

depling
↓ 2011
Proceedings

edited by Kim Gerdes, Eva Hajičová, and Leo Wanner

depling.org

ISBN 978-84-615-1834-0, Barcelona 2011

**International Conference on
Dependency Linguistics**

Depling 2011, Barcelona, September 5-7 2011

exploring dependency grammar, semantics, and the lexicon

Index

	Kim Gerdes (Sorbonne Nouvelle), Eva Hajičová (Charles University), and Leo Wanner (Universitat Pompeu Fabra and ICREA)	<i>Introduction</i>	iii
Theoretical foundations	Igor Mel'čuk (University of Montreal)	<i>Dependency in Language</i>	1
	Kim Gerdes (Sorbonne Nouvelle) and Sylvain Kahane (Université Paris Ouest)	<i>Defining dependencies (and constituents)</i>	17
	Nicolas Mazziotta (Universität Stuttgart)	<i>Coordination of verbal dependents in Old French: coordination as a specified juxtaposition or apposition</i>	28
	Timothy Osborne	<i>Type 2 Rising: A Contribution to a DG Account of Discontinuities</i>	38
	Thomas Groß (Aichi University)	<i>Clitics in Dependency Morphology</i>	47
	Thomas Groß (Aichi University)	<i>Catenaes in Morphology</i>	58
	Federico Gobbo and Marco Benini (University of Insubria)	<i>From Structural Syntax to Constructive Adpositional Grammars</i>	69
Semantics	Ke Wang and Rongpei Wang (Dalian University of Technology)	<i>Implementing Categorical Grammar in Semantic Analysis: from Frame Semantics' View</i>	79
	Orsolya Vincze and Margarita Alonso Ramos (Universidade da Coruña)	<i>A proposal for the multilevel linguistic representation of Spanish person names</i>	85
	Michael Hahn and Detmar Meurers (Universität Tübingen)	<i>On deriving semantic representations from dependencies: A practical approach for evaluating meaning in learner corpora</i>	94
Tree banks	Alicia Burga (Universitat Pompeu Fabra), Simon Mille (Universitat Pompeu Fabra) and Leo Wanner (Universitat Pompeu Fabra and ICREA)	<i>Looking Behind the Scenes of Syntactic Dependency Corpus Annotation: Towards a Motivated Annotation Schema of Surface-Syntax in Spanish</i>	104
	Katri Haverinen, Filip Ginter, Veronika Laippala, Samuel Kohonen, Timo Viljanen, Jenna Nyblom, and Tapio Salakoski (University of Turku)	<i>A Dependency-based Analysis of Treebank Annotation Errors</i>	115
	Henrik Høeg Müller (Copenhagen Business School)	<i>The Copenhagen Dependency Treebank (CDT). Extending syntactic annotation to morphology and semantics</i>	125
	Markus Dickinson and Marwa Ragheb (Indiana University)	<i>Dependency Annotation of Coordination for Learner Language</i>	135
	Eva Maria Duran Eppler (Roehampton University)	<i>The Dependency Distance Hypothesis for bilingual code-switching</i>	145
	Kristiina Muhonen and Tanja Purtonen (University of Helsinki)	<i>Creating a Dependency Syntactic Treebank: Towards Intuitive Language Modeling</i>	155
	Xinying Chen, Chunshan Xu, and Wenwen Li (Communication University of China)	<i>Extracting Valency Patterns of Word Classes from Syntactic Complex Networks</i>	165



Linguistic issues	Jarmila Panevová and Magda Ševčíková (Charles University)	<i>Delimitation of information between grammatical rules and lexicon</i>	173
	Katerina Rysová (Charles University)	<i>The Unmarked Word Order of Inner Participants in Czech, With the Focus on the Systemic Ordering of Actor and Patient</i>	183
	Dina El Kassas (Minya University)	<i>Representation of Zero and Dummy Subject Pronouns within multi-strata dependency framework</i>	193
	Taiki Yoshimura (Osaka University)	<i>The 'Errant' Scope of Question in Turkish: A Word Grammar Account</i>	204
	Andreas Pankau (Goethe Universität Frankfurt and Universiteit Utrecht)	<i>Wh-Copying in German as Replacement</i>	214
	Kensei Sugayama (Kyoto Prefectural University)	<i>Why kono akai hana and akai kono hana Are Both Possible in Japanese: A Word Grammar Account</i>	224
Pavlna Jinová, Lucie Mladová, and Jiří Mírovský (Charles University)	<i>Sentence Structure and Discourse Structure: Possible Parallels</i>	233	
Formal issues	Vered Silber-Varod (The Open University of Israel and Afeka Tel Aviv Academic College of Engineering)	<i>Dependencies over prosodic boundary tones in Spontaneous Spoken Hebrew</i>	241
	Bernd Bohnet (Universität Stuttgart), Leo Wanner (Universität Pompeu Fabra and ICREA), and Simon Mille (Universität Pompeu Fabra)	<i>Statistical Language Generation from Semantic Structures</i>	251
	Alexander Dikovsky (Université de Nantes)	<i>Categorial Dependency Grammars: from Theory to Large Scale Grammars</i>	262
	Ramadan Alfareed, Denis Béchet, and Alexander Dikovsky (Université de Nantes)	<i>"CDG LAB": a Toolbox for Dependency Grammars and Dependency Treebanks Development</i>	272
Parsing	Bernd Bohnet (Universität Stuttgart)	<i>Comparing Advanced Graph-based and Transition-based Dependency Parsers</i>	282
	Niels Beuck, Arne Köhn and Wolfgang Menzel (Universität Hamburg)	<i>Incremental Parsing and the Evaluation of Partial Dependency Analyses</i>	290
	Ozlem Cetinoglu, Anton Bryl, Jennifer Foster and Josef Van Genabith (Dublin City University, Ireland)	<i>Improving Dependency Label Accuracy using Statistical Post-editing: A Cross-Framework Study</i>	300
	Julia Krivanek and Walt Detmar Meurers (Universität Tübingen)	<i>Comparing Rule-Based and Data-Driven Dependency Parsing of Learner Language</i>	310
	Igor Boguslavsky, Leonid Iomdin, Leonid Tsinman, Victor Sizov, and Vadim Petrochenkov (Russian Academy of Sciences)	<i>Rule-Based Dependency Parser Refined by Empirical and Corpus Statistics</i>	318

Introduction

Ibn Mada was the first grammarian ever to use the term *dependency* in the grammatical sense that we use it today. He was born in 1119 in Cordoba, studied in Sevilla and Ceuta, and died 1195 in Sevilla. He is known for his only book, *Radd: the refutation of grammarians*, in which he tackles subjects that still seem modern today: He criticizes the use of ellipsis, i.e. underlying invisible forms, in the analyses of grammatical phenomena, for example when discussing whether the unmarked nominative form consists of a zero marker. He further speaks out against semantic interpretation and justification of grammatical rules, and thus in favor of an independence of syntax, semantics, and cognitive interpretation: “*And why is the agent in the nominative?*” *The correct answer is [...] : “This is how the Arabs speak”*.

And he used the term *تعلق* *Ta'alluq* which translates as *being suspended to, dependent upon, connected with; attachment; love of the world; dependence ; connection; relation; relationship ; concern, reference, regard ; consideration, reflection ; commerce ; means of support, employment, office ; property, possession ; a manor ; a small division of a district*, when referring to the relation between verbs and their direct and indirect *dependents*. He prefers this term to *عمل* ‘*amal* ('operation', 'government'), the commonly used term at his time for relations between *governing* and *dependent* words, because, following Ibn Mada, the *head* word does not *operate* on its dependents, but he only sees a *relation*, a *dependency*. He goes as far as calling it *heretic* to use *amal* because words cannot act on other words and *cause* an inflection. Since this was merely a change in vocabulary, the use of *dependency* did not catch on until the 20th century. The importance of a dependency type analysis for the description of language, however, was well-established in the Arabic grammatical tradition before Ibn Mada and can even partially be traced back to Panini.

So why, you might ask, do we need a conference on Dependency Linguistics in Barcelona, when grammarians have done dependency linguistics in Spain for 1000 years?

The generative grammatical tradition that, in its origins, solely attempts to construct a system that distinguishes grammatical from ungrammatical sentences, left linguistics in a state where the grammatical analysis, phrase structure, was difficult to connect to deeper (semantic, conceptual) structures. The result was a complete separation between, on one side, Natural Language Processing that needed deeper analyses, for translation, classification, generation etc. and, on the other side, generative linguistics that built complex structures with the declared goal to model Language as a whole, where the structures got more and more complicated the further the described language is from English. In the second half of the 20th century, only a few linguists, often referring themselves to Lucien Tesnière, continued to describe language in terms of dependency, mainly because they were working on free word order languages, where the use of phrase structure is more clearly maladaptive.

Since the 1990s, NLP is turning towards dependency analysis, and in the past five years dependency has become quasi-hegemonic: The very large majority of parsers presented in recent NLP conferences are explicitly dependency-based. It seems, however, that the connection between computational linguists and dependency linguists remains sporadic: What happens commonly is that someone transfers an existing tree bank into a dependency format that fits his or her needs, and other researchers attempt to reproduce this annotation, with statistical or rule-based grammars. Not that the situation was any better when parsers still automatized phrase structure construction and linguistics discussed *move alpha*. Yet, we believe that the situation is different today and dependency linguists and computational linguists have a lot to share:

We know that statistical parsers give better results if we have a linguistically coherent corpus analysis. We need to know what the differences are between surface and deep dependency. How to define dependency? What are the units that appear in dependency analysis? What set of labels (particularly syntactic functions) do we use? Do we agree on the same syntactic representations? Or simply, what are the others doing? What kind of analysis works for which application? How to link dependency to structures to the lexicon and to semantics?



Not all questions will find a direct answer, but we believe that Dependency Linguistics 2011 provides a forum allowing for an interchange between researchers on the theoretical and the applicative sides of current linguistics, on various languages, and on various underlying models.

The conference is organized in thematic sessions:

We will start with the *theoretical foundations* of dependency: What types of dependency exist, how to define dependency, how to handle coordination and discontinuities, how to relate dependency and morphology, as well as, more specifically, how to handle clitics, and finally how to translate Tesnièrean notions into a grammar formalism.

In the *semantics* session, we learn about the relations of dependency structures to Frame Semantics, about the semantic analysis of person names and about semantic structures on learner corpora.

A big part of the work presented at this conference concerns *treebanks*: A syntactic annotation scheme for Spanish, error analysis for Finnish, a multi-layer corpus annotated in terms of the Generative Lexicon, the analysis of coordination on a learner corpus of English, the detection of code switching in an English-German by means of the dependency distance, user-centered syntactic annotation for Finnish, and the extraction of valency patterns from a Chinese treebank.

Linguistic issues include the relationship of grammar and lexicon, the definition of unmarked word order in Czech, the Prodrop problem in Arabic, the interrogative clitic of Turkish, wh-copying in German, free word order in Japanese noun phrases, and parallels between syntax and discourse.

The session on *formal topics* presents the prosody syntax interface in an analysis of Hebrew, statistical language generation, and categorical dependency grammars, as well as tools for their development.

Last but not least, *dependency parsing* will be presented under its various aspects: A comparison of graph-based and transition-based parsers, incremental parsing, improving dependency label accuracy, a comparison of rule-based and data-driven parsers, and a rule-based dependency parser for Russian.

Overall, these proceedings include 33 articles from 16 countries: Canada, China, Czech Republic, Denmark, Egypt, Finland, France, Germany, Great Britain, Ireland, Israel, Italy, Japan, Russia, Spain, and the United States.

We would like to thank Igor Mel'čuk and Joakim Nivre for having accepted our invitation to give talks on the two fundamental sides of our conference: dependency analysis and dependency parsing. We would also like to thank the program committee for their participation and their astonishingly thorough reviews that have certainly contributed to the quality of the papers presented here:

Margarita Alonso Ramos University of La Coruña

Lorraine Baqué Autonomous University of Barcelona

David Beck University of Alberta, Edmonton

Xavier Blanco Autonomous University of Barcelona

Bernd Bohnet Stuttgart University

Igor Boguslavsky Polytechnical University of Madrid

Marie Candito University Paris 7

Éric de la Clergerie University Paris 7

Michael Collins Columbia University, New York

Benoit Crabbé University Paris 7

Denys Duchier University of Orléans

Jason Eisner Johns Hopkins University, Baltimore

Dina El Kassas Miniya University

Gülşen Cebiroğlu Eryiğit Istanbul Technical University

Charles J. Fillmore University of California, Berkeley

Koldo Gojenola University of the Basque Country, Bilbao

Jan Hajič Charles University in Prague



Hans-Jürgen Heringer	University of Augsburg
Richard Hudson	University College London
Leonid Iomdin	Russian Academy of Sciences, Moscow
Lidija Iordanskaja	University of Montreal
Aravind Joshi	University of Pennsylvania, Philadelphia
Sylvain Kahane	University Paris Ouest
Marco Kuhlmann	Uppsala University
François Lareau	Macquarie University, Sydney
Alessandro Lenci	University of Pisa
Leonardo Lesmo	University of Turin
Haitao Liu	Zhejiang University, Hangzhou
Henning Lobin	University of Gießen
Chris Manning	Stanford University
Igor Mel'čuk	University of Montreal
Wolfgang Menzel	University of Hamburg
Kemal Oflazer	Carnegie Mellon University, Qatar
Ryan McDonald	Google Research, New York
Piet Mertens	University of Leuven
Jasmina Milićević	Dalhousie University, Halifax
Dipti Misra Sharma	IIT, Hyderabad
Henrik Høeg Muller	Copenhagen Business School
Jee-Sun Nam	Hankuk University of Foreign Studies, Seoul
Alexis Nasr	University of Marseille
Joakim Nivre	Uppsala University
Gertjan van Noord	University of Groningen
Martha Palmer	University of Colorado, Boulder
Jarmila Panevova	Charles University in Prague
Alain Polguère	Nancy University
Prokopis Prokopidis	ILSP, Athens
Owen Rambow	Columbia University, New York
Ines Rehbein	Saarland University, Saarbrücken
Petr Sgall	Charles University in Prague
Davy Temperley	University of Rochester
Robert Van Valin	Heinrich Heine University, Düsseldorf

Many thanks also to Joana Clotet and Bea Abad for taking care of practically all matters related to the local organization of the conference, to Simon Mille for assisting them and to all the other members of the local organization team: Stefan Bott, Alicia Burga, Gerard Casamayor, Gaby Ferraro, Estela Mosquera, Luz Rello, Orsi Vincze, and Alexandra Vorobyova. Financial support for Depling was provided by the Natural Language Processing research group TALN of the Pompeu Fabra University (UPF), the Department of Communication and Information Technologies, UPF, and the Department of French and Romance Philology at the Autonomous University of Barcelona.

Kim Gerdes, Eva Hajičová, and Leo Wanner
September 2011



Dependency in Language-2011

Igor Mel'čuk

OLST

Université de Montréal, Montréal

igor.melcuk@umontreal.ca

Abstract

The paper aims at summarizing knowledge about linguistic dependency. Three types of dependency are considered: semantic, syntactic, and morphological; fourteen possible combinations thereof are presented. Each type of dependency is described in some detail. An overview of Deep-Syntactic relations is given, as well as the criteria for establishing Surface-Syntactic relations in particular languages. Some domains in which the advantages of dependencies manifest themselves in the clearest way are briefly sketched (diathesis and voice, lexical functions, paraphrasing, word order). The place of the notion of phrase within a dependency framework is characterized; an analysis of a “bracketing paradox” in terms of linguistic dependency is proposed.

1 Introductory Remarks

1.1 The Task Stated

This talk does not present new facts or new ideas about known facts. Its goal is to sum up my own experience of more than half a century of work on linguistic dependencies and to better organize the knowledge acquired over this period. It is based on materials that have been published (Mel'čuk 1963, 1974, 1979, 2002, 2003 and 2009) and that are easily accessible. Therefore, I will not explain the nature of linguistic dependency; I will also abstain from rigorously presenting the necessary notions and formalisms of Meaning-Text theory (the reader is kindly invited to consult the appropriate titles: e.g., Mel'čuk 1974: 31ff, 1981, 1988: 43-101, 1997, 2006: 4-11 and Kahane 2003). Finally, there will be only a dire minimum of references.

The task of this talk is three-pronged:

- To present an overview of what must be known about linguistic dependencies to successfully use them (“Dependencies 101”).
- To emphasize the advantages of dependencies (with respect to constituents) in linguistic description.
- To sketch the place of phrases (\approx constituents), within a strict dependency approach.

But first, a bit of personal experience.

1.2 Some History

I met (syntactic) dependency for the first time in the 1950's while developing a Hungarian-Russian machine-translation system: Mel'čuk 1957. Here is an example from this paper: translation of the Hungarian sentence (1a) into Russian.

(1) a. *A legtöbb nyelvnek sok idegen eredetű szava van.*
the most language-SG.DAT many foreign “originary” word-SG.NOM.3SG is

b. *V bol'šinstve jazykov est' mnogo slov inostrannogo proisxoždenija.*
in majority-SG.PR language-PL.GEN is many word-PL.GENforeign-N.SG.GEN origin-SG.GEN

At least four problems have to be dealt with by an automatic translation system to obtain (1b) from (1a):

- The grammatical number of the nouns ‘language’ and ‘word’: singular in Hungarian, because of a quantifier (which requires the singular of the quantified N in Hungarian), and plural in Russian—for the same reason, except that Russian quantifiers require the plural of nouns.
- The agreement of the adjective ‘foreign’ with the noun ‘origin’ in Russian (in Hungarian, adjectives do not agree with nouns).
- The dative of ‘language’ in Hungarian, induced by the verb VAN ‘[there] is’, corresponds to the Russian preposition V ‘in’, induced by the verb EST’ ‘[there] is’.
- Word order: some Hungarian modifiers precede the lexemes they modify while their Russian equivalents follow theirs; cf.:

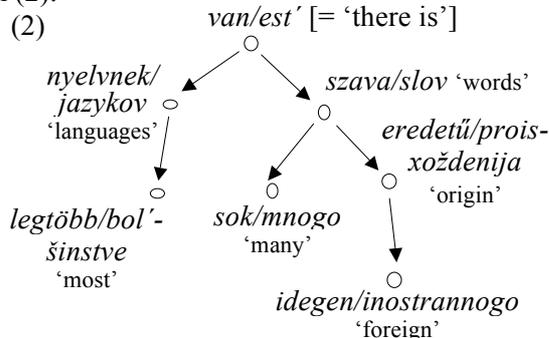
Hung. *szava van* \Leftrightarrow Rus. *est' ... slov*

Hung. *eredetű szava* \Leftrightarrow Rus. *slov ... proisxoždenija.*

However, I was unable back then, and I am still unable now, to figure out how to formulate the corresponding rules if the sentence is simply parsed into constituents, that is, supplied only with a “pure” phrase structure. The constituency approach, borrowed by computational linguists in the ex-USSR from the USA and UK, was then the only well-known formal framework, yet I felt strongly that there was no way you could translate by means of bracketed phrases. And from my fu-

tile attempts to find a way to do so syntactic dependencies were born.¹

The above problems can be easily solved by using syntactic dependencies. Let us consider an approximate dependency tree for both sentences in (2):



Based on dependency arrows linking the lexemes, it is easy to formulate the rules for the necessary changes between Hungarian and Russian in the four above cases. It became soon clear that automatic translation needs—as a kind of hinge between sentences of two languages—a syntactic structure, and this structure must be written in terms of dependencies.

1.3 Dependency and Meaning-Text Approach

To see all advantages of dependency representation, one has to use it in a package with several other techniques. Three conditions must be met for dependencies to show their full power:

- A semantic representation as a starting point — that is, the very first thing to do in any linguistic study is to present a formal description of the meaning of the expressions examined (in order to establish the correspondences between the expression a given meaning and its possible expression). The guiding slogan here is: “We say what we think!”
- A synthetic perspective—that is, a linguistic description is done from meaning to text. You aim at modeling the activity of the Speaker, who produces texts, rather than that of the Addressee, who interprets/understands them. The guiding slogan: “To use a language is to speak it!”

¹ Of course I was not alone: at least in Germany, France and Czechoslovakia, several researchers were inching forward along the same difficult path, and for the same reasons, as myself. Interestingly, in the USA, David Hays and Julia Robinson formulated explicitly the basic tenets of dependency syntactic description as far back as 1960 and published their proposals, but theirs remained voices crying out in the desert...

- A stratificational description—that is, each type of major linguistic unit (such as sentences and words) is represented in terms of those properties that are specific to it, so that we need different formalisms for each type. Several levels of linguistic representation and different structures within the representation of a given level are distinguished; these representations and structures are related by means of formal rules of the linguistic model. The guiding slogan: “Dead flies and meatballs should be served separately!”²

1.4 Simplifications Used in This Talk

Concerning the characterization of a Meaning-Text model, two simplifications are resorted to:

- 1) While the bottom level is the Semantic representation [= SemR], the upper level in all the examples below is the Deep-Morphological representation [= DMorphR]. This means that the discussion of morphology will be completely left out, one of the reasons being that many languages (like Vietnamese or Mandarin Chinese) have no or very little morphology.

- 2) Instead of full linguistic representations, the paper deals only with their central structures. For instance, instead of the complete SemR of a sentence (which includes the Semantic Structure, the Sem-Communicative Structure, the Rhetorical Structure and the Referential Structure), only its central structure—i.e., the Semantic structure [= SemS]—will be considered.

Concerning the proposed definitions of linguistic phenomena, only prototypical cases are considered. This means that several definitions and characterizations given below are incomplete—that is, strictly speaking, incorrect. However, they are sufficient for my purposes here.

2 Different Types of Linguistic Dependency

Let us take a simple sentence:

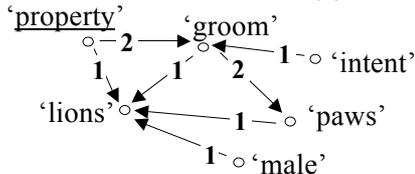
- (3) *Male lions carefully groom their paws.*

² This is a punch line of an old Jewish joke. A poor guy comes to a shabby diner, a typical greasy spoon, and asks for a helping of meatballs. When the dish arrives, he sees several dead flies on the meatballs; calling up the waiter, he indicates the problem to the latter. The waiter explodes in self-assured indignation: —Screw off! If you don’t like our meatballs, go some-where else!!—and starts rolling up the sleeves, getting ready for a physical assault. —No, no, you misunderstood me,—screams the customer. —I have nothing against your meatballs, but I would like to have my dead flies and my meatballs separately.

The task of a linguist is to write a system of rules that, applied—among other things—to a formal representation of the meaning of this sentence, or to its SemR, produce the representation of its physical form, or its Phonic representation [= PhonR]. A system of rules such that it is valid for a language as a whole, or a linguistic model, is a correspondence $\{\text{SemR}_i\} \Leftrightarrow \{\text{PhonR}_j\}$; however, as indicated, I will stop at the DMorphR.

Sentence (3) has the SemS in (4a) and the DMorphS (4b):

(4) a. The SemS of sentence (3)



Literal reading of the SemS in (4a):

‘Male lions have the property of intently grooming their paws’

The SemS of (4a) in predicate-argument notation:

Property(Male(lions_i);
Careful(lions_i; Groom(Paws(lions_i))))

b. The DMorphS of sentence (3)

MALE < LION_{PL} < CAREFULLY <
GROOM_{IND, PRES, 3, PL} < THEIR < PAW_{PL}
(The symbol “<” means ‘immediately precedes’.)

This example illustrates three types of dependency:

—The SemS in (4a) is written in terms of semantic dependency (see 4).

—In order to go from (4a) to (4b), the Deep-Syntactic structure [= DSyntS] and the Surface-Syntactic structure [= SSyntS] are needed; both are based on syntactic dependency (see 5.4).

—The rules for the “SSyntS \Leftrightarrow DMorphS” transition use morphological dependency (see 6); the MorphS itself does not show them.

Dependency is a binary relation that is anti-reflexive, anti-symmetrical and non-transitive; it will be figured by an arrow:

Governor $\circ \longrightarrow \circ$ Dependent

Semantic Dependency [= Sem-D]

If the SemS is written in a formal language derived from the language of predicate calculus,³ semantic elements in it, or **semantemes** (= signified of lexemes), are linked by a dependency relation.

³ I don’t think there is or can be another formal language fit for describing linguistic meaning. At least, all projects of ‘semantic metalanguages’ I have seen propose something fully equivalent to the language of predicate calculus.

This is **semantic dependency**, corresponding to a “predicate ~ argument” relation; the predicate is the Sem-Governor of its arguments. Since predicative semantemes have been found in various languages with up to six arguments, six relations of Sem-D are distinguished: 1, 2, ..., 6. (These distinguishers are asemantic: see 4.)

Syntactic Dependency [= Synt-D]

As can be seen from (4), in Meaning-Text approach, the SemS of a sentence is a **network**, and the MorphS, a **chain**. The SyntS as a convenient bridge between the SemS and the MorphS must be a **dependency tree**. Synt-Ds link lexemes that label the nodes of the SyntS; these links do two things:

1) Synt-D between the elements of a (syntactic) phrase determines the distribution of the phrase within sentences—that is, its capacity to be used in a particular syntactic position. Thus, in the phrase L_1 -synt- L_2 , the Governor is L_1 , if and only if L_1 -synt- L_2 is used like L_1 (\approx can replace L_1) rather than like L_2 .

2) Synt-D controls the linear position of the Synt-dependent with respect to its Synt-governor. Thus, for instance, in **English**, in Basque and in French we have Adj \leftarrow synt-N (the Adj \leftarrow synt-N phrase is used like an N and not like an Adj), and Adj is positioned with respect to N (in English, before N; in Basque, after N; and in French, before or after N, according to several conditions).

Morphological Dependency [= Morph-D]

Sem-D and Synt-D are cross-linguistically universal in the following two senses:

—there is no language without Sem-D and Synt-D;

—in a language, there is no sentence without Sem-D and Synt-D, which link all the words of the sentence.

But Morph-D is found only in some languages—those that feature at least one of two types of Morph-D: **agreement** and **government**; and even in a language with morphology, not all words in any sentence are morphologically linked. Thus, in (3), the verb GROOM agrees with the subject LION_{PL}, and this is the only morphological link in this sentence.

Sem-D holds between semantemes, which are signified of lexemes:

‘ L_1 -sem \rightarrow L_2 ’ means ‘ $L_1(L_2)$ ’,

that is, semanteme ‘ L_2 ’ is a semantic argument of predicative semanteme ‘ L_1 ’.

Synt-D holds between lexemes: L_1 -synt \rightarrow L_2 means that it is L_1 that determines the distribution

(i.e., the passive valence) of the phrase L_1 -**synt**- L_2 within sentences. At the same time, L_2 's linear position in the sentence is determined with respect to L_1 : L_2 precedes L_1 , follows it, or can precede or follow (as a function of some particular conditions).

Morph-**D** holds between grammemes and syntactic features of lexemes: L_1 -**morph**- L_2 means that a grammeme or a syntactic feature of L_1 determines some grammemes of L_2 .

Sem-**D**s and Synt-**D**s form connected structures (within sentences); they are directly reflected in sentence representations—as semantic networks and syntactic trees. Morph-**D**s do not form a connected structure (within a sentence); they are not explicitly shown in any sentence representations,

but are used only in syntactic rules that ensure the morphologization of the SSynt-structure.

These three types of dependency do not exhaust all linguistic dependencies: for instance, there is communicative dependence, which will be ignored here.

3 Fourteen Combinations of the Three Types of Linguistic Dependency

The mutual logical autonomy of the three types of dependency is demonstrated by the fact that they cooccur: two lexemes L_1 and L_2 in a sentence can be linked by any combination of dependencies out of the 14 logically possible ones. Here is an overview of these possibilities, with minimal examples.

1. L_1 L_2 : No dependency whatsoever between L_1 and L_2 ; e.g., HERE and POSSIBILITY_{PL} in the preceding sentence.
2. L_1 —**sem**→ L_2 : Only Sem-**D** between L_1 and L_2 ; e.g., JOHN and LAUGH in *John broke out laughing*.
3. L_1 —**synt**→ L_2 : Only Synt-**D** between L_1 and L_2 ; e.g., TAKUSAN ‘many/much’ and YOMU ‘read’ in Jap. *Yoko+wa hon+o takusan yom+u* lit. ‘Yoko_{THEME} book_{ACC} many read_{PRES}’ = ‘Yoko reads many books’; semantically, ‘takusan’ bears on ‘hon’, and morphologically, *takusan* is an invariable adverb.
4. L_1 —**morph**→ L_2 : Only Morph-**D** between L_1 and L_2 ; e.g., IČ ‘our’ and HEBGNU-(*jič*) ‘ran.away.our’ in Tabasaran *Ič mudur uc^{wh}u+na hebgnu+jič* lit. ‘Our goat.kid you.to ran.away.our’ = ‘Our goat kid ran away to you’, where HEBGNU depends morphologically on the pronoun IČ ‘our’, without any Sem- or Synt-link with it.
5. L_1 L_2 :
—**sem**→
—**synt**→ Sem-**D** and Synt-**D** between L_1 and L_2 go in the same direction, no Morph-**D**; e.g., READ and NEWSPAPER in *John is reading a newspaper*.
6. L_1 L_2 :
—**sem**→
←**synt**— Sem-**D** and Synt-**D** between L_1 and L_2 go in opposite directions, no Morph-**D**; e.g., INTERESTING and NEWSPAPER in *an interesting newspaper*, where NEWSPAPER semantically depends on INTERESTING, since the former is a Sem-argument of the latter.
7. L_1 L_2 :
—**sem**→
—**morph**→ Sem-**D** and Morph-**D** between L_1 and L_2 go in the same direction, no Synt-**D**; e.g., the clitic *le*_{DAT} ‘to.him/to.her’ in Sp. *Juan le quiere dar un libro* ‘Juan wants to give him a book’ depends semantically and morphologically on the verb DAR, while syntactically it depends on the Main Verb QUERER ‘want’, since it forms a phrase with it (for the notion of phrase, see 5.3) and is positioned with respect to it.
8. L_1 L_2 :
—**sem**→
←**morph**— Sem-**D** and Morph-**D** between L_1 and L_2 go in opposite directions, no Synt-**D**; e.g., MARIE and BELLE ‘beautiful’ in Fr. *Marie est devenue belle* ‘Mary has become beautiful’: MARIE depends semantically on BELLE, being its argument, but BELLE depends morphologically—for its number and gender—on MARIE.
9. L_1 L_2 :
—**synt**→
—**morph**→ Synt-**D** and Morph-**D** between L_1 and L_2 go in the same direction, no Sem-**D**; e.g., AB ‘from’ and URBS ‘city’ in Lat. *ab urbe condita* lit. ‘from city founded’ = ‘from the founding of the City [= of Rome]’.
10. L_1 L_2 :
—**synt**→
←**morph**— Synt-**D** and Morph-**D** between L_1 and L_2 go in opposite directions, no Sem-**D**; e.g., TEMPERATURE and BEGIN in *The temperature begins to fall*: syntactically, TEMPERATURE depends on BEGIN, but morphologically, the other way around.
11. L_1 L_2 :
—**sem**→
—**synt**→
—**morph**→ Sem-**D**, Synt-**D** and Morph-**D** between L_1 and L_2 go all in the same direction; e.g., *vižu* ‘I.see’ and *Maš+u_{ACC}* ‘Mary’ in Rus. *Vižu Mašu* ‘I see Mary’.

12. $L_1 \begin{array}{l} \text{---sem---} \\ \text{---synt---} \\ \text{---morph---} \end{array} L_2$: Sem-**D** and Synt-**D** between L_1 and L_2 go in the same direction, Morph-**D** is opposite; e.g., polypersonal agreement of the Main Verb in a case-less language, as in Abkhaz *Nadš'a sara i+s+əl+teixxt' aš'q'w'ə* lit. 'Nadsha me gave a book', where the Main Verb *isəlteit'* agrees, by its prefixes, with all three invariable actants (in person and gender); semantically and syntactically, actants depend on the verb, which depends on them morphologically (on each of them, in different categories).
13. $L_1 \begin{array}{l} \text{---sem---} \\ \text{---synt---} \\ \text{---morph---} \end{array} L_2$: Sem-**D** and Morph-**D** between L_1 and L_2 go in the same direction, Synt-**D** is opposite; e.g., the idafa construction in Iranian languages: Persian *ketab+e nav* 'book-IDAFA new', where KETAB 'book' is a semantic argument of NAV 'new' and receives from it the morphological marker -e, while syntactically being its governor.
14. $L_1 \begin{array}{l} \text{---sem---} \\ \text{---synt---} \\ \text{---morph---} \end{array} L_2$: Synt-**D** and Morph-**D** between L_1 and L_2 go in the same direction, Sem-**D** is opposite; e.g., NOUVELLE 'piece.of.news' and INTÉRESSANT 'interesting' in Fr. *nouvelle*_{(fem)SG} *intéressant*+e_{SG.FEM} 'interesting piece of news'.

4 Semantic Dependency

Speaking of Sem-**D**, one has to insist that there are no “meaningfully” distinguished Sem-relations that would correspond to Fillmore’s Deep Cases or “Semantic Roles” (= “θ-roles”) of Generative Grammar. It is linguistically and logically inconsistent to explicitly indicate in a SemS that in *John loves Mary*, ‘John’ is related to ‘love’ as Experiencer, and ‘Mary’, as Source/Object. “Experiencer” is actually a binary predicate ‘X is Experiencer of Y’ = ‘X experiences Y’, and as such, it would require a meaningful indication of the Sem-relations between itself and its arguments, which will in turn require the same thing, etc. This creates infinite regression, and it can be stopped only by an arbitrary decision about which Sem-relations and under which conditions must be considered non-predicates—or, at least, not quite normal predicates. However, postulating some Sem-relations that are not full-fledged predicates is a *contradictio in adjecto*. Moreover, any such “not quite normal” predicate is also capable of appearing as quite a normal predicate, when it is associated with a node, and not with an arc, of a semantic network. The bottom line is that Sem-**D**s are simply distinguished (by arbitrary symbols, e.g., by numbers), but they cannot be positively identified. The semantic role of an argument is given by the semantic decomposition of the predicate:

‘John←1-loves-2→Mary’ =
‘John←1-experiences strong affection [for] and sexual attraction-[to]-2→Mary’.

NB: However, the names of “semantic relations” can be used informally—for better clarity, as a kind of abbreviation. Thus, L_1 can be called Experiencer with respect to L_2 to mean that ‘ L_1 ’ is the SemA 1 of the predicate ‘experience’ in the semantic decomposition of ‘ L_2 ’; etc.

5 Syntactic Dependency

5.1 Deep- vs. Surface-Synt-Dependency

Speaking of Synt-**D**, one has to emphasize the distinction of two sublevels of linguistic representation in syntax: Deep-Syntactic vs. Surface-Syntactic representation, *resp.* structure [= DSyntR vs. SSyntR]. While DSyntR is cross-linguistically universal, SSyntR is language-specific. The DSynt- vs. SSynt-distinction allows for useful generalizations in syntax and for the formulation of simpler and more efficient semantic rules, i.e., rules of the {SemR} ⇔ {DSyntR} transition. For instance, in English, the verb HELP takes a DirO (*help-[the]-dir-objectiveal→neighbor*), and its Russian equivalent POMOGAT’ an IndirO (in the dative: *pomogat’-indir-object→sosed+u*): two different syntactic constructions; but at the DSynt-level, where surface particularities are not taken into account, the two constructions are “homogenized:”

HELP-**II**→NEIGHBOR and POMOGAT’-**II**→SOSED

The DSynt- vs. SSynt-distinction requires establishing two sets of syntactic relations: Deep-Syntactic vs. Surface-Syntactic relations.

5.2 Deep-Synt-Relations

The DSyntRels are supposed to be language-independent; all the DSyntRels are necessary and the set thereof is sufficient:

Necessity: Each DSyntRel is found in many, if not all, languages.

Sufficiency: The DSyntS of any sentence of any language can be conveniently represented in terms of the DSyntRels available.

The last statement is true only if we allow for the use, in the DSyntS, of fictitious lexemes, called upon to represent lexical-type meanings expressed by syntactic constructions.

Each DSyntRel stands for a family of particular syntactic constructions found in particular language-

ges; the DSyntRel is intended to represent them in a more abstract way. DSyntRels are semantic-gearred generalizations over specific SSyntRels of various languages; at the DSynt-level, only most general Synt-**D**s are distinguished. Thus, as shown above, the **direct-objective** construction, the **indirect-**

objective construction and the prepositional **oblique-objective** construction governed by different verbs are all reduced to DSyntRel **II**.

The full inventory of DSyntRels is represented in Fig. 1:

coordinate DSyntRels		subordinate DSyntRels									
		weak subordinate DSyntRel	strong subordinate DSyntRels								
			modification: attributive DSyntRels			complementation: actantial DSyntRels					
COORD 1	QUASI-COORD 2	APPEND 3	ATTR 4	ATTR_{descr} 5	I 6	II 7	III 8	IV 9	V 10	VI 11	II_{dir.sp} 12

Figure 1: Inventory of DSynt-relations

The set of DSyntRels is determined by the following five binary DSynt-oppositions:

1. Coordination vs. Subordination: constructions which represent lists (of lexical expressions) ~ constructions which represent texts other than lists. The first class—coordinate constructions—manifest two DSyntRels, called **COORD**(inative) [*Mary, -COORD→Peter, -COORD→Alan; New York -COORD→or Boston*] and **QUASI-COORD** [*in Boston -QUASI-COORD→on Fleet Street -QUASI-COORD→at her parents*’]; the DSyntRels of the second class of constructions are subordinate.

2. Weak Subordination vs. Strong Subordination: constructions with no strong structural links ~ constructions with strong structural links. The first class—weak subordinate constructions—is represented by the DSyntRel **APPEND**(itive) [*John is, -APPEND→unfortunately, absent*].

3. Modification vs. Complementation: modification-based constructions ~ complementation-based constructions. Modification is a Synt-**D** L_1 -synt $\rightarrow L_2$ such that ‘ $L_1 \leftarrow \text{sem} - L_2$ ’; complementation is a Synt-**D** L_1 -synt $\rightarrow L_2$ such that ‘ $L_1 - \text{sem} \rightarrow L_2$ ’. The DSyntRels of the first class are **ATTR**(ibutive) [*Alan works -ATTR→hard*]; the DSyntRels of the second class are actantial.

4. Restrictive Modification vs. Descriptive Modification: constructions with restrictive modification ~ constructions with descriptive modification. The first class—restrictive, or identifying, modification—is represented by the DSyntRel **ATTR** (which by default is understood as restrictive): *He reads only interesting <Spanish> books*; the second class—descriptive, or qualifying, modification—is represented by the DSyntRel **ATTR_{descr}**: *These three students, who just returned from Europe, were selected to represent the department*.

5. Different Actantial Roles: I, II, ..., VI, II_{dir.sp}. Constructions with actantial DSyntRels are divided into seven classes, according to the maximal number of DSyntAs that a lexical unit in natural language can have, which is six, plus a special DSyntRel for Direct Speech:

‘WOW!’ \leftarrow **II_{dir.sp}** - SAY_{PAST} - **I** \rightarrow ALAN \Leftrightarrow
‘Wow!’, said Alan.

5.3 Surface-Synt-Relations: Criteria for Establishing Surface-Syntactic Relations in a Language

Given the abstract nature of Synt-**D** (this dependency is not directly perceivable by our mind or senses), three groups of formal criteria are needed for establishing inventories of SSynt-relations for particular languages: **A**. A criterion for SSynt-connectedness between two lexemes L_1 and L_2 in a sentence (= for the presence of a SSyntRel between them); **B**. Criteria for the SSynt-dominance between L_1 and L_2 (= for the orientation of the SSyntRel between them); **C**. Criteria for the specific type of the given SSyntRel between L_1 and L_2 .

SSyntRels hold between lexemes in a SSyntS; however, for simplicity’s sake, I will allow myself to use in the examples actual wordforms, where this does create confusion.

SSynt-Connectedness: Criterion A

Criterion A (prosody and linear order): Potential prosodic unity and linear arrangement

In a sentence, the lexemes L_1 and L_2 have a direct Synt-**D** link, only if L_1 and L_2 can form in language **L** an utterance—i.e., a prosodic unit, or a prosodic phrase of **L**—such as *the window, of John, spouts water or stained glass*, out of any context; the linear position of one of these lexemes in the sentence must be specified with respect to the other.

A prosodic phrase is not formally defined: it is determined by the linguistic intuition of speakers. A prosodic phrase in language **L**, or potential prosodic phrase, is an utterance of **L** that can exist outside of any context; a prosodic phrase in a sentence *S* of **L**, or actual prosodic phrase, is a fragment of *S* separated by pauses and featuring a particular intonation contour. A potential prosodic phrase is always an actual phrase, but not vice versa: thus, in the sentence *For his, so to speak, one-sheet atlas he needs a support system*, the sequence *for his* is an actual prosodic phrase, but not a potential prosodic phrase of English. The difference between potential prosodic phrases, or phrases of language, and actual prosodic phrases, or phrases of discourse parallels that between wordforms of language and wordforms of discourse.⁴

In the sequence *for several decades*, FOR and DECADE_{PL} are syntactically linked: *for decades* is a prosodic phrase of English, and *for* has to be positioned before *decades*.

A caveat: The real state of affairs is, as always, more complicated. The formulation of Criterion A is simplified. First, in fact, Synt-**D** can link lexemes L_1 and L_2 that do not form a prosodic phrase in the language, but do form phrases L_1-L_2-L and L_2-L . For instance, since *left*_{L1} *with*_{L2} *John*_L is a prosodic phrase of English and *with*_{L2} *John*_L also is, it follows that *left* and *with* are syntactically linked. Second, we have to reason in terms of syntactic classes rather than individual lexemes. Thus, if *by John* or *with caution* are prosodic phrases of English, we allow Synt-**D** between any preposition and any noun.

The formulations of Criteria B and C use a different notion of phrase: a syntactic phrase, which is, roughly speaking, a syntactic subtree and/or its projection (see 8). In principle, “prosodic phrase” ≠ “syntactic phrase”; thus, in the Serbian sentence (5), the boldfaced fragment is a prosodic phrase (in this context, not in the language) but by no means a syntactic phrase (neither in this sentence, nor in the language); on the other hand, Serbian syntactic phrases *video—ga* ‘having.seen him’ and *sam—video* ‘am having. seen’ are not prosodic phrases in this sentence (but they are in the language).

(5) **Juče sam ga, kao znaš, video**
 yesterday am him as know- having.seen
PRES. 2SG

‘Yesterday, I have, as you know, seen him’.

SAM ‘am’ and GA ‘him’ are clitics, which explains their specific linear position.

⁴ Wordforms of language exist outside of any context: *birds, sprang, to*, etc. Wordforms of discourse appear in particular contexts only—as a result of an amalgam, such as Fr. *à le* ⇒ *au* /o/ or as that of a syntactic splitting, such as separable prefixes in German: *hört ... auf* ⇐ *auhört* ‘stops, ceases’. See Mel’čuk 1992a: 188ff.

SSynt-Dominance: Criteria B

Criterion B1 (syntactic): The passive Synt-valence of the syntactic phrase

In the syntactic phrase L_1 -synt— L_2 , the lexeme L_1 is the Synt-governor, if the passive SSynt-valence of the whole phrase is determined to a greater extent by the passive Synt-valence of L_1 rather than by that of L_2 .

Thus, the passive SSynt-valence of the syntactic phrase *for decades* is fully determined by the preposition; therefore, *for—synt*→*decades*.

If, and only if, Criterion B1 does not establish the Synt-governor, the next criterion should be applied.

Criterion B2 (morphological): The inflectional links between the phrase and its external context

In the syntactic phrase L_1 -synt— L_2 , the lexeme L_1 is the Synt-governor, if L_1 controls the inflection of lexemes external to the phrase or its own inflection is controlled by such lexemes.

The lexeme L_1 is called the morphological contact point of the phrase L_1 -synt→ L_2 .

Thus, in the Russian phrase *divan-krovat’* lit. ‘sofa-bed’ Criterion B1 does not establish the Synt-governor (both components have the same passive valence); but Criterion B2 singles out DIVAN_(MASC) as the Synt-governor: *èt+ot* [SG. MASC] *divan-krovat’ byl+Ø* [SG.MASC]... ‘this sofa-bed was...’, where the external agreement is with DIVAN_(MASC), and not with KROVAT’_(FEM) ⟨**èt+a divan-krovat’ byl+a...*⟩; therefore, DIVAN-synt→KROVAT’.

If, and only if, Criterion B2 does not establish the Synt-governor, the next criterion should be applied.

Criterion B3 (semantic): The denotation of the phrase

In the syntactic phrase L_1 -synt— L_2 , the lexeme L_1 is the Synt-governor, if L_1 -synt— L_2 denotes a kind/an instance of the denotation of L_1 rather than a kind/an instance of the denotation of L_2 .

In the phrase *noun suffix*, the Synt-governor is SUFFIX, because *noun suffix* denotes a kind of suffix, rather than a kind of noun.

One can say with Zwicky (1993: 295-296) that in a two-word phrase the Synt-governor is the phrase syntactic class determinant, or—if there is no such syntactic determinant—the phrase morphological behavior determinant, or—in case both syntactic and morphological determinants are absent—the phrase semantic content determinant.

Types of SSynt-Relations: Criteria C

For each syntactic phrase $L_1-r \rightarrow L_2$, one has to know exactly which type r of Synt-D links the corresponding lexemes. If at least one of Criteria C is not satisfied, the presumed SSyntRel $r[?]$ should be split in two (or more) SSyntRels.

Criterion C1 (minimal pairs): Absence of semantic contrast

$w(L)$ stands for “a wordform w of the lexeme L .”

An SSyntRel r cannot describe two phrases

$w_1(L_1)-r[?] \rightarrow w_2(L_2)$ and $w_3(L_1)-r[?] \rightarrow w_4(L_2)$, which 1) contrast semantically and 2) differ formally by some syntactic means of expression—i.e., by word order, syntactic prosody or syntactic grammemes.

The configuration Rus. DESJAT'← $r[?]$ →DOLLAR has two implementations with different meanings: *desjat' dollarov* ‘10 dollars’ vs. *dollarov desjat'* ‘maybe 10 dollars’. The formal difference between the two phrases is purely syntactic: word order; therefore, the presumed SSyntRel $r[?]$ is to be split in two SSyntRels:

DESJAT'←**quantitative**-DOLLAR ⇔ *desjat' dollarov*
vs.

DESJAT'←**approx-quant**-DOLLAR ⇔ *dollarov desjat'*.

Criterion C2 (substitutability in context): Syntactic substitutability

$\Delta_{(X)}$ stands for “a SSynt-subtree whose head is a lexeme of the syntactic class X .”

An SSyntRel r of L must possess the following (= “quasi-Kunze”) property: L has a syntactic class X , different from substitute pronouns and such that, for any SSynt-phrase $L-r \rightarrow D_{(Y)}$, replacing $\Delta_{(Y)}$ by $\Delta_{(X)}$ (but not necessarily vice versa!) in any SSyntS of L does not affect its syntactic well-formedness.

This means that an SSyntRel must have a prototypical Dependent, which passes with any possible Governor. In the phrases *have-r[?]*→*been* and *be-r[?]*→*going* the presumed SSyntRel $r[?]$ does not possess the quasi-Kunze property:

**have-r[?]*→*going* and **be-r[?]*→*been*

Therefore, there are two different SSyntRels:

HAVE→**perfect-analytical**→BE
vs.

BE→**progressive-analytical**→GO.

Criterion C3 (repeatability): Repeatability with the same Synt-governor

A SSyntRel r must be either non-repeatable (= no more than one branch labeled r can start from a Synt-governor) or unlimitedly repeat-

able (= any number of branches labeled r can start from a Synt-governor).

In Persian, expressions of the following type are extremely widespread:

(6) *Ramin+ra*←**r-kārd-r[?]**→*bedar*

Ramin DirO made awakening[Noun]

lit. ‘[He/she/it] made [the] awakening Ramin’. = ‘He/she/it awoke Ramin’.

These expressions are built on verbal collocations of the type *bedar kārd* ‘awakening made’ = ‘woke up’ or *dārs dād* lit. ‘lesson gave’ = ‘taught’, which, although they seem to include a DirO, such as BEDAR or DĀRS, behave as transitive verbs and take—as a whole—a “genuine” DirO (the suffix **-ra** is an unmistakable marker of DirO). The presumed SSyntRel $r[?]$ (direct-objectival?) in such expressions would be limitedly repeatable—just twice. Therefore, there are two different SSyntRels:

RAMIN←**dir-obj**-KĀRD→**quasi-dir-obj**→BEDAR

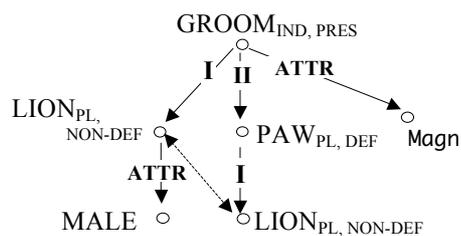
The nominal element in such verbal collocations is considered to be a Quasi-Direct Object.

Using the above criteria (plus considerations of analogy), a list of SSyntRels for a particular language can be obtained; in the Annex, I give such a list for English (Mel’čuk and Pertsov 1987: 85-156, Mel’čuk 2009: 52-58).

5.4 Examples of Deep- vs. Surface-Synt-Structures

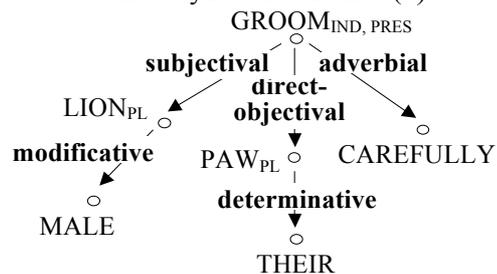
In order to show how Synt-relations work, the two (Deep- and Surface-) SyntSs of the sentence in (3) are given.

(7) a. The DSyntS of sentence (3)



In the DSyntS of (7a) the link of coreferentiality is shown (←-----→).

b. The SSyntS of sentence (3)



6 Morphological Dependency

The two types of morphological relations—agreement and government—are conveniently described in terms of dependency. Let us consider the Latin sentence (8), a fragment of a poem by Catullus (for more on agreement and government, see Mel’čuk 2006: 31-105):

(8) *Tu solebas meas esse*
 you-NOM used-2SG my-FEM.PL.ACC be-INF
aliquid putare nugas.
 something-NOM think-INF trifles(FEM)-PL.ACC
 ‘You used to think that my trifles are something’.

Take a pair of lexemes linked by Morph-D:

$L_1 \leftarrow \text{morph-} L_2$.

6.1 Agreement

Lexeme L_1 agrees with lexeme L_2 in inflectional category \mathbf{C}_1 , if and only if the following two conditions are simultaneously satisfied:

1) L_1 is not a substitute pronoun that replaces an occurrence of L_2 .

2) L_1 must receive the grammeme $\mathbf{G}_1 \in \mathbf{C}_1$ that is selected depending—either upon a grammeme $\mathbf{G}_2(L_2)$ such that $\mathbf{G}_2 \in \mathbf{C}_2$ and \mathbf{C}_1 is mirroring⁵ for \mathbf{C}_2 ,

—or upon the value of a syntactic feature $\Sigma_2(L_2)$, this feature being an agreement class, pronominal person or pronominal number.

Sentence (8) presents two cases of agreement:

- MEUS_{L_1} ‘my’ agrees with NUGAE_{L_2} ‘trifles’—in gender (a syntactic feature of L_2), and in number/case (grammmemes of L_2 in this sentence)

- SOLERE_{L_1} ‘use to’ agrees with TU_{L_2} ‘you’—in person and number (syntactic features of L_2)

6.2 Government

Lexeme L_1 is governed by lexeme L_2 ($\leftarrow L_2$ governs L_1) with respect to inflectional category \mathbf{C}_1 , if and only if the grammeme $\mathbf{G}_1 \in \mathbf{C}_1$ is selected depending

—either upon the value of a syntactic feature $\Sigma_2(L_2)$ that is neither agreement class, pronominal person, or pronominal number [standard case];

—or upon a grammeme $\mathbf{G}_2 \in \mathbf{C}_2$ such that \mathbf{C}_1 is not mirroring for \mathbf{C}_2 [special case].

Sentence (8) presents the following instances of government:

⁵ An inflectional category \mathbf{C}_1 is mirroring for the category \mathbf{C}_2 if and only if the grammemes of \mathbf{C}_1 simply “reflect” the grammemes of \mathbf{C}_2 and do not do anything else.

- SOLERE_{L_2} governs the nominative of TU_{NOM} and the infinitive of $\text{PUTARE}_{\text{INF}}$

- PUTARE_{L_2} governs the accusative of $\text{NUGAE}_{\text{ACC}}$ and the infinitive of ESSE_{INF}

- ESSE_{L_2} governs the nominative of $\text{ALIQUID}_{\text{NOM}}$

7 What Syntactic Dependency Is Good For

Among different linguistic phenomena that can be described adequately in terms of syntactic dependency, but cannot be in terms of constituency, I will consider the following four.

7.1 Diatheses and voices

A diathesis of a lexeme L is the correspondence between its Sem-actants [= SemAs] and DSyntAs. To give an example, the verbs FOLLOW and PRECEDE have inverted diatheses: X_I follows $Y_{II} \equiv Y_I$ precedes X_{II} ; symbolically, their respective diatheses appear as $X \Leftrightarrow I, Y \Leftrightarrow II$ for FOLLOW and $X \Leftrightarrow II, Y \Leftrightarrow I$ for PRECEDE. Such a formulation, as well as the notion itself of actant—on three different levels (SemAs, DSyntAs and SSyntAs, see Mel’čuk 2004)—is possible only within a dependency framework.

This description of diathesis leads to clear definition of voice: a voice is a particular diathesis explicitly marked grammatically. Among other things, the correlation between the active and the passive voices can be represented in the same way: X_I follows $Y_{II} \equiv Y_I$ is followed by X_{II} . One can develop a calculus of voices by combining all permutations of DSyntAs of L with respect to its SemAs, DSyntA suppression and their referential identification (see Mel’čuk 2006: 181-262).

7.2 Lexical Functions

To describe regular collocations of the type *wield authority, pursue a policy* or *honor a commitment*, Meaning-Text theory proposes an inventory of a few dozen Lexical Functions [= LFs]; cf. $\text{Real}_1(\text{AUTHORITY}) = \text{wield} [\sim]$, $\text{Real}_1(\text{POLICY}) = \text{pursue} [\text{ART } \sim]$, $\text{Real}_1(\text{COMMITMENT}) = \text{honor} [\text{ART } \sim]$. Similarly, $\text{AntiVer}(\text{PROMISE}) = \text{empty}$, $\text{AntiVer}(\text{EXAMPLE}) = \text{poor}$, $\text{AntiVer}(\text{DREAM}) = \text{pipe} [\sim]$. An LF is applied to the base of a collocation (in small caps above) and returns the corresponding collocates. LFs, specified for a lexeme in its lexical entry, allow for correct lexical choices under text generation or automatic translation, as well as for efficient paraphrasing, equally necessary for these tasks. No less is their role in lexicography, in language teaching and learning.

However, the base of a collocation and its collocates are always linked by a particular Synt-D, specific for a given LF:

$\text{Real}_1(L) - \text{II} \rightarrow L, L - \text{ATTR} \rightarrow \text{AntiVer}(L), \text{etc.}$

Thus, the LF formalism is only possible based on a dependency syntactic approach.

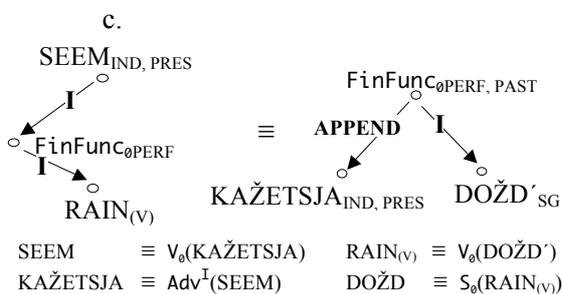
7.3 Paraphrasing

Expressing the syntactic structure of a sentence in terms of Synt-D opens the way for powerful **paraphrasing**—that is, the calculus of sets of semantically equivalent DSyntSs. Such paraphrasing proves to be absolutely necessary in translation because of lexical, syntactic and morphological mismatches between sentences of different languages that translate each other (Mel’čuk and Wanner 2001, 2006, 2008). An example of such mismatches can be the translation of the English sentence (9a) into Russian (and *vice versa*):

(9) a. *It seems to have stopped raining.*

b. *Dožd’, kažetsja, perestal*
lit. ‘Rain, [it] seems, stopped’.

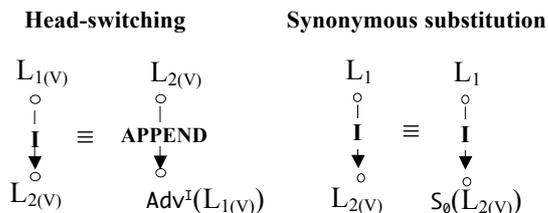
The respective DSyntSs of these sentences and lexical equivalences are given in (9c):



FinFunc₀ in both trees is a lexical function meaning roughly ‘cease to take place’; FinFunc₀(RAIN_N) = *stop*, while FinFunc₀(DOŽD') = *končit'sja, perestat', prekratit'sja, projti*.

The DSynt-paraphrasing rules necessary for this transition are as follows (with serious simplifications):

(10) Two DSynt-Paraphrasing Rules



These rules are formulated in terms of Lexical Functions and simple DSynt-transformations. Given the limited number of LFs and of DSyntRels, on the one hand, and the fact that all

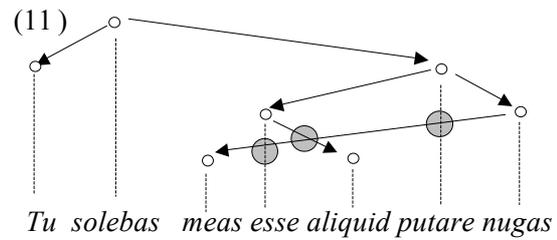
DSynt-transformations can be easily reduced to a few minimal ones, on the other, it is possible to develop an exhaustive set of DSynt-paraphrasing rules, which cover all potential paraphrases in all languages (Mel’čuk 1992b and Milićević 2007).

7.4 Word order

One of the most universal properties of word order in different languages—so-called **projectivity**—can be remarked and described only in terms of dependency.

The word order in the sentence *S* is **projective**, if and only if in the projection of the SSyntS(*S*) on *S* no dependency arrow crosses another dependency arrow or a projection perpendicular.

Sentence (8) is non-projective, cf. the SSyntS projected on it in (11); shaded circles indicate “crime scenes”—that is, the spots of projectivity violations:



However, a **crushed majority** of sentences in texts are projective, which allows for a simpler and more general word order rule. Namely, under synthesis or analysis, it is required that the sentence produced or analyzed be projective. Non-projective sentences are not only very rare, but are possible solely under stringent conditions, which can be easily verified.

8 Where Syntactic Dependency Is Not Sufficient

As far as I know, there is only one syntactic phenomenon for whose description “pure” dependencies prove insufficient: a coordinated phrase with a modifier (boldfaced below) that bears either on the whole phrase (i.e., on all its elements) or just on one element. Here is the stock example:

(12) a. **old men and women:**
either ‘old men’ + ‘women’
or ‘old men’ + ‘old women’

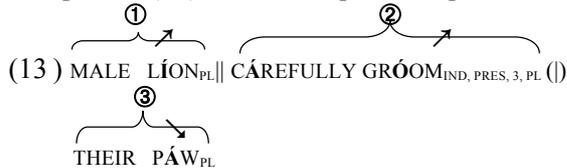
This contrast cannot be expressed in a natural way in terms of dependency so as to preserve the arborescent structure. Therefore, an additional technique is necessary: in case the suspicious element bears on the whole phrase, the corresponding subtree must be explicitly indicated, as in (12b):

- b. *old* ← [–*men* → *and* → *women*]:
 ‘old men + old women’
 vs.
old ← –*men* → *and* → *women*:
 ‘old men + women’

The subtree specified in such a way is called a **syntactic grouping**; a grouping corresponds to a syntactic phrase, but it is not a constituent in the classical sense of the term.

9 Constituents vs. Phrases

Now, what about “classical” constituents? They cannot be part of a syntactic structure, simply because they—no matter how we define them—are a linguistic means used to express the syntactic structure of a sentence. Therefore, their natural place is in the Deep-Morphological representation, where they appear in the DMorph-Prosodic structure—but not as constituents in the strict sense of the term (constituents coming together to form a constituent of a higher rank and thus forming a hierarchy): as specification of actual prosodic phrases, with the corresponding pauses, stresses and contours. Sentence (3) has the DMorphR in (13), with three prosodic phrases:



Prosodic phrases fragments are by no means constituents: there is no hierarchy between them (= no embeddings).

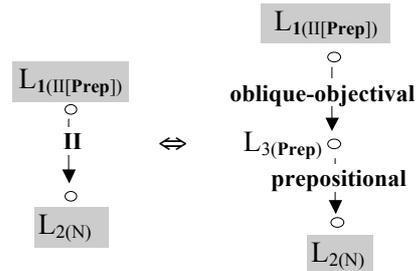
However, as it often happens in linguistics, the term *phrase* is also widely used in a different sense: as a **syntactic phrase**. (Although I am trying to avoid polysemy of terms, I did not dare to replace *phrase*.) Syntactic phrases are of two major types:

- **Potential syntactic phrases** are abstract schemata of basic syntactic constructions of a language; they are stated in terms of parts of speech and syntactic features, such as $N \leftarrow V_{FIN}$, $V \rightarrow N$,

$V \rightarrow N$, $A \leftarrow N$, $Prep \rightarrow N$, $Adv \leftarrow V$, etc. Potential phrases are necessarily minimal, i.e., binary; they do not appear in syntactic representations, but are used in syntactic rules, both deep and surface. For instance, here are a DSynt-rule and an SSynt-rule.

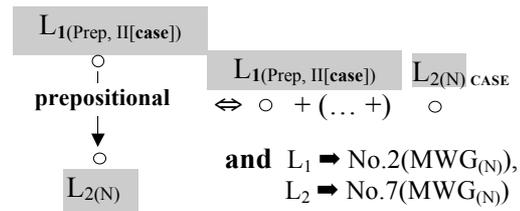
While a DSynt-rule describes a mapping of a deep subtree on a surface subtree, an SSynt-rule linearizes and morphologizes a surface subtree, using, among other means, general schemata, or patterns, of Minimal Word Groups.

A Deep-Synt-rule



The shaded zones represent the context—that is, the elements that are not affected by the given rule, but control its applicability.

A Surface-Synt-rule



“MWG” stands for ‘minimal word group,’ see below; No.2 and No.7 refer to the corresponding positions in an MWG pattern.

The left-hand part of any syntactic rule consists of a potential (Deep or Surface) syntactic phrase. The right-hand part of a Surface-Syntactic rule gives the basic information on the linear arrangement of the elements by specifying their mutual disposition, the possible “gap” between them and their positions in the corresponding MWG pattern. For instance, a nominal MWG_(N) pattern for Russian looks as follows:

1	2	3	4	5	6	7	8
coordinate conjunction	preposition	demonstrative	numeral	possessive adjective	adjective	noun	formula
ILI ‘or’	DLJA ‘for’	ÈTI ‘these’	TRI ‘three’	NAŠ ‘our’	INTERESNYJ ‘interesting’	PRIMER ‘example’	(11)
<i>ili dlja ètix trëx našix interesnyx primerov (11)</i> ‘or for these three our interesting examples (11)’							

Figure 2: Pattern of the Russian Nominal Minimal Word Group

- **Actual syntactic phrases** are real utterances of the language, such as *John depends, depends on John, for her survival, depends on John for her*

survival, etc. These phrases can be simple (= minimal: two lexemes) or complex (= of any length: any number of lexemes). An actual syntactic phrase is a



subtree of an SSyntS and/or its linear projection.

The DSynt-rule above covers such actual syntactic phrases as *depend on John*; more specifically, it produces their SSyntS:

DEPEND-II → JOHN ⇔
DEPEND-obl-obj → ON-prepos → JOHN

The SSynt-rule ensures the linearization and morphologization of such actual syntactic phrases as Rus. *ot Džona* ‘from/on John’:

OT-prepos → DŽON ⇔ OT DŽON_{GENITIVE}.

An actual syntactic phrase corresponds, most of the time, to a potential prosodic phrase—yet, as stated above, these two entities are conceptually different; thus, sentence (8) has the DMorphR as in (14a), with four prosodic phrases, while it contains only three actual syntactic phrases, shown in (14b):

(14) a. DMorphR of (8) (the symbol “<” indicates the immediate linear precedence)

TU_{NOM} < SOLERE_{IND, IMPF, 2, SG} | < MEUS_{FEM, PL, ACC} |

< ESSE_{INF} < ALIQUID_{NOM} | < PUTARE_{INF} < NUGA_{PL, ACC}

b. *tu solebas putare; meas nugas;
esse aliquid*

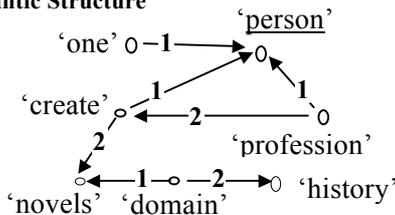
10 “Bracketing Paradox”

I became aware of the so-called “bracketing paradox” thanks to an exchange with T.M. Gross; I thank him for explaining to me why the phrases of the type *historical novelist* or *nuclear physicist* are problematic for some theoretical frameworks. The suffix *-ist* seems to be added to a phrase rather than to a nominal stem, which would be the normal case: [*historical novel*]+*ist* ‘one whose profession is to write + historical novels’ and [*nuclear physics*]+*ist* ‘one whose profession is to study + nuclear physics’. But if our task as linguists is to formally describe the correspondence between the meaning and the structure of these phrases, here is what we obtain.

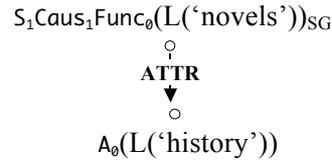
First, we need the representations of the phrase in question at different levels: semantic, deep-syntactic, surface-syntactic and deep-morphological.

Four representations of the phrase *historical novelist*

Semantic Structure

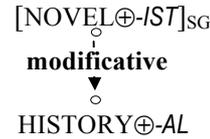


Deep-Syntactic Structure



S₁Caus₁Func₀ is a complex lexical function meaning roughly ‘one who causes to exist’.

Surface -Syntactic Structure

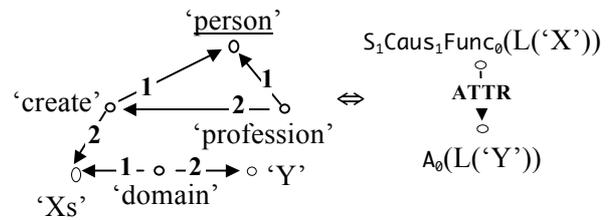


Deep-Morphological Structure

HISTORY ⊕ AL < [NOVEL ⊕ -IST]_{SG}

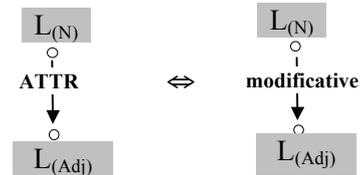
Second, we write rules that relate these representations, for instance:

Semantic rule (SemR ⇔ DSyntR)



Deep-Syntactic rules (DSyntS ⇔ SSyntS)

1. S₁Caus₁Func₀(NOVEL) ⇔ NOVEL ⊕ -IST
2. A₀(HISTORY) ⇔ HISTORY ⊕ AL
- 3.



Rules 1 and 2 are fragments of the lexical entries for the respective lexemes; HISTORY ⊕ AL will be turned into *historical* by morphological rules of allomorphy and morphological rules. Rule 3 realizes DSyntR ATTR by the SSyntRel *modificative*.

And nothing resembling a paradox can be found... The moral of the story: if you do not want paradoxes, don’t create them by your own descriptive means!

11 Conclusion

After this longish text, the conclusion can be very short: To describe the structure of linguistic expressions on all levels linguistic dependencies are necessary and sufficient. Constituents (in the classical sense) do not exist; phrases do of course exist, but they are of two types—prosodic and syntactic, and only prosodic phrases appear in a linguistic repre-

sentation (in the DMorphR); syntactic phrases are used in syntactic rules only.

Acknowledgments

The text of this talk has been read and criticized by D. Beck, L. Iordanskaja, S. Kahane, J. Milićević and L. Wanner; I express to them my heartfelt gratitude. At the same time, I assume the full responsibility for all mistakes and inconsistencies that survived their scrutiny.

Appendix: A Tentative List of English SSynt-Relations

I. Subordinate SSyntRels: 1 - 50

CLAUSE-LEVEL (= CLAUSAL) SSYNTRELS: 1 - 21

These SSyntRels link between themselves the elements of the sentence—the maximal syntactic phrases.

Valence-controlled SSyntRels: Complementation

Actantial SSyntRels

1. Subjectival:

I←subj→*am old*.

Intervention←subj→*seems [impossible]*.

Smoking←subj→*is [dangerous]*.

That←subj→[*Alan can do that*]→*is [clear]*.

It←subj→*is [clear that Alan can do that]*.

2. Quasi-Subjectival:

[*It*←subj→]*is*→[*clear*]→quasi-subj→*that* [Alan can do that].

3. Direct-Objectival:

sees→dir-obj→*me*

[*to have*] *written*→dir-obj→*novels*

[*Helen*] *wants*→dir-obj→*Alan* [*to read*].

worth→[*a*]→dir-obj→*trip*

prefer→[*her*]→dir-obj→*staying* [*home*]

explain→[*to me*]→dir-obj→*that* [*Alan was absent*]

make→dir-obj→*it* [*possible to neutralize the consequences*]

4. Quasi-Direct-Objectival:

make→[*it possible*]→quasi-dir-obj→*to* [*neutralize the consequences*]

5. Indirect-Objectival:

gives→indir-obj→*Alan /him* [*some money*]

convince→[*Alan*]→indir-obj→*that* [*he should work less*]

6. Oblique-Objectival:

depends→obl-obj→*on* [*Alan*]

my respect→obl-obj→*for* [*Alan*]

translation→obl-obj→*from* [*French into Polish*]

translation→[*from French*]→obl-obj→*into* [*Polish*]

7. Infinitival-Objectival:

can→inf-obj→*read*; *want*→inf-obj→*to* [*read*]

[*Helen*] *wants*→[*Alan*]→inf-obj→*to* [*read*].

[*Helen*] *makes*→[*Alan*]→inf-obj→*read*.

[*her*] *desire*→inf-obj→*to* [*come home*]

8. Completive:

find→[*this*]→compl→*easy*

consider→[*Alan*]→compl→*happy*

make→[*it*]→compl→*possible*

make→[*Helen a good*]→compl→*wife*

9. Copular:

be→copul→*easy*; *be*→[*a*]→copul→*teacher*

be→copul→*without* [*a hat*]

seem→copul→*in* [*a difficult position*]

10. Agentive:

written→agent→*by* [*Alan*]

arrival→agent→*of* [*Alan*]

shooting→agent→*of* [*the hunters*: ‘the hunters shoot’]

[*a*] *translation*→agent→*by* [*Alan*]

[*I like*] *for*←agent→[*Alan to*]→*play* [*cards*].

11. Patientive:

translation→patient→*of* [*this text*]

shooting→patient→*of* [*the hunters*: ‘the hunters are shot’]

Copredicative SSyntRels

12. Subject-copredicative:

[*Alan*] *returned*→subj-copred→*rich*.

13. Object-copredicative:

[*Alan*] *likes*→[*Helen*]→obj-copred→*slim*.

[*Alan*] *hammered*→[*the coin*]→obj-copred→*flat*.

Comparative SSyntRel

14. Comparative:

older→compar→*than* [*Leo*]

[*He loves Helen*] *more*→compar→*than* [*Leo*].

more→[*important*]→compar→*than* [*Leo*]

as→[*important*]→compar→*as* [*Leo*]

Non-Valence-controlled SSyntRels: Modification

Absolutive SSyntRel

15. Absolute-predicative:

[*His first*] *attempt*→[*a*]→abs-pred→*failure*, [*he...*]

[*He went out, his*] *anger*→abs-pred→*gone*.

[*He ran, his*] *gun*→abs-pred→*in* [*his left hand*].

Adverbial SSyntRels

16. Adverbial:

walk→adverb→*fast*; *delve*→adverb→*deeply*

[*He*] *works*→adverb→*there* (in [*this office*]).

[*will*] *write*→[*next*]→adverb→*week*

[*He*] *ran*,→[*his*]→adverb→*gun* [*in his left hand*].

With←adverb→[*the text finished, Helen*]→*can afford* *this trip*.



17. Modificative-adverbial:
[As always] **elegant**, ← **mod-adverb**-[Alan]-walked [away].
18. Appositive-adverbial:
[An old] **man**, ← **appos-adverb**-[Alan]-works [less].
19. Attributive-adverbial:
Abroad, ← **attr-adverb**-[Alan]-works [less].

Sentential SSyntRels

20. Parenthetical:
Oddly, ← **parenth**-[Alan] works [less].
Alan, **naturally**, ← **parenth**-accepted it.
As ← **parenth**-[we know, Alan]-works [less].
To ← **parenth**-[give an example, I]-consider [now nominal suffixes].
21. Adjunctive:
OK, ← **adjunct**-[I]-agree

PHRASE-LEVEL (= PHRASAL) SSYNTRELS: 22 - 50

These SSyntRels function within elements of the sentence—inside maximal phrases.

General Phrase SSyntRels

Non-valence-controlled SSyntRels: Modification

22. Restrictive:
still ← **restr-taller**; **most** ← **restr-frequent**
not ← **restr-here**
[Alan has] **just** ← **restr-arrived**.

Noun Phrase SSyntRels

Valence-controlled SSyntRels: Complementation

23. Elective:
[the] **poorest** ← **elect** → **among** [peasants]
[the] **best** ← [ones] ← **elect** → **of** (**from**) [these boys]
five ← **elect** → **of** these books
[the] **most** ← [expensive car] ← **elect** → **in** [France]

Mixed Type SSyntRels = Valence-controlled/ Non-Valence-controlled: Modification

24. Possessive:
Alan's ← **poss-arrival**; Alan's ← **poss-bed**
Alan's ← **poss-garden**
25. Compositive:
man ← **compos** [-machine]-interaction;
car ← **compos-repair**
noun ← **compos-phrase**; **color** ← **compos-blind**

Non-Valence-controlled SSyntRels: Modification

26. Determinative:
my ← **determ-bed**; **a** ← **determ-bed**;
those ← **determ-beds**
27. Quantitative:
three ← **quant-beds**
[three ← **num-junct**] - **thousand** ← **quant-people**
28. Modificative:
comfortable ← **modif-beds**

visible ← **modif-stars**
French ← **modif-production**

29. Post-modificative:
stars ← **post-modif** → **visible** (vs. *visible stars*)
30. Descriptive-Modificative:
[these beds, -**descr-modif** → **comfortable** [and not expensive], ...
31. Relative:
[the] **paper** ← [that I] ← **relat** → **read** [yesterday]
[the] **paper** ← [I] ← **relat** → **read** [yesterday]
the girl ← [who] ← **relat** → **came** [first]
32. Descriptive-Relative:
[this] **paper** ← [which I] ← **descr-relat** → **read** [yesterday]
Alan, -[who] ← **descr-relat** → **loves** [her so much]
33. Appositive:
Alan - [the] ← **appos** → **Powerful**
General ← **appos-Wanner**
[the] **term** ← **appos** → **'suffix'**
34. Descriptive-Appositive:
[This] **term** ← **descr-appos** → (**'suffix'**) [will be considered later].
[You forget about] **me**, -[your] ← **descr-appos** → **mother**
35. Sequential:
man ← **sequent** → **machine** [interaction]
fifty ← **sequent** → **to** [seventy dollars]
36. Attributive:
learner ← **attr** → **with** [different backgrounds]
dress ← **attr** → **of** [a beautiful color]
years ← **attr** → **of** [war]; **bed** ← **attr** → **of** [Alain]
man - [the same] ← **attr** → **age**
37. Descriptive-Attributive:
[Professor] **Wanner**, -**descr-attr** → **from** [Stuttgart, was also present]

Prepositional Phrase SSyntRels

A valence-controlled SSyntRel: Complementation

38. Prepositional:
in ← **prepos** → **bed**;
without - [three hundred] ← **prepos** → **dollars**
a year ← **prepos-ago**

A non-valence-controlled SSyntRel: Complementation (by analogy)

39. Prepositional-infinitival:
to ← **prepos-inf** → **go** [to bed]

Verb Phrase (= Analytical Form) SSyntRels

Non-valence-controlled SSyntRels: Ancillary

40. Perfect-analytical:
has ← **perf-analyt** → **written**
has ← **perf-analyt** → **been** [beaten]
41. Progressive-analytical:
was ← **progr-analyt** → **writing**



42. Passive-analytical:

was–pass-analyt→*written*

Conjunction Phrase SSyntRels

Valence-controlled SSyntRels: Complementation

43. Subordinate-Conjunctive:

[*Suppose*] *that*–[*Alan*]–subord-conj→*comes*.

[*so*] *as*–[*not*]–subord-conj→*to* [*irritate Leo*]

44. Coordinate-Conjunctive:

[*Alan*] *and*–coord-conj→*Helen*

45. Comparative-Conjunctive:

than–compar-conj→*Helen*

as–compar-conj→*always*

46. Absolute-Conjunctive:

If–[*a*]–abs-conj→*pronoun*, [*the grammatical subject may...*]; *while*–abs-conj→*in* [*bed*]

Word-like Phrase SSyntRels

Non-valence-controlled SSyntRels: Ancillary

47. Verb-junctive:

give–verb-junct→*up*

bring–verb-junct→*down*

48. Numeral-junctive:

fifty←num-junct–*three*

fifty←num-junct–*third*

49. Binary-junctive:

if–[...]-bin-junct→*then...*

the–[*more...*]-bin-junct→*the* [*more...*]

till–bin-junct→*after*

from–[...]-bin-junct→*to* [...]

either–[...]-bin-junct→*or* [...]

50. Colligative:

[*is*] *dealt*–collig→*with* [*stranded prepositions*]

II. Coordinate SSyntRels: 51 – 52

Non-valence-controlled SSyntRels: Coordination

51. Coordinative:

Alan–coord→*and* [*Leo*]

rich,–coord→*intelligent*–coord→*and* [*beautiful*]

52. Quasi-coordinative:

[*He was*] *abroad*–quasi-coord→*without*–[*a penny*]–quasi-coord→*in* [*a desperate situation*].

[*These moneys we keep hidden*] *under*–[*a loose board*]–quasi-coord→*under*–[*the floor*]–quasi-coord→*under*–[*a chamber pot*]–quasi-coord→*under* [*my friend's bed*] [T. Capote, “A Christmas Memory”].

References

Lidija Iordanskaja and Igor Mel'čuk. 2009. Establishing an Inventory of Surface-Syntactic Relations: Valence-controlled Surface-dependents of the Verb in French. In: Polguère and Mel'čuk (eds) 2009: 151-234.

Sylvain Kahane. 2003. The Meaning-Text Theory. In: V. Agel, L. Eichinger, H.-W. Eroms, P. Hellwig, H. J. Herringer, H. Lobin (eds), *Dependency and Valency. An International Handbook of Contemporary Research*, vol. 1, W. de Gruyter, Berlin/New York, 546-570.

Igor' Mel'čuk. 1957. O nekotoryx voprosax MP s vengerskogo jazyka na russkij [On Some Problems of Automatic Translation from Hungarian into Russian]. *Bjulleten' ob'edinenija po problemam MP*, 4:1-75.

Igor' Mel'čuk. 1963. Avtomatičeskij analiz tekstov (na materiale russkogo jazyka) [Automatic Text Analysis (Based on Russian Data)]. In: *Slavjanskoe jazykoznanie*, Nauka, Moskva, 477-509.

Igor' Mel'čuk. 1974. *Opyt teorii lingvističeskix modelej «Smysl ⇔ Tekst»* [Outline of a Theory of Meaning-Text Linguistic Models]. Nauka, Moskva [1999: Škola «Jazyki russkoj kul'tury», Moskva].

Igor Mel'čuk. 1979. *Studies in Dependency Syntax*. Karoma, Ann Arbor, MI.

Igor Mel'čuk. 1981. Meaning-Text Models: A Recent Trend in Soviet Linguistics. *Annual Review of Anthropology*, 10:27-62.

Igor Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. The SUNY Press, Albany, NY.

Igor Mel'čuk. 1992a. *Cours de morphologie générale. Vo. I*. Les Presses de l'Université de Montréal/CNRS Édition, Montréal/Paris.

Igor Mel'čuk. 1992b. Paraphrase et lexique: la théorie Sens-Texte et le Dictionnaire explicatif et combinatoire. In: Mel'čuk et al., *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques III*, Les Presses de l'Université de Montréal, Montréal, 9-58.

Igor Mel'čuk. 1997. *Vers une linguistique Sens-Texte. Leçon inaugurale*. Collège de France, Paris.

Igor Mel'čuk. 2002. Language: Dependency. In: N. Smelser and P. Baltes (eds), *International Encyclopedia of the Social and Behavioral Sciences*, Pergamon, Oxford, 8336-8344.

Igor Mel'čuk. 2003. Levels of Dependency in Linguistic Description: Concepts and Problems. In: V. Agel, L. Eichinger, H.-W. Eroms, P. Hellwig, H. J. Herringer, H. Lobin (eds), *Dependency and Valency. An International Handbook of Contemporary Research*, vol. 1, W. de Gruyter, Berlin/New York, 188-229.

Igor Mel'čuk. 2004. Actants in Semantics and Syntax. I: Actants in Semantics. *Linguistics*, 42(1):1-66; II: Actants in Syntax. *Linguistics*, 42(2):247-291.

Igor Mel'čuk. 2006. *Aspects of the Theory of Morphology*. W. de Gruyter, Berlin/New York.

Igor Mel'čuk. 2009. Dependency in Natural Language. In: Polguère & Mel'čuk (eds) 2009: 1-110.



- Igor Mel'čuk and Nikolaj Pertsov, Nikolaj. 1987. *Surface Syntax of English. A Formal Model within the Meaning-Text Framework*. Benjamins, Amsterdam/Philadelphia.
- Igor Mel'čuk and Leo Wanner. 2001. Towards a Lexicographic Approach to Lexical Transfer in Machine Translation (Illustrated by the German-Russian Language Pair). *Machine Translation*, 16(1): 21-87.
- Igor Mel'čuk and Leo Wanner. 2006. Syntactic Mismatches in Machine Translation. *Machine Translation*, 20(2):81-138.
- Igor Mel'čuk and Leo Wanner. 2008. Morphological Mismatches in Machine Translation. *Machine Translation*, 22:101-152.
- Milićević, Jasmina. 2007. *La paraphrase. Modélisation de la paraphrase langagière*. Peter Lang, Bern.
- Alain Polguère and Igor Mel'čuk (eds). 2009. *Dependency in Linguistic Description*. Benjamins, Amsterdam/Philadelphia.
- Arnold Zwicky. 1993. Heads, Bases and Functors. In: G. Corbett, N. Fraser & S. McGlashan (eds), *Heads in Grammatical Theory*, 1993, Cambridge University Press, Cambridge, 292-315.

Defining dependencies (and constituents)

Kim Gerdes

LPP

Sorbonne Nouvelle, Paris

kim@gerdes.fr

Sylvain Kahane

Modyco

University Paris Ouest

sylvain@kahane.fr

Abstract

The paper proposes a mathematical method of defining dependency and constituency provided linguistic criteria to characterize the acceptable fragments of an utterance have been put forward. The method can be used to define syntactic structures of sentences, as well as discourse structures for texts or morphematic structures for words.

Keywords: connection graph, dependency tree, phrase structure.

1 Introduction

Syntacticians generally agree on the hierarchical structure of syntactic representations. Two types of structures are commonly considered: Constituent structures and dependency structures (or mixed forms of both, like headed constituent structures, sometimes even with functional labeling). However, these structures are rarely clearly defined and often purely intuition-based as we will illustrate with some examples. Even the basic assumptions concerning the underlying mathematical structure of the considered objects (ordered constituent tree, unordered dependency tree) are rarely motivated (why syntactic structures should be trees?).

In this paper, we propose a definition of syntactic structures that supersedes constituency and dependency, based on a minimal axiom: *If an utterance can be separated into two fragments, we suppose the existence of a connection between these two parts.* We will show that this assumption is sufficient for the construction of rich syntactic structures.

The notion of *connection* stems from Tesnière who says in the very beginning of his *Éléments de syntaxe structurale* that “Any word that is part of a sentence ceases to be isolated as in the dictionary. Between it and its neighbors the mind perceives **connections**, which together form the structure of the sentence.” Our axiom is less strong than Tesnière's, because we do not presuppose that the connections are formed between words only.

We will investigate the linguistic characteristics defining the notion of “fragment” and how this notion leads us to a well-defined graph-based structure, to which we can apply further conditions leading to dependency or constituent trees. We will start with a critical analysis of some definitions in the field of phrase structure and dependency based approaches (Section 2). Connection structures are defined in Section 3. They are applied to discourse, morphology, and deep syntax in Section 4. The case of surface syntax is explored in Section 5. Dependency structures are defined in Section 6 and constituent structures in Section 7.

2 Previous definitions

2.1 Defining dependency

Tesnière (1959) does not go any further in his definition of dependency and remains on a mentalist level (“the mind perceives connections”). The first formal definition of dependency stems from Lecerf (1960) and Gladkij (1966) (see also Kahane 1997) who showed that it is possible to infer a dependency tree from a constituent tree with heads (what is commonly called *phrase structure*). Further authors have tried to overcome these first definitions of constituency. Mel'čuk (1988: 130-132) proposes a definition of fragments of two words connected together. But



it is not always possible to restrict the definition to two-word fragments. Consider:

(1) *The dog slept.*

Neither *the slept* nor *dog slept* are acceptable syntactic fragments. Mel'čuk resolves the problem by connecting *slept* with the head of *the dog*, which means that his definitions of fragments and heads are mingled. Moreover Mel'čuk's definition of the head is slightly circular: "In a sentence, wordform w1 directly depends syntactically on wordform w2 if the passive [surface] valency of the phrase w1+w2 is (at least largely) determined by the passive [surface] valency of wordform w2." However, the concept of passive valency presupposes the recognition of a hierarchy, because the passive valency of a word or a fragment designates the valency towards its governor (see Section 6.1).

Garde (1977) does not restrict his definition of dependency to two-words fragments but considers more generally "significant elements" which allows him to construct the dependency between *slept* and *the dog*. However, he does not show how to reduce such a dependency between arbitrary "significant elements" to links between words. The goal of this article is to formalize and complete Garde's and Mel'čuk's definitions.

Schubert (1987:29) attempts to define dependency as "directed co-occurrence" while explicitly including co-occurrence relations between "distant words". He explains the directedness of the co-occurrence by saying that the "occurrence of certain words [the dependent] is made possible by the presence of other words," the governor. However, "form determination should not be the criterion for establishing co-occurrence lines." This adds up to lexical co-occurrences rather than syntactic dependencies. Hudson (1994) precisely proposes to keep this type of dependencies. For our part, we want to restrict connection and dependency to couples of elements which can form an acceptable text fragment in isolation (which is not the case of the radio *playing*). We do not disagree that some sort of dependency exists between radio and *playing*, but we consider this link as a lexical or semantic dependency (Mel'čuk 1988, 2011) rather than as a surface syntactic one.

2.2 Defining constituency

In order to evaluate the cogency of a definition of dependency based on a pre-existing definition of constituency, we have to explore how constituents are defined.

Bloomfield (1933) does not give a complete definition of syntactic constituents. His definition of the notion of *constituent* is first given in the chapter *Morphology* where he defines the morpheme. In the chapter on syntax it is said that "Syntactic constructions are constructions in which none of the immediate constituents is a bound form. [...] The actor-action construction appears in phrases like: *John ran, John fell, Bill ran, Bill fell, Our horses ran away.* [...] The one constituent (*John, Bill, our horses*) is a form of a large class, which we call *nominative expressions*; a form like *ran* or *very good* could not be used in this way. The other constituent (*ran, fell, ran away*) is a form of another large class, which we call *finite verb expressions*." Bloomfield does not give a general definition of constituents: They are only defined by the previous examples as instances of distributional classes. The largest part of the chapter is dedicated to the definition of the head of a construction. We think that in some sense Bloomfield should rather be seen as a precursor of the notions of connection (called *construction*) and dependency than as the father of constituency.

For Chomsky, a constituent exists only inside the syntactic structure of a sentence, and he never gives precise criteria of what should be considered as a constituent. In Chomsky (1986), quarreling with the behaviorist claims of Quine (1986), he refutes it as equally absurd to consider the fragmentation of *John contemplated the problem* into *John contemplated – the problem* or into *John contemp – lated the problem* instead of the "correct" *John – contemplated the problem*. No further justification for this choice is provided.

Gleason (1961:129-130) proposes criteria to define constituents (like substitution by one word, possibility to be a prosodic unit) and to build a constituent structure bottom up: "We may, as a first hypothesis, consider that each of [the words of the considered utterance] has some storable relationships to each other word. If we can describe these interrelationships completely, we will have described the syntax of the utterance in its entirety. [...] We might start by marking those pairs of words which are felt to have the closest relationship." But he makes the following assumption without any justification: "We will also lay down the rule that each word can be marked as a member of only one such pair." Gleason then declares the method of finding the best among all the possible pairings to be "the basic problem of syntax" and he notes him-

self that his method is “haphazard” as his “methodology has not as yet been completely worked out” and lacks precise criteria. We are not far from agreeing with Gleason but we do not think that we need to choose between various satisfactory pairings. For instance, he proposes the following analysis for the NP *the old man who lives there*:

the	old	man	who	lives	there
the	graybeard		who	survives	
the	graybeard		surviving		
survivor					
he					

We think that other analyses like

the	old	man	who	lives	there
he	graybeard		living		there
someone			surviving		
he					

are possible, that they are not in competition, but complementary, and that both (and others) can be exploited to find the structure of this NP.

Today, the definition of 'constituent' seems no longer be a significant subject in contemporary literature in syntax. Even pedagogical books in this framework tend to skip the definition of constituency, for example Haegeman (1991) who simply states that “the words of the sentence are organized hierarchically into bigger units called phrases.”

Commonly proposed tests for constituency include the “stand-alone test”, meaning that the segment can function as an “answer” to a question, the “movement test” including clefting and topicalization, and coordinability, the latter causing the “problems” of coordination of multiple constituents, gapping, and right-node raising.

In phrase structure frameworks, constituents are nothing but a global approach for the extraction of regularities, the only goal being the description of possible orders with few rules. However, it is never actually shown that the proposed phrase structure really is the most efficient way of representing the observed utterances.

We see that the notion of constituency is either not defined at all or in an unsatisfactory way, often based on the notion of one element, the *head*, being linked to another, its *dependent*, modifying it. It is clear that the notion of dependency cannot be defined as a derived notion of constituency, as the definition of the latter presupposes head-daughter relations, making such a definition of dependency circular.

2.3 Intersecting analyses

An interesting result of the vagueness of the definitions of constituency is the fact that different scholars invent different criteria that allow to choose among the possible constituent structures. For example, Jespersen's lexically driven criteria select particle verbs as well as idiomatic expressions. For instance, the sentence (2) is analyzed as “S W O” where W is called a “composite verbal expression” (Jespersen 1937:16)

(2) *She [waits on] us.*

Inversely, Van Valin & Lapolla (1997:26) oppose *core* and *periphery* of every sentence and obtain another unconventional segmentation of (3).

(3) [*John ate the sandwich*] [*in the library*]

Imposing one of these various fragmentations supposes to put forward additional statements (all legitimate) based on different types of information like head-daughter relations (for X-bar approaches), idiomaticity (for Jespersen) or argument structure or information packaging (for VanValin & Lapolla) and serve merely for the elimination of unwanted fragments.

We consider the fact that we find multiple decomposition of an utterance not to be a problem. There is no reason to restrict ourselves to one particular fragmentation as it is done in phrase-structure based approaches. On the contrary, we think that the best way to compute the syntactic structure of an utterance is to consider all its possible fragmentations and this is the idea we want to explore now. Steedman (1985) was certainly one of the first linguists to develop a formal grammar that allows various groupings of words. This work and later articles by Steedman corroborated the multi-fragment approach to syntactic structure.

3 Fragmentation and connection

3.1 Fragments

We will relax the notion of syntactic constituent. We call *fragment* of an utterance any of its subparts which is a linguistically acceptable phrase with the same semantic contribution as in the initial utterance. Let us take an example :

(4) *Peter wants to read the book.*

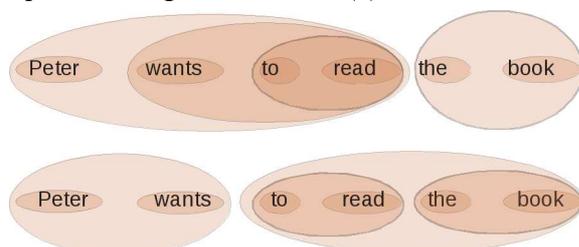
We consider the acceptable fragments of (4) to be: *Peter, wants, to, read, the, book, Peter wants, wants to, to read, the book, Peter wants to, wants to read, read the book, Peter wants to read, to read the book, wants to read the book.*

We will not justify this list of fragments at this point. We just say for the moment that *wants to*

read, just like *waits on*, fulfills all the commonly considered criteria of a constituent: It is a “significant element”, “functions as a unit” and can be replaced by a single word (*reads*). In the same way, *Peter wants* could be a perfect utterance. Probably the most unnatural fragment of (4) is the VP *wants to read the book*, traditionally considered as a major constituent in a phrase structure analysis.

3.2 Fragmentations

A *fragmentation (tree)* of an utterance U is a recursive partition of U into acceptable fragments. The following figure shows two of the various possible fragmentations of (4):



More formally, if X is set of minimal units (for instance the words of (4)), *fragments* are subsets of X and a *fragmentation* F is a subset of the powerset of X ($F \subset P(X)$) such that:

1. for every $f_1, f_2 \in F$, either $f_1 \subseteq f_2$, $f_2 \subseteq f_1$, or $f_1 \cap f_2 = \emptyset$;
2. Each fragment is partitioned by its immediate sub-fragments.

A fragmentation whose fragments are constituents is nothing else than a constituency tree.

A fragmentation is *binary* if every fragment is partitioned into 0 or 2 fragments.

3.3 Connection structure and fragmentation hypergraph

We consider that each segmentation of a fragment in two pieces induces a *connection* between these two pieces.¹ This allows us to define graphs on the fragments of a set X . An *hypergraph* H on X is a triplet (X, F, φ) where $F \subset$

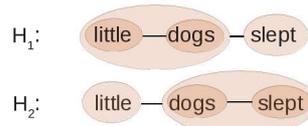
¹ The restriction of the connections to binary partitions can be traced back all the way to Becker (1827:469), who claims that “every organic combination within language consists of no more than two members.” (*Jede organische Zusammensetzung in der Sprache besteht aus nicht mehr als zwei Gliedern*). Although we have not encountered irreducible fragments of three or more elements in any linguistic phenomena we looked into, this cannot be *a priori* excluded. It would mean that we encountered a fragment XYZ where no combination of any two elements forms a fragment, i.e. is autonomizable in any without the third element. Our formal definition does not exclude this possibility at any point and a connection can in theory be, for example, ternary.

$P(X)$ and φ is a graph on F . If F is only composed of singletons, H corresponds to an ordinary graph on X . For each binary fragmentation F on X , we will define a *fragmentation hypergraph* $H = (X, F, \varphi)$ by introducing a connection between every couple of fragments which partitions another fragment.

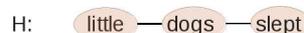
Let us illustrate this with an example:

(5) *Little dogs slept.*

There are two natural fragmentations of (5) whose corresponding hypergraphs are:²



As you can see, these two hypergraphs tell us that *little* is connected to *dogs* and *dogs* to *slept*. H_2 also show a connection between *little* and *dogs slept*, but in some sense, this is just a rough version of the connection between *little* and *dogs* in H_1 . The same observation holds for the connection between *little dogs* and *slept* in H_1 , which correspond to the connection between *dogs* and *slept* in H_2 . In other words, the two hypergraphs contains the same connections (in more or less precise versions). We can thus construct a finer-grained hypergraph H with the finest version of each connection:



We will call this hypergraph (which is equivalent to a graph on the words in this case) the *connection structure* of the utterance. We will now see how to define the connection structure in the general case.

3.4 A complete partial order on hypergraphs

We saw with our example that the connection structure is a finer-grained version of the different fragmentation hypergraphs of the utterance. So we propose to define the connection structure as the *infimum*³ of the fragmentation hypergraphs for a natural order of fineness.

A connection $f - g$ is *finer* than a connection $f' - g'$ if $f \subseteq f'$ and $g \subseteq g'$. For instance the con-

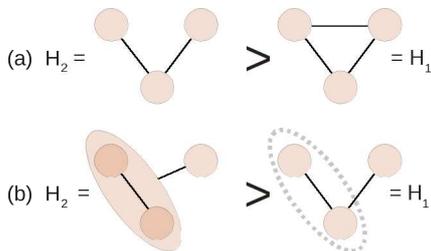
² It is possible that, for most readers, H_1 seems to be more natural than H_2 . From our point of view, it is not the case: *dogs slept* is a fragment as valid as *little dogs*. See nevertheless footnote 6.

³ If \leq is a partial order on X and A is a subset of X , a *lower bound* of A is an element b in X such that $b \leq x$ for each x in A . The *infimum* of A , noted $\wedge A$, is the greatest lower bound of A . A partial order for which every subset has an infimum is said to be *complete*. (As a classical example, consider the infimum for the divisibility on natural integers, which is the greatest common divisor: $9 \wedge 12 = 3$).

nection $[dogs]-[slept]$ is finer than the connection $[little\ dogs]-[slept]$. A connection is *minimal* when it cannot refine.

Intuitively, the *fineness order*, henceforth noted \leq , represents the precision of the hypergraph, ie. $H_1 \leq H_2$ if H_1 is a finer-grained analysis than H_2 . A hypergraph H_1 is *finer* than a hypergraph H_2 (that is $H_1 \leq H_2$) if every connection in H_2 has a finer connection in H_1 .

In other words, H_1 must have more connections than H_2 , but H_1 can have some connections pointing to a smaller fragment than in H_2 , and in this case the bigger fragment can be suppressed in H_1 (if it carries no other connections) and H_1 can have less fragments than H_2 . This can be resumed by the following schemata:



In case (a), H_1 is finer because it has one connection more. In case (b), H_1 is finer because it has a finer-grained connection and the dotted fragment can be suppressed. It is suppressed when it carries no further connection.

We think that this partial order on hypergraphs is *complete* (see note 3). We have not proven this claim but it appears to be true on all the configurations we have investigated.

If we have an utterance U and linguistic criteria characterizing the acceptable fragments of U , we define the *connection structure* of U as the infimum of its all fragmentation hypergraphs.

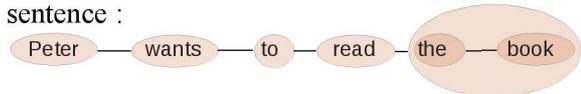
3.5 Constructing the connection structure

Our definition could appear as being slightly complicated. In practice, it is very easy to build the connection graph of an utterance as soon as you have decided what the acceptable fragments of an utterance are. Indeed, because the fineness order on hypergraphs is complete, you can begin with any fragmentation and refine its connections until you cannot refine them any further. The connection structure is obtained when all the connections are minimal. The completeness ensures, due to the uniqueness of the greatest lower bound, that you obtain always the same structure. The only problem stems from cycles and sometimes connections must be added (see 3.7). Let us see what happens with example (4). Suppose the first step of your fragmentation is :

$$f_1 = \textit{Peter wants to}$$

$$f_2 = \textit{read the book}$$

This means that you have a connection between f_1 and f_2 that will correspond in the final connection structure to a link between two minimal fragments, possibly words. Now, you want to discover these minimal fragments. For that you are looking for the minimal fragment g overlapping both f_1 and f_2 : $g = \textit{to read}$. It is fragmentable into *to* and *read*. Therefore the connection between f_1 and f_2 is finally a connection between *to* and *read*. It now remains to calculate the connection structures of f_1 and f_2 in order to obtain the complete connection structure of the whole sentence :



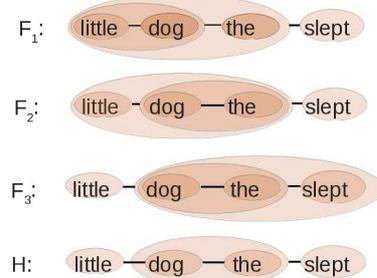
Connection structure of (4)

3.6 Irreducible fragment

The connection structure of (4) is not equivalent to a graph on its words because some fragments are irreducible. An *irreducible fragment* is a fragment bearing connections which cannot be attributed to one of its parts. For instance, *the book* in (4) is irreducible because there is no fragment overlapping *the book* and including only *the* or only *book* (neither *read the* nor *read book* are acceptable).

(6) *The little dog slept.*

Example (6) poses the same problem, because *little* can be connected to *dog* (*little dog* is acceptable), but *slept* must be connected to *the dog* and cannot be refined (neither *dog slept* or *the slept* is acceptable). One easily verifies that (6) has the fragmentation hypergraphs F_1 , F_2 , and F_3 and the connection graph H (which is their infimum). Note that the fragmentation *the dog* persists in the final connection graph H because it carries the link with *slept* but *little* is connected directly to *dog* and not to the whole fragmentation *the dog*.



Connection structure of (6): $H = F_1 \wedge F_2 \wedge F_3$

Irreducible fragments are quite common with grammatical words. We have seen the case of determiners but conjunctions, prepositions, or relat-

ive pronouns can also cause irreducible fragments:

(7) *I think [that [Peter slept]]*

(8) *Pierre parle [à Marie]*

Peter speaks [to Mary]

(9) *[the (old) man] [who lives] there*

3.7 Cycles

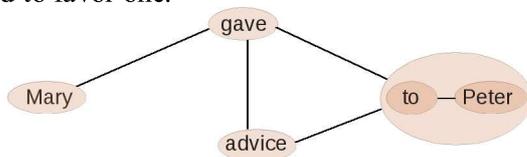
Usually the connection graph is acyclic (and could be transformed into a tree by choosing a node as the root, as we have shown for example (5). But we can have a *cycle* when a fragment XYZ can be fragmented into XY+Z, YZ+X, and XZ+Y. This can happen in examples like :

(10) *Mary gave advice to Peter.*

(11) *I saw him yesterday at school.*

(12) *the rise of nationalism in Catalonia*

In (10), *gave advice*, *gave to Peter*, and *advice to Peter* are all acceptable. We encounter a similar configuration in (11) with *saw yesterday*, *saw at school*, and *yesterday at school (It was yesterday at school that I saw him)*. In (12), *in Catalonia* can be connected both with *nationalism* and *the rise* and there is no perceptible change of meaning. We can suppose that the hearer of these sentences constructs both connections and does not need to favor one.⁴



Cyclic connection graph for (10)⁵

3.8 Connection structures and fragments

We have seen that the connection structure is entirely defined from the set of fragments. Conversely the set of fragments can be reconstructed from the connection graph. Every initial fragment can be obtained by cutting connections in

⁴ The fact that we cannot always obtain a tree structure due to irreducible fragment and cycle suggests that we could add weights on fragments indicating that a fragment (or a fragmentation) is more likely than another. We do not pursue this idea here, but we think that *weighted connection graphs* are certainly cognitively motivated linguistic representations.

Note also that the preferred fragmentation is not necessary the constituent structure. For instance, the most natural segmentation of (i) is just before the relative clause, which functions as a second assertion in this example and can be preceded by a major prosodic break (Deulofeu *et al.* 2010).

(i) *He ran into a girl, who just after entered in the shop.*

⁵ The irreducibility of *to Peter* is conditioned by the given definition of fragments. If we considered relativization as a criteria for fragments, the possibilities of preposition stranding in English may induce the possibility to affirm that *gave* and *advice* are directly linked to the preposition *to*.

the structure and keeping the segment of the utterance corresponding to continuous pieces of the connection structure.

For instance in the connection structure of (4) cutting the connections between *to* and *read*, gives the segment *read the book*. But the segment *read the* cannot be obtained because even when cutting the connection between *the* and *book*, *read* remains connected to the entire group *the book*.

4 Discourse, morphology, semantics

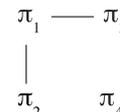
Dependency structures are usually known to describe the syntactic structures of sentences, i.e the organization of the sentence's words. In the next sections, we will give a precise definition of fragments for surface syntax in order to obtain a linguistically motivated connection structure and to transform it into a dependency tree. Let us now at first apply our methodology to construct connection structures for discourse, morphology, and the syntax-semantics interface.

4.1 Discourse

Nothing in our definition of connection graphs is specific to syntax. We obtain syntactic structures if we limit our maximal fragment to be sentences and our minimal fragments to be words. But if we change these constraints and begin with a whole text and take “discourse units” as minimal fragments, we obtain a discourse connection graph. This strategy can be applied to define discourse relations and discourse structures such as RST or SDRT. Of course, to obtain linguistically motivated structures, we need to define what is an acceptable sub-text of a text (generally it means to preserve coherency and cohesion).

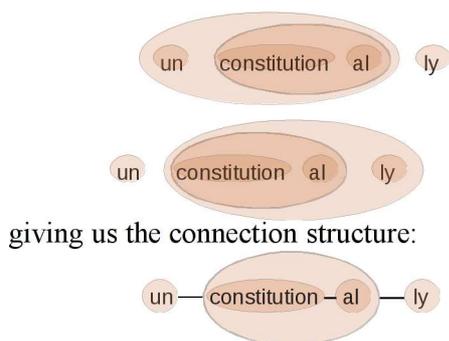
(13) (π_1) *A man walked in.* (π_2) *He sported a hat.*
 (π_3) *Then a woman walked in.* (π_4) *She wore a coat.* (Asher & Pogodalla 2010)

We have the fragments $\pi_1\pi_2$, $\pi_1\pi_3$, $\pi_3\pi_4$ but we don't have $\pi_2\pi_3$ nor $\pi_1\pi_4$. This gives us the following connection graph:



4.2 Morphology

On the other side, we can fragment words into morphemes. To define the acceptable fragmentations of a word, we need linguistic criteria like the commutation test. As an example for constructional morphology consider the word “*unconstitutionally*”. The two possible fragmentations are:



giving us the connection structure:

4.3 Deep Syntax

The *deep syntactic representation* is the central structure of the semantics-syntax interface (Mel'čuk 1988, Kahane 2009). If we take compositionality as a condition for fragmentation, we obtain a structure that resembles Mel'čuk's deep syntactic structure. In other words, idioms must not be fragmented and semantically empty grammatical words are not considered as fragments.

(14) *Pierre donne du fil à retordre à ses parents.*
lit. Peter gives thread to twist to his parents
'Peter has become a major irritant to his parents'



5 Fragmentations for surface syntax

5.1 Criteria for syntactic fragments

The connection structure we obtain completely depends on the definition of acceptable fragments. We are now interested in the linguistic criteria we need in order to obtain a connection structure corresponding to a usual surface syntactic structure. As a matter of fact, these criteria are more or less the criteria usually proposed for defining constituents. A *surface syntactic fragment* of an utterance U:

- is a subpart of U (in its original order),
- is a linguistic sign and its meaning is the same when it is taken in isolation and when it is part of U,⁶
- can stand alone (for example as an answer of a question),⁷

⁶ This condition has to be relaxed for the analysis of idiomatic expressions as they are precisely characterized by their semantic non-compositionality. The fragments are in this case the elements that appear autonomizable in the paradigm of parallel non-idiomatic sentences.

⁷ Mel'čuk (1988, 2011:130-132) proposes a definition of two-word fragments. Rather than the stand alone criterion, he propose that a fragment must be a prosodic unit. This is a less restrictive criterion, because the possibility to stand alone supposes to be a speech turn and therefore to be a prosodic unit. For instance *little dog* can never be a prosodic unit in *the little dog* but it is a prosodic unit when it stands

- belongs to a distributional class (and can for instance be replaced by a single word).

Mel'čuk (2006) proposes, in his definition of wordforms, to weaken the stand-alone property (or autonomizability). For instance in (6), *the* or *slept* are not autonomizable, but they can be captured by subtraction of two autonomizable fragments: *slept* = *Peter slept* \ *Peter*, *the* = *the dog* \ *dog*.⁸ We call such fragments *weakly autonomizable*.⁹

Of course, even if our approach resolves most of the problems arising when trying to directly define constituents, some problems remain. For instance, if you consider the French noun phrase *le petit chien* 'the little dog', the three fragments *le chien*, *petit chien*, and *le petit* 'the little one' are acceptable. Eliminating the last fragment *le petit* supposes to put forward non trivial arguments: *le petit*, when it stands alone, is an NP (it commutes with NPs) but it cannot commute with NPs like for example *la fille* 'the girl' in *le petit chien* as **la fille chien* "the girl dog" is ungrammatical. Many exciting questions posed by other phenomena like coordination or extraction cannot be investigated here for lack of space.

5.2 Granularity of the fragmentation

Syntactic structures can differ in the minimal units. Most of the authors consider that the wordforms are the basic units of dependency structure, but some authors propose to consider dependencies only between chunks and others between lexemes and grammatical morphemes. The following figure shows representations of various granularity for the same sentence (15).

(15) *A guy has talked to him.*

Tree A is depicting an analysis in chunks (Vergne 1990), Tree B in words, Tree D in lexemes and inflectional morphemes (and can be

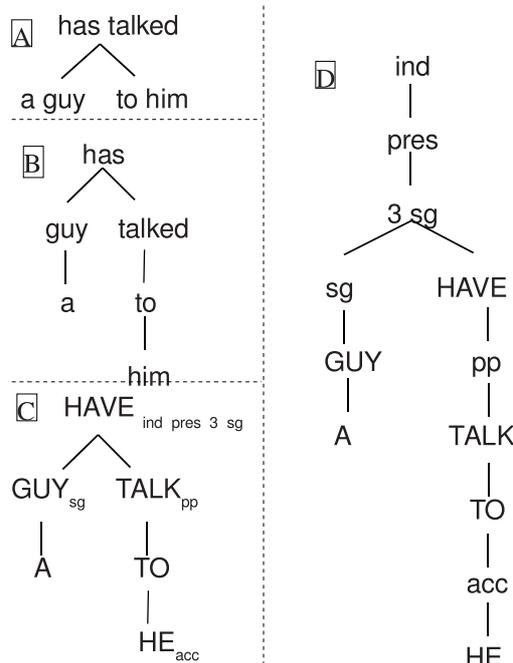
alone. We think that this criterion is interesting, but not easy to use because the delimitation of prosodic units can be very controversial and seems to be a gradual notion. Note also that clitics can form prosodic units which are unacceptable fragments in our sense, like in:

- (i) *the king* | *of England's* | *grandmother*
(ii) *Je crois* | *qu'hier* | *il n'est pas venu*
'I think | that yesterday | he didn't come'

⁸ Note that singular bare noun like *dog* are not easily autonomizable in English, but they can for instance appear in titles.

⁹ Some complications arise with examples like Fr. *il dort* 'he slept'. Neither *il* (a clitic whose strong form *lui* must be used in isolation), nor *dort* are autonomizable. But if we consider the whole distributional class of the element which can commute with *il* in this position, containing for example *Peter*, we can consider *il* to be autonomizable by generalization over the distributional class.

compared to an X-bar structure with an IP, governed by agreement and tense). The tree C (corresponding to the surface syntactic structure of Mel'čuk 1988) can be understood as an under-specified representation of D.



These various representations can be captured by our methods. The only problem is to impose appropriate criteria to define what we accept as minimal fragments. For instance, trees C and D are obtained if we accept parts of words which commute freely to be “syntactic” fragments (Kahane 2009). Conversely, we obtain tree A if we only accept strongly autonomizable fragments.

6 Heads and dependencies

6.1 Defining head and dependency

Most of the syntactic theories (if not all) suppose that the syntactic structure is hierarchized. This means that connections are directed. A directed connection is called a *dependency*. For a dependency from A to B, A is called the *governor* of B, B, the *dependent* of A, and A, the *head* of the fragment AB.¹⁰ The introduction of the term “head” into syntax is commonly attributed to Henry Sweet (1891-96, I:16, sections 40 and 41): “The most general relation between words in sentences from a logical point of view is that of

¹⁰Dependency relation are sometimes called head-daughter relations in phrase structure frameworks. Note the distinction between *head* and *governor*. For a fragment *f*, the governor of *f* is necessary outside *f*, while the head of *f* is inside *f*. The two notion are linked by the fact that the governor *x* of *f* is the head of the upper fragment composed of the union of *f* and *x*.

adjunct-word and **head-word**, or, as we may also express it, of **modifier** and **modified**. [...] The distinction between adjunct-word and head-word is only a relative one : the same word may be a head-word in one sentence or context, and an adjunct-word in another, and the same word may even be a head-word and an adjunct-word at the same time. Thus in *he is very strong*, *strong* is an adjunct-word to *he*, and at the same time head-word to the adjunct-word *very*, which, again, may itself be a head-word, as in *he is not very strong*.”

Criteria for the recognition of the direction of relations between words have been proposed by Bloomfield (1933), Zwicky (1985), Garde (1977), or Mel'čuk (1988). The most common criterion is that the head of a constituent is the word controlling its distribution, which is the word that is most the sensitive to a change in its context. But for any fragment, its distribution does not depend only on its head (and, as we have said in the introduction, constituents cannot easily be defined without using the notion of head). As an example, consider the fragment *little dogs* in (16):

(16) *Very little dogs slept.*

As *little* is connected to *very* and *dogs* to *slept*, *little dogs* does not have the distribution of *dogs* nor of *little* in (16) as *very dogs slept* and *very little slept* are both unacceptable. Determining the head of the fragment *little dogs* (i.e. the direction of the relation between *little* and *dogs*) is equivalent to the identification of the governor of this fragment (between *very* and *slept*). But, as soon as we have identified the governor of the fragment, the head of the fragment is simply the word of the fragment which is connected to the governor, that is the main word outside the fragment. For example, in (16), the identification of *slept* as the governor of the fragment *little dogs* also chooses *dogs* as the head of *little dogs*.

Problems occur only if we are dealing with an irreducible fragment like the determiner-noun connection.¹¹ To sum up: In order to direct the

¹¹Various criteria have been proposed in favor of considering either the noun or the determiner as the head of this connection, in particular in the generative framework (Principles and Parameters, Chomsky (1981), remains with NP, and, starting with Abney (1986), DP is preferred). It seems that the question is triggered by the assumption that there has to be one correct directionality of this relation, in other words that the syntactic analysis is a (phrase structure) tree. This overly simple assumption leads to a debate whose theoretical implications do not reach far as any DP analysis has an isomorphic NP analysis. The NP/DP debate was triggered by the observation of a parallelism in the relation between the lexical part of a verb and its inflection (reflec-

connections and to define a dependency structure for a sentence, it is central to define the head of the whole sentence (and to resolve the case of irreducible fragments if we want a dependency tree). We consider that the head of the sentence is the main finite verb, because it bears most of the illocutionary marks: Interrogation, negation, and mood morphemes are linked to the main finite verb. In English, interrogation changes the verbal form, and in French, interrogation, negation, or mood can be marked by adding clitics or inflectional morphemes on the finite verb even if it is an auxiliary verb.

(17) a. *Did very little dogs sleep?*

b. *Pierre a-t-il dormi?*

lit. Peter has-he slept? ‘Did Peter sleep?’

c. *Pierre n'a pas dormi.*

lit. Peter neg has neg slept ‘Peter didn't sleep’

d. *Pierre aurait dormi.*

lit. Peter have-COND slept?

‘Peter would have slept’

Once the head of the sentence has been determined, most of the connections can be directed by a top down strategy. Consequently the main criterion to determine the head of a fragment f is to search if one of the words of f can form a fragment with the possible governors of f , that is if one of the words of f can be connected with the possible governors of f . If not, we are confronted with an irreducible fragment, and other criteria must be used, which we cannot discuss here (see Mel'čuk 1988, 2011).¹² Nevertheless, it is well known that in many cases, the head is difficult to find (Bloomfield called such configurations *exocentric*). It could be advocated not to attempt to direct the connections and to settle with an only *partially directed connection structure*.¹³

6.2 Refining the dependency structure

Even when the connection structure is completely directed, the resulting dependency structure is not necessary a tree due to irreducible fragments and cycles. We can use two principles to refine the dependency structure and to get closer to a dependency tree. The fineness order

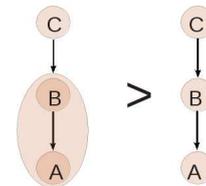
ted by the opposition between IP and VP in the generative framework). This carries over to dependency syntax: The analysis D of sentence (15) captures the intuition that the inflection steers the passive valency of a verb form.

¹²Conversely, whenever the fragmentation tests do not give clear results on whether or not a connection must be established, criteria used to determine the head can be helpful to confirm the validity of the connection.

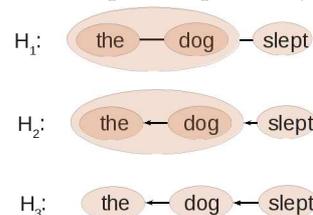
¹³Equally, the problem of PP attachment in parsing is certainly partially based on true ambiguities, but in many cases, it is an artificial problem of finding a tree structure where the human mind sees multiple connections, like for instance

on hypergraphs will be prolonged for directed hypergraph in accordance with these principles.

The first principle consists of avoiding double government: if C governs AB and B is the head of AB, then the dependency from C to AB can be replaced by a dependency from C to B (if $[A \leftarrow B] \leftarrow C$, then $A \leftarrow B \leftarrow C$). In other words, the directed hypergraph with the connection $B \leftarrow C$ is finer than the hypergraph with the connection $[AB] \leftarrow C$.



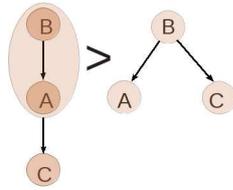
Suppose, for instance, that for the sentence (1) *The dog slept*, we obtained the connection graph H_1 below. We can then add directions: The head principle easily gives the link from *slept* to the rest of the sentence, and some additional criteria may direct the connection between *dog* and *the* to give us H_2 . We can now carry over this directionality to a complete dependency graph H_3 .



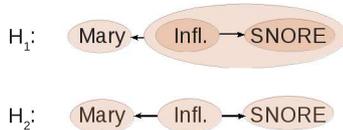
Inversely, the second principle consist of avoiding the creation of unacceptable projections: if C depends on AB and B is the head of AB, then the dependency from AB to C can be replaced by a dependency from B to C (if $[A \leftarrow B] \rightarrow C$, then $A \leftarrow B \rightarrow C$). In other words, the directed hypergraph with the connection $B \rightarrow C$ is finer than the hypergraph with the connection $[AB] \rightarrow C$.¹⁴

in *He reads a book about syntax* or in the examples (10) to (12). We can assume that a statistical parser will give better results when trained on a corpus that uses the (circular) graph structure, reserving the simple tree structures for the semantically relevant PP attachments.

¹⁴The two principles could be generalized into only one: if C is connected to AB and B is the head of AB, then the connection between AB and C can be replaced by a connection between B and C (if $[A \leftarrow B] - C$, then $A \leftarrow B - C$). Nevertheless we think that the two principles are different and that the second one is less motivated. For instance, *the most famous of the world* can be analyzed in $[[the\ most] \leftarrow [famous]] \rightarrow [of\ the\ world]$ and neither *famous of the world* or *the most of the world* are acceptable, but we think that $[of\ the\ world]$ is rather selected by the superlative marker *the most* rather than by the adjective *famous* (because for any adjective X we have *the most X of the world*). The problem can be also solved by declaring *the most of the world* acceptable based on previous more general arguments.



For example, in the sentence *Mary snored*, based on the observation that the distribution of the sentence depends on the inflection of the verb, we decide to direct the relation between the inflection and the lexical part of the verb *snored* as $\text{Infl.} \rightarrow \text{SNORE}$. This implies, following Principle 2, that the subject depends on the inflection, and not on the lexical part of the verb. This corresponds to the observation that other, non-finite forms of the verb cannot fill the subject slot of the verbal valency.



7 Constituency

We saw in section 3.8 that any fragmentation can be recovered from the connection structure. As soon as the connections have been directed, some fragmentations can be favored and constituent structures can be defined.

Let us consider nodes A and B in a dependency structure. A *dominates* B if $A = B$ or if there is a path from A to B starting with a dependency whose governor is A. The fragment of elements dominated by A is called the *maximal projection* of A. Maximal projections are major constituents (XPs in X-bar syntax). The maximal projection of A can be fragmented into $\{A\}$ and the maximal projections of its dependents. This fragmentation gives us a flat constituent structure (with possibly discontinuous constituents).

Partial projections of A are obtained by considering only a part of the dependencies governed by A. By defining an order on the dependency of each node (for instance by deciding that the subject is more “external” than the object), we can privilege some partial projections and obtain our favorite binary fragmentation equivalent to the phrase structure trees we prefer. In other words, a phrase structure for a given utterance is just one of the possible fragmentations and this fragmentation can only be identified if the notion of *head* is considered.

We can thus say that phrase structure contains a definition of dependency at its very base, a fact that already appears in Bloomfield's work, who

spends much more time on defining head-daughter relations than on the notion of constituency. Jackendoff's X-bar theory is based on a head-centered definition of constituency, as each XP contains an X being the (direct or indirect) governor of the other elements of XP.

If we accept to mix criteria for identifying fragments and heads, it is possible to directly define a constituent structure without considering all the fragmentations. The strategy is recursive and top-down (beginning with the whole sentence at first constituent); each step consists of first identifying the head of the constituent we want to analyze and then looking at the biggest fragments of the utterance without its head: These biggest fragments are constituents.¹⁵

8 Conclusion

We have shown that it is possible to formally define a syntactic structure solely on the basis of fragmentations of an utterance. The definition of fragments does not have to keep the resulting constituent structure in mind, but can be based on simple observable criteria like different forms of autonomizability. Even (and especially) if we obtain intersecting fragmentations, we can obtain a connection graph. This operation can be applied to any type of utterance, yielding connections from the morphological to the discourse level. This delegates the search for the head of a fragment to a secondary optional operation. It is again possible to apply the known criteria for heads only when they provide clear-cut answers, leaving us with partially unresolved connections, and thus with a hypergraph, and not necessarily a tree structure. It is possible, and even frequent, that the syntactic structure is a tree, but our definition does not presuppose that it must be one. This two step definition (connection and directionality) allows for a more coherent definition of dependency as well as constituency avoiding the commonly encountered circularities. It finds *connection* as a primary notion, preliminary to constituency and dependency.

¹⁵ If the head of the constituent is a finite verb, clefting can be a useful test for characterizing sub-constituents. But clefting can only capture some constituents and only if the head of the constituent has been identified and is a finite verb. As noted by Croft (2001), such constructions can only be used to characterize the constituents once we have defined them. We know that constructions like clefting select constituents because we were able to independently define constituents with other techniques. We cannot inversely define constituents by use of such language-specific constructions.

Another interesting feature of our approach is not to presuppose a segmentation of a sentence into words and even not suppose the existence of words as an indispensable notion.

In this paper we could explore neither the concrete applicability of our approach to other languages nor the interesting interaction of this new definition of dependency with recent advances in the analysis of coordination in a dependency based approach, like the notion of pile put forward in Gerdes & Kahane (2009). It also remains to be shown that the order on hypergraphs is really complete, i.e. that we can actually always compute a greatest connection graph refining any set of fragmentation hypergraphs. We also leave it to further research to explore the inclusion of weights on the connection which could replace the binary choice of presence or absence of a connection.

Acknowledgments

We would like to thank Igor Mel'cuk, Federico Sangati, and our three anonymous reviewers.

References

- Steven Abney. 1986. *The English Noun Phrase in its Sentential Aspect*. Unpublished Ph.D., MIT.
- Nicholas Asher, Sylvain Pogodalla. 2010. "SDRT and Continuation Semantics", *Logic and Engineering of Natural Language Semantics* 7 (LENLS VII).
- Karl Ferdinand Becker. 1841 [1827]. *Organismus der Sprache*. 2nd edition. Verlag von G.F. Kettembeil, Frankfurt am Main.
- Leonard Bloomfield. 1933. *Language*. Allen & Unwin, New York.
- Rens Bod. 1998, *Beyond grammar: an experience-based theory of language*. Stanford, CA: CSLI Publications.
- Andrew Carnie. 2011. *Modern Syntax: A Coursebook*. Cambridge University Press.
- Noam Chomsky. 1981. *Lectures On Government and Binding*. Foris, Dordrecht.
- Noam Chomsky. 1986. *New horizons in the study of language and mind*, Cambridge University Press.
- William Croft. 2001. *Radical construction grammar: syntactic theory in typological perspective*. Oxford University Press.
- José Deulofeu, Lucie Dufort, Kim Gerdes, Sylvain Kahane, Paola Pietrandrea. 2010. "Depends on what the French say", *The Fourth Linguistic Annotation Workshop (LAW IV)*.
- Paul Garde. 1977. "Ordre linéaire et dépendance syntaxique : contribution à une typologie", *Bull. Soc. Ling. Paris*, 72:1, 1-26.
- Aleksej V. Gladkij. 1966. *Leckii po matematicheskoi lingvistike dlja studentov NGU*, Novosibirsk (French translation: *Leçons de linguistique mathématique*, fasc. 1, 1970, Paris, Dunod)
- Henry A. Gleason. 1955. *An Introduction to Descriptive Linguistics*. New York: Holt, Rinehart & Winston, 503 pp. Revised edition 1961.
- Kim Gerdes, Sylvain Kahane. 2009. "Speaking in piles: Paradigmatic annotation of a French spoken corpus", *Corpus Linguistics 2009*, Liverpool.
- Otto Jespersen. 1937. *Analytic syntax*. Copenhagen.
- Yves Lecerf. 1960. "Programme des conflits, module des conflits", *Bulletin bimestriel de l'ATALA*, 4,5.
- Liliane M. V. Haegeman. 1991. *Introduction to Government and Binding Theory*. Blackwell Publishers
- Richard Hudson. 1994. "Discontinuous phrases in dependency grammars", *UCL Working Papers in Linguistics*, 6.
- Sylvain Kahane. 1997. "Bubble trees and syntactic representations", *MOL5*, Saarbrücken, 70-76.
- Sylvain Kahane. 2009. "Defining the Deep Syntactic Structure: How the signifying units combine", *MTT 2009*, Montreal.
- Igor Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. The SUNY Press, Albany, N.Y.
- Igor Mel'čuk. 2006. *Aspects of the Theory of Morphology*. de Gruyter, Berlin, New York.
- Igor Mel'čuk. 2011. "Dependency in language", *Proceedings of Dependency Linguistics 2011*, Barcelona.
- Klaus Schubert. 1987. *Metataxis: Contrastive dependency syntax for machine translation*. <http://www.mt-archive.info/Schubert-1987.pdf>
- Henry Sweet. 1891-1896. *A New English Grammar*, 2 vols. Clarendon Press. Oxford.
- Willard Quine. 1986. "Reply to Gilbert H. Harman." In E. Hahn and P.A. Schilpp, eds., *The Philosophy of W.V. Quine*. La Salle, Open Court.
- Mark Steedman. 1985. "Dependency and coordination in the grammar of Dutch and English", *Language*, 61:3, 525-568.
- Lucien Tesnière. 1959. *Éléments de syntaxe structurale*. Klincksieck, Paris.
- Arnold M. Zwicky. 1985. "Heads", *Journal of Linguistics*, 21: 1-29.

Coordination of verbal dependents in Old French: Coordination as a specified juxtaposition or apposition

Nicolas Mazziotta

Universität Stuttgart, Institut für Linguistik/Romanistik

nicolas.mazziotta@ulg.ac.be

Abstract

Scholars have proposed many different models to describe coordination of verbal dependents. We give a brief presentation of the most common ways to deal with this construction from a general point of view. Then, we evaluate the adequacy of the models using data from Old French. In this particular language, coordination is a more elaborated form of juxtaposition and apposition, which differs only at the semantic level. For this reason, the coordinating conjunction has to be considered as a dependent of the following conjunct.

Introduction

Our purpose is to present an adequate way to describe simple coordination of verbal dependents in Old French (hereafter “OF”) within a dependency framework. We will mainly focus on the question of the hierarchical position of the conjunction.

As far as coordination constructions are concerned, OF is not very different from modern European languages, such as English or modern French. However, some uses of the conjunction *et* in OF would not be possible nowadays. For example, the construction *cel pris et celle somme d’argent* in ex. 1 would be ungrammatical in modern French (or English), because both nouns refer to the same object, and modern French does not allow the coordination of two noun phrases with identical referents.

- (1) *cel pris et celle somme d’argent*
this price and this amount of money
doit li glise Saint-Donis paier a
must the church S-D pay to
mun saingor Wilhame
my sir W.

“Saint Denis church owes this price and

amount of money to Sir W.” (Charter, 1278, 8)

This phenomenon is named *pairs of synonyms* (Fr. *binôme synonymiques*), and the link between this kind of structure and translations in the Middle Ages has often been studied from the perspective of stylistics. The semantic relation between the synonyms varies, and it is generally assumed that pairs of synonyms are used for the sake of clarity (Buridant, 1977; Buridant, 1980). Buridant (1977, 294, our translation) proposes the following definition:

a sequence of two synonyms normally belonging to the same part of speech and sharing the same level in the syntactic hierarchy

We would like to compare this kind of coordination with cases that can be analysed in the same way as modern variants, and to propose an adequate and accurate hierarchy to model them. The focus of our presentation will gradually shift from general considerations about coordination toward specific OF properties.

We begin this paper (section 1) with a review of the main descriptive options that have been used to analyse coordination in a dependency framework. In section 2, we briefly highlight the fact that OF sentences can often be grammatically correct without the use of segmental grammatical devices such as prepositions and conjunctions. In section 3, we survey OF juxtaposition and apposition. We provide evidence that both constructions can be syntactically and semantically complemented by the use of the same conjunction – a process very close to the one called *specification* by Lemaréchal (1997) – thus forming two different kinds of coordination.

1 Coordination in the dependency framework

To begin with, we provide a general overview of models of coordination in the dependency framework. Since the concept of *dependency* varies among theories, we will briefly introduce the different definitions when necessary. We illustrate this section with English translations for the sake of simplicity. We conclude section 1 with a summary of the descriptive options provided by these different models. The appropriate formalism to model OF coordination will be elaborated in the following sections.

1.1 Tesnière's baseline

Lucien Tesnière (1965, ch. 134 sqq.) introduces the concept of *jonction* (we use the translation *junction* hereafter), used to model coordination. Junction is a “horizontal” relation. Words linked in junction are hierarchically equivalent (Tesnière, 1965, ch. 135). This characteristic makes junction very different from *connexion* (fr. *connection*), which represents a governor/dependent “vertical” relation, where the governor (the top node in the stemma) is hierarchically more prominent than the dependent. Dependency as such is never defined by Tesnière, but Garde (1981, 159-160), in the same framework, defines the governor as the word that controls the passive valency of the phrase (the potential it has to be dependent on some external governor).

As a simple example of junction, we can analyse ex. 2: see fig. 1 (Tesnière, 1965, ch. 136, §3).

- (2) *Alfred and Bernard fall* (translation of stemma 248 in Tesnière's book)

As the graphical (bi-dimensional) representation is very important to him, Tesnière adds (we will see in section 3.2 how this compares with the way appositions are handled):

Two joined nodes each retain equivalent vertical connections [i.e. dependency]. As a result, the graphical representation derived from two vertical connections and the junction line will always form a triangle. (Tesnière, 1965, ch. 136, §4, our translation)

Graphically, the conjunction *and* is placed directly on the horizontal line.

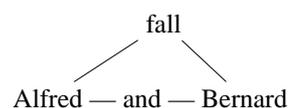


Figure 1: Coordination according to Tesnière

When the conjunction is not present, the representation is exactly the same, except the horizontal line is unbroken. Tesnière's model of coordination multiplies the number of dependents that can be connected to a verb.

1.2 Mel'čuk's unidimensional approach

In the Meaning-Text Theory (MTT) framework, coordination is described as a dependency relation.

MTT has developed a comprehensive list of criteria to find syntactic dependencies, to identify the governor in such a relation, and to classify them (Mel'čuk, 2009, 25-40). To identify a governor, syntactic (with higher priority), morphological and semantic (with lower priority) aspects have to be investigated. Syntactically, the passive valency of the phrase formed by the governor and its dependents should lead us to identify the governor of the phrase. Morphologically, the governor controls agreement between the phrase and its context. Semantically, the governor is a better sample of the referential class denoted by the phrase (e.g.: *a ham sandwich* is a kind of *sandwich*, therefore, *ham* is the dependent).

In fact, Mel'čuk (2009, 50-51) defines coordination from both a semantic and a syntactic perspective: no conjunct semantically depends on the other, but the second conjunct syntactically depends on the first one. Coordination often uses a conjunction and displays the following properties (Mel'čuk, 1988, 41):

1. In a phrase of the form *X and Y*, no element can remain “independent”, i.e., unrelated to any other element. [...]
2. In the phrase *X and Y*, the conjunction cannot be the head, since the distribution of the phrase is determined by its conjuncts and by no means by the conjunction. [...]
3. *X* is the head of the phrase, since the distribution of *X and Y* is that of *X*, and by no means that of *and Y*.

4. In the chunk *and Y*, the conjunction is the head: it determines the distribution of the expression to a greater degree than *Y*. [...]

As a result, the analysis (see fig. 2) forces the dependency between *Bernard* and *fall* to become indirect, which was not the case with Tesnière’s model.

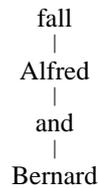


Figure 2: Coordination according to the MTT

According to the MTT, coordination can be direct, and it corresponds to traditional *juxtaposition*.

The author himself acknowledges that his pure-dependency model cannot describe constituent coordination efficiently (Mel’čuk, 2009, 93). For instance, there is no difference in the description of *old men and women* meaning “old men + old women” and “old men + women (either old or not)” (Mel’čuk, 2009, 93). Another limit of the formalism appears in gapping coordinations or valency slot coordinations (non-constituent coordination). There is no way to correctly describe clustering as observed in: *John loves Mary; and Peter, Ann and John gets a letter from Mary and roses from Ann*.

1.3 Two dimensional formalisms

It is a common idea that the limits of the MTT syntactic description of coordination are linked to the unidimensionality of the formalism (generally called *projectivity*). However, as Kahane (1997, § 5.5) states,

Subordination and coordination are two orthogonal linguistic operations and we need a two dimensional formalism to capture this [...]

Bubbles. Kahane (1997) introduces the concept of the *bubble*. Bubbles are formal objects that represent embeddable clusters of nodes. Clustered elements are linked together by a dependency (this concept is defined formally) or an embedding relation. Therefore, coordination bubbles allow the

grouping of sub-bubbles without any dependency relation between them. The advantage of this model is that it can cope with gapping and valency slot coordination, but our main interest is the hierarchical position of the conjunction. In the representation shown in fig. 3,

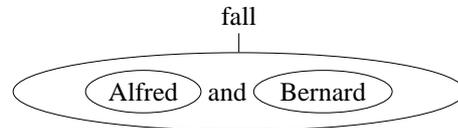


Figure 3: Coordination in a Bubble-tree

it can be seen that the representation leaves the exact hierarchical position of the coordinating conjunction unspecified: it is simply a sibling of the conjuncts. Note that the dependency links the whole bubble to its governor, thus assuming functional equivalence of the conjuncts.

Paradigmatic piles. The so-called *paradigmatic pile* device is aimed at easing transcription and analysis of oral performance, mainly to deal with disfluencies and reformulations. It inherits the ideas of the *grid analysis* (Blanche-Benveniste and Jeanjean, 1987, 167-171). Kahane and Gerdes (2009) argue that the same device can be used to describe coordination and apposition – the same idea already appears in Bilger (1999), but without further formalisation. For instance, the following example presents a disfluency (Kahane and Gerdes, 2009, § 3.2):

- (3) *okay so what what changed your mind*

what and *what ...mind* form some kind of paradigm. Production is indeed interrupted, and one could not reasonably think that both elements are part of the same syntactic structure; as far as reformulation and coordination are concerned,

we consider that a segment *Y* of an utterance piles up with a previous segment *X* if *Y* fills the same syntactic position as *X*. (Kahane and Gerdes, 2009, § 4)

Such an analysis is represented in fig. 4, where curly brackets delimit the pile, and the vertical bar divides the elements of the pile.

Besides, paradigmatic piles can also be used to sketch a coordination relation: the analysis of ex. 2 is shown in fig. 5, where the italicised *and* is called a *pile marker*. It is related to the conjuncts, but their exact dependency is not stated:

okay so { what
| what changed your mind }

Figure 4: Disfluency

{ Alfred
| and Bernard } fall

Figure 5: Coordination in a pile

[...] pile markers like *and* or *or*, usually called coordinating conjunctions, are in a syntagmatic relation only with the conjuncts and do not play any role in the combination of the conjuncts with the context as they can only appear between two conjuncts (Kahane and Gerdes, 2009, § 3.1)

Formally, bubbles and piles can be combined. The resulting formalisation displays three sets of relations: plain syntactic dependencies, in a tree equivalent to Mel'čuk's, orthogonal paradigmatic relations, and pile marking relations (Kahane, forthcoming). As a result, the analysis of ex. 2 is represented in fig. 6, where solid arrows are regular dependencies, the double line expresses the paradigmatic link, and the dashed arrows express the value of the pile marker.

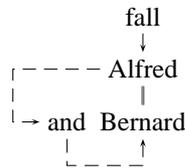


Figure 6: Coordination with tree sets of relations

Word grammar. Word grammar has a mainly semantic definition of dependency: a dependent makes the meaning of its governor more precise (Hudson, 2010, 147).

Following most recent formulations of the word grammar dependency model (Hudson, 2010, 176-181), a coordinating conjunction has no governor and is itself the governor of the conjuncts. These also depend on the verb. Ex. 2 would thus be analysed as in fig. 7.

Another option (Rosta, 2006, 189-191) would be to make the conjunction the dependent of the verb, which would govern each conjunct if there was no coordination (fig. 8).



Figure 7: Coordination according to Hudson

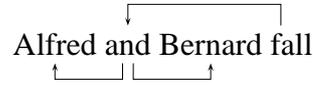


Figure 8: Coordination according to Rosta

1.4 Summary of options

Regarding simple coordination of verbal dependents, differences between models are all linked to the hierarchical position of the conjunction. The coordinating conjunction can depend on:

- the coordination relation (Tesnière, 1965);
- nothing (Hudson, 2010; Kahane, 1997; Kahane and Gerdes, 2009);
- the first conjunct (Mel'čuk, 1988);
- the first conjunct in a parallel set of dependencies (Kahane, forthcoming);
- the verb (Rosta, 2006).

It can govern:

- nothing (Tesnière, 1965);
- [undefined] (Kahane, 1997; Kahane and Gerdes, 2009);
- both conjuncts (Hudson, 2010; Rosta, 2006);
- the following conjunct (Mel'čuk, 1988);
- the following conjunct in a parallel set of dependencies (Kahane, forthcoming).

As far as the concept of *dependency* is concerned, we will retain Mel'čuk's definition hereafter. This first choice compels us to reject Tesnière's description, because a word cannot depend on a relation.

2 Segmental underspecification in OF

OF is the ancestor of Modern French. It can be roughly described as a V2 analytic language. Some remnants of Latin nominal declension remain, but they are often too poor to guarantee the univocity of the form/function relation (Moignet, 1988, 87).

Being a major written language from the 11th century to the 14th century, OF has been well described in several more or less traditional grammars, e.g. Foulet (1968), Moignet (1988), Ménard (1994), Buridant (2000). However, grammars do not investigate the syntactic description of coordination phenomena in detail, and their contribution to the problem is generally limited to a list of coordinating conjunctions and their semantic or discursive values, with the main focus on coordination of clauses or sentences. More useful is the very comprehensive study by Antoine (Antoine, 1958; Antoine, 1962), which examines many aspects of coordination from a diachronic and a synchronic point of view, but lacks a proper syntactic modelisation of the structure. However, it contains many well-classified examples and remains very useful.

We use the concept of *specification* (section 2.1) to show that OF has many “segmentally underspecified” constructions (section 2.2). The adequacy of the models can be evaluated with this property (section 2.3).

2.1 Minimal relation and specification concepts

Following Alain Lemaréchal’s work, we assume that every syntactic relation has an underlying minimal relation (Fr. *relation minimale*) that has hardly any formal mark. Put simply, some words are connected simply by being used together, without the need for grammatical information other than the part-of-speech class they belong to. For instance, using *red* and *book* together will generate an understandable phrase that “works” (Lemaréchal, 1997, esp. 3 and 103). At this “minimal” level, the orientation of the dependency relation is not important.

However, languages tend to add grammatical marks that help to distinguish different functions: prosodic marks, segmental morphemes, etc. The addition of such marks over a minimal relation is called *specification* (Fr. *spécification*) by Lemaréchal (1997, 107-114). Specifications are generally combined in complex context-dependant mark sets. The use of marks make the definition of the relation more precise, and generally allows the governor of a relation to be identified. For example, it is the lexical verb that controls the form of its dependents: most constraints over the dependents are stored in the lexicon.

From a diachronic perspective, specification

may vary for the same dependency relation. For example, it is well known that the Latin subject was marked using the nominative case, while in Modern French, the subject is marked by its position in the clause. Once a specification becomes tightly bound to the way a function is expressed, its use becomes compulsory.

2.2 Segmental underspecification in OF

However, there is never a compulsory segmental mark for every function. Moreover, marks tend to be polyfunctional; e.g.:

- nominal structures expressing the semantic recipient are generally indirect (prepositional specification with *a*), but the preposition can be absent (Moignet, 1988, 296), as in:

(4) *Nos avons donet Warnier une mason*
 “We have given W. a house” (Charter, 1252, 3)

- nominal structures expressing a genitive relation can be specified by the preposition *de*, but this specification is not compulsory when the possessor is a human being, as in *la fille le roi* [“The king’s daughter”] (Moignet, 1988, 94);
- subordination is generally marked by conjunction, but parataxis also exists (Moignet, 1988); see also the extensive study by Glikman (2009).
- even when these prepositions and conjunctions are used, they can have multiple meanings (Moignet, 1988).

Hence we claim, following Mazziotta (2009, 149-150), that OF can be seen as a language in which the syntax relies less on segmental specification than on semantic categories and situational/contextual factors. Consequently, models used to describe OF should not systematically treat segmental specification morphemes as governors.

2.3 Consequences

The segmental underspecification of many structures in OF has a direct impact on the choice of the model best suited to describe the language. Given the fact that grammatical words such as conjunctions and prepositions are, in some cases, optional, grammatical words cannot *always* be considered as governors of prepositional or conjunctive phrases, because these words do not fully

determine the passive valencies of these structures (i.e. the way they combine with a governor), which is the prominent criterion in evaluating directin of dependency (Meřćuk, 2009, 27-28). It is quite probable that many grammatica units are indeed compulsory (Moignet, 1988, 293), but the dependency description of OF is not complete enough to state it firmly in every case. It is better to keep the description at the level of the minimal relation while dependency remains unclear.

Hence, if we want to investigate such questions with respect to the coordinating conjunction, it is important to choose a model in which the hierarchical position of the conjunction remains undefined. At first glance, the bubble-tree and the pile models, as well as a combination of the two, seem a perfect fit, because they do not state dependencies regarding the conjunction.

3 Coordination as a specified juxtaposition or apposition

In this section, we show that there exist two types of coordination. The first must be considered as a special case of juxtaposition (section 3.1). Relying on the structural equivalence between juxtaposition and apposition, we will also demonstrate that the second type of coordination can be seen as a special case of apposition (3.2).

3.1 Specified juxtaposition

Given the possibly underspecified status of coordination, we follow Antoine’s insight, focusing our survey at first on what one might call “implicit” coordination, in order not to assign too important a role to the conjunction initially (Antoine, 1958, 461).

Argument types. Let us first try to define what one may call *juxtaposition* at clause level (not *between* clauses). There may be juxtaposition between dependents of the verb, but what makes juxtaposition different from simultaneous use of different arguments of the same verb?

From a syntactic-semantic perspective, the verb, as a selected lexical unit, has a predetermined set of valency patterns, constraining the semantic role and the morphosyntactic expression of its arguments (Lemaréchal, 1989, 102). For instance, in its prototypical transitive use, the verb *to kill* has a first argument of which the grammatical form is that of a subject (possible agreement with the verb, substitiability with *he*, etc.) and

which expresses the semantic AGENT. *To kill* the second argument has the form of an object and is the semantic PATIENT. One can say that *to kill* can govern two types of arguments combining a specific form to a specific meaning. Only one occurrence of each argument type can occur in the same clause. On the other hand, adjuncts are not subject to such constraints of form, meaning or presence.

For all languages, juxtaposition is the construction that allows speakers to multiply each argument type of one verb or adjuncts. Simultaneously using arguments of different types (such as a subject expressing the agent and an object expressing the patient) is not juxtaposition.

Juxtaposed dependents. Orientations 1, 2, etc. of a verb can thus be duplicated without using any grammatical device:

- (5) *Homes, bestes, sont en repos*
humans animals are in rest
“Humans, animals are resting” (Antoine, 1958, 561, quoting Eneas, 2163)
- (6) *Bien li siet cele ventaille,*
well to him is suited this faceguard
li hiaumes, li escus, la lance
the helmet the shield the spear
“He is well clad with this faceguard, the helmet, the shield, the spear” (Stein et al., 2008, BretTournD, 2202)

The same is true of the adjunct position, which is naturally unbounded.

Specification. From our point of view, the coordinating conjunction that can be used between juxtaposed arguments is a specification device that is *added* to a relation that already exists. In other words, there cannot be a coordination if there is no multiplication of any type of argument. As a result, although the word *et* is present in ex. 7, there is no juxtaposition, and therefore no coordination:

- (7) *Nos oiemes che ke li veritauuele dissent*
we heard what the witnesses said
et par serement
and under oath
“We heard what the witnesses declared under oath indeed” (Charter, 1260, 10)

Although *et* is present, the adjunct *et par serement* is not coordinated, because there is no other juxtaposed adjunct in the clause. Therefore, *et* has to be considered as a mark of specification of the

relation bounding the adjunct to its verbal governor *dissent* (we will not elaborate on the structural position of the preposition *par* here). From a semantic perspective, the word *et* adds emphasis to the adjunct.

If the coordinating conjunction is a specification mark that combines with an *already existing* relation, the conjunction cannot be the governor of the second conjunct, nor can it be a third co-head in a common bubble. If the coordinating conjunction is secondary, Mel'čuk's description presented in 1.2 does not hold for OF.

Moreover, following Mel'čuk's definition of dependency if the conjunction forms a phrase with the second conjunct and is directly linked in a dependency relation with the first one, it should be described as the governor of the second conjunct (Mel'čuk, 2009, 26-27), which cannot be the case. Therefore, there is no dependency relation between the first conjunct and the conjunction, which *must* be described as a dependent of the conjunct following it.

In other words, we also reject the classical assumption that juxtaposition is a coordination from which the conjunction has been deleted (Tesnière, 1965, ch. 137, § 1). This is a matter of *frequency*, rather than of grammatical organisation: specification is more frequent, but it does not mean that it is more basic from a structural point of view. Fig. 9 shows our simplified analysis of ex. 8.

- (8) *Prenez mon escu et ma lance*
 “Take my shield and my spear” (De-fourques and Muret, 1947, Bérout, v. 3586)

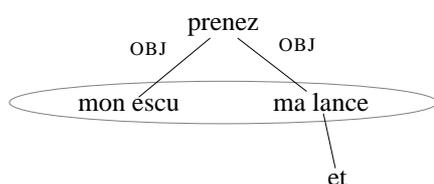


Figure 9: Juxtaposition specification

The coordination relation takes the form of a bubble and the conjunction depends on the second conjunct. The juxtaposition has duplicated the object-PATIENT argument of the verb without changing its valency. Note that the model is not exactly a bubble tree, because dependency relations cannot normally cross the border of a bubble, but the main idea of coordination being an or-

thogonal relation between (groups of) dependents is inherited from this model.

Such model integrates seamlessly polysyndeton (ex. 9):

- (9) *li baisse et le bouche*
 to him DATIVE kisses and the mouth OBJ
et le nes
 and the nose OBJ
 “He kisses him on the mouth and on the nose” (Stein et al., 2008, ElieB, 2599)

Here, the first coordinating conjunction depends on the first conjunct, as shown in fig. 10.

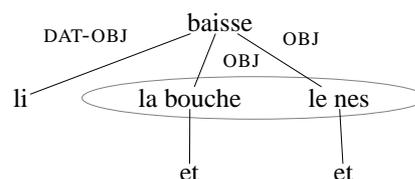


Figure 10: Representation of polysyndeton

Indeed, there are many simple examples of the specified construction in OF. According to our knowledge of this language, and to the texts we have read so far, we have found that juxtaposition is very often specified in the case of a coordination of genuine arguments (which excludes coordination of adjuncts). We believe that in the written language underspecification tends to become rarer over time (a diachronic survey would be necessary). Note that adjuncts are obviously not subject to this emerging constraint.

3.2 Specified apposition

We claim that coordination can also be a specified case of apposition, which is a property of OF but not modern French – Bilger (1999, 263-264), among others, gives no example of specified apposition.

Comparing apposition and juxtaposition. Intuitively, appositions are generally described as simultaneous expressions of the same object; e.g.:

- (10) *Li enemy, li aduersaire dunc*
 the enemy warriors the opponents then
se desrengent
 are restless
 “Then, the foes are restless” (Stein et al., 2008, EdmK, 2065)

- (11) *Tu vouloies ceste angoisse, ceste dolor,*
 You wanted this anguish this pain
ceste painne pour nostre amor [...]
 this mourning for the love of us
 “You wanted to experience this pain for
 our sake” (Stein et al., 2008, PassJonglGP,
 497)
- (12) *Adont m’ arés vous retenu*
 Then me OBJ will have you retained
a vostre ami, a vostre dru
 as your lover as your lover
 “Then, you will hold on to me as your
 lover” (Stein et al., 2008, JacAmArtK,
 1972)

Tesnière has the following insight:

The form of the junction line is identical
 to the form of the apposition line, since
 both are horizontal (Tesnière, 1965,
 ch. 136, § 5, our translation)

But he argues (Tesnière, 1965, ch. 69, §§ 5-6 and
 ch. 139, § 6) that the apposed node, even if it is
 bound by an horizontal line, remains dependent
 upon the node to which it is apposed (the rela-
 tion that unites them is a *connexion*). Underlying
 his argumentation is the assumption that apposi-
 tion is not a clause-level relation: apposed nouns
 are governed by a node that may be an argumen-
 tal dependent. This may be true, but there is a
 major difficulty in determining what is apposed to
 what. Moreover, apposed dependents of the verb
 share the same constraints bound to their function
 (e.g. the use of the preposition *a* in ex. 12).

It is often not possible to decide which apposed
 word would be the governor in an apposition rela-
 tion. As they share the same argument type, ap-
 posed words have the same passive valency, and
 therefore would trigger the same agreement in the
 same context. From a semantic point of view, they
 are lexical synonyms (*enemy/adversaire* in ex. 10
 or *ami/dru* in ex. 12) or they refer to the same ob-
 ject or fact (*angoisse/dolor/paine* in ex. 11). The
 hierarchy remains undefined.

The difference between argumental apposition
 and juxtaposition is only semantic – the fact has
 been highlighted by Blanche-Benveniste and Cad-
 déo (2000) for spoken modern French, and by
 Touratier (2005, 290) in a constituent-based ap-
 proach – as it is a case of *coreference* (Hudson,
 2010, 229-232). Where several dependents refer

to the same object, they are said to be coreferent.
 For instance, a noun and the pronoun replacing it
 are coreferent. Coreference is a major semantic
 characteristic of apposition, distinguishing it from
 juxtaposition: apposed nouns share the same *des-
 ignatum*. Note that subject/verb agreement cannot
 be considered as a reliable grammatical mark of
 the difference between apposition and juxtaposi-
 tion (Foulet, 1968, 201-202).

Specification. The apposition relation can be
 specified by the use of a coordinating conjunction,
 as seen in ex. 1, and in the following excerpt.

- (13) *Poor en ont tuit et esfroi*
 Fear of it have all and fright
 “They are all afraid of it” (Defourques and
 Muret, 1947, Bérroul, 1722)

Since we consider juxtaposition and apposition to
 be syntactically equivalent, our analysis of *paier
 cel pris et celle summe* is shown in fig. 11, where
 the dashed line represents the coreference relation.

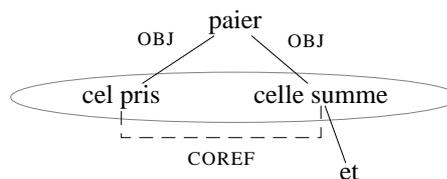


Figure 11: Specified apposition

Contrary to juxtaposition, we suggest (again,
 this should be verified), underspecification has
 generalised in apposition over time. Note that
 modern French can still specify appositions when
 they are not directly dependent on the verb. Thus,
 14 is grammatical (the unique determiner implies
 that there is only one noun phrase), but 15 is not:

- (14) Je vois ma chère et tendre
 “I see my dear and sweet”
 (15) **Je vois ma chère et ma tendre

3.3 Conclusion

As far as verbal dependents of OF are concerned,
 coordination is a form of juxtaposition or appo-
 sition that is specified by the use of a coordinat-
 ing conjunction. The fact that apposition can be
 specified in the same manner as juxtaposition is a
 property of OF that has not survived into modern
 French.

Since both constructions occur without this specification, the coordinating conjunction has to be described as a dependent of the conjunct following it. Of course, this position of the conjunction should be reserved to languages where its presence is not compulsory: where the conjunction is mandatory, it has the position of a governor. However, according to Caterina Mauri (2008, 60), juxtaposition without specification is always possible at clause level, in all languages she has investigated:

Asyndetic constructions consist of the simple juxtaposition of the two SoAs [i.e.: ‘states of affairs’, “hyperonym for the words ‘situation’, ‘event’, ‘process’ and ‘action’” (Mauri, 2008, 32)], and the specific coordination relation existing between them is inferred from the context of communication and from their semantic properties. Asyndesis is always possible and occurs in every language as a more or less stylistically marked strategy.

It means that the dependent position of the conjunction can be generalised in the case of juxtaposition.

Acknowledgements

We would like to thank Thomas M. Rainsford, Julie Glikman, Brigitte Antoine, Sylvain Kahane and Lene Schøsler for proofreading and content suggestions.

References

- Gérald Antoine. 1958. *La coordination en français. Tome I*. D’Artrey, Paris.
- Gérald Antoine. 1962. *La coordination en français. Tome II*. D’Artrey, Paris.
- Mireille Bilger. 1999. Coordination: analyses syntaxiques et annotations. *Recherches sur le français parlé*, 15:255–272.
- Claire Blanche-Benveniste and Sandrine Caddéo. 2000. Préliminaires à une étude de l’apposition dans la langue parlée. *Langue française*, 125(1):60–70.
- Claire Blanche-Benveniste and Colette Jeanjean. 1987. *Le français parlé. Transcription et édition*. Didier érudition, Paris.
- Claude Buridant. 1977. Problèmes méthodologiques dans l’étude des traductions du latin au français au XII^e siècle: le domaine lexical. Les couples de synonymes dans l’histoire de France en français de Charlemagne à Philippe-Auguste. In Danielle Buschinger, editor, *Linguistique et philologie: Application aux textes médiévaux. Actes du colloque des 29 et 30 avril 1977*, Champion, Paris. 293–324.
- Claude Buridant. 1980. Les binômes synonymiques. Esquisse d’une histoire des couples de synonymes du moyen âge au XVII^e siècle. *Bulletin du centre d’analyse du discours*, 4:5–79.
- Claude Buridant. 2000. *Grammaire nouvelle de l’ancien français*. Sedes, Paris.
- Charter. 1252. *1st of March 1252*. Archives de l’État à Liège (Belgium), couvent de Robermont.
- Charter. 1260. *9th of May 1260*. Archives de l’État à Liège (Belgium), couvent du Val-Benoît.
- Charter. 1278. *1st of August 1278*. Archives de l’État à Liège (Belgium), collégiale Saint-Denis.
- L. M. Defourques and E. Muret, editors. 1947. *Béroul. Le roman de Tristan. Poème du XII^e siècle*. Number 12 in CFMA, Paris, Champion, 4th edition.
- Lucien Foulet. 1968. *Petite syntaxe de l’ancien français*, Paris, Champion, 3rd edition.
- Paul Garde. 1981. Des parties du discours, notamment en russe. *Bulletin de la Société de Linguistique de Paris*, 76(1):155–189.
- Julie Glikman. 2009. *Parataxe et Subordination en Ancien Français. Système syntaxique, variantes et variation*. Université Paris Ouest Nanterre La Défense and Universität Potsdam. Thèse de doctorat.
- Richard Hudson. 2010. *An introduction to word grammar*. Cambridge Textbooks in Linguistics. Cambridge University Press, Cambridge.
- Sylvain Kahane and Kim Gerdes. 2009. Speaking in piles. Paradigmatic annotation of a French spoken corpus. In *Proceedings of Corpus Linguistics Conference 2009, Liverpool*, 16 pages.
- Sylvain Kahane. 1997. Bubble trees and syntactic representations. In T. Becker and H.-U. Krieger, editors, *Proceedings of the 5th meeting of Mathematics of Language (MOL 5), Saarbrücken*, 70–76.
- Sylvain Kahane. forthcoming. De l’analyse en grille à la modélisation des entassements. In Sandrine Caddéo, Roubaud Marie-Noëlle, Magali Rouquier, and Frédéric Sabio, editors, *Hommage à Claire Blanche-Benveniste*. Aix-en-Provence, Publications de l’Université de Provence.
- Alain Lemaréchal. 1989. *Les parties du discours. Sémantique et syntaxe*. Linguistique nouvelle. Presses Universitaires de France, Paris.

- Alain Lemaréchal. 1997. *Zéro(s)*. Linguistique nouvelle. Presses universitaires de France, Paris.
- Caterina Mauri. 2008. *Coordination relations in the languages of Europe and beyond*. Number 42 in *Empirical approaches to language typology*. Mouton de Gruyter, Berlin and New York.
- Nicolas Mazziotta. 2009. *Ponctuation et syntaxe dans la langue française médiévale. Étude d'un corpus de chartes écrites en français à Liège entre 1236 et 1291*. Number 354 in *Beihefte zur Zeitschrift für romanische Philologie*. Niemeyer, Tübingen.
- Igor Mel'čuk. 1988. *Dependency syntax: theory and practice*. State University of New York, Albany.
- Igor Mel'čuk. 2009. Dependency in natural language. In Alain Polguère and Igor Mel'čuk, editors, *Dependency in linguistic description*, John Benjamins, Amsterdam and Philadelphia, 1–110.
- Philippe Ménard. 1994. *Syntaxe de l'ancien français*. Études médiévales, Bordeaux, Bière, 4th edition.
- Gérard Moignet. 1988. *Grammaire de l'ancien français. Morphologie – Syntaxe*. Number 2 in *Initiation à la linguistique. Série B. Problèmes et méthodes*, Paris, Klincksieck, 2nd edition.
- Andrew Rosta. 2006. Structural and distributional heads. In Kensei Sugayama and Richard Hudson, editors, *Word grammar. New perspectives on a theory of language structure*. Continuum, London and New York, 171–203.
- Achim Stein, Pierre Kunstmann, and Martin-Dietrich Gleßgen, editors. 2008. *Nouveau Corpus d'Amsterdam. Corpus informatique de textes littéraires d'ancien français (ca 1150-1350), établi par Anthonij Dees (Amsterdam 1987)*, Stuttgart, Institut für Linguistik/Romanistik, 2nd edition. DEAF identifiers.
- Lucien Tesnière. 1965. *Éléments de syntaxe structurale*. Klincksieck, Paris, 2nd edition.
- Christian Touratier. 2005. *Analyse et théorie syntaxiques*. Publications de l'Université de Provence, Aix-en-Provence.

Type 2 Rising

A Contribution to a DG Account of Discontinuities

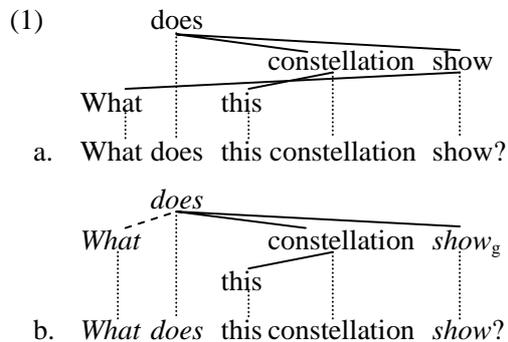
Timothy Osborne
tjo3ya@yahoo.com

Abstract

This contribution examines discontinuities in DG. Discontinuities are addressed in terms of *catenae* and *rising*. The catena is defined as A WORD OR A COMBINATION OF WORDS THAT IS CONTINUOUS WITH RESPECT TO DOMINANCE. The definition identifies any tree or subtree of a tree as a catena. Rising occurs when a governor fails to dominate one (or more) of its governees. Two sorts of rising are distinguished: type 1 and type 2. Type 1 rising obtains when the risen catena is a constituent, whereas type 2 rising obtains when the risen catena is a non-constituent. The Rising Principle expresses the main trait of instances of rising. Discontinuity sorts (e.g. *wh*-fronting, topicalization, scrambling, extraposition, NP-internal displacement) are classified in terms of type 1 and type 2 rising.

1 Introduction

Many dependency grammars (DGs) address discontinuities in terms of a flattening of structure. A displaced unit takes on a word as its head that is not its governor. Example (1a) illustrates a standard *wh*-discontinuity and example (1b) shows the manner in which the discontinuity is “overcome”:



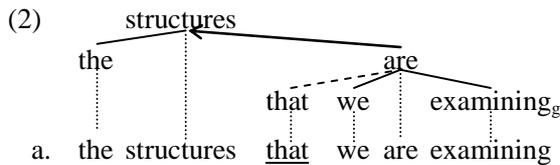
Tree (1a) illustrates a typical projectivity violation (=discontinuity). The fronted *wh*-element is separated from its governor in such a manner

that crossing lines obtain in the tree. The tree in (1b) shows the manner in which the crossing lines are “remedied”. The displaced unit takes on a word as its head that is not its governor.

The tree conventions shown in (1b) follow Groß and Osborne (2009). The dashed dependency edge indicates the presence of rising by which the discontinuity is overcome; the underline marks the displaced unit; the *g* subscript marks the governor of the displaced unit; and the italics mark the chain (=catena) of words the end points of which are the displaced unit and the governor of the displaced unit. These conventions will become clear as the discussion continues.

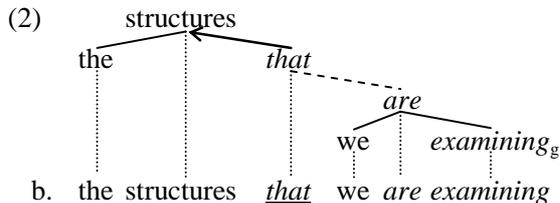
The flattening of structure illustrated in (1b) represents a widespread approach to discontinuities in DGs, although the terminology certainly varies: Hudson (2000:32) employs the term “raising” to address such constellations; Duchier and Debusmann (2001) use the term “climbing”; Gerdes and Kahane (2001) assume “emancipation”; Bröker (2003:294) posits “lifting”. Eroms and Heringer (2003:26) suggest movement and “adjunction”; and Groß and Osborne (2009) posit “rising”. This contribution follows the terminology of the latter. Discontinuities are addressed in terms of *rising*. While the accounts of these linguists certainly vary, the underlying idea pursued is consistent. This idea is that a flattening of structure occurs in order to overcome projectivity violations.

While there seems to be a measure of agreement concerning the manner in which DGs should address discontinuities like the one shown in (1), there are other structures involving discontinuities that pose major challenges and for which there seems to be much less consensus about the correct analysis. Consider, for instance, the structural analysis of the following example involving a relative clause:



The arrow dependency edge identifies an adjunct (as opposed to an argument). While the tree conventions shown again follow Groß and Osborne (2009), the actual hierarchy of words assumed is similar to proposal by Kunze (1975:160); the finite verb is seen as the root of the relative clause (not the relative pronoun).¹

The difficulty with the analysis in (2a) is that there are indications that the relative pronoun should be the root of the relative clause, not the finite verb. In German for instance, the presence of a relative pronoun evokes VF (=verb final) order just like subordinators do. Since subordinators are unanimously viewed as the root of the clause they introduce, the inference is that relative pronouns should also be the roots of the clauses that they introduce. This insight motivates the following structural analysis of (2):



The relative pronoun is now the root of the relative clause. The major difference between (2a) and (2b) is that the displaced unit, i.e. *that*, in (2a) is a constituent (=a complete subtree), whereas it alone is a non-constituent in (2b) (because it dominates other words).

This contribution argues that the analysis in (2b) should be preferred over the analysis in (2a). This situation necessitates that the theory distinguish between two types of discontinuities. Discontinuities like the one in (1b) are instances of *type 1 rising*, whereas discontinuities like the one in (2b) are instances of *type 2 rising*. The defining trait of type 1 rising is that the risen unit is a constituent (=a complete subtree), whereas the risen unit of type 2 rising is a non-constituent. Since type 2 rising is more likely to

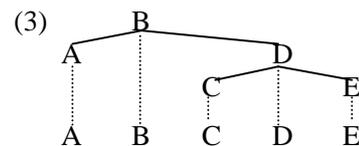
¹ The term “root” is used throughout to denote the one word in a given unit (e.g. constituent, catena) that is not dominated by any other word in that unit.

be controversial for DG theory, this contribution focuses more on it. The data examined are mostly from English and German.

2 Catena

Before exploring the distinction between type 1 and type 2 rising, the fundamental unit of syntactic analysis assumed in the current DG must be established. Following O’Grady (1998), Osborne (2005), and Osborne et al. (in press), the *catena* (Latin for ‘chain’, plural *catenae*) is posited as the fundamental unit of syntactic analysis.² The catena is defined as A WORD OR A COMBINATION OF WORDS THAT IS CONTINUOUS WITH RESPECT TO DOMINANCE. This definition identifies any dependency tree or any subtree of a dependency tree as a catena.

The catena unit is illustrated using the following abstract structure:



The capital letters represent words. The following 17 combinations qualify as catenae: A, B, C, D, E, AB, BD, CD, DE, ABD, BCD, BDE, CDE, ABCD, ABDE, BCDE, and ABCDE. The following 14 combinations, in contrast, qualify as non-catenae: AC, AD, AE, BC, BE, CE, ABC, ABE, ACD, ACE, ADE, BCE, ABCE, and ACDE. As the number of words increases, the percentage of non-catena combinations increases.

Given a theory neutral definition of the constituent (=A WORD/NODE PLUS ALL THE WORDS/NODES THAT THAT WORD/NODE DOMINATES), there are five constituents in (3): A, C, E, CDE, and ABCDE. Examining the combinations that qualify as catenae and that qualify as constituents, one sees that every constituent is a catena, but many catenae are not constituents. Thus THE CONSTITUENT IS A SUBTYPE OF THE CATENA.

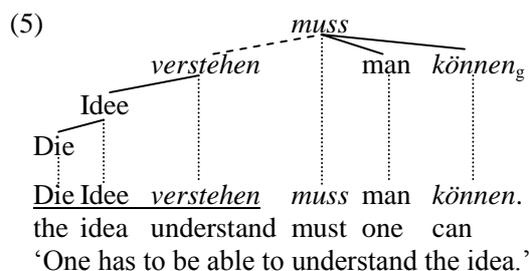
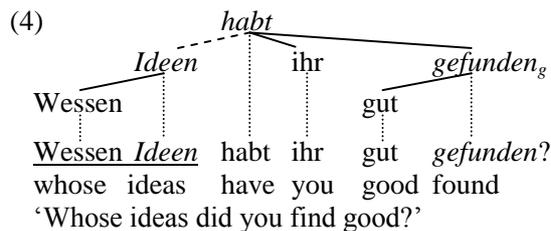
² O’Grady (1998), Osborne (2005), and Groß and Osborne (2009) employed the term “chain” (instead of “catena”). Osborne et al. (in press), however, replace the term “chain” with “catena” in order to avoid confusion coming from constituency-based derivational grammars, where “chain” has a much different meaning.

The detailed discussions of the catena unit in the sources cited at the beginning of this section establish the validity and importance of the concept. The discussion below can therefore take the concept for granted and base its account of discontinuities thereupon.

3 Type 1 rising

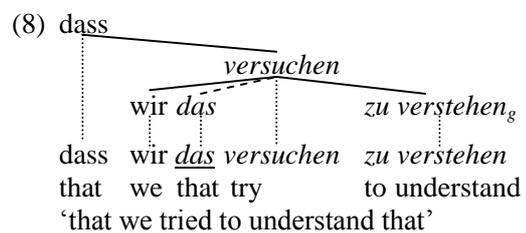
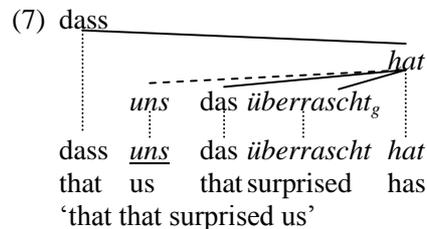
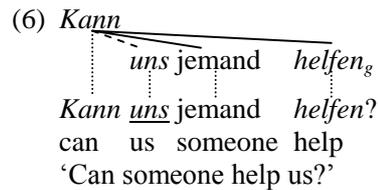
Type 1 rising occurs when the risen catena is a constituent. A number of discontinuity types involve Type 1 rising (e.g. *wh*-fronting in English and German, scrambling in German, and extraposition in English and German). This section briefly illustrates these discontinuity types and in so doing, establishes the particular terminology of the current DG theory of discontinuities. The tree conventions introduced with tree (1b) are again employed.

Example (1b) illustrated *wh*-fronting in English. The next two examples illustrate *w*-fronting rising and topicalization rising in German:

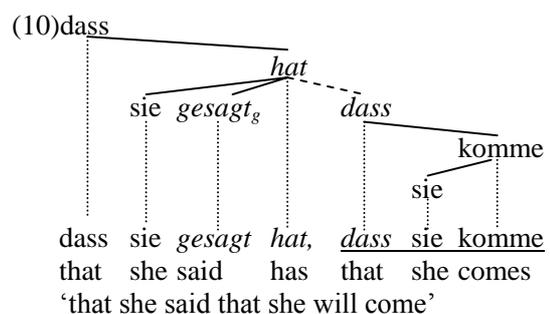
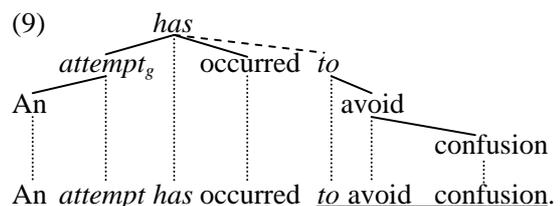


As mentioned above, the dashed dependency edge indicates the presence of rising, the underlined unit is the *risen catena*, the *g* subscript marks the governor of the risen catena, and the italicized words constitute what is now called the *rising catena*.

The following examples illustrate scrambling rising in German (Scrambling does not exist in English, of course):



And the following examples illustrate extraposition rising:

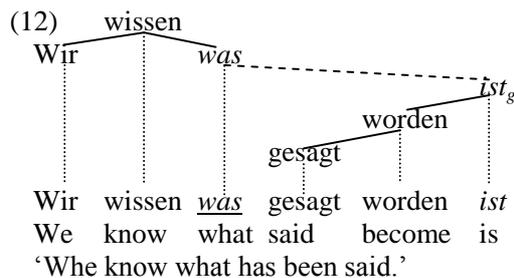
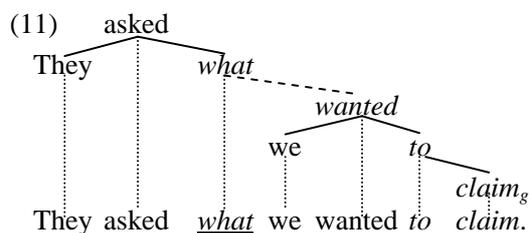


These instances of rising all show the major trait of type 1 rising. This trait is that the risen catena is a constituent (as defined above). While there are many aspects of these discontinuity types that deserve attention, the main point that is pertinent for the account of type 2 rising below has now been established. This point is that the risen catena of type 1 rising is a constituent.

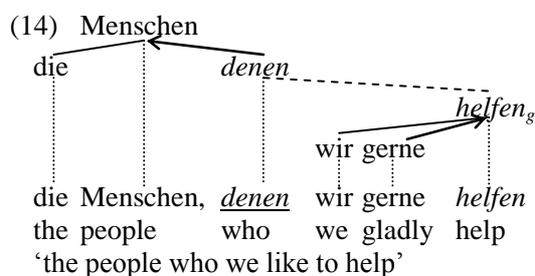
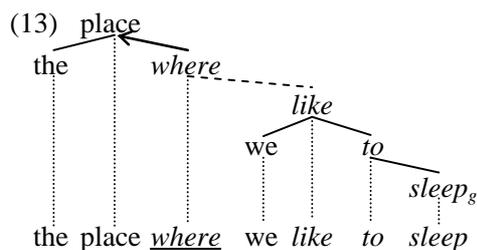
4 Type 2 rising and the Rising Principle

The instance of type 2 rising illustrated with (2b) shows the risen catena as a non-constituent. This aspect of type 2 rising allows one to easily distinguish the two types of rising. Any instance of rising where the risen catena is a non-constituent is type 2 rising. Type 2 rising occurs with *wh*-fronting in subordinate clauses (in indirect questions), with relative pronouns of all types, and with NP-internal displacement.

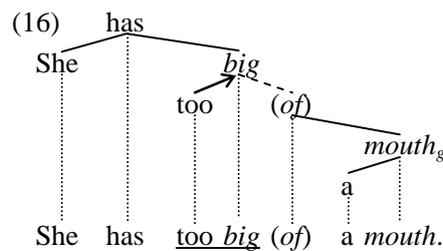
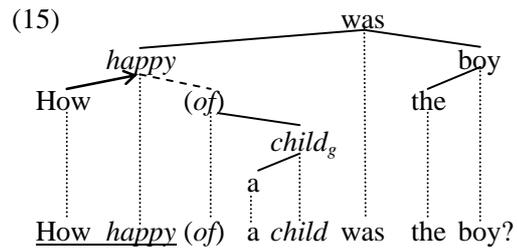
The following trees illustrate type 2 rising in indirect questions:



The following two examples illustrate type 2 rising in relative clauses:



And the following two examples illustrate type 2 rising inside NPs in English:



These two examples show type 2 rising within an NP. The parentheses indicate that the appearance of the preposition *of* in each case is optional. The risen adjective is focused by the adverb, i.e. by *how* and *too*. When the adjective is focused in this manner, it must be fronted within the NP. Interestingly, this sort of type 2 rising is completely absent from German. The pertinent observation in this regard that there are numerous discontinuity types, and languages vary with respect to the inventory of discontinuities that they allow.

The tree conventions in these instances of type 2 rising remain consistent. The risen catena in each case is underlined; the governor of the risen catena carries the *g* subscript, and the rising catena is in italics. Two things should be acknowledged about type 2 rising: again that the risen catena is a non-constituent and that the root of the risen catena necessarily dominates its governor.

Comparing the instances of type 1 rising in (4-10) with the instances of type 2 rising in (11-16), one sees that in cases of type 1 rising, the head of the risen catena dominates the governor of the risen catena,³ whereas with type 2 rising, the root of the risen catena itself dominates the governor of that risen catena. These two observations exhaust the possibilities, and they motivate the Rising Principle:

³ The head of a given catena is the one word (outside of that catena) that dominates that catena.

Rising Principle

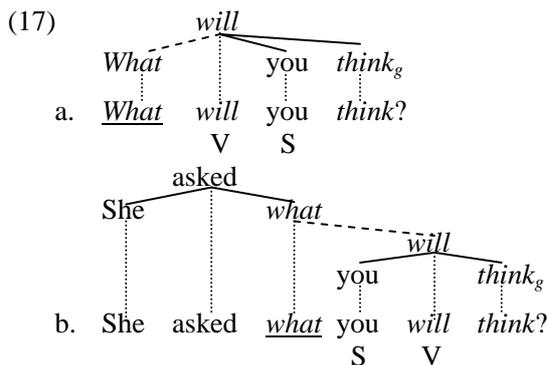
The head or the root of the risen catena must dominate the governor of the risen catena.

The examples above all obey this principle. Either the head of the risen catena dominates the governor of the risen catena (=type 1 rising) or the root of the risen catena dominates the governor of the risen catena (=type 2 rising). The Rising Principle is the major guideline that all discontinuities must obey. It helps limit the discontinuities that can obtain.

Since many DGs address discontinuities in terms of a mechanism like type 1 rising, type 1 rising should not be too controversial. Type 2 rising, however, is unique to the current DG. To my knowledge, no other DG has proposed something similar. Furthermore, there are some aspects of type 2 rising that generate questions about the nature of discontinuities and head-dependent relations in general. For these reasons, the following subsections motivate the current understanding of type 2 rising.

3.1 SV order

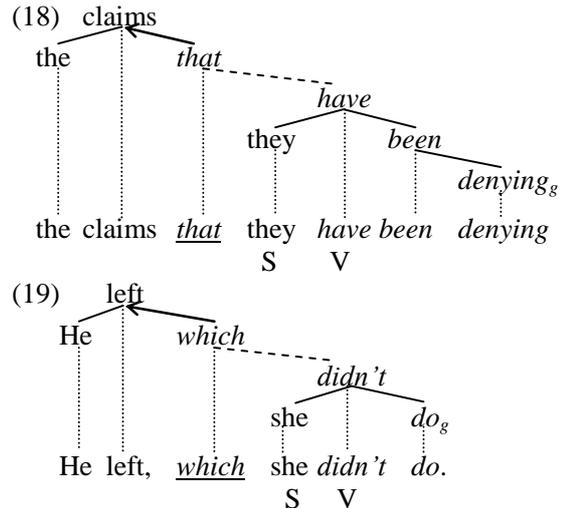
The first observation that supports type 2 rising comes from word order across direct and indirect clauses in English. Direct constituent questions in English can have VS order, where V is an auxiliary. In indirect questions in contrast, SV order obtains. These facts are illustrated with the following examples:



The direct question in (17a) has VS order, where V is an auxiliary verb. The indirect question (17b), in contrast, has SV order. Both sentences necessarily involve a discontinuity. By assuming the distinction between type 1 and type 2 rising, the VS vs. SV distinction can be accommodated.

If type 1 rising were the only type of rising that the theory had at its disposal, accommodating the contrast in a principled manner would be difficult.

The SV order of indirect questions is also seen in relative clauses of all sorts. This fact supports the type 2 rising analysis of these clauses.

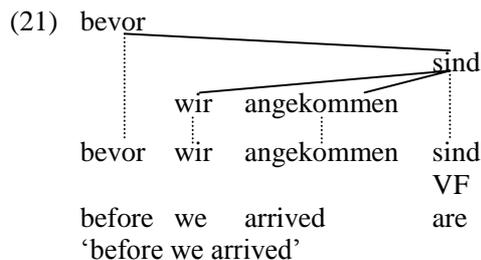
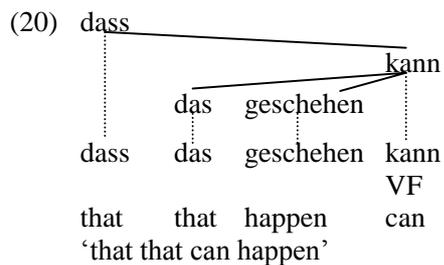


The same SV order seen in the indirect questions is present in relative clauses like these. The combination SV-order plus pronoun fronting is thus an indication of type 2 rising.

Beyond the VS vs. SV distinction, subcategorization considerations support type 2 rising. Question verbs (e.g. *ask*, *wonder*, *inquire*, *know*, etc.) subcategorize for an indirect question, whereby the question word is the most distinctive trait of a question (direct or indirect). And regarding relative clauses, the relative pronoun is the most distinctive word of a relative clause, so it makes sense that it should be the root of the relative clause.

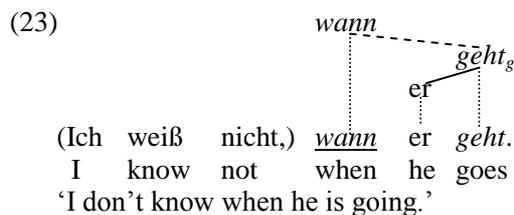
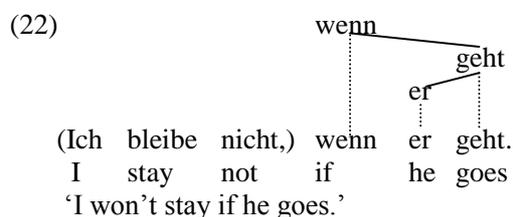
3.2 VF order

VF order in German subordinate clauses provides similar support for type 2 rising. Type 2 rising in many subordinate clauses in German provides a principled means of accounting for VF (=verb final) order. An initial observation in this regard is that the appearance of a typical subordinator evokes VF order, e.g.



The subordinators *dass* 'that' and *bevor* 'before' evoke VF order, which means the finite verb appears in clause-final position. Since this word order occurs with indirect questions and relative clauses as well, one can assume that such subordinate clauses should have a similar structure.

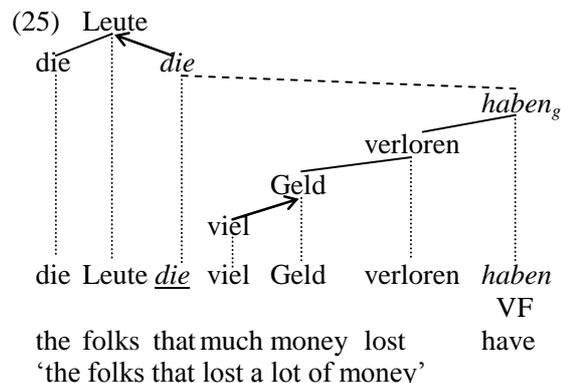
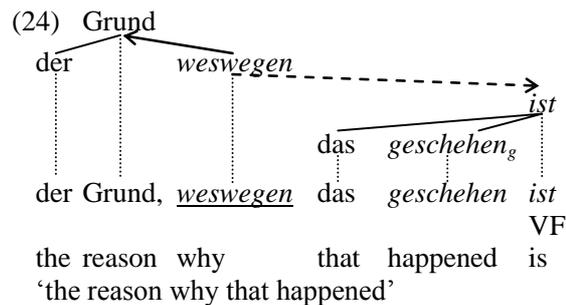
But only if type 2 rising is allowed can the structure of all VF clauses be parallel. Examine the parallelism of structure across the following clauses:



The closeness in form and meaning across the two clauses suggests strongly that they should have similar structures. The subordinator *wenn* 'when/if' and the interrogative proform *wann* 'when' convey similar meanings and they both evoke VF order. Type 2 rising allows for the parallelism to be acknowledged in the structure. If type 1 rising were all the theory had at its dis-

posal, there would be no way to establish the desired parallelism across all VF clauses.

The same observation speaks for type 2 rising in relative clauses in German. The appearance of the relative pronoun evokes VF order, which means that the relative proform should appear in a position where it can have this impact on the clause it introduces, e.g.⁴



The relative proforms *weswegen* 'why' and *die* 'that/who' evoke VF order. They should therefore appear in a position where they can exert this influence. Assuming type 2 rising allows them to serve as the root of the clause that they introduce.

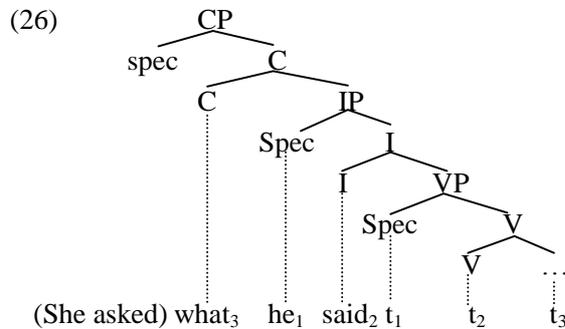
As mentioned above, distributional considerations provide a second source of support for type 2 rising in indirect questions and relative clauses in German. The defining trait of these clauses is the *wh*-element or relative proform. Since the presence of these elements influences greatly the distribution of the clauses in which they appear, granting them root status in the clause is appropriate.

⁴ The dependency arrow connecting *weswegen* to *ist* indicates that *weswegen* is an adjunct. Since the arrow always points away from the adjunct towards the governor of the adjunct, the arrow points downwards in this case.

3.3 Constituency-based hierarchies

A third observation supporting type 2 rising is of a much different nature (from the previous two observations). This third observation concerns the analysis of indirect interrogative clauses and relative clauses in constituency grammars (as opposed to in DGs). Constituency grammars of the GB/MP tradition see the *wh*-element or relative proform occupying the head position of the clause, e.g. the C position of CP. The type 2 rising analysis therefore mirrors this analysis of the GB/MP tradition.

The point is illustrated with the following GB analysis of a simple indirect question.



The details of this analysis (e.g. the traces) are not important for the matter at hand. What is important is the surface hierarchy shown. The *wh*-element *what* occupies C, whereby CP, the maximal projection of C, is the root node of the entire structure. This constituency-based analysis is therefore analogous to the DG type 2 rising analysis now under consideration.

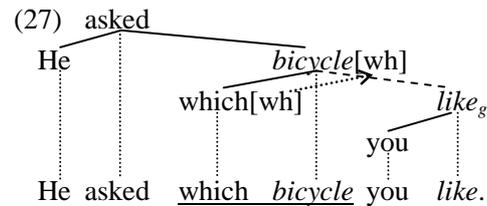
The type 2 rising analysis therefore opens the door to the massive body of literature on subordinate clauses in constituency-based systems. Many of the insights gained by the constituency-based systems are now applicable to dependency-based systems (that assume type 2 rising). A bridge of sorts now spans the two traditions (at least in this one area).

3.4 Pied-piping

Pied-piping in subordinate clauses presents a difficulty for the current analysis in terms of type 2 rising. The seriousness of this difficulty should not, however, be overestimated, since pied-piping challenges most analyses regardless of whether (something like) type 2 rising is assumed or not. The analysis of pied-piping that is

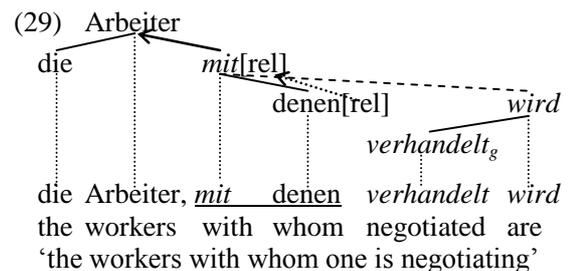
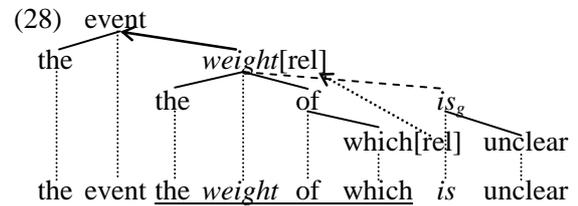
now proposed assumes that *wh*-features and relative proform features can in a sense percolate up a catena to a higher node.

Interrogative verbs subcategorize for a word with a *wh*-feature. When pied-piping occurs, this feature has percolated upward to the root of the pied-piped catena, e.g.



The *wh*-feature associated with *which* percolates to the root node of the risen catena. How exactly this percolation should be conceived of is not clear at this point, but that some sort of percolation mechanism is necessary is apparent. In fact regardless of the particular approach at hand, this passing of information up the structure is needed for pied-piping in general, in matrix as well as in embedded clauses.

Two more examples involving relative pronouns further illustrate the mechanism:



In these cases, the feature [rel] (indicating the presence of a relative pronoun) must percolate to the root node of the clause. By doing so, this feature is in a position to elicit the obligatory SV order of English or VF order of German associated with relative clauses.

Worth emphasizing again is that no matter the approach, some sort of percolation mechanism is needed to accommodate pied-piping. This

necessity is consistent across matrix and embedded clauses.

4 Rising in other languages

The discussion so far has focused on type 2 rising in English and German. The question arises as to whether type 2 rising exists in other languages. I believe that it does. Any time a *wh*-word or relative proform (or the encompassing pied-piped expression) must introduce a clause, one can make a case for type 2 rising. The discussion now briefly considers examples from French and Russian. These languages also exhibit type 2 rising.

The following example from French illustrates type 2 rising in an embedded interrogative clause:

- (30)
-
- a. Je- veux savoir où il est allé.
I want to know where he is gone
'I want to know where he went.'
- b. *Je veux savoir il est allé où ?
- c. *Je veux savoir où est-il allé.

The lack of vertical projection edge but presence of a hyphen on *Je-* identifies *Je-* as a clitic. This aspect of the tree is not pertinent to the point at hand and is therefore taken for granted.

The question word *où* is fronted in the relative clause in (30a). When this fronting fails to occur, the result is bad, as illustrated in (30b). And sentence (30c) demonstrates that fronting is incompatible with subject-auxiliary inversion of the sort that one encounters in matrix questions in French. These data can be accommodated if type 2 rising is seen as obligatory in embedded interrogative clauses in French (just like it is in such clauses in English and German). Note as well that subcategorization requirements in French in such cases are the same as in English and German. Since the matrix predicate subcategorizes for an interrogative element, it makes sense to view the *wh*-element as having risen in

the embedded clause to a hierarchical position that allows the subcategorization requirement of the matrix predicate to be satisfied.

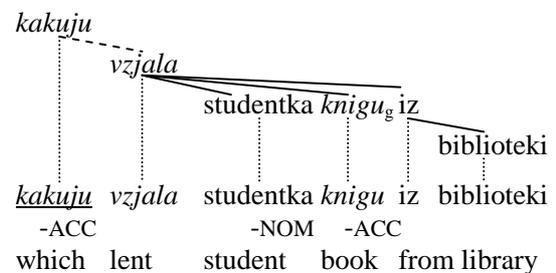
The following example contains a standard relative clause:

- (31)
-
- a. le client que vous reconnaissez
the client that you recognize
- b. *le client vous reconnaissez que.

As in English and German, the relative pronoun must undergo type 2 rising. If it does not (i.e. it remains in situ), the result is clearly unacceptable, as example (31b) shows. Based on such data, one can conclude that type 2 rising is occurring consistently in the same environments across English, German, and French (and certainly across many other languages as well).

The following example provided by an anonymous reviewer illustrates an embedded interrogative in Russian:

- (30) Skazhi emu,
tell him



'Tell him which book the student checked out of the library.'

The interrogative element *kakuju* 'which' must be fronted within the embedded interrogative clause, a fact that is consistent with an analysis in terms of type 2 rising.

The interesting aspect of this example is that the interrogative word *kakuju* fails to pied-pipe its governor *knigu*. Note that in English, French, and German, such sentences are bad, e.g. *Tell him which the student checked out book from the library. This contrast across the languages is explained in terms of Ross (1967)

left branch condition. Languages like English, German, and French cannot extract an element on a left branch out of an NP. Such cases require pied-piping, e.g. *Tell him which book the student checked out of the library*. Apparently, the left branch condition is not in force in Russian. This fact should perhaps not be a surprise, since the word order of Slavic languages like Russian is known to be much freer than that of the Germanic (and Romance) languages.

In sum, there is evidence that type 2 rising is the key to producing a principled DG analysis of many embedded clauses across numerous languages.

7 Conclusion

This contribution has provided a DG account of discontinuities in English and German. Displaced units are addressed in terms of catenae and rising. Two types of rising are acknowledged: type 1 and type 2. Since many DGs posit some mechanism akin to type 1 rising, it should not be too controversial. Type 2 rising, however, is unique to the current DG. Type 2 rising occurs when the risen catena is a non-constituent.

By acknowledging type 2 rising, DG is in a position to address all discontinuities in a principled fashion. All displacement obeys the Rising Principle, which requires that either the head (type 1) or the root (type 2) of a risen catena dominate the governor of that risen catena. This principle significantly limits the type of discontinuities that the grammar allows. The second half of the discussion concentrated on aspects of type 2 rising. Word order considerations (SV, V2, VF) provide the primary support for type 2 rising.

Finally, something should be said about rising catenae. This concept was introduced and shown in the trees (via italics), but almost nothing has been said about why the concept is important. A more comprehensive account of discontinuities would show that each specific discontinuity type (*wh*-fronting, topicalization, scrambling, extraposition, NP-internal displacement) can be described and explained in terms of the rising catenae that each allows. Since the concept is important in this regard, drawing attention to it here was warranted.

References

- Bröker, N. 2003. Formal foundations of dependency grammar. Ágel, V. et al. (eds), *Dependency and valency: An international handbook of contemporary research*, vol. 1, 294-310. Berlin: Walter de Gruyter.
- Duchier, D. and R. Debusmann. 2001. Topology dependency trees: a constraint based account of linear precedence. Proceedings from the 39th annual meeting of the Association Computational Linguistics (ACL) 2001, Toulouse, France, 180-187.
- Eroms, H.-W. and H. J. Heringer. 2003. Dependenz und lineare Ordnung. Ágel, V. et al. (eds.), *Dependency and valency: An international handbook of contemporary research*, vol. 1, 247-262. Berlin: Walter de Gruyter.
- Gerdes, K. and S. Kahane. 2001. Word order in German: A formal dependency grammar using a topology model. Proceedings from the 39th annual meeting of the Association Computational Linguistics (ACL) 2001, Toulouse, France, 220-227.
- Groß, T. and T. Osborne 2009. Toward a practical DG theory of discontinuities. *Sky Journal of Linguistics* 22. 43-90.
- Hudson, R. 2000. Discontinuities. Kahane, S. (ed.), *Les grammaires de dépendance* (Dependency grammars), *Traitement automatique =des langues* 41, 7-56. Paris: Hermes.
- Kunze, J. 1975. *Abhängigkeitsgrammatik*. *Studia Grammatica* 12. Berlin: Akademie Verlag.
- O'Grady, W. 1998. The syntax of idioms. *Natural Language and Linguistic Theory* 16:79-312.
- Osborne, T. 2005. Beyond the constituent: A DG analysis of chains. *Folia Linguistica* 39(3-4). 251-297.
- Osborne, T., M. Putnam and T. Groß. (forthcoming). Catenae: Introducing a novel unit of syntactic analysis. *Syntax*.
- Ross, John R. 1967. *Constraints on Variables in Syntax*. Ph.D. dissertation, MIT.

Catenae in Morphology

Thomas Groß

Aichi University, Nagoya, Japan

tmgross@vega.aichi-u.ac.jp

Abstract

This paper argues for a renewed attempt at morphology in dependency grammar. The proposal made here is based on the concept of the “catena” proposed by Authors (in press). The predecessor to this notion was the “chain” introduced by O’Grady (1998), and employed by Osborne (2005) and Groß and Osborne (2009). In morphology and morphosyntax, a *morph catena* is A MORPH OR A COMBINATION OF MORPHS THAT IS CONTINUOUS WITH RESPECT TO DOMINANCE. This concept allows for a parsimonious treatment of morphology on the surface. The fact that no additional terms and concepts are necessary for the analysis of morphological data is highly desirable because it makes a fluid transition from syntax to morphology possible. This paper introduces the relevant dependency relationships seen operating in morphology, and shows how they can be used to explain compound structure, bracketing paradoxes, and multiple periphrasis.

1 Introduction

Hays (1964: 517f; see in particular the second example on page 518) may have been the first to recognize the merit of extending the notion of dependency into morphology. The motivation for doing so is clear: the complexity of word structure in languages differs, and if dependency grammar desires to say something enlightening about languages with different word structure, then it must have the means to do so. Heringer (1970: 96f) provided perhaps the first dependency trees that included separate nodes for morphs. Anderson (1980) was the first to use the label “dependency morphology”, in his analysis of Basque verbs. Both Heringer’s and Anderson’s analyses are characterized by the assumption that derivational and inflectional morphs depend on the lexical morphs with which they form words. This assumption has carried on to the present (e.g. Eroms 2010: 38f). Speculating on the reasons for this assumption, the European tradition sees dependency grammar as the theoretical background for valency theory. A brief look at Ágel and Fischer (2010) confirms this evaluation; valency theory is treated prominently and initially on 14 pages, while dependency grammar

takes the backseat with just 8 pages. Valency theory is characterized by putting valency-bearing lexical items at center stage. Assuming that non-lexical material is somehow subsumed by lexical material seems on a logical trajectory. But research in typology, foremost Bybee (1985), has confirmed that affixes as expressions of valency, voice, aspect, modality, tense, mood, and person obtain in a specific linear order (or hierarchy), and developments in generative grammar during the 1980’s emphasized the dominance structure of the IP/TP, where such affixes are thought to be located. Similar statements also concern NP structure: if case or plural is expressed by morphs, then these morphs appear in peripheral position, an indication that they dominate their nouns. In general, it is safe to say that dependency grammar has missed out on important trends and insights, and this has severely hampered any formulation of a dependency-based morphology. The fact that Anderson went on to establish “dependency phonology” (Anderson & Ewen 1987) instead of pursuing his initial program of dependency morphology, is a case in point. Among the widely known dependency grammars, only Mel’čuk’s Meaning-Text-Theory (1988) and Hudson’s Word Grammar (1984, 1990, 2007) explicitly address morphology. While the notion of dependency can be considered as established in syntax and phonology, morphology is still underdeveloped. In recent times, Harnisch (2003) and Maxwell (2003) have argued again that dependency grammar must achieve a better understanding of the morphological component.

This paper outlines a proposal for a dependency morphology based on the notion of “chain”, which was introduced by O’Grady (1998). O’Grady shows that many idioms do not qualify as constituents, rather they form incomplete dependency trees, which he called “chains”. Osborne (2005) recognized the versatility of this notion for dependency grammar. Groß and Osborne (2009) use the chain concept to address discontinuous structure in syntax, and Groß (2010) endeavors, in a first attempt, to apply the chain to word structure, arguing that bracketing paradoxes and multiple auxiliary constructions

can be quite easily resolved. Below, however, the term *catena* will be used instead of “chain” because “chain” is understood in an entirely different way in derivational theories of syntax. This decision is also motivated by the work of Osborne et al (in press), who show that the catena, rather than the constituent, is implicated in idiom formation, ellipsis, and predicate formation. They define a catena (in syntax) as A WORD OR A COMBINATION OF WORDS THAT IS CONTINUOUS WITH RESPECT TO DOMINANCE. This definition identifies any dependency tree or subtree of a tree as a catena. By replacing “word” with “morph”, the catena is also available for morphology.

This paper proceeds as follows: Section 2 informs on the central notions and shows how they are used to explain morphological dependencies within and across words and with clitics. It also illustrates briefly that non-concatenative morphology can be dealt with. Section 3 concerns compounds: gradient compound structure as well as exocentric compounds are explained. Section 4 addresses bracketing paradoxes. Section 5 demonstrates that a catena-based approach can parsimoniously account for multiple periphrasis. A final section concludes the paper.

2 Catena-based morphology

Building on Osborne et.al. (in press), a *morph catena* is a MORPH OR A COMBINATION OF MORPHS THAT IS CONTINUOUS WITH RESPECT TO DOMINANCE. The choice of “morph” instead of “morpheme” is motivated by the need to maintain a surface-oriented level of analysis. A *morph* is loosely defined as any meaning bearing unit that cannot be reduced any further, but that can be separated from other meaning bearing units in the horizontal AND/OR vertical dimension.¹ The inclusion of the notion “vertical dimension” allows for the treatment of phenomena subsumed under non-concatenative morphology (trans- and suprafixation, reduplication, etc.), as briefly demonstrated below. This section addresses morph dependencies within and across words, clitics, and non-concatenative morphology.

2.1 Within words

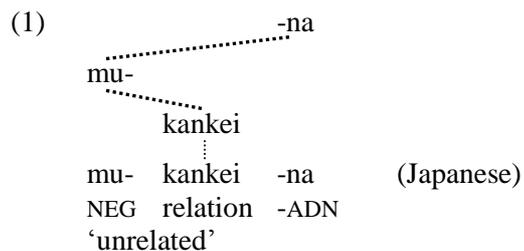
Morph catenae obtain in morphology proper, i.e. inside words, and in morphosyntax, i.e. across

¹ While there are certainly difficulties with the notions “morph” and “morpheme” (cf. Mel’čuk 2006: 384ff), the proposal here is sufficient in the present context.

words. A dependency relationship between morphs inside the same word is called an *intra-word* dependency. Intra-word dependencies are determined by distribution:

If the combination of two morphs M_1 and M_2 distributes more like M_2 than like M_1 , then M_1 is a dependent of M_2 .

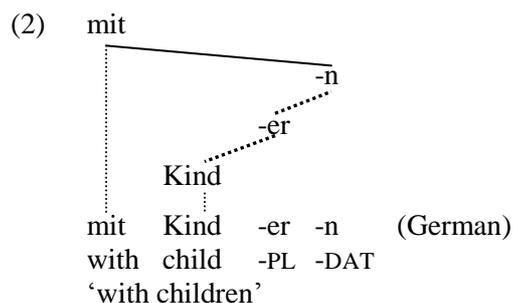
This definition is similar to Mel’čuk’s definition of “surface syntactic dominance” (2003: 200f). The next example from Japanese illustrates intra-word dependencies:



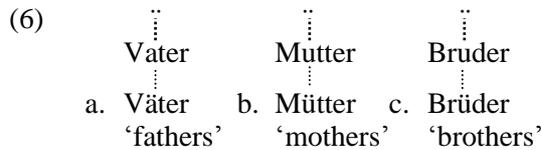
The intra-word dependencies are represented by the dotted edges (as opposed to solid edges). The lexical morph *kankei* receives a (vertical) projection edge. The hyphens represent phonological attachment (in the horizontal dimension). The negation prefix *mu-* phonologically attaches to the next morph to its right, and the attributive suffix phonologically *-na* attaches to the next morph to its left; in (1) this morph is *kankei*. The prefix *mu-* must depend on the suffix *-na* because the morph combination *mu-kankei* distributes like a member of the lexical class of nominal adjectives “*keiyō meishi*”. The morph catena *kankei-na* is not possible because *kankei* is a noun rather than a nominal adjective. Intra-word dependencies are thus motivated on the basis of distribution.

2.2 Across words

An inter-word dependency is a morphosyntactic relationship between a morph and a word. If the morph licenses the appearance of the word, then the morph *governs* the word. The next example illustrates that with an example from German:



er – *Brüder* etc. Since the appearance of the ablaut changes the distribution of the whole expression, it is seen as the root:



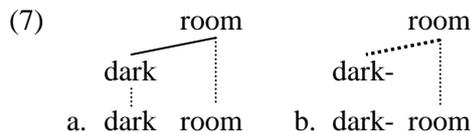
The ablaut, represented by “”, now constitutes an individual node that can be accessed. The dotted dependency edge is now completely vertical, a feature also present in infixation, transfixation, and suprafixation. Reduplication, suprafixation, and infixation can be accommodated in a similar vein.

3 Compounds

Compounds are words containing at least two lexical morphs. Because lexical morphs have the ability to constitute prosodic words, the appearance of two lexical morphs in one prosodic word requires one of these morphs to be integrated into the prosodic word structure of the other.

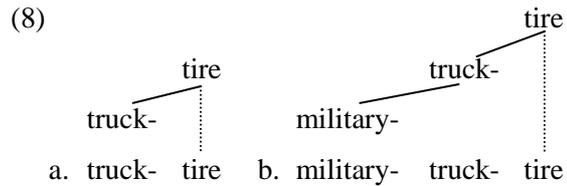
3.1 Compound gradience

Compounds are of particular interest for dependency morphology because the semanto-syntactic connection between compound parts exhibits gradience. Consider the next English examples:



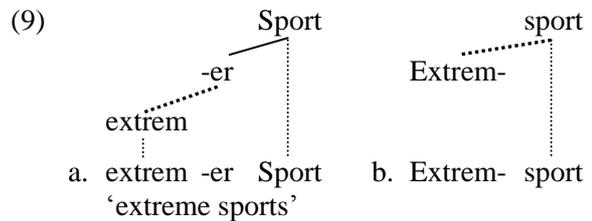
Example (7a) shows a purely syntactic dependency relationship. The attributive adjective can still be modified by e.g. *very*. In (7b), that is impossible, hence this expression is a compound. Because *dark-room* denotes a kind of room, not a kind of dark(ness), *room* is seen as the root dominating the adjective. The adjective is integrated into the prosodic word structure of the morph *room*, which is represented by the hyphen on *dark-*. Morphs must either be marked by a hyphen or receive a projection edge (but never both).

The words in (7a-b) represent the endpoints of a compound continuum. English allows compounds to reside between these two end points, as the next examples demonstrate:



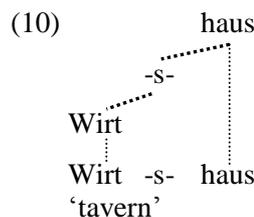
Example (8a) is a compound, but unlike (7b). Here *truck-*, can still be modified, as (8b) illustrates. The truck is a military type of truck, rather than the tire being a military type of tire. This kind of compound is less syntactic than (7a), but more syntactic than (7b); this fact is represented by the solid dependency edge between the compound parts.

German seems to dislike (8a)-type compounds. Modifying adjectives must appear without their attributive suffixes, an indication that the modified noun has lost the ability to license the appearance of attributives:



In (9a) the adjective is a regular attributive adjective, and it can be modified by *sehr* ‘very’. In (9b) however, the adjective is integrated into the prosodic word structure of *sport*, and it cannot be marked with the attributive suffix *-er* (or any other inflectional suffix), thus indicating compounding.

But German can build compounds by using the Fugen *-s-*:

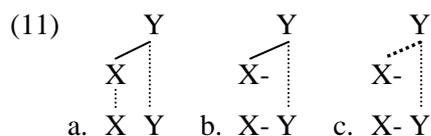


Example (10) is very simple, and much more complex examples exist (e.g. *Einzugsermächtigung* ‘collection authorization’). The important issue here is that *-s-* combines two units, each of which requires one of its morphs to be marked with a projection edge (here: *Wirt* and *haus*). The hyphens on either side of *-s-* signal this important function; technically, *-s-* functions as an infix.

3.2 “Exocentric” compounds

Exocentric compounds come in different types:⁴ in *bahuvrihi* compounds, the meaning of the entire expression cannot be deduced from its parts, or only with great difficulty, e.g. *skinhead*, *old-money*, *bluecollar*, etc. Other types of exocentric compounds defy morphological categorization. The words *musthave* and *kickback* are nouns (rather than verbs), auxiliaries, or prepositions. Furthermore, there are *dvandva* compounds: copulative *dvandva* have two (or more) semantic heads such as *bitter-sweet* or *sleep-walk*, and in appositional *dvandva* the compound parts contribute to a similar degree to the meaning of the entire expression, such as in *maid-servant*.

At first blush, *bahuvrihi* and *dvandva* compounds are removed from productive compounds to a significant degree. *Bahuvrihi* such as *skinhead*, which means a certain type of person, rather than a body part, are in the process of idiom formation or have already completed this process. Applying O’Grady’s (1998) lesson of syntactic idioms to compounding leads to the straightforward assumption that the units involved in these types of compound must qualify as catenae if they are to be retained in the lexicon. But the lexicon, as understood in construction grammar, also contains constructions, which is why Goldberg (1995) calls it “constructicon” rather than lexicon. Concerning compound constructions, English requires the root of the compound to be a nominal, i.e. a noun, adjective, or some other nominal form. In other words, the English compound construction continuum could look like this (with the horizontal order being free):



Construction (11a) is purely syntactic, like (7a). In the next step (11b), X loses its ability to constitute a prosodic word, but still retains the ability to govern modifiers. At stage (11c), the ability to govern modifiers is relinquished. Beyond that stage, a new morph obtains. The example *truck-tire* in (8a) is at stage (11b), while (11c) is accurate for *dark-room* in (7b). In general, a construc-

⁴ The literature on this topic is quite extensive. Compounding and their types are treated in Fabb (1998), Olsen (2000), Ten Hacken (2000), Bauer (2001, 2009), etc. *Dvandva* are addressed in Bauer (2008). See Scalise and Bisetto (2009) and Arcodia (2010) for an overview.

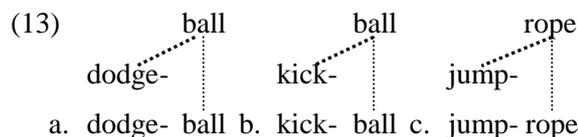
tion with closer association of its parts should be preceded by a construction with freer association at an earlier time. When and how the association changes is a matter for specialists. The assumption of such a continuum is, however, compatible with much research in grammaticalization theory, see Bybee’s (2010:136-50) analysis of Engl. *in spite of*. The important issue here is that in order to undergo this process, the individual parts of the complex expression must form catenae.

Since the *bahuvrihi* compound classes are very extensive, the discussion concentrates on four classes that contain verbal morphs:

- (12) a. VERB + NOUN
 b. VERB + PARTICLE
 c. PARTICIPLE + PARTICLE
 d. AUXILIARY + VERB

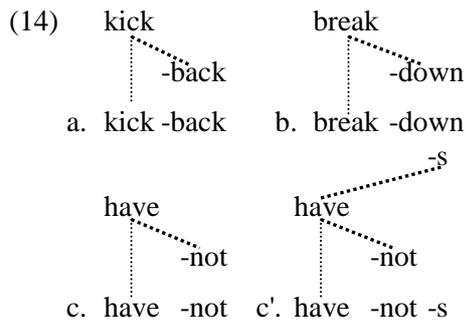
Examples for type (12a) are *dodgeball*, *kickball*, *jumprope* etc. For type (12b), one finds *kickback*, *breakdown*, *havenot* etc, and examples for type (12c) are *rundown*, *letdown*, *shutout*, etc. Type (12d) includes *musthave* and *hasbeen*.

Even though the noun *ball* depends on the verbs *dodge* and *kick* in the VPs *dodge (a) ball* and *kick (a) ball*, the noun dominates the verb in the compounds because these compounds denote specific objects or activities using these objects, and these objects are represented by *ball* and *rope*. Type (12a) exhibits the following morph dependencies:



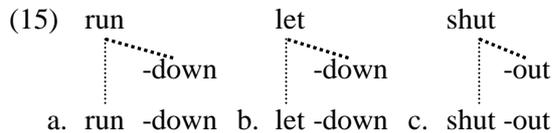
Examples (13a-c) show that the initial compound part depending on the final compound part.

Type (12b) compounds differ from type (12a) insofar as the initial compound part is seen as the root. Expressions such as *kickback*, *breakdown*, *havenot* etc. are clearly nominals, because they can be pluralized: *kickbacks*, *breakdowns*, *havenots*. It is, however, the initial compound parts that undergo plural formation, i.e. *kicks*, *breaks*, *haves*, rather than **backs*, **downs*, **nots*. Multiple *jumpropes* are still multiple *ropes*, while multiple *kickbacks* are not multiple *backs*, but multiple instances of kicking back. Hence the assumption that the initial parts form the roots, and that the plural morph vertically attaches to the initial parts is also justified when seen from semantics. The structure of type (12b) compounds is shown next:



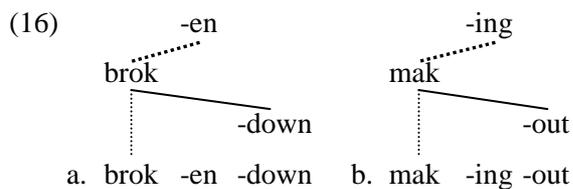
(14a-c) show the structure of *kickback*, *break-down*, and *havenot*. Example (14c') shows the plural form of (14c).

Type (12c) is a variation of type (12b). The difference lies with the form of the verb morph, which appears as a stem in (12b) but as a participle form in (12c). As long as the participle forms do not contain overt participle morphs, type (12c) compounds are seen as structured along the lines of (14):



Type (12c) compounds such as (15a-c) appear as nominal compounds because the participle is a nominal form. In the examples (13a-c), (14a-d), and (15a-c), dotted dependency edges obtain because no material can intervene between the compound parts.

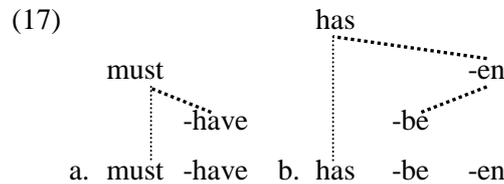
When a participle morph is present, a solid dependency edge between the verb morph and the adverb must obtain because the participle morph must intervene in the horizontal dimension:



In (16a-b), the participle morphs *-en* and *-ing* mark the expressions as nominals, but they appear in medial position. The adverbs must therefore be connected by solid dependency edges. This indicates that, in the compound continuum, the expressions in (15a-c) are located closer to the lexical endpoint of the continuum than the expressions (16a-b). More precisely, the expressions (15a-c) are at stage (11c), while the expressions (16a-b) reside at stage (11b). Since highly irregular verbs such as *run*, *let*, *shut*, etc. do not appear with a participle morph, they can lexical-

ize more readily than expressions that contain such morphs.

Finally, type (12d) compounds like *musthave* seem to be very rare. Nevertheless, their structure must be like (15):



Compare the structure (17b) with periphrasis in Section 5 below.

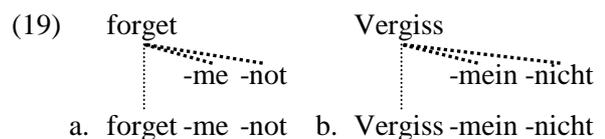
Once an expression has reached the stage (11c), it can be converted into a verb: *babysit*, *benchpress*, *bodycheck*, *bullrush*, *carpetbomb*, *crashdive*, *fieldtest*, *housebreak*, *housesit*, *proofread*, *slamdunk*, *tapdance*, etc.⁵ Many of these examples are considered to have undergone backformation; for instance, *baby-sit* is derived from *babysitter*, *carpetbomb* from *carpetbombing*, etc. Other examples such as *benchpress* or *crashdive* are seen as zero-conversion. One real-life example shows the conversion of the compound noun *cake-walk* into a verb:

(18) ...as Joseph Addai really *cakewalked* into the endzone...

This example appeared in the commentary of the Colts-Raiders game (season 2010/11, week 17), and it illustrates the productivity of the reconversion of apparent compounds to lexical morphs.

3.3 Clausal compounds

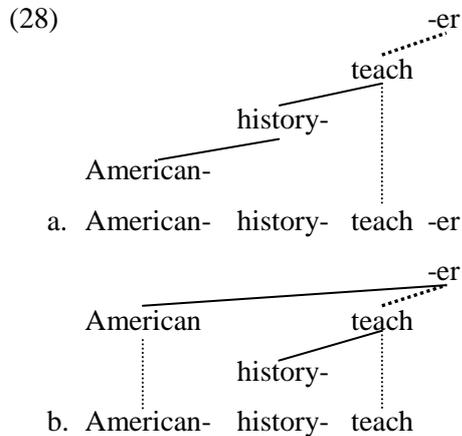
A further phenomenon of interest is compounds containing whole clauses. Well known examples include the fully lexicalized English *forget-me-not* and its German version *Vergissmeinnicht*. Both are based on imperative clauses: evidence for this assumption is the ablaut of *vergiss*, the stem of which is *vergess*. In German verbs with an /e→i/ ablaut, the ablaut version serves as the imperative form. Since the verb is the clausal root, it retains this role in compounding within its compound part. The structure of *forget-me-not* and *Vergissmeinnicht* are given next:



⁵ Contrary to spelling conventions, none of these expressions is written with a hyphen here, because these words are fully lexicalized.

quired for the correct semantic interpretation of the entire expression (and which is missing in (25a)).

A phenomenon related to bracketing paradoxes appears in compounding. Fabb (1998: 72f) calls this phenomenon “subconstituency”. He uses the example *American history teacher*:



In (28a) *American history* is traditionally seen as a subconstituent of the whole expression, which refers to a teacher of American history, the teacher not necessarily being an American. In (28b), *history teacher* is seen as a subconstituent of the entire NP, which now refers to an American teacher of history, the history not necessarily being that of America.

5 Multiple periphrases

That multiple auxiliary constructions, i.e. multiple periphrases, are a problem was acknowledged early on by Chomsky (1957: 39). He posits “affix hopping” in order to explain why the morphemes expressing aspect and voice do not appear together on the surface. Consider the next sentence:

(29) The problem has be-en be-ing discuss-ed.

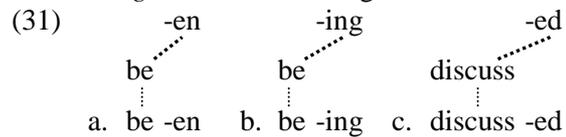
The units *has* and *-en* express perfective aspect, the first *be* and *-ing* express progressive aspect, and the second *be* and *-ed* express passive voice. The problem is that these units of functional meaning are not contiguous, because parts of other functional units intervene on the surface. For instance, *be* of the progressive unit intervenes between *has* and *-en* forming the perfective aspectual unit. Chomsky (1957: 39) proposed that the respective units are contiguous at a deeper level, and the affix of the unit “hops” over the verb of the next unit. The next example, based on Anderson (1992: 16), shows how this proposal plays out:

(30) ... (has -en) (be -ing) (be -ed) (discuss)

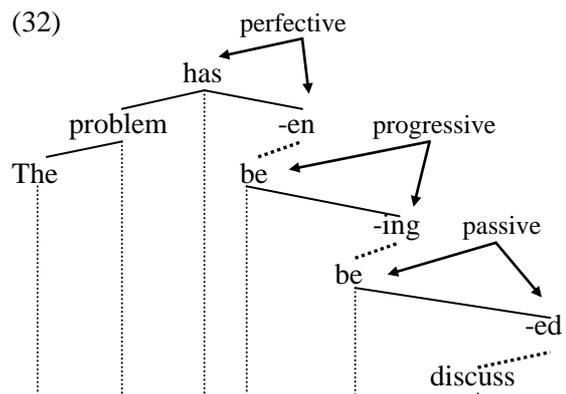
This “hopping” guaranteed that there was one level at which the respective units were contiguous, a prerequisite to establishing a semantic relationship.

In Distributed Morphology (DM) (Halle & Marantz 1993, Harley & Noyer 2003, Embick and Noyer 2001/2007, Embick 2003), affix hopping is now seen as the predecessor of “lowering” and “local dislocation”.⁶ Whatever one calls the mechanism, the core assumption is that if some unit is displaced on the surface, this unit must have moved to its surface position from a position at which it was contiguous with other units with which it forms a greater semantic unit.

Based on the concepts introduced in the previous sections, example (29) can now be reexamined. The structure of the individual words *been*, *being*, and *discussed* is given below:



In (31), the suffixes invariably dominate their lexical verbs: in (31a), *-en* dominates *be* because *be-en* distributes like a participle rather than as the infinitive. The same is true for (31c). In (31b), *be-ing* distributes like a progressive marked verb form rather than like the infinitive. The complete morph dependency structure of example (29) is now shown:



The problem has be -en be -ing discuss -ed.

The dependency structure in (32) must first be compared to the affix hopping/lowering analysis in (30): the units expressing the respective functional meanings are present as units on the surface. *has* and *-en* (=perfective aspect), *be* and *-ing* (=progressive aspect), and *be* and *-ed*

⁶ See Sternefeld (2009: 481-88) for an overview.

(=passive voice) qualify as morph catenae. The assumption of movement is unnecessary, since the respective morph combinations are discernible in the vertical dimension (rather than the horizontal dimension).

Two issues are of importance here: 1. The analysis in (32) obeys the Bybee hierarchy (1985: 196-7), because the perfective morph catena dominates the progressive morph catena, which in turn dominates the voice catena. 2. The respective functional meanings are expressed by units that qualify neither as constituents nor as words. As a corollary, the morph catena is – like its syntactic equivalent – a unit of meaning, available on the surface.

6 Conclusion

This paper has argued that morphological structure can be captured in dependency grammar by extending the notion of the catena from syntax into morphology. The fact that no additional concepts are necessary – and thus that morphology plays out as syntax inside words is desirable. Section 2 introduced the morph catena as A MORPH OR COMBINATION OF MORPHS THAT IS CONTINUOUS WITH RESPECT TO DOMINANCE. The two relevant dependency relationships between morphs were then established: intra-word dependencies obtain between morphs contained in the same word; they are based on distribution. Inter-word dependency, or government, plays out between a morph and a word, so that the morph licenses the appearance of the word. Using these two concepts, morphs can be connected into catenae regardless of the complexity of the structure. It has also been demonstrated that this account can accommodate non-concatenative morphology (although these phenomena were not in focus).

The main message of this paper is that dependency grammar should and can make more of morphology. At present, dependency grammar operates in syntax. However, the same meaning can be encoded at different levels in different languages. For instance, causative constructions are periphrastic in English and German, but morphological in Japanese. In order to compare languages, the concentration on syntax alone is insufficient; rather it is necessary to provide a system that enables a fluid transition of description from syntax to morphology and back. This is possible if dependency relationships are seen as operating not only in syntax, but also in morpho-syntax and morphology. The catena concept al-

lows for a fluid transition between syntax, morpho-syntax, and morphology, and thus simplifies the theoretical apparatus.

References

- Ágel, V. and K. Fischer. 2010. 50 Jahre Valenztheorie und Dependenzgrammatik. *Zeitschrift für germanistische Linguistik* 16. 249-290
- Anderson, J. 1980. Towards dependency morphology: the structure of the Basque verb. Anderson, J. and C. Ewen eds., *Studies in Dependency Phonology*, 221-271. Ludwigsburg.
- Anderson, J. M. and C. J. Ewen. 1987. *Principles of dependency phonology*. Cambridge: Cambridge University Press.
- Anderson, S. R. 1992. *A-Morphous Morphology*. Cambridge: Cambridge University Press.
- Anderson, S.R. 2005. *Aspects of the Theory of Clitics*. Oxford/New York: Oxford University Press.
- Arcodia, G.F. 2010. Coordinating compounds. *Language and Linguistics Compass* 4/9, 863-873.
- Bauer, L. 2001. Compounding. Haspelmath, M., E. König, W. Oesterreicher and W. Raible eds., *Language typology and language universals: an international handbook*, 695–707. Berlin/New York: Mouton de Gruyter.
- Bauer, L. 2008. Dvandva. *Word Structure* 1. 1–20.
- Bauer, L. 2009. Typology of compounds. Lieber, R. and P. Štekauer eds., *The Oxford handbook of compounding*, 343–56. Oxford: Oxford University Press.
- Becker, T. 1993. Back-formation, cross-formation, and ‘bracketing paradoxes’ in Paradigmatic Morphology. In Booij, G. & van Marle, J. (eds.), *Yearbook of Morphology* (vol. 6). Dordrecht: Foris Publications. 1–25.
- Booij, G. 2007. *The Grammar of Words*. Second Edition. Oxford: Oxford University Press.
- Bybee, J.L. 1985. *Morphology: A study of the relation between meaning and form*. Amsterdam/Philadelphia: John Benjamins Publishing
- Bybee, J. 2010. *Language, Usage and Cognition*. Cambridge: Cambridge University Press.
- Embick, D. and R. Noyer. 2001. Movement operations after syntax. *Linguistic Inquiry* Vol 32, No. 4, 555–595
- Embick, D. and R. Noyer. 2007. Distributed Morphology and the Syntax/ Morphology Interface. Ramchand, G. and C. Reiss eds. *The Oxford Handbook of Linguistic Interfaces*. 289-324. Oxford University Press.

- Embick, D. 2003. Linearization and local dislocation: Derivational mechanics and interactions. *Linguistic Analysis* 33/3-4. 303-336.
- Eroms, Hans-Werner. 2010. Valenz und Inkorporation. Kolehmainen Leena, Hartmut E. Lenk and Annikki Liimatainen eds. *Infinite Kontrastive Hypothesen*. 27-40. Frankfurt: Peter Lang
- Fabb, N. 1998. Compounding. Spencer A. and A. M. Zwicky eds., *Handbook of morphology*, 66–83. Oxford: Blackwell
- Goldberg, Adele. 1995. *Constructions: A Construction Grammar approach to argument structure*. Chicago: The University Press of Chicago.
- Groß, T. 2010. Chains in syntax and morphology. In *Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation at Tohoku University*, eds. O. Ryo, K. Ishikawa, H. Uemoto, K. Yoshimoto & Y. Harada, 143-152. Tokyo: Waseda University.
- Groß, T. and T. Osborne 2009. Toward a practical DG theory of discontinuities. *Sky Journal of Linguistics* 22. 43-90.
- Halle, M. and A. Marantz. 1993. Distributed Morphology and the Pieces of Inflection. Hale, K. and S. J. Keyser eds., *The View from Building 20*, 111-176, Cambridge: MIT Press.
- Harley, H. and R. Noyer. 2003. Distributed Morphology. In *The Second GLOT International State-of-the-Article Book*. Berlin: Mouton de Gruyter. 463-496.
- Harnisch, R. 2003. Verbabhängige Morphologie und Valenzmorphologie der Subjekt-Verb-Kongruenz. Ágel, V. et al. (eds.), *Dependency and valency: An international handbook of contemporary research*, vol. 1, 411-421. Berlin: Walter de Gruyter.
- Hays, D. 1964. Dependency theory: A formalism and some observations. *Language* 40. 511-525.
- Heringer, H-J. 1970. Einige Ergebnisse und Probleme der Dependenzgrammatik. *Der Deutschunterricht* 4. 42-98.
- Hudson, R. 1984. *Word Grammar*. New York: Basil Blackwell.
- Hudson, R. 1990. *An English Word Grammar*. Oxford: Basil Blackwell.
- Hudson, R. 2007. *Language networks: the new Word Grammar*. Oxford University Press.
- Kaisse, E. M. 19985. *Connected Speech*. New York: Academic Press.
- Klavans, J. L. 1985. The Independence of syntax and phonology in cliticization. *Language* 61, 95-120.
- Maxwell, D. 2003. The concept of dependency in morphology. Ágel, V. et al. (eds.), *Dependency and valency: An international handbook of contemporary research*, vol. 1, 678-684. Berlin: Walter de Gruyter.
- Mel'čuk, I. 1988. *Dependency syntax: Theory and practice*. Albany: State University of New York Press.
- Mel'čuk, I. 2003. Mel'čuk, I. 2003. Levels of dependency in linguistic description: concepts and problems. In Ágel, V. et.al (eds.), *Dependency and valency: an international handbook of contemporary research*, vol. 1, 188-229. Berlin: Walter de Gruyter.
- Mel'čuk, I. 2006. *Aspects of the Theory of Morphology*. Berlin, New York: Mouton de Gruyter.
- Müller, S. 2003. Solving the bracketing paradox: an analysis of the morphology of German particle verbs. *Journal of Linguistics* 39. 275–325.
- Nevis, J. A. 1986. *Finnish Particle Clitics and General Clitic Theory*. Working Papers in Linguistics 33. Columbus, Ohio: Dept. of Linguistics, Ohio State University.
- O'Grady, W. 1998. The syntax of idioms. *Natural Language and Linguistic Theory* 16. 79-312.
- Olsen, S. 2000. Composition. Geert B., C. Lehmann and J. Mugdan eds. *Morphologie-morphology*, 897–916. Berlin/New York: Mouton de Gruyter
- Osborne, T. 2005. Beyond the constituent: a dependency grammar analysis of chains. *Folia Linguistica* 39/3-4: 251-297.
- Osborne, T., M. Putnam and T. Groß (in press). Cate-nae: Introducing a novel unit of syntactic analysis. *Syntax*.
- Pesetsky, D. 1985. Morphology and logical form. *Linguistic Inquiry* 16:193-246.
- Scalise, S. and A. Bisetto. 2009. The classification of compounds. Lieber, R. and P. Štekauer eds., *The Oxford handbook of compounding*, 34–53. Oxford: Oxford University Press.
- Spencer, A. 1988. "Bracketing paradoxes and the English lexicon." *Language* 64:663-682.
- Sproat, R. 1988. Bracketing paradoxes, cliticization, and other topics: The mapping between syntactic and phonological structure. In Everaert et al. (eds), *Morphology and Modularity*. Amsterdam: North-Holland. 339-360.
- Stump, G. T. 1991. A paradigm-based theory of morphosemantic mismatches. *Language* 67/4. 675-725.
- Stump, G. T. 2001. *Inflectional Morphology: A Theory of Paradigm Structure*. Cambridge: Cambridge University Press.
- Sternefeld, W. 2009. *Syntax: Eine morphologisch motivierte generative Beschreibung des Deutschen*.

Band 2. Third Edition. Tübingen: Stauffenberg.

Ten Hacken, P. 2000. Derivation and compounding. Geert B., C. Lehmann and J. Mugdan eds. *Morphologie-morphology*, 349–60. Berlin/New York: Mouton de Gruyter.

Williams, E. 1981. On the notions ‘lexically related’ and ‘head of a word’. *Linguistic Inquiry* 12. 245-274.

Zwicky, A. M. 1987. Suppressing the Zs. *Journal of Linguistics* 23, 133-148.



Clitics in Dependency Morphology

Thomas Groß

Aichi University, Nagoya, Japan

tmgross@vega.aichi-u.ac.jp

Abstract

Clitics are challenging for many theories of grammar because they straddle syntax and morphology. In most theories, cliticization is considered a phrasal phenomenon: clitics are affix-like expressions that attach to whole phrases. Constituency-based grammars in particular struggle with the exact constituent structure of such expressions. This paper proposes a solution based on catena-based dependency morphology. This theory is an extension of catena-based dependency syntax. Following Authors et.al. (in press), a word or a combination of words in syntax that are continuous with respect to dominance form a *catena*. Likewise, a morph or a combination of morphs that is continuous with respect to dominance form a *morph catena*. Employing the concept of morph catena together with a hyphenation convention leads to a parsimonious and insightful understanding of cliticization.

1 Introduction

“Dependency morphology” was a short-lived affair. Anderson (1980) coined this label in his attempt to extend the dependency-based structuring of syntax to morphology. Yet even earlier, Heringer (1970: 96) considered the possibility of individual morphs entertaining dependency relationships. Morphological dependency structures crop up occasionally (Heringer 1973:283-294, 1996:117f, Eroms 2010: 38f), but a consistent discussion of morphological structure is curiously lacking from dependency-based approaches in general. The only exceptions are Mel’čuk (1988: 107, 2003: 193f.), where morphological dependency is discussed in detail, and within the Word Grammar framework of Creider and Hudson (1999) and Hudson (2003: 514, 518).¹ Due to this dearth of solid dependency-based explorations into morphological structure, it is not surprising that Maxwell (2003) bases his account of dependency concepts in morphology entirely on constituency-based proposals.

The possibility of complex words being struc-

ured in much the same fashion as sentences was proposed first in Williams (1981), and further discussed in the famous “head-debate” between Zwicky (1985a) and Hudson (1987). In contemporary morphological theories that attempt to inform syntax (predominantly within the generative framework) such as Di Sciullo (2005) and the theory of Distributed Morphology (Halle and Marantz 1993, Embick and Noyer 2001, 2007, Harley and Noyer 2003, Embick 2003), words are now seen as hierarchically structured items.

Seen in the light of this development, it is time for dependency grammar (DG) to make up for its neglect of morphological matters. The assessment by Harnisch (2003) that the development of a dependency-based morphology requires immediate attention is accurate. In this spirit, a proposal for a dependency-based morphology is sketched in the next section. The central idea builds on the notion of syntactic *catenae* as defined by Osborne et.al. (in press). Concepts defined in Section 2 are then used to address clitics.

2 Catena-based morphology

Adapting the definition of syntactic *catenae* by Osborne et.al. (in press), a *morph catena* is a MORPH OR A COMBINATION OF MORPHS THAT IS CONTINUOUS WITH RESPECT TO DOMINANCE. This definition identifies any morph tree or subtree of a morph tree as a morph catena. The choice of “morph” instead of “morpheme” is motivated by the need to maintain a surface-oriented level of analysis.² A *morph* is loosely defined as any meaning bearing unit that cannot be reduced any further, but that can be segmented from other meaning bearing units in the horizontal AND/OR vertical dimension. The inclusion of the notion “vertical dimension” allows for the treatment of phenomena subsumed under non-concatenative morphology. For reasons of space, however, non-concatenative morphology is not addressed in this paper.

If one wishes to see the interactions of morphs in the same manner as the interactions of words,

¹ In MTT morphological dependencies operate at strata entirely different from syntactic dependencies. In Word Grammar, morphology is feature-based, rather than morph-based.

² While there are certainly difficulties with the notions “morph” and “morpheme” (cf. Mel’čuk 2006: 384ff), the proposal here is sufficient in the present context.

then one must first distinguish dependency relationships between morphs within the same word, and then second between morphs across separate words.

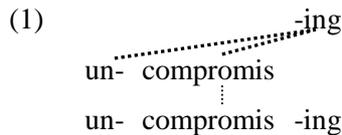
2.1 Intra-word dependencies

A dependency relationship between morphs inside the same word is called an *intra-word* dependency. Intra-word dependencies are determined by distribution. The formal definition is presented first:

Intra-word dependency

A morph M_1 is an intra-word dependent of another morph M_2 , if the morph combination M_1 - M_2 distributes more like an M_2 -type unit than like an M_1 -type unit.

This definition is similar to Mel'čuk's definition of "surface syntactic dominance" (2003: 200f). The next example illustrates intra-word dependencies:



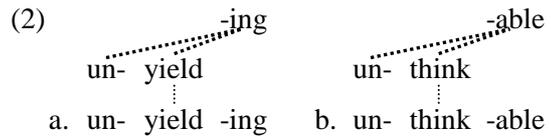
The intra-word dependencies are represented by dotted edges (as opposed to solid edges). Only the lexical morph *compromis* receives a (vertical) projection edge.

Hyphens are an important tool in this account. They represent prosodic dependencies (in the horizontal dimension). For instance, the negation prefix *un-* prosodically depends on the next morph to its right (here: *compromis*). The progressive suffix *-ing* prosodically depends on the next morph to its left (here: *compromis*).

A morph must receive either a hyphen or a projection edge, but never both. Morphological affixes always receive a hyphen, and therefore they can never receive a projection edge.

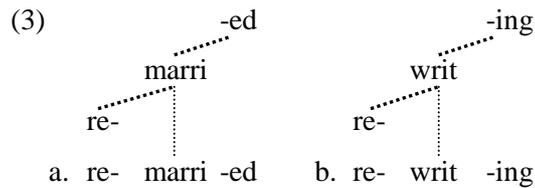
Reexamining example (1), the peripheral morphs are affixes and must therefore appear with hyphens and dotted edges. Note that the progressive immediately dominates both the prefix and the lexical morph. The progressive suffix dominates the lexical morph because *compromising* is a valid word. The expression **un-compromise*, however, does not exist, hence the prefix cannot depend on the lexical morph. Rather the negative prefix must depend on a morph that has some adjectival features. Since the progressive morph can appear as an adjective-like expression, such as *an uncompromising person*, the negative prefix must depend on the progres-

sive suffix. Further examples of this ilk are shown below:



Since **un-yield* and **un-think* are bad, the prefixes must depend on the final (adjectival) morphs *-ing* and *-able*.

Somewhat different structures from those in (1) and (2a-b) appear with the prefix *re-* in *re-marri-ed* and *re-writ-ing*:



The analyses in (3a-b) are correct because the expressions *re-marry* and *re-write* are good.

2.2 Inter-word dependencies

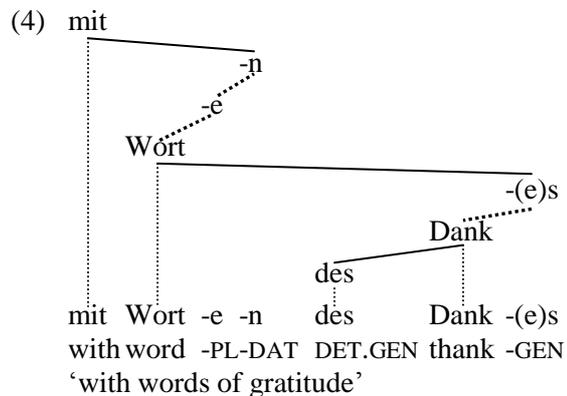
An inter-word dependency is a morphosyntactic relationship between a morph and a word. If the morph licenses the appearance of the word, the morph *governs* the word. The formal definition is again presented first:

Inter-word dependency (government)

A morph M in a word W_1 governs another word W_2 , if M licenses the appearance of W_2 .

This definition is similar to Mel'čuk's omissibility and cooccurrence properties of syntactic dominance (2003: 205).

The next example from German illustrates two inter-word dependencies:



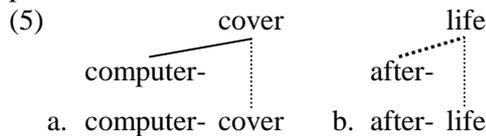
Example (4) shows two instances of inter-word dependency relationships. The first concerns the morph *mit* and the word *Wort-e-n*. The structure of the latter is established independently through intra-word dependency: *Wort-e* distributes like

any plural noun, and *Wort-e-n* distributes like any dative marked noun. The preposition *mit* is both a morph and a word. Because this prepositional morph only licenses *Wort-e-n*, but not *Wort* or *Wort-e*, *mit* governs *Wort-e-n*.

The second inter-word dependency concerns the morph *Wort* and the word *Dank(-e)s*. The bracket indicates the phoneme /e/ is optional. The morph *Wort* requires the masculine noun *Dank* to appear with a genitive case suffix (here: *-(-e)s*). In other words, the morph *Wort* licenses the appearance of *Dank(-e)s*, but not of *Dank*. The dependency relationship between the article *des* and *Dank(-e)s* is purely syntactic.

2.3 Compound structure

A lexical morph does not automatically receive a projection edge. In some cases, lexical morphs appear very similar to affixes, barring their meaning, of course. Compounding is a case in point:



In (5a), the initial morph *computer-* is certainly a lexical morph because it can appear on its own. The initial *after* usually appears as a preposition. Nevertheless, in *computer-cover* and *after-life*, both *computer-* and *after-* have lost the ability to stand alone and have been integrated into their respective compound. The hyphens symbolize the inability to constitute a prosodic word alone.

The next matter concerns the angled dependency edges. In (5a) the dependency edge is solid, much like a syntactic dependency edge. In (5b) however, the dependency edge is dotted. This distinction addresses a semantic difference. In (5a) *computer-* is still subject to further modification, as in *desk-top-computer-cover*, where the computer is of the desktop type. The morph *after-* in (5b), however, cannot undergo modification. In *after-life*, *after-* functions much in the manner of a lexical prefix. On the other hand, *computer-* in (5a) lies between a pure syntactic dependency relationship and the type of morphological relationship that affixes have with their lexical morphs.

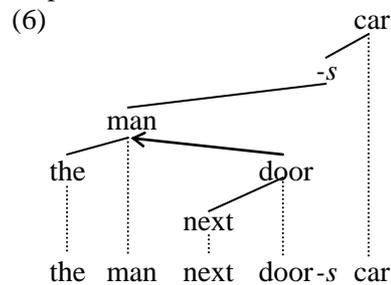
In compounds, a non-initial compound part must appear with a hyphen and the dependency edge must be solid if this compound part can still be modified, or it must be dotted if modification is impossible.

The distinctions drawn above open the door to a principled analysis of clitics. Clitics share much with initial compound parts such as *computer-* in (5a): *computer-* has lost its ability to constitute a prosodic word. Clitics never constitute prosodic words. Therefore all clitics must receive a hyphen. While *computer-* in (5a) has retained much of its semantic autonomy, clitics are syntactically autonomous. Therefore the dependency edge of a clitic must be solid, as opposed to a dotted edge which connects affixes to lexical morphs (or other affixes).

3 Clitics

Clitics are morphs on the borderline between free and bound morphs (Zwicky 1977, 1985b, 1987, Klavans 1985, Kaisse 1985, Borer 1986, Nevis 1986, Anderson 1992, 2005, Halpern 1995, 1998, Halpern and Zwicky 1996, Gerlach 2002, Hudson 2007:104f). Clitics express meanings usually reserved for free morphs, but fail – for whatever reasons – to appear as individual prosodic words. In the current system, these properties are expressed by the following tree conventions: A clitic appears with a hyphen and a solid dependency edge but without a projection edge.

This convention is illustrated with the next example:

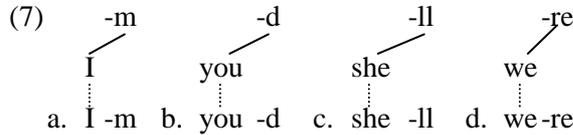


The (italicized) possessive *-s* depends on the following noun *car*, seemingly like a full word. It also governs the noun *man* like a full noun. However, the clitic appears without a projection edge in exactly the fashion affixes would. Like affixes, the clitic is prosodically dependent on a morph capable of constituting a prosodic word (here: *door*), or it must depend on a morph that depends on such a morph, and so on, recursively.

Clitics also subsume contractions, cf. Zwicky and Pullum (1983). The parts after the apostrophe in English *I'm*, *you'd*, *she'll*, *we're*, etc. are cliticized to the pronouns³. The phonological reduction of the auxiliaries causes them to be-

³ Pronouns are used here for simplification. But cliticization to other word classes is possible (cf. Zwicky and Pullum 1983: 504).

come prosodically dependent on the pronominal morphs. Hence a hyphen is required for the reduced auxiliaries. A solid dependency edge must connect the pronominal morphs and the contractions because the latter are still syntactically autonomous. Their structure is shown next:



Even though the reduced auxiliaries are prosodically dependent on their pronouns, they dominate their pronouns as they would if not reduced. Many clitics, though, do not entertain dependency relationships with their hosts. Prosodic dependency and dominance are therefore logically independent properties. The necessity to distinguish these two dimensions is addressed in the next section.

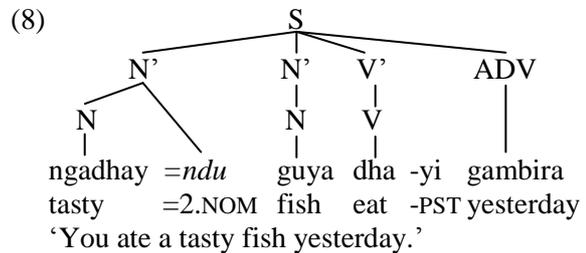
3.1 Horizontal and vertical dimension

Comparing affixes and clitics, one sees an important difference: affixes must entertain intra-word dependency relationships with morphs contained WITHIN the same word. I.e. for a morph to be an affix, this morph must either be an intra-word dependent or an intra-word head of another morph contained within the same word. Examine again the word *Wort-e-n* of example (4): the morphs *-e* and *-n* are affixes because they must reside within the prosodic word structure constituted by the lexical morph *Wort*. The plural suffix *-e* dominates and prosodically depends on the morph *Wort*. As a result, this suffix is integrated into the prosodic word structure of the morph *Wort*. The dative suffix *-n* dominates and prosodically depends on the plural suffix *-e*. Because the plural suffix is already a part of the prosodic word structure of *Wort*, the dative suffix can – via prosodic dependency – be integrated into the same prosodic word structure. In sum, prosodic and dependency structure coincide for affixes.

In cliticization, however, prosodic dependency structure and standard dependency structure are logically independent. The prosodic dependency preference of a clitic says nothing about the hierarchical status of that clitic. Since a description of clitics requires the distinction between prosodic dependency (a linear/horizontal order phenomenon) and standard dependency/dominance (a vertical order phenomenon), those grammars that do not sufficiently distinguish between these dimensions experience considerable difficulties

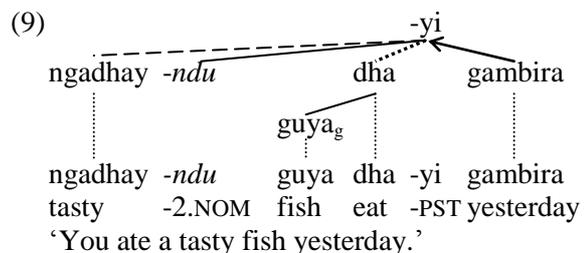
when attempting to produce an insightful and intuitive treatment of clitics.

These difficulties are evident in two diametrically opposed approaches to syntax. The first approach is epitomized by constituency-based grammars. The apparatus in GB-theory, for instance, is geared toward generating the appropriate word order, i.e. the linear order of utterances. In the account of Klavans (1985) for instance, the GB apparatus leads her to posit parameters that exclusively apply to the horizontal dimension. Klavans' (1985: 97f) posits “dominance”, “precedence”, and “phonological liaison”.⁴ These concepts are illustrated with a Ngiyambaa example (a Pama-Nyungan language spoken in New South Wales) taken from Klavans (1985: 101, her tree conventions and gloss):



According to Klavans (1985: 97f), the italicized clitic *=ndu* appears in a domain characterized by the following parameters: “dominance” is “initial” because the clitic appears with the first constituent under S. “Precedence” is “after” because the clitic appears after the first constituent under S. “Phonological liaison” is “enclitic” because the first constituent is also the host of the clitic.

The structure proposed in (8) displays one serious problem concerning the first N', however: it does not in any insightful manner clarify the dependency structure of the clitic. The reason for this insufficiency is the inability of constituency-based grammars to represent dependency relationships. The clitic should, namely, depend on the verb *dha-yi*. A dependency tree of (8) is shown next:



The adjective *ngadhay* has undergone *rising* (see Section 3.2) due to the splitting of the NP. The

⁴ Anderson (2005: 81) suggests “anchor” instead of “dominance”.

adverb *gambira* receives an arrow, the arrow-head pointing towards the head; this type of dependency edge marks adjuncts (cf. ex.6). The comparison of (8) and (9) shows that (9) accurately displays the relevant information concerning the clitic *-ndu*: *-ndu* depends on the tense suffix on the verb (dependency structure), AND it prosodically depends on to the preceding adjective. Klavans' (8) suggests, however, that the clitic is somehow part of the constituent formed by the adjective. This assumption is wrong, but the motivation by which one arrives at this assumption is clear. Unlike the current dependency-based apparatus, the constituency-based apparatus employed in (8) is not capable of representing both the syntactic and prosodic relationships simultaneously.

Examples of the second type of the two diametrically opposed approaches are dependency-based grammars that see linear order as derived, e.g. Mel'čuk's MTT or dependency-based topology models (Duchier and Debusmann 2001, Gerdes and Kahane 2001, 2006). In general, this type of grammar has no problem representing dependency structure, but must derive linear order by an additional topological model. With respect to cliticization, it is difficult to assess this approach fairly because only Gerdes and Yoo (2003) seem to address the matter (focusing on Modern Greek). They posit a clitic field within an embedded domain. Due to the scarcity of information on this matter within topological models, it is impossible for me to present a topology-based structure of (8).

It may be relatively safe to assume, though, that topological models are going to face problems with K^wak^wala clitics. Consider the next fragment from an example by Anderson (2005: 16):

- (10) a. *yəlk^wəmas -ida bəgwanəma -x -a...*
 cause hurt -DEM man -OBJ-DEM
 'The man hurt [the dog with a stick].'

Even though the italicized demonstrative clitic prosodically depends on the preceding verb *yəlk^wəmas* 'cause hurt', it modifies the following noun *bəgwanəma* 'man'. Similarly, the two clitics *-x-a* do not modify the preceding noun *bəgwanəma* 'man' to which they attach, but rather they modify a following noun (which is not shown). Constituency-based models as well as topological models must now reconcile two different structures: prosodic and constituent structure:

- (10) b. [*yəlk^wəmas -ida*] [*bəgwanəma...*]
 [cause hurt -DEM] [man...]
 c. [*yəlk^wəmas*] [*-ida bəgwanəma*]...
 [cause hurt] [-DEM man]

(10b) shows the prosodic word structure; the clitic *-ida* is shown as a part of the prosodic word structure of the verb. The noun constitutes a separate prosodic word, of which the clitic is NOT a part. (10c) shows the constituent structure: here the clitic forms a constituent with the noun. In this structure, the clitic is excluded from the word structure of the verb.

Prima facie it is not evident how one proceeds from the prosodic structure (10b) to the dependency structure (10c), which is what constituency-based grammars would like to accomplish. Nor is it clear how topological models might distinguish the prosodic/topological structure (10b) from the dependency structure (10c), which they see as primary.

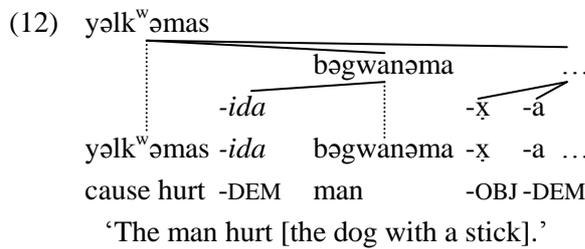
Topological models might point to the fact that K^wak^wala clitics are enclitics, and they must therefore prosodically depend on immediately preceding material. The distinction between proclitics and enclitics, while self-evident at first blush, is not as clear-cut as it seems. In some languages, one and the same clitic can appear with both orientations, a fact that blocks any attempt at positing universal orientation preferences. The next European Portuguese example, taken from Anderson (2005: 85), shows that orientation preference is not a property inherent to the clitic, but rather that it is contingent on the prosodic context:

- (11) a. *Só o Pedro o- viu.*
 only ART Pedro him-saw
 'Only Pedro saw him.'
 b. **Só o Pedro viu-o.*
 c. *Viu-o só o Pedro.*
 d. **O-viu só o Pedro.*

(11a) shows the object clitic *o-* as a proclitic. (11b) shows that this clitic may not follow the final verb. (11c) shows that it must be enclitic on an initial verb, but may not precede the initial verb (11d). A topological modal can, of course, simply posit respective clitic fields after an initial verb field, and before a final verb field. Doing so, however, seems *ad hoc*. The contingency that the prosodic context poses (for a clitic to appear as a proclitic as opposed to an enclitic, or vice versa) does not – in any discernible way – follow from its dependency structural context. In contrast, Klavans' (1985) account can easily provide a

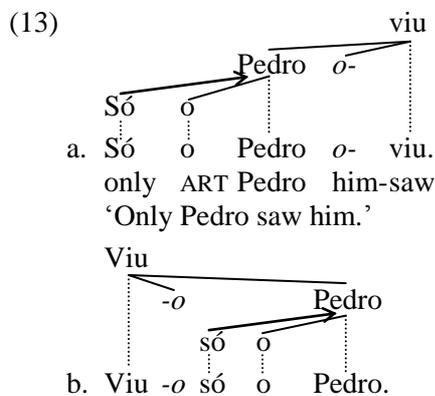
systematic distinction between the cases (11a) and (11c), and rule out the cases (11b) and (11d).

The examples from Ngiyambaa, K^wak^w’ala, and European Portuguese and the difficulties they pose force one to the assumption that linear/horizontal order and dominance/vertical order are ultimately distinct, and that neither is derivable from the other. The current theory accommodates this insight by positing two different tools to represent these distinct dimensions: hyphens for linear order, and solid, dotted, or dashed (in case of rising) dependency edges for vertical order. Reexamining the K^wak^w’ala data from (10a), the current theory can provide a tree representation that visualizes the linear (prosodic) relationships and the vertical (dominance) relationships:



The clitic is marked in two ways, the one way indicating its prosodic dependency and the other its standard vertical dependency. The hyphen on its left side indicates that *-ida* must prosodically depend on the initial verb. The solid dependency edge, however, indicates that it is dependent on the noun. Equally for the clitics *-x-a*: *-x* prosodically depends on *bəgwanəma*, and *-a* prosodically depends on *-x*. Hence both clitics end up integrated into the prosodic word structure of *bəgwanəma*. These clitics depend, however, on a following noun (not shown), which they modify.

The European Portuguese example receives an equally parsimonious analysis: (11a) is shown as (13a), and (11c) as (13b):



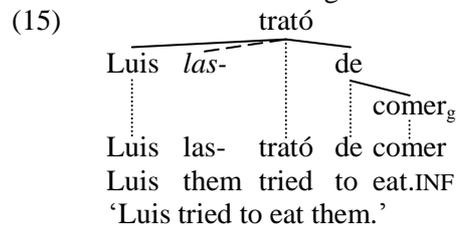
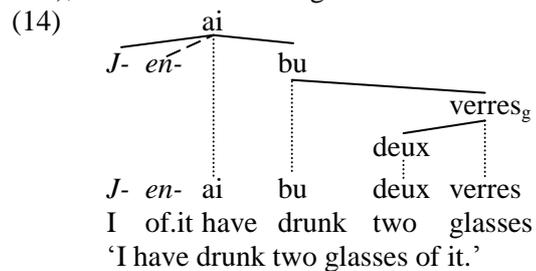
The fact that (11b,d) are ungrammatical has nothing to do with clitic orientation preference,

rather orientation is forced on the clitic by the prosodic context of its head, the verb *viu*. If the verb is in V1 position, the clitic must appear as an enclitic; if the verb is in VF position, then the clitic must appear as a proclitic.

3.2 Clitic rising

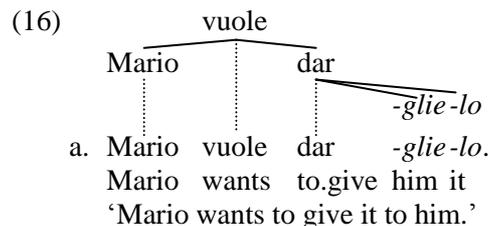
A well known fact is that clitics can exhibit displacement. This phenomenon is known as “clitic climbing”. Building on the work of Groß and Osborne (2009), displacement is understood here as *rising*. The displaced catena is seen as *risen*, which is indicated by the dashed edge. The governor of the risen catena is marked with a g-subscript. The *Rising Principle* states that the head or the root of a risen catena must dominate the governor of that catena. Clitics fully obey this principle when they appear displaced.

Clitic rising is well documented throughout the Romance language family. A French and a Spanish example, taken from Halpern (1998: 106), illustrate clitic rising:



In the French example (14), two clitics appear: the subject clitic *J-* and the clitic *en-*. The latter has risen, its governor being *verres*. The Spanish example (15) shows the object clitic *las-* as risen, its governor being *comer*.

Some languages require all clitics to either rise or stay. Italian is such a language, as the next example demonstrates (taken from Anderson 2005: 246f):



-
- b. Mario *gliē-lo-* vuole dar.
- c.*Mario *lo-*vuole dar-*gliē*.
- d.*Mario *gliē-*vuole dar-*lo*.

(16a) shows both clitics dominated by their governor *dar*. (16b) shows both clitics as risen: they are now dominated by *vuole*, which dominates their governor *dar*, thus obeying the Rising Principle. (16c,d) show that individual rising is ungrammatical. Either no clitic rises, or all clitics rise.

Surmiran, a dialect of the Romansh language group (Switzerland), allows clitic rising, but disallows multiple occurrences of clitics. The data are again from Anderson (2005: 247f):

- (17)
-
- a. Ia vi dar el ad ella.
I want to.give it.m to her
'I want to give it to her.'
-
- b. Ia *igl-* vi dar ad ella.
it.m
-
- c. Ia *la-* vi dar el.
to.her
- d.*Ia *igl-la-*vi dar.
- e.*Ia *la-igl-*vi dar.

Example (17a) does not contain clitics, nor does it exhibit rising. In (17b), the direct object clitic *igl-* rises to attach to the matrix verb *vi*. In (17c), it is the indirect object clitic *la-* that rises and attaches to *vi*. Examples (17d,e) show that multiple rising of clitics is disallowed. (17f,g) show that the occurrence of multiple clitics is bad.

3.3 Clitic doubling

Another phenomenon which merits attention is “clitic doubling”. Clitic doubling obtains when a clitic co-occurs with a full NP carrying the same grammatical function. While French prohibits

clitic doubling, Spanish clitic doubling is sensitive to a variety of criteria. Clitic doubling is optional in the presence of an indirect object or an animate direct object, both preceded by the preposition *a*. But doubling of an inanimate direct object without this preposition is ungrammatical. And with a pronominal object, doubling is obligatory. Four examples from Halpern (1998: 107f) illustrate the differences:

- (18) a. (*le-*) puso comida al canario.
him put.3sg food to.the canary
'S/he gave food to the canary.'
- b. (*la-*) oían a Paca.
her listened.3pl to Paca
'They listened to Paca.'
- c.**lo-* compró el libro.
it bought.3sg the book
'S/he bought the book.'
- d. ellos *(*la-*) llamaron a ella.
they her called.3pl to her
'They called her.'

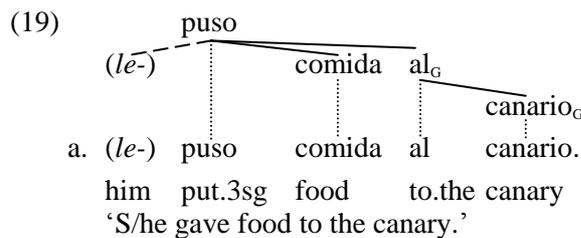
Here the clitics are italicized and their doubles underlined. The brackets on the clitics indicate that the occurrence of the clitic is optional. In (18a), *al canario* is the indirect object; since the preposition is present, optional doubling is grammatical. (18b) shows the direct object *a Paca*. Here, too, optional doubling is allowed. In (18c) the direct object *el libro* is inanimate and the preposition *a* is absent. Hence doubling is ungrammatical. (18d) shows the pronominal object *a ella*. Here the asterisk indicates that optionality is ungrammatical; clitic doubling must occur in this case.

While it is understood that clitic doubling is sensitive to animacy and specificity, such that animate objects and specified objects allow clitic doubling, while inanimate objects and unspecified objects disallow it, the status of the clitic in terms of syntax and subcategorization remains beyond principled understanding (see the discussion in Halpern 1998: 107f). Concerning the syntactic status of doubling clitics, the traditional view is to treat them as adjuncts. This assumption, however, causes problems with subcategorization, in particular concerning case assignment.

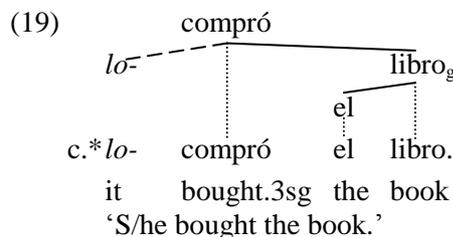
In order to explain the Spanish examples (18a-d) an augmentation of the notion *governor* is necessary. Two facts back this step: first, clitic doubling in Spanish occurs in the presence of the preposition *a*. Second the pronominal clitics are sensitive to animacy (and pronominal) features. These two facts imply that neither the preposition *a* nor the nominal governed by this preposi-

tion alone suffice as the governor of the clitic. The combination of the preposition *a* AND the nominals, however, does fulfill all requirements for a governor of the clitics. The preposition *a* and the nominal qualify as a catena, hence they constitute the *governor catena* of the clitic.

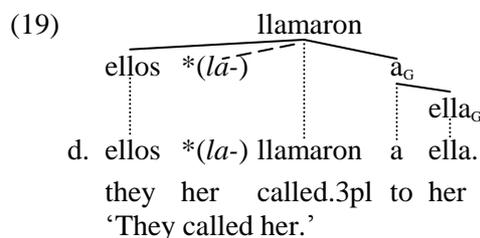
The second issue concerns the syntactic status of the clitics. As long as the clitics are optional, they are seen as adjuncts. The dependency edges of optional clitics must therefore be arrows (cf. ex. 6, 9, 13). An analysis of the Spanish examples (18a-d) is now provided:



The governor catena is the word combination *al canario*. Both words receive a G-subscript, which is capitalized to help indicate that the entire catena is the governor of the clitic *le-*. Finally, rising must obtain (because the clitic is separated from its governor catena) so that domination is impossible. Note that the Rising Principle is obeyed: *puso*, the head of the clitic *le-*, dominates the governing catena *al canario* of the clitic. A similar analysis also holds for (18b).



(19c) is bad because the governing catena of the clitic is incomplete; the preposition *a* being absent; case cannot be assigned to the clitic.

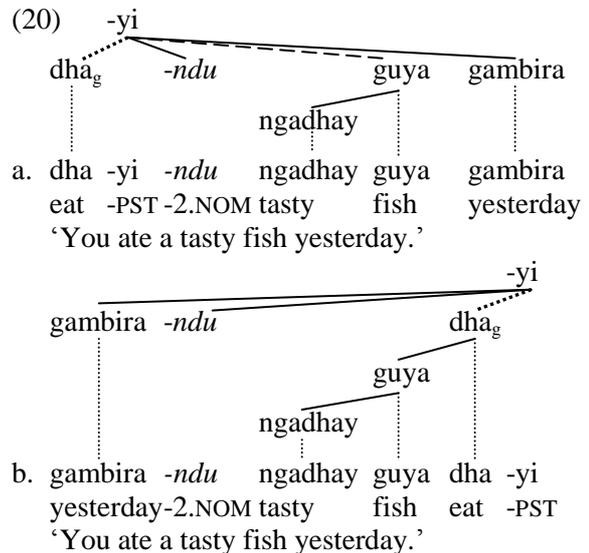


Here, the governor catena is *a ella*.

3.4 Second position clitics

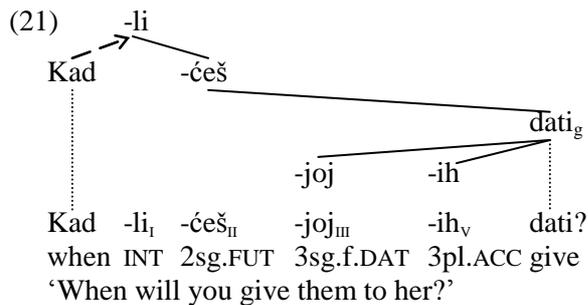
“Wackernagel” or “second position” clitics challenge many theories. In quite a number of languages, clitics tend to cluster in a position rough-

ly called the “second position” or the “Wackernagel position”. Ngiyambaa (cf. ex.8, 9) is a case in point. The subject clitic *-ndu* ‘2.NOM’ must appear after the first prosodic word, regardless of that word’s category or syntactic function. Therefore, change of word order does not affect the positional appearance of the clitic as the next examples taken from Klavans (1985: 101) demonstrate:



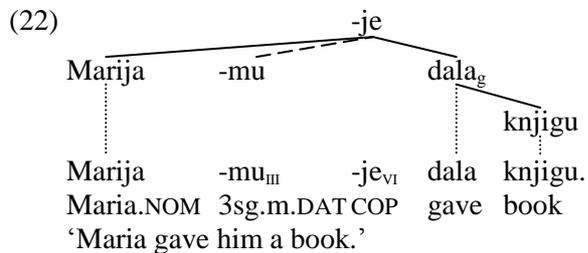
The difference between (9) and (20a,b) is a matter of focus. The first position is a focus position. Hence the adjective *ngadhay* ‘tasty’ is focused in (9), the verb *dha-yi* ‘ate’ in (20a), and the adverb *gambira* ‘yesterday’ in (20b). Regardless, the subject clitic must prosodically depend on the first prosodic word. Its dependency structure, however, is constant because it must always depend on the verb.

In Serbo-Croat, multiple clitics appear in second position, obeying a specific order. Following Corbett (1987: 406), the Serbo-Croat second position has six slots in the following order: I. interrogative *-li*, II. verbal auxiliaries, III. dative, IV. genitive, V. accusative (weak) pronouns, and VI. *-je*, the 3sg copula. The following dominance order among these slots can be assumed: the first slot dominates everything else; slot II tends to dominate to the right, but depends to the left on a slot I clitic if such a clitic is present. Slots III-V are dependent to the left, but can undergo clitic climbing. Slot VI tends again to dominate everything else. The plausibility of this assumption is now illustrated with two examples taken from Halpern (1998: 109). The indices on the clitics indicate their slot position.

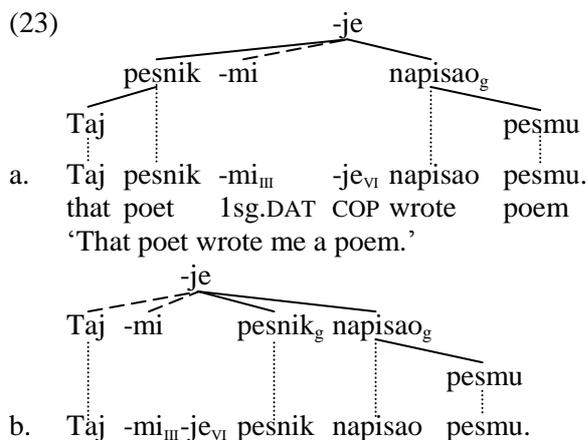


The slot II clitic depends on the slot I clitic which is the root. The slot II clitic dominates the infinitive verb to the right. The pronominal clitics of the slots III and V depend on the verb to the right (as would slot IV clitics). The question word *Kad* has risen.

The next example shows that the slot VI copula *-je* must again dominate other material:



The pronominal clitic *-mu* must rise, its governor being *dala*. Note that in (21, 22), the clitics invariably depend prosodically on the first prosodic word, but that the clitics’ dependency structure can vary considerably. The Serbo-Croat second position is purely defined in prosodic terms: the second position is located after the first stressed prosodic unit. This can lead to NP splitting as the next example from Corbett (1987: 406) illustrates:



In (23a) the first stressed prosodic unit is the prosodic phrase *Taj pesnik*. Hence the pronominal clitic *-mi* rises in order to attach to *pesnik*; as a result it is dominated by *-je*. In (23b) one sees the splitting of the NP: here the demonstrative is the

first stressed prosodic unit. This causes the demonstrative to undergo topicalization rising. The pronominal clitic must again rise and it depends prosodically on the demonstrative.

4 Conclusion

This paper has demonstrated that cliticization can be captured parsimoniously in a catena-based dependency morphological account. Due to the capability of dependency-based structures to distinguish clearly between the horizontal dimension of precedence, where the phonology and prosody of cliticization operates, and the vertical dimension of dominance, where standard syntactic dependency operates, a unified account of cliticization is possible. Cliticization phenomena can be captured with the same notions that are already required to perform a morphological analysis within and across words. The intuition that clitics seem to operate much as individual words in syntax was shown to hold true with clitic rising, which is indistinguishable from syntactic rising. The catena and the Rising Principle together make a fluid transition from morphology to morphosyntax and further to syntax possible.

References

- Anderson, J. 1980. Towards dependency morphology: the structure of the Basque verb. In Anderson, J. and C. Ewen (eds.), *Studies in Dependency Phonology*, 221-271. Ludwigsburg.
- Anderson, S.R. 1992. *A-Morphous Morphology*. Cambridge: Cambridge University Press.
- Anderson, S.R. 2005. *Aspects of the Theory of Clitics*. Oxford/New York: Oxford University Press.
- Borer, H. 1986. *The Syntax of Pronominal Clitics*. Syntax and Semantics Vol. 19. Orlando et.al.: Academic Press.
- Creider, C. and R. Hudson. 1999. Inflectional morphology in Word Grammar. *Lingua* 107: 163-187.
- Corbett, G. 1987. Serbo-Croat. In Comrie, B. (ed.), *The World’s Major Languages*. 391-409. New York: Oxford University Press.
- Di Sciullo, A.M.. 2005. *Asymmetry in Morphology*. Linguistic Inquiry Monographs 46. Cambridge: MIT Press.
- Duchier, D. and R. Debusmann. 2001. Topological Dependency Trees: A Constrained-based Account of Linear Precedence. *ACL 2001*: 180-187.
- Embick, D. and R. Noyer. 2001. Movement operations after syntax. *Linguistic Inquiry* Vol 32, No. 4, 555-595

- Embick, D. and R. Noyer. 2007. Distributed Morphology and the Syntax/ Morphology Interface. Ramchand, G. and C. Reiss eds. *The Oxford Handbook of Linguistic Interfaces*. 289-324. Oxford University Press.
- Embick, D. 2003. Linearization and local dislocation: Derivational mechanics and interactions. *Linguistic Analysis* 33/3-4. 303-336.
- Eroms, Hans-Werner. 2010. Valenz und Inkorporation. Kolehmainen Leena, Hartmut E. Lenk and Annikki Liimatainen eds. *Infinite Kontrastive Hypothesen*. 27-40. Frankfurt: Peter Lang
- Gerdes, K. and H.-Y. Yoo. 2003. La topologie comme interface entre syntaxe et prosodie, une système de génération appliqué au grec modern. *TALN 2003*. Batz-sur-Mer, 123-34.
- Gerdes, K. and S. Kahane. 2001. Word Order in German: A Formal Dependency Grammar Using a Topological Hierarchy. *ACL 2001*: 220-227.
- Gerdes, K. and S. Kahane. 2006. A polynomial parsing algorithm for the topological model: Synchronizing Constituency and Dependency Grammars, Illustrated by German Word Order Phenomena. *ACL 2006*: 1097-1104.
- Gerlach, B. 2002. *Clitics between Syntax and Lexicon*. Amsterdam: John Benjamins.
- Groß, T. and T. Osborne 2009. Toward a practical DG theory of discontinuities. *Sky Journal of Linguistics* 22. 43-90.
- Halle, M. and A. Marantz. 1993. Distributed Morphology and the Pieces of Inflection. Hale, K. and S. J. Keyser eds., *The View from Building 20*, 111-176, Cambridge: MIT Press.
- Harley, H. and R. Noyer. 2003. Distributed Morphology. In *The Second GLOT International State-of-the-Article Book*. Berlin: Mouton de Gruyter. 463-496.
- Halpern, A. 1995. *On the Placement and Morphology of Clitics*. Stanford: CSLI.
- Halpern, A.L. 1998. Clitics. In Spencer, A. and A. M. Zwicky (eds.), *The Handbook of Morphology*. 101-122. Oxford: Blackwell Publishers.
- Halpern, A.L. and A.M. Zwicky. 1996. *Approaching Second: Second Position Clitics and Related Phenomena*. Stanford: CSLI.
- Harnisch, R. 2003. Verbabhängige Morphologie und Valenzmorphologie der Subjekt-Verb-Kongruenz. In Ágel, V. et al. (eds.), *Dependency and valency: An international handbook of contemporary research*, vol. 1, 411-421. Berlin: Walter de Gruyter.
- Heringer, H.J. 1970. Einige Ergebnisse und Probleme der Dependenzgrammatik. *Der Deutschunterricht* 4. 42-98.
- Heringer, H.J. 1973. *Theorie der deutschen Syntax*. Second edition. München: Hueber.
- Heringer, H.J. 1996. *Deutsche Syntax Dependentiell*. Tübingen: Staufenberg.
- Hudson, R. 1987. Zwicky on Heads. *Journal of Linguistics* 23, 109-132.
- Hudson, R. 2003. Word Grammar. In Ágel, V. et al. (eds.), *Dependency and valency: An international handbook of contemporary research*, vol. 1, 508-526. Berlin: Walter de Gruyter.
- Hudson, R. 2007. *Language Networks*. Oxford: Oxford University Press.
- Kaisse, E.M. 19985. *Connected Speech*. New York: Academic Press.
- Klavans, J.L. 1985. The Independence of syntax and phonology in cliticization. *Language* 61, 95-120.
- Lieber, R. 1981. *On the Organization of the Lexicon*. Bloomington: Indiana University Linguistics Club.
- Maxwell, D. 2003. The concept of dependency in morphology. In Ágel, V. et al. (eds.), *Dependency and valency: An international handbook of contemporary research*, vol. 1, 678-684. Berlin: Walter de Gruyter.
- Mel'čuk, I. 1988. *Dependency syntax: Theory and practice*. Albany: State University of New York Press.
- Mel'čuk, I. 2003. Levels of dependency in linguistic description: concepts and problems. In Ágel, V. et.al (eds.), *Dependency and valency: an international handbook of contemporary research*, vol. 1, 188-229. Berlin: Walter de Gruyter.
- Mel'čuk, I. 2006. *Aspects of the Theory of Morphology*. Berlin, New York: Mouton de Gruyter.
- Nevis, J.A. 1986. *Finnish Particle Clitics and General Clitic Theory*. Working Papers in Linguistics 33. Columbus, Ohio: Dept. of Linguistics, Ohio State University.
- Osborne, T., M. Putnam and T. Groß (in press). Cate-nae: Introducing a novel unit of syntactic analysis. *Syntax*.
- Williams, E. 1981. On the notions 'lexically related' and 'head of a word'. *Linguistic Inquiry* 12. 245-274.
- Zwicky, A.M. 1977. *In clitics*. Bloomington: Indiana University Linguistics Club.
- Zwicky, A.M. 1985a. Heads. *Journal of Linguistics* 21, 1-29.
- Zwicky, A.M. 1985b. Clitics and particles. *Language* 61, 283-305
- Zwicky, A.M. 1987. Suppressing the Zs. *Journal of Linguistics* 23, 133-148

Zwicky, A.M. and G.K. Pullum. 1983. Cliticization vs.
Inflection: English *n't*. *Language* 59 (3): 502–513.



From Structural Syntax to Constructive Adpositional Grammars

Federico Gobbo

Research Center “Informatica Interattiva”

University of Insubria, Varese, Italy

federico.gobbo@uninsubria.it

Marco Benini

DICOM

University of Insubria, Varese, Italy

marco.benini@uninsubria.it

Abstract

The importance of the research made by Tesnière (1959) for the concepts of dependency and valency cannot be underestimated. However, his Structural Syntax remains still uninvestigated in most part. In this paper, a formal grammar model that follows Tesnière’s intuitions and concepts as much as possible is proposed. This model is called constructive adpositional grammar. This paper explains the linguistic and formal reasons behind such a research plan.

1 Introduction

Research in dependency linguistics acknowledges a lot from the work by Lucien Tesnière, the French linguist who introduced, in modern times, the key concepts of dependency and valency. Nonetheless, unlike valency, there is no agreement among scholars and specialists on how to treat precisely the concept of dependency. In fact, even if the theoretical assumption behind all dependency-based models is fairly the same, namely “the syntactic structure of sentences resides in binary asymmetrical relations holding between lexical elements” (Nivre, 2005, 6), in practice the use of this assumption is different among authors. For example, in Topological Dependency Grammar (TDG), proposed by Debusmann (2001), there are two different forms of dependencies, called “syntactic dependency tree (ID tree)” and “topological dependency tree (LP tree)”, while Mel’čuk (1988) postulates three types of syntagmatic dependency relations: semantic dependency, syntactic dependency and morphological dependency. How did it all begin? In other words, how Tesnière *really*

defined dependency? What can be saved – and adapted – for a dependency-based linguistic model that is formally feasible with modern mathematical and computational tools?

2 Governor, dependent, connection

Tesnière (1959) can be considered the *summa* of his work, being more than 600-pages long, where his language analysis system, called Structural Syntax (in French: *syntaxe structurale*), is explained in detail.¹ That work was published posthumously, and for this reason it is not always coherent in all parts; however, every paragraph is numbered referring to a Chapter that belongs to an internal Book (from A to F) belonging to a Part (from 1 to 3). In the sequel, references to that work will take the original form. For instance, paragraphs 1–8 of Chapter 21 belonging to Book A of Part 1 will be referred like this: (1, A, ch. 21, par. 1–8). Analogously, it was decided to retain the original numbers of Tesnière’s examples (*stemma*) in order to help the reader in the comparison between Structural Syntax and the model presented in this paper, while new examples are numbered through capital letters.

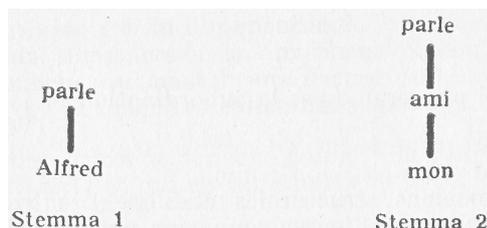


Figure 1: How connection works for Tesnière

¹Unfortunately Tesnière (1959) is still not available in English. All English translations are proposals written especially for this paper.

In (1, A, ch. 1, 4-12) the notion of connection (*connexion*) is presented. In Figure 1 the first examples of Tesnière (1959) are shown: in *Alfred parle* ('Alfred speaks'), the verb *parle* is the governor (*régissant*), the noun *Alfred* being the dependent (*élément subordonné*). Their relation, "indicated by nothing" (1, A, ch. 1, 4) is their connection. Connections can be recursive, as in example 2 *mon ami parle* ('my friend speaks'): governors are put above, dependents are put thereunder. It is interesting to note that Tesnière did not use the word 'dependency' (*dependance*) but 'connection'. This choice becomes clear when the dichotomy 'structural vs. semantic influence' is introduced (1, A, ch. 2, 3).

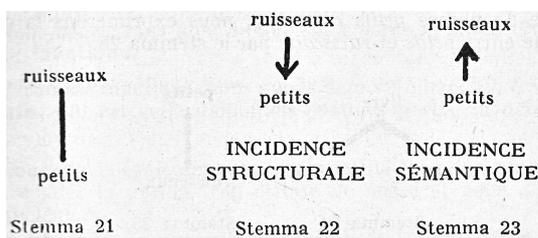


Figure 2: Structural and semantic influence

Figure 2 shows that two connections between the elements of *petits ruisseaux* ('little streams') are possible: either the governor *ruisseaux* structurally influences the dependent *petits*, or the dependent semantically influences the governor – i.e., by grammaticalization, e.g. in the proverb *Les petits ruisseaux font les grandes rivières*, 'tall oaks from little acorn grow', literally, "little streams form big rivers". Here, it seems that the French linguist wants to give the apart status of dependency only to semantically influenced connections. Unfortunately, this crucial point is never mentioned anymore throughout the book (more than 600 pages); in fact, only generic, underspecified connections are actually used.

In sum, Tesnièreian Structural Syntax shows a triple in order to describe the connection between two linguistic elements: governor, dependent, connection. Moreover, it is admissible that connections can be generic, structurally or semantically influenced. The depicting of this triple through unary trees – called *représentation stemmatique*, let it be 'structural syntactic trees' hereafter – made by the author seems not to be the best way to describe such a structure, under a formal point of view. For this reason, the model proposed here makes use of a special form of binary trees,

called 'adpositional trees', in brief *adtrees*.

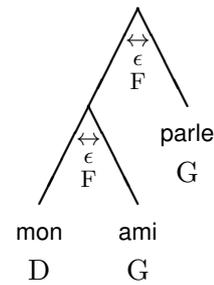


Figure 3: How connection is reinterpreted

Figure 3 shows the reinterpretation of example 2 in terms of adtrees instead of structural syntactic trees, where all structural elements become far more evident. Governors (indicated by G) are put on the right leaf of the binary trees, while dependents (indicated by D) are put on the left ones. The third element of the triple, is put as a "hook" under the root of the tree, (indicated by F, for 'final'). What Tesnière conceived as the connection, can be represented as *adposition*. In fact, in many languages of the world what gives the final character (F) to the connection is a preposition, a postposition or another form of adposition: this fact gives the same dignity to morphology and syntax, unlike Tesnière's tenet (see section 3 below). In the case of example 2, as the connections between *mon ami* and *parle* and *mon* and *ami* are morphologically unmarked, i.e., they are syntactic adpositions, epsilons (ε) are put accordingly (figure 3). The influences behind connections are left underspecified through left-right arrows (↔).

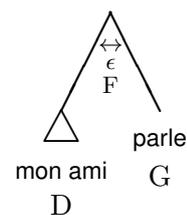


Figure 4: Information hiding explained

The main linguistic difference between the proposed structure, using adpositional trees, which are binary, and Tesnièreian structural syntactic trees, which are unary, is the following: from the point of view of the verbal governor *parle*, the dependent is the whole tree depicting *mon ami* (Figure 4). On the contrary, in Structural Syntax, apparently only *ami* is dependent of *parle* (figure 1 right). Furthermore, the small triangle (Δ) indi-

cates that a binary tree was “packed” in order to increase human legibility: however, no information was lost, but only *hidden* – i.e., it is always possible to get that information explicit. No such possibility is present in Tesnière (1959). However, a single example of something similar to information hiding is provided, when dealing with grammaticalization (1, A, ch. 29, par. 21).

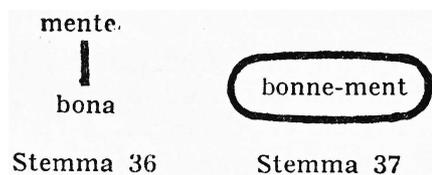


Figure 5: Grammaticalization example

Figure 5 shows how the Latin syntactic expression *bona mente* (‘with a good mind’) became in French *bonne-ment* (‘quite simply’) for grammaticalization.²

This example lead to the goal of providing a coherent treatment in terms of binary trees of all features explained in Structural Syntax – and consequently, in terms of the underlying formal model, as explained below. In fact, one of the great problems in Tesnière (1959) is that the examples (*stemma*) are illustrated in different ways throughout the work, where some information got lost during the way, and other introduced – for instance, connection influence, as presented above, got completely lost.

The explicitation of the triple ‘governor, dependent, connection’ let the structure to be illustrated with the recursively use of adtrees – partially hidden when needed – retrieving Tesnière’s structural information whenever possible.

3 Word classes and syntactic functions

Tesnière (1959) quite early introduces a set of symbols which “express the deep nature [of structure] without keeping the accidental contingencies” (A, ch. 33, par. 1). For Tesnière, morphology is the “shallow facet” while syntax is the “essential facet” of structure, i.e., Humboltian *Innere Sprachform* – in modern terms, deep structure (1, A, ch. 12, note 1). This emphasis on syntax is a severe limit, perhaps a cultural heritage of the times when the French linguist lived, where

²Grammaticalization is, roughly speaking, the ‘paradox of change’ (Coseriu) for which yesterday’s morphology becomes today’s syntax, and vice versa, paraphrasing Givón.

so much importance was given to morphology, almost neglecting syntax. However, now times changed, and it is possible to express both syntactic and morphological information in the same binary tree structure (although, from a formal and computational point of view, syntax and morphology still may be kept separate for practical reasons). In other words, the model proposed here aims to extends Structural Syntax in order to comprehend morphologic phenomena. As it considers morphosyntax as a whole, and it has adpositions (interpreted as morphosyntactic connectors) as the central concept, it was called ‘Constructive Adpositional Grammars’ – the term ‘constructive’ will be explained in section 6 below, devoted to the formal model.

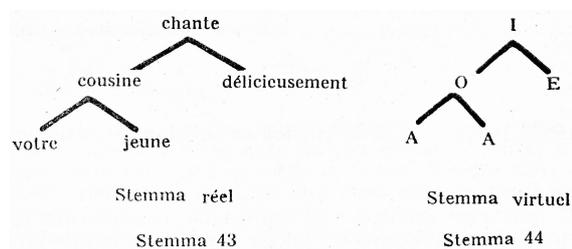


Figure 6: Tesnierian classes and functions at work

Figure 6 shows the instantiated example 43 and its abstract syntactic tree (example 44, i.e. *stemma virtuel*).

(43) *vous* *jeune* *cousine* *chante* *délicieusement*
‘your young cousin sing lovely’

Tesnière individuates four ‘word classes’ (*classes de mots*) as generally and cross-linguistically valid. Their markers, which indicate their respective ‘syntactic functions’ (*fonctions syntactiques*), are the following:

- I = verbants (presence of predication),
- O = statives (expressions of reference),
- E = circumstantials (modifiers of verbants),
- A = adjunctives (modifiers of statives).

There is general agreement among linguists that the presence of expression of reference (i.e., “things”) and the presence of predication (i.e., “events”) are conceptual archetypes, i.e., always-valid universals of language (Langacker, 1987; Tomasello, 2003, for example).

Within the Standard Average European sprachbund, verbants (I) include verbs and interjections, while statives (O) include common and proper nouns, personal pronouns. Normally verbants and

statives are the governors of their respective structural trees, while their modifiers play the role of dependents. Let adjunctives (A) be the modifiers of statives, including adjectives, determiners, possessive pronouns. Finally, let circumstantials (E) be the modifiers of verbants, e.g., in English, adverbs and adverbials. Figure 6 (right) shows that both modifiers (A and E) are dependents, respectively of the stative (O) *cousine* and the verbant (E) *chante*, and in fact they are put below in the Tesnièrean abstract syntactic tree.

Tesnière (1959) explains that the choice of the vowels is a borrowing from the planned language Esperanto, used as a “mnemonic tool” (1, A, ch. 33, par. 1). While the original Tesnièrean vowels are retained here for adherence with the original, in order to help the reader in the comparison of the two models, their original names, like “substantives” or “verbs”, were not adopted in Constructive Adpositional Grammars, being too closely related to the grammar tradition belonging to the Standard Average European sprachbund (Haspelmath, 2001). However, it is worth noticing that Tesnière (1959) gives examples in many different languages through the book, e.g. French, German, Latin, Russian, Greek, but also Coptic, Chinese, Samoan, Turc, Tatar, Votjak, in order to show how Structural Syntax is valid across sprachbunds.

In a completely independent way, Whorf (1945) addressed the problem of grammar categorization out of Standard Average European, with results similar to Tesnière’s. Since Whorf’s names are valid across typologically distant sprachbunds, they were adopted here, with some adaptation.

The main difference between the two authors is the concept of selective and collocational lexemes introduced by Whorf (1945). He noticed that in every language some lexemes he calls selective have their proper grammar category carved inside, as in the English adjunctive (A) *honest*. No collocation can turn the grammar category of the selective adjunctive, but only morphology, e.g., *honest-y*, in order to obtain a stative (O), or *honest-ly*, in order to obtain a circumstantial (E).

By contrast, collocational lexemes are defined only if put into the syntagmatic axis: in isolation, we can have cues about their most probable function, but we cannot be certain. For instance, the English lexeme *walk* is probably a verbant (I), as in the phrase *I walk in the park*. Nonetheless, it can also be a stative (O), as in *Let’s have a walk*

or even an adjunctive (A), as in *walking distance*.

For this reasons, within Constructive Adpositional Grammars instead of ‘word classes’ it is preferred to say ‘grammar characters’, as the characters are can be applied or not to morphemes following the adtree where they are collocated, while selective lexemes are retained as a special case.

4 Adpositional trees and valency

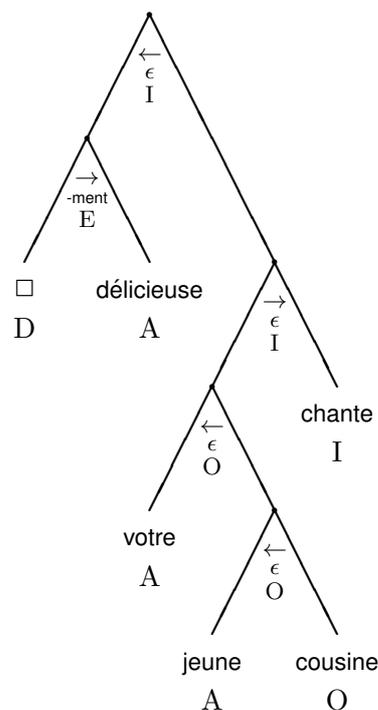


Figure 7: Reinterpretation of examples 43-44

Adtrees retain all the features of Tesnière’s model in a single, unique representation, as shown in the adtree of examples 43-44 (figure 7).

First, both the concrete and abstract syntactic trees (i.e., *stemma réel* and *virtuel*) are represented together. Moreover, the verbant *chante* and the stative *cousine* are the governors of their respective adtrees, as expected from example 44. The reader is invited to note that the final grammar character of the stative group *votre jeune cousine* is indicated by the syntactic adpositions (ϵ); analogously, the stative-verbant connection is syntactic as well. On the contrary, the adverb *délicieusement* is obtained by the application of the suffix *-ment* which act as an adposition, imposing the circumstantial grammar character E to the adjective *délicieuse*. The dependent branch in this case is left underspecified (D), while structurally it is blocked (\square), i.e., it cannot be furtherly expanded by the application of other morphemes. This adtree shows

how syntactic and morphological connections are treated in the same way. Finally, unlike structural syntactic trees, the trajectories of information prominence are rendered explicitly in the adtree of examples 43-44 (figure 7).

4.1 Trajectories of information prominence

The Tesnièrean dichotomy ‘structural vs. semantic’ influence was the source of one of the core features of Constructive Adpositional Grammars. Typological research on ergativity has shown that a good grammar theory “would have to recognise that there are three basic syntactic-semantic primitives (A, S and O) rather than just two ‘subject’ and ‘object’ – however these are defined” (Dixon, 1994, 236). The arrows proposed by Tesnière (1959) are a cue for the solution of this problem. Within a stative-verbant connection, if the stative actively “does” the action, then the stative will be the most prominent element of the pair: in the terms proposed by Langacker (1987), the stative (O) will be the trajector (tm) while the verbant (I) will be the landmark (lm). Therefore, the trajectory of information prominence will be left-right (\rightarrow). In other words, the stative, being the dependent, is prominent (tr), and hence the connection will be a *dependency* (‘semantic influence’, according to Tesnière). Conversely, if the action marked by the verbant (I) “happens” to the stative (O), then the verbant will be trajector (tr) and the stative landmark (lm): the trajectory will be right-left (\leftarrow) accordingly. As the verbant is the governor, the connection will be a *government* (‘structural influence’, according to Tesnière). Therefore, the word ‘dependency’ assumes a very technical and precise meaning within the adpositional paradigm. It is important to note that what stated for the stative-verbant connection is valid for every grammar character connection, as exemplified in figure 7.

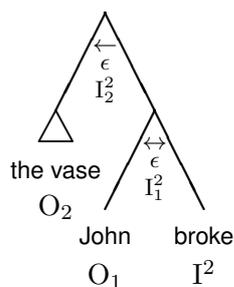


Figure 8: Adtree of *John broke the vase*

The adtree of *John broke the vase* is a good

example of how trajectory of information prominence is treated within the adpositional paradigm. Let assume that our world model is standard, i.e., vases are inanimated objects, without will or beliefs, and John is a man.³ While John can have broken the vase by accident (government, \leftarrow) or willingly (dependency, \rightarrow), the vase for sure happened to be broken, from its point of view, and hence its connection is a government (\leftarrow).

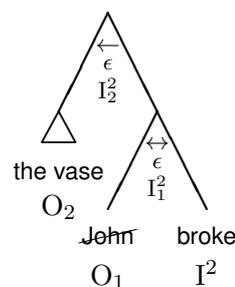


Figure 9: Adtree of *the vase broke*

Trajectory of information prominence explains why some “subjects” are statives in dependency – \overrightarrow{O}_1 , ‘A’ for Dixon (1994) – while others are in government, i.e. \overleftarrow{O}_1 , ‘S’ for Dixon (1994). In fact, the adtree of *the vase broke* (figure 8) can be considered a reduced or *transferred* adtree of *John broke the vase* (figure 9), where the subject (either in government or dependency, i.e., generically O_1) got lost. Before to deal with the concept of transference, which is derived from the Tesnièrean *translation* – explained in Part 3 of Tesnière (1959) – it is necessary to explain how valency is treated within the model proposed here.

4.2 The treatment of valency

The introduction of valency by Tesnière (1959) is one of the most successful part of his Structural Syntax, as it was adopted in most dependency-based frameworks in its fundamental traits:

one could indeed compare the verb to a kind of **crossed atom**, which can attract a number more or less high of actants, in proportion to the number more or less high of hooks needed to maintain the respective dependencies (2, D, ch. 97, par. 3).

The concepts of valency and actants, i.e., how many statives are needed to saturate the valency

³Constructive Adpositional Grammars are agnostic in respect of world models.

value, are taken as such in Constructive Adpositional Grammars.

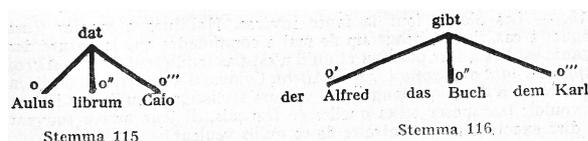


Figure 10: Examples of trivalent verbs

Figure 10 shows how Tesnière sometimes indicates the numbers of the actants saturating the valency value, in case of a trivalent verb. The examples are in Latin and in German, where an English equivalent can be *Alfred gave the book to Charles*.

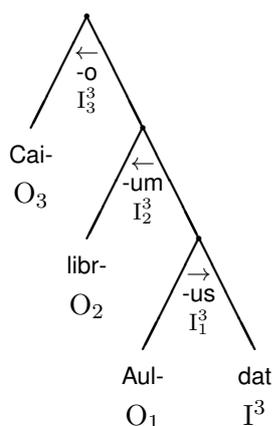


Figure 11: Adtree of example 115

Figure 11 shows the adtree of example 115 in Latin. In Constructive Adpositional Grammars, the verbant is the governor of the phrasal adtree (with ‘phrase’ meaning a tree governed by a uniquely identified verbant). If the verbant is a trivalent verb, as in example 115, three actants (i.e., O_1, O_2, O_3) are provided to saturate the valency value, along with their respective adtrees. The superscript number of the verbant indicates the absolute valency value – e.g., I^2 for a bivalent verb. The subscript number of the verbant indicates the degree of saturation in that point of the adtree, while the subscript of the actant indicates the actant number, following Tesnière’s usage (figure 10). Example 115 shows that Latin substantive finals act as adpositions of the stative-verbant connection, with an indication of information prominence: *Aulus* (‘Alfred’) performs the giving (*dat*) and hence it is in dependency (\rightarrow), while the giving happens both to *Caio* (‘Carl’), being the beneficiary, and *librum* (‘the book’), i.e., the actual object which was given, are both in government

(\leftarrow).

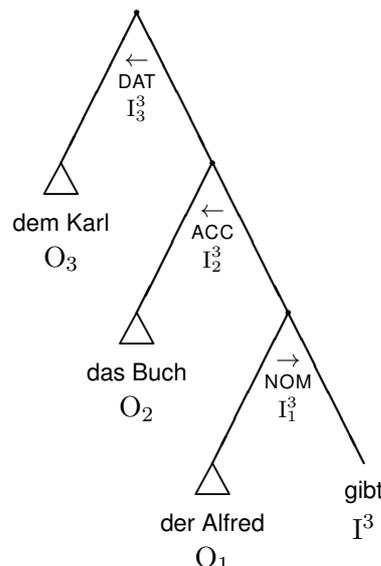


Figure 12: Adtree of example 116

Sometimes adpositions are marked through sememes, i.e., structural well-defined traits within a given language, even if the morph – the explicit morphemic signifier – is absent. For instance, example 116 shows that in German the case markers, like DATIVE, are not morphologically marked in the stative-verbant connection, but still well present in every German speaker’s competence. In these cases, sememes can be written explicitly instead of epsilons, for clarity, if there is no possible ambiguity.

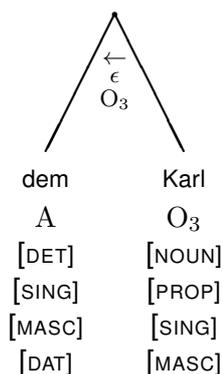


Figure 13: Detail of example 116

The detail of the adtree hidden under the third actant *dem Karl* (figure 13) shows that the sememe DATIVE is an additional trait under the adjunctive grammar character. Moreover, the adtree clarifies that there is a number agreement requirement, indicated by the sememe SINGULAR, between the stative *Karl* and the adjunctive *dem*, in order to

get everything work under a grammatical point of view. No such level of detail is present in Structural Syntax, most probably because Tesnière was not interested in such a direction. However, it is important that such level of detail is possible within the model here proposed if needed, e.g. for language learning purposes.

5 Transference

Every language – regardless of its sprachbund – has a class apart within its morphemes devoted to convey the most part of semantics, called lexemes. In fact, while the concept of ‘word’ is central only in the grammar traditions of the Standard Average European, the distinction of lexemes within a language’s morpheme set is valid in general. For example, in Japanese non-lexemes are written in *kana* (syllabic script), while lexemes are written in *kanji* (Chinese logographic characters).

Lexemes are morphemes devoted to represent the relation between the language and the non-linguistic world, with a particular attention to reference. In structural syntactic trees, they are put above, being the governors, and similarly in the adpositional paradigm they are put in the right-most leaves of their respective adtrees.

Tesnière (1959) noted that the most part of the non-lexical morphemes have the function of “turning” the grammar character of the lexemes they are applied to (3, A, ch. 161, par. 6).

(290) un exemple frapp-**ant** (I > A)

‘a strik-**ing** example’

(292) liber Petr-**i** (O > A)

‘Peter’s book’

The French suffix *-ant* (in 290) is applied to verbant lexemes in order to transfer their syntactic function to adjunctive, while the Latin suffix *-i* (in 292) is applied to stative lexemes in order to transfer their syntactic function to adjunctive as well. Of course, there are a lot of differences between the two adjunctives: in Constructive Adpositional Grammars, they would be expressed by different sememes and trajectories of information prominence. This kind of differences are not well formalized in Structural Syntax; however, the fundamental intuition that the morpheme set of a language can be divided in lexemes and non-lexemes on a functional syntactic basis is a remarkable part of Tesnière’s heritage in the adpositional paradigm, since its definition in

Gobbo (2009). This kind of morphemes (prepositions, pospositions, derivational suffixes and so on) were called by Tesnière (1959) *translatifs* (in the model proposed here, morphological, explicit adpositions) and the phenomenon as a whole was called *translation*, while in English “an equivalent may be *transference*, as the word *translation* has already the meaning of the French ‘traduction’” (3, A, ch. 154, par. 2).

In the development of the formal model on which Constructive Adpositional Grammars are based, the role of transference is growing of importance. Tesnière (1959) devoted a lot of pages to transfer chains, from ‘simple transference’ (*translation simple*, e.g. I > O) to ‘double transference’ (*translation double*, e.g. O > A > O) until, at a limit, sevenfold transference (*translation septuple*). Complex transfer chains, i.e., double or more, can be explained in terms of recursive, nested adtrees, but this solution has two limits. First, from a linguistic point of view, there is no relation between an abstract adtree and the others belonging to the same language – ‘abstract adtree’ meaning what Tesnière called a *stemma virtuel*, i.e., an adtree without morphemic instantiation. Second, from a computational point of view, the constructive adpositional grammar of a given language, which should contain at least two sections – the first for morphemes, their grammar characters, basic transfers and the appropriate sememes, the second for the set of admissible abstract adtrees – will grow inconveniently. In fact, one of the goals of the adpositional paradigm is to give a convenient description of natural language grammars, both linguistically and computationally.

5.1 Abstract adtrees as constructions

The Tesnièrean concept of transference shows that most part of the dictionary is in reality the result of transfer chains: for this reason, a constructive dictionary can be built upon the lexemes and a set of transfer chain *patterns* in order to perform grammar character changes. In a cognitive linguistic perspective, these patterns of usage of form-meaning correspondences, that carry meaning beyond the meaning of the single composing morphemes, are called *constructions* (Croft, 2001; Goldberg, 2006). As a side note, the community of cognitive linguistics recognized Structural Syntax as a complementary, although dated, approach (Langacker, 1995).

After the study of Tesnièrean transference, it seemed more reasonable to see abstract adtrees as constructions instead of describing grammar only in terms of adtrees, so that the relations between constructions are formally represented in terms of adtree transformations, i.e., Tesnièrean transference rendered in formal terms. For example, the active-passive diathesis transference (2, D, ch. 101–102) can be expressed in terms of adtree transformations. Basically, the most primitive construction is the active diathesis, with all valency saturated by the possible actants, then a chain of adtree transformations permits to obtain the desired construction.

- (A) (Carl)_O (slept in)_I (the beds)_O.
 (B) (the beds)_O (were slept in)_{I>I} (by Carl)_{O>O}.
 (C) (Carl's)_{O>A} (sleeping)_{I>O}.

Examples A-B-C were annotated with the main grammar characters of the respective adtrees in order to help the reader in the knowledge of the use of transference within the model proposed here. In particular, example A shows an instantiation of the active diathesis construction of the English verbant *to sleep in*, while example B shows the correspondent passive construction. It is worth noticing that two transfers were performed in order to obtain the appropriate form of the verb ($I > I$) and of the SLEEPER actant ($O > O$). Moreover, example C is an example of nominalization: the SLEEPER actant was transferred into a saxon genitive construction ($O > A$) while the *ing*-construction transferred the verbant into a stative ($I > O$).

It is possible to write down classes of lexemes following the admissible patterns of adtree transformations. For example, it can be easily tested that the verbants *to sleep in* and *to melt in* belong to different classes of English verbants:

- (D) the ice cube melted in the oven.
 (E) *the oven was melted in by the ice cube.
 (F) the melting of the ice cube.

Example D is structurally identical to example A; nevertheless, the passive construction obtained by the adtree transformation is ungrammatical (example E), while a different adjunctive construction, head by the adposition *of*, is to be preferred to saxon's genitive (example F). A full treatment of adtree transformation would need at least another paper devoted to it, so it is left as a further work.

English	wh-ere	th-ere	wh-en	th-en
French	où	là	qu-and	alors
Latin	u-bi	i-bi	qu-ando	t-um
German	w-er	d-a	w-ann	d-ann

Table 1: Tesnièrean analysis of correlatives

5.2 Second-order transference

Tesnière (1959) introduces the second-order transference (*translation du second degré*) in order to explain “what the traditional grammar had already implicitly regarded apartly with the name of ‘subordination’.” (3, D, ch. 239, par. 2). For example, the sentence *Alfred espère qu'il réussira* (‘Alfred hopes that he will achieve’) is a second-order transference from the verbant phrase *Alfred réussira* (‘Alfred will achieve’) to the stativized phrase *qu'il réussira* (‘that he will achieve’; 3, D, ch. 241, par. 15). This kind of second-order transference is indicated with the symbol: \gg ; e.g., a verbant-stative second-order transfer will be indicated as such: $I \gg O$.

Tesnière (1959) noticed that the *translatifs* – in the model proposed here, adpositions – devoted to second-order transference show a high degree of regularity in many different languages (3, D, ch. 240, par. 6, adapted in Table 1).

In Constructive Adpositional Grammars there is no need of a second-order level because of the expressive power of the mathematics underlying the formal model (see next section 6). What is retained from the Tesnièrean analysis is the observation that correlatives are *double* morphemes, made by a fixed part (e.g., *wh-* in English), that is appended to the governor phrase, and a flexible part (e.g., the English *-ere* for PLACE and *-en* for TIME) that is put in the adtree of the adtree of the subordinate phrase.

- (H) I know where she goes.

Figure 14 shows the adtree of example H. The adtree of *where she goes* is intact in its inner construction: the relevant fact is that the correlative adposition *wh-ere* transfers the phrase from verbant to the second actant stative ($I > O_2$), from the point of view of the construction of *I know [where she goes]*. As the reader can see, adtrees can represent correlatives without any need of a second-order level of analysis.

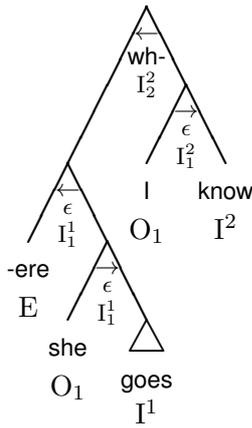


Figure 14: Adtree of example H

6 Sketches of the formal model

Tesnière (1959) asserts that “the use of symbols [grammar characters O, I, E, A, authors’ note] in grammar is equivalent to the use of calculus in algebra” (1, A, ch. 33, par. 10). This statement implies that Structural Syntax can be formalised, at least theoretically.

In the fields of mathematical and computational linguistics there are many natural language grammar formalisms currently under investigation. In particular, the constructive adpositional formalism can be put into the realm of the so-called “categorical grammars”—i.e., representations of natural language grammars in terms of categories (Morril, 2007). At the present stage, the formal model is intended as a guiding reference for the development of linguistic concepts (GobboBenini, 2011). In fact, ‘constructive’ is intended linguistically as pertaining constructions (as already defined) and mathematically as pertaining constructive mathematics, i.e., any formal, mathematical construct used here have a constructive existence. In other words, it is possible to find an algorithm, non necessarily efficient, to construct any entity of the model.

In particular, adtrees and constructions together form a category, called **AdTree**, in the mathematical sense (MacLane, 1998; Borceux, 1994). A mathematical category is an algebraic structure composed by two classes, the *objects* and the *arrows*; arrows lie between two objects, the *source* or *domain*, and the *target* or *codomain*. Also, a category states that there are distinct arrows, the *identities*, one for every object A and such that the source and the target are A . Moreover, a category is equipped with a partial operation allowing to

compose two arrows whenever one has the domain which is the target of the other one. Composition is required to be associative and identities act as one expects with respect to composition.

Intuitively, there is an arrow f from A to B whenever we can construct the B tree starting from the A tree applying the construction f . We do allow complex constructions obtained by sequentially composing simpler ones; if f and g are constructions such that $f(A) = B$ and $g(B) = C$, that is, if f maps A into B , and g constructs C from B , then $g \circ f$ is the construction which maps A into C by doing g after f .

It is possible to observe that, calling M the free monoid over the alphabet of morphemes of some natural language, i.e., the set of all possible (finite) sequences of morphemes obtained by juxtaposition, the functions mapping the trees in **AdTree** into the sequences of M comprehend the textual renderings of adpositional trees. If the attention is restricted to *contravariant functors*, i.e., the functions preserving the identical transformation and the reverse composition of adpositional trees, what is obtained is a class of functions which is called *presheaves over M* . Requiring that a presheaf maps the morphemes in the adtree into themselves in the monoid, what is obtained is exactly the lexicalizations of adtrees. In other words, there is a subclass of presheaves which directly corresponds to the texts the adtrees represent and which encodes the transformations that constitute the grammar. It is this space of presheaves which is generally understood as the subject of linguistics. Moreover, considering *endofunctors* on **AdTree**, i.e., functions mapping each adtree into another adtree, and each construction into another one such that they preserve identities and composition, it easily follows that each linguistic transformation, e.g., the mapping of active to passive diathesis, is an endofunctor. In turn, an endofunctor can be represented as an arrow between presheaves, thus showing that the mathematical model of the presheaves space is rich enough to represent and to reason about the foundational elements of Constructive Adpositional Grammars.

As a side effect of this intended model of interpretation, it follows that whatever construction over adtrees which is built by combinatorially composing the fundamental constructions, is an arrow. Lifting the structure of the **AdTree** category into the spaces of presheaves, which is a cat-

egory, it is possible to reason in a larger and richer environment, where the full power of mathematical methods can be applied: in fact, the presheaves space is a *Grothendieck topos* (MacLane, 1992; Johnstone, 2002), one of the richest mathematical structures available.

7 Conclusion

The impressive work by Tesnière (1959) is a constant source of inspiration for the definition of Constructive Adpositional Grammars. It is quite astonishing that nobody until now – as far as the authors know – has proposed a dependency-based model that makes use of the grammar characters proposed by the French linguist, i.e. O, I, E, A, which are the ground on which Structural Syntax is actually built. Such heritage could be the topic of another paper.

Directly formalising Structural Syntax, which was the first hypothesis considered, is simply not possible, essentially for two reasons. First, pragmatically Tesnière (1959) is a posthumous publication, and hence there are formal and linguistic incongruences which cannot be overcome; in particular, the unary tree (*représentation stemmatique*) used by the author is ever-changing within the text, and not optimal to represent the triple ‘governor, dependent, connection’, for the reasons exposed in this paper. Second, Tesnière, working in the 1930-50 years, was a son of his time: he could take advantage of the fruits of the great tradition of linguistic structuralism that spread out in francophone Europe in the first half of the past century, but on the other hand he could not have the proper formal and mathematical instruments to be applied to his linguistic results – as category theory was introduced by Samuel Eilenberg and Saunders Mac Lane in the 1940s, and in those times it was not mature enough for linguistic applications.

Nonetheless, Constructive Adpositional Grammars, standing on the shoulders of Tesnière, can be considered a derivative work of Structural Syntax in many aspects, all of which were presented in this paper.

References

- Francis Borceux. 1994. *Handbook of Categorical Algebra I*. Cambridge University Press: Cambridge.
- William Croft. 2001. *Radical Constructions Grammar*. Oxford University Press: Oxford.
- Ralph Debusmann. 2001. *A declarative grammar formalism for dependency grammar*, PhD thesis. Universität des Saarlandes.
- Robert M. W. Dixon. 1994. *Ergativity*. Cambridge University Press, Cambridge.
- Federico Gobbo and Marco Benini. 2011. *Constructive Adpositional Grammars: Foundations of Constructive Linguistics*. Newcastle upon Tyne: Cambridge Scholars Publishing.
- Federico Gobbo. 2009. *Adpositional Grammars: A Multilingual Grammar Formalism for NLP*, PhD thesis. University of Insubria.
- Adele E. Goldberg. 2006. *Constructions at work*. Oxford University Press: Oxford.
- Martin Haspelmath. 2001. The European linguistic area: Standard Average European, in: Martin Haspelmath, Wulf Oesterreicher and Wolfgang Raible. *Handbuch der Sprach- und Kommunikationswissenschaft*. Mouton de Gruyter, Berlin & New York. 1492–1510.
- Peter Johnstone. 2002. *Sketches of an Elephant: A Topos Theory Compendium*, Volume 2. Cambridge University Press: Cambridge.
- Ronald W. Langacker. 1995. Structural syntax: the view from cognitive grammar, in: Germain Dondelinger, Françoise Madray-Lesigne and Jeannine Richard-Zappella. *Lucien Tesnière aujourd’hui*. Peeters Publishers, Leuven. 13–37.
- Ronald W. Langacker. 1987. *Foundations of Cognitive Grammar: Theoretical Prerequisites*. Stanford University Press, Stanford.
- Igor Mel’čuk. 1988. *Dependency Syntax: Theory and Practice*. The SUNY Press, Albany, N.Y.
- Saunders Mac Lane. 1998. *Categories for the Working Mathematician*. Springer Verlag, Berlin.
- Saunders Mac Lane and Ieke Moerdijk. 1992. *Sheaves in Geometry and Logic*. Springer Verlag, Berlin.
- Glyn Morrill. 2007. A Chronicle of Type Logical Grammar: 1935–1994. *Research on Language & Computation*, 5(3):359–386.
- Joakim Nivre. 2005. *Dependency grammar and dependency parsing*, Technical report. Växjö University.
- Lucien Tesnière. 1959. *Éléments de syntaxe structurale*. Klincksieck, Paris.
- Michel Tomasello. 2003. *Constructing a language: A Usage-Based Theory of Language Acquisition*. Harvard University Press, Harvard.
- Benjamin L. Whorf. 1945. Grammatical Categories. *Language*, 21(1):1–11.



Implementing Categorical Grammar in Semantic Analysis: from a Frame Semantics' View

Ke Wang

Dalian University of Technology,
116024, P.R. China
wang.coco.ke@gmail.com

Rongpei Wang

Dalian University of Technology,
116024, P.R. China
rpwang1@yahoo.com

Abstract

In this paper, we propose a new idea that semantic frames are taken as the functions, and semantic categories (usually labeled with semantic roles) are taken as arguments. Thus, a semantic frame can apply to semantic categories if semantic categories are consistent with the semantic frame. Beta-reduction is used to represent the idea of the application of semantic frame to semantic categories. Semantic consistency is tested through β -unification. It is concluded semantic consistency problems are decidable if verbs are typable in the system of frames.

1 Introduction

Grammar is the set of rules that governs the composition of phrases or words to be meaningful and interpretable in a given natural language, i.e. a grammar should explain why a sentence is acceptable while others are not. In this case, syntax and semantics are not opposite to each other. However, many of semantic issues cannot be explained in CGs¹. For example, the following examples share the same construction, coordination-reduction, which has been finely explained in Combinatory Categorical Grammar (Mark Steedman, 1987). Both (1) and (2) are grammatical in CGs; however, (2) is completely unacceptable in semantics.

- (1) Mary planted and Harry cooked the beans.
- (2) *Harry cooked and Mary planted the beans.

¹ CGs is the general name of variants of Categorical Grammar. A better introduction of variants of CG can be found in Mary M. Wood's work (1995).

Mostly, CGs can distinguish sentences from non-sentences, but it is inefficient when to explain this kind of semantic issues. In this paper, we tried to diagnose the above semantic problem through combining the ideas of frame semantics and logic inference methods. We propose a new idea that semantic frames are considered as functions, and semantic categories (usually labeled with semantic roles) are taken as arguments. Thus, a semantic frame can apply to semantic categories if these semantic categories are consistent with the semantic frame.

We used semantic roles to replace the syntactic categories of CGs so as to enrich it with a stronger capability in semantic analysis. Then, the combinator C (Haskell Curry, 1942) is introduced, with which the disturbed positions of arguments in a complex sentence could be reordered. After that, beta-reduction was used to represent the idea of the application of semantic frame to semantic categories. In seeking of a method to resolve this problem, it is proposed that the unification of typed feature structures that represent the semantic categories and semantic frames is right the one we are pursuing. However, it is still quite difficult to decide whether an instance of unification could have a solution in lambda calculus. Finally, β -unification (A.J. Kfoury, 1999) is discussed, which says that an instance of unification problems in lambda calculus can have a solution if and only if lambda term M (from which the instance is transformed), is strongly β -normalizable. M is strongly β -normalizable if and only if M is typable in the lean fragment of the system of intersection types. Thus, it was hypothesized that the semantic frame system is the lean fragment of the system of intersection types

(15) Strangely enough, Paul refuses to talk.
reorder: X Z Y
rewrite: X' (X'=X) Y' (Y'=Y\Z)
convert: X'/Y' Y'
compose: X'/Y' Y' → X'

where, *rewrite*, *convert*, and *compose* are the operators that have been introduced in section 2. (16) and (17) are similar examples, if we consider 'must' in (17) as a disjunct, for example 'I guess', rather than a modal verb:

(16) He, *frankly speaking*, is not good enough.

(17) He *must* be angry.

3.2 Coordination-reduction

The examples (1) and (2) mentioned at the beginning of this paper share the same construction, coordination-reduction. The omitted constituents did not disappear; actually they exist in deep semantic layers, as shown in the followings:

(1)'Mary planted [...] ^{Theme} and Harry cooked [the beans] ^{Patient}.

(2)* Harry cooked [...] ^{Patient} and Mary planted [the beans] ^{Theme}.

To give a further explanation on why (2) is not acceptable in semantics, we should use world knowledge. We may share the common sense that an action of change cannot be withdrawn by human power. In logics, this kind of world knowledge can be represented as the application of semantic frames (functions) to semantic roles (variables), as shown in (18) and (19):

(18) Mary planted and Harry cooked the beans.
Y X

Harry cooked the beans Mary planted [...].
reorder: X Y
rewrite: X' Y'
convert: X'/Y' Y'/[...] ^{Theme}
compose: X'/Y' Y'/[...] ^{Theme} → X'/[...] ^{Theme}

(X' = X\Cook/Patient; Y' = Y\Agent/Theme)

(19) λPatient. X'[Patient:=Theme]

Note that the composition of semantic frames can be realized either by application rules or

composition rules. In (18), X'/[...] ^{Theme} means semantic frame X' applies to [...] ^{Theme}. As shown in (19), if [...] ^{Theme} is consistent with X', then, it can replace the variable Patient in X'. Technically, this replacement could be implemented through the unification of semantic frame and semantic categories. It is expected to find a way, such as the one in figure 1, through which the semantic consistency of [...] ^{Theme} can be tested.

$$\left[\begin{array}{l} \text{variable} \\ \text{lexical: } \dots^{\text{Theme}} \\ \text{semantic: [Patient:animate]} \end{array} \right]$$

Unifies with

$$\left[\begin{array}{l} \text{frame} < \text{cook} > \\ \text{lexical: } \text{cook} \\ \text{semantic: [Agent: intelligent} \\ \text{Patient: animate/inanimate]} \end{array} \right]$$

Figure 1. The unification of [...] ^{Theme} and frame X' in example (19).

In figure 1, the meanings of [...] ^{Theme} and the frame X' are represented by a particular notation called typed feature structures. When the variable unifies with the frame, the semantic consistency is tested through the compatibility of the two structures. As it is shown, [...] ^{Theme} is compatible with requirements of frame <cook>. Analogously, in figure 2, [...] ^{Patient} is not compatible with the requirements of frame <plant>. This explains why (1) is acceptable in semantics, while (2) is not.

$$\left[\begin{array}{l} \text{variable} \\ \text{lexical: } \dots^{\text{Patient}} \\ \text{semantic: [Theme:inanimate]} \end{array} \right]$$

Unifies with

$$\left[\begin{array}{l} \text{frame} < \text{plant} > \\ \text{lexical: } \text{plant} \\ \text{semantic: [Agent: intelligent} \\ \text{Theme: animate]} \end{array} \right]$$

Figure 2. The unification of [...] ^{Patient} and frame <plant> in example (2)'.

The decidability of the unification is discussed in section 4. For more information

about unification of typed feature structures, please refer to Carpenter (1992) and Gerald (2000)

4. Discussion

In Kfoury's work (1996), he proved that an instance Δ of unification problem U (β -unification) has a solution iff M is β -strongly normalizable, (where M is a lambda term, from which Δ can be transformed); and that M is β -strongly normalizable iff M is typable in the lean fragment of the system of intersection types.

Apart from the precise definitions and proofs, intuitively, if semantic frame were the lean fragment of the system of intersection types, and if verbs that bear the meanings of semantic frames could be typable in such system, then the semantic consistency in (19) is decidable.

Linguistically, being typable in the system of semantic frame means verbs, such as 'cook' and 'plant' in (1) and (2), are of completely different types. Therefore, verb types can explain why the semantic changes of 'the beans' caused by 'cook' is unacceptable in the semantic frame represented by verb 'plant'.

5. Conclusion

In this paper, a new idea is proposed, that semantic frames are seen as the functions, and semantic categories (usually labeled with semantic roles) are taken as the arguments of functions. Thus, a semantic frame can apply to arguments, the variables. Many complex constructions, such as insertion and co-ordination reduction can be well explained with this set of approaches.

The combinator C is used for reordering the disturbed positions of arguments in a complex sentence. Beta-reduction is used to represent the idea of the application of semantic frame to semantic categories. The idea of the proof of decidability of unification problems in β -reduction is borrowed from Kfoury's work (1999). It is concluded semantic consistency problems are decidable if verbs are typable in the system of semantic frames.

The ultimate goal of computational linguistics is to let machines understand

human's language. It is hoped that the idea proposed in this paper could help to implement a real NLU system, suppose, if there were some resources that finely describe types of verbs and lexical meanings of each word of a language. Actually, there already have been some (such as, WordNet, VerbNet, and FrameNet).

Acknowledgments

We are grateful to the reviewers for their expending time on reading our drafts and for their valuable suggestions.

References

- Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. The University of Chicago Press, Chicago and London.
- Charles J. Fillmore. 1968. The Case for Case. In Bach and Harms (ed.): *Universals in Linguistic Theory*. New York: Holt, Rinehart, and Winston.
- Charles J. Fillmore. 1982. Frame semantics. In The Linguistic Society of Korea, eds. *Linguistics in the Morning Calm*. Seoul: Hanshin.
- Joachim Lambek. 1958. The mathematics of sentence structure. *The American Mathematical Monthly*, Vol. 65, No. 3.
- Kazimierz Ajdukiewicz. 1935. Die Syntaktische Konnexitat. *Studia Philosophica* 1: 1-27; translated as 'Syntactic Connexion' in S. McCall (ed.), *Polish Logic*, Oxford, 1976.
- Mary M. Wood. 1993. *Categorial Grammars*. Routledge, USA and Canada.
- Yehoshua Bar-Hillel. 1953. A Quasi-arithmetical Notation for Syntactic Description. *Language* 29.
- Mark Steedman. 1987. Combinatory Grammars and Parasitic Gaps. *Natural Language and Linguistic Theory*. Vol. 5, 403-439.
- Mark Steedman. 2000. *The Syntactic Process*. The MIT Press. Cambridge, Massachusetts, London, England.
- Julia Hockenmaier and Mark Steedman. 2005. *CCGbank: User's Manual*. Technical Report MS-CIS-05-09. Department of Computer and Information Science. University of Pennsylvania, Philadelphia.
- A.J. Kfoury. 1999. *Beta-Reduction as Unification*. This can be downloaded from: <http://types.bu.edu/index.php?p=reports>.

- Haskell Curry. 1942. The Combinatory Foundations of Mathematical Logic. *The Journal of Symbolic Logic*. Volume. 7, Number 2.
- Bob Carpenter. 1992. *The Logic of Typed Feature Structures*. Cambridge University Press. Printed in the USA.
- Gerald Penn. 2000. *The Algebraic Structure of Attributed Type Signatures*. Doctoral Thesis, Carnegie Mellon University. Pittsburgh PA.



A proposal for a multilevel linguistic representation of Spanish personal names

Orsolya Vincze

Universidade da Coruña
ovincze@udc.es

Margarita Alonso Ramos

Universidade da Coruña
lخالonso@udc.es

Abstract

This paper proposes a multilevel representation of personal names, with the aim of offering an economical treatment for these expressions, which makes a clear distinction between ontological information, described in a *name database*, and linguistic levels of representation. Adopting the linguistic model and formalisms provided within the Meaning \leftrightarrow Text framework (Mel'čuk 1988), it is argued that, contrary to other proper names (e.g. organizations, toponyms, etc.), which should be treated similarly to idioms, complex personal names such as *José Luis Rodríguez Zapatero* should not be represented as single units at any linguistic level nor in the lexicon. Variant forms referring to a concrete person (e.g. *José Luis Rodríguez Zapatero*, *Rodríguez Zapatero*, *Zapatero*, *Z.P.*) are accounted for by a set of rules connecting the *name database* and the semantic level.

1 Introduction

Proper names have traditionally occupied a rather marginal position in linguistic description. As a consequence, the systematic and formalized description of their syntactic and morphological behavior remains largely unattended. More recently, in the field of natural language processing (NLP), the treatment of proper names has been put into focus, as a consequence of the growing interest in tasks involving the recognition of *named entities*, a set of expressions characterized by having a unique reference (e.g. Vitas et al. 2007).

A problem going further than the mere identification of segments of texts as proper names, which is generally solved using simple heuristics (cf. Krstev et al. 2005: 116), is that of the treatment of the various ways a particular entity can be referred to (Nadeau and Sekine 2007). For instance, in journalistic texts, the current Spanish prime minister can be designated by either one of the following

strings: *José Luis Rodríguez Zapatero*, *Zapatero*, or *Z.P.* It has been found that NLP applications dealing with this latter, more complex question can profit from information on the linguistic properties of names (e.g. Charniak 2001; Gaizauskas et al. 2005; Vitas et al. 2007). One way of tackling the problem, proposed by the authors of *Prolexbase* (Krstev et al. 2005), a multilingual ontology of proper names, is that of explicitly listing variant forms of names in a lexical database.

The aim of the present paper is to propose a representation of Spanish personal names, wherein variant forms can be treated in a more economical way. For this, we have adopted the linguistic model proposed within the Meaning \leftrightarrow Text framework (MTT, Mel'čuk 1988). To our knowledge, no attempt has been made to formally integrate personal names in any such comprehensive linguistic model, therefore, this proposal should be considered as rather tentative.

The most important feature of our description is that we suggest a clear distinction between ontological information, contained in the *person database*, where a person is conceived as a single entity, and linguistic representation, where personal name strings are analyzed as complex structures constituted by name elements. Consequently, as we will show, variant forms can be accounted for by a set of rules establishing correspondences between the person database and the linguistic levels of representation.

Note that, in what follows, we will use the more generic term *proper name* to refer to those expressions which constitute the names of geographical locations, organizations, institutions, persons, etc., while the more specific term *personal name* will be used for the expressions that name particular individuals.

2 Related work

2.1 Encyclopedic vs. linguistic description of proper names

The definition of the notion of proper names has been formulated in various ways in linguistics, mainly proposing an opposition of this class to that of common nouns on the basis of their different semantic and/or referential properties. We do not intend to discuss this issue in detail; however, it is relevant to note that the existence of such an obvious difference lies at the root of the lexicographical tradition of excluding proper names from dictionaries, and transferring them to encyclopedias (Marconi 1990). This practice has been challenged by some authors, (e.g. Lázaro Carreter 1973; Mufwene 1988; Higgins 1997) arguing that, whatever the content of these expressions, their linguistic properties, such as gender, number, pronunciation, variant spellings, etc. should be described systematically.

Concentrating on the case of personal names, we find that, like other proper names, these are generally excluded from dictionaries; that is, we will not find dictionary entries with names of specific persons, given that this information is considered to belong to the encyclopedia. More importantly, name elements such as given names like *José*, their non-standard diminutive form *Pepe*, and surnames like *Rodríguez* are also excluded from the lexicographical tradition. Nevertheless, we do find some cases of derived relational adjectives that make reference to specific persons, e.g. *Freudian* with reference to Sigmund Freud. This latter aspect has been pointed out by, for instance, Lázaro Carreter (1973) and Higgins (1998), who claim that it violates the self-sufficiency principle in lexicography, namely, definitions of these adjectives point to entities – specific persons – on whom often no information is provided in the dictionary.

Within the field of NLP, it is claimed that named entity recognition systems are able to function quite efficiently on the basis of simple heuristics (Krstev et al. 2005: 116). This may be the reason why researchers working in this field are generally not concerned with describing specific linguistic properties of these expressions in a systematic way. Although lexical resources such as ontologies or knowledge-based systems are created for named entity tasks (e.g. Morarescu and Harabagiu 2004; Rahman and Evens 2000), these are generally

applied for the semantic classification of named entities. In consequence, they are merely designed to incorporate encyclopedic information in a formal, computerized lexicon, leaving linguistic properties of proper names unattended.

On the contrary, the description of the linguistic properties, together with the formal and orthographic variants of proper names, seems to be rather important in the case of more complex tasks such as identifying *aliases*, that is, the various ways an entity can be spelled out in a text (cf. Nandreau and Sekine 2007: 16), or for computer-assisted translation and multilingual alignment (Maurel 2008). For instance, as illustrated in (1), a person, such as *Sigmund Freud* can be referred to by variant name forms, as well as by a derived relational adjective. Moreover, some languages may prefer one formulation to the other, and a language may completely lack a particular derivative form (Vitas et al. 2007: 119).

- (1) *Sigmund Freud's/S. Freud's/Freud's/the Freudian theory of human personality*

Prolexbase (Krstev et al. 2005; Maurel 2008, etc.), a multilingual relational database of proper names has been created with the aim of proposing a solution for the problem posed by variant forms of proper names. Consequently, besides conceptual or encyclopedic information, it also contains description of formal variants. Each entity is represented by a single, language independent node, which is linked to a lemma in each specific language, representing the base form of the given proper noun, which is in turn specified for all of its variant forms. For example, as shown in Figure 1, the same ID is associated with the French and the English lemmas, *États-Unis* and *United States* respectively, and the latter is specified for its variant realizations *United States of America*, *USA*, *as well as the adjective American*.

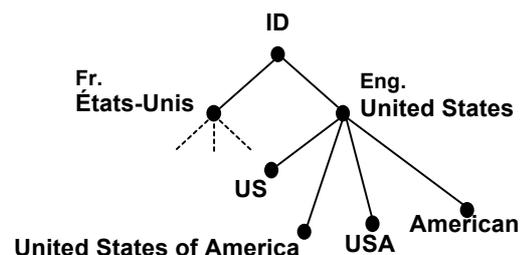


Figure 1: Representation of different forms of proper names in Prolexbase (adapted from Maurel 2008: 335)

2.2 Representation of the structure of personal names in syntactically annotated corpora

The syntactic representation of personal names and of proper names in general, to our knowledge, has not received sufficient attention. In descriptive grammars, authors tend to limit their analysis of the structure of these expressions to proposing a classification based on their lexical elements: for instance, many proper names are composed only of word forms that can be classified as proper names themselves (e.g. *Socrates*, *Switzerland*), while others are more similar in their structure to regular noun phrases (e.g. *United Kingdom*, *University of Cambridge*), given that they contain adjectives and common nouns (e.g. Quirk et al. 1985: 288-294; Allerton 1987: 67-69). At the same time, after a brief look at syntactically annotated corpora, we arrive at the conclusion that within the field of NLP, there is no consensus on whether to analyse the syntactic structure of names. Namely, treebanks in general differ in treating multilexemic proper names as single nodes, e.g. Spanish *AnCora Corpus* (Martí et al. 2007) and Portuguese *Floresta Sintá(c)tica* (Afonso et al 2002), or as proper subtrees, e.g. *Prague Dependency Treebank* (PDT, Hajičová et al. 1999; Böhmová et al. 2005).

As for the more specific case of the representation of the syntactic structure of personal names, the number of existing proposals is rather limited. For instance, Anderson (2003: 374) suggests that complex forms of personal names are headless compounds, whose elements, the given name and the family name are juxtaposed, given that, one may perceive the given name modifying the family name, or vice versa, depending on the context. Within the dependency framework, the PDT provides an analysis where all other elements are syntactic dependents of the rightmost element of the name string, generally the last name, and are represented as adnominal modifiers (Böhmová et al. 2005: 836). From the perspective of the MTT, Bolshakov (2002) suggests representing Spanish personal names as dependency chains where the family name depends on the given name, and proposes a specific type of surface syntactic relation, *nomination appositive*, to describe the dependencies between their components.

3 The linguistic status of personal names

Given that we aim at proposing a linguistic description for personal names, we have to raise the question of what kind of linguistic units these expressions are and how they should be represented on each level of representation proposed by the linguistic model of the MTT (e.g. Mel'čuk 1988).

An important feature of our framework is the clear split between linguistic and non-linguistic level. Following this idea we propose to describe ontological information on each entity in the *name database*, separate from the linguistic representation, attending solely to linguistic properties of name elements. In this way, we obtain a more economical treatment of variant forms of personal names via a set of rules operating between these two main levels of representation, and avoid explicitly listing variant forms of names in a lexical entry (cf. Tran and Maurel 2006:119-120).

In syntactic analysis, as in the case of some *treebanks*, proper names are often treated as idioms, that is, indecomposable chains. However, the MTT proposes a more multifaceted treatment for idioms. Within this framework the semantic unity of full idioms is reflected by representing them as a single lexical unit in the dictionary and, consequently, as a single node at the deep syntactic level, while they are assigned a proper subtree representation at the surface syntactic level, indicating their internal structure. The reason for this is not only the lack of semantic compositionality characterizing these expressions, but also the structural irregularities they present in comparison with regular, compositional expressions.

We would like to underline that, from our point of view, an important distinction should be made between the representation of names of organizations, toponyms, etc., on the one hand, and personal names, on the other hand. We claim that expressions belonging to the first group (e.g. *Organization of United Nations*) should be treated similarly to full idioms, attending to semantic non-compositionality. In contrast, we suggest complex personal names to be represented as complex structures at all linguistic levels: as various lexical units in the dictionary, as a compound lexical node at the deep syntactic level, that is, a lexeme constructed from various actual lexemes, similarly to the case of com-

pound lexemes (Mel'čuk forthcoming), and as a tree at the surface syntactic level.

This proposal is based, on the one hand, on the assumption that the structure of personal names can be considered as regular, that is, it can be sufficiently accounted for by a specialized mini-grammar. On the other hand, we claim that, contrary to full idioms which cannot be analysed in terms of the meanings of their components, in the case of names, the meaning of each element, that is, the meaning of each given name and each family name, can be represented as an independent *denomination predicate*, e.g. *José = person X is called José*. We have adopted this concept from Gary-Prieur (1994), according to whom the *meaning* of a proper name is distinct from its *content* defined as a set of properties attributed to the referent.

We assume that the possibility of referring to a person by variant name forms suggests that name elements retain their meaning and can have the same referential content whether used as a full string or independently (as in 2a). Thus, as we show in sentence (2b), meanings of name elements seem to be autonomous within a name string, which is further demonstrated by the fact that they are accessible for coordination (see 2c). Finally, we consider that utterances like (2d) and (2e) can be considered potential cognitive, or, more precisely, referential paraphrases (cf. Milićević 141-145).

- (2a) *That was the first time I met María Lamas, although I'd known María's sister for a long time.*
- (2b) *The author Manuel Rivas is called the same as your father (who is called Manuel González).*
- (2c) *Ana and María Lamas/Ana Lamas and María Lamas are visiting us this week.*
- (2d) *María Lamas*
- (2e) *the girl whose name is María and whose surname is Lamas*

4. Linguistic representation of Spanish personal names

As we have said, our proposal distinguishes two main levels of description: the person database and the linguistic representation. As for the linguistic description, in accordance with the MTT framework, we foresee a *dictionary*, where name elements, that is, both given names and family names are listed and speci-

fied for their linguistic properties. Furthermore, we deal with the following three levels of linguistic representation: *semantic representation* (SemR), the *deep syntactic* (DSyntR) and the *surface syntactic representations* (SSyntR). Each two consecutive levels are connected by a set of rules that serve to establish correspondences between them. Among these, we will limit ourselves to those operating between the person database and the semantic level.

For the purpose of the present paper, we will limit ourselves to the analysis of the most common forms of personal names in European Spanish, which, in their full form, consist of a number of given names, followed by two family names, e.g. *José Luis Rodríguez Zapatero*. Note that full forms of Spanish names usually contain two family names, the first of these being the first family name of the father and the second the first family name of the mother.

4.1 The person database

The *person database* contains a list of all individuals relevant in a given application. Naturally, it would be impossible to compile a list of all human beings in the world, so, for practical purposes, the content of this component will always have to correspond to specific research objectives. For each individual, several name attributes are specified, such as a) first family name, b) second family name, c) first given name, d) second given name, e) nickname, and f) derived lexical units. Sometimes an individual can be referred to by different full names depending on the context, in these cases, attributes have to be specified under such fields as *birth name*, *public name*, *maiden name*, etc. (Cf. Bartkus et al. 2007). See Figure 2 for an example of the representation corresponding to *José Luis Rodríguez Zapatero*.

ID=134567 First family name=Rodríguez Second family name=Zapatero First given name=José Second given name=Luis Nickname=Z.P. Derivate = zapateriano

Figure 2: Representation in the person DB

At this level, the attribute *nickname* refers to a form that is used to name a particular individual. This form does not correspond to

standard nicknames or diminutives (see section 3.2), which can make reference to any individual carrying a particular name. Likewise, as we have already explained, derivative forms included in the ontological representation also make reference to a specific person e.g. *freudiano* → *Sigmund Freud*; *cervantino* → *Miguel de Cervantes*, *isabelino* → *Queen Elizabeth I and Queen Elizabeth II of Spain*, *Queen Elizabeth I of England*.

The name database should also include relevant extralinguistic or encyclopedic information on each individual. This information may have certain importance in the identification of a name as referring to a specific person on the basis of context, for instance, appositives like *presidente*, *general*, *secretario*, *director*, etc. (cf. Arévalo et al. 2002). As we have seen, encyclopedias and certain resources developed for NLP applications generally concentrate on this kind of information. However, since our purpose is to look at personal names from a strictly linguistic point of view, we won't discuss this aspect in more detail.

4.2 The dictionary

The *dictionary* should include a complete list of name elements, that is, given names and family names together with their variant and derivative forms. This implies that our formal dictionary does not include the full form of the name, and hence, encyclopedic information on a specific person, e.g. *José Luis Rodríguez Zapatero*, instead, it specifies the following information (see Figure 3).

José:	proper name, given name, masculine Nickname: <i>Pepe</i>
Luis:	proper name, given name, masculine
Rodríguez:	proper name, family name, weak
Pepe:	[= Nickname (<i>José</i>)] proper name, nickname, masculine
Zapateriano:	adjective, related to <i>ID134567</i>
Zapatero:	proper name, family name
Z.P.:	nickname for <i>ID134567</i>

Figure 3: Representation in the dictionary

Note that in the case of each name element, we include information on syntactic class (*proper name*) and specify the subcategory (*given name* or *family name*). We consider the latter distinction necessary, given that, as

we will show later, we perceive a difference in the syntactic combinability of these classes¹.

Lexical entries of given names indicate irregularly derived standard nicknames. For instance, in the case of *José*, we include the form *Pepe* but not regularly derived *Josito*². These variant forms also receive their own dictionary entry, while derived forms or non-standard nicknames, like *Zapateriano* or *Z.P.*, constitute an individual entry, without any link to the base form. Note that, as we have already discussed, these forms make reference to a specific person, instead of e.g. all persons called *Zapatero*, that is why, their reference is specified via an ID, assigned to the person in the person database.

Another property of both given- and family names that we find important from the point of view of lexical description, is the feature of *weakness*. In the case of female compound given names such as *María Teresa*, *María Dolores*, etc. Spanish speakers will generally opt for using the second element, contrary to other compound cases like *Fernando Manuel* or *Rosa María*, where generally the second given name is omitted. Similarly, in the case of family names, there is a preference towards retaining the second family name when it is perceived as more salient. An example would be the case of the Spanish president *José Luis Rodríguez Zapatero*, commonly referred to as *Zapatero* and not as *Rodríguez*. In both cases, the attribute *weakness* seems to be related to the frequency of use of these name elements, however, further empirical research would be needed to establish clear criteria. For some frequency information on compound given names, see (Albaigès 1995: 82-83).

Finally, we find worth mentioning that there are certain forms of given names for which it may be problematic to decide whether they should be treated as compounds containing two independent name elements or they

¹ Naturally, the choice of one or another combination of these name elements to refer to an individual also reflects pragmatic, sociolinguistic, etc. differences, factors which are beyond the scope of this study.

² Note that the distinction between regularly and not regularly derived standard nicknames may not be as straightforward as it may seem at first sight. Spanish given names generally, but not always, receive the diminutive ending *-ito/a* as in *Miguel* → *Miguelito*, *Rosa* → *Rosita*, but *Carlos* → *Carlitos*, and not **Carlosito*; *Mercedes* → *Merceditas*, and not **Mercedesita*. (We would like to thank one of the anonymous reviewers for pointing this out.)

should be stored as a single lexical unit. For instance, in the forms *María del Carmen*, *María del Pilar*, etc., similarly to cases we have just seen, *María* tends to behave as a weak element, however, the second part *del Pilar* or *del Carmen* is not autonomous, e.g. *María del Carmen Álvarez/Carmen Álvarez/*del Carmen Álvarez*. Furthermore, certain compounds correspond to a single diminutive form, e.g. *María del Carmen*=*Maricarmen/Mari Carmen*, *José Miguel*=*Josemi*, *José María*=*Chema*, *María Jesús*=*Chus*, while others, like *José Luis* or *Miguel Angel*, although they do not have a corresponding single diminutive form, are often perceived as a single word form.

4.3 Semantic representation (SemR)

As we have already suggested, in formulating the SemR, we have adopted the concept of *denomination predicate*, (Gary-Prieur 1994) to represent the *meaning* of names. Consequently, we conceive of each name element as including a predicate, e.g. *José* = *person X is called José* so that the representation of the sequence used to refer to a specific person called *José Luis Rodríguez Zapatero* would be as in Figure 4.

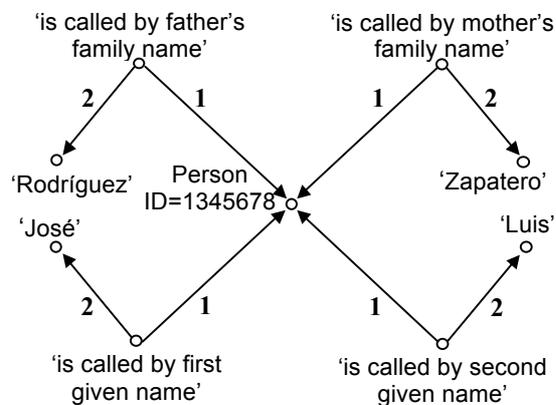


Figure 4: SemR of the name *José Luis Rodríguez Zapatero*

As shown in Figure 4, in the cases where more than one name element of the same category (i.e. given name or family name) is used, the semantic representation is enriched with more specific information. For instance, full forms of Spanish names usually contain two family names, as we have said, the first of these coming from the father and the second from the mother. When only one name element of a category is used, this information would not necessarily be present in SemR. As a consequence, simpler semantemes could be used,

e.g. if the current Spanish president is referred to by the form *Zapatero*, the semanteme ‘family name’ instead of ‘mother’s family name’ would be used in the SemR.

4.4 Deep- and surface syntactic representation (DSyntR and SSyntR)

The syntactic representation of personal names has not been studied in detail within the Meaning⇔Text framework, the only proposal we know about being that of Bolshakov (2002).

We propose representing personal names on the DSynt by a single node, in a similar way as compound lexemes are represented. As pointed out by Mel’čuk (forthcoming), compound lexical units that are fully compositional potential lexical units do not need a lexical entry in the dictionary, given that they are constructed in a regular way through the combination of full lexemes. Their internal structure is considered a syntactic island for DSyntS rules, but it is specified as a tree-like structure whose nodes are labelled with the compounding lexemes, in order to provide information for the linear ordering of components. In a similar way, we propose representing personal names as potential lexical units constructed out of element names, see (3a) and (3b).

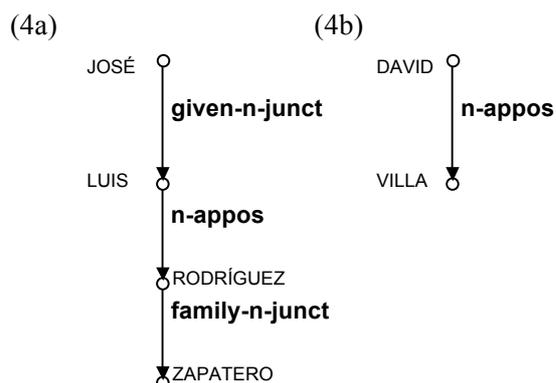
(3a)
○ [JOSÉ→LUIS→RODRÍGUEZ→ZAPATERO]

(3b)
○ [DAVID→VILLA]

However, on the SSynt level, personal names will be represented as proper sub-trees, in the same way as idioms, following Mel’čuk (1988 and 2009). Nevertheless, we have found that the special characteristics of personal names do not lend themselves easily to determining the directionality of syntactic relations on the basis of the criteria proposed by Mel’čuk (2009). As a consequence, we have decided to adopt Boshakov’s (2002) scheme, where, as we have already mentioned, name elements form a dependency chain headed by the first given name. Considering the lack of other criteria, we believe this kind of representation to be convenient, given that it facilitates linearization, contrary to, for instance, PDT type representation (see section 2.2).

For labelling dependencies, we have decided to introduce three different syntactic relation types to represent relations between the name elements that concern us, that is, given names and family names. Our decision was

based on one of the criteria provided by Mel'čuk (2009: 34-35), namely that every relation type has to have its prototypical dependent, which can be substituted for the actual dependent in any configuration, resulting in a correct structure. Consequently, we propose *name appositive* to represent the relation between the last given name or a standard nickname and the first family name, *given name junctive* will stand between any two given names and, finally, *family name junctive* connects the two family names, see (4a) and (4b).



4.5 Mapping between the *person database* and the semantic level

In the MTT framework correspondences between two consecutive levels of linguistic representations are established by a set of rules. Similarly, we propose a series of rules for mapping between the *person database* and the semantic level of our model, with the aim of providing a systematic account for the formal variants of personal names referring to the same individual. These rules reflect all possible combinations of the name elements.

By way of illustration, we will discuss the case of the complex name form consisting of one single given name and one family name³. For the mapping rules applied in this case see Figure 5. G1 and G2 stand for the forms filling the first and second given name attribute respectively, and F1 and F2 are the forms filling the father's and the mother's family name attribute respectively. Note that in the semantic

³ Other possible variant patterns are: 1) Given name+Given name+Family name+Family name (*José Luis Rodríguez Zapatero*); 2) Given name+Given name (*José Luis*); 3) Given name+Family name+Family name (*Federico García Lorca*), 4) Family name (*Aznar*) and 5) Non-standard nickname (*ZP*).

representation, as we have discussed, a proper sub-network will correspond to each selected attribute.

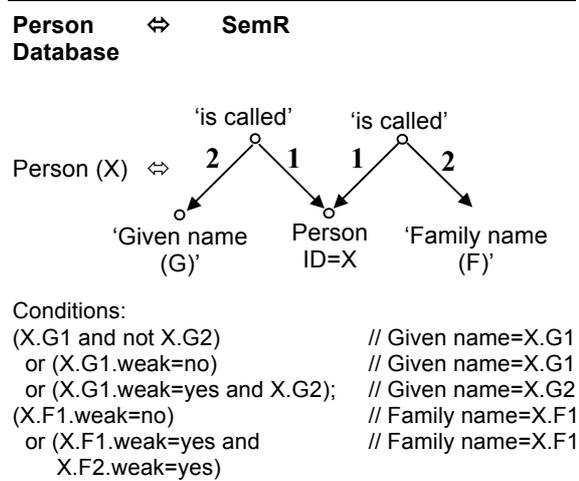


Figure 5: Mapping Rule between the *name database* and the semantic level

We assume that on the basis of these rules and making use of both types of information stored in the *name database* and the *dictionary*, correct forms agreeing with the first name+given name pattern (G F) can be generated. For instance, for a person whose corresponding attributes are G1=*María*, G2=*Teresa*, F1=*Álvarez*, F2=*Fernandez*, we can generate the form *Teresa Álvarez*, given that name elements *María*, *Álvarez* and *Fernández* are specified as [+weak] in the dictionary. Similarly, these rules can serve to associate the form *Teresa Álvarez* with persons with matching attributes in the *name database*.

Note that the name of current Spanish president José Luis Rodríguez Zapatero is generally not to be used with this pattern, since first [+weak] family names followed by a [-weak] family name are rarely, and second family names are never used alone. That is, for any Spanish speaker it would result rather strange to refer to the current prime minister as *José Luis Rodríguez* and they would never refer to him as *José Luis Zapatero*. As we have already mentioned, the compound name *José Luis* shows a particular behaviour, for now, not covered by our rules. A single element, either *José* or *Luis* is used only without family names, on the contrary, when family names are used as well, these given names tend to obligatorily appear in the compound form, which may point towards the fact that this form should be treated as a single word form.

5 Conclusion

This paper has presented a proposal for a multilevel representation of personal names with the aim of accounting for variant combinations of name elements that can be used to refer to a specific person. We have suggested that a clear distinction is necessary between ontological information and linguistic levels of representation. Adopting the linguistic model and formalisms provided by the MTT framework, we have argued that, contrary to other proper names, such as names of organizations, toponyms, etc., which should be treated similarly to full idioms, personal names are to be represented as complex structures on all linguistic levels: as various lexical units in the dictionary, a “quasi-compound” lexical node on the deep- and as a tree on the surface syntactic level. Finally, variant forms of personal names referring to a given individual have been accounted for by a set of rules establishing correspondences between the name database, containing ontological information, and the semantic level.

Acknowledgments

This work has been supported by the Spanish Ministry of Science and Innovation and the FEDER Funds of the European Commission under the contract number FFI2008-06479-C02-01. We would also like to thank Simon Mille, Igor Mel'čuk and Leo Wanner, as well as the anonymous reviewers for their valuable remarks and comments on the previous versions of this text.

References

- Afonso, S., E. Bick, R. Haber and D. Santos. 2002. Floresta sintá(c)tica: a treebank for Portuguese. In González Rodríguez, M. And C. P. Suárez Araujo (eds.) *Proceedings of LREC 2002*, Paris, ELRA, 1698-1703.
- Albaigès, J. 1995. *Enciclopedia de los nombres propios*. Barcelona: Planeta.
- Allerton, D. J. 1987. The linguistic and sociolinguistic status of proper names. *Journal of Pragmatics* XI: 61-92.
- Anderson, J. 2003. On the structure of names. *Folia Linguistica: Acta Societatis Linguisticae Europaeae* XXVII(3-4): 347-398.
- Arévalo, M., X. Carreras, L. Márquez, M. A. Martí, L. Padró and M. J. Simón. 2002. A proposal for wide-coverage Spanish named entity recognition. *Procesamiento de Lenguaje Natural*, 28: 63-80.
- Bartkus, Kim, Paul Kiel and Mark Marsden. Person name: Recommendation, 2007 April 15. HR-XML Consortium. http://ns.hr-xml.org/2_5/HR-XML-2_5/CPO/PersonName.html
- Böhmová, A., A. Cinková and E. Hajičová. 2005. A manual for tectogrammatical layer annotation of the Prague Dependency Treebank. Available: <http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/t-layer/pdf/t-man-en.pdf>
- Bolshakov, I. 2002. Surface syntactic relations in Spanish. In Gelbukh, A. (ed.) *Proceedings of CICLing-2002*, Berlin/Heidelberg: Springer-Verlag, 210-219.
- Charniak, E. 2001. Unsupervised learning of name structure from coreference data. In *NAACL*.
- Gaizauskas, R., T. Wakao, K. Humphreys, H. Cunningham and Y. Wilks. 2005. Description of the LaSIE System as Used for MUC-6. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann.
- Gary-Prieur, M-N. 1994. *Grammaire du nom propre*, Vendôme, Presses Universitaires de France.
- Hajičová, E., Z. Kirschner and P. Sgall. 1999. A manual for analytical layer annotation of the Prague Dependency Treebank. Available: <http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/a-layer/pdf/a-man-en.pdf>
- Higgins, Worth J. 1997. Proper names exclusive of biography and geography: maintaining a lexicographic tradition. *American Speech*, 72(4): 381-394.
- Krstev, C., D. Vitas, D. Maurel, M. Tran. 2005. Multilingual ontology of proper names In *Proceedings of 2nd Language & Technology Conference*, 116-119.
- Lázaro Carreter, F. 1973. Pistas perdidas en el diccionario. *Boletín de la Real Academia española*, 1973, 53(199): 249-259.
- Marconi, D. 1990. Dictionaries and proper names. *History of Philosophy Quarterly*, 7:1, 77-92.
- Martí, M. A., M. Taulé, M. Bertran and L. Márquez. 2007. AnCora: Multilingual and multilevel annotated corpora. Available: clic.ub.edu/corpus/webfm_send/13
- Maurel, D. 2008. Prolexbase: A multilingual relational lexical database of proper names. In Calzolari N. et al. (eds.) *Proceedings of LREC-2008*, Paris: ELRA, 334-338.

- Mel'čuk, I. 1988. *Dependency syntax: Theory and practice*, Albany, State University of New York Press.
- Mel'čuk, I. 2009. Dependency in natural language. en Mel'čuk, I. and A. Polguère (eds.), *Dependency in linguistic description*, Amsterdam/Philadelphia, John Benjamins, 1-110.
- Mel'čuk, I. forthcoming. *Semantics*, Amsterdam/Philadelphia: John Benjamins.
- Morarescu, P. and S. Harabagiu. 2004. NameNet: a self-improving resource for name classification. In *Proceedings of LREC 2004*.
- Milićević, Jasmina. 2007. *La paraphrase: Modélisation de la paraphrase langagière*, Bern, Peter Lang.
- Mufwene, S. S. 1988. Dictionaries and proper names. *International Journal of Lexicography*, 1(3): 268-283.
- Nadeau, D. and S. Sekine. 2007. A survey of *named entity* recognition and classification. *Linguisticae Investigationes*, 30(1): 3-26.
- Quirk, R., S. Greenbaum, G. Leech and J. Svartvik. 1985. *A contemporary grammar of the English language*, London/New York: Longman.
- Rahman M. A. and Evens, M. 2000. Retrieving knowledge for a lexical database from proper noun entries in Collins English Dictionary. In *Proceedings of MAICS-2000*, 63-67.
- Tran, M. and D. Maurel. 2006. Un dictionnaire relationnel multilingüe de noms propres. In *Traitement Automatique des Langues XLVII(3)*: 115-139.
- Vitas, D., C. Krstev and D. Maurel. 2007. A note on the semantic and morphological properties of proper names in the Prolex Project. *Linguisticae Investigationes*, 30(1): 115-133.

On deriving semantic representations from dependencies: A practical approach for evaluating meaning in learner corpora

Michael Hahn

Detmar Meurers

Seminar für Sprachwissenschaft

Universität Tübingen

{mhahn, dm}@sfs.uni-tuebingen.de

Abstract

One of the reasons for the popularity of dependency approaches in recent computational linguistics is their ability to efficiently derive the core functor-argument structure of a sentence as an interface to semantic interpretation. Exploring this feature of dependency structures further, in this paper we show how basic dependency representations can be mapped to semantic representation as used in *Lexical Resource Semantics* (Richter and Sailer 2003), an underspecified semantic formalism originally developed as a semantic formalism for the HPSG framework (Pollard and Sag 1994) and its elaborate syntactic representations.

We describe a two stage process, which in the first stage establishes a syntax-semantics interface representation abstracting away from some differences in surface dependencies. It ensures the local reconstruction of arguments for middle and long-distance dependencies, before building the actual LRS semantics in the second stage. We evaluate the approach on the CREG-109 corpus, a small dependency-annotated corpus with answers to reading comprehension questions written by American learners of German.

1 Introduction

Computational linguistics in recent years has seen a rise in interest in tasks that involve the evaluation and comparison of meaning. A good example is the research strand which has developed around the Recognizing Textual Entailment Challenge (Dagan et al. 2009). De-

pendency representations have received a lot of attention in this context given that they provide access to functor-argument structures without requiring a computationally costly commitment to a more complex syntactic constituent structure.

In our work, the task is the evaluation of answers to reading comprehension questions. We want to determine, whether the answer given by a student expresses the same meaning as that expressed in a target answer given by the teacher. Such a comparison can be carried out at different levels of abstraction, starting with direct comparisons of the surface forms of words, as, for example, used in the BLEU and ROUGE metrics for evaluating machine translation and summarization approaches (Lin and Och 2004; Lin 2004). At the other extreme are comparisons on the basis of deep semantic analysis and logical inference, which, however, in practice do not necessarily outperform the shallow methods (Bos and Markert 2006). For exploring the space in-between these two extremes, looking for representations on the basis of which meaning can be compared and how they can robustly be obtained, in this paper we discuss the derivation of underspecified semantic representations on the basis of a dependency analysis.

We make this process concrete on the basis of *Lexical Resource Semantics* (LRS, Richter and Sailer 2003), an underspecified semantic formalism which provides explicit semantic representation while at the same time exposing all minimal building blocks of the semantics. This is relevant for our overall goal of having access to a rich space of representations for comparing meanings even in situations where no complete semantics can be obtained. Using LRS essentially allows us to (re)use semantic analyses developed in model-theoretic frame-

works. The goal of this research is to further the understanding of the mapping and the information needed to derive semantic representations – with a specific focus on LRS representations as a kind of normal form facilitating meaning comparison, an abstraction away from the significant well-formed and ill-formed variation exhibited by learner language.

2 Creating lexically enriched syntax-semantics interface representations

To derive semantic LRS representations from dependency structures, we use a two-step approach. In the first step, the syntactic structure is transformed into a syntax-semantics interface representation, from which the semantic contribution of each word can be computed in the second step, independent of the syntactic structure.

Adopting a two-step approach instead of a single-step mapping makes the system more robust and flexible. The feature-based interface representation abstracts away from variation in form and grammaticality. The semantic representation as such then can be built on this strongly constrained interface representation and the system constructing the semantics does not need to take into account the large space of possible underlying dependency configurations. The explicit interface representation thus makes the semantic construction process more robust against unexpected parses. It also makes the procedure building the semantic representation more transparent. As we will show, only a small number of rather simple rules is needed. And implementing new semantic analyses is rather straightforward if the relevant information is encoded in the interface representation. Last but not least, the interface representation also allows us to build on previous work on semantics within the HPSG architecture since the system can model the HPSG syntax-semantics interface on the lexical level directly.

2.1 The nature of the representations

We focus on the core local, middle-distance and long-distance relations in a sentence. The goal is to achieve good coverage of language

phenomena in general as well as to deal with well-known argument-structure challenges.

The representation used to capture the properties needed to identify these relations are represented by a set of features which are defined for every word. They mainly provide information about valency, modification and a more fine-grained labelling of syntactic categories. The following features are used:

- PRED: the core semantic relation expressed by the word, generally represented by its lemma
- CAT: the syntactic category
- ARGS: the set of arguments with their role labels. Only semantically non-empty arguments are represented. The elements of ARGS are feature structures of the form

$$\left[\begin{array}{ll} \text{ROLE} & (\text{role label}) \\ \text{ARG} & (\text{the argument itself}) \end{array} \right].$$
- MOD: a modified word, if there is one
- MODTYPE: type of modification (possessor, time, etc.)
- CONJUNCTS: the conjuncts, if this is a first conjunct
- INTERROGATIVE: *true* if this is an interrogative
- IS-PREDICATIVE: *true* if this is a verb or a predicative argument
- TENSE: the tense of a predicative head

An example interface representation for *schreibt* ‘writes’ as in (1) is shown in (2):

- (1) *Peter schreibt einen Brief.*
 Peter writes a letter
 ‘Peter writes a letter.’

- (2)
$$\left[\begin{array}{ll} \text{PRED} & \text{‘schreiben’} \\ \text{ARGS} & \left\langle \left[\begin{array}{ll} \text{ROLE} & \text{subj} \\ \text{ARG} & \left[\begin{array}{ll} \text{PRED} & \text{‘Peter’} \\ \text{CAT} & \text{noun} \\ \text{ISPRED} & \text{false} \end{array} \right] \end{array} \right] , \left[\begin{array}{ll} \text{ROLE} & \text{obja} \\ \text{ARG} & \left[\begin{array}{ll} \text{PRED} & \text{‘Brief’} \\ \text{CAT} & \text{noun} \\ \text{ISPRED} & \text{false} \end{array} \right] \end{array} \right] \right\rangle \\ \text{ISPRED} & \text{true} \\ \text{TENSE} & \text{present} \\ \text{CAT} & \text{verb-finite} \end{array} \right]$$

Extracting this information involves the recursive processing of dependencies and also identifying where dislocated elements are to be interpreted. Some of the features are straightforward to specify locally. CAT, INTERROGATIVE, for example, can simply be assigned based on the dependency labels and part-of-speech tag in the input. However, a well-investigated set of linguistic phenomena, such as strong and weak unbounded dependencies and non-finite constructions, involves dependents which are not realized locally. Our system starts out by building interface representations for local dependency relations only. The structure is then transformed by a procedure which tries to reconstruct a correct representation by moving, copying and adding arguments to the representations of different heads.

2.2 Argument structure challenges

German, as the language we are focusing on here, includes several phenomena which cause arguments to be realized separate from the head they belong to semantically. These include fronting, extraposition, raising, control, passive, and the so-called coherent constructions of the verbal complex (Bech 1955). Since relative pronouns and relative clauses are marked in the dependency parse, identifying relative pronouns with their antecedent can be achieved by recursively searching the dependency graph for a dependency whose dependent is labelled as relative clause and which dominates a given relative pronoun. Other extraction phenomena and the interaction of raising, control, passive and the German verbal complex are more complex to handle since they can interact to form sequences of several verbs, with sequences of three or four verbs being relatively common in written text. While an in-depth discussion of these phenomena clearly is beyond the scope of this paper (cf. Meurers 2000 for an overview), let us illustrate the issue with two examples. Sentence (3) shows a basic example including a future perfect construction and a modal verb.

- (3) *dass ihn Peter [wird [haben [treffen können]]]*
 that him Peter will have meet be able to
 ‘that Peter will have been able to meet him.’

Here, *Peter* is interpreted as the subject of *treffen* ‘meet’, yet it must also be identified as

the subject of the equi predicate *können* ‘be able to’ which is raised further to become the syntactic subject of the perfect tense auxiliary *haben* and to finally be realized as the subject of the future auxiliary *wird*, which shows subject-verb agreement with it. Similarly, *ihn* ‘him’ is interpreted as the object of *treffen* ‘meet’, but given that the other predicates all construct in a coherent verbal cluster, it is ultimately realized as a syntactic argument of the matrix verb *wird* ‘will’ together with the raised subject *Peter*.

That there is indeed a complex interaction of the different types of lexically triggered argument sharing phenomena going on that needs to be captured can readily be illustrated with the so-called long-distance passivization (Höhle 1978, pp. 175ff) shown in (4).

- (4) *wenn der Wagen [[zu reparieren] versucht] wird]*
 when the_N car to repair tried is
 ‘when it is attempted to repair the car’

Here, the passive auxiliary *wird* ‘is’ selects the verbal complement *versuchen* ‘try to’, which however is not a verb selecting an NP object that could be promoted to become the subject. Instead, *versuchen* selects the verbal argument *reparieren* ‘to repair’. Since this is a coherent verbal complex, the argument of *reparieren* also becomes a syntactic dependent of *versuchen* and as such can then be lifted up by the passive auxiliary *wird* to become the subject of the sentence, with nominative case and showing subject-verb agreement with the finite verb.

Building an adequate interface representation clearly requires lexical information about the verbs selecting nonfinite complements. This includes knowledge about whether a verb is a raising or equi predicate and what its orientation is, i.e., which argument slot the raised or controlled subject fills. Furthermore, we need to know which arguments of its own such a verb selects (as, e.g., the dative NP object required by the subject control equi verb *versprechen*).

A basic reconstruction algorithm The procedure for reconstructing functional structures is based on the general observation that all argument sharing constructions involve a predicate which specifies something about the

dependents of its verbal complement. A reconstruction algorithm thus only has to increase the depth of embedding of arguments, but never has to decrease them. Therefore, reconstruction starts from the least embedded verb. Some arguments are moved or copied to the ARGS list of the nonfinite argument, and the same procedure is applied recursively to the embedded predicate, until a predicate without a nonfinite or predicative complement is reached. We furthermore assume that the decision to move an argument can be made locally and depends only on the two verbs under consideration.

In each recursive step, the embedded predicate is identified by its function label PRED, OBJ, or AUX. If the dependency parse is correct and the sentence grammatical, at most one such argument will be present. If no or more than one are found, the algorithm stops. Else, the following operations are carried out:

1. If the matrix verb is *not a passive marker*, the argument with the role label matching the verb's orientation is selected and copied to the ARGS list of the embedded verb, where it has role label *subject*. If the matrix verb is a raising predicate, the copied dependent is deleted from its ARGS list.
2. If the matrix verb is a *tense-marking auxiliary*, the TENSE value of the embedded verb is updated.
3. All arguments which do not match a slot in the verb's argument frame are moved to the ARGS list of the embedded verb. If the surplus arguments cannot be unambiguously determined, no argument is selected.
4. If the matrix verb is the passive auxiliary *werden* or the dative passive marker *bekommen* and the embedded verb is a passive participle, the subject becomes an object of the embedded verb. If a *von* ('by') PP is available, which might encode the agent, its relation is changed to *von_or_subj*. Otherwise, an unspecified subject is added.

As an example, we apply this procedure to the long-distance passivization example we

saw in (4) in the way illustrated in (5). The example shows the ARGS lists before reconstruction (a) and after the two recursive steps (b-c).

- (5) a.
$$\begin{array}{l} \boxed{1} \left[\begin{array}{l} \text{PRED } \textit{werden} \\ \text{ARGS } \langle \textit{AUX } \boxed{2}, \textit{SUBJ } \textit{Wagen} \rangle \end{array} \right] \\ \boxed{2} \left[\begin{array}{l} \text{PRED } \textit{versuchen} \\ \text{ARGS } \langle \textit{OBJI } \boxed{3} \rangle \end{array} \right] \\ \boxed{3} \left[\begin{array}{l} \text{PRED } \textit{reparieren} \\ \text{ARGS } \langle \rangle \end{array} \right] \end{array}$$
- b.
$$\begin{array}{l} \boxed{1} \left[\begin{array}{l} \text{PRED } \textit{werden} \\ \text{ARGS } \langle \textit{AUX } \boxed{2} \rangle \end{array} \right] \\ \boxed{2} \left[\begin{array}{l} \text{PRED } \textit{versuchen} \\ \text{ARGS } \langle \textit{OBJI } \boxed{3}, \textit{OBJ } \textit{Wagen}, \textit{SUBJ } \textit{PRO} \rangle \end{array} \right] \\ \boxed{3} \left[\begin{array}{l} \text{PRED } \textit{reparieren} \\ \text{ARGS } \langle \rangle \end{array} \right] \end{array}$$
- c.
$$\begin{array}{l} \boxed{1} \left[\begin{array}{l} \text{PRED } \textit{werden} \\ \text{ARGS } \langle \textit{AUX } \boxed{2} \rangle \end{array} \right] \\ \boxed{2} \left[\begin{array}{l} \text{PRED } \textit{versuchen} \\ \text{ARGS } \langle \textit{OBJI } \boxed{3}, \textit{SUBJ } \boxed{4} \textit{PRO} \rangle \end{array} \right] \\ \boxed{3} \left[\begin{array}{l} \text{PRED } \textit{reparieren} \\ \text{ARGS } \langle \textit{OBJ } \textit{Wagen}, \textit{SUBJ } \boxed{4} \textit{PRO} \rangle \end{array} \right] \end{array}$$

In the first step, resulting in (5b), the passive marker *wird* is dealt with, for which *Wagen* is removed and turned into the subject of the passivized verb *versucht*. It has no overt agent, therefore a *pro* subject is added. In the second step, resulting in (5c), the subject control equi verb *versuchen* is considered and its subject is copied to the ARGS list of *reparieren*. The accusative object *Wagen* does not match an argument slot in the lexical entry of *versuchen* and is moved to the embedded verb. The verb *reparieren* does not embed a predicate so that the algorithm terminates.

To some extent, this procedure can also deal with fronting in V2 clauses and with relative clauses. However, the lexical information only allows handling dislocated arguments – the correct attachment of adjuncts cannot be determined.

2.3 Relation to other formalisms

Our interface representations are related to LFG f-structures (Kaplan and Bresnan 1995). Most of our features directly translate into common f-structures features. However, our interface representations differ from standard assumptions about f-structures in that they

are closer to the underlying argument structure, i.e., the LFG a-structure. In the interface representations of passive verbs, the agent has the role SUBJ and the patient roles like OBJA. Non-thematic subjects and complements are not represented. This treatment allows a straightforward analysis of some aspects of German syntax such as long-distance passivization, as we will show below. Furthermore, semantic composition is simpler than in LFG, since the arguments represented in the interface representation of some word are always exactly those having a semantic role.

Our two-step approach is also similar to some aspects of the architecture of Meaning Text Theory (Mel’cuk 1988). Our interface representations can be compared to Deep Syntactic Structure, as it also acts as the interface between the surface syntactic dependency structure and a deep semantic representation. While we chose a feature-structure based representation for interface representations, our features ARGS, MOD, MODTYPE and CONJUNCTS can be seen as direct encodings of labelled dependency arcs. However, our interface representations differ from Deep Syntactic Structure in Meaning Text Theory in that they are invariant under phenomena such as passivization, which are already encoded in Deep Syntactic Structure.

The representations are also reminiscent of the linguistic encodings used in HPSG (Pollard and Sag 1994), in particular the treatment of adjuncts as selecting their heads by the MOD feature, which is useful for lexicalized semantic composition. The ARGS list is related to the ARG-ST list often assumed in HPSG, which can be seen as representing the underlying argument structure (Manning and Sag 1998). Furthermore, it appears that all the information contained in our representations is inherent in HPSG analyses and could easily be automatically extracted.

3 The semantic formalism: LRS

Lexical Resource Semantics (LRS, Richter and Sailer 2003) is an underspecified semantic formalism which embeds model-theoretic semantic languages like Ty2 into typed feature structures as used in HPSG. It is formalized in the *Relational Speciate Reentrancy Language*

(RSRL, Richter 2000). While classical formal semantics uses fully explicit logical formulae, the idea of underspecified formalisms such as LRS is to derive semantic representations which are not completely specified and subsume a set of possible resolved expressions, thus abstracting away from scope ambiguities.

While other underspecified formalisms used in HPSG such as MRS (Copestake et al. 2005) encode only an underspecified representation, whose relation to resolved representations is external to the representation language, an LRS representation includes both a resolved representation and a representation of its subexpressions, on which scope constraints can be expressed by the relation \triangleleft ‘is a subexpression of’.

An *lrs* object has three features: INCONT (INTERNAL CONTENT) encodes the core semantic contribution of the head, EXCONT (EXTERNAL CONTENT) the semantic representation of the head’s maximal projection, and PARTS is a list containing the subterms contributed by the words belonging to the constituent. An example is given in (6), a semantic representation for *schreibt* in (2).

$$(6) \text{ a. } \left[\begin{array}{l} \text{INCONT} \quad \boxed{1} \text{ schreiben}'(e) \\ \text{EXCONT} \quad \boxed{2} \\ \text{PARTS} \quad \left\langle \begin{array}{l} \exists e[\boxed{3} \wedge \boxed{4}], \\ \boxed{7} \text{ present}(\wedge \boxed{5}), \\ \boxed{6}(\boxed{1} \text{ schreiben}'(e) \wedge \\ \text{subj}(e, \text{peter}) \wedge \text{obj}(e, y)), \\ \dots \end{array} \right\rangle \end{array} \right]$$

b. $\boxed{6} \triangleright \boxed{2} \wedge \boxed{6} \triangleright \boxed{3} \wedge \boxed{6} \triangleright \boxed{5}$

The INCONT value *schreiben'*(*e*) is the core semantic contribution. The value of EXCONT is not specified, because it also contains the semantics of arguments and modifiers of the verb. The PARTS list contains three ‘maximal’ terms: $\exists e[\boxed{3} \wedge \boxed{4}]$ is the quantifier for the event variable, *present*($\wedge \boxed{5}$) is the semantic representation of tense marking and $\boxed{6}(\boxed{1} \text{ schreiben}'(e) \wedge \text{subj}(e, x) \wedge \text{obj}(e, y))$ represents the verb with its argument structure. Furthermore, PARTS contains every one of their subexpressions with the exception of those which are contributed by another word, but they are omitted in the figure for reasons of readability. The three subexpression constraints in (6b) ensure that the core semantic contribution and the specification of the ar-

guments is part of the representation of the maximal projection, that the event variable is bound by a quantifier, and that the tense predicate outscopes the core semantic contribution.

A possible resolved value for EXCONT of *schreibt* in example (1) is shown in (7).

$$(7) \boxed{2}(\exists y[\text{brief}(y) \wedge \boxed{7} \text{present}(\wedge \exists e[\boxed{5}\boxed{6}(\boxed{3}\boxed{1} \text{schreiben}'(e) \wedge \text{subj}(e, \text{peter}) \wedge \text{obj}(e, y))])])$$

All elements of PARTS are subterms of the complete representation and the subexpression constraints are satisfied.

Unlike some other implementations of deep semantic frameworks, LRS does not employ the lambda calculus as its combinatorial mechanism. Instead, a grammar with an LRS semantics contains three sets of constraints linking syntax and semantics. The INCONT PRINCIPLE ensures that the core semantic contribution (INCONT) is part of the representation of the maximal projection and lexically contributed by the word. The EXCONT PRINCIPLE essentially states that all semantic expressions have to be introduced lexically via the PARTS list. The SEMANTICS PRINCIPLE is grammar-dependent and we show only one exemplary clause:

- INCONT PRINCIPLE:

INCONT is a subterm of EXCONT and a member of PARTS.

- EXCONT PRINCIPLE:

In a maximal projection, EXCONT is a member of PARTS.

In an utterance, α is a member of PARTS iff it is a subexpression of EXCONT.

- SEMANTICS PRINCIPLE:

- If the nonhead is a quantifier, then the INCONT value of the head is a component of the restrictor.
- ...

Adapting LRS for Interface Representations LRS was originally developed for constituency-based grammars such as HPSG, and the combinatorial constraints make reference to phrasal notions such as *maximal projection*. Nevertheless, the formalism can easily be used for our syntax-semantics interface

representations or standard dependency representations. Unlike other underspecified formalisms used in HPSG such as MRS (Copestake et al. 2005), LRS is strictly lexicalized in the sense that all subexpressions of the complete semantic representation have to be introduced at the word level, and INCONT and EXCONT are the same in all projections of a head. Therefore, combinatorial constraints in the SEMANTICS PRINCIPLE which make reference to *non-maximal* projections can straightforwardly be reformulated in terms of dependencies or the features ARGS and MOD. Representations on the level of nonmaximal projections are not necessary for the combinatorial mechanisms of LRS to work.

The EXCONT PRINCIPLE refers to the elements PARTS list of maximal projections, but this can be replaced by referring to the union of the semantic contributions of the direct and indirect dependents. Technically, this can be implemented in the feature-structure-based LRS formalism by a second list DOMAIN-PARTS which is defined recursively as the concatenation of PARTS and the DOMAIN-PARTS lists of all dependents of the word. Thus, all combinatorial constraints of LRS can be translated into lexicalized, dependency-like formalisms such as our interface representations.

In the next section, we will show how INCONT and PARTS values on the lexical level can be obtained from interface representations.

4 Building LRS representations

For building the LRS representation, only the interface representation built in the first step is required. Building the semantic representation is completely local and rather straightforward since all the required information is included in the interface representation.

In the beginning, INCONT and EXCONT are initialized to unspecified objects; PARTS to the empty list. This structure is successively built up to a full semantic representation by applying rewrite rules, which can be applied in any order. Each rule consists of a condition which is a partial description of the syntactic representation and a consequence, which is a set of operations for adding information to the semantic representation. These operations include: *identifying two objects*, *adding ob-*

jects to PARTS, and *adding subexpression constraints*. In the following, we discuss some exemplary rules to illustrate the nature of the procedure. We will use the name of a feature to refer to its value, e.g., $TENSE(\hat{\alpha})$ denotes the application of a function with the TENSE value as its name on $\hat{\alpha}$. The semantic representations given are a selection of items from PARTS and in some cases relevant subexpression constraints. The word to which the rule applies and the terms added by the rule are printed in boldface.

cat = verb: Besides a term defining the predicate, such as *schreiben(e)*, where *e* is the event variable, and a quantifier binding the variable, terms relating the event variable and the semantic arguments are introduced. If the argument is marked as predicative, the term $R(e, \hat{\alpha})$ is added, where R is the role label of the argument. α is constrained to contain the EXCONT-value of the argument. Otherwise, the term is simply $R(e, x)$, where x is the variable associated with the argument. (6) illustrates this rule. As the figure shows, the PARTS list also contains the subexpressions of the terms added.

cat = aux: Since the semantically relevant information was already transported to embedded predicates, auxiliaries are not interpreted at all. Their PARTS list is empty and their INCONT and EXCONT values are equated with those of their complement.

cat = preposition: The treatment of prepositions is designed to maximize the invariance of the semantic representation with regard to the variation between adjunct and argument PPs, between argument PPs and argument NPs, and between PPs and pro-forms such as *dahin* ‘thither’ and *woher* ‘whither’, which also receive *CAT preposition* in the syntactic analysis. Adjunct and argument PPs are assimilated already in the interface representation, where it is assumed that all prepositions select the head by MOD. The INCONT value of a preposition is always $PRED(A_1, A_2)$. If the ARGS list does not contain a complement, A_2 is set to a new variable, i.e., as the referent of a pronoun or as variable bound by an interrogative quantifier, which is built by a different rule operating on all interrogatives.

If there is in argument, A_2 is a first- or higher-order expression as explained for arguments of verbs. A_1 is the index of either the MOD value or the subject. Some aspects of the representation are illustrated by these examples:

- (8) *Hans war im Haus.*
Hans was in.the house
‘Hans was in the house’
 $\langle in(hans, x), haus(x), past(\hat{\alpha}), \dots \rangle$
with $\alpha \triangleright in(hans, x)$
- (9) *Wohin geht Hans?*
where goes Hans
‘Where does Hans go to?’
 $\langle gehen(e) \wedge subj(e, hans) \wedge$
 $wohin(e, x), interrog-q x \alpha, \dots \rangle$
- (10) *Hans geht nach Berlin.*
Hans goes to Berlin
‘Hans goes to Berlin.’
 $\langle gehen(e) \wedge subj(e, hans) \wedge$
 $nach(e, berlin), \dots \rangle$

cat = adverb, mod \neq none: The INCONT value of adverbial modifiers is $PRED(\hat{\alpha})$ with $\alpha \triangleright MOD|INCONT$, i.e., they outscope the core semantic contribution of the verb, while the relative scope of modifiers is not specified.

cat = noun, mod \neq none: For nominal modifiers, the term $MODTYPE(MOD|INDEX, INDEX)$ is added. This for example accounts for:

- (11) *das Buch des Kindes*
the book the.GEN child.GEN
‘the child’s book’
 $\langle POSS(x, y), buch(x), kind(y),$
 $def-q x [\alpha \circ \beta], def-q y [\gamma \circ \delta], \dots \rangle$
- (12) *Hans kochte zwei Stunden*
Hans cooked two hours
‘Hans cooked for two hours’
 $\langle kochen(e) \wedge subj(e, hans), TIME(e, x),$
 $stunde(x), 2 x [\alpha \wedge \beta], \dots \rangle$

tense \neq none: The term $TENSE(\hat{\alpha})$ with $\alpha \triangleright INCONT$ is added. Note that also predicative NPs and PPs will receive tense marking, as *Peter* in (13):

- (13) *Hans war Peter.*
Hans was Peter
‘Hans was Peter.’
 $\langle PAST(\hat{\alpha}), hans = peter, \dots \rangle$
with $\alpha \triangleright hans = peter$

The total system consists of 22 rules building the semantic representation from interface representations. Besides basic head-argument and head-modifier structures, some

of the covered phenomena are the verb complex, fronting in V2 sentences, relative clauses, coordination and interrogatives. Phenomena which we have not implemented yet include extraposition, ellipsis, focus-sensitive modifiers and discontinuous realization of NPs.

5 Experiment

5.1 Setup

To evaluate the quality and robustness of the systems, we ran two experiments on a small German learner corpus. In the first experiment, we ran the system on a manual dependency annotation and evaluated the resulting LRS structures. To evaluate the meaning of an ungrammatical learner sentence, we constructed a grammatical target hypothesis and then compared it with the automatic semantic analysis. Usually, only one possible analysis was deemed correct, with the exception of adverbs or adjectives modifying verbs, where both an intensional representation (e.g., $really(\hat{come}(e))$) and a representation using the verb's event variable ($real(f) \wedge subj(f, e)$) were admitted. In a second experiment, we ran the same procedure on automatic parses obtained from the statistical MaltParser (Nivre and Hall 2005) trained on Tüba-D/Z (Telljohann et al. 2004) to test the robustness against parsing errors.

5.2 The corpus used

Starting point is the CREG-109 corpus created by Ott and Ziai (2010), a sub-corpus of the Corpus of Reading Comprehension Exercises in German (CREG, Meurers et al. 2010). It consists of 109 sentences representing answers to reading comprehension exercises written by US college students at the beginner and intermediate levels of German programs. Of these, 17 sentences were classified as ungrammatical in that they clearly involved errors in word order, agreement, and case government.

The average sentence length is 8.26; the longest sentence has 17 tokens. CREG-109 was manually annotated by Ott and Ziai (2010) according to the dependency annotation scheme of Foth (2006), which distinguishes thirty-four dependency labels.

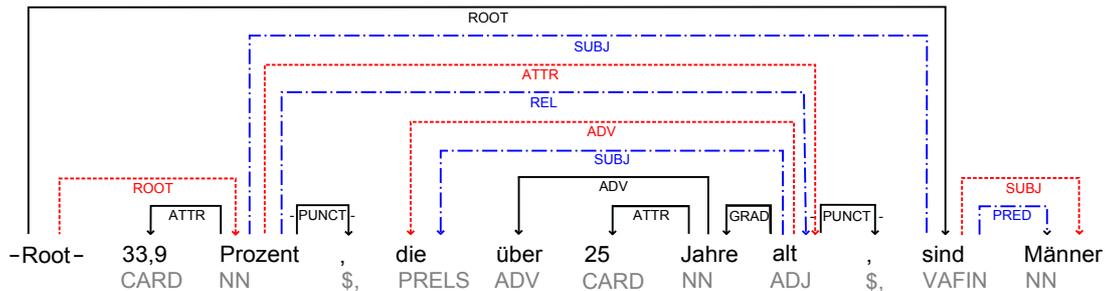
5.3 Results

Using the manual dependency annotation, the semantic representations of 86.2% of the grammatical sentences were fully correct. For 70.5% of the ungrammatical sentences, the analysis was a correct semantic representation of the target hypothesis. Using the automatic parses as input for semantic construction, 65.5% of the grammatical and 47.1% percent of the ungrammatical ones receive a correct representation.

5.4 Error analysis

Apart from ungrammatical input, most errors in the output arise from difficulties with coordination or ellipsis. Problems with coordination are even more severe in the case of automatic parses. Other typical problems caused by noisy parser output are the confusion of subjects and complements, PP-attachment and missing dependencies which isolate some words. The impact of other parser errors on the semantic output is often minor due to the flexibility of the semantic representation language. For example, errors in the attachment of adverbs in the verbal complex are handled to some extent by scope underspecification.

In other cases, even clearly ungrammatical structures receive a correct semantic interpretation, even if an automatic parse which differs from the manual annotation is used. An example for this is given in Figure 1, which Ott and Ziai (2010) give as an example of bad parser performance on ungrammatical input. The dashed blue dependencies are the human annotation, and the red dotted the automatic parse. Inside the relative clause, the copula is missing and the relative clause has no finite verb. The human annotators took the predicative adjective *alt* as the head of the relative clause and *die* as its subject, which yields a correct semantic representation. The parser, on the other hand, interpreted the predicative adjective as adjectival modifier and the relative pronoun as an adverbial modifier. This usage of pronouns is not expected in correct parses and there is no rule dealing with it; therefore, the semantic contribution is empty. Because the noun modified by an adjective is interpreted like an adjective's subject, the ad-



Target Hypothesis: *33,9 Prozent, die über 25 Jahre alt [sind], sind Männer.*
 33.9 percent who over 25 years old are are men

Figure 1: Parse of an ungrammatical sentence

jective has exactly the same semantic representation. Thus, the correct semantic representation is obtained for the NP *33,9 Prozent, die über 25 Jahre alt*. The example illustrates that abstracting away from the syntactic structure before building the semantic representation can help the system perform well for unexpected syntactic structures which may arise from learner and parser errors.

6 Related work

Spreyer and Frank (2005) use term rewrite rules to build RMRS representations for the TIGER Dependency Bank for German. RMRS is a robust version of Minimal Recursion Semantics, an underspecified semantic formalism used in HPSG. Jakob et al. (2010) present an RMRS system for the Prague Dependency Treebank of Czech. Our work differs in that the input data is learner language and that the semantic representation language is LRS. Furthermore, the dependency parses our system uses contain much less syntactic information than the two dependency banks, in particular no tectogrammatical information.

The first step in our system is related to work on automatically deriving richer feature structure representations such as f-structures from treebank parses (cf. Frank 2000; Cahill et al. 2002). The treebanks used likewise contain more information than the bare dependency parses we use.

7 Conclusion

We presented a system that automatically derives underspecified, model-theoretic semantic representations from dependency parses

of German learner sentences. We argued that it is beneficial to first transform dependency structures into syntax-semantics interface representations, which reduce the syntactic structure to semantically important information. In particular, they are invariant under phenomena such as passivization and dislocation. We discussed how such representations can be obtained from dependency parses and presented an algorithm for reconstructing the argument structures of verbs in the German coherent verbal complex, where arguments are commonly realized as dependents of other verbs. We showed that Lexical Resource Semantics, although developed for HPSG, can straightforwardly be adapted to dependency-based syntactic representations, and we presented a sample of a simple rule system building semantic representations in LRS from interface representations. Our evaluation showed that the architecture can often deal robustly with learner and parser errors. In future work, we intend to put these results on a more expressive quantitative basis by evaluating the system on larger native corpora.

References

- Gunnar Bech, 1955. *Studien über das deutsche verbum infinitum*. Historisk-filologiske Meddelelser udgivet af Det Kongelige Danske Videnskabernes Selskab. Bind 35, no. 2, 1955; Bind 36, no. 6, 1957; Kopenhagen. Reprinted 1983, Tübingen: Max Niemeyer.
- Johan Bos and Katja Markert, 2006. When logical inference helps determining textual entailment (and when it doesn't). In *The Second PASCAL Recognising Textual En-*



- tailment Challenge. Proceedings of the Challenges Workshop*. pp. 98–103.
- Aoife Cahill, Mairead McCarthy, Josef van Genabith and Andy Way, 2002. Parsing with PCFGs and Automatic F-Structure Annotation. In *Proceedings of the LFG-02 Conference*. CSLI Publications.
- Ann Copestake, Dan Flickinger, Carl Pollard and Ivan Sag, 2005. Minimal Recursion Semantics: An Introduction. *Research on Language & Computation*, 3:281–332.
- Ido Dagan, Bill Dolan, Bernardo Magnini and Dan Roth, 2009. Recognizing textual entailment: Rational, evaluation and approaches. *Natural Language Engineering*, 15(4):i–xvii.
- Kilian Foth, 2006. *Eine umfassende Constraint-Dependenz-Grammatik des Deutschen*. Manual, Universität Hamburg.
- Anette Frank, 2000. Automatic F-structure Annotation of Treebank Trees. In *Proceedings of LFG-00*. CSLI Publications.
- Tilman N. Höhle, 1978. *Lexikalistische Syntax. Die Aktiv-Passiv-Relation und andere Infinitkonstruktionen im Deutschen*. Max Niemeyer, Tübingen.
- Max Jakob, Marketa Lopatkova and Valia Kordoni, 2010. Mapping between Dependency Structures and Compositional Semantic Representations. In *Proceedings of LREC 2010*.
- Ronald M. Kaplan and Joan Bresnan, 1995. Lexical-Functional Grammar: A Formal System for Grammatical Representations. In Mary Dalrymple, John T. Maxwell, Ronald M. Kaplan and Annie Zaenen (eds.), *Formal issues in Lexical-Functional Grammar*, CSLI Publications, Stanford, CA.
- Chin-Yew Lin, 2004. Rouge: a package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*.
- Chin-Yew Lin and Franz Josef Och, 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the ACL*. Barcelona.
- Christopher D. Manning and Ivan A. Sag, 1998. Argument Structure, Valence, and Binding. *Nordic Journal of Linguistics*, 21.
- Igor A. Mel'cuk, 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press.
- Detmar Meurers, 2000. Lexical Generalizations in the Syntax of German Non-Finite Constructions. Phil. dissertation, Eberhard-Karls-Universität Tübingen. <http://pur1.org/dm/papers/diss.html>.
- Detmar Meurers, Niels Ott and Ramon Ziai, 2010. Compiling a Task-Based Corpus for the Analysis of Learner Language in Context. In *Proceedings of Linguistic Evidence*. Tübingen, pp. 214–217.
- Joakim Nivre and Johan Hall, 2005. Malt-parser: A language-independent system for data-driven dependency parsing. In *Proceedings of the Fourth Workshop on Treebanks and Linguistic Theories*. pp. 13–95.
- Niels Ott and Ramon Ziai, 2010. Evaluating Dependency Parsing Performance on German Learner Language. In *Proceedings of the Ninth International Workshop on Treebanks and Linguistic Theories (TLT9)*.
- Carl Pollard and Ivan A. Sag, 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago, IL.
- Frank Richter, 2000. A Mathematical Formalism for Linguistic Theories with an Application in Head-Driven Phrase Structure Grammar. Phil. dissertation, Eberhard-Karls-Universität Tübingen.
- Frank Richter and Manfred Sailer, 2003. Basic Concepts of Lexical Resource Semantics. In Arnold Beckmann and Norbert Preining (eds.), *ESSLLI 2003 – Course Material I*. Kurt Gödel Society, Wien, volume 5 of *Collegium Logicum*, pp. 87–143.
- Kathrin Spreyer and Anette Frank, 2005. Projecting RMRS from TIGER Dependencies. In *The Proceedings of the 12th International Conference on HPSG*. CSLI Publications, Stanford, pp. 354–363.
- Heike Telljohann, Erhard Hinrichs and Sandra Kübler, 2004. The Tüba-D/Z Treebank: Annotating German with a Context-Free Backbone. In *Proceedings of LREC 2004*. pp. 2229–2235.

Looking Behind the Scenes of Syntactic Dependency Corpus Annotation: Towards a Motivated Annotation Schema of Surface-Syntax in Spanish

Alicia Burga¹, Simon Mille¹, and Leo Wanner^{1,2}

¹Universitat Pompeu Fabra ²ICREA, Barcelona

firstname.lastname@upf.edu

Abstract

Over the last decade, the prominence of statistical NLP applications that use syntactic rather than only word-based shallow clues increased very significantly. This prominence triggered the creation of large scale treebanks, i.e., corpora annotated with syntactic structures. However, a look at the annotation schemata used across these treebanks raises some issues. Thus, it is often unclear how the set of syntactic relation labels has been obtained and how it can be organized so as to allow for different levels of granularity in the annotation. Furthermore, it appears questionable that despite the linguistic insight that syntax is very much language-specific, multilingual treebanks often draw upon the same schemata, with little consideration of the syntactic idiosyncrasies of the languages involved. Our objective is to detail the procedure for establishing an annotation schema for surface-syntactic annotation of Spanish verbal relations and present a restricted set of easy-to-use criteria which facilitate the decision process of the annotators, but which can also accommodate for the elaboration of a more or a less fine-grained tagset. The procedure has been tested on a Spanish 3,500 sentence corpus, a fragment of the AnCora newspaper corpus.

1 Introduction

Over the last decade, the prominence of statistical Natural Language Processing (NLP) applications (among others, machine translation parsing, and text generation) that use syntactic rather than only word-based shallow clues increased very significantly. This prominence triggered, in its turn, the creation of large scale treebanks, i.e., corpora annotated with syntactic structures, needed for training of statistical algorithms; see, among others, the Penn Treebank (Marcus et al., 1993) for English, the Prague Dependency Treebank (Hajič et al., 2006) for Czech,

the Swedish Talbanken05 (Nivre et al., 2006), the Tiger corpus (Thielen et al., 1999) for German, and the Spanish, Catalan, and Basque AnCora treebank (Taulé et al., 2008). Even though this is certainly a very positive tendency, a look at the annotation schemata used across the treebanks of different languages raises some issues. Thus, despite the linguistic insight that syntax is very much language-specific, many of them draw upon the same more or less fine-grained annotation schemata, i.e., sets of syntactic (dependency) relations, with little consideration of the languages themselves. Often, it is unclear how the individual relations in these sets have been determined and in which linguistic theory they are grounded, and occasionally it is not obvious that the annotation schema in question uses only syntactic (rather than also semantic) criteria.

Our objective is to detail the process of elaboration of an annotation schema for surface-syntactic verbal relation annotation of Spanish corpora,¹ which has already been used to annotate a 3,500 sentence corpus of Spanish. The corpus is a fragment of the AnCora corpus which consists of newspaper material.

Our work draws heavily on the principles of the Meaning-Text Theory (MTT) as far as the nature of dependency in general and (surface-) syntactic dependency in particular are concerned.

In the next section, we analyze the state of affairs in some of the well-known dependency treebanks and justify why we set out to write this paper. In Section 3, we present the notion of surface-syntactic structure and the general principles of dependency as defined in MTT. Section 4 outlines the annotation schema we propose and the principles used to distinguish between different relations. Section 5, finally, summarizes the paper and draws some conclusions.

¹“Surface-syntactic” is used here in the sense of the Meaning-Text Theory (Mel’čuk, 1988).

2 A Glance Behind the Scenes

It is well-known that surface-syntactic relations (SSyntRels) as usually used in dependency treebanks are language-specific. A dependency relation annotation schema should thus, on the one hand, facilitate the annotation of all language-specific syntactic idiosyncrasies, but, on the other hand, offer a motivated generalization of the relation tags such that it could also serve for applications that prefer small generic dependency tag sets. However, as already mentioned above, in a number of dependency treebanks containing corpora in different languages, the same arc tag set is used for all languages involved—no matter whether the languages in question are related or not. For instance, AnCora (Taulé et al., 2008) contains the related Spanish and Catalan, but also Basque; the treebank described in (Megyesi et al., 2008) contains Swedish and Turkish, etc. This makes us think that little work has been done concerning the definition of the relation labels. In general, for all parallel and non-parallel treebanks that we found—the Czech PDT2.0-PDAT (Hajič et al., 2006) and (Hajič and Zemánek, 2004)) and PCET (Čmejrek et al., 2004), the English-German FuSe (Cyrus et al., 2003), the English-Swedish LinEs (Ahrenberg, 2007), the English Penn Treebank (Marcus et al., 1993), the Swedish Talbanken (Nivre et al., 2006), the Portuguese Bosque (Afonso et al., 2002), the Dutch Alpino (Van der Beek et al., 2002), etc.—the justification of the choice of dependency relation labels is far from being central and is largely avoided. This may lead to the conclusions that the selection of the relations is not of great importance or that linguistic research already provides sets of relations for a significant number of languages. Each of these two conclusions is far from being correct. In our work, we found the question of the determination of SSyntRels very crucial, and we observed the lack of an appropriate description of the language through a justified description of the SSyntRels used even for languages for which treebanks are available and widely used.

In MTT, significant work has been carried out on SSyntRels—particularly for English and French. Thus, (Mel’čuk and Percov, 1987; Mel’čuk, 2003) present a detailed inventory of SSyntRels for English, and (Iordanskaja and Mel’čuk, 2009) sug-

gest criteria for establishing an inventory of labeled SSyntRels headed by verbs as well as a preliminary inventory of relations for French. However, we believe that both inventories are not thought for large scale corpus annotation to be used in statistical NLP in that the criteria are generally difficult to apply and do not separate enough surface-syntactic phenomena from the phenomena at other levels of the linguistic description. For instance, one important distinction in (Iordanskaja and Mel’čuk, 2009) is whether a dependent is actantial or not—in other words, if a dependent forms part of the definition of its governor or not—which is, however, a clear semantic distinction.

We attempt to avoid recourse to deep criteria. Instead, we replace deep criteria by a list of strictly syntactically motivated, easy-to-use criteria in order to make their application efficient on a large scale, and detail the process from the very beginning. This list is as reduced as possible, but still sufficient to capture fine-grained idiosyncrasies of Spanish. Obviously, we intensely use the cited works on SSyntRels in MTT as a source of inspiration.

3 MTT Guide to SSynt Dependencies

The prerequisite for the discussion of the compilation of a set of SSyntRels for a particular language is a common understanding of (i) the notion of a surface-syntactic dependency structure (SSyntS) that forms the annotation of a sentence in the corpus; (ii) the principles underlying the determination of a dependency relation, i.e., when there is a dependency relation between two lexical units in a sentence, and what is the direction of this dependency, or in other words, who is the governor and who is the dependent. In the presentation of both, we widely follow (Mel’čuk, 1988).

3.1 Definition of SSyntS

In MTT, an SSyntS is defined as follows:

Definition 1 (Surface-Syntactic Structure, $SSyntS$)

Let L , G_{sem} and R_{ssynt} be three disjoint alphabets, where L is the set of lexical units (LUs) of a language \mathcal{L} , G_{sem} is the set of semantic grammemes, and R_{ssynt} is the set of names of surface-syntactic relations (or grammatical functions).

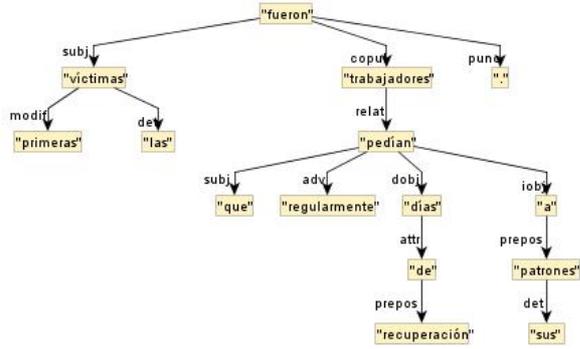


Figure 1: SSyntS of the sentence *Las primeras víctimas fueron trabajadores que pedían regularmente días de recuperación a sus patrones*. ‘The first victims were employees who regularly asked days-off to their bosses’

An SSyntS of \mathcal{L} , S_{SSynt} , is a quintuple over $L \cup G_{sem} \cup R_{ssynt}$ of the following form:

$$S_{SSynt} = \langle N, A, \lambda_{l_s \rightarrow n}, \rho_{r_s \rightarrow a}, \gamma_{n \rightarrow g} \rangle$$

where

- the set N of nodes and the set A of directed arcs (or branches) form an unordered dependency tree (with a source node n^s and a target node n^t defined for each arc),
- $\lambda_{l_s \rightarrow n}$ is a function that assigns to each $n \in N$ an $l_s \in L$,
- $\rho_{r_s \rightarrow a}$ is a function that assigns to each $a \in A$ an $r_s \in R_{ssynt}$,
- $\gamma_{n \rightarrow g}$ is a function that assigns to the name of each LU associated with a node $n_i \in N$, $l_i \in \lambda_{n \rightarrow g}(N)$, a set of corresponding grammemes $G_t \in G_{sem}$.

For illustration, consider the SSyntS of a Spanish sentence in Figure 1.²

We are particularly interested in the assignment of surface-syntactic relation labels to the arcs (i.e., the function $\rho_{r_s \rightarrow a}$). These labels are of the nature as used by many other treebanks: ‘subject’, ‘direct/indirect object’, ‘copulative’, ‘modificative’, ‘determinative’, ‘adverbial’, etc, i.e., *grammatical*

²The nominal node labels reflect the number (*víctimas* ‘victims’, *trabajadores* ‘workers’, *patrones* ‘bosses’) only to facilitate the reading; semantic plural is encoded as a grammeme in terms of an attribute/value pair on the node: *number=PL*. Note also that we consider each node label to be a disambiguated word, i.e., lexical unit (LU). For details on grammemes, see (Mel’čuk, 2006).

functions. We want to determine when to use each of them and how to build the tag set such that it can be enriched or reduced in a prescribed way under clearly defined conditions. For instance, in Figure 1, the indirect object of the verb *pedían* ‘asked_{PL}’ is introduced by a preposition *a* ‘to’. However, in Spanish, direct objects can also be introduced by it. So, obviously, looking at the units of the sentence is not enough to establish the dependency relations.

Each relation has to be associated with a set of central properties. These properties must be clearly verifiable. For instance, a direct object is cliticizable by an accusative pronoun, an indirect object by a dative pronoun, and every relation must have one type of dependent that can be used with any governor.

3.2 Principles for Determination of SSynt-Dependencies

The central question faced during the establishment of the SSyntS as defined above for each sentence of the corpus under annotation is related to:

- the elements of A : when is there a dependency between two nodes labeled by the LUs l_i and l_j and what is the direction of this dependency,
- the elements of R_{ssynt} : what are the names of the dependencies, how they are to be assigned to $a \in A$, and how they are to be distinguished,

or, in short, to the determination of SSynt-Dependencies. In what follows, we address this question in terms of two corollaries.

Corollary 1 (Dependency between nodes) *Given any two unordered nodes n_1 and n_2 , labeled by the LUs l_1 and l_2 respectively, in the sentence S of the corpus, there is a dependency between n_1 and n_2 if either*

- (a) *in order to position l_i in S , reference must be made to l_j , with $i, j = 1, 2$ and $i \neq j$ (linear correlation criterion)*

and

- (b) *between l_i and l_j or between syntagms of which l_i and l_j are heads ($i, j = 1, 2$ and $i \neq j$), a prosodic link exists (prosodic correlation criterion)*

or

- (c) *l_i triggers agreement on l_j ($i, j = 1, 2$ and $i \neq j$) (agreement criterion)*

Thus, in *John has slept well today*, *John* has to be positioned before the auxiliary *has* (or after in a question) and a prosodic link exists between *John* and the syntagm headed by *has*. This means that *John* and *has* are likely to be linked by a dependency relation. *Well* has to be positioned compared to *slept* (not compared to *has*), hence there is a dependency between *slept* and *well*.

With respect to agreement, we see that the verb is *has* and not *have*, as it would be if we had *The boys* instead of *John*. This verbal variation in person, which depends on the preverbal element, implies that a dependency links *John* and *has*.

Once the dependency between two nodes has been established, one must define which node is the governor and which one is the dependent, i.e., the direction of the SSynt arc that links those two nodes. The following corollary handles the determination of the direction of the dependency:

Corollary 2 (Direction of a dependency relation)

Given a dependency arc a between the nodes n_1 and n_2 of the SSyntS of the sentence S in the corpus, n_1 is the governor of n_2 , i.e., n_1 is the source node and n_2 is the target node of a if

(a) *the passive valency (i.e., distribution) of the group formed by the LU labels l_1 and l_2 of n_1/n_2 and the arc between n_1 and n_2 is the same as the passive valency of l_1 (passive valency criterion)*

or

(b) *l_1 as lexical label of n_1 can be involved in a grammatical agreement with an external element, i.e., a label of a node outside the group formed by LU labels l_1 and l_2 of n_1/n_2 and the arc between n_1 and n_2 (morphological contact point criterion)*

If neither (a) nor (b) apply, the following weak criteria should be taken into account:

- (c) *if upon the removal of n_1 , the meaning of S is reduced AND restructured, n_1 is more likely to be the governor than n_2 (removal criterion),*
- (d) *if n_1 is not omissible in S , it is more likely to be the governor than n_2 (omissibility criterion),*
- (e) *if l_2 as label of n_2 needs (“predicts”) l_1 as label of n_1 , n_2 is likely to be a dependent of n_1 (predictability criterion).*

As illustration of the passive valency criterion,³ consider the group *the cats*. It has the same distribution as *cats*: both can be used in exactly the same paradigm in a sentence. On the other side, *the cats* does not have the distribution of *the*. We conclude that *cats* is the head in the group *the cats*. It is important to note that, for instance, in the case of prepositional groups, the preposition does not have its own passive valency since it always needs an element directly after it. It does not prevent the passive valency criterion from applying since, e.g., the distribution of *from [the] house* is not the same as the distribution of *house*. It is the presence of the preposition that imposes on the group a particular distribution.

The morphological contact point criterion is used as follows: considering the pair *sólo felinos* in *sólo felinos ronronean* ‘only felines_{PL} purr_{PL}’, *felinos* is the unit which is involved in the agreement with an external element, *ronronean*. As a consequence, *felinos* is more prone to be the governor of *sólo*.

We illustrate the omissibility criterion in Section 4.2, but do not elaborate on the removal criterion nor on the predictability criterion; for more details see (Mel’čuk, 1988).

3.3 Labelling the dependencies

With the two corollaries from above at hand, we should be able to state when there is a dependency arc between two nodes, and which node governs which other node. Now, labels to the dependency arcs need to be assigned. The assignment may be very intuitive and straightforward (as, e.g., the assignment of *subject* to the arc between *caen* ‘fall’ and *bolas* ‘balls’ in *bolas caen*, lit. ‘balls fall’, or the assignment of *object* to the arc between Sp. *tiran* ‘throw’ and *bolas* ‘balls’ in *tiran bolas*, lit. ‘[they] throw balls’) or less clear (as, e.g., the assignment of a label to the dependency arc between *caen* ‘fall’ and *bolas* ‘balls’ in *caen bolas*, lit. ‘[it] falls balls’: is it the same as in *bolas caen*, namely *subject* or a different one?).⁴

³For the definition of the notion “passive valency”, see (Mel’čuk, 1988).

⁴We do not encode linear order in the SSyntRels: in practice, this allows us to limit the tagset size. However, it does not mean that some relations do not impose a particular linear order between the governor and dependent (see Section 4.2). The dependency tree as such remains unordered.

The following corollary addresses the question whether two given dependency arcs are to be assigned the same or different labels:

Corollary 3 (Different labels) *Be given an arc a_1 and an arc a_2 such that*

- a_1 holds between the nodes $n_{s_{a_1}}$ (labeled by $l_{s_{a_1}}$) and $n_{st_{a_1}}$ (labeled by $l_{t_{a_1}}$), with the property set $P_{a_1} := \{p_{a1_1}, p_{a1_2}, \dots, p_{a1_i}, \dots, p_{a1_n}\}$,

- a_2 holds between the nodes $n_{s_{a_2}}$ (labeled by $l_{s_{a_2}}$) and $n_{st_{a_2}}$ (labeled by $l_{t_{a_2}}$), with the property set $P_{a_2} := \{p_{a2_1}, p_{a2_2}, \dots, p_{a2_j}, \dots, p_{a2_m}\}$

Then, $\rho_{r_s \rightarrow a}(a_1) \neq \rho_{r_s \rightarrow a}(a_2)$, i.e., a_1 and a_2 are assigned different labels, if

(a) $\exists p_k : (p_k \in P_{a_1} \wedge p_k \notin P_{a_2}) \vee (p_k \in P_{a_2} \wedge p_k \notin P_{a_1})$ and p_k is a central property

or

(b) one of the following three conditions apply; cf. (Mel'čuk, 1988):

1. semantic contrast condition: $l_{s_{a_1}}$ and $l_{s_{a_2}}$ and $l_{t_{a_1}}$ and $l_{t_{a_2}}$ are pairwise the same word-forms, but either $l_{s_{a_1}}$ and $l_{s_{a_2}}$ or $l_{t_{a_1}}$ and $l_{t_{a_2}}$ have different meanings.

2. prototypical dependent condition (quasi-Kunze property): given the prototypical dependents d_{p_1} of a_1 and d_{p_2} of a_2 , when $l_{t_{a_1}}$ in $l_{s_{a_1}} - a_1 \rightarrow l_{t_{a_1}}$ is substituted by d_{p_2} the grammaticality of $l_{s_{a_1}} - a_1 \rightarrow l_{t_{a_1}}$ is affected or when $l_{t_{a_2}}$ in $l_{s_{a_2}} - a_2 \rightarrow l_{t_{a_2}}$ is substituted by d_{p_1} the grammaticality of $l_{s_{a_2}} - a_2 \rightarrow l_{t_{a_2}}$ is affected.

3. SSyntRel repeatability criterion: If $l_{t_{a_1}}$ and its dependency a_1 from $l_{s_{a_1}}$ can be repeated and $l_{t_{a_2}}$ and its dependency a_2 from $l_{s_{a_2}}$ cannot (or vice versa).

Condition (a) entails first of all that a relation should have clear properties associated to it. Associating properties to a relation is exactly what means to define a relation. This can only be done in opposition to other relations, which means that this is the result of numerous iterations after the inspection of numerous examples. As a consequence, paradoxically, the list of properties of a relation is one of the last things which is defined.⁵

⁵A restricted property set of the *direct objectival* relation in Spanish includes: the direct object (1) is cliticizable (2) by an accusative pronoun, (3) can be promoted, (4) does not receive any agreement, and (5) is typically a noun.

The semantic contrast condition (b1) states that for a given relation and a given minimal pair of LUs, there must not be any semantic contrast; the arc orientation has to be the same for both members of the minimal pair, and the deep-morphologic representation should be different (different possible orders or different case on the dependent for instance). Both pairs have the property to be able to occupy the same syntactic role in a sentence. Consider the two LUs *matar* 'kill' and *gatos* 'cats': they can form an ambiguous sentence *Matan gatos*, lit. 'Cats kill'/'[They] kill cats'. The ambiguity cannot be explained by the difference of meaning of the components of the sentence (since they are the same). Hence, the semantic contrast criterion prevents both dependencies to be the same; in one case, *gatos* is subject, and in the other case, it is object of *matar*.

The semantic contrast condition does not apply to *una casa* 'indefinite + house' / *una casa* 'one + house' because *una* does not have the same meaning (i.e., is not the same lexeme) in both cases.

The quasi-Kunze criterion (b2) states that any SSyntRel must have a prototypical dependent, that is, a dependent which can be used for ANY governor of this SSyntRel; see (Mel'čuk, 2003). Consider, for illustration, *poder*—R→*caer* 'can fall' vs. *cortar*—R→*pelo* 'cut hair': it is not possible to have an N as dependent of *poder* 'can' nor an V_{inf} as dependent of *cortar* 'cut'. More generally, no element of the same category can appear below both *poder* and *cortar*. This implies that the prototypical dependents in both cases do not coincide, so it is not the same relation.

The SSyntRel repeatability criterion (b3) indicates that a particular SSyntRel should be, for any dependent, either always repeatable or never repeatable. If one dependent can be repeated and another one cannot, then we have two different relations. In a concrete case, we can start with the hypothesis that we have ONE relation R for which we want to know if it is suitable to handle two dependents with different properties (in particular, two different parts-of-speech). If the same relation R can be used to represent the relation, for instance, between a noun and an adjective, and, on the other side, between a noun and a numeral quantifier, R should be either repeatable or not repeatable in both cases. We observe

that R is repeatable for adjectives but not for quantifiers and conclude, thus, that R should be split in two relations (namely ‘modifier’ and ‘quantificative’).

4 Towards a SSynt Annotation Schema

In Section 3, the general principles have been presented that allow us to decide when two units are involved in a dependency relation and who is the governor. Furthermore, some generic cases have been identified in which it seems clear whether a new relation should be created or not. With these principles at hand, we can set out for the definition of a motivated SSynt annotation schema. To be taken into account during this definition is that (a) (unlike the available MTT SSyntRel sets,) the schema should cover only syntactic criteria; (b) the granularity of the schema should be balanced in the sense that it should be fine-grained enough to capture language-specific syntactic idiosyncrasies, but be still manageable by the annotator team (we are thinking here of decision making and inter-agreement rate). The latter led us target a set of 50 to 100 SSyntRels.

4.1 Principles for the criteria to distinguish between different relations

The following properties are particularly important:

- **Applicability:** The criteria should be applicable to the largest number of cases possible. For instance, a head and a dependent always have to be ordered, so a criterion implying order can be applied to every relation whatever it is. One advantage here is to keep a set of criteria of reasonable size, in order to avoid to have to manage a large number of criteria which could only be applied in very specific configurations. The other advantage in favouring generic criteria is that it makes the classification of dependency relations more readable: if a relation is opposed to another using the same set of criteria, the difference between them is clearer.

- **Visibility:** When applying a criterion, an annotator would rather *see* a modification or the presence of a particular feature. Indeed, we try to use only two types of criteria: the ones that transform a part of the sentence to annotate—promotion, mobility of an element, cliticization, etc.—, and the ones that check the presence or absence of an element in the sentence to annotate (is there an agreement on the depen-

dent? does the governor impose a particular preposition? etc.). In other words, we avoid semantically motivated criteria. The main consequence of this is the absence of opposition complement/attribute as discriminating feature between syntactic relations.

- **Simplicity:** Once the annotator has applied a criterion, he/she must be able to make a decision quickly. This is why almost all criteria involve a binary choice.

All of the resulting selected criteria presented below have been used in one sense or the other in the long history of grammar design. However, what we believe has not been tackled up to date is how to conciliate in a simple way fine-grained syntactic description and large-scale application for NLP purposes. In what follows, we present a selection of the most important criteria we use in order to assign a label to a dependency relation. Then, we show how we use them for the annotation of a Spanish corpus with different levels of detail.

4.2 Main criteria to distinguish between different relations

- **Type of linearization:** Some relations are characterized by a rigid order between the head and the dependent (in any direction), whereas some others allow more flexibility with respect to their positioning. Thus, e.g., the relations that connect an auxiliary with the verb imply a fixed linearization: the auxiliary (head) always appears to the left of the verb (dependent):

He comido mucho. ‘[I] have eaten a-lot’ /

**Comido he mucho.*

On the other hand, even if Spanish is frequently characterized as an SVO language, the relation ‘subject’ does allow flexibility between the head and the dependent:

Juan come manzanas. ‘Juan eats apples’/

Come Juan manzanas./Come manzanas Juan.

Given that it is possible to apply this criterion to all the relations, the linearization criterion is very relevant to our purposes.

- **Canonical order:** As just stated, some relations are more flexible than others with respect to the order between head and dependent. When the order is not restricted, there is usually a canonical order. Thus, although it is possible to have a postverbal subject, the canonical order between the subject and

the verb is that the former occurs at the left of the latter. On the other hand, the relations introducing the non-clitic objects have the opposite canonical order, i.e. the object appears at the right of the verb.

• **Adjacency to the governor:** There are some relations that require that the head and the dependent are adjacent in the sentence, and only accept a very restricted set of elements to be inserted between them, but there are some other relations that allow basically any element to appear between them. We believe that the fact to keep a dependent very close in the sentence is an important syntactic feature. All the relations involving clitics belong to the first type, and a relation such as determinative belongs to the second type:

Cada día, lo miraba. ‘Every day, [I] watched it’/

**Lo cada día miraba.*

El hombre bueno. lit. ‘The guy good’ /

El buen hombre.

• **Cliticization:** Concerning only elements for which the order between the verbal head and its dependent is not restricted, an important criterion refers to the possibility for the dependent to be replaced or duplicated by clitic pronouns. Thus, the relation *indirect object* allows cliticization, as opposed to the *oblique object* that does not:

Miente—IObj→*a Carla.* / *Le miente.* / *A Carla le miente.*

lit. ‘[He] lies to Carla.’ / ‘[He] to-her lies.’ / ‘To Carla [He] to-her lies.’

Invierte—OblObj→*en bolsa.* / **La invierte.* / **En bolsa la invierte.*

lit. ‘[He] inverts in stock-market.’ / ‘[He] in-it inverts.’ / ‘In stock-market [He] in-it inverts.’

• **Promotion/demotion:** Promotion and demotion refer to the possibility of becoming, respectively, a closer or a further argument in a parallel sentence. Thus, the dependent of the relation *direct object* can be promoted to the dependent of the relation *subject* in a passive sentence (and, from the opposite point of view, the subject can be demoted to the dependent of the relation *agent* in a passive sentence):

Juan compuso las canciones. / *Las canciones fueron compuestas por Juan.*

‘Juan wrote the songs’ / ‘The songs were written by Juan’

Cliticization and promotion/demotion can only be applied if the head is a finite verb and from this per-

spective, do not seem comply with the Applicability principle. However, since there are many different relations that can appear below a verb, this is not totally true. In addition, they are very efficient with respect to the other two principles, Visibility and Simplicity.

• **Agreement:** Agreement appears when head and dependent share some morphological features, such as gender, number, person, etc., which one passes to the other. The agreement actually depends on two parameters: on the one hand, the target of the agreement must have a Part of Speech which allows agreement, and on the other hand, the dependency relation itself must allow it. For example, the *copulative* relation allows agreement, but if the dependent is not an adjective, it is not mandatory: *Pedro y Carla son relajados* ‘Pedro and Carla are relaxed_{PLU}’ as opposed to *Pedro y Carla son una pareja*, ‘Pedro and Carla are a couple_{SING}’. Inversely, the past participle in the perfect analytical construction is intrinsically prone to agreement (as shows the second example that follows), but the relation does not allow it: *Carla está perdida*, ‘Carla is lost_{FEM}’ as opposed to *Carla ha perdido* ‘Carla has lost_{noFEM}’. This is why the notion of prototypical dependent is important (see next paragraph): if a relation licences agreement, it doesn’t mean that any dependent must have agreement, but that there is always agreement for its prototypical dependent.

There are different types of agreements allowed by a syntactic relation:

- dependent agrees with head:
sillas—modificative→*rotas* ‘chairs broken_{FEM.PL}’,
- head agrees with dependent:
Juan←subject—*viene* ‘Juan comes’,
- dependent agrees with another dependent:
Juan←subject—*parece*—copulative→ *enfermo* ‘Juan seems sick_{MASC.SG}’.

When there is agreement, secondary criteria can be applied, concerning the type of inflection of the agreeing element: in some cases, the agreement can vary, in some cases it cannot (see the opposition between *subject* and *quotative subject* in the next subsection).

• **Prototypical dependent:** As mentioned in Section 3, every relation must have a prototypical dependent. This criterion is more useful for designing

the set of dependency relations than for assigning a tag to a relation, since it involves a generalization over a large number of cases, which are not accessible during the process of annotation. However, it can be used during annotation as well, especially in order to infirm/confirm a relation: if a dependent of a SSyntRel cannot be replaced by the prototypical dependent of this relation, then the relation should be changed. It can also be useful when looking for a relation in the hierarchical representation of the criteria (see Figure 2), for instance in combination with the Agreement criterion: if the pair *son-??*→*pareja* in the sentence *Pedro y Carla son una pareja* ‘Pedro and Carla are a couple_{SING}’ has to be annotated, although there is no visible agreement, the native speaker annotator has the knowledge that the typical dependent in that case for that verb is an adjective and then should consider that an agreement is usually involved.

- **Part-Of-Speech of the Head:** The actual PoS of the governor is relevant in that there are very few syntactic dependents that behave the same with heads of different syntactic categories once a certain level of detail has been reached in the annotation. As a consequence, we decided to separate the tags of our tagset by PoS of the governor.

- **Governed Preposition/ Conjunction/ Gramme (P/C/G):** There are some relations that require the presence of a preposition, a subordinating conjunction or a grammeme. For instance, the relation *oblique object* implies the presence of a preposition which has no meaning to introduce the dependent (*viene de comer* ‘he/she has just eaten’), and the relation *subordinate conjunctive* requires the presence of a feature in the verb indicating that it is finite.

- **Dependent omissibility:** This syntactic criterion is defined within an “out-of-the-blue” context, given that otherwise it is very difficult to determine whether or not a dependent is omissible: it is always possible to create pragmatic contexts whereas the dependent can be perfectly omitted. There are two cases: on the one hand, relations such as *prepositional* always require the presence of the dependent and, on the other hand, relations as *modifier* do not require the presence of the dependent. Consider:

Juan viene para—prepos→*trabajar*. /

**Juan viene para*.

‘Juan comes to work’ / ‘Juan comes to’

Tiene sillas—modif→*verdes*. / *Tiene sillas*.

lit. ‘[He] has chairs green’ / ‘[He] has chairs’

4.3 Application of the Schema to Spanish

We organized all the criteria into a tree-like hierarchy so that if an annotator identifies a pair a governor/dependent, but wonders which relation holds between the two, he only has to follow a path of properties that leads to the relation. The order in which the criteria are applied is only important for a generalization over the relations, since it allows to keep close in the graphical representation the relations that have the same type (see Figure 2).

Due to space restrictions, we only present in this paper a part of the hierarchy, namely, the relations headed by a verb which do not impose a rigid order between governor and dependent; our complete hierarchy contains 70 different arc labels and covers the annotation of a 100,000 word corpus. We use here nine criteria: removability of dependent, possible cliticization, agreement type, inflection type, PoS of prototypical dependent, behaviour to promotion, presence of governed P/C/G, presence of quotes, presence of parentheses or dashes. With this level of detail, we get sixteen different relations; c.f. Figure 2.

In the following, we give an example for each relation; the governor of the relation appears in bold uppercase, the dependent in bold lowercase:

-*adjunctive*: **Vale, VAMOS!** lit. ‘Ok, let’s-go!’

-*adverbial*: **Hoy PASEO** lit. ‘Today I-go-for-a-stroll’

-*copulative*: El gato **ES negro** ‘The cat is black’

-*direct objectival*: **CONSTRUYEN una casa** lit. ‘They-build a house’

-*indirect objectival*: Les **MOLESTA** el ruido **a los peces**, lit. ‘(to-them) bothers the noise (to) the fish’, ‘The fish are bothered by the noise’

-*modificative adverbial*: **Llegados** a ese extremo, el trabajo se **VUELVE** insoportable lit. ‘Arrived-MASC-PL to that extremity, the work becomes unbearable’

-*object completive*: Pedro **CONSIDERA tontos** a los gatos lit. ‘Pedro considers stupid to the cats’

-*object copredicative*: Pedro **VE felices** a los gatos lit. lit. ‘Pedro sees happy to the cats’, ‘Pedro sees the cats happy’

-*oblique objectival*: **PASA de Pedro** lit. ‘He-ignores from Pedro’

-*quasi subjectival*: **LLUEVE(N) ranas**, lit. ‘it/they-rain(s) frogs’

-*quotative copulative*: La pregunta **ERA ‘Va a volver?’** lit. ‘The question was/ ‘Is-he-going to come-back?’

-*quotative direct objectival*: ‘Dogs’ **SIGNIFICA “perros”** (‘ “Dogs” means “perros”

-*quotative subjectival*: **“Dogs” SIGNIFICA “perros”** ‘ “Dogs” means “perros” ’

-*subjectival*: **Pedro CORRE** ‘Pedro runs’

-*subject completive*: La frase **RESULTA buena** lit. ‘The sentence turns-out fine’

-*subject copredicative*: Pedro **VUELVE feliz** lit. ‘Pedro comes-back happy’

By selecting only a few criteria, it is possible to diminish the number of relations and thus, by doing so, to tune the level of detail of the annotation. For example, keeping only four of the nine criteria presented above, we end up with only five relations, instead of sixteen:

1. Cliticization: *objectival (type 1)*

2. No Cliticization

2.1 Dep not Removable: *completive*

2.2 Removable Dep.

2.2.1 Prototypical Dep.=N

2.2.1.1 Dep. controls Agreement *subjectival*

2.2.1.2 No Agreement *objectival (type 2)*

2.2.2 Prototypical Dep.=A/Adv *adverbial*

Figure 2 summarizes the use of some criteria for Spanish and shows the correspondence between the fine-grained relations and generalized relations (rightmost side of the figure). On the left side, each intermediate node corresponds to the application of one criteria, and the leaves are the SSyntRels. The path from the root of the tree to one leaf thus indicates a list of properties of this relation. Within the brackets, some properties are listed which are entailed by the criterion they appear next to. For example, the *Canonical Order* (CO Right/Left) can always be predicted by a particular property: for instance, all elements that can be cliticized are usually linearized on the right of their governor. If *Canonical Order* is not mentioned for a relation, it is because there is no canonical order, as it is the case for three adverbial relations (*modificative adverbial*, *adjunct*, and *adverbial*). Obviously, every relation

usually has many more properties than those listed in this hierarchy.

Although we use only syntax-based criteria, it is possible to reach the semantic level by indicating whether the dependent of a relation is accounted for in the valency of its governor (no (-), actant I, actant II, etc.), which is indicated by the numbers in the column to the right of SSYNTRELS.⁶ This helps for generalizing the relations, as illustrated on the right side of the figure. This second hierarchy, over relations, is similar to those proposed by, among others, (De Marneffe et al, 2006) or (Mille and Wanner, 2010).

5 Conclusions

Even if there are dependency corpora in different languages and some of them widely used for NLP applications, it is not yet clear how the set of syntactic relations can be obtained and how it can be organized so as to allow for different levels of granularity in the annotation. In this paper, we attempt to fill this gap by detailing the procedure for establishing a tagset for Spanish verbal relations. We present a restricted selection of easy-to-use criteria which facilitate the work of the annotators, but which also can accommodate for the elaboration of a more or less fine-grained tagset. An advantage of such hierarchical schema is its potential application to any other language, although it is possible that some criteria are not needed anymore for a specific language (e.g., linearization for order-free languages) or, on the contrary, that new syntactic criteria are needed. We already successfully began to apply this method to a radically different language, namely, Finnish, and are annotating a 2,000 sentence corpus with a restricted set of about 25 relations.

The use of the fine-grained tagset and the application of the hierarchized criteria for the annotation of a 100,000 word corpus has proven feasible.

References

- S. Afonso, E. Bick, R. Haber, and D. Santos. 2002. Floresta sintá(c)tica: A treebank for Portuguese. In *Proceedings of LREC 2002*, Las Palmas de Gran Canaria, Spain.

⁶We actually have a version of our corpus with such valency information (to be released).

- STTS. Technical report, Institute for Natural Language Processing, University of Stuttgart.
- L. Van der Beek, G. Bouma, R. Malouf, and G. Van Noord. 2002. The Alpino Dependency Treebank. In *Proceedings of Computational Linguistics in the Netherlands CLIN 2001*.
- M. Čmejrek, J. Hajič, and V. Kuboň. 2004. Prague Czech-English Dependency Treebank: Syntactically Annotated Resources for Machine Translation. In *Proceedings of the fourth international conference on Language Resources and Evaluation (LREC2004)*, Lisbon, Portugal.

A Dependency-based Analysis of Treebank Annotation Errors

Katri Haverinen,^{1,3} Filip Ginter,³ Veronika Laippala,² Samuel Kohonen,³
Timo Viljanen,³ Jenna Nyblom³ and Tapio Salakoski^{1,3}

¹Turku Centre for Computer Science (TUUS)

²Department of French studies

³Department of Information Technology

20014 University of Turku, Finland

first.last@utu.fi

Abstract

In this paper, we investigate errors in syntax annotation with the Turku Dependency Treebank, a recently published treebank of Finnish, as study material. This treebank uses the Stanford Dependency scheme as its syntax representation, and its published data contains all data created in the full double annotation as well as timing information, both of which are necessary for this study.

First, we examine which syntactic structures are the most error-prone for human annotators, and compare these results to those of a baseline automatic parser. We find that annotation decisions involving highly semantic distinctions, as well as certain morphological ambiguities, are especially difficult for both human annotators and the parser. Second, we train an automatic system that orders for inspection sentences ordered by their likelihood of containing errors. We find that the system achieves a performance that is clearly superior to the random baseline: for instance, by inspecting 10% of all sentences ordered by our system, it is possible to weed out 25% of errors.

1 Introduction

In the field of natural language processing (NLP), human-annotated training data is of crucial importance, regardless of the specific task. The creation of this data requires a large amount of resources, and the data quality affects applications. Thus it is important to ensure that first, the quality of the data is as sufficiently high for the desired purpose,

and second, that the amount of expensive manual work is kept to a reasonable amount. Considering the importance of manual annotation for NLP, studies on different aspects of the annotation process are of great interest.

This work strives to examine the difficulty of syntax annotation in the context of Finnish. Our primary objective is to study human annotation and the errors in it, so as to make observations beneficial for future treebanking efforts. As dependency representations have been argued to be a good choice for the purposes of evaluating the correctness of an analysis as well as the general intuitiveness of evaluation measures (see, for instance, the work of Lin (1998) and Clegg and Shepherd (2007)), and as there exists a recently published, dependency-based treebank for Finnish, also this study uses dependency-based evaluation.

Our experiments are twofold. First, we conduct an experiment to find which phenomena and constructions are especially error-prone for human annotators. We also compare human errors to those of an automatic baseline parser. Second, as a practical contribution, we build an automatic system that orders annotated sentences in such a way that those sentences most likely to contain errors are presented for inspection first.

The difficulty of annotation is not a heavily studied subject, but there has been some previous work. For instance, Tomanek and Hahn (2010) have studied the difficulty of annotating named entities by measuring annotation time. They found that cost per annotated unit is not uniform, and thus suggested that this finding could be used to improve models for active learning (Cohn et al., 1996), the goal of which is to select for annotation those examples that are expected to most benefit an existing machine learning system. Tomanek et



Figure 1: The Stanford Dependency scheme. The sentence can be translated as *The commission must ask for clarification from the minister and his assistant.*

al. (2010) have conducted a follow-up study using eye-tracking data, and found that annotation time and accuracy depend on both the syntactic and semantic complexity of the annotation unit.

Dligach et al. (2010) have studied annotation costs in the context of word sense disambiguation and concluded that for data annotated solely for machine learning purposes, single-annotating a large amount of data appears to be preferable over double-annotating a smaller amount of data. On the level of discourse annotation, Zikánová et al. (2010) have examined typical disagreements between annotators in the context of discourse connectives and their scopes, and on the level of syntax, Dickinson (2010) has studied the possibilities of finding errors in automatic parses in the context of producing *parsebanks*.

However, studies in the context of manual syntax annotation in particular have been rare. One reason for this may be that data which would enable such studies is not generally available. Many treebanks, such as the well-known Penn Treebank (Marcus et al., 1993), are single-annotated, after an initial annotator training period, and thus agreement of the annotators cannot be measured across the whole treebank. Also, timing data for the annotation process is usually not recorded and made available.

2 Data: The Turku Dependency Treebank

In our experiments, we use the first Finnish treebank, the Turku Dependency Treebank (TDT) by Haverinen et al. (2010). TDT is a treebanking effort still in progress, and the new version used in this work is a superset of the recent second release of the treebank and consists of 7,076 sentences (100,073 tokens). Approximately 10%¹ of this data is not used in our experiments, except for parser parameter optimization as described below, and this portion of the data will be held secret for the purpose of possible future parser com-

parisons and scientific challenges. The remaining 90% of TDT, the portion that was used in this work, consists of 6,375 sentences (89,766 tokens). This data will be made available at the address <http://bionlp.utu.fi/>.

The annotation scheme of the treebank is a slightly modified version of the well-known Stanford Dependency (SD) scheme (de Marneffe and Manning, 2008a; de Marneffe and Manning, 2008b). The annotation in TDT is based on the *basic* variant of the scheme, in which the analyses are trees of dependencies. In total, the scheme version of Haverinen et al. contains 45 different dependency types, whereas the original scheme version contains 54 types. The scheme modifications include both omissions of types where the corresponding phenomenon does not occur in Finnish, and additions where a phenomenon has not been accounted for in the original SD scheme. Figure 1 illustrates the usage of the SD scheme on a Finnish sentence. In this paper, we only discuss those aspects of the SD scheme that are relevant for the current study. For further details of the scheme, we refer the reader to the annotation manual by de Marneffe and Manning (2008a), and for changes made during the annotation process of TDT, the paper by Haverinen et al. (2009).

The Turku Dependency Treebank is exceptional in the sense that the whole treebank has been created using *full double annotation*, where each sentence is first independently annotated by two different annotators, and all differences are then jointly resolved. This results in a single analysis that is called the *merged* annotation. Afterward, the treebank data is subjected to consistency checks, the purpose of which is to ensure that the final release of the treebank, called the *final* annotation, consists of analyses that are updated to conform to the newest annotation decisions. Consistency checks are needed, as some decisions may need revision when the annotation team comes across new examples, and thus the annotation scheme undergoes slight changes.

The treebank also contains the morphologi-

¹10% on the level of full text documents

cal analyses of two Finnish morphology tools by Lingsoft Ltd., FinTWOL and FinCG (Koskeniemi, 1983; Karlsson, 1990).² Out of these, FinTWOL gives each token all of its possible morphological readings, and FinCG disambiguates between these readings. When unable to fully disambiguate, FinCG can select multiple readings.

In addition to the actual treebank — the *final* annotations — TDT releases contain the *individual* annotations of each annotator, two per sentence, and the *merged* annotations. In addition, the documents include a full edit history with millisecond-resolution timestamps.

In total five different annotators have taken part in the annotation of TDT. The annotators have backgrounds including PhD and Master’s students in computer science and linguistics, and also their prior knowledge of linguistics varies substantially.

Our experiments have been conducted against the *merged* annotations, not the *final* annotations of the treebank. This is because we want to avoid penalizing an annotator for a decision that was correct at annotation time but has later become outdated. In addition, the numbers of tokens and sentences in the individually annotated documents and in the final treebank documents do not necessarily match, as possible sentence splitting and tokenization issues are corrected at the consistency fix stage of the annotation process. The only exception to this strategy of comparing *individual* annotations against the *merged* annotation is the experiment detailed in Section 4, where an annotator re-annotated some of the treebank sentences, to estimate the quality of the *final* annotation.

For experiments where a baseline parser was needed, we used the MaltParser³ (Nivre et al., 2007). Of the treebank documents, 10% were used for parameter optimization and excluded from the experiments. The remaining portion of the treebank was parsed using *ten-fold crossvalidation*, meaning that 90% of the data was used to train a parser and the remaining 10% was then parsed with it, and this process was repeated ten times in order to parse the whole data (disregarding the parameter optimization set) while ensuring that the training and testing data do not overlap.

²<http://www.lingsoft.fi>

³<http://www.maltparser.org/>

3 Error-prone constructions

As the first part of our study, we have examined the numbers of different errors by the human annotators as well as the baseline parser. In these experiments, all dependencies that remain unmatched between the *merged* annotation (henceforth discussed as *gold standard*, *GS*) and the *individual* annotation (human or automatic), are considered errors. In our measurements, we have used the standard F_1 -score, defined as $F_1 = \frac{2PR}{P+R}$, where P stands for precision and R stands for recall. Precision, in turn, is the proportion of correctly annotated dependencies out of all dependencies present in the *individual* annotation, and recall is the proportion of correctly annotated dependencies out of all dependencies present in the gold standard. In some experiments, we also use the *labeled attachment score* (*LAS*), the proportion of tokens with the correct governor and dependency type.

In addition to the measurements described below, we have also studied the overall annotator performance on the different *sections*⁴ of TDT, in order to find how genre affects the agreement. However, the differences found were small, and it appears that the annotator performance on different genres is similar to the overall performance.

3.1 Most difficult dependency types

In our first set of measurements, we examined which dependency types were the most difficult for the human annotators and the baseline parser. This was done by calculating an F_1 -score for each of the dependency types, and the types with the lowest F_1 -scores were considered the most difficult ones. Only those dependency types that occur in the gold standard at least 150 times were considered, in order to avoid taking into account types that may have extremely low F_1 -scores, but which are also very rare, meaning that their being incorrect hardly affects the overall treebank at all. Table 1 shows the ten most difficult types for the annotators, as well as for the baseline parser.⁵

From this table it can be seen that several of the most difficult dependency types for human annotators represent a complement of the verb. The annotation scheme of the treebank contains sev-

⁴Current sections include Wikipedia and Wikinews texts, articles from a university online magazine and from student-magazines, blog entries, EU text and grammar examples.

⁵In this experiment, we have disregarded the small single-annotated proportion of TDT constructed in the very beginning of the annotation process in so called *trial annotations*.

Human					Parser				
type	P	R	F	freq.	type	P	R	F	freq.
icomp	68.8	70.9	69.8	261 (0.3%)	parataxis	24.2	8.2	12.3	280 (0.4%)
parataxis	69.9	71.6	70.7	280 (0.4%)	advcl	34.9	39.2	36.9	982 (1.3%)
acomp	74.1	70.5	72.2	154 (0.2%)	appos	41.3	38.5	39.8	658 (0.9%)
compar	77.0	71.6	74.2	178 (0.2%)	compar	62.4	35.3	45.2	178 (0.2%)
dep	85.8	69.4	76.7	291 (0.4%)	acomp	53.2	43.5	47.9	154 (0.2%)
advcl	79.2	79.1	79.2	982 (1.3%)	rcmod	49.7	48.2	49.0	897 (1.2%)
auxpass	84.9	75.7	80.0	282 (0.4%)	ccomp	57.2	49.2	52.9	835 (1.1%)
ccomp	82.2	79.4	80.8	835 (1.1%)	icomp	64.4	47.9	54.9	261 (0.3%)
appos	81.7	80.2	81.0	658 (0.9%)	name	50.7	68.0	58.1	1,925 (2.5%)
gobj	88.6	77.4	82.6	579 (0.8%)	conj	61.7	62.9	62.3	4,041 (5.3%)
overall	89.9	89.1	89.5	76,693 (100%)	overall	71.4	70.2	70.8	76,693 (100%)

Table 1: The ten hardest dependency types for the human annotators and the parser. The standard F_1 -score was calculated for each dependency type separately, considering only those types that occur in the gold standard at least 150 times. This table presents the ten dependency types with the lowest F_1 -scores. For each type is given its precision, recall and F_1 -score, and its frequency in the gold standard.

eral different types for these complements, such as clausal complement (*ccomp*) and infinite clausal complement (*icomp*), as well as a clausal complement with external subject (*xcomp*). Distinguishing these types, especially *ccomp* and *icomp*, is often challenging, as the distinction depends on only the form of the complement verb. Adjectival complements (*acomp*) likely fall victim to the difficulty of assessing whether a sentence element is a complement. The attachment of sentence elements can also be a source of difficulty. For instance, in word order variations of an example like *The man in the brown coat came into the train* it may be difficult to determine whether *in the brown coat* should modify *man* or *came into the train*. In these cases, the analysis in the treebank follows rules similar to those used in the Prague Dependency Treebank (Hajič, 1998), where in *The man in the brown coat came into the train* there is considered to be a *man in the brown coat*, but in *The man came into the train in a brown coat* the coming into the train happened while wearing a brown coat. These rules, however, are easy to overlook especially in fast-paced annotation. Adverbial clause modifiers (*advcl*), non-complement subordinate clauses, are an example of a phenomenon where the difficulty of annotation may be partly due to attachment issues and partly the difficulty of distinguishing complements and modifiers.

The dependency type *parataxis* is used to mark two different phenomena: direct speech and certain types of implicit clausal coordination, for instance, clauses combined using a semicolon. Especially the latter use can be difficult due to the phenomenon being closely related to coordination. Comparative structures (marked with the

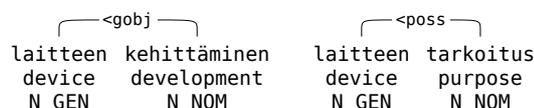


Figure 2: Genitive objects (left) and other genitive modifiers (right). The examples can be translated as *the development of the device* and *the purpose of the device*, respectively. The word *laitteen* (genitive form of *device*) is a genitive attribute of the noun in both examples, but on the left, the noun *kehittäminen* has been derived from the corresponding verb *kehittää* (*to develop*), and the device acts as the object of the developing. Direct derivations of a verb are morphologically marked in the treebank, but other verb-related nouns are not.

type *compar*), in turn, are often elliptical, and it may be unclear what is being compared with what.

Passive auxiliaries (*auxpass*) may suffer from the annotator simply forgetting them, as there is also a more general dependency type for auxiliaries (*aux*). In some cases drawing the line between passives and other subjectless expressions⁶ may be difficult. In addition, some passive participles can also be interpreted as adjectives, and thus clauses containing these participles can be read as copular. Another mistake that is easily made out of carelessness is that of mistaking genitive objects (*gobj*) for more general genitive modifiers (*poss*). On the other hand, the distinction of genitive objects and general genitive modifiers is also highly semantic in nature. For an illustration of genitive objects in the SD scheme, see Figure 2.

⁶such as the *zeroth person*, *nollapersoon* (Hakulinen et al., 2004, §1347)

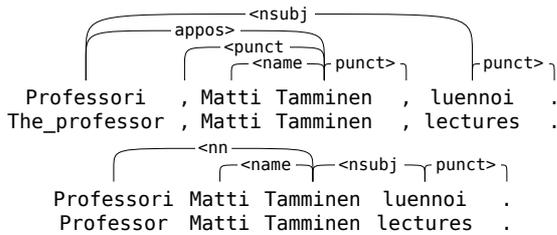


Figure 3: Appositions (top) and appellation modifiers (bottom). The examples can be translated as *The professor, Matti Tamminen, lectures* and *Professor Matti Tamminen lectures*, respectively. The key difference between the examples is that the apposition structure includes commas, while the one with the appellation modifier does not.

Another difficult phenomenon seen in Table 1 is the apposition (*appos*). Appositions are often hard to distinguish from nominal modifiers (*nommod*) due to the semantic requirement that an apposition should have the same referent as its head word. In addition, the annotation scheme distinguishes between appositions and appellation modifiers (marked with the type *nn* alongside with noun compound modifiers), where the distinction usually depends on small details such as the inflection forms of the words involved or the presence or absence of punctuation. Figure 3 illustrates appositions and appellation modifiers in the Finnish-specific version of the SD scheme. Finally, the most generic dependency type *dep* (dependent) is also among the most difficult types. This type is meant for cases where no other, more specific type applies, and in the treebank, it is mostly used for idiomatic two-word expressions.

The most difficult dependency types for the automatic parser are in some respects similar compared to humans, although there are differences as well. Like human annotators, the parser had difficulties with different clausal complements and modifiers (types *ccomp*, *advcl* and *icomp*), and unlike humans, it also scored low on relative clause modifiers (*rmod*). Appositions were also clearly difficult for the parser, which is understandable due to the semantic distinctions involved. Another two types that were difficult for the parser but not particularly for humans, were *conj* (coordinated element, see Figure 1) and *name*. With coordinations, it is difficult for a parser to decide which sentence element is coordinated with which, and additionally, for instance an apposition structure may seem coordination-like

without any semantic information. The closely related *parataxis* was also especially difficult for the parser. The low F_1 -score of the type *name*, which is used for multi-word named entities, has to do, at least partly, with lack of morphological information. Many of the words that are marked with *name* in the training material are unknown to the morphological analyzer, and thus the parser is eager to mark unknown words as multi-word named entities. The overall F_1 -score of the parser is 70.8%, compared to the overall human performance of 89.5%.

3.2 Dependency type confusions

Seeing that for many of the most difficult dependency types, the potential explanation seemed to include a possible confusion with another type, we have investigated this matter further. We have calculated the numbers of those errors where the governor is correct, but where the dependency type is wrong, that is, where a dependency type has been replaced by another type. Table 2 shows the five most common type confusions for all five annotators as well as the parser. In total, approximately 32.4% of all erroneous dependencies assigned by annotators only had an incorrect dependency type.

The confusion errors can be divided into several different classes. One error type that can be seen from the table are errors arising from both morphological and semantic closeness of two phenomena. For instance, a common type confusion for nearly all annotators was that of confusing the types *nommod* (*nominal modifier*) and *dobj* (*direct object*). The distinction between nominal modifiers and direct objects is based on both structure and morphology; objects are complements of the verb that can only take certain cases of the Finnish case system (Hakulinen et al., 2004, §925). It is likely that the semantic closeness of objects and certain nominal modifiers misled annotators. In addition, some measures of amount take the same cases as objects and closely resemble them. A nominal modifier like this is called an *object-cased amount adverbial*⁷ (Hakulinen et al., 2004, §972).

Also a second confusion seemed to be affected by morphological and semantic closeness. This confusion occurred particularly for Annotators 2 and 4, who notably confused subjects and objects on occasion. For other annotators this confusion occurred as well, but not as frequently. Subjects

⁷*objektin sijainen määrän adverbiaali (OSMA)*

Annotator 1			Annotator 2			Annotator 3		
GS type	annot. type	fr. (%)	GS type	annot. type	fr. (%)	GS type	annot. type	fr. (%)
advmod	nommod	5.6	dobj	nommod	6.8	advmod	nommod	6.8
dobj	nommod	3.7	gobj	poss	5.5	dobj	nommod	5.7
auxpass	aux	3.0	nsubj	dobj	4.8	nommod	dobj	4.2
gobj	poss	2.9	advmod	nommod	4.4	advmod	advcl	3.0
nommod	advmod	2.6	nommod	dobj	4.3	nommod	appos	3.0
Annotator 4			Annotator 5			Parser		
GS type	annot. type	fr. (%)	GS type	annot. type	fr. (%)	GS type	annot. type	fr. (%)
dobj	nommod	11.5	dobj	nommod	7.1	nommod	dobj	5.5
nommod	dobj	6.0	acomp	nommod	7.1	gobj	poss	5.4
gobj	poss	5.4	partmod	advcl	5.4	partmod	amod	4.8
nsubj	nommod	5.1	appos	conj	5.4	nsubj	dobj	4.1
nsubj	dobj	4.0	nommod	dobj	5.4	dobj	nommod	4.0

Table 2: The five most common dependency type confusions for each annotator and the parser. For each confusion is given the gold standard dependency type (GS type) and the type suggested by the annotator (annot. type), as well as the frequency of the confusion, out of all type confusions by the annotator/parser.

and objects may at first seem like a surprising confusion pair, but actually, due to several reasons these two can rather easily be confused in Finnish, especially when annotating quickly. First, both subject and object use the same cases of the Finnish case system: the nominative, the partitive, and the genitive. Second, Finnish is a free word-order language, and thus the word-order does not necessarily reveal the role of a word. Also, certain verbs that are passive-like in nature, but in fact take a subject and not an object, so called *derived passives*⁸ (Hakulinen et al., 2004, §1344), further add to the misleading characters of subjects and objects. In the majority of cases, it is not difficult to decide which of the two analyses is correct in the annotation scheme, once the disagreement is brought into attention, but rather it is the case that annotators are easily misled by the similar properties of these two sentence elements.

A second error type seen in the table is a confusion that is based on a difficult morphological distinction. The distinction between nominal (*nommod*) and adverbial modifiers (*advmod*) was, for several annotators, among the most difficult ones. It is not always clear whether a word should be analyzed as an adverb or rather as an inflected noun, as it is possible for many adverbs to inflect in certain cases, similarly to nouns. For instance, the Finnish word *pääasiassa* (*mainly*) could be analyzed as an adverb, or it could be seen as an inflected form of the noun *pääasia* (*main thing*).

One unexpected type of confusion errors was typical for Annotator 3 in particular. These errors are not due to linguistic similarity, but are simply typographical errors. The annotator has confused

adverb modifiers (*advmod*) with adverbial clause modifiers (*advcl*), which are linguistically rather easily distinguishable, but in the annotation software user interface, the shortcut key for *advmod* is capital *V*, while the shortcut key for *advcl* is non-capital *v*. Similarly, this annotator has confused also other dependency types where the respective shortcut keys were capital and non-capital versions of the same letter, but these were not as frequent. Annotator 1 also used the shortcut keys of the annotation user interface and made some typographical errors, although not frequently enough to appear among the five most common type confusions. An example of such an error by Annotator 1 is the confusion of subjects (*nsubj*) and adjectival modifiers (*amod*). The explanation for this otherwise peculiar error is that the shortcut key for *nsubj* is *s* and the one for *amod* is *a*, which are adjacent on a regular Finnish keyboard.

The automatic parser also displayed confusion errors in its output (approximately 16.3% of all erroneous dependencies), involving many of the same semantic distinctions that were difficult for humans, such as genitive objects versus other genitive modifiers and nominal modifiers versus direct objects. Notably the confusion of subjects and objects was also present. Also one morphological distinction was among the most difficult ones for the parser: participial versus adjectival modifiers, where the distinction is, in fact, between participles and adjectives. The same confusion was present for human annotators, but not among the five most common ones. As an example, consider the Finnish word *tunnettu* (*well-known*). It could be a form of the verb *tuntea* (*to know*), but on the other hand, it can be given the comparative and

⁸*johdospassiivi*

superlative forms, which are typical of adjectives. The only type of confusions that did not, naturally, occur for the parser were the typographical errors.

3.3 Correlation of human and parser errors

An interesting question to study is whether the annotator and parser errors correlate with respect to their position in the sentence. Such correlation would indicate that certain structures are in some sense “universally difficult”, regardless of whether the annotator is human or machine. This correlation is easy to analyze on the level of tokens: a token is deemed correct if its governor and dependency type are correct. Since we have two independent human annotations for each sentence, we take the union of the individual annotators’ errors, thus defining a token as correct only if it was correctly analyzed by both of the annotators. In this experiment, we can only take a sentence into account, if it has both human analyses available. This is the case for a total of 82,244 tokens not used for parser optimization, as a small portion of TDT has, in the very beginning of the annotation process, been constructed in so called *trial annotations*, where a single annotator has annotated the sentence and it has then been jointly inspected (Haverinen et al., 2009).

The results are shown in Table 3. We find that 35.9% (8,677/24,152) of parser errors co-occur with human errors, whereas only a 18.9% (15,548/82,244) co-occurrence, corresponding to the human error-rate, would be expected by chance. Similarly, we find that 55.8% (8,677/15,548) of human errors co-occur with parser errors, whereas only a 29.3% (24,152/82,244) co-occurrence, corresponding to the parser error-rate, would be expected by chance. We can thus conclude that there is a notable positive association between human and parser errors, strongly statistically significant with $p \ll 0.001$ (Pearson’s chi-square test on Table 3).

		human	
		error	correct
parser	error	8,677	15,475
	correct	6,871	51,221

Table 3: Token-level correlation between human and parser errors.

4 Correctness of double-annotated data

As part of our investigation on the number of errors by human annotators, we have conducted a small-scale experiment on the correctness of the final treebank annotation. We sampled a random set of 100 sentences from the *final* annotations of the treebank and assigned them to an annotator who had not annotated them previously. This annotator then independently re-annotated these sentences, and the resulting annotation was compared to the previously existing *final* annotation in a regular meeting between all the annotators.

Effectively, we thus gained a set of triple-annotated sentences. The *final* annotation of the corresponding portion of the treebank was compared against these triple-annotated sentences, and thus we gained an estimate of the error-rate of the *final* annotation in the treebank. The LAS for the final treebank annotation against the triple-annotated sample as gold standard was 98.1%, which means that the minimum error-rate of the *final* annotation is 1.9%. This is a lower bound, as it is possible (although unlikely) that further errors go unnoticed because three annotators have given a sentence the same, erroneous analysis.

We thus find that *full double annotation* is an efficient way to produce annotation of high quality. The triple annotation agreement of 98.1% together with the original inter-annotator agreement of 89.6% in LAS (89.5% in F_1 – score) implies that approximately 82% $((98.1-89.6)/(100-89.6))$ of errors remaining in the single annotated documents can be weeded out using double annotation.

5 Automated recognition of annotation errors

While full double annotation produces high-quality results, as shown in the previous section, it is undoubtedly a resource-intensive approach to annotation. In many cases, particularly when building large treebanks, a compromise between single and double annotation will be necessary. Under such a compromise annotation strategy, only some proportion of sentences would be double annotated or otherwise carefully inspected for errors, while the rest would remain single-annotated. If we were to select these sentences randomly, we would expect to correct the same proportion of annotation errors present in the treebank, assuming that the errors are approximately evenly distributed throughout the treebank. Thus,

for example, by randomly selecting 25% of the sentences for double annotation we would expect to visit 25% of annotation errors present in the treebank. The necessary effort would naturally decrease if we used a strategy that is better than random at selecting sentences which contain annotation errors. In the following, we investigate a machine-learning based method which, given a single-annotated sentence, assigns each token a score that reflects the likelihood of that token being an annotation error, i.e., not having the correct governor and dependency type in the tree.

We approach the problem as a supervised binary classification task where incorrectly annotated tokens are the *positive class* and correctly annotated tokens are the *negative class*. Training data for the classifier can be obtained from the individual annotators' trees, by comparing them against the *merged* trees resulting from the double annotation protocol. If for any token its governor or dependency type do not match those in the merged tree, this token is considered an annotation error (a positive instance), otherwise it is considered correct (a negative instance). Since the average LAS of our annotators is about 90%, the training data contains about 10% positive instances and 90% negative instances, a considerably disbalanced distribution.

The features that represent the tokens in classification are as follows:

Annotator The annotator who produced the tree.

Morphology/POS The lemma, POS, and morphological tags given for all possible morphological readings of the word (prefixed by "cg_" if the reading was selected by the FinCG tagger). The number of possible morphological readings of the word, and the number of readings selected by FinCG.

Dependency Whether the token acts as a dependent, the dependency type, and all morphology/POS features of the governor, given both separately and in combination with the dependency type. The same features are also generated for all dependents of the token under consideration.

We split the available data randomly into a training set (50%), a parameter estimation set (25%), and a test set (25%). The split is performed on the level of documents, so that all instances generated from both annotations of a single document are always placed together in one of the three sets. This

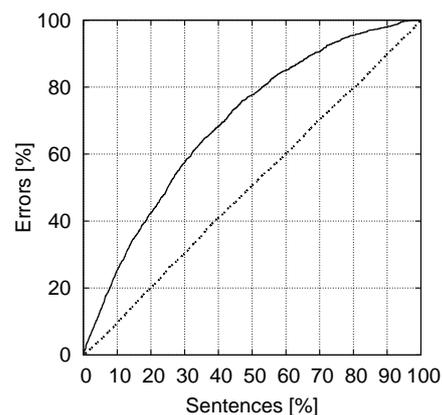


Figure 4: Proportion of annotation errors recovered. The full line represents the machine-learning based ordering of sentences, while the dashed line represents a baseline obtained by ordering the same sentences randomly.

prevents any possible leak of information between data used for training and that used for testing. As the classifier, we use the support vector machine (SVM)⁹ (Joachims and Yu, 2009) with the radial basis kernel. We select the C and γ parameters by a wide grid search on the parameter estimation set. To account for the pronounced disbalance in the positive and negative class distribution, we use the standard *area under ROC curve* (AUC) performance measure, which is not sensitive to class distribution, and is thus preferred in this case over the usual F_1 or accuracy. We use AUC as both the SVM loss function and the performance criterion to select the best parameter combination.

To evaluate the accuracy of the classification, and its practical impact on annotation, we first calculate for each sentence the maximum of the classification scores over all of its tokens and then order the sentences in descending order by this value. The results on the test set are shown in Figure 4. The classifier is notably better than the random baseline: the first 10% of the sentences contain 25% of all annotation errors, and the first 25% of the sentences contain 50% of all annotation errors. These differences are large enough to provide a notable decrease in annotation effort. For instance, the effort to correct 50% of annotation errors is halved: only 25% of all sentences need to be double-annotated, instead of the 50% random baseline. For a treebank of 10,000 sentences, this

⁹Implemented in the *SVM^{perf}* package available at <http://www.joachims.org>

would mean that 2,500 sentences less would need to be double annotated, a notable reduction of effort. Here it should be noted that the classification problem is relatively difficult: we are asking a classifier to recognize human mistakes, at a task at which the humans are 90% correct to start with.

We have also investigated, as an additional feature for the classification, the time spent by the annotator to insert all dependencies for the given token (governor and all dependents), including dependencies that are removed or relabeled in the course of the annotation. Our hypothesis was that those parts of the sentence on which the annotator spent an unusually long time are more difficult to analyze, and thus prone to error as well. This experiment is possible since the treebank contains annotation history data with millisecond-resolution timestamps. However, a substantial part of the treebank is annotated so that of the two individual annotations for each sentence, one is constructed from scratch with all dependencies inserted manually, while the other is constructed on top of the output of a parser, with the annotator correcting the parser output (Haverinen et al., 2010). Complete timing data can naturally be extracted only in the former case, amounting to 119,117 tokens.

We further normalize the annotation timing data to account for the different baseline annotation speeds of the annotators, as well as for the simple fact that the annotation of a token with more dependencies takes longer to complete. We first divide the annotation time of each token by the number of its dependencies in the completed tree and then, for each sentence separately, subtract from each time the mean and divide by standard deviation of the times in that particular sentence. Thus normalized annotation times were then included as a feature in the classification. However, no measurable gain in the performance of the classifier could be observed.

To investigate the correlation between annotation speed and annotation accuracy further, we define a token as “slow” if the time it took to complete is more than one standard deviation above the mean¹⁰ time in the given sentence (we first divide by the number of the token’s dependencies, as previously). We then correlate the correctness and speed of annotation in a contingency

¹⁰Variations of this definition were tested and had no effect on the overall conclusion.

	correct	incorrect
slow	14,752	2,288
normal	92,290	9,787

Table 4: Correlation between annotation speed and correctness of tokens. Tokens are defined as “slow” if their annotation took longer than one standard deviation above the mean time.

table (Table 4). We find that incorrectly annotated tokens are overrepresented among “slow” tokens (13.4%), compared to the rest of the tokens (9.6%), as per our original hypothesis. This positive association is strongly statistically significant ($p \ll 0.001$, Pearson’s chi-square test on Table 4). While this observation is of some interest, the magnitude of the difference is likely too small for practical applications and annotation times do not seem to provide new information — on top of the features listed above — to a classifier predicting incorrectly annotated tokens.

6 Conclusions and future work

In this paper, we have studied the difficulty of syntax annotation in a dependency-based framework, in the context of the Finnish language and the Stanford Dependency (SD) scheme. We have studied the different kinds of errors by the annotators and compared these errors with those of a baseline parser. In addition, we have trained an automatic system that orders single-annotated sentences so that sentences that are most likely to contain errors are offered for inspection first.

We find that there are several different kinds of mistakes that humans make in syntax annotation. In this data, different kinds of clausal complements and modifiers were often erroneously marked, as were comparatives, appositions and structures with parataxis. Nearly one third of the erroneous dependencies marked by annotators were such that only the type of the dependency was wrong. Morphological and semantic closeness of two phenomena seemed to mislead annotators, as for instance adverbial modifiers were often confused with nominal modifiers, and nominal modifiers with direct objects. Annotators also made some mistakes that were not due to any linguistic resemblance, but rather an artifact of annotation user interface shortcut keys that were adjacent or capital and non-capital versions of the same letter. The last type of errors suggests how

this particular annotation user interface in question could be improved, or how the usability of possible future software could be increased.

We also find that our automatic sentence ranker notably outperforms a random baseline. This means that using this classifier to order single annotated sentences for inspection, it is possible to significantly reduce the amount of double annotation or other careful inspection needed in a compromise setting where full double annotation is not possible or desired. For instance, if one wanted to correct 50% of errors in a treebank, using the proposed method, they could inspect only 25% of all sentences instead of the 50% expected by random selection — a remarkable decrease in effort.

In the future, the knowledge gained in this work could be used for developing new methods helpful for inspecting manual annotations, and for the benefit of large annotation efforts in general. Also studies in for instance the field of active learning, where the goal is to keep the amount of data annotated for machine learning purposes to a minimum, could be conducted.

Acknowledgments

We would like to thank Lingsoft Ltd. for their kind permission to use FinTWOL and FinCG analyses in TDT. We are also grateful to the corpus text authors for the use of their text. This work has been supported by the Academy of Finland.

References

- A. B. Clegg and A. Shepherd. 2007. Benchmarking natural-language parsers for biological applications using dependency graphs. *BMC Bioinformatics*, 8(1):24.
- D. Cohn, Z. Ghahramani, and M. Jordan. 1996. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145.
- M. Dickinson. 2010. Detecting errors in automatically-parsed dependency relations. In *Proceedings of ACL'10*, pages 729–738.
- D. Dligach, R. Nielsen, and M. Palmer. 2010. To annotate more accurately or to annotate more. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 64–72.
- J. Hajič. 1998. Building a Syntactically Annotated Corpus: The Prague Dependency Treebank. In *Issues of Valency and Meaning. Studies in Honour of Jarmila Panevová*, pages 106–132. Karolinum, Charles University Press, Prague, Czech Republic.
- A. Hakulinen, M. Vilkuna, R. Korhonen, V. Koivisto, T-R. Heinonen, and I. Alho. 2004. *Iso suomen kielioppi / Grammar of Finnish*. Suomalaisen kirjallisuuden seura.
- K. Haverinen, F. Ginter, V. Laippala, T. Viljanen, and T. Salakoski. 2009. Dependency annotation of Wikipedia: First steps towards a Finnish treebank. In *Proceedings of TLT8*, pages 95–105.
- K. Haverinen, T. Viljanen, V. Laippala, S. Kohonen, F. Ginter, and T. Salakoski. 2010. Treebanking finnish. In *Proceedings of TLT9*, pages 79–90.
- T. Joachims and C-N. John Yu. 2009. Sparse kernel SVMs via cutting-plane training. *Machine Learning, Special Issue from ECML PKDD 2009*, 76(2–3):179–193.
- F. Karlsson. 1990. Constraint Grammar as a framework for parsing unrestricted text. In *Proceedings of COLING'90*, pages 168–173.
- K. Koskenniemi. 1983. Two-level model for morphological analysis. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, pages 683–685.
- D. Lin. 1998. A dependency-based method for evaluating broad-coverage parsers. *Natural Language Engineering*, 4(2):97–114.
- M. Marcus, M. A. Marcinkiewicz, and B. Santorini. 1993. Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19(2):313–330.
- M-C. de Marneffe and C. Manning. 2008a. Stanford typed dependencies manual. Technical report, Stanford University, September.
- M-C. de Marneffe and C. Manning. 2008b. Stanford typed dependencies representation. In *Proceedings of COLING'08, Workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8.
- J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryğiç, S. Kübler, S. Marinov, and E. Marsi. 2007. Malt-Parser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.
- K. Tomanek and U. Hahn. 2010. Annotation time stamps — temporal metadata from the linguistic annotation process. In *Proceedings of LREC'10*, pages 2516–2521.
- K. Tomanek, U. Hahn, S. Lohmann, and J. Ziegler. 2010. A cognitive cost model of annotations based on eye-tracking data. In *Proceedings of ACL'10*, pages 1158–1167.
- Š. Zikánová, L. Mladová, J. Mírovský, and P. Jínová. 2010. Typical cases of annotators disagreement in discourse annotations in prague dependency treebank. In *Proceedings of LREC'10*, pages 2002–2006.

The Copenhagen Dependency Treebank (CDT)

Extending syntactic annotation to morphology and semantics

Henrik Høeg Müller
Copenhagen Business School
Copenhagen, Denmark
E-mail: hhm.isv@cbs.dk

Abstract

This paper has two main objectives. The first is to provide an overview of the CDT annotation design with special emphasis on the modeling of the interface between syntactic and morphological structure. Against this background, the second objective is to explain the basic fundamentals of how CDT is marked-up with semantic relations in accordance with the dependency principles governing the annotation on the other levels of CDT. Specifically, focus will be on how Generative Lexicon theory has been incorporated into the unitary theoretical dependency framework of CDT by developing an annotation scheme for lexical semantics which is able to account for the lexico-semantic structure of complex NPs.

1. Introduction

The Copenhagen Dependency Treebank (CDT)¹ is a set of parallel text collections (treebanks) of approx. 60.000 words each for Danish, English, German, Italian and Spanish with a unified annotation of morphology, syntax and discourse, as well as an alignment system of translational equivalences (Kromann, 2003; Buch-Kromann et al., 2009). The treebanks are annotated on the basis of the dependency-based grammar formalism Discontinuous Grammar (Buch-Kromann, 2006) and can be used to train natural language parsers, syntax-based machine translation systems, and other statistically based natural language applications. CDT is unique in creating parallel treebanks for 5 languages and combining this effort with a unitary level of analysis which can provide annotations that span all levels of linguistic analysis, from morphology

¹ The project is hosted on Google Code – <http://code.google.com/p/copenhagen-dependency-treebank/> – and all the sources are freely available.

to discourse, on a principled basis.² Here, however, the centre of attention will be morpho-syntax and semantics.

This paper is structured as follows. In Section 2, it is explained how syntactic structure is annotated in CDT. In Section 3, focus is on how morphological structure is marked-up on the basis of an operator notation system. In section 4, building on the insights reached in the previous sections, the annotation principles for lexical-semantic structure are presented, and, finally, Section 5 sums up the most central points.

2. Syntactic annotation

The syntactic annotation of the treebanks is based on the principles accounted for in the dependency theory Discontinuous Grammar (Buch-Kromann, 2006) and in the CDT-manual (Buch-Kromann et al., 2010). In accordance with other dependency theories, it is assumed that the syntactic structure of a sentence or an NP can be represented as directed relations between governors and complements and adjuncts. Complements function as arguments and are lexically licensed by the governor, whereas adjuncts are modifiers that take the governor as argument.

Figure 1 below shows the primary dependency tree for the sentence *Kate is working to earn money* (top arrows), enhanced with secondary subject relations (bottom arrows). The arrows point from governor to dependent, with the relation name written at the arrow tip.

² Many treebank projects focus on annotating a single linguistic level or a single language: The Penn Treebank (Marcus et al., 1993) focuses on syntax; the Penn Discourse Treebank (Prasad et al., 2008ab) and the RST Treebank (Carlson et al., 2001) on discourse, and the GNOME project (Poesio, 2004) on coreference annotation. Others, like the TuBa-D/Z treebank (Hinrichs et al., 2004), include both morphology and coreference annotation, and the Prague Dependency Treebank (Böhmová et al., 2003) comprises Czech, English and Arabic.

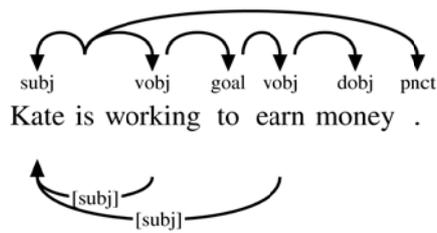


Figure 1. Primary dependency tree (top) augmented with secondary subject dependency relations (bottom).

The finite verb *is* functions as head and top node of the sentence. The arrow from *is* to *Kate* identifies *Kate* as the subject (“subj”), while the arrow from *is* to *working* indicates that *working to earn money* is a verbal object (“vobj”) governed by *is*. The finite verb also functions as governor to the punctuation mark (“pnct”), which is categorized as an adjunct relation. The lexical main verb *working* establishes the adjunct relation (“goal”) to *to earn money*, and inside the adverbial phrase *earn* functions as verbal object (“vobj”) governed by the infinitive marker *to*. Finally, *money* is the direct object (“dobj”) of *earn*. So, in the primary dependency structure every word heads a phrase consisting of all words that can be reached from the phrasal head by following the arrows above the text. The arrows below the text specify secondary subject relations [subj] in the sense that *Kate* is the logical subject of both *working* and *earn*.

The annotation scheme for syntax currently includes approx. 20 complement relations and 60 adjunct relations of which ten of the most frequent ones are listed in Table 1.³

Complement relations	Adjunct relations
nobj (nominal object: <i>for the_{nobj} child_{nobj}</i>)	time (time adverbial: <i>We_{subj} leave now_{time}</i>)
subj (subject: <i>They_{subj} saw me_{dobj}</i>)	loc (location adverbial: <i>I_{subj} fell here_{loc}</i>)
vobj (verbal object: <i>He_{subj} had left_{vobj} it_{dobj}</i>)	man (manner adverbial: <i>I_{subj} read slowly_{man}</i>)
dobj (direct object: <i>He_{subj} left us_{dobj}</i>)	quant (degree adverbial: <i>very_{quant} hard</i>)
pobj (prepositional obj.: <i>one of_{pobj} them_{nobj}</i>)	neg (negation: <i>I_{subj} will not_{neg} leave_{vobj}</i>)
preds (subject predic.: <i>It_{subj} was blue_{preds}</i>)	pnct (punctuation: <i>It_{subj} is !_{pnct}</i>)
@loc (locative object: <i>living in_{@loc} Rome_{nobj}</i>)	attr (attribution: <i>la tarea_{nobj} difficil_{attr}</i>)
predo (object predicative: <i>We_{subj} found it_{dobj} disappointing_{predo}</i>)	appr (restrictive apposition: <i>the genius_{nobj} Einstein_{appr}</i>)
iobj (indirect object: <i>We_{subj} gave him_{iobj} flowers_{dobj}</i>)	appa (parenthetic apposition: <i>Einstein, the_{appa} genius_{nobj}</i>)
avobj (adverbial object: <i>as before_{avobj}</i>)	relnr (restrictive relative clause: <i>the cat_{nobj} that_{subj} died_{relnr}</i>)

Table 1. Ten of the most frequent complement and adjunct relations in the syntactic annotation.

Generally, on the sentence level we do not annotate semantic relations, only syntactic dependency relations. However, in order to achieve a more fine-grained analysis of modifiers, adverbials are annotated according to the semantic relation established between governor and adverbial, cf., e.g., the relations (“time”), (“loc”) and (“man”) in Table 1. The introduction of semantic relations into the syntactic annotation is, of course, debatable, but was preferred over annotating all adverbial adjuncts as (“mod”) relations in accordance with their syntactic function, an alternative adopted in earlier versions of CDT.

With respect to interannotator agreement an experiment has been conducted where two annotators annotated 21 English and Danish texts with a total of 4287 relations. The results were the following:⁴

81%: *Full labeled agreement*, i.e. the probability that another annotator assigns the same label and out-node to the relation.

³ For a full specification of the relation inventory see CDT manual on <http://code.google.com/p/copenhagen-dependency-treebank/>.

⁴ See CDT manual (op.cit).

93% : *Unlabeled agreement*, the probability that another annotator assigns the same out-node (but not necessarily label) to the relation.

85% : *Label agreement*, the probability that another annotator assigns the same label (but not necessarily out-node) to the relation.

In general, the results are satisfactory and prove the system to be quite solid.

3. Morphological annotation

The morphological annotation in CDT only deals with derivation and composition, since inflectional morphology can be detected and analysed automatically with high precision for the treebank languages.

The internal structure of words is encoded as a dependency tree. However, in order to annotate dependency relations inside solid orthography compounds and derivationally constructed words, which appear as tokens in the automatically produced word tokenisation, an operator notation scheme has been developed (Müller, 2010). The operator notation is an abstract specification of how the dependency tree for a morphologically complex word is constructed from roots, annotated as lemmas or in some cases imperatives, dependent on the specific language, in combination with morphological operators. Examples of this notation form, applied to derived nouns and nominal compounds in Danish, are shown in figure 2 to 5.⁵

Antistof [antibody]:
stof –anti/NEG:contr

Figure 2. Operator notation of the Danish prefixed derivation *antistof* [antibody].

Lancering [launching]:
lancer +ing/DERvn:core

Figure 3. Operator notation of the Danish suffixed derivation *lancering* [launching].

Loftslampe [ceiling lamp]:
lampe –[loft]/s/LOC

Figure 4. Operator notation of the Danish compound *loftslampe* [ceiling lamp].

Vindmølle [wind mill]:
mølle –vind/FUNC

Figure 5. Operator notation of the Danish compound *vindmølle* [wind mill].

In Figure 2, the Danish word *antistof* [antibody] is constructed from the root *stof* [body] by attaching the prefix *anti-* as a “NEG:contr” dependent of the root. The “NEG:contr” relation indicates that *anti-* negates the meaning of *stof* so that the new word acquires the opposite meaning of the base. The minus sign introducing the notation specifies the pre-head position of the prefix. In Figure 3, the word *lancering* [launching] is constructed from *lancer* [launch] by transforming the verbal root into a predicative eventive core noun by means of the transformative suffix *-ing* which takes *lancer* as its dependent. Here, the plus sign indicates the post-head position of the suffix. With respect to dependency, the operator notation follows the convention that transformative affixes take the root as dependent, whereas non-transformative affixes are dependents to the root.

The analyses of the minimally complex Danish compounds in Figure 4 and 5 can be explained in the following way: *Loftslampe* [ceiling lamp] in Figure 4 is composed of the modifier *loft* [ceiling], the head *lampe* [lamp] and the linking consonant or interfix *-s*. The annotation is to be understood as follows: The minus sign specifies the pre-head position of the modifier, the lexical material of the modifier itself occurs in square brackets, then comes the interfix which is a phonetically induced morpheme which only acts as a glue between the head and the modifier, and finally, following the oblique slash, the meaning aspect of the head noun selected by the non-head modifier, in this case a locative meaning relation. The analysis of

⁵ In CDT, the three word-classes nouns, adjectives and verbs are marked-up according to the operator notation scheme, but, for matters of space, we only provide examples with nouns. Moreover, CDT has a system for separating linking elements such as thematic vowels, infixes and interfixes, on the one hand, from what is the suffix proper, on the other hand, and it allows CDT to regenerate the word form in question on the basis of the operator instructions. This system is also not detailed here.

vindmølle [wind mill] in Figure 5 follows the same scheme, but here the meaning component activated by the modifier is functional.

Of course, the system must also be able to handle more complex expressions, such as, e.g., the combination of derivation and compounding, cf. Figure 6 below.

<i>Flerbrugersystem</i> [multiple user system]: system -[[brug@V] +er/DERvn:agent -fler/MOD:quant]/GOAL
--

Figure 6. Operator annotation of the Danish compound *flerbrugersystem* [multiple user system].

The head of the compound is the simple lexeme *system* [system], and the non-head is the complex lexeme *flerbruger-* [multiple user]. The operator notation of the complex non-head lexeme, i.e. “-[[brug@V] +er/DERvn:agent -fler/MOD:quant]/GOAL”, should be analyzed step by step as follows:

1. the minus sign introducing the square brackets that delineate the non-head indicates the pre-head position of the non-head.
2. ”[[brug@V] +er/DERvn:agent” specifies that the derivationally complex head *bruger* [user] is an agent nominalization of the verb *bruge* [use] triggered by the suffix *-er*. (The indication of word class in separate square brackets with the specification “@word-class” is optional, but it should be indicated when the form is ambiguous, as in this case between a noun and a verb.)
3. “-fler/MOD:quant” indicates via the minus sign the pre-head position of *fler* [multiple] with respect to *bruger* [user], and that the semantic relation established is one of quantificational modification, cf. “MOD:quant”.
4. Finally, the last part of the operator, i.e. “/GOAL”, specifies that the primary level non-head prompts a semantic (“goal”)-relation between the non-head and the head in the sense that the interpretation of *flerbrugersystem* is a system which has the goal/purpose of several people being able to use it.

Summarizing, in the operator annotation the dependency tree for a morphological complex

lexeme is annotated as a root – given abstractly by means of its lemma or imperative form – followed by one or more operators “*lemma op₁ op₂...*” applied in order. Each operator encodes an abstract affix and a specification of how the abstract affix combines with the base (root or complex stem) in its scope. Here, *abstract affix* is used to denote either a traditional affix (prefix or suffix) or the non-head constituent of a compound. The operator itself has the form “*pos affix/type*”. The field *pos* specifies whether the abstract affix is attached to its base in prefix position (“-”) or suffix position (“+”), or a combination of these in case of parasynthetic verbs, cf. Table 2 (*adormecer* [lull to sleep]). The field *type* specifies the derivational orientation (e.g., “DERvn”, {fig. 3}), either in the form of a categorial shift, or not. Moreover, the field *type* semantically and functionally identifies the type and, where relevant, the subtype, of the semantic relation created between the base and the abstract affix (e.g., “NEG:contr”, {fig 2}). The field *affix* specifies the abstract affix and its possibly complex internal structure. The abstract affix may be encoded either as a simple string representing a simple affix or a simple root (e.g., *-er*, “brug”, {fig. 6}), or as a complex string of the form “[*stem*]” or “[*stem*]*interfix*”, where “*stem*” encodes the internal structure of the abstract affix in operator notation (e.g., “-[loft]s/LOC” or “-vind/FUNC”, {fig. 4 and 5}).

As mentioned previously, the abstract affix functions as a dependent of the base when it is non-transformational, whereas if it triggers word class change or a significant change of meaning, the base is assumed to function as a dependent of the abstract affix.

Finally, it is important to keep in mind that the operator notation is merely an abstract specification of a dependency tree, not an autonomous annotation system which follows individual rules.

A sample of morphological relation types is listed in Table 2 below.⁶ The system is flexible in the sense that all relations can be annotated as either prefixes or suffixes, or non-head roots in case of compounds; here they are just listed as they typically appear in the CDT languages.

⁶ The different relation types have taken inspiration from the works on morphological categories by Rainer (1999) and Varela and Martín García (1999). The total number of morphological relation types in CDT is 70, out of which 57 are derivational relations (17 prefix; 40 suffix) and 13 compositional relations (see CDT-manual, cf. footnote 3).

<p>Relations that typically appear with prefixes</p> <p>SPACE:loc (location: <i>intramural</i> = <i>mural</i> –<i>intra</i>/SPACE:loc)</p> <p>TIME:pre (precedency: <i>prehistorical</i> = <i>historical</i> –<i>pre</i>/TIME:pre)</p> <p>NEG:contr (contrast: <i>antihero</i> = <i>hero</i> –<i>anti</i>/NEG:contr)</p> <p>AGENT (causative: <i>acallar</i> ‘silence’ = <i>callar</i> –<i>a</i>/AGENT)</p> <p>TELIC (telic: <i>oplåse</i> ‘open’ = <i>låse</i> –<i>op</i>/TELIC)</p> <p>MOD:quant (quantification: <i>multicultural</i> = <i>cultural</i> –<i>multi</i>/MOD:quant)</p> <p>TRANS (transitivity: <i>påsejle</i> ‘colide’ = <i>sejle</i> –<i>på</i>/TRANS)</p> <p>Relations that typically appear with suffixes</p> <p>AUG (augmentative: <i>perrazo</i> ‘big dog’ = <i>perro</i> +<i>azo</i>/AUG)</p> <p>DIM (diminutive: <i>viejecito</i> ‘little old man’ = <i>viejo</i> +<i>ecito</i>/DIM)</p> <p><i>Verb derivation</i></p> <p>DERnv (noun→verb derivation: <i>salar</i> ‘to salt’ = <i>sal</i> +<i>ar</i>/DERnv)</p> <p>DERav (adjective→verb derivation: <i>darken</i> = <i>dark</i> +<i>en</i>/DERav)</p> <p>DERvv (verb→verb derivation: <i>adormecer</i> ‘lull to sleep’ = <i>dormir</i> –+[a][ecer]/DERvv)</p> <p><i>Noun derivation</i></p> <p>DERvn:agent (verb→noun derivation: <i>singer</i> = <i>sing</i> +<i>er</i>/DERvn:agent)</p> <p>DERvn:core (verb→noun derivation: <i>exploitation</i> = [<i>exploit@V</i>] +<i>ation</i>/DERvn:core)</p> <p>DERnn:cont (noun→noun derivation: <i>azucarero</i> ‘sugar bowl’ = <i>azucar</i> +<i>ero</i>/DERnn:cont)</p> <p><i>Adjective derivation</i></p> <p>DERva:pas.epi (deverbal adjective: <i>transportable</i> = <i>transport</i> +<i>able</i>/DERva:pas.epi)</p> <p>DERna:rel (denominal adjective: <i>presidential</i> = <i>president</i> +<i>ial</i>/DERna:rel)</p> <p>Relations that typically appear with compounds</p> <p>CONST (constitutive: <i>træbord</i> ‘wooden table’ = <i>bord</i> –<i>træ</i>/CONST)</p> <p>AGENT (agent: <i>politivold</i> ‘police violence’ = <i>kontrol</i> –<i>politi</i>/AGENT)</p> <p>SOURCE (source: <i>rørsukker</i> ‘cane sugar’ = <i>sukker</i> –<i>rør</i>/SOURCE)</p> <p>GOAL (goal: <i>krigsskib</i> ‘war ship’ = <i>skib</i> –[<i>krig</i>]/s/GOAL)</p> <p>FUNC (function: <i>vindmølle</i> ‘wind mill’ = <i>mølle</i> –<i>vind</i>/FUNC)</p> <p>LOC (location: <i>loftlampe</i> ‘ceiling lamp’ = <i>lampe</i> –[<i>loft</i>]/s/LOC)</p>

Table 2. Relation types in the morphological notation system.

4. The semantic dimension

4.1 Basic annotation of NPs

A number of semantic annotation projects have developed over the years.⁷ In CDT, the dependency structure has been enhanced with semantic annotation with respect to sentence level adverbials, derivations and different kinds of NPs. In this context, we limit ourselves to focusing on the description of how Generative Lexicon theory (GL) has been integrated into the current dependency framework in order to account for the lexical semantics of certain NPs.

GL (Pustejovsky, 1991, 1995, 2001) is based on the assumption that any lexeme can be defined by the four qualia, FORMAL, CONSTITUTIVE, TELIC and AGENTIVE, which constitute the fundamental rules according to which the integration of mental representations of entity types is produced. In other words, Qualia can be described as a template representing the relational force of a lexical item, a system of constructive understanding and inference.

Below, we exemplify the integration of lexical semantic knowledge in the dependency-based multilevel CDT annotation scheme by describing the annotational challenges posed by one single type of NPs, viz. Spanish N+PP constructions.

In N+PP constructions like *taza de café* [coffee cup] and *taza de porcelana* [china cup], the PP-modifiers *de café* and *de porcelana* are syntactic dependents of the head *taza*, but they select different sub-senses of *taza*, Telic and Constitutive, respectively, and act semantically as governors (Johnston and Busa, 1999).⁸ The relationship between syntactic and semantic dependencies is implemented in terms of annotation in the following way.

⁷ *PropBank* (Palmer et al., 2005) is a corpus semantically annotated with verbal propositions and their arguments; *NomBank* (Meyers et al., 2004ab) marks up the sets of arguments that co-occur with nouns; *VerbNet* marks up the sets of syntactic frames a verb can appear in to reflect underlying semantic components constraining allowable arguments; and *FrameNet* (Ruppenhofer et al., 2006) is an on-line lexical resource for English based on frame semantics and supported by corpus evidence.

⁸ In practice, CDT operates with an expanded set of qualia-roles. For instance, the Telic-role can manifest itself either as Goal or Function (see Table 2), dependent on the specific interpretation.

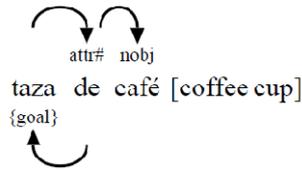


Figure 7. Syntactic and semantic annotation of the Spanish phrasal NP-compound *taza de café* [coffee cup].

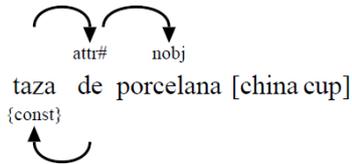


Figure 8. Syntactic and semantic annotation of the Spanish phrasal NP-compound *taza de porcelana* [china cup].

The arrows above the text from the head *taza* [cup] to the PPs *de café* [of coffee] and *de porcelana* [of china] in Figure 7 and 8, respectively, indicate that the relation is non-argumental, i.e. what we understand as one of contribution (“attr”) – basically because the head is non-predicative or non-relational. In other words, the non-head is not lexically licensed by the governing head. The hash symbols following the (“attr”) label stipulate that the phrases in question show composite structure (see later discussion). The nouns *café* and *porcelana* are syntactically governed by the preposition *de* and function as noun objects (“nobj”). The “reversed” arrows below the text indicate semantic structure. The non-heads activate the Telic quale – we refer to it as a (“goal”) relation – and the Constitutive quale of the head, respectively, being the general assumption that the qualia of the head can be triggered by different modifiers, in these cases PPs.⁹

Moreover, *taza de café* is ambiguous as it allows yet another interpretation equivalent to cup of coffee, where *taza* functions as a specifier of quantity. In these cases it is the complement *café* which has to respect the selectional restrictions imposed by, e.g., the predicate, and, consequently, the construction must be re-analyzed as yielding a specifier+head structure, i.e. a case of head switching, cf. Figure 9 below.

⁹ Of course, the preposition *de* in itself is purely syntactic, but we have chosen to see the whole PP as the unit which activates the semantic relation between head and non-head.

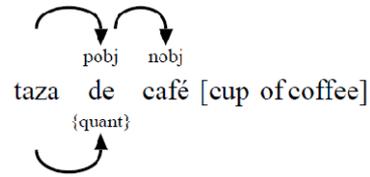


Figure 9. Syntactic and semantic annotation of Spanish NP expressing quantification.

In terms of annotation the difference between Figure 7 and 9 is that in Figure 9 the noun *taza* is relational and thus selects the PP *de café* as a dependent. Therefore *de café* functions as an argument to the head, which is made clear by the fact that the relation name written at the arrow tip is (“pobj”), a lexically governed prepositional object. Consequently, the syntactic labels (“pobj”) and (“nobj”) indicate that the modifying noun or PP is lexically governed by the head, whereas the (“attr”)-label indicates that this is not the case. The label (“nobj”) is also used more widely when a noun is governed by an article or a preposition. The arrow below the text indicates that *taza* does not function as a semantic head, but as a specifier which imposes a quantificational reading on the PP. Therefore the arrows showing syntactic and semantic dependency, respectively, are oriented in the same direction in this case.

Apart from the Qualia inspired inventory of semantic relations, CDT also operates with a set of “standard” semantic roles in the form of Agent, Patient, Recipient, etc. These roles are used when the head noun is deverbal or deadjectival and thus projects an argument structure, cf. Figure 10.

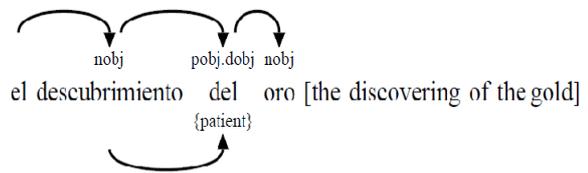


Figure 10. Full syntactic and semantic annotation of Spanish NP with deverbal head.

In Figure 10, the bottom arrow specifies that the PP *del oro* [of-the gold] functions as Patient with respect to the deverbal head noun *descubrimiento* [discovering]. The top arrow from head noun to PP demonstrates that the PP is a syntactically governed (“pobj”) with the function of direct object (“dobj”).

Generally, the qualia-structure has been a guiding principle for the organization of the semantic inventory of CDT on all levels, i.e. with respect to adverbial adjuncts, NPs and derivational morphology.¹⁰ This attempt to unify the inventory through the qualia-structure, which provides a rather general template for structuring semantic relations, is theoretically appealing because it accommodates the fact that similar semantic relations are found on different linguistic levels. However, this does not mean that any semantic relation can be accounted for with point of departure in the qualia-structure. For instance, the nature of the arguments to a predicate (semantic labeling), cf. Figure 10, or certain adverbial adjunct relations, such as condition, concession, contrast, etc., fall outside the explanatory frame of the qualia-structure.

4.2 Compounding

As mentioned before, we use the hash symbol to indicate when a phrasal constellation of words should be regarded as a compound. Of course, in the non-English Germanic languages it is not a problem as they have unitary stress (e.g. in Danish, head nouns are reduced prosodically and pronounced with secondary stress) and solid orthography, which means that in CDT they are tackled in accordance with the so-called operator notation scheme. However, when a word constellation should be regarded as a free syntactic phrase formation or a compound is not an uncontroversial issue, which can be appreciated, for instance, in the Spanish grammatical literature about the subject.

Briefly, the problem is that the criteria for compounding in Spanish, and other Romance languages for that matter, are often based on the notion of degree of lexicalization – the more lexicalized the more compound status – which seems to be difficult to deal with both empirically and theoretically in a setting of annotation.

In the standard approach (e.g., Escandell Vidal, 1995; Val Alvaro, 1999), degree of lexicalization is measured by the parameters of internal solidity, i.e. cohesion between the constitutive elements, and, secondarily, possibility of substitution of elements, and

finally, as an effect of these criteria, degree of semantic transparency.¹¹

According to this approach, good examples of phrasal compounds would be such as the ones in (1) and (2). They have a solid internal structure, and, moreover, the foot-examples in (2) are not semantically transparent. They are exocentrically structured, and they are metaphoric extensions of some original meaning of which we have more or less lost track.

- (1) *un punto de vista*
[a point of view]

**un punto agudo de vista*
[a point **sharp** of view]

un agudo punto de vista/un punto de vista agudo
[a **sharp** point of view/a point of view **sharp**]

- (2) *pie de liebre*
[foot-of-hare] “sort of clover”

pie de atleta
[foot-of-athlete] “sort of skin disease”

pie de gallina
[foot-of-chicken] “sort of knot”

However, the examples in (3) and (4) below are not so good phrasal compounds. They do not show a solid internal structure, and the ones in (4) are even headed by the event denoting deverbal noun *venta* [sale], which means that

¹¹ Other authors (see, e.g., Corpas Pastor, 1997; Ferrando Aramo, 2002; Ruiz Gurillo, 2002; Alonso Ramos, 2009) intend to establish more or less solid distinctions between compounds, locutions/idiomatic expressions, and collocations on the basis of a wide range of syntactic, semantic and denotative criteria, such as cohesion, transparency and unity of meaning. Although a continuum, rather than an attempt to make clear delimitations, probably is the more adequate way to represent these types, there is no doubt that important phraseological distinction can be identified between different N+PP constructions. However, the point deserving emphasis here is that, contrary to the current discussion in the Spanish literature, the definition of compounding in the non-English Germanic languages, such as Danish, does not hinge on the extent to which a certain construction fulfils an array of criteria, but is solely based on the criterion of unitary stress and, consequently, solid orthography. Therefore, although Germanic compounds can show all kinds of semantic “peculiarities”, Germanic compounding is well-defined, while Romance N+PP compounding is a fuzzy edged phenomenon.

¹⁰ This also goes for the CDT annotation of anaphoric relations and discourse structure, which, however, has not been the topic of this paper.

they are completely productive and that they resample the corresponding “free” sentence structure.

- (3) *lazo de luto*
[bow of grief/mourning]

bolsa de viaje
[bag-of-travel/travel bag]

*lazo **negro** de luto*
[bow **black** of grief]

*bolsa **negra** de viaje*
[bag **black** of travel]

- (4) *venta de carne/ trigo/ caballos/ teléfonos*
[sale of meat/ wheat/ horses/ telephones]

It is not the intention here to enter into a theoretical discussion about compounding, but it must be acknowledge that in general the understanding of compounding in Spanish and other Romance languages deviates substantially from a Germanic understanding of the “same” phenomenon, cf. also footnote 11.

In order to cope with these interlingual discrepancies in CDT we have chosen a very liberal approach to Romance compounding in the sense that if the constellation of words in question can be said to designate a single entity or type of entity, we add a hash symbol indicating that the relevant construction shows some kind of tendency towards being a lexical unit. Good signs of such a status is, of course, if the modifying noun, N2, is naked, i.e. appears without determiners, or if an analogous expression in German or Danish manifests itself as a compound (with respect to Germanic compounding see, e.g., Mellenius, 1997; ten Hacken, 1999; Müller, 2001, 2003).

Another problem of compounding is coreless (exocentric) compounds, cf. what with Sanskrit terms is referred to as “bahuvrihi” (e.g., *redskin*), “dvandva” (e.g., *marxism-leninism*) and “imperavitic” (e.g., *forgetmenot*). These constructions are not especially productive, but they do not fit in so neatly in a dependency framework which builds on the assumption that every expression must have a head. This issue also concerns a number of synthetic compounds such as *darkhaired* and *blueeyed*, where it is difficult to decide which element is the head.

With respect to the headedness problem, the CDT, by stipulation, follows the general

principle that the element which carries the inflectional endings also is considered the head. However, one exception to this standard is the issue of verbo-nominal compounds illustrated in (5) and (6) below and annotated according to the operator scheme. In these cases, we follow the principle that the verbal part is the head, and the nominal part, although it carries the inflectional endings, is a modifier, very often in the form of a direct object. The problem arises because there is a discrepancy between the inner dependency structure of the compound, which follows the corresponding sentence structure, and its instantiation in syntax, which dictates an inflectional declension of the modifier, when relevant.

- (5) *un tocadiscos* [a play-records/record player]:
tocar ! +discos/DOBJ.patient

- (6) *un guardapolvo* [a protect-dust/working coat]:
guardar ! +polvo/GOAL

4.3 Semantic agreement figures

Interannotator agreement has also been calculated for semantic relation. This has been done on the basis of the same 21 English and Danish texts that were used for the syntax annotation task, and in this case with a total of 358 semantic relations. The results were the following:¹²

48% : *Full labeled agreement*, i.e. the probability that another annotator assigns the same label and out-node to the relation.

96% : *Unlabeled agreement*, the probability that another annotator assigns the same out-node (but not necessarily label) to the relation.

50% : *Label agreement*, the probability that another annotator assigns the same label (but not necessarily out-node) to the relation.

Obviously, the scores with respect to semantic annotation are rather low in comparison with the syntactic level. A specific analysis of the major disagreement cases has not been conducted yet, but it seems reasonable to suspect that at least

¹² See CDT manual (op.cit).

some of the explanation lies in the fact that the semantic annotation of CDT covers both NPs and derivational morphology, as well as adverbial adjuncts. This makes the system fairly complex and, perhaps, in some respects too detailed. Specifically, informal investigations of compound annotation show that the annotators in many cases tend to disagree on which semantic label should be assigned to the relation between head and non-head. However, we expect to be able to improve the system by introducing a more hierarchical ordering of relations and a higher degree of label specificity.

5. Conclusion

This paper has explained how the basic dependency principles behind the sentence level syntactic analyses, through an operator notation, has been transferred to the morphological level to account for the inner structure of tokens in the form of derivations and compounds. There is a clear analogy between syntactic and morphological annotation in CDT. On both levels we depart from the basic assumption that coherent linguistic units, in the form of either sentences or words, are determined by a dependency structure in which each word or morpheme is assumed to function as complement or adjunct to another word or morpheme, called the governor. In the last part of the paper, we show from a limited subset of examples how GL semantics has been incorporated into a coherent annotation scheme compatible with the CDT dependency principles on the other descriptive levels.

It is expected that the enhancement of CDT with morphological and semantic annotation will enable inquiries into interface issues between different linguistic layers, cross-linguistic contrasts and typological variations between the languages involved in CDT, thereby supporting CDT's applicability in multilingual language processing systems. Of course, these aspects have not been dealt with in the paper, which only introduces the system.

Finally, we have seen that interannotator agreement scores confirm that the system functions robustly with respect to syntax, whereas the annotation of semantic relations is not sufficiently performant yet. Larger scale analyses of the functionality of the morphological annotation system have not been conducted so far, but preliminary studies are generally positive in terms of the user

friendliness of the system, despite its obvious complexity. However, on the critical side the annotators find the system time-consuming to get familiar with.

References

- Alonso Ramos, M. (2009). Delimitando la intersección entre composición y fraseología. *Lingüística española actual* (LEA), 31(2). 5-37.
- Böhmová, A., Hajič, J., Hajičová, E. & Hladká, B. (2003). The Prague Dependency Treebank: a three-level annotation scenario. In A. Abeillé (ed.). *Treebanks: Building and Using Parsed Corpora*. Dordrecht: Kluwer Academic Publishers.
- Buch-Kromann, M. (2006). *Discontinuous Grammar. A dependency-based model of human parsing and language learning*. Doctoral dissertation. Copenhagen: Copenhagen Business School.
- Buch-Kromann, M., Korzen, I. & Müller, H.H. (2009). Uncovering the 'lost' structure of translations with parallel treebanks. In I.M. Mees, F. Alves, & S. Göpferich (eds). *Methodology, Technology and Innovation in Translation Process Research. Copenhagen Studies in Language* 38: 199-224.
- Buch-Kromann, M., Gylling, M., Knudsen, L.J., Korzen, I. & Müller, H.H. (2010). *The inventory of linguistic relations used in the Copenhagen Dependency Treebanks*. Technical report. Copenhagen: Copenhagen Business School. Available at: <http://code.google.com/p/copenhagen-dependency-treebank/>.
- Carlson, L., Marcu, D. & Okurowski, M. E. (2001). Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In *Proceedings of the 2nd SIGdial Workshop on Discourse and Dialogue*.
- Corpas Pastor, G. (1997). *Manual de fraseología española*. Madrid: Gredos.
- Escandell Vidal, M.V. (1995). *Los complementos del nombre*. Madrid: Arco Libros.
- Ferrando Aramo, V. (2002). Colocaciones y compuestos sintagmáticos. In A. Veiga Rodríguez, M. González Pereira & M. Souto Gómez (eds.). *Léxico y Gramática*. TrisTram, Lugo. 99-107.
- Hinrichs, E., Kubler, S., Naumann, K., Telljohann H. & Trushkina, J. (2004). Recent developments in linguistic annotations of the TuBa-D/Z treebank. In *Proceedings of the Third Workshop on Treebanks and Linguistic Theories*. Tübingen, Germany. 51-62.

- Johnston, M. & Busa, F. (1999). The compositional interpretation of compounds, In E. Viegas (ed.). *Breadth and Depth of Semantics Lexicons*. Dordrecht: Kluwer Academic. 167-87.
- Kromann, M. T. (2003). The Danish Dependency Treebank and the DTAG treebank tool. In *Proceedings of the Second Workshop on Treebanks and Linguistic Theories (TLT 2003), 14-15 November, Växjö*. 217-220.
- Marcus, M. P., Marcinkiewicz, M. A. & Santorini, B. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics* 19(2). 313-330.
- Mellenius, I. (1997). *The Acquisition of Nominal Compounding in Swedish*. Lund: Lund University Press.
- Meyers, A. et al. (2004a). The NomBank Project: An interim report. In *Proceedings of the HLTNAACL Workshop on Frontiers in Corpus Annotation*, Boston, MA. 24-31.
- Meyers, A. et al. (2004b). Annotating noun argument structure for NomBank. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*. Lisbon, Portugal.
- Mladová, L., Š. Zikánová & Hajičová, E. (2008). From sentence to discourse: building an annotation scheme for discourse based on Prague Dependency Treebank. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*. 2564-2570.
- Müller, H.H. (2001). Spanish N de N-structures from a cognitive perspective. In I. Baron, M. Herslund, & F. Sørensen (eds.). *Dimensions of Possession*. Amsterdam/Philadelphia: Benjamins. 169-186.
- Müller, H.H. (2003). Strategies de lexicalisation des noms composés en espagnol. In M. Herslund (éd.). *Aspects linguistiques de la traduction*. Bordeaux: Presses Universitaires de Bordeaux. 55-84.
- Müller, H.H. (2010). Annotation of Morphology and NP Structure in the Copenhagen Dependency Treebanks. In M. Dickinson, K. Müürisepp, & M. Passarotti, (eds.). *Proceeding of the Ninth International Workshop on Treebanks and Linguistic Theories*. (NEALT Proceedings Series). 151-162.
- Palmer, M., Gildea, D. & Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1). 71-106.
- Poesio, M. (2004). Discourse annotation and semantic annotation in the GNOME corpus. In *Proceedings of the ACL Workshop on Discourse Annotation*.
- Prasad, R., Miltsakaki, E., Dinesh, A, Lee, A., Joshi, A., Robaldo L. & Webber, B. (2008a). *The Penn Discourse Treebank 2.0. Annotation Manual*. (IRCS Technical Report IRCS-08-01). University of Pennsylvania: Institute for Research in Cognitive Science.
- Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A. & Webber, B. (2008b). The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*.
- Pustejovsky, J. (1991). The Generative Lexicon. *Computational Linguistics* 17. 409-441.
- Pustejovsky, J. (1995). *The Generative Lexicon*. Cambridge (Mass.). London, England: MIT Press.
- Pustejovsky, J. (2001). Generativity and Explanation in Semantics: A Reply to Fodor and Lepore. In P. Bouillon & F. Busa (eds.). *The Language of Word Meaning*. Cambridge University Press. 51-74.
- Rainer, F. (1999). La derivación adjectival. In I. Bosque. & V. Demonte (eds). *Gramática Descriptiva de la Lengua Española*. Madrid: Espasa Calpe. 4595-4643.
- Ruiz Gurillo, L. (2002). Compuestos, colocaciones, locuciones: intent de delimitación. In A. Veiga Rodríguez, M. González Pereira & M. Souto Gómez (eds.). *Léxico y Gramática*. TrisTram, Lugo. 327-339.
- Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C. & Scheffczyk, J. (2006). *FrameNet II: Extended Theory and Practice*.
- ten Hacken, P. (1999). Motivated Tests for Compounding. *Acta Linguistica Hafniensa* 31. 27-58.
- Val Álvaro, J.F. (1999). La composición. In I. Bosque & V. Demonte (eds.). *Gramática descriptiva de la lengua española*. Madrid: Espasa Calpe. 4757-4841.
- Varela, S. & Martín García, J. (1999). La prefijación. In I. Bosque. & V. Demonte (eds.). *Gramática Descriptiva de la Lengua Española*. Madrid: Espasa Calpe. 4993-5040.

Dependency Annotation of Coordination for Learner Language

Markus Dickinson
Indiana University
md7@indiana.edu

Marwa Ragheb
Indiana University
mragheb@indiana.edu

Abstract

We present a strategy for dependency annotation of corpora of second language learners, dividing the annotation into different layers and separating linguistic constraints from realizations. Specifically, subcategorization information is required to compare to the annotation of realized dependencies. Building from this, we outline dependency annotation for coordinate structures, detailing a number of constructions such as right node raising and the coordination of unlikes. We conclude that branching structures are preferable to treating the conjunct as the head, as this avoids duplicating annotation.

1 Introduction and Motivation

While corpora containing the language of second language learners have often been annotated for errors (e.g., Nicholls, 2003; Rozovskaya and Roth, 2010), they have rarely been annotated for linguistic properties. Those which mark part-of-speech (POS) tend to do so only for illicit forms (e.g., Granger, 2003) and those with syntactic annotation generally first map the learner forms to target forms (e.g., Hirschmann et al., 2010). While these annotations serve many purposes, what has been lacking is linguistic annotation of the learner data itself, in particular syntactic annotation (Dickinson and Ragheb, 2009). As argued in Ragheb and Dickinson (to appear), such annotation has the potential to be beneficial for much second language acquisition (SLA) research, to address questions such as complexity (e.g., Pendar and Chapelle, 2008) and stage of acquisition (e.g., Pienemann, 1998). Such annotation is also suited to evaluate the parsing of learner data (Ott and Ziai, 2010).

We outline an annotation framework for applying syntactic dependency annotation to learner

corpora, focusing on the challenges stemming from coordination for learner structures. The first issue in annotating dependencies for learner language has to do with the fact that learner data diverges from canonical language use. We build from proposals which thus split the annotation into separate levels, one for each piece of evidence. In (1), from (Díaz Negrillo et al., 2010), the word *jobs* is distributionally in a singular noun slot, but has the English plural marker. Díaz Negrillo et al. propose separate layers of part-of-speech (POS) annotation to account for this (see section 2).

- (1) ... for almost every **jobs** nowadays ...

Splitting annotation into different layers for different types of linguistic evidence is applicable to dependency annotation (Dickinson and Ragheb, 2009), but as we will describe in section 3, there is also a need to separate linguistic constraints from the actual realizations, in order to capture non-native properties. Subcategorization requirements, for example, do not always match what is realized.

Coordination is one particularly difficult area for dependency annotation (e.g., Nivre, 2005). When linguistic constraints are separated from realizations, coordination becomes a prominent issue for learner annotation, as the constraints (subcategorization) and the realizations (dependencies) need to be appropriately matched up. Our annotation scheme should: 1) be useful for SLA research (Ragheb and Dickinson, to appear), 2) be as simple as possible to annotate, and 3) cover any learner sentence, regardless of the proficiency level. Balancing these concerns and taking our multi-layered approach to annotation into account (sections 2 and 3), we will advocate a branching approach to coordination in section 4. Such an approach treats every dependency independently, avoiding the duplication of information.

2 Annotating learner language

There has been a recent trend in annotating the grammatical properties of learner language, independent of errors (Díaz Negrillo et al., 2010; Dickinson and Ragheb, 2009; Rastelli, 2009). While error annotation has been the standard annotation in learner corpora (e.g., Granger, 2003; Díaz Negrillo and Fernández Domínguez, 2006), annotation of linguistic properties such as POS and syntax provides SLA researchers direct indices to categories of interest for studying interlanguage (Pienemann, 1992; Ragheb and Dickinson, to appear). One does not posit a correct version of a sentence, but annotates only what is observed.

Consider again example (1): a single POS is not appropriate, as the distributional evidence for *jobs* is of a singular noun, and the morphological evidence is plural. Díaz Negrillo et al. (2010) propose annotating 3 tags, representing the morphological, distributional, and lexical evidence. Each POS layer, then, contains a separate description of a linguistic property. The POS is not claimed to be a single category; rather, the evidence is represented in different layers, thereby providing access for searching. Errors in this framework are epiphenomena, arising from conflicts between layers.

Using SUSANNE tags (Sampson, 1995), we see an example of two layers in (2), where the distributional layer contains a present tense verb (VVZt) and the morphological layer a base form verb (VV0t).¹ In a sense, this parallels the multi-layered annotation in Lüdeling et al. (2005), where each error interpretation is given its own layer.

(2) Tin Toy can **makes** different music ...
NP1x NP1x VMo **VVZt** JJ NN1u ...
NP1x NP1x VMo **VV0t** JJ JJ ...

These annotation efforts are still in the early stages of development, making the conceptual issues clear. Because much SLA research is framed in terms of linguistic categories—e.g., the use of extraction from embedded clauses (e.g., Juffs, 2005; Wolfe-Quintero, 1992)—the annotation has much potential to be useful. We turn next to annotating dependencies in this framework.

3 Dependencies for learner language

We will provide a sketch of the annotation layers we use, emphasizing the split between the anno-

¹Unless otherwise noted, our learner examples come from a corpus of narratives from the 1990s (Bardovi-Harlig, 1999).

tation of realized dependencies (section 3.2) and subcategorization (section 3.3).

3.1 Completeness, Coherence, & Consistency

Leaving aside the separation of linguistic evidence for the moment, we start with the general use of dependencies, which directly capture selection and modification relations. We focus on capturing selectional properties, which means dealing with issues of: 1) completeness, 2) coherence, and 3) consistency (cf. Lexical-Functional Grammar (LFG), Bresnan, 2001). Violations of these are given in the constructed examples in (3). Example (3a) represents an incomplete structure, in that the verb *devour* selects for an object, which is not realized. For *completeness* to hold, all the arguments of a predicate must be realized.

- (3) a. *Max devoured.
b. *Max slept a tree.
c. *Max devoured of a sandwich.

In (3b), there is an incoherent structure, as there is an extra argument: for *coherence*, there must be no additional arguments. Finally, (3c) is inconsistent, as there is a prepositional phrase, but *devoured* selects a noun phrase. To be *consistent*, the realized arguments must match those selected for. Since learners produce structures with a mismatch between the selectional requirements and the realized arguments, we want to represent both.

3.2 Modeling dependencies

3.2.1 Distributional dependencies

We first annotate the relations occurring in the sentence, using the target language (English) as a reference frame to define the relations, e.g., what it means to be a subject. By **distributional dependencies**, we refer to dependencies between words based strictly on syntactic distribution, i.e., primarily word order. Building from Dickinson and Ragheb (2009), we focus on these dependencies; other layers are discussed in section 3.2.3.

In (4), for example, *baby* is in the distributional slot of the subject of *had*, as defined by English declarative structure.

- (4) The **baby** had no more interest ...

To see the need for defining dependencies on a strictly syntactic basis, consider (5). The word *dull* (cf. *doll*) is ambiguous: it could be an object of *escape* (with a missing subject), or it could be

the subject in the wrong location. To fully disambiguate requires knowing learner intention, a difficult proposition for consistent and reliable annotation. Looking only at distribution, however, this position in English is an object position.

(5) After the baby down, escape the **dull**.

The tree for this example is shown in figure 1, where *dull* is the object (OBJ). The non-nativeness of this sentence is captured via the encoding of subcategorization requirements (section 3.3).

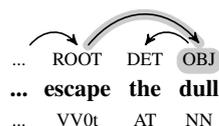


Figure 1: Distributionally-based dependencies, with distributional POS tags

We use the CHILDES annotation scheme (Sagae et al., 2010, 2007) as the basis for our annotation, as it was developed for language being acquired (albeit, first language), with two main differences: 1) They treat main verbs as heads, with auxiliaries and infinitive markers (*to*) as dependents, whereas we mark auxiliaries as heads, following work treating them on a par with raising verbs (e.g., Pollard and Sag, 1994). 2) They treat the conjunct in coordinate structures as the head, whereas we investigate this approach and a binary-branching approach, ultimately arguing for branching. For branching, we introduce a new label, CC (coordinating conjunction), for the relation with the conjunction as a dependent.

3.2.2 Secondary dependencies

Given the widely-held assumption that each word has only one head in a dependency graph (Kübler et al., 2009, ch. 2), basic dependencies cannot capture every relationship. In the learner example (6), for instance, *I* is the subject for the verbs *hope* and *do*. Allowing for additional dependencies to be specified (cf. Kromann, 2003; Sgall et al., 2004), this can be fully represented.

(6) ... the only thing that **I** hope to do ...

We thus annotate **secondary dependencies**, which encode non-local syntactic relationships between words. Such secondary dependencies are represented in figure 2 with arcs below the words. One could argue that secondary dependencies are

semantic; we try to restrict usage to cases where: a) a syntactic process is involved, in this case control, and b) the subcategorization of predicates is at stake (section 3.3). As we will see in section 4, secondary dependencies are crucial to capturing the selected dependents of coordinated functors.

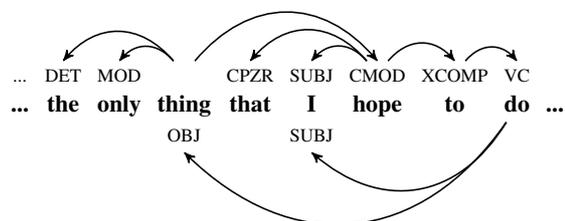


Figure 2: Encoding secondary dependencies

3.2.3 Other types of dependencies

We focus on distributional dependencies in this paper, as this is sufficient to illustrate the issues faced with coordination. Other types of dependencies can and should be annotated for learner language, including **morpho-syntactic** and **semantic dependencies**. Splitting dependencies into different layers of evidence has precedence in a variety of frameworks (e.g., Mel'čuk, 1988; Debusmann et al., 2004; Deulofeu et al., 2010).

For morpho-syntactic dependencies, consider the constructed example (7): *Him* is in the subject distributional position, but morphologically has object marking. The interplay between morphological and distributional layers will vary for different language types (e.g., freer word order).

(7) **Him** slept.

Semantic dependencies would capture the canonical linking of dependencies to meaning (e.g., Ott and Ziai, 2010; Hirschmann et al., 2010). Consider *see* in (8). The distributional position of the subject is filled by *Most (of the movie)*, while the object is *adults*, but on a semantic layer of dependencies, *adults* may be the subject and *Most* the object. Again, this is an orthogonal issue.

(8) **Most of the movie** is seem to see **adults**, but the chieldern like to movie.

3.3 Modeling subcategorization

Dependencies are based on evidence of what learners are doing, but to capture completeness, coherence, and consistency, we need to model

which dependencies are selected for, namely **subcategorization** information.

We annotate subcategorization frames on the basis of the requirements in the target language (English). For example, in (5), the subordinate clause is missing a verb. One way to capture this is in figure 3, where *baby* is the subject (SUBJ) of *down*, but *down* has an empty subcategorization list (<>). Since subjects are arguments, this mismatch indicates an issue with coherence. By contrast, *baby* subcategorizes for a determiner (<DET>), which is realized.

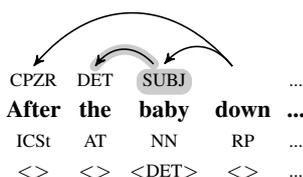


Figure 3: Partial tree with dependencies, distributional POS tags, and subcategorization frames

Words may have many subcategorization frames (Levin, 1993), and we annotate the one which is the best fit for a given sentence. In the constructed cases in (9), for example, *loaded* receives different annotations. In (9a), it is <SUBJ, OBJ>, while in both (9b) and (9c), it is <SUBJ, OBJ, IOBJ-*with*>. For (9c), this is the best fit; while still not matching what is in the sentence, it means that only one element (OBJ) is missing, as opposed to, e.g., <SUBJ, OBJ, IOBJ-*into*>, where two elements would be wrong.

- (9) a. Max **loaded** the wagon.
 b. Max **loaded** the wagon with hay.
 c. *Max **loaded** with hay.

Treatment of raising and control Consider (6) again: in *hope to do*, the subject of *do* is essentially the same as that of *hope*, and in many theories, *to* “raises” the subject, keeping relations local. We can see subcategorization information in figure 4.

It is not immediately clear whether we should explicitly annotate raising and put SUBJ on *to*’s subcategorization frame. We are trying to base the annotation on well-founded grammatical theory, but the primary criteria are: a) to make the data useful for SLA research, and b) to be able to annotate efficiently. Thus, even if a theoretical model supports the annotation, we do not necessarily need to annotate all parts of it.

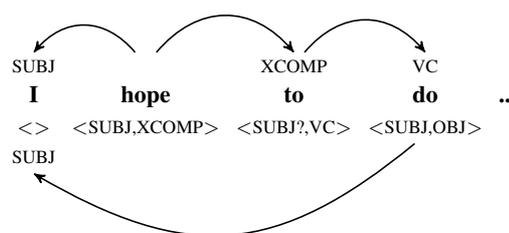


Figure 4: Treating raising and control

We advocate not annotating raising in all cases. This is simpler for annotation, especially as we get into the sharing of elements between conjuncts. We expect more efficient and reliable annotation by annotating the minimal required elements. Additionally, keeping subcategorization simple makes us less committed to any theoretical claims for, for example, right node raising (section 4.2). When coordinated verbs share an object, we do not have to determine whether the object is percolated up to the conjunction; there is simply a long-distance relationship where appropriate.

Technical details We encode our annotation by extending the CoNLL format (Buchholz and Marsi, 2006) to account for secondary dependencies (see details in Dickinson and Ragheb, 2009). We are also extending the format to encode both distributional and morpho-syntactic dependencies.

4 Our treatment of coordination

There are many ways to handle coordination in dependency annotation (see, e.g., Osborne, 2008, sec. 5), of which we will examine two main ones.² With our basic layers as defined above, we will show that a binary-branching analysis is preferable for annotating learner language, in that it minimizes the number of mismatches between subcategorization and realization.

4.1 Basic coordination

In the learner example (10), two arguments (of *about*) are conjoined. One treatment of this is with the conjunction as the head, as in figure 5,³ while an alternate view is to have a branching structure, as in figure 6.⁴ We will use these two treatments of coordination throughout, in order to illustrate what

²If one allows for limited amounts of constituency, there are even more ways to treat coordination (cf. Hudson, 1990).

³We often abbreviate: C=COORD, S=SUBJ, O=OBJ.

⁴Branching could go in either direction; while we choose right-branching, nothing hinges on this.

needs to be captured for learner language; these are also the main analyses considered for parsing (Kübler et al., 2009). The conjunction-as-head analysis treats coordination as involving some degree of a “phrase,” whereas right-branching treats the conjuncts independently.

(10) The story about a tin toy and a baby .

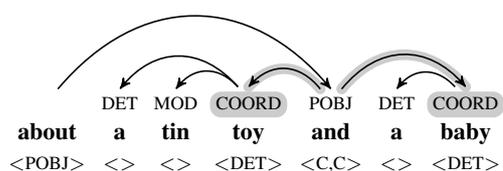


Figure 5: Conjunction-as-head coordination

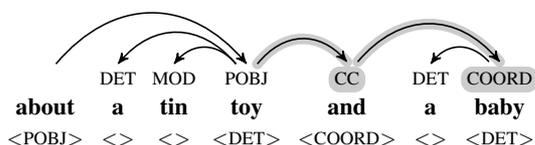


Figure 6: Right-branching coordination

For either analysis, we must consider how subcategorization interacts with the dependencies. In this case, it must be clear that *about*—which selects for a prepositional object (POBJ)—actually realizes it. Both analyses meet this requirement.

Additionally, we need to consider how subcategorization should be handled for the conjunction itself. A learner could potentially use a conjunction like *and* without one of its conjuncts. Thus, it should select for at least one coordinating element. In figure 5, this is done by *and* selecting for two COORD elements, while in figure 6, it selects for one element, as only one conjunct is realized at a time. The CC relation is not selected for, consistent with the fact that the head of *and* is not required to have a conjoined phrase.⁵

For the moment, we are simplifying the dependency graphs; in section 4.3, we will discuss the need to further articulate the COORD labels. In this case, we will have <COORD-POBJ> in the branching analysis, i.e., passing down the POBJ requirement from the head of *and* onto *and* itself.

⁵Another branching analysis has the conjunct be a dependent of the second noun (*baby*) (e.g., Buch-Kromann, 2009). While selection works differently, our general points about branching analyses should apply.

Saturated functors For the coordination of functors—i.e., words selecting for arguments—these can be treated on a par with basic argument coordination if they have realized all their requirements. Looking at the coordination of sentences in (11), for example, both *found* and *hid* are functors, but are saturated when they coordinate. Thus, the treatment of coordination is the same as before (trees not shown for space reasons).

(11) the tin toy **found** the very safety place where he should hide , and he **hid** under a sofar .

4.2 Coordination of unsaturated functors

Consider now the case where two unsaturated elements are coordinated, i.e., both words are still looking for an argument. In (12), for example, *walk* and *run* both have the same subject. The trees in figures 7 and 8 show that *He* is the subject of *begins*, with *walk* and *run* having a secondary connection to it. For this sentence, there is not a great difference between the two different analyses, in terms of connecting dependencies and subcategorizations. If the sentence were *He walks and runs*, however, then *and* would take *He* as a SUBJ for the conjunction-as-head analysis and thus also explicitly include SUBJ on its subcategorization; we take this issue up in the next section.

(12) He begins to walk and at to run .

As a side point, note in this example that *at* has an empty subcategorization list because we cannot determine what it is distributionally. For the morphologically-defined tree (see section 3.2.3), the subcategorization for *at* would be <POBJ> without a POBJ being realized.

Right node raising Moving from a fairly straightforward analysis of shared subjects, let us now consider the more challenging shared object between conjuncts, as in the constructed example (13), a case of right node raising (cf. Ross, 1967).⁶

(13) He begins to walk and to run the race.

Trees for this example are presented in figures 9 and 10. In both cases, the analyses are relatively theory-neutral, in that they do not state anything explicitly about how the object came to be shared between these verbs (see section 3.3).

⁶Most of the remaining examples in the paper are constructed, due to these types of coordination not having been observed in our data thus far.

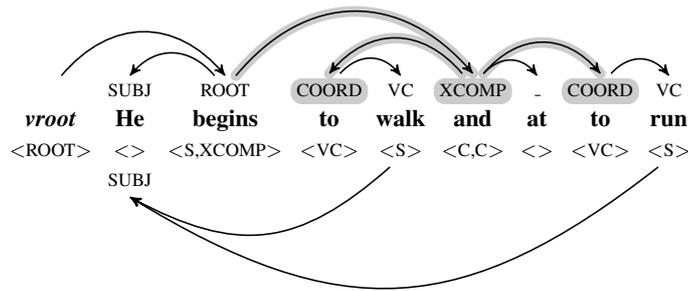


Figure 7: Functor coordination, where functors are unsaturated (conjunction-as-head)

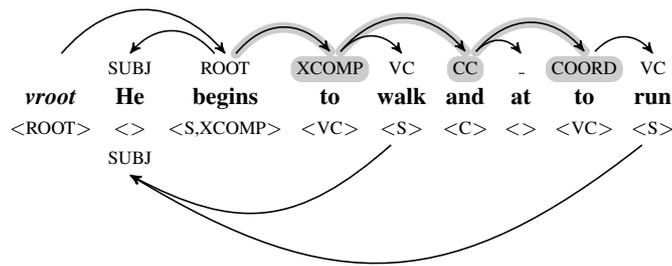


Figure 8: Functor coordination, where functors are unsaturated (right-branching)

What is noticeable in comparing the figures is the extra secondary dependency in the conjunction-as-head analysis. Recall that part of our goal is to accurately encode whether a learner’s sentence obeys completeness, coherence, and consistency. With *and* as the head of the coordinate structure, it must have the object as its dependent and must thus have the object on its subcategorization list. This means that all three words (*walk*, *and*, *run*) have the same object in their subcategorization.

Consider now if there were to be an error in consistency, as in the constructed example (14), where the verbs expect OBJ, but instead find the prepositional IOBJ. There are now 3 mismatches, as *bakes*, *eats*, and *and* all have the same OBJ subcategorization requirement. In general, the conjunction-as-head analysis reduplicates dependency requirements, leading to more mismatches.

(14) He bakes and eats **to** the cookies.

In the branching analysis in figure 10, on the other hand, only the verbs have the object requirement listed in their subcategorization, and the number of secondary dependencies is reduced from 4 to 3. To handle (14), there would be only two mismatches, one for each verb. As we argue below, this is desirable, as each verb can have its

own separate requirements.

Note that we are not claiming that the branching analysis is better theoretically. We are claiming that it is a simpler way to annotate learner language, especially as it posits fewer errors.

Functor coordination with different requirements

Consider an example of right node raising where there are slightly different verbal requirements. In the constructed example (15), for instance, *is fond of* selects for a prepositional object (POBJ), while *buys* selects for an object.

(15) She is fond of and buys toys.

In figures 11 and 12, this is partly handled by the (secondary) dependencies between *of* and *toys*, on the one hand, and between *buys* and *toys*, on the other. The relation is POBJ in the former cases, and OBJ in the latter. Whether primary or secondary, each relation has a unique label.

The issue is in the label between *and* and *toys* in the conjunction-as-head analysis (figure 11): should it be POBJ or OBJ? We can posit a category hierarchy (e.g., POBJ as a subtype of OBJ) or an intersection of categories (e.g., OBJ+POBJ), but this requires additional machinery. The branching analysis (figure 12) requires nothing extra, as no extra relations are used, only those between the

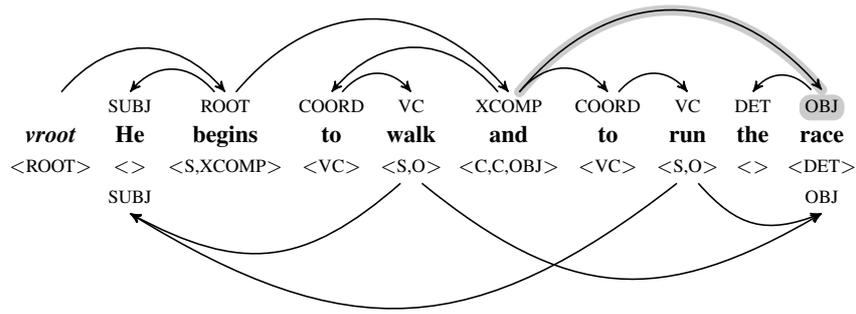


Figure 9: Functor coordination, with right node raising (conjunction-as-head)

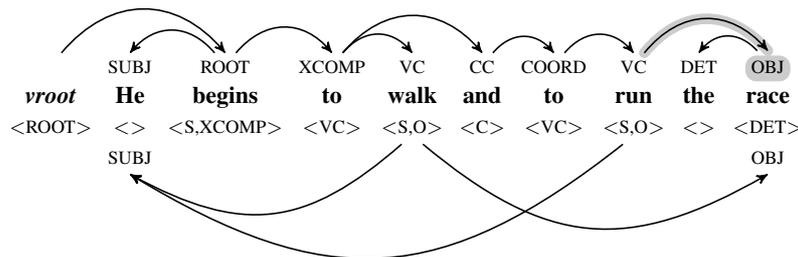


Figure 10: Functor coordination, with right node raising (right-branching)

functors and *to*s. This independent treatment of verbs also means that if verb saturation differs, the conjunction does not have to represent this, as in the learner example (16), where *run* is saturated and *stumbled over* is not (missing POBJ).

(16) ... it **run** after him and **stumbled over** and began to cry.

4.3 Coordination of unlikes

One difficulty that arises in annotating coordination is in how we annotate the coordination of unlike elements. Coordination of unlikes is well-known (Sag, 2003; Sag et al., 1985), though when we refer to the coordination of unlike elements, we are referring to elements which have different dependency relations. For instance, (17) features a coordination of an adjective and a noun phrase. But, in terms of their dependencies, they are both predicatives, so their dependency will be the same (PRED), as our dependency inventory does not distinguish adjectival from nominal predicatives.

(17) Pat is [wealthy and a Republican]. [AP & NP] (Sag et al., 1985)

The kind of case we are concerned about occurs in the constructed example (18), where we have a

non-finite and a finite verb conjoined.⁷ Because learners can head a sentence with a non-finite verb (e.g., *to apparer a baby*) or no verb at all (e.g., *the baby down* in (5)), we distinguish finite ROOT relations from non-finite ROOT-nf. In (18), then, we have one conjunct (*running*) which should be ROOT-nf and one (*eats*) which should be ROOT.

(18) He running and eats.

Walking through figures 13 and 14, we first consider the label on the arc between *and* and its head. For the conjunction-as-head analysis, we need to indicate that the whole *and* phrase is not consistent. This is essentially the same issue we saw with OBJ+POBJ; in this case, we need to annotate the label as ROOT+ROOT-nf or use a hierarchy. This makes the connection to the subcategorization list transparent: *vroot* looks for ROOT, but finds both ROOT and ROOT-nf. The branching structure, on the other hand, only takes the first conjunct is its dependent. Thus, if *running* comes first—as it does in figure 14—its label is ROOT-nf; if *eats* were first, the label would be ROOT.

⁷We have an attested example of unlike coordination in *I want to make happy and love and nice family*, but use the simpler (18) to explain our approach; the points are similar.

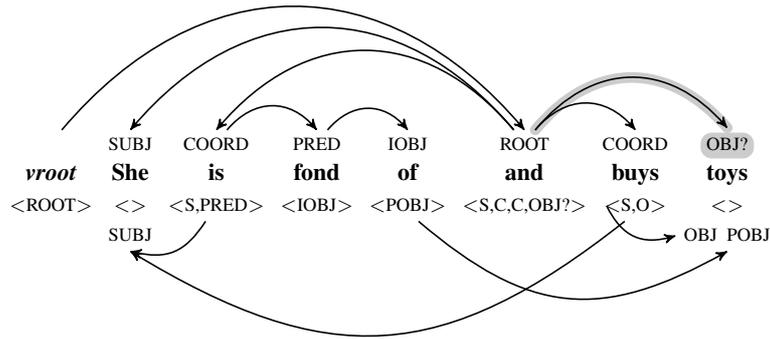


Figure 11: Coordination between two elements with different requirements (conjunction-as-head)

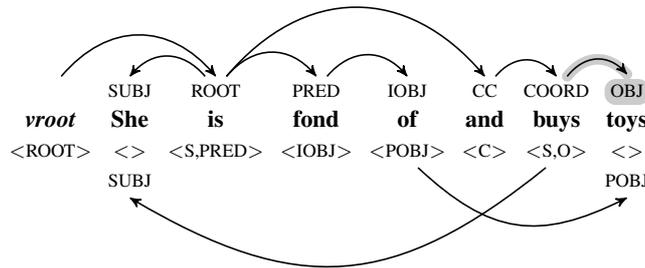


Figure 12: Coordination between two elements with different requirements (right-branching)

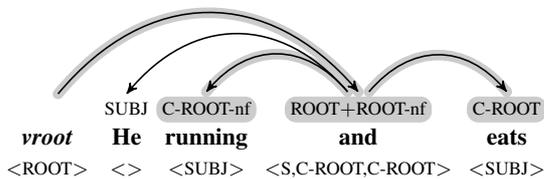


Figure 13: Coordination of unlikes; secondary dependencies not shown (conjunction-as-head)

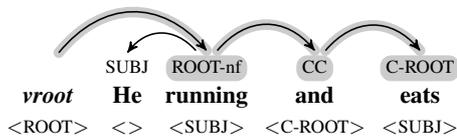


Figure 14: Coordination of unlikes; secondary dependencies not shown (right-branching)

Secondly, there is the relation between *and* and its dependents. To determine which conjunct is finite and which non-finite for the conjunction-as-head analysis and to exactly pinpoint the inconsistency, we augment the COORD labels. COORD only tells us that the element is a coordinating element, but does not tell us if the word is functioning

as a subject, a verbal complex, etc. Incorporating the actual relation, we create COORD-ROOT and COORD-ROOT-nf labels in this case.

For subcategorization, the requirements of the head of *and* (the virtual root *vroot*) are passed down to *and* and added to its conjunct requirements. Thus, in figure 13, *and* selects for two COORD-ROOT elements: COORD because it is a conjunction, and ROOT because its head selects for a ROOT. Thus, in the case of *running*, we identify a mismatch between the selected-for COORD-ROOT and the realized COORD-ROOT-nf.

For the branching analysis in figure 14, we also use COORD-ROOT. If the sentence were *He eats and running*, we would want to know that *and* selects for COORD-ROOT, but realizes COORD-ROOT-nf (*running*). Though not indicated in previous figures, this applies for all the trees in this paper, to ensure that requirements can be checked.

Again, the conjunction-as-head analysis is more complicated to annotate: in figure 13, there are two mismatches—between the subcategorization and realization for *vroot* and also for *and*—for what is only one issue. And unlike the use of ROOT+ROOT-nf, with the branching analysis

there is no confusion about the problem's source.

5 Summary and Outlook

We have outlined a way of annotating dependencies for learner language, relying upon a division of labor between basic dependencies, secondary dependencies to capture long-distance relations, and subcategorization marking for every word. Comparing two different exemplar analyses of coordination, we illustrated why a branching analysis is preferable over one which duplicates information, in terms of keeping annotation simple and allowing one to find mismatches between annotation layers. We are attempting to maintain a relatively simple annotation scheme, but as coordination illustrates, even this can become complex.

This treatment handles the cases of coordination we have observed so far, and in this paper we covered the main constructions we expect to see in learner language. A few other cases need to be fully borne out in the future, however, including cases of missing conjunctions and of non-constituent coordination (Steedman and Baldrige, 2011). For missing conjunctions, one would have to use a non-conjunction head, i.e., one of the conjuncts, in the conjunction-as-head analysis (e.g., Sagae et al., 2010, p. 716), while for the right-branching analysis, there has to be a direct link between conjuncts. This means a CC relation will not have a conjunction as its dependent. Working out the details requires a fuller treatment of modification, but neither case seems to supersede our proposal.

The annotation effort is still relatively new, and we are beginning to move out of the pilot phase. With the different layers in place, we are currently investigating inter-annotator agreement.

Acknowledgments

We thank Detmar Meurers for discussion and four anonymous reviewers for their helpful feedback.

References

Kathleen Bardovi-Harlig. 1999. Examining the role of text type in L2 tense-aspect research: Broadening our horizons. In *Proceedings of the Third Pacific Second Language Research Forum*, volume 1, pages 129–138. Tokyo.

Joan Bresnan. 2001. *Lexical-Functional Syntax*. Blackwell Publishing, Oxford.

Matthias Buch-Kromann. 2009. *Discontinuous Grammar. A dependency-based model of human parsing and language learning*. VDM Verlag.

Sabine Buchholz and Erwin Marsi. 2006. Conll-x shared task on multilingual dependency parsing. In *Proceedings of CoNLL-X*, pages 149–164. New York City.

Ralph Debusmann, Denys Duchier, and Geert-Jan M. Kruijff. 2004. Extensible dependency grammar: A new methodology. In *Proceedings of the COLING 2004 Workshop on Recent Advances in Dependency Grammar*. Geneva/SUI.

José Deulofeu, Lucie Duffort, Kim Gerdes, Sylvain Kahane, and Paola Pietrandrea. 2010. Depends on what the french say - spoken corpus annotation with and beyond syntactic functions. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 274–281. Uppsala.

Ana Díaz Negrillo and Jesús Fernández Domínguez. 2006. Error tagging systems for learner corpora. *Revista Española de Lingüística Aplicada (RESLA)*, 19:83–102.

Ana Díaz Negrillo, Detmar Meurers, Salvador Valera, and Holger Wunsch. 2010. Towards interlanguage POS annotation for effective learner corpora in sla and flt. *Language Forum*, 36(1–2).

Markus Dickinson and Marwa Ragheb. 2009. Dependency annotation for learner corpora. In *Proceedings of the TLT-8*. Milan, Italy.

Sylviane Granger. 2003. Error-tagged learner corpora and CALL: A promising synergy. *CALICO Journal*, 20(3):465–480.

Hagen Hirschmann, Anke Lüdeling, Ines Rehbein, Marc Reznicek, and Amir Zeldes. 2010. Syntactic overuse and underuse: A study of a parsed learner corpus and its target hypothesis. Talk given at the Ninth Workshop on Treebanks and Linguistic Theory.

Richard A. Hudson. 1990. *English Word Grammar*. Blackwell, Oxford, UK.

Alan Juffs. 2005. The influence of first language on the processing of wh-movement in English as a second language. *Second Language Research*, 21(2):121–151.

Matthias Trautner Kromann. 2003. The danish dependency treebank and the underlying linguistic theory. In *Proceedings of TLT-03*. Växjö, Sweden.

- Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. In Graeme Hirst, editor, *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers.
- Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL.
- Anke Lüdeling, Maik Walter, Emil Kroymann, and Peter Adolphs. 2005. Multi-level error annotation in learner corpora. In *Proceedings of Corpus Linguistics*. Birmingham.
- Igor Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press.
- Diane Nicholls. 2003. The Cambridge Learner Corpus - error coding and analysis for lexicography and ELT. In *Proceedings of the Corpus Linguistics 2003 Conference (CL 2003)*, pages 572–581. Lancaster University.
- Joakim Nivre. 2005. Dependency grammar and dependency parsing. MSI report 05133, Växjö University: School of Mathematics and Systems Engineering.
- Timothy Osborne. 2008. Major constituents and two dependency grammar constraints on sharing in coordination. *Linguistics*, 46(6):1109–1165.
- Niels Ott and Ramon Ziai. 2010. Evaluating dependency parsing performance on German learner language. In *Proceedings of TLT-9*, volume 9, pages 175–186.
- Nick Pendar and Carol Chapelle. 2008. Investigating the promise of learner corpora: Methodological issues. *CALICO Journal*, 25(2):189–206.
- Manfred Pienemann. 1992. Coala—a computational system for interlanguage analysis. *Second Language Research*, 8(1):58–92.
- Manfred Pienemann. 1998. *Language Processing and Second Language Development: Processability Theory*. John Benjamins, Amsterdam.
- Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. The University of Chicago Press.
- Marwa Ragheb and Markus Dickinson. to appear. Avoiding the comparative fallacy in the annotation of learner corpora. In *Second Language Research Forum Conference Proceedings*. Cascadilla Proceedings Project, Somerville, MA.
- Stefano Rastelli. 2009. Learner corpora without error tagging. *Linguistik online*.
- John Robert Ross. 1967. *Constraints on Variables in Syntax*. Ph.D. thesis, MIT.
- Alla Rozovskaya and Dan Roth. 2010. Annotating ESL errors: Challenges and rewards. In *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 28–36. Los Angeles, CA.
- Ivan Sag, Gerald Gazdar, Thomas Wasow, and Steven Weisler. 1985. Coordination and how to distinguish categories. *Natural Language and Linguistic Theory*, 3:117–171.
- Ivan A. Sag. 2003. Coordination and underspecification. In *Proceedings of the Ninth International Conference on HPSG*. CSLI Publications, Stanford.
- Kenji Sagae, Eric Davis, Alon Lavie, and Brian MacWhinney and Shuly Wintner. 2010. Morphosyntactic annotation of CHILDES transcripts. *Journal of Child Language*, 37(3):705–729.
- Kenji Sagae, Eric Davis, Alon Lavie, Brian MacWhinney, and Shuly Wintner. 2007. High-accuracy annotation and parsing of CHILDES transcripts. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 25–32. Prague.
- Geoffrey Sampson. 1995. *English for the Computer: The SUSANNE Corpus and Analytic Scheme*. Clarendon Press, Oxford.
- Petr Sgall, Jarmila Panevová, and Eva Hajičová. 2004. Deep syntactic annotation: Tectogrammatical representation and beyond. In *Proceedings of the Workshop on Frontiers in Corpus Annotation*, pages 32–38. Boston.
- Mark Steedman and Jason Baldridge. 2011. Combinatory categorial grammar. In Robert Borsley and Kersti Borjars, editors, *Non-Transformational Syntax: Formal and Explicit Models of Grammar*. Wiley-Blackwell.
- Kate Wolfe-Quintero. 1992. Learnability and the acquisition of extraction in relative clauses and wh questions. *Studies in Second Language Acquisition*, 14:39–70.

The Dependency Distance Hypothesis for bilingual code-switching

Eva Duran Eppler
Roehampton University, London
e.eppler@roehampton.ac.uk

Abstract

This paper addresses the questions why and where, i.e. in which syntactic dependency relations, multilingual speakers are likely to code-switch. Code-switching (CS) is the linguistic behaviour of producing or comprehending language which is composed from lexical items and grammatical structures from two (or more) languages. This paper proposes that code-switching is more likely in syntactic relations with long dependency distances (Distance hypothesis DH). Dependency distance is the number of words intervening between a head and a depended. The DH is tested on a 93,235 word corpus of German/English monolingual and code-mixed discourse analyzed in Word Grammar (WG). This data set supports the DH in general and on specific syntactic functions. In ongoing work the DH is being tested on Welsh/English and Spanish/English corpora and with self-paced silent reading experiments using eye-tracking.

1 Introduction

This paper suggests that a property of dependency structures, i.e. dependency distance, accounts in part for syntactic code-switching. The idea that long dependency distances facilitate code-switching is an original proposal and can therefore only be indirectly linked to existing theories of code-switching.

The concept of dependency distance was first used in Heringer et al. (1980: 187) who call it 'Abstand'; the term 'dependency distance' was introduced in Hudson (1995: 16) who defines it as 'the linear distance between words and their heads, measured in terms of intervening words'. For an illustration of individual and mean distances see Figure 1

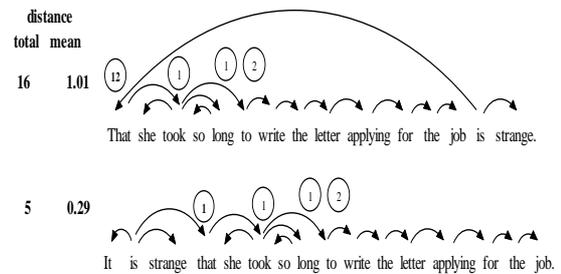


Figure 1 Dependency Distance

Mean dependency distances are cross-linguistically different (Liu 2008). In English most words are next to the word on which they depend (Collins 1996, Pake 1998). The main factor increasing distance is a change in dependency direction, i.e. a combination of left- and right-dependents (Termperley 2008, Hudson, personal communication).

Distance is an important property of a dependency relation because of its implications for the cost of processing the syntactic relation. Distance has been shown to be associated with syntactic memory load (keeping track of incomplete dependencies / obligatory syntactic requirements) and parsing complexity / integration cost (connecting a word into sentence structure) (Gibson 1998, Hiranuma 1999, Liu 2008). In terms of integration cost of long-distance dependencies, Gibson's (1998) Dependency Locality Theory (DLT) proposes that the longer a predicted category must be kept in memory before the prediction is satisfied, the greater the cost for maintaining that prediction. The greater the distance between an incoming word and the most local head or dependent to which it attaches, the greater the integration cost. In other words, the structural integration complexity depends on the distance between the two elements being integrated. That average dependency distance of a sentence can be used as a measure for its parsing complexity has been shown for centre-embedded vs. right-dependent sentences, subject vs. object relative

clauses¹ and Garden Path² sentences (Liu 2008).

Hiranuma (1999) demonstrated for Japanese and English that memory load increases with the length of a dependency, measured in terms of the number of intervening words. Liu et al. (2009) show that Chinese has a considerably longer mean dependency distances than other languages and propose that this may make Chinese more difficult to process than other languages. The average dependency distance of a text is an important comparative measure and can shed light on the cognitive demands of the language concerned relative to other languages (Liu 2008).

The present paper investigates the effects of dependency distance on syntactic code-switching, a linguistic phenomenon for which classical phrase-structure based models have proven to be unsatisfactory because over-generating (Eppler 0026).

2 The data

The study is based on a 93,235 words corpus of German-English monolingual and code-mixed discourse. The data is drawn from a community of Austrian Jewish refugees from the National Socialist regime who settled in London in the late 1930s. The L1 of the informants is Austrian German. The age of onset of the L2, British English, was during adolescence (15 - 21 years) for all speakers included in this study. At the time the audio-recordings were made (1993) all informants were in their late sixties or early seventies. A bilingual mode of interaction called 'Emigranto' developed among a close-knit network of community members. Linguistically the mixed code is characterized by frequent switching at speaker turn boundaries and heavy intra-sentential code-switching.

3 Dependency distance in English and German

English is generally considered to be a head-first language and allows for relatively little word order variation. As a consequence we get

¹ Processing cost of subject vs. object extraction, however, seem to be cross-linguistically different in English and German (Jackson and Dussias 2009)

² Late Closure (Frazier 1978) is preferred by the parser because it tends to minimize average dependency distance.

few changes in dependency direction and short dependency distances. 63 – 74% (Collins 1996 and Pake 1998 respectively) of English words that are syntactically related are also adjacent i.e. they have a distance of 0.

The mean distance between two syntactically related German words is hypothesised to be longer than the mean distance between two related English words. The main reasons why I assume German to have a longer mean distance are,

- the generally freer word order in German, which allows for more changes in dependency direction which trigger longer distances
- scrambling, i.e. word order variation of argument noun phrases with respect to each other (Example 1a & b), and/ or with respect to adverbial phrases (Examples 2) or even with respect to subjects (Example 3)

(1a) *Er hat ihr dieses Buch vielleicht gegeben.*

³%glo: he has her this book maybe given

(1b) *Er hat dieses Buch vielleicht ihr geben.*

%glo: he has this book maybe her given

(2) *Er hat ihr vielleicht dieses Buch gegeben*

%glo: he has her maybe this book given

(3) *dass jeder den Geruch erkennt*

%glo: that everybody this smell recognises

- the Verbalklammer, i.e. the discontinuity between AUX/MOD and main verbs
- different word orders in German main (V2) and subordinate clauses (V final or SOV).

4 Dependency distance in 'mixed' dependencies

'Mixed' dependencies are syntactic relations in which words A and B are from different languages. For mixed dependencies the main point of interest will be whether greater dependency distance influences / affects the chances of code-mixing. If code-switching is found to cause extra processing load, we might either expect

- shorter distances in mixed dependencies, because they 'counteract' the processing cost that is associated with code-switching (for some speakers)

³ CHILDES / LIDES transcription conventions are used throughout the paper
<http://childes.psy.cmu.edu/manuals/chat.pdf>
<http://www.ling.lancs.ac.uk/staff/mark/lipps/easyliides.htm>

- a dependency distance between the mean distances for German and English monolingual dependencies, because syntactic dependency properties of both languages are involved

- longer mixed dependency distances, if we assume that the influence of a word's language on that of its dependent will decrease with increased distance. In other words, the longer the distance, the more likely we are to encounter an other language dependent, i.e. a code-switch. The latter assumption is similar to Gibson's computational motivation for the DLH. In an activation-based framework like WG, the head's activation will decay with distance (because of the limited quantity of activation in the system). The process of structural integration therefore involves reactivating the head to a target threshold level so that aspects of the head can be retrieved from memory. This reactivation is not only costly, but may also be incomplete.

For mixed dependency relations I am going to work on the hypothesis that the distance of mixed dependencies with a German head should be longer than the distance of mixed dependencies with an English head. This is based on the assumption that monolingual German dependencies are longer than English ones, and the possibility that heads influence dependency distance more than dependents. Furthermore, a change in dependency direction should be more frequent in mixed dependency relations with a German head, because verbs are main heads and they are involved in construction types like the Verbalklammer and V2 placement in main clauses and SOV placement in subordinate clauses.

The calculation of the mean distances in monolingual and mixed dependencies will reveal if these ideas are supported by the Emigranto data or not. The results on mean distances together with the standard deviation from the mean are presented in Table 1 Section 5.

5 General findings

Table 1 presents the mean dependency distances for monolingual German, monolingual English and mixed dependencies with German and English heads respectively.

	German	English	Average
Monolingual	0.87 ($\sigma=0.78$)	0.49 ($\sigma=0.41$)	0.68
Mixed with head	0.85 ($\sigma=0.81$)	1.26 ($\sigma=1.08$)	1.06

Table 1. Mean distances (and σ) in monolingual and mixed dependencies

These numbers tie in with pervious findings about dependency distance and the hypotheses formulated in Sections 1-4 as follows: Table 1 shows that:

1. monolingual German dependencies are longer than English ones. This supports the hypothesis made on the basis of the word order properties of the two languages (Section 3);
2. the mean distance of mixed dependencies with a German head is marginally shorter than the mean distance of monolingual German dependencies. This difference is unexpected but too small to support the idea that mixed dependencies counter-balance a potentially greater processing load for mixed utterances with a shorter dependency distance. This finding may, however, indicate that the word class that is assumed to increase dependency distance through a change in dependency direction, i.e. German verbal heads, is infrequently involved in mixed dependencies. Most importantly, however, it suggests that German words do not seem to affect the distance to their dependent, i.e. at least in terms of distance, they behave similarly in monolingual and mixed syntactic relations.
3. the mean distance of mixed dependencies with an English head is much longer than the mean distance of monolingual English dependencies. English heads thus seem to enter into 'looser', literally more remote, syntactic relations with German dependents. We would then expect English words to 'head' more dependency relations that are characterised by long distances, e.g. adjunct, extractee and extraposee relations. And we would expect German dependents of English heads to be more frequently located at the clause periphery. If we found more mixed dependents at the clause periphery in the Emigranto data, this would tie in nicely with the literature

on code-switching. Treffers-Daller (1994) first noted a high propensity of switching for ‘dislocated constituents’ in her French / Dutch data. Muysken (2000) subsequently adopted the idea that code-switching is favoured in peripheral positions as one of four primitives of code-switching.

4. the mean distance in mixed dependencies with a German head is approximately two thirds of the mean distance of mixed dependencies with an English head. This last finding completely contradicts the assumption that mixed dependencies with German heads are longer than mixed dependencies with English heads. This idea was based on the assumption that heads determine more characteristics of dependency relations than dependents, including the linear distance between them measured in terms of the number of words from one to the other.
5. the difference in mean distances between monolingual and mixed dependencies is highly significant ($X^2 = 18.6$, $df = 1$, $p < 0.001$);
6. The mean distance of mixed dependencies (1.06) is longer than that of both English and German monolingual dependencies. This finding supports the third possibility outlined above, i.e. that more distant words may have less influence on each other’s language, because of the decay in activation as intervening words are processed and integrated into the structure of the input. If we assume that the influence of a word’s language on that of its dependent decreases with increased distance, mixed dependencies may be the result of distance. By their very nature long distance dependencies in SVO and V2 languages are more likely to be located at the clause periphery. Treffers-Daller (1994) and Muysken (2000: 25) have both proposed peripherality as a factor favouring code-mixing.
7. and the standard deviation from the mean is higher for mixed dependencies. In other words, there is more variation in the distances of mixed dependencies and there are more mixed outliers.

These findings seem to suggest that the syntactic relations German heads enter with English dependents are not very different to the ones they enter with same language

dependents, at least as far as distance is concerned. English heads, on the other hand, may enter into ‘looser’ and – literally – more remote (e.g. adjunct, extractee, extraposee) syntactic relations with German dependents. As a consequence, English words may function more frequently as heads of syntactic material that is located at the clause periphery.

The long dependency distances of mixed syntactic relations may furthermore point towards a processing motivation behind code-switching: the influence of a word’s language on that of its dependent may decrease with increased distance. This would then mean that the longer the dependency distance, the more likely we are to encounter an other language dependent, i.e. a code-switch. This assumption, in combination with the findings presented in Table 1 discussed above, has led to the formulation of a claim about bilingual language use which combines features of grammar (syntactic relations) and psycholinguistics processes of speech production (dependency distance), the Distance Hypothesis.

Greater dependency distance of syntactic relations increases the chances of code-mixing.
(Eppler 2005)

The Distance Hypothesis is a syntactic processing hypothesis. Evidence in its support would therefore potentially shed light on both grammatical and psycholinguistics aspects of code-switching.

6 Specific findings

The analysis of individual syntactic functions in the Emigranto corpus statistically supports some of the constraints on code-switching proposed in the literature, but not others. The findings, for example, support the equivalence (Poplack 1980) and the subcategorization constraints (Bentahila and Davies 1983) in a probabilistic way. Both of these constraints are similar to the null hypothesis Eppler (2010) is based on, i.e. that each word in a syntactic dependency relation must satisfy the constraints imposed on it by its own language. The Complement Adjunct distinction (Mahootian and Santorini 1996: 470), on the other hand, is not supported.

The syntactic analysis of the Emigranto corpus moreover confirms that some syntactic functions are more easily switched than others.

Noun complements of determiners, for example, are clearly at the top of the ‘borrowability’ hierarchy; objects are more easily switched than subjects; and syntactically unrelated utterance elements are at the top of the ‘switchability’ hierarchy.

In the following three sub-sections, I will focus on the dependency distances of individual syntactic relations, comparing monolingual types with each other (Section 6.1), monolingual German ones with mixed ones with a German head (Section 6.2), and monolingual English ones with mixed ones with an English head (Section 6.3).

6.1 Monolingual German and monolingual English syntactic functions

Comparing monolingual German and English grammatical functions with each other on the one hand shows that German and English are typologically similar (they can be analysed with the same set of dependency types), but also reveals the main word order differences between the two languages. Sharers⁴, objects, negatives, particles and prepositional are exclusively right-dependency relations of verbs in English. In German, sharers, objects, negatives particles and prepositionals of V2 verbs are also right-dependents, while they are left-dependents of clause final verbs. These results furthermore indicate that the German/English bilinguals possess two identifiable linguistic systems or languages, each with its identifiable grammatical rules and lexicon.

In Section 3 I outlined why I expect a longer mean distance for German dependency relations than for English ones, and we found this assumption confirmed by the data (Tables 1 and 2). The mean distance between two syntactically related German words is 0.87 in the Emigranto data; the mean distance between two syntactically related English words, on the other hand, is only 0.49. This is approximately 0.1 longer than what Hiranuma (1999) found for a comparable (conversational) corpus of 1,035 words, and closer to the 0.51 Liu et al. (2009) calculated for a written English sample text of about 100 words. Table 2 (Appendix B), however, shows that those monolingual English syntactic functions from the Emigranto

corpus that yield substantial enough a number of tokens to be included in the more fine grained analysis, have a mean distance of 0.4 and are therefore very close to Hiranuma’s 0.386.

Tables 1 and 2 (Appendix B) therefore confirm that mean dependency distances differ cross-linguistically and that different dependency types have different mean distances (cf. Liu, Hudson and Feng 2009: 170). Table 2 furthermore reveals which grammatical functions differ most between German and English in terms of dependency distances. They are complements, subjects, sharers, objects and especially extractees.

The clause final placement of German finite verbs depending on complementizers could cause the longer mean distance of German complements: the head and the complement are literally at opposite ends of the subordinate clause. Subordination, however, tends not to be frequent enough in spoken language corpora to have this big an effect. The longer mean distance of German complements can be traced back to the same reason why we have significantly more German pre-adjectives than English ones, i.e. attribute adjectives between determiners and nouns. The distance of German subjects (in their ‘normal’ position, i.e. as left-dependents) from their head verbs also deserves comment. The following two word order properties of German cause the, in comparison with English, longer mean distance. Firstly, the subjects of clause final finite verbs in subordinate clauses are almost at opposite ends of clauses. Secondly, material (e.g. adverbs) intervening between subjects and the verb that functions as the head / root of the sentences increases the distance of German subjects. Given that the Emigranto corpus contains a lot of Verbalklammern (because reference to past time is made with the present perfect rather than the simple past in spoken German), I find the mean distance for German sharers (1.64) relatively short, although it is of course three times longer than that of English sharers. The, for standard German, ungrammatically extraposed objects in the Emigranto corpus⁵ shorten the mean distance of monolingual German sharers.

⁴ ‘Sharer’ is a kind of verb complement. In other syntactic theories sharers are called xcomps or predicatives.

⁵ Sub-stratum influence from Yiddish has rendered examples like these marginally acceptable in the Viennese dialect.

(4) hat mich gekostet zwei pfund zwanzig [*].
 %glo: has me cost two pounds twenty
 Jen1.cha, line 465

The mean distance for German sharers may also be a result of the monolingual German data containing more sharer relations between verbs and predicative adjectives than between auxiliaries / modals and non-finite verbs. Adjectives intervening between objects and their head verbs give rise to the longer mean distance of German object dependencies. The biggest difference in the mean distances between monolingual German and English dependencies, however, clearly lies in the extractees. An example that illustrates the ‘damaging’ effect of extraction (and the word order in subordinate clauses) on the mean distance of monolingual German extractees is

(4)
 *MEL: aber wenn man einen huegel
 hinauf#gehen muss -, das ist schon +...
 %tra: and if one must walk up a hill, then
 that is already +...

Jen1.cha, line 447

Example (4) is a fragment, but the relevant syntactic relations are there and the extractee *wenn*, is six words away from its head, the main clause finite verb *ist*; the complement of the extractee (*muss*) is four words away from it; and the subordinate clause’s subject (*man*) is three words away from its head. In extracted subordinate clauses that are not ‘small’ clauses, we get three changes in dependency direction between the words that build the basic syntactic structure of these clauses. This naturally increases distance.⁶

6.2 Monolingual German and mixed syntactic functions with a German head

Out of the most common syntactic relations (complements, subjects, adjuncts, sharers and objects), three show a significant difference between how often they occur monolingually and how often they enter mixed dependency relations with a German head in the Emigranto corpus. They are complements, subjects and adjuncts. The frequently switched

complements are borrowed English nouns; subjects are infrequently switched, particularly subject pronouns like (5)

(5)
 *LIL: **you** kannst # jauchzen .
 %tra: you can # rejoice

Jen2.cha, line 1019

despite linguistic constraints on switching subjects (Gumperz and Hernandez-Chavez 1971 and Pfaff 1975); and adjuncts a very frequently switched.

Table 1 showed that the mean distance of mixed dependency relations with a German head is actually a bit shorter than the mean distance of monolingual German dependencies (0.85 to 0.87). Table 3 (Appendix B), however, reveals that the distances for most mixed grammatical functions (subjects, adjuncts, pre-dependent sharers and post-dependent objects) are longer than their monolingual German equivalents. The slightly shorter mean distance of mixed dependencies with a German head (in comparison with monolingual German dependencies) is only attributable to three dependency types: complements, post-dependent sharers and left-dependent objects. Out of these three, it is the very large number of English complements with a German head, the borrowed English nouns, that brings the mean distance down.

This result also tells us something about the syntactic structure of mixed complement relations with an English noun: they are hardly ever pre-modified. A lot of the English predicative adjectives are also very close to their German head; and so are the English objects that depend on German clause final / SOV verbs. The fact that English post-dependent adjuncts are almost three times as far away from their German head as monolingual post-dependent adjuncts seems to support Treffers-Daller (1994), Mahootian and Santorini (1996) and Muysken (2000), i.e. that code-mixing is favoured in adjoined peripheral positions.

(6)
 *MEL: nein # ich bin draussen # **as per usual**.
 %tra: no # I am out

Jen2.cha: line 185.

In Section 5 we hypothesised that the mean distance of mixed dependencies with a German head might be marginally shorter than the mean distance of monolingual German dependencies because the word class that is assumed to increase dependency distance

⁶ Dependency distance can be quantified in different ways. Gibson, for example, quantifies it in terms of new intervening discourse referents. According to this measure we would expend 5 energy units when processing sentence (5).

through a change in dependency direction, i.e. German verbal heads, is infrequently involved in mixed dependencies. An analysis of all German verbs in the Emigranto corpus revealed that this word class does function as roots/heads in mixed dependencies.

A separate test performed on verb types (main vs. AUX/MOD) showed that overall German verbs are not significantly less frequently involved in mixed dependencies than monolingual ones ($p=0.112$). The same holds true for German main verbs ($p=0.192$). German auxiliaries and modals, however, are significantly more frequently involved in mixed dependencies than English ones ($p=0.001$). This finding is interesting as AUX / MOD are frequently in V2 position, which often coincides with the placement of verbs in SVO structures. German AUX and MOD are therefore placed in congruence sites (Sebba 1998). Congruence sites / equivalent surface word orders have been identified as factors that facilitate code-switching (cf. Muysken's four primitives of code-switching).

6.3 Monolingual English and mixed grammatical functions with an English head

In the Emigranto corpus five syntactic functions occur significantly more or less frequently switched with an English head than with both the head and the dependent from the English language. They are - again - subjects and (pre-)adjuncts, as well as sharers, extrapositions and extractions.

As for German, the corpus yields disproportionately fewer German subjects depending on an English verbal head than monolingual English ones, but they do exist. See (7)

(7)
 *DOR: die **do-'nt mind ##** aber **I do** .
 %tra: they don't mind ### but I do
 jen1.cha: line 220.

The Emigranto data therefore provide probabilistic support for constraints on switching subjects or subject pronouns (Gumperz & Hernandez-Chavez 1971, Pfaff 1975).

Hardly any English verbs share their subjects with German words. This is unexpected and interesting for several reasons. For one, in this direction, i.e. $h_E \rightarrow d_G$, we do not encounter the conflict in dependency

direction we get in this syntactic relation with a German head (where the dependents can be both, right-dependents or a left-dependents of clause final verbs). We would therefore expect switching to be 'easier' in this direction. Second, Hawkins (1986: 96) notes that German is much more resistant to sharer/xcomp structures than English and puts this down to the generally increased semantic diversity of basic grammatical relations in English. For the code-switched German / English data this means that it seems to be the semantics of the German dependent that constrains code-switching, not the English head.

The pre-adjunct relation, on the other hand, is very popular for switching between an English head and a German dependent among the Jewish refugees.

(8)
 *LIL: die xx hat es # **in high heaven** gelobt.
 %glo: xx has it # praised
 Jen2.cha, line 1570

(9)
 *MEL: als kind **I didn't like anything**
 aber **I love food** .
 %tra: as a child I didn't like anything
 but I love food
 Jen2.cha, line 2058

Note that the pre-adjunct in (9) is also extracted, that is moved out of its default word order position and moved to the left clause periphery.

The difference between monolingual English and German extractees and extraposees depending on an English head is also highly significant. The next example illustrates a German long-distance (distance = 8) extraction.

(10)
 *MEL: was die Dorit wieder geschmissen hat,
I [/] I would have liked.
 %glo: what the Dorit again thrown has,
 It appears that for emphasis reasons speaker *MEL increases the distance of a mixed dependency relation from zero to eight in the above example.

The results presented in Table 4 (Appendix B), which compares the mean distances of monolingual English and mixed dependencies with an English head, strongly support the hypotheses formulated on the basis of Table 1 in Sections 1-5.

Hypothesis three proposes that English heads seem to enter into 'looser', literally more

remote, syntactic relations with German dependents. It furthermore predicts that we would expect English words to ‘head’ more dependency relations that are characterised by long distances, e.g. adjunct, extractee and extraposee relations. And we would expect German dependents of English heads to be more frequently located at the clause periphery (cf. Treffers-Daller 1994). This is exactly what we find in the data (see Table 4).

The Distance Hypothesis, states that greater distance seems to increase the chances of code-mixing. On the basis of Table 1 we assumed that the influence of a word’s language on that of its dependent may decrease with increased distance, and mixed dependencies would therefore be the result of distance. As a consequence of their long dependency distances code-switches were also expected to be more frequently located at the clause periphery. This is again what we find in the data (see Table 4).

Focusing on the mean distances of individual syntactic functions in Table 4 we notice that ALL mixed dependencies with an English head (apart from objects) are longer than their monolingual English counterparts (this is unlike the mean distances of monolingual German and mixed grammatical relations with a German head (h_G) (Table 3)). Table 4 furthermore illustrates that all dependency relations that yield a significantly higher number of mixed tokens than monolingual ones (German adjuncts, extractees), are further away from their English heads than their English counterparts. The results presented in Table 4 therefore lend support to the finding that code-mixing is favoured in peripheral and adjoined positions.

The hypothesis that greater dependency distance of syntactic relations increases the chances of code-mixing therefore appears to apply particularly to mixed syntactic relations with an English head. Mixed grammatical functions with an English head seem to pose a particular processing complexity for the German/English bilinguals. The activation of English heads seems to decay especially rapidly in long-distance dependencies and render retrieving aspects of h_E , e.g. it’s language, from memory particularly difficult. This appears to lead to the significantly larger number of mixed long distance syntactic relation with an English head in the Emigranto corpus.

7 Summary and Conclusion

The analysis of syntactic dependency relations in a 93,235 word corpus of German/English monolingual and code-mixed discourse analyzed in Word Grammar (WG) showed that the bilinguals possess two identifiable linguistic systems, each with its grammatical rules and lexicon. The code-switched speech results from the interaction between lexical elements and grammatical rules from these languages.

The data support some of the syntactic constraints on code-switching proposed in the literature in a probabilistic way: the constraint on switching subject (pronouns), the equivalence of structure (Poplack 1980) and the subcategorization constraints (Bentahila and Davies 1983). The Complement Adjunct distinction (Mahootian and Santorini 1996: 470), on the other hand, is not supported, not even if we analyse the English noun complements of German determiners discussed in Section 6.2 as borrowings.

The most interesting finding to emerge from the comparison of monolingual and mixed syntactic relations (Table 1) in the corpus is that mixed syntactic relations have a longer mean dependency distance than monolingual ones. This led to the formulation of the Distance Hypothesis and a set of corpus-specific hypotheses on the syntactic behaviour of linguistic elements in Emigranto. The central syntactic processing claim to emerge from the quantitative analysis is that the influence of a word’s language on that of its dependent appears to decay with the number of words intervening between it and its dependent. In other words, the longer the dependency distance, the more likely we are to encounter an other-language dependent, i.e. a code-switch.

The analysis of individual grammatical functions in Section 6 revealed that (with three exceptions) ALL mixed dependency relations are, on average, longer than the corresponding monolingual ones. In particular, the Emigranto corpus contains a considerable number of very long-distance mixed (post-)adjuncts with a German head, and English heads generally tend to enter into ‘looser’, literally more remote, syntactic relations with German dependents, i.e. syntactic relations that are not essential for building sentence structures, like adjunction, extraction (and extraposition). These

grammatical relations are also associated with long distances.

Including syntactically unrelated sentence element in the analysis, we can conclude that the ease with which elements are switched in the Emigranto corpus can be arranged on a continuum ranging from syntactic relations with very short dependency distances (such as subjects), to syntactically loosely connected grammatical functions with long dependency distances (such as adverbials, extractees and extraposees), to syntactically unrelated discourse elements at the utterance periphery (such as interjections, discourse markers and tags).

Dependency distance has been shown to play an important role in code-switching: syntactically related words are significantly more often in the same language when they are adjacent (Muysken's Adjacency Principle), and more distant words seem to have less influence on each other's language and are therefore more frequently switched. The Distance Hypothesis is an original proposal and can only be indirectly linked to existing theories of code-switching. It incorporates the idea that code-switching is favoured in peripheral (Treffers-Daller 1994, Muysken 2000) and adjoined positions but captures this notion on a more general syntactic processing level.

In collaboration with the Centre for Research on Bilingualism in Theory and Practice at the University of Wales, Bangor the Distance Hypothesis is being tested on other bilingual corpora (Spanish/ English and Welsh/English) and in self-paced silent reading studies supported with eye-tracking technology. In future work I also intend to investigate the effects of different kinds of 'interim' words, i.e. words intervening between the head and the dependent.

References

- Bentahila, Abdelali, and Eirlys E. Davies. 1983. The Syntax of Arabic - French Code - Switching. *Lingua* 59: 301 - 30.
- Collins, Michael John. 1996. A new statistical parser based on bigram lexical dependencies. *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics. University of California, Santa Cruz, 24-27 June 1996.* 184 - 191.
- Gumperz, John J., and Eduardo Hernandez-Chavez. 1971. Cognitive aspects of bilingual communication. Language use and social change. In: W. H. Whiteley (ed.). Oxford: Oxford University Press 111 - 25.
- Hawkins, John A. 1986. *A Comparative Typology of English and German. Unifying the Contrasts.* London and Sydney: Croom Helm.
- Eppler, Eva. 2003. German/English database <http://talkbank.org/data/LIDES/Eppler.zip>
- Eppler, Eva. 2006. Word Grammar and syntactic code-mixing research. In Sugayama, K. and Richard A. Hudson (eds.), *Studies in Word Grammar. New Perspectives on a Theory of Language Structure.* London, New York: Continuum, 117-39.
- Gibson, Edward. 1998. Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68, 1-76.
- Heringer, H.J., Strecker, B. and R. Wimmer. 1980. *Syntax: Fragen-Lösungen-Alternativen.* München: Wilhelm Fink Verlag.
- Hiranuma, So. 1999. Syntactic difficulty in English and Japanese: A textual study. *UCL Working Papers in Linguistics 11* (1999): 309-322.
- Hudson, Richard A. 2010. *An Introduction to Word Grammar.* Cambridge: Cambridge University Press.
- Hudson, Richard A. 2007. *Language Networks. The New Word Grammar.* Oxford: Oxford University Press.
- Liu, Haitao. 2008. Dependency distance as a metric of language comprehension difficulty. *Journal of Cognitive Science* 9(2): 161-174.
- Liu, H., R. Hudson, Z. Feng. 2009. Using a Chinese treebank to measure dependency distance. *Corpus Linguistics and Linguistic Theory* 5(2): 161-174.
- Mahootian, Sharzad, and Beatrice Santorini. 1996. Code-Switching and the Complement/Adjunct Distinction. *Linguistic Inquiry* 27: 464 - 79.
- Muysken, Pieter. 2000. *Bilingual Speech. A Typology of Code-Mixing.* Cambridge: Cambridge University Press.
- Pake, James. 1998. The Marker Hypothesis. A constructivist theory of language acquisition. PhD thesis, University of Edinburgh.
- Pfaff, Carol. Constraints on language mixing: intrasentential code-switching and borrowing in Spanish/English. *Language* 55: 291 - 318.
- Poplack, Shana. 1980. Sometime I'll start a sentence in Spanish y termino en Espanol: toward a typology of code-switching. *Linguistics* 18: 581 - 618.

Sebba, Mark. A congruence approach to the syntax of codeswitching. *International Journal of Bilingualism* 2(1): 1 - 19.

Temerpley, D. 2008. Dependency length minimization in natural and artificial languages. *Journal of Quarterly Linguistics* 15: 256-282.

Treffers-Daller, Jeanine. 1994. *Mixing Two Languages: French-Dutch Contact in a Comparative Perspective*. Berlin: de Gruyter.

Appendix A Notation – summary

Dependency types/ syntactic functions in WG	
Post-adjunct	>a
Pre-adjunct	a<
Complement	c

Pre-complement (before 's, -ing)	c<
Particle	e
Free complement	f
Indirect object	i
Negative (not)	n
Sharer/x-comp	r
Subject	s
Object	o
Prepositional complement	p
Extraposee	>x
Extractee	<x

(<http://www.phon.ucl.ac.uk/home/dick/enc.html>)

Appendix B – mean dependency distances for monolingual and mixed syntactic functions in Emigranto

	>c	s<	>s	>a	a<	>r	r<	>o	o<	>x	x<	>n	n<	>p	p<	total
G	0.65	0.54	0.07	1.1	0.37	1.64	0.07	0.78	0.83	-	2.16	0.33	0	-	0	0.73
E	0.22	0.07	- ⁷	1.26	0.38	0.53	-	0.5	-	-	0	0	-	-	-	0.4

Table 2. Mean distances of monolingual German and English syntactic functions

	>c	s<	>s	>a	a<	>r	r<	>o	o<	>x	x<	total
G	0.65	0.54	0.07	1.1	0.37	1.64	0.07	0.78	0.83	-	2.16	0.73
h _G	0.1	0.7	0.5	2.9	0.52	0.95	0.29	1.38	0.5	0.33	2.07	0.6

Table 3. Mean distances of selected monolingual German and mixed syntactic functions with a German head

	>c	s<	>a	a<	>r	>o	>x	x<	>n	Total
E	0.22	0.07	1.26	0.38	0.53	0.5	-	0	0	0.4
h _E	0.84	0.9	1.33	0.78	2.12	0.18	0.45	3.5	-	1.05

Table 4. Mean distances of selected monolingual English and mixed syntactic functions with an English head

⁷ For empty cells mean distances are not available.

Creating a Dependency Syntactic Treebank: Towards Intuitive Language Modeling

Kristiina Muhonen and Tanja Purtonen

Department of Modern Languages

University of Helsinki

kristiina.muhonen@helsinki.fi, tanja.purtonen@helsinki.fi

Abstract

In this paper we present a user-centered approach for defining the dependency syntactic specification for a treebank. We show that by collecting information on syntactic interpretations from the future users of the treebank, we can model so far dependency-syntactically undefined syntactic structures in a way that corresponds to the users' intuition. By consulting the users at the grammar definition phase we aim at better usability of the treebank in the future.

We focus on two complex syntactic phenomena: elliptical comparative clauses and participial NPs or NPs with a verb-derived noun as their head. We show how the phenomena can be interpreted in several ways and ask for the users' intuitive way of modeling them. The results aid in constructing the syntactic specification for the treebank.

1 Introduction

Building a treebank is an expensive effort consuming a lot of time and resources. To ensure the usability of the result, it is wise to ascertain that the chosen syntactic modeling responds to needs of its users. The Finnish CLARIN, FIN-CLARIN, project¹ provides language resources for researchers by creating a treebank and a dependency parser for unrestricted text. Because the main user groups of the Finnish treebank are presumably language researchers and students, it is necessary to ensure that the syntactic modeling used in the treebank accords with their linguistic intuition. In this paper we present a case study of improving the syntactic representation of the

Finnish treebank on the basis of its user groups' judgment.

The FIN-CLARIN treebank project² is in a phase in which the first specification of the dependency syntactic representation and the first manually annotated FinTreeBank are ready, and the morphological definition is in progress (Voutilainen and Lindén, 2011). The base for the first version of the treebank is a descriptive grammar of Finnish (Hakulinen et al., 2004a). The treebank consists of the grammar's example sentences³. The advantage of this approach is that already in the first version of the treebank every phenomenon described in the grammar must also be described in the dependency syntactic framework.

During the creation of the first treebank and the syntactic specification, the annotators encountered some phenomena in which it was hard to define the one and only best dependency syntactic representation. The problems in defining such phenomena are due to two reasons. Sometimes the descriptive grammar did not state only one specific representation for a phenomenon. In other cases the annotators reported that the traditional way of representing a phenomenon covered only the most typical cases but that the traditional representation seemed uninformative and unsuitable for covering the whole phenomenon.

In this paper we concentrate on two complex syntactic structures for which the wide-coverage descriptive grammar of Finnish (Hakulinen et al., 2004a) does not offer a complete solution: elliptical comparative clauses and NPs with either a participial construction or a verb-to-noun derivation. The two structures are only roughly defined in the first version of the treebank, and they need to be fully formulated in the second version. We

¹<http://www.ling.helsinki.fi/finclarin/>

²<http://www.ling.helsinki.fi/kieliteknologia/tutkimus/treebank/>

³The online version of the grammar:
<http://kaino.kotus.fi/visk/etusivu.php>



show that the dependency syntactic representation of undefined or complex structures can be better outlined when consulting the user groups of the treebank for their intuitive solution at the syntactic definition phase.

The user-centered approach guarantees that the syntactic representation complies with the majority's view which ensures maximizing the usability of the treebank. For this purpose we composed an e-query, in which we collected the answerers' intuitive interpretations of the two structures. Recording the user groups' intuitive solution complements, but does not replace the approximate syntactic representation already created in the project.

The first purpose of our experiment is to see how native speakers interpret elliptical comparative sentences, participial NPs with sentence-like structures and NPs with a verb-derived head. This sheds light on how the complex phenomena can be parsed in a natural way. The second aim is to estimate, is it beneficial to use an e-query at the syntactic specification phase. In this estimation we consider the number, the quality and the distribution of the answers. The third benefit of the test is to see whether there is a hidden consensus on the phenomena uncovered in the descriptive grammar and not yet described in the dependency syntactic framework. This, however, is not the main focus of our pilot study, but rather a side-product of the experiment.

2 Linguistic Background

In this section we outline the linguistic phenomena. We also show why the phenomena have alternative solutions.

2.1 Elliptical Comparative Sentences

The first phenomenon we concentrate on is the elliptical comparative structure. Finnish and English comparative structures are formed in a rather similar way. Typically a Finnish comparative structure contains:

- the comparative form of an adjective or an adverb formed with the comparative ending *-mpi*,
- the item being compared (subject of the main clause), and
- the subordinating conjunction *kuin*.

The next example shows a typical comparative structure:

- (1) Ana on pidempi kuin Maria.
Ana is taller than Maria
Ana is taller than Maria.

In example (1) the target of the comparison is Maria and the item being compared is Ana. It is also possible that the target is not semantically equivalent with the item being compared, like in the following example:

- (2) Ana on (nyt) pidempi kuin ennen.
Ana is (now) taller than before
Ana is now taller than before.

In this sentence, Ana is still the item being compared, but the comparative clause (*ennen/before*) is not comparable with the subject of the main clause (*Ana*), but with another word (*nyt/now*) in the previous clause. This equivalent word (*nyt/now*) is not necessarily even mentioned.

The diversity of comparative structures is a challenge for parsing: semantically oriented dependency parsing aims at an analysis in which the head is semantically, not only grammatically, considered the head. In our experiment, we investigate should sentences (1) and (2) be analyzed similarly with each other by marking e.g. the adjective, verb or the conjunction as the head. The other option is to link two equivalent words (e.g. *Ana–Maria*, *now–before*) with each other.

The comparative conjunction *kuin* can be followed by a whole, or an elliptical, sentence:

- (3) Ana on nyt pidempi kuin Maria ennen.
Ana is now taller than Maria before
Ana is now taller than Maria before.

The comparative clause can be seen as a common structure of its own or as an elliptical clause. In principle, all cases where the comparative conjunction is not followed by a verb are elliptical clauses. In Finnish it is common to have a whole elliptical sentence after the comparative conjunction, like in example 3. Thus, the way of analyzing the comparative clause is significant; it can be analyzed as a structure of its own, or as an elliptical clause. In the tradition of dependency grammar, the subordinate clauses are linked to the main clause via the verb and all other head-dependent-relations stay inside the subordinating clause (Tesnière, 1980, p. 231). If the words following the comparative conjunction are seen as a clause, it is justifiable to have only one link from

this clause to the main clause also in elliptical structures .

It is also possible to see the comparative as a conventional structure with a) no need to link the word following the conjunction to the main verb or b) no need to have only one link to the main clause. Thus the head-dependent relations can be seen e.g. in the following way (for the glossed sentence, see example (3)):

(4) Ana on nyt pidempi kuin Maria ennen.

In our experiment, we try to find out the most natural and informative way to describe different kinds of comparative structures. The main research question relating to comparative clauses is to clarify which word(s) the answerers mark intuitively as the head of the word(s) following the comparative conjunction.

2.2 NPs with Participles and Derived Nouns

NPs with sentence-like structures are challenging to parse. Making decisions on how the NP-internal structure should be represented in the dependency grammar framework is a challenging task with no absolute correct solution.

The standard work on Finnish grammar (Hakulinen et al., 2004a) states that if a participle functions as an attribute, it can take an object or an adverbial as a premodifier. The internal structure of an NP with a verb-derived noun as the head of the phrase resembles that of a participial NP. The semantics of the arguments of the head nouns in the following sentences are thus alike.

- (5) päivittäin vihanneksia syövä
daily vegetables eating-PR-PRT-ACT
eating vegetables daily
- (6) päivittäinen vihannesten syönti
daily vegetables eating-DER
eating vegetables daily

In both examples (5) and (6) the head *syövä/syönti* (*eating*) takes a direct object: *vihanneksia/vihannesten* (*vegetables*). In the participial construction, example (5), the premodifier *päivittäin* (*daily*) is an adverb directly dependent on the participial head, *syövä* (*eating*). In NP (6) the premodifier *päivittäinen* (*daily*) is an attribute directly dependent on the head noun *syönti* (*eating*).

We want to examine whether *vihannesten/vihanneksia* (*vegetables*) is interpreted as the

object in both cases (5) and (6). Traditionally the object has only been seen as the complement of a verb, not of a noun (Hakulinen et al., 2004b).

With the help of an e-query, in which the answerers assign grammatical functions to the premodifiers, we want to examine whether the two constructions, the participial construction, example (5), and the NP with a verb-derived noun as its head, example (6), get analyzed similarly. In addition, we anticipate new insight on the distinction between an adverb and attribute defining a participle or a verb-derived noun.

We extend the research question to cover subjects as well. If a derived noun can take an object as a premodifier, it seems natural that it would analogously be able to take a subject. Consider the following NP:

- (7) murhaajan ensimmäinen tappo
murderer's first killingDER
the murderer's first killing

In example (7) the verb-derived noun *tappo* (*killing*) has a premodifier, *murhaajan* (*murderer*). Since the semantics of the sentence cannot be interpreted as the killer being the object of the killing, we want to investigate whether speakers assign *murhaajan* the grammatical function of a subject.

The test we conducted seeks to give new insight on whether the NP's internal grammatical functions are assigned in a parallel manner in participial NPs and NPs with derived nouns. In section 4 we present the results of the experiment.

3 The Experiment

The test is conducted as an online query. We asked Finnish native speakers to answer multiple-choice questions regarding the dependency relations of elliptical verb phrases and sentences and the grammatical function of a participial NP or an NP with a verb-derived head noun. A similar way of using crowdsourcing for collecting linguistic data is described in e.g. Munro et al. (2010).

We presented the respondents a set of ten sentences and asked them to choose the most intuitive answer to the questions from a list of choices. We did not give the respondents the option of inserting a missing element to the elliptical comparative structures because we want to stick to a surface syntax representation.

The 428 answerers are mainly language students and researchers at the University of Helsinki.

They were encouraged to answer the questions swiftly based on their intuition, not according to their knowledge of Finnish grammar. Since the purpose of the query is to find out the users' opinion on the two structures, it does not matter whether their language competence influences their intuitive answers. Most importantly we want to ensure that the future users of the treebank agree with the annotation scheme and that the scheme does not contradict with their language sense.

In the query we collected information about dependency relations (see example question in figure 1) and grammatical functions (figure 2) separately. (For the word-to-word translations, see Appendix A.) To better conceal the aim of the questionnaire, questions on dependency relations alternated with questions on grammatical functions.

Unicafe tarjoaa parempaa ruokaa kuin ennen.

“Unicafe offers better food than before.”

What is the head of the word “ennen”, i.e. which word is it closest related to?

- a. Unicafe
- b. tarjoaa
- c. parempaa
- d. ruokaa
- e. kuin

Figure 1: A sample question regarding dependency relations (Sentence 8 in Appendix A.2)

Ojaan pudonnut auto kaivettiin ylös.

“The car that fell into a ditch was dug out.”

What is the grammatical function of “ojaan”?

- a. predicate
- b. subject
- c. object
- d. adverbial
- e. attribute

Figure 2: A sample question regarding grammatical functions (Sentence 1 in Appendix A.1)

Our aim was to estimate if it is possible to get reliable answers to both kinds of questions. The main reason for asking either about dependencies or functions was to not make the questionnaire too time-consuming. Also, we were particularly interested in how the answerers perceive dependency relations in comparative structures on the one hand, and how they assign grammatical functions to complex NPs on the other.

The respondents filled in the questionnaire independently without supervision so we did not monitor the average time taken for answering. We also

do not precisely know the background of the answerers, only that most of them are either language students or researchers who heard about the query via mailing lists. The phrasing of the questions did not point the answerers towards dependency grammar but asked the answerers to base their answers purely on intuition.

In order to get a better understanding on the competence of the respondents, the first question in the questionnaire was a control question without elliptical structures or complex NPs. We simply asked the answerers to specify a dependency relation in the following sentence:

Tuuli käy päivisin koulua, ja Vesa työskentelee kotona.

“During the day Tuuli goes to school and Vesa studies at home.”

What is the head of the word “kotona”, i.e. which word is it closest related to?

- a. Tuuli
- b. käy
- c. päivisin
- d. koulua
- e. ja
- f. Vesa
- g. työskentelee

Figure 3: The control question (Sentence 6 in Appendix A.2)

The dependencies in the control question presented in figure 3 are unambiguous so that giving an illogical answer to the question reveals us either that the answerer is not familiar with the notion “head word” or that the answer was marked by accident. The responses to the control question are encouraging: 71% marked *työskentelee* (*works*) as the head of *kotona* (*at home*), and 22% *Vesa*. This leaves us with only 7% illogical answers. Notwithstanding, we regard the results of the questionnaire merely indicative of the answerers intuitive language modeling.

Even though a part of the answers to the control question are not predictable, see example sentence 6 in Appendix A.2, we take all answers into account and do not consider any answers counter-intuitive. Still, further research might benefit from narrowing down the results based on the control question.

The experiment presented here is a case study with only 10 questions including one control question. If the experiment would be repeated to cover more phenomena, there should be more questions and different types of control questions. E.g. the elliptical sentences should have a non-elliptical

equivalent as a control question to test whether the dependencies are interpreted identically.

4 Results: Modeling the Phenomena

Before determining the syntactic specification for the phenomena, we explore the different ways of modeling them. At this point of the Finnish tree-bank project, the main goal is not to follow any kind of formalism but to investigate the most natural and semantically informative representation of syntax. Dependency grammar allows for a natural representation of e.g. long-distance relationships because of the non-hierarchical nature of dependency relations (Kübler et al., 2006). At this point we do not try to avoid crossing branches in the dependency trees, since we allow e.g. linking the words of the elliptical comparative sentences to their semantic equivalents in the main clause.

4.1 Elliptical Comparative Structure

The main clause of the comparative clause does not necessarily contain any semantically equivalent word with the word after the subordinating conjunction (see sentence 8 in Appendix A.2). In such a case the most used solution by the answerers is to link the word to the conjunction (55%). The second popular solution is to mark the adjective as the head (20%) and the third popular option for the head is the verb of the main clause (14%).

If the final annotation scheme prefers marking content words as heads, it is worth noticing that 20% of the answerers mark the adjective as the head in a typical elliptical comparative clause with only one word after the conjunction. Also, the conjunction is the most popular choice for the head only when there are no clear semantic or grammatical equivalents in the main clause and no other words in the elliptical clause.

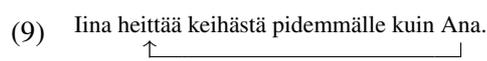
Based on the test, it is intuitively most popular (24%) to link two equivalent words with each other, when the verb of the main clause is *olla* (*be*). Example (8) illustrates⁴ this solution where the equivalent words, expressions of location, are linked with each other. This tendency to link two compared items to each other supports selecting a representation in which crossing branches are possible.

- (8)  Täällä on kuumempaa kuin Espanjassa.

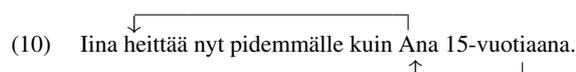
⁴See sentence 7 in Appendix A.2 for the complete answers and the full sentence.

Täällä on kuumempaa kuin Espanjassa.
Here is warmer than Spain-(ine)
It is warmer here than in Spain.

According to our working hypothesis, the results suggest that when the verb of the main clause is “semantically heavier”, the verb is seen as the head more often (33%). This solution is shown in the example (9) where the answerers marked the verb as the head of the elliptical clause even when there is an equivalent in the subject position in the main clause.

- (9) 
Iina heittää keihästä pidemmälle kuin Ana.
Iina throws javelin further than Ana
Iina throws the javelin further than Ana.

In the examples above, there is only one word in the comparative clause. When the comparative clause contains an elliptical clause with two or more words, the solutions depend on the interpretation. When there is a primary object of comparison in the comparative clause and the other words are semantically clearly connected to this primary word, it is clearly seen as a head (79%), even if there are equivalent words in the main clause. For example:

- (10) 
Iina heittää nyt pidemmälle kuin Ana 15-vuotiaana.
Iina throws now further than Ana
15-vuotiaana.
15 years old
Iina throws the javelin further now than Ana when she was 15 years old.

When the semantic link between the words of an elliptical comparative clause is not so clear as in example (10), the solutions are so variable that there is no clear conclusion we can draw. Still, based on the answers it is clear that this phenomenon, an elliptical comparative clause, is a real challenge for parsing.

Above we have shown how to treat comparative structures which include elliptical clauses. The comparative sentence can also consist of elliptical phrases, like in the following example⁵:

- (11) Matka Tukholmasta Tallinnaan on pidempi
Distance Stockholm-ELA Tallinn-ILL is longer
kuin Riiasta Wieniin.
than Riga-ELA Vienna-ILL

⁵ELA=elative, ILL=illative

The distance from Stockholm to Tallinn is longer than from Riga to Vienna.

The most popular solution (52%) is to connect first part of the elliptical phrase (*Riiasta/*from *Riga*) to the head of the phrase (*matka/*distance). The latter part of the elliptical phrase (*Wieniin/*to *Vienna*) was mostly (41%) seen as a dependent of the word (*Riiasta/*from *Riga*).

Even though in many cases a semantically heavy word is seen as the head of a comparative clause, throughout the test and in all different kinds of elliptical comparative clauses, the conjunction has always clear support. In all cases, *kuin* is marked as the head of the whole comparative clause by at least 15% of the answerers.

Based on this experiment, we can now roughly sketch the main principles of representing comparative structures intuitively:

- When there is an equivalent sentence element in the main clause, mark it as the head of the dependent in the comparative clause. Link the other parts of the elliptical sentence to this word.
- When there is no equivalent sentence element in the main clause, mark the conjunction as the head of the elliptical comparative clause. When favoring semantically heavier words, mark the adjective as the head as 20% of the answerers do in question 8. (Appendix A.2.).

4.2 Participles and Derived Nouns

The participial NP constructions we wanted the respondents to assign grammatical functions to are the following:

(12) Ojaan pudonnut auto kaivettiin ylös.
Ditch fallenPAST-PRT-ACT car dug up
The car that fell into a ditch was dug out.

(13) Kirkon penkillä itki tekojaan syvästi
Church bench cry deeds deeply
katuva mies.
regrettingPRES-PRT-ACT man
A/the man who deeply regretted his deeds was crying on the church bench.

The primary results of the e-query are assembled in table 1. For conciseness' sake only the three most popular answers are displayed in the table. For the complete results, see Appendix A.1.

The past participles indicate a completed action and have corresponding pluperfect forms. The past participle active form *pudonnut* (*fallen*) corresponds to a relative clause:

(12) OJAAN PUDONNUT AUTO KAIVETTIIN YLÖS.			
(13) KIRKON PENKILLÄ ITKI TEKOJAAN SYVÄSTI KATUVA MIES.			
Word	Obj	Adv	Attr
ojaan	47 (11%)	246 (57%)	120 (28%)
tekojaan	250 (58%)	51 (12%)	96 (22%)
syvästi	27 (6%)	236 (55%)	158 (37%)
N=428			

Table 1: Grammatical functions of participial NPs

(14) auto, joka oli pudonnut ojaan
car which had fallen into ditch
a/the car which had fallen into a ditch

A participle can get an adverbial modifier (Hakulinen et al., 2004a). In the corresponding relative clause (14) the grammatical function of the premodifier *ojaan* (*into a ditch*) is adverb. Based on the answers of the e-query, the distinction is not clear in the participial construction. As can be seen from table 1, in fact 57% of the answerers regard *ojaan* an adverb, but as many as 28% consider it an attribute. This might be explained by participles possessing traits of both verbs and adjectives, and the typical modifier of an adjective would be an attribute. Some, 11%, see *ojaan* as an object. This can possibly be explained by the whole NP being the object of the sentence and with semantics: *ojaan* is the target of falling.

In the second participial construction, example (13), we asked the answerers to assign a grammatical function to both of the premodifiers of the participle: *tekojaan* (*deeds*) and *syvästi* (*deeply*). Analogously to the past participle, the present participle *katuva* (*regretting*) corresponds to a relative clause with a present tense verb.

(15) mies, joka katuu tekojaan syvästi
man who regrets deeds deeply
a/the man who regrets his deeds deeply

Again, the relative clause (15) has clearly distinguishable grammatical functions: *tekojaan* is the direct object of the head verb *katuu*, and *syvästi* is an adverb postmodifying the head.

Analogously, in the participial construction corresponding to the relative clause, 58% of the answerers see *tekojaan* as the object of the sentence. 22% give it the attribute-label, and 12% name it an adverb (see table 1). This indicates that the object premodifier of a participle is a rather straightforward case: a vast majority of the answerers see it as an object.

NPs with a derived noun as their head constitute a similar problem with assigning phrase-internal grammatical functions. Take, for example, the following three sentences from the e-query. We present the most frequent answers in table 2.

- (16) Puolet rehtorin ajasta meni oppilaiden
Half principal time went student
ohjaukseen.
guidance
Half of the principal's time was spent on guiding the students.
- (17) Päivittäinen vihannesten syönti pitää sinut
Daily vegetables eating keeps you
terveenä.
healthy
Eating vegetables daily keeps you healthy.
- (18) Murhaajan ensimmäinen tappo sai paljon
Murderer first kill receive a lot
julkisuutta.
publicity
The murderer's first killing received a lot of publicity.

Word	Subj	Obj	Adv	Attr
oppilaiden		127 (30%)	43 (10%)	243 (57%)
vihannesten	45 (11%)	130 (30%)		218 (51%)
murhaajan	73 (17%)		38 (9%)	280 (65%)
N=428				

Table 2: Grammatical functions of derived NPs

In examples (16) and (17) the NP investigated is in the object position. Both cases reflect a very similar way of intuitive modeling among the respondents: *oppilaiden* and *vihannesten* are given the function of an attribute, 57% and 51%, respectively.

We will now proceed to examine whether a noun can receive an object based on the answers' intuition. Traditionally only verbs get an object (Hakulinen et al., 2004b), but we want to see if a noun derived from a verb retains this feature of a verb.

The difference between the intuitive response and the object-attribute distinction is clear when comparing the results of the participial NP of sentence (13) and the NPs with a verb-to-noun derivation as the head in sentences (16) and (17). The

vast majority (58%) of the respondents label *tekojaan* as an object in (13), whereas only 30% see *oppilaiden* and *vihannesten* in sentences (16) and (17) as the object. This suggests that the verb-to-noun derivations do not possess the traits of a verb, and the traditional definition of the object prevails.

The object-attribute distinction can also be seen from another point of view. As many as 30% of the respondents do in fact think that a noun can receive an object despite the option being excluded by traditional grammars. This suggests that the answerers have a strong semantic way of modeling the phrase alongside with the morphological view.

In sum, intuitive modeling of participial NPs or NPs with a verb-derived head should follow these principles:

- The premodifier of a verb-to-noun derivation is interpreted as an attribute.
- The premodifier of a participial is treated analogously to premodifiers of verbs. It is seen as an object when the verb would take an object, and an adverbial when the verb would have one too.

5 Conclusion

In this paper we have shown that an e-query is a useful tool for collecting information about a treebank's user groups' intuitive interpretations of specific syntactic phenomena. This information is needed to ensure that the syntactic representation used in the treebank does not deviate from its user's language intuition.

Using an e-query for probing for the respondents' intuitive way of modeling syntactic phenomena moves from separate cases to general modeling: A respondent does not need to be consistent with her answers and have one specific answering policy throughout the e-form. Our aim is to collect information about modeling the whole phenomena coherently so these collected opinions are not seen as an unquestionable base for the syntactic model.

Based on this experiment we can also conclude that the variation between the answers results from the fact that these phenomena – the structure of the verb-based NP and the elliptical comparative clause – are semantically ambiguous, and representing them in the dependency grammar framework is not a univocal task. To exclude the possibility of having the same kind of variation in

the answers also between other phenomena, we had a control question in the test. The majority of the answers to this question are homogeneous (71%), and the second popular answer (22%) is also semantically valid. This means that 7% of the answers were illogical in a clear-cut case, so at least 7% of the answers should be considered ill-advised. Thus, again we consider the results only as advisory.

Even though the answers to the e-query are varied, some general principles can be made based on our experiment. Interestingly, contradicting the tradition of dependency grammar, where the verb of the main clause is seen as the core of the sentence to which other clauses are related, in some comparative structures the answerers consider e.g. the adjective as the head of the whole comparative clause. This questions the traditional verb-centric modeling of the comparative clauses and suggests perhaps a more informative representation, where the objects of the comparison are more clearly visible.

Based on the number and quality of the answers, an e-query seems to be suitable a suitable method for getting a general view of the users' intuitive way of modeling syntactic phenomena. The large number of the answers also allows for the possibility to eliminate a part of the answers on the grounds of the control question. Before finalizing the syntactic representation of the treebank, we will scrutinize the answers in a more thorough way to receive a more accurate and valid model where the nonsensical answers do not skew the results.

Our experiment shows that the method employed provides new information on how to define the phenomena in the dependency syntactic framework. This information can be used when determining the syntactic specification. The results point towards a way of modeling the syntactic phenomena so that the final syntactic representation used in the treebank does not argue against the view of its users.

Acknowledgements

The ongoing project has been funded via CLARIN, FIN-CLARIN, FIN-CLARIN-CONTENT and META-NORD by EU, University of Helsinki, and the Academy of Finland. We would like to thank Atro Voutilainen and the three anonymous reviewers for their constructive comments.

References

- Auli Hakulinen, Maria Vilkuna, Riitta Korhonen, Vesa Koivisto, Tarja-Riitta Heinonen, and Irja Alho. 2004a. *Iso suomen kielioppi*. Suomalaisen Kirjallisuuden Seura, Helsinki. ISBN: 951-746-557-2.
- Auli Hakulinen, Maria Vilkuna, Riitta Korhonen, Vesa Koivisto, Tarja-Riitta Heinonen, and Irja Alho. 2004b. Ison suomen kieliopin verkkoversio: määritelmät. Suomalaisen Kirjallisuuden Seura. <http://kaino.kotus.fi/cgi-bin/visktermit/visktermit.cgi>.
- Sandra Kübler, Jelena Prokić, and Rijksuniversiteit Groningen. 2006. Why is german dependency parsing more reliable than constituent parsing. In *Proceedings of the Fifth Workshop on Treebanks and Linguistic Theories (TLT)*, pages 7–18.
- Robert Munro, Steven Bethard, Steven Kuperman, Vicky Tzuyin Lai, Robin Melnick, Christopher Potts, Tyler Schnoebelen, and Harry Tily. 2010. Crowdsourcing and language studies: the new generation of linguistic data. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 122–130. Association for Computational Linguistics.
- Lucien Tesnière. 1980. *Grundzüge der strukturalen Syntax*. Klett-Cotta, Stuttgart. ISBN: 3-12-911790-3.
- Atro Voutilainen and Krister Lindén. 2011. Designing a dependency representation and grammar definition corpus for finnish. In *Proceedings of III Congreso Internacional de Lingüística de Corpus (CILC 2011) (upcoming)*.

A Complete Results

The total number of answers is 428.

Percentages are rounded to the nearest whole number.

A.1 NP Constructions

Word	Predicate	Subject	Object	Adverb	Attribute	NA
1. OJAAN PUDONNUT AUTO KAIVETTIIN YLÖS. "The car that fell into a ditch was dug out."						
<i>ojaan</i> into a ditch	5 (2%)	5 (2%)	47 (11%)	246 (57%)	120 (28%)	5 (2%)
2. PUOLET REHTORIN AJASTA MENI OPPILAIDEN OHJAUKSEEN. "Half of the principal's time was spent on guiding the students."						
<i>oppilaiden</i> students'	3 (1%)	6 (1%)	127 (30%)	43 (10%)	243 (57%)	6 (1%)
3. PÄIVITTÄINEN VIHANNESTEN SYÖNTI PITÄÄ SINUT TERVEENÄ. "Eating vegetables daily keeps you healthy."						
<i>vihannesten</i> vegetables-GEN	3 (1%)	45 (11%)	130 (30%)	22 (5%)	218 (51%)	3 (2%)
4. MURHAAJAN ENSIMMÄINEN TAPPO SAI PALJON JULKISUUTTA. "The murderer's first killing received a lot of publicity."						
<i>murhaajan</i> murderer's	2 (0%)	73 (17%)	14 (3%)	38 (9%)	280 (65%)	21 (5%)
5. KIRKON PENKILLÄ ITKI TEKOJAAN SYVÄSTI KATUVA MIES. "The man who deeply regretted his deeds was crying on the church bench."						
<i>tekojaan</i> deeds-PAR	1 (0%)	7 (2%)	250 (58%)	51 (12%)	96 (22%)	23 (5%)
PAR=PARTITIVE, GEN=GENITIVE						

A.2 Comparative Constructions

The following tables show what is seen as the head of the word in italics:

6. TUULI KÄY PÄIVISIN KOULUA, JA VESA OPISKELEE KOTONA. "During the day Tuuli goes to school and Vesa studies at home."						
Word	<i>Tuuli</i> Tuuli	<i>käy</i> goes	<i>päivisin</i> daily	<i>koulua</i> to school	<i>Vesa</i> Vesa	<i>opiskelee</i> studies
<i>kotona</i> at home	2 (0%)	14 (3%)	6 (1%)	6 (1%)	96 (22%)	304 (71%)

7. TÄÄLLÄ ON KUUMEMPAA KUIN TURISTEILLA KESÄLLÄ ESPANJASSA. "It is hotter here than what tourists experience in Spain during the summer."								
Word	<i>Täällä</i> Here	<i>on</i> is	<i>kuumempaa</i> hotter	<i>kuin</i> than	<i>turisteilla</i> tourists-ADE	<i>kesällä</i> in the summer	<i>Espanjassa</i> in Spain	NA
<i>turisteilla</i> tourists-ADE	25 (6%)	46 (11%)	59 (14%)	105 (25%)	- -	36 (8%)	126 (29%)	31 (7%)
<i>kesällä</i> in the summer	26 (6%)	30 (7%)	50 (12%)	32 (7%)	83 (19%)	- -	175 (41%)	32 (7%)
<i>Espanjassa</i> in Spain	103 (24%)	29 (7%)	52 (12%)	64 (15%)	84 (20%)	63 (15%)	- -	33 (8%)
ADE=ADESSIVE								



8. UNICAFE TARJOAA PAREMPAA RUOKAA KUIN ENNEN. "Unicafe offers better food than before."						
Word	unicafe Unicafe	tarjoaa offers	parempaa better	ruokaa food	kuin than	NA
<i>ennen</i> before	10 (2%)	59 (14%)	87 (20%)	17 (4%)	234 (55%)	21 (5%)

9. IINA HEITTÄÄ KEIHÄSTÄ JO NYT PIDEMMÄLLE KUIN ANA 15-VUOTIAANA. "Iina throws the javelin further already now than Ana when she was 15 years old."										
Word	Iina Iina	heittää throws	keihästä javelin	jo already	nyt now	pidemmälle further	kuin than	Ana Ana	15-vuotiaana 15 years-ESS	NA
<i>Ana</i>	59 (14%)	142 (33%)	16 (4%)	0 (0%)	1 (0%)	38 (9%)	129 (30%)	-	31 (7%)	12 (3%)
<i>15-vuotiaana</i> 15 years-ESS	7 (2%)	21 (5%)	5 (1%)	5 (1%)	21 (5%)	6 (1%)	15 (4%)	338 (79%)	-	10 (2%)
ESS=ESSIVE										

10. MATKA TUKHOLMASTA TALLINNAAN ON PIDEMPI KUIN RIIASTA WIENIIN. The distance from Stockholm to Tallinn is longer than from Riga to Vienna.									
Word	Matka Distance	Tukholmasta Stockholm-ELA	Tallinnaan Tallinn-ILL	on is	pidempi longer	kuin than	Riiasta Riga-ELA	Wieniin Vienna-ILL	NA
<i>Riiasta</i> Riga-ELA	222 (52%)	41 (10%)	1 (0%)	5 (1%)	27 (6%)	67 (16%)	-	11 (48%)	17 (4%)
<i>Wieniin</i> Vienna-ILL	138 (32%)	3 (1%)	40 (9%)	2 (0%)	26 (6%)	22 (5%)	176 (41%)	-	21 (5%)
ELA=ELATIVE, ILL=ILLATIVE									



Extracting Valency Patterns of Word Classes from Syntactic Complex Networks

Chen Xinying Xu Chunshan Li Wenwen

Communication University of China, Beijing

cici13306@gmail.com

Abstract

Our study extracted different valency patterns of Chinese word classes from 3 different Chinese dependency syntactic networks and compared their similarities and differences. The advantages and disadvantages of network approach are discussed at the end of this paper. The results show that there are some persisting properties in Chinese which are not affected by style. There are also some word classes which are more sensitive to the stylistic impact in Chinese. The network approach to linguistic study can make the complex data concise and easy to understand. However, it also has some deficiencies. First of all, when the network size is large, the structure will become so complex that easy understanding is impossible. Secondly, although the network can easily provide an overview of the language, it usually fails to be much helpful when it comes to language details.

Introduction

Reductionism has driven 20th century science, with the result being that we have experts who know more and more about less and less while leaving us devoid of generalists and multi-disciplinary artists and scientists who can "connect the dots" across these fragmented foci (Albert-László Barabási 2002). Now, more and more people realize that we can not figure out the overall structure by researching parts. In this situation, the network science, which provides a way to study the relationship between parts from an overall perspective, has a rapid development.

Language system is a complex network (Hudson, 2007). Therefore, the use of complex networks is a necessary attempt to study language (Liu, 2011). The research of language network can give a global perspective about language structure and about the relationship between language units.

There have been many researches on language complex networks (Liu, 2010; Ferrer i Cancho *et al*, 2004; Yu, 2011). Although the networks are built at different levels of language and with different concerns, most studies put the emphasis on the common features of various networks, such as small world and scale-free characteristics. This research approach is novel, and the results are often difficult to interpret in terms of linguistic theories. It seems that the study of language network lacks a solid foundation of linguistic theories. For linguists, network is simply a means and a tool for linguistic study but not the goal. We hope to establish a close link between the network and linguistic theories and study how network can serve to local syntactic studies or semantic studies, so the network can play a more important role in linguistic study. The paper tries to making an insightful exploration in this direction.

In language networks, such as syntactic and semantic networks, the nodes are language units on the same linguistic level which have direct relationships with one another. From this perspective, the Valence Theory is rather network-friendly, which provides a suitable linguistic theory to explain the findings of studies of language networks.

The theoretical basis of our study is Probabilistic Valency Pattern (PVP, Liu 2006), which is developed from the classic Valence Theory. The study extracted 3 different valency patterns of Chinese word classes from 3 different Chinese dependency syntactic networks and compared their similarities and differences. The discussion about the advantages and disadvantages of the network approach on linguistic study will be presented at the end of the paper.

PVP

The Valence Theory has been developed and revised in many ways since Tesnière (1959) integrated valence into syntactic theory. The



concept of valency can be found in almost all modern linguistic theories.

Traditionally, the Valence Theory is a syntactic-semantic theory. It is a term used to describe the relationship between a language unit and its complement. With the development of computer technology, people began to use computers to analyze and process real language. To analyze real language, taking only the complement into account is not enough. Under such circumstances, Liu (2006) propose PVP. After surveying the definitions of valence claimed by Tesnière (1959), Helbig (1978 2002), Schenkel (1978), Fischer (1997), Mel'čuk (2003), Hudson (2004) and others, Liu found that, despite some differences among these definitions, one thing remains unvarying: valence is the combinatorial capacity of a word. They argued that this is the general definition of valence: the combinatorial capacity shared by all words as one of their fundamental attributes. The capacity is a potential one whose realization is constrained by syntactic, semantic and pragmatic factors. When a word comes into a text, the potential capacity is activated, producing a dependency relation and establishing a pattern of sentence structure. They also put forward that the dependencies involved in the valence of a word may distribute unevenly. The probability information should be added to the descriptions of words' valences so as to indicate the strength of corresponding combinations. According to PVP, it is necessary to qualitatively describe the dependencies involved in the valence of a word or word class.

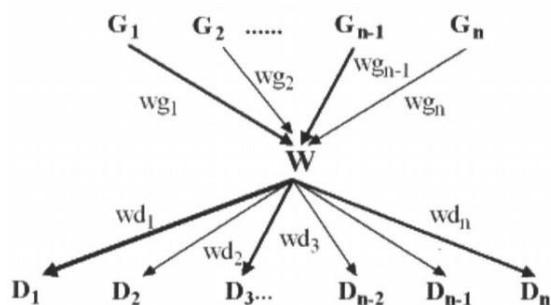


Figure 1. Valency Patterns in PVP (Liu 2006)

The 'W' can be a word class or a specific word. $G_1, G_2 \dots G_n$ are dependencies that take 'W' as the dependent. $D_1, D_2 \dots D_m$ are de-

pendencies that take 'W' as the governor. $wg_1, wg_2 \dots wgn$ are the probabilities of different dependencies and $wg_1 + wg_2 + \dots + wgn = 1$. It is the same with $wd_1, wd_2 \dots Wdm$.

Extracting valency patterns of Chinese word classes from syntactic complex networks

PVP represents the probabilistic dependencies between words or word classes. Before the exploration into language details, an overview of Chinese dependency structure will be of much help. So we chose to study the valency patterns of word classes instead of words. There are no similar researches so far in literature. First, we built 3 Chinese dependency syntax treebanks and then converted them into dependency syntactic networks. After that, from these networks we extracted the valency patterns of Chinese word classes and compared their similarities and differences. Treebanks are the basis of this study. Taking stylistic influences into account, we selected the “实话实说” shi-hua-shi-shuo ‘name of a famous Chinese talk show’ (hereinafter referred to as SHSS) and “新闻联播” xin-wen-lian-bo ‘name of a Chinese TV news program’ (hereinafter referred to as XWLB), two corpora with different styles, for annotation. We transcribed and annotated these two oral corpus. The annotation scheme is the Chinese Dependency Annotation System proposed by Liu (2006). SHSS is colloquial, containing 19,963 words. XWLB is of a quite formal style, containing 17,061 words. In order to get a corpus which can reflect the general structure of Chinese without the reflections of language styles, we put the SHSS and XWLB together and get the third corpus. We respectively built 3 treebanks with SHSS, XWLB and SHSS+XWLB (hereinafter referred to as the S-treebank, X-treebank, A-treebank). Table 1 shows the format of our Chinese dependency treebanks.

This format includes all the three mentioned elements of the dependency relation, and can easily be converted into a graph as shown in Figure 2.

Order number of sentence	Dependent			Governor			Dependency type
	Order number	Character	POS	Order number	Character	POS	
S1	1	这	r	2	是	v	subj



S1	2	是	v	6	。	bjd	s
S1	3	一	m	4	个	q	qc
S1	4	个	q	5	苹果	n	atr
S1	5	苹果	n	2	是	v	obj
S1	6	。	bjd				

Table 1. Annotation of a sample sentence in the Treebank¹

¹ The details of all codes and symbols in tables and figures in this paper are available in Appendix A.



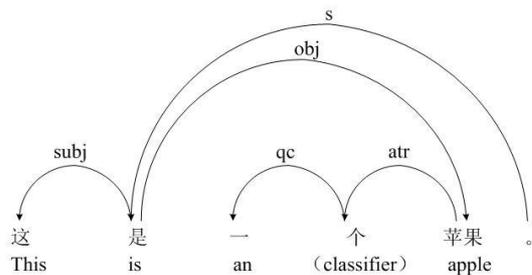


Figure 2. The graph of the dependency analysis of a sentence

With words as nodes, dependencies as arcs and number of dependencies as the value of arcs, networks are built, in which the direction of arc is defined as from governor nodes to dependent nodes. For example, the sample shown in Figure 2 can be converted to a network structure as shown in Figure 3 (excluding punctuation). Figure 4 presents the syntactic network converted from A-treebank.

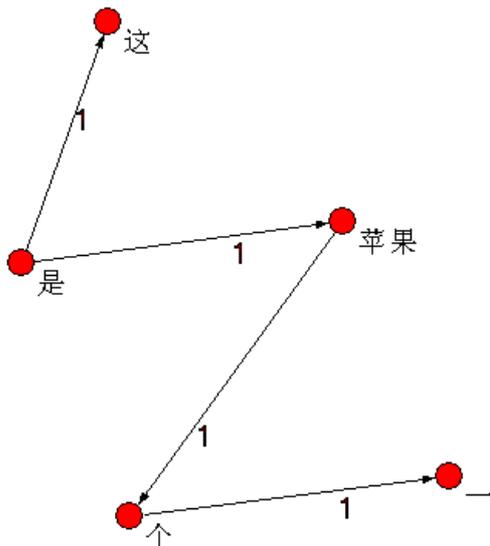


Figure 3. Network of 这是一个苹果 zhe-shi-yi-ge-ping-guo 'this is an apple'

Governor \ Dependent	d	r	m	a	u	c	p	q	v	n	o	e	zmen	zdi
d	9	0	4	103	8	7	7	1	567	17	0	0	0	0
r	2	5	8	6	35	6	18	39	154	164	0	0	0	0
m	3	0	80	7	11	5	5	396	84	160	0	0	0	0
a	1	1	7	46	135	29	8	8	384	422	0	0	0	0
u	0	1	17	20	2	58	3	6	415	749	0	0	0	0
c	1	1	1	7	44	8	24	0	293	83	0	0	0	0
p	3	1	0	5	56	6	3	0	638	13	0	0	0	0
q	1	0	2	4	13	5	10	6	121	274	0	0	0	0
v	5	4	15	23	365	289	119	8	2216	635	0	0	0	0
n	4	13	15	105	349	474	527	24	3046	2558	0	0	0	0
o	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	0	0	0	0	0	0	0	0	0	0	0	0	0	0
zmen	0	0	0	0	0	0	0	0	0	4	0	0	0	0
zdi	0	0	9	0	0	0	0	0	0	0	0	0	0	0

Table 2. values of arcs (X-treebank)

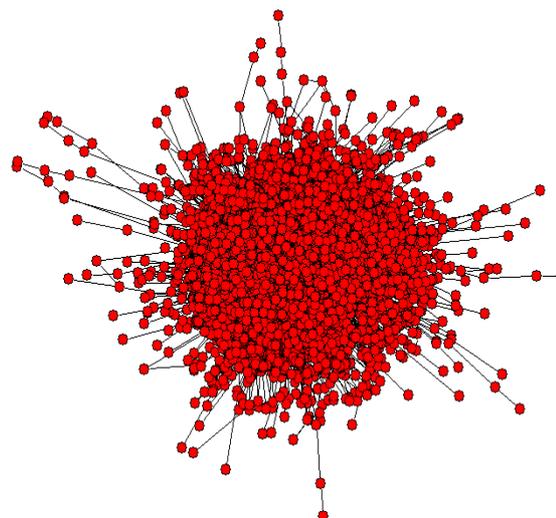


Figure 4. Network of A-treebank

In Figure 4, the nodes are words. We can also cluster the nodes which belong to the same word class into a new node and the new node will inherit all the dependencies of these nodes. In this way, we can obtain a dependency network of word classes. We extracted 3 networks of word classes from the S-treebank, X-treebank and A-treebank. For the sake of clarity, the values of arc, which are given in Table 2, 3 and 4, are not shown in Figure 5, 6, 7.

In Table 2, the first column is the list of dependent word classes and the first row is the list of governing word classes. In each cell in this table is the frequency of the dependency relation between a certain governing word class and a certain dependent governing word class, or, the value of the corresponding arc.

Governor Dependent	d	r	m	a	u	c	p	q	v	n	o	e	zmen	zdi
d	7	6	9	341	26	3	16	3	1481	13	0	0	0	0
r	2	62	19	93	229	14	136	432	1652	325	0	0	0	0
m	0	1	64	15	6	0	0	603	40	79	0	0	0	0
a	1	2	5	55	240	6	7	7	321	220	0	0	0	0
u	10	10	2	97	16	9	8	9	1090	668	0	0	0	0
c	0	6	0	28	7	3	10	2	338	32	0	0	0	0
p	0	1	0	26	18	3	1	0	460	10	0	0	0	0
q	2	15	3	26	19	1	6	10	278	692	0	0	0	0
v	11	8	2	124	315	22	44	7	3484	187	0	0	0	0
n	5	58	3	141	198	69	302	7	2382	688	0	0	0	0
o	0	0	0	0	0	0	0	0	1	0	0	0	0	0
e	0	0	0	2	0	0	0	0	1	0	0	0	0	0
zmen	0	361	0	0	0	0	0	0	1	8	0	0	0	0
zdi	0	0	31	0	0	0	0	0	0	0	0	0	0	0

Table 3. values of arcs (S-treebank)

Governor Dependent	d	r	m	a	u	c	p	q	v	n	o	e	zmen	zdi
d	16	6	13	444	34	10	23	4	2048	30	0	0	0	0
r	4	67	27	99	264	20	154	471	1806	489	0	0	0	0
m	3	1	144	22	17	5	5	999	124	239	0	0	0	0
a	2	3	12	101	375	35	15	15	705	642	0	0	0	0
u	10	11	19	117	18	67	11	15	1505	1417	0	0	0	0
c	1	7	1	35	51	11	34	2	631	115	0	0	0	0
p	3	2	0	31	74	9	4	0	1098	23	0	0	0	0
q	3	15	5	30	32	6	16	16	398	966	0	0	0	0
v	16	12	17	147	680	311	163	15	5701	822	0	0	0	0
n	9	71	18	246	547	543	829	31	5428	3246	0	0	0	0
o	0	0	0	0	0	0	0	0	1	0	0	0	0	0
e	0	0	0	2	0	0	0	0	1	0	0	0	0	0
zmen	0	361	0	0	0	0	0	0	1	12	0	0	0	0
zdi	0	0	40	0	0	0	0	0	1	0	0	0	0	0

Table 4. values of arcs (A-treebank)

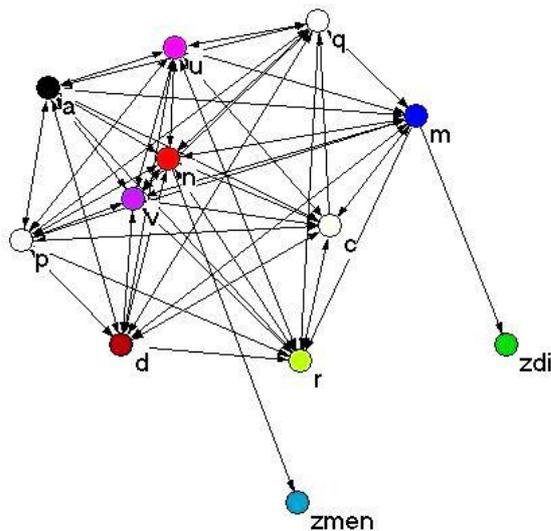


Figure 5. Network of word classes (X-treebank)

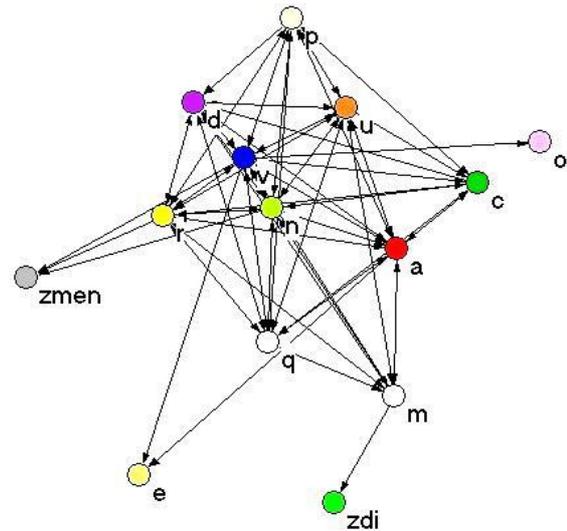


Figure 6. Network of word classes (S-treebank)

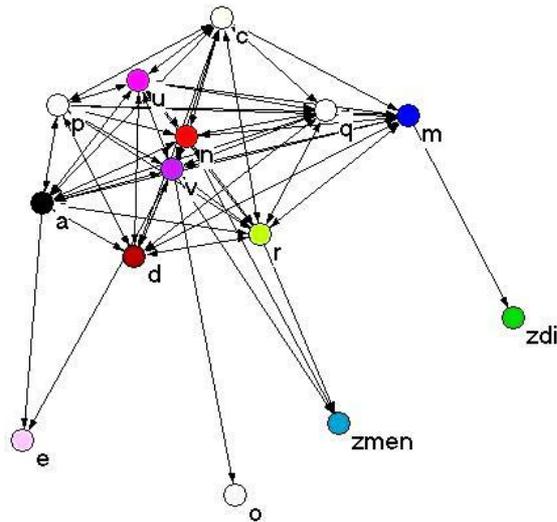


Figure 7. Network of word classes (A-treebank)

Since the PVP describes the combinatorial strength between words or word classes, the frequencies of dependencies, which reflect the actual realization of valence, can be used to quantitatively measure the combinatorial strength. Because the values of arcs indicate frequencies of dependencies between the nodes, the greater the value is, the greater this strength is in network.

Figure 1 can be seen as the valency pattern of an unknown word class. Obviously, the valence patterns of all word classes will merge into a complex network whose nodes are word classes, that is, a dependency networks as shown in Figure 5, 6 and 7. The network structure can be understood as roughly reflecting valency patterns, the nodes representing word classes and the arcs representing dependencies. The direction of arcs distinguishes governing nodes from dependent nodes and the values of arcs indicate the frequencies of dependencies, from which the probability can easily derive. We use, instead of numbers, the distance between nodes to indicate the combinatorial strength between network nodes, which, we hope, can make for an easy understanding of the combinatorial strength between nodes. The higher the value of arc, the greater the combinatorial strength and the shorter the distance.

Results and discussion

Based on these 3 networks, we found that the valency patterns of X-treebank, S-treebank and A-treebank have a few things in common:

- (1) 'zdi', 'zmen', mimetic word and interjection cannot be governors.
- (2) 'zdi' can only be the dependent of numeral.

- (3) mimetic word can only be the dependent of verb.
- (4) The distance between the verb and noun is the shortest. In other words, the dependency between noun and verb has the highest probability in all dependencies between different word classes.
- (5) The governor of adjective is most likely to be verb. The dependent of adjective is most likely to be adverb.
- (6) The dependent of pronoun is most likely to be 'zmen'.
- (7) The governor of numeral is most likely to be classifier. The dependent of numeral is most likely to be numeral.
- (8) The two most probable governors of classifier are noun and verb. The dependent of classifier is most likely to be numeral.
- (9) The governor of preposition is most likely to be verb. The dependent of preposition is most likely to be noun.
- (10) The dependent of auxiliary is most likely to be verb.
- (11) The governor of adverb is most likely to be verb.
- (12) The governor of conjunction is most likely to be verb. The dependent of conjunction is most likely to be noun.

These phenomena are found in all 3 networks which mean that they are unaffected by stylistic impact and are stable properties of this language. When we process the real language, it is a good choice to give these properties the priority, which can promote the efficiency avoiding random analysis.

We have also found several dissimilarities:

- (1) There are no mimetic word and interjection in valency pattern of X-treebank.
- (2) The value of arcs involving pronoun is the smallest in X-treebank while it is largest in S-treebank. It means that pronoun is sensitive to language styles.
- (3) In X-treebank, the probability of auxiliary linking with noun is higher than that of auxiliary linking with verb. It is opposite to the valency pattern of S-treebank. It is also may be seen as the different property between language styles.

These results prove that some word classes could show different valence patterns in texts with different styles. If we want to describe the valency pattern accurately, we should find a way to reduce the stylistic impact. So A-treebank may present a more reliable valence pattern of word classes. From the data, we can

see that mimetic word, interjection, pronoun and auxiliary are more sensitive to the stylistic impact. It shows that network approach and PVP may be able to provide some effective parameters for Chinese style research.

Conclusion

With three dependency networks, we have found that the valence pattern can be affected by style; simultaneously we investigated the similarities and differences among them. This work is trying to study the valence patterns of the language from an overall perspective and compare different valence patterns then figure out the real structure of language systems. It is different from traditional statistical works on words or word classes, for example collocation extraction from tree banks etc., which pay more attention to some specific structures.

In the study, we found that the language network approach and PVP are beneficial to each other. PVP can explain the language network data, such as the node, arc, value of arc, direction of arc, distance between nodes, etc. At the same time, as a method of language study, complex network can provide an intuitionistic but concise representation of data, which is easy to perceive and understand. However network approach also has some deficiencies. First of all, when the network size is large, the structure will become so complex that easy understanding is impossible. Secondly, although the network can easily provide an overview of the language, it usually fails to be much helpful when it comes into language details. For example, we cannot give the arcs qualitative descriptions in the network, which implies the loss of valuable information. Extracting valency patterns of word classes from syntactic complex networks is an interesting experiment. The integration of language network and PVP makes us believe that further research will bring more valuable results.

References

- Albert-László Barabási. 2002. *Linked*. Cambridge, Perseus Publishing.
- Ferrer i Cancho, R., R. Koehler and R. V. Solé. 2004. Patterns in Syntactic Dependency Networks. *Physical Review E*, 69.
- Fischer, K. 1997. *German-English Verb Valency*. Tübingen, Gunter Narr Verlag.

Helbig, G. and Schenkel, W. 1978. *Wörterbuch zur Valenz und Distribution deutscher Verben*. Leipzig, Bibliographisches Institut.

Helbig, G. 2002. *Linguistische Theorien der Moderne*. Berlin, Weidler Buchverlag.

Hudson, R. 2004. *An Encyclopedia of English Grammar and Word Grammar*. <http://www.phon.ucl.ac.uk/home/dick/enc-gen.htm>.

Hudson, R. 2007. *Language Networks: The New Word Grammar*. Oxford, Oxford University Press.

Liu, H. 2006. Syntactic Parsing Based on Dependency Relations. *Grkg/Humankybernetik*, 47:124-135.

Liu, H. 2010. Language Clusters based on Linguistic Complex Networks. *Chinese Science Bulletin*, 55.

Liu, H. 2011. Linguistic Networks: Metaphor or Tool? *Journal of Zhejiang University (Humanities and Social Science)*. 41: 169-180.

Liu, H. and Huang, W. 2006. A Chinese Dependency Syntax for Treebanking. *Proceedings of The 20th Pacific Asia Conference on Language, Information and Computation*. Beijing, Tsinghua University Press, 126-133.

Mel'čuk, I. 2003. Levels of Dependency in Linguistic Description: Concepts and Problems. In V. Agel, L. Eichinger, H.-W. Eroms, P. Hellwig, H. J. Herringer, H. Lobin (eds): *Dependency and Valency. An International Handbook of Contemporary Research*, vol. 1. Berlin - New York: W. de Gruyter, 188-229.

Tesnière, L. 1959. *Éléments de syntaxe structurale*. Paris, Klincksieck.

Yu, S., Liu, H. and Xu, C. 2011. Statistical Properties of Chinese Phonemic Networks. *Physica A*, 390.

Appendix A. codes meaning

code	meaning
d	adverb
r	pronoun
m	numeral
a	adjective
u	auxiliary
c	conjunction
p	preposition

q	classifier
v	verb
n	noun
o	mimetic word
e	interjection
zmen	“们” men ‘kind of suffix’
zdi	“第” di ‘kind of prefix’
bjd	punctuation
subj	subject
s	main governor
qc	complement of classifier
atr	attributer
obj	object



Delimitation of information between grammatical rules and lexicon

Jarmila Panevová, Magda Ševčíková

Charles University in Prague

Faculty of Mathematics and Physics

Institute of Formal and Applied Linguistics

{panevova|sevcikova}@ufal.mff.cuni.cz

Abstract

The present paper contributes to the long-term linguistic discussion on the boundaries between grammar and lexicon by analyzing four related issues from Czech. The analysis is based on the theoretical framework of Functional Generative Description (FGD), which has been elaborated in Prague since 1960's. First, the approach of FGD to the valency of verbs is summarized. The second topic, concerning dependent content clauses, is closely related to the valency issue. We propose to encode the information on the conjunction of the dependent content clause as a grammatical feature of the verb governing the respective clause. Thirdly, passive, resultative and some other constructions are suggested to be understood as grammatical diatheses of Czech verbs and thus to be a part of the grammatical module of FGD. The fourth topic concerns the study of Czech nouns denoting pair body parts, clothes and accessories related to these body parts and similar nouns. Plural forms of these nouns prototypically refer to a pair or typical group of entities, not just to many of them. Since under specific contextual conditions the pair/group meaning can be expressed by most Czech concrete nouns, it is to be described as a grammaticalized feature.

1 Introduction

Theoretical approaches to natural languages, regardless of which particular theory they subscribe to, usually work with grammar and lexicon as two basic modules. The delimitation between these modules

is not given by the language itself, the “balance” between the modules is “entirely an empirical issue” (Chomsky, 1970). There are core grammatical and lexical topics, such as agreement or lexical meaning, respectively, whose classification as belonging to the grammar on the one hand and to the lexicon on the other is shared across languages and linguistic theories, while classification of borderline cases as either grammatical or lexical ones is strongly theory-dependent.

A brief overview of selected approaches laying more stress either on the lexical or on the grammatical module is given in Sect. 2 of the present paper; the approach of Functional Generative Description, used as the theoretical framework of our analysis, is briefly presented. In Sect. 3 to 6, the delimitation of information between the two modules is exemplified by four topics, which have been studied for Czech.

2 Grammar vs. lexicon in selected theoretical approaches

The interplay between grammar and lexicon has been discussed for decades in linguistics. Although the former or the latter module plays a predominant role in particular frameworks, the preference of one of the modules does not mean to exclude the other one from the description, they are both acknowledged as indispensable.

According to Bloomfield (1933), the lexicon has a subordinated position.¹ The grammatical rules are the main component either within Chomskyan generative (transformational) grammar, though the im-

¹“The lexicon is really an appendix of the grammar, a list of basic irregularities.” (Bloomfield, 1933)

portance of the lexical component was strengthened by the decision to treat certain types of nominalizations within the lexicon rather than within the transformational (grammatical) component (Chomsky, 1970).

On the other side of the scale of grammatically vs. lexically oriented approaches,² there is the lexicalist approach of Meaning-Text Theory by Mel'čuk et al. Within this framework, a richly structured lexicon, so-called Explanatory Combinatorial Dictionary, has been systematically compiled for individual languages; the Dictionary is considered as a central component of description of language, cf. (Mel'čuk, 1988; Mel'čuk, 2006). Lexicon plays a crucial role in categorial grammars (Ajdukiewicz, 1935) as well as in the lexicalized tree adjoining grammar, see Abeillé – Rambow (2000), just to give two further (chronologically distant) examples.

Functional Generative Description (FGD) works with both grammatical and lexical modules since the original proposal of this framework in 1960's (Sgall, 1967); nevertheless, the main focus has been laid on grammatical, in particular syntactic, issues (Sgall et al., 1986). FGD has been proposed as a dependency-based description of natural language (esp. Czech). The meaning-expression relation is articulated here into several steps: the representations of the sentence at two neighboring levels are understood as the relation between form and function. The "highest" level (tectogrammatics) is a disambiguated representation of the sentence meaning, having the counterparts at lower levels. On the FGD framework the multi-layered annotation scenario of Prague Dependency Treebank 2.0 (PDT 2.0) has been built.

PDT 2.0 is a collection of Czech newspaper texts from 1990's to which annotation at the morphological layer and at two syntactic layers was added, namely at the layer of surface syntax (so-called analytical layer) and of deep syntax (layer of linguistic meaning, tectogrammatical layer) (Hajič et al., 2006).³ At the morphological layer each token is assigned a lemma (e.g. nominative singular for nouns) and a positional tag, in which the part of speech and formal-morphological categories are specified (e.g.

case, number etc. for nouns). At the analytical layer, the surface-syntactic structure of the sentence is represented as a dependency tree, each of the tree nodes corresponds exactly to one morphological token and is labeled as a subject or object etc. At the tectogrammatical layer, the linguistic meaning of the sentence is captured as a dependency tree, whose nodes correspond to auto-semantic words only.⁴ The nodes are labeled with a tectogrammatical lemma (which is often different from the morphological one), functor (semantic role, label; e.g. Actor ACT) and a set of grammatemes, which are node attributes capturing the meanings of morphological categories which are indispensable for the meaning of the sentence.⁵ The tectogrammatical representation is further enriched with valency annotation, topic-focus articulation and coreference.

The lexical issues have been becoming more central in the FGD framework for the recent ten years as two valency lexicons of Czech verbs based on the valency theory of FGD (Panevová, 1974/75) have been built; cf. the VALLEX lexicon (Lopatková et al., 2008) and the PDT-VALLEX (Hajič et al., 2003), which is directly interconnected with the tectogrammatical annotation of PDT 2.0.

The approach of FGD to valency is summarized in Sect. 3 of the present paper. The delimitation of information between grammar and lexicon in FGD is further illustrated by the description of dependent content clauses in Czech (Sect. 4), grammatical diatheses of Czech verbs (Sect. 5) and representation of a particular meaning of plural forms of Czech nouns (Sect. 6).

3 Valency

The problem of valency is one of most evident phenomenon illustrating the interplay of lexical and grammatical information in the language description. Lexical units are the bearers of the valency information in any known theoretical framework. The form of this information is of course theory-dependent in many aspects, first of all in (i) to (iii):

²For this opposition, the terms 'transformationalist' vs. 'lexicalist' approaches are used, the former ones are called also 'syntactic' or simply 'non-lexicalist' approaches, depending on the theoretical background.

³See also <http://ufal.mff.cuni.cz/pdt2.0>

⁴There are certain, rather technical exceptions, e.g. coordinating conjunctions used for representation of coordination constructions are present in the tree structure.

⁵Compare the related term 'grammems' in Meaning-Text Theory (Mel'čuk, 1988).

- (i) how the criteria for distinguishing of valency and non-valency complementations are determined,
- (ii) how the taxonomy of valency members looks like,
- (iii) how the relation between the deep valency labels (arguments in some frameworks, inner participants in FGD) are reflected (see also Sect. 4).

3.1 Valency approach of FGD

In FGD we use the valency theory presented in Panevová (1974/75; 1994) and Sgall (1998) and in its application in valency dictionaries (VALLEX, PDT-VALLEX). Valency complementations enter the valency frame as an obligatory part of lexical information. The empirically determined set of inner participants (Actor ACT, Patient PAT, Addressee ADDR, Origin ORIG and Effect EFF) and those free modification which were determined as semantically obligatory with the respective verb (by the criterion of grammaticality or by the so-called dialogue test see (Panevová, 1974/75)) are included in the valency frame.

FGD avoids the concept of a wide number of valency positions known, for instance, from the Meaning-Text Model (recently see Apresjan et al. (2010)), where e.g. the verb *vyzvat* ‘to call’ has six valency slots, cf. ex. (1). In FGD the valency frame of the corresponding verb *povolat* consists of three slots: ACT(Nom) PAT(Acc) DIR3.

- (1) *vyzvat*’ *kogo-libo iz Peterburga v Moskvu po telefonu na soveshchanije*
 ‘to call somebody from Petersburg to Moscow by phone for the meeting’

3.2 Valency in the lexicon and grammar

In FGD functors are defined with respect to their language patterning and to their position in the valency frame (a verb with two valency slots includes ACT and PAT, a verb with three valency slots includes ACT, PAT and its third position is determined according to its semantics). Inner participants are divided into obligatory and optional, this information is a part of the valency frame in the lexicon.

Any grammatical module, whatever its aim is (be it analysis, or generation, or having determinative or declarative character) is based on the combinatorial nature of the verb (as a center of the sentence) with its obligatory valency slots. If the valency requirements are not fulfilled, the sentence is in some sense wrong (either as to its grammaticality or as to its semantic acceptability). The surface deletions are checked by the dialogue test or by the contextual conditions.

4 Dependent content clauses in Czech

The next topic concerns the description of dependent content clauses in Czech. Dependent content clauses are object, subject and attributive clauses that express a semantic complementation of the particular governing verb (or noun, these cases are not addressed in the paper).

4.1 Dependent content clauses in FGD

Within the valency approach of FGD, dependent content clauses are inner participants of a verb, they (more precisely, main verbs of these clauses) are classified as PAT and EFF with most verbs, less often as ACT and rarely as ADDR or ORIG in the tectogrammatical structure of the sentence according to the PDT 2.0 data.

Dependent content clauses are connected with their governing verbs by subordinating conjunctions or by pronouns and pronominals (cf. ex. (2) and (3)).⁶ Pronouns and pronominals are considered as semantically relevant parts of the tectogrammatical sentence structure and thus represented as nodes in the tectogrammatical tree whereas subordinating conjunctions are not. Conjunctions introducing content clauses are listed within the dictionary entry of the particular verb in the PDT-VALLEX, VALLEX as well as in Svozilová et al. (1997).

- (2) *Rozhodl se, že zůstane.*
 Decided.3.sg.pst.anim REFL, that stays.3.sg.fut.
 ‘He decided to stay.’
- (3) *Otec často vyprávěl,*
 Father.nom.sg.anim often recounted.3.sg.pst.anim,
jak jezdíval.EFF autem.
 how drove.3.sg.pst.anim car.instr.sg.neut.
 ‘Father often recounted how he drove a car.’

⁶The Czech examples are translated literally first, followed by a standard English translation.

4.2 Modality in dependent content clauses

Since conjunctions introducing dependent content clauses correspond to the modality expressed by these clauses, it seems to be theoretically more adequate to interconnect the choice of the conjunction with the modality information and to handle it within the grammatical component of the linguistic description. A pilot study based on the valency theory of FGD was carried out by Kettnerová (2008), who suggested a classification of selected Czech verbs of communication into four classes of assertive, interrogative, directive and “neutral” verbs.

Though Kettnerová’s approach concerns just verbs of communication (and so do nearly all theoretical studies dealing with dependent content clauses in Czech, e.g. Bauer (1965)), Daneš (1985), according to our preliminary analysis of dependent content clauses in the tectogrammatical annotation of PDT 2.0 clauses of this type occur with a number of other verbs. Besides verbs of communication, a dependent content clause is governed by verbs classified as verbs of mental actions according to (Lopatková et al., 2008) (e.g. *dočíst se* ‘to learn by reading’, *přehodnotit* ‘to rethink’), further by verbs of “preventing” somebody or oneself from an action (*odradit* ‘to discourage’, *předejít* ‘to prevent’) and many other which do not share a common semantic feature (*usilovat* ‘to aim’, *divit se* ‘to be surprised’).

4.3 Interconnecting lexical and grammatical information

Aiming at an extension of Kettnerová’s approach, all the verbs governing a dependent content clause were analyzed in order to find a correspondence between the conjunctions introducing the respective dependent content clause and the modality expressed by this clause. As the dependent content clause is closely related to the meaning of the governing verb, the repertory of modality types of the dependent content clauses and the conjunctions used is mostly restricted:

Most of the analyzed verbs occurred only with a dependent content clause expressing assertive modality; assertive dependent content clauses are mostly introduced by the conjunction *že* ‘that’ (less often also by *jestli*, *zda*, *zdali* or *-li* ‘whether/if’ – see the next paragraph). Substantially fewer verbs

governed only either an imperative or an interrogative dependent content clause; imperative clauses are introduced by *aby* or *ať* ‘that’, the interrogative ones by *jestli*, *zda*, *zdali* or *-li* ‘whether/if’. Only with a restricted number of verbs dependent content clauses of more modality types (and thus with several introducing conjunctions) occurred, most of them belong to verbs of communication;⁷ the conjunctions corresponded to the modality in the same way as with verbs with dependent content clauses of only a single modality type.

However, since there are semantic nuances in the modality of the dependent content clauses that cannot be described by means of the common inventory of modality types, the inventory has to be extended or revised; cf. ex. (4) and (5) that both are classified as assertive but the difference between them consist in the fact that in the former example the content of the dependent content clause is presented as given and in the latter one as open.

- (4) *Ověříme, že robot*
Check.1.pl.fut, **that** robot.nom.sg.anim
místnost uklidil.
room.acc.sg.fem cleaned_up.3.sg.pst.anim.
‘We check **that** the robot had cleaned up the room.’
- (5) *Ověříme, zda robot*
Check.1.pl.fut, **whether** robot.nom.sg.anim
místnost uklidil.
room.acc.sg.fem cleaned_up.3.sg.pst.anim.
‘We check **whether** the robot had cleaned up the room.’

After a solution for this question is found, at least the following issues have to be discussed before modality of the dependent content clauses is included into the grammatical module of FGD:

- the differences between conjunctions introducing dependent content clauses of a particular modality type have to be determined in a more detailed way,
- the impact of the morphological characteristics of the governing verb on the modality of the dependent content clause should be clarified.

⁷They are classified as “neutral” verbs of communication by Kettnerová (2008).

5 Grammatical diatheses of Czech verbs

The number of the diathesis proposed for the modified version of FGD as well as for the new version of Prague Dependency Treebank (PDT 3.0) was broadened in comparison with the previous version and with the usual view considering namely passivization; see Panevová – Ševčíková (2010). We exemplify below three types of grammatical diatheses with their requirements on syntax, morphology and lexicon.

5.1 Passivization

Passivization is commonly described as a construction regularly derived from its active counterpart by a transformation or another type of grammatical rule. However, though this operation is productive, there are some constraints blocking it and requirements how the result of this diathesis looks like. It is very well known that, at least for Czech, passivization cannot be applied for intransitive verbs, moreover it is not applicable for some object-oriented verbs and for reflexive verbs in Czech (ex. (6) to (8), respectively).

(6) *spát* – **je* *spáno*
sleep.inf – is.3.sg.pres slept.nom.sg.neut
'to sleep' – 'it is slept'

(7) *přemýšlet* *o něčem* – **je*
think.inf about something.loc.sg.neut – is.3.sg.pres
přemýšleno *o něčem*
thought.nom.sg.neut about something.loc.sg.neut
'to think about something' – 'it is thought about something'

(8) *rozloučit* *se* – **je* *se*
say_good.bye.inf REFL – is.3.sg.pres REFL
rozloučeno
said_good_bye.nom.sg.neut
'to say good bye' – 'it is said good bye'

There are also constraints on passivization which are lexical dependent: Some verbs having direct object in Accusative cannot be passivized, e.g. *mít* 'to have', *znát* 'to know', *umět* 'to know' at all, with some verbs this constraint concerns only their imperfective form (e.g. *jíst* 'to eat', *potkat* 'to meet'). From the other side, the passivization is not restricted only on the verbs with direct object in Accusative (e.g. *věřit komu* 'to believe + Dat', *plýtvat čím* 'to waste + Instr', *zabránit čemu* 'to avoid + Dat').

The operation of passivization is based on the shift of the certain verbal participant to the position of the surface subject. However, reminding the theory of valency in FGD sketched briefly in Sect. 3, which participant is shifted, depends on the type of the verb. Sometimes it is PAT (ex. (9)), with other verbs it is ADDR (ex. (10)), or EFF (ex. (11)). These constraints and requirements concerning passivization have to be marked in the lexicon.⁸

(9) *vykopat jámu*.PAT – *jáma*.PAT
dig.inf hole.acc.sg.fem – hole.nom.sg.fem
je *vykopána*
is.3.sg.pres dug.nom.sg.fem
'to dig a hole' – 'the hole is dug'

(10) *informovat někoho*.ADDR *o*
inform.inf somebody.acc.sg.anim about
něčem – *někdo*.ADDR
something.loc.sg.neut – somebody.nom.sg.anim
je *o něčem*
is.3.sg.pres about something.loc.sg.neut
informován
informed.nom.sg.anim
'to inform somebody about something' – 'somebody is informed about something'

(11) *O zemětřesení* *napsal*
About earthquake.loc.sg.neut wrote.3.sg.pst.anim
reportáž.EFF – *O zemětřesení*
report.acc.sg.fem. – About earthquake.loc.sg.neut
byla *napsána*
was.3.sg.pst.fem written.nom.sg.fem
reportáž.EFF
report.nom.sg.fem.
'He wrote a report about an earthquake.' – 'A report was written about the earthquake.'

5.2 Resultative constructions

The other constructions understood as the grammatical category of diathesis are less productive than passivization, but they are produced by a regular grammatical operation, which fact points out to their systemic (grammatical) nature. Resultative constructions display this character in both of their forms: so-called objective (Giger, 2003) and possessive forms. The auxiliary *být* 'to be' and passive participle are used for the former type (ex. (12) and

⁸The technical means how to mark this information is left aside here. We can only say that the feature reflecting the relation between the type of the participant and the surface subject must be included in the lexicon.

(13)); the auxiliary *mít* ‘to have’ and passive participle constitute the latter type (ex. (14)).⁹

- (12) *Je otevířeno.*
 Is.3.sg.pres **opened**.nom.sg.neut.
 ‘It is **opened**.’
- (13) *Je zajiřřeno, ře*
 Is.3.sg.pres **arranged**.nom.sg.neut, that
dřkan na schůzi
 dean.nom.sg.anim for meeting.acc.sg.fem
přijde.
 come.3.sg.fut.
 ‘It is **arranged** that the dean will come for the meeting.’
- (14) *Dohodu o spolupřáci*
 Agreement.acc.sg.fem about cooperation.loc.sg.fem
uř máme podepsánu.
 already have.1.p.presl **signed**.acc.sg.fem.
 ‘We **have** an agreement about the cooperation **signed**.’

The form for objective resultative is ambiguous with the passive form. However, the slight semantic difference is reflected here by the grammateme values passive vs. resultative1 (see Table 1); cf. the ex. (15) and (16):

- (15) *Bylo navřřeno, aby se*
 Was.3.sg.pst.neut **proposed**.nom.sg.neut, that REFL
o změně zákona
 about change.loc.sg.fem law.gen.sg.inan
hlasovalo ihned.
 voted.3.sg.cond.neut immediately.
 ‘It **was proposed** to vote about the change of the law immediately.’
- (16) *Tento zákon se pořád*
 This.nom.sg.inan law.nom.sg.inan REFL still
pouřívá, ačkoli byl
 uses.3.sg.pres, though was.3.sg.pst.inan
navřřen uř dávno.
proposed.nom.sg.inan already long_time_ago.
 ‘This law is still used, though it **has been proposed** long time ago.’

These constructions are rarely used for intransitives and for verbs in imperfective aspect (ex. (17) and (18)) (Načeva Marvanová, 2010). The syntactic rules for the objective resultative are based either on the shift of the PAT into the position of the surface

subject, or on the surface deletion of the PAT. In the possessive form either ACT or ADDR converts into surface subject.¹⁰ The mark about compatibility of the verb with the resultative meanings (grammateme values resultative1, resultative2) must be introduced into the lexicon, see Table 1.

- (17) *má nakročeno*
 has.3.sg.pres **stepped forward**.nom.sg.neut
 ‘he **has stepped forward**’
- (18) *Toto území máme*
 This.acc.sg.neut area.acc.sg.neut **have**.1.pl.pres
chráňeno před povodněmi.
protected.acc.sg.neut against floods.instr.pl.fem.
 ‘We **have** this area **protected** against the flood.’

5.3 Recipient diathesis

The recipient diathesis is a more limited category than the resultativeness, see also Daneř (1968), Panevová – řevčříková (2010) and Panevová (in press); however, it is again a result of the syntactic process constituting a recipient paradigm. An ADDR of the verb is shifted to the position of surface subject, the auxiliary verb *dostat* ‘to get’, marginally *mít* ‘to have’ with passive participle are used in recipient diathesis (ex. (19)). The semantic features of verbs (such as verb of giving, permitting etc.) are responsible for the applicability of this diathesis rather than the presence of ADDR in the verbal frame (ex. (20)). The mark about a possibility to apply the recipient diathesis will be a part of the lexical information within the respective verbs (see Table 1).

- (19) *Pavel dostal za*
 Paul.nom.sg.anim **got**.3.sg.pst.anim for
posudek zapláceno.
 review.acc.sg.inan **payed**.acc.sg.neut.
 ‘Paul **got payed** for the review.’
- (20) *řřkal někomu*
told.3.sg.pst.anim somebody.dat.sg.anim
*něco – *dostal*
 something.acc.sg.neut – **got**.3.sg.pst.anim
něco řřčeno
 something.acc.sg.neut **told**.acc.sg.neut
 ‘he **told** somebody something’ – ‘he **got told** something’

⁹These constructions studied from the contrastive Slavic view are called “new perfect constructions” by Clancy (2010); see, however, the description of these constructions given by Mathesius (1925).

¹⁰The shift of the ADDR into the subject position is a syntactic operation and it is not connected with the participant shifting; however, the subject has a function of the possessor of the resulting event rather than an agentive role.

PLATIT-1 ‘TO PAY-1’	
Formal morphology:	Vf - - - - - A - - - -
Aspect:	processual (→ complex ZAPLATIT)
Grammatemes:	+passive: PAT ^{Sb} +resultative1, +resultative2, +recipient
Reflexivity:	cor3
Reciprocity:	ACT – ADDR
Valency frame:	ACT(Nom) (ADDR(Dat)) PAT(Acc) (EFF(od+Gen/za+Acc))
Semantic class:	exchange
ŘÍCI-1 ‘TO TELL-1’	
Formal morphology:	Vf - - - - - A - - - -
Aspect:	complex (→ processual ŘÍKAT)
Grammatemes:	+passive: EFF ^{Sb} +resultative1, -resultative2, -recipient
Reflexivity:	cor3
Reciprocity:	ACT – ADDR
Valency frame:	ACT(Nom) ADDR(Dat) (PAT(o+Loc)) EFF(4/V-ind(assert/deliber),imper)
Semantic class:	communication

Table 1: Examples of lexical entries in the lexicon – a preliminary proposal

5.4 Grammatical diatheses in the lexicon and grammar

The exemplified diatheses belong to the grammatemes in FGD, representing the morphological meanings of specific analytical verbal forms. They differ from fully grammaticalized paradigmatic verbal categories (such as verbal tense or mood) in this respect that they use for their constitution not only morphemic means, but also syntactic operations. Due to this fact they are not applicable for all verbs and for their application a lexical specification in the lexicon is needed as well as general syntactic rules in the grammatical module.

Examples of lexical entries according to the suggestions in Sect. 3 to 5 are given in Table 1. In the Table the following notation is used:

- the number accompanying the lemma delimits the particular meaning of an ambiguous lexical item,
- formal morphological features of the lemma are described by the positional tag,
- processual and complex are the tectogrammatical values of the grammateme aspect corresponding to the imperfective and perfective aspect, respectively; a link to the aspectual counterpart is included,
- +/- with a grammateme value indicate the non/applicability of this value,
- for passive the participant converting into subject of the passive construction (if present) is marked as the upper index,

- resultative1, resultative2, and recipient (objective resultative, possessive resultative, and recipient, respectively) are proposed as the values of selected grammatical diatheses whose non/applicability is expressed with +/-,
- the reflexivity value cor3 means that reflexive binding ACT – ADDR is possible with the verb,
- the reciprocity ACT – ADDR is to be interpreted as indicating that syntactic operation of reciprocity can be applied if ACT and ADDR play both roles in this event,
- as for the valency frame, obligatory participants are without brackets, optional participants are in brackets; the morphemic realization of noun participants in brackets is attached to the functor; the verbal participant filling the role of PAT or EFF is denoted as V with the possible modalities compatible with the verb (assert – factual indicative, *deliber* – nonfactual indicative, *imper* – imperative).

6 Pair/group meaning of Czech nouns

The last issue that we use to exemplify the interconnection of grammar and lexicon in FGD is related to the category of number of Czech nouns. For this category some “irregularities” occur in their paradigms. The pluralia tantum nouns as *nůžky* ‘scissors’, *brýle* ‘glasses’ perform the formal deviation – they use the same form (plural) for singular as well as for the plural. They constitute a closed class and the

feature for their number deviation must be included into the morphological zone of their respective lexical entry. With the nouns denoting collectives such as *listí* ‘leaves’, *mládež* ‘young people’ the plural forms are semantically excluded; however, they represent again the closed class, so that their semantic deviation have to be included into the semantic zone of their respective lexical entry as a feature blocking the plural value of the grammateme number in their prototypical usage.

6.1 Nouns with pair/group meaning

There are many nouns in Czech that refer by their plural forms prototypically to a pair or typical group of entities and not just to many of them, which is acknowledged as the ‘proper’ meaning of plurals in Czech. This is the case for nouns denoting the human body parts occurring in pairs or typical groups (e.g. *ruce* ‘hands’, *prsty* ‘fingers’), nouns denoting clothes and accessories related to these body parts (*ponožky* ‘socks’), further nouns denoting objects used or sold in collections or doses, such as *klíče* ‘keys’ and *sirky* ‘matches’.

In contrast to other languages (Corbett, 2000), in Czech the pairs or groups are expressed by common plural forms of these nouns, these nouns are not formally marked for the pair/group meaning. However, the pair/group meaning manifests in the form of the numeral in Czech. When denoting pair(s) or group(s) of entities, the nouns are compatible with so-called set numerals only (cf. *jedny ruce* ‘a pair of hands’, *patery sirky* ‘five boxes of matches’), while if they refer simply to a number of entities, they are accompanied with cardinal numerals (*dvě rukavice* ‘two gloves’, *pět sirek* ‘five matches’).

The primary meaning of the set numerals is to express different sorts of the entities denoted by the noun (cf. *dvoje sklenice – na bílé a červené víno* ‘two sets of glasses – for the white and red wine’). However, the same set numerals, if combined with pluralia tantum nouns, express either the amount of single entities (i.e. the same meaning which is expressed by cardinal numerals with most nouns), or the number of sorts, cf. *troje nůžky* ‘three types/pieces of scissors’. The set numerals in combination with the nouns which we are interested in in the present paper express the number of pairs or groups; it means that the set numerals are used

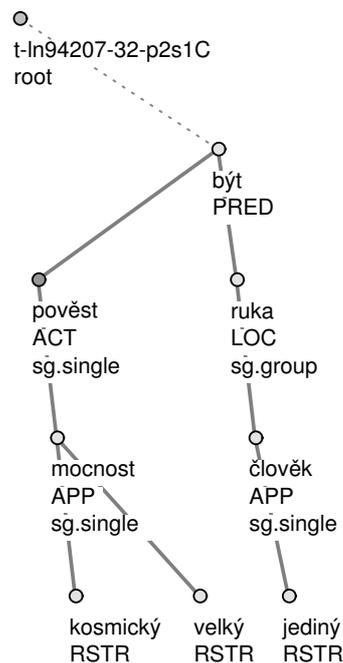


Figure 1: Tectogrammatical tree of the sentence *Pověst velké kosmické moci je v rukách jediného člověka*. ‘Reputation of the big space power is in the hands of a single man.’ For each node the tectogrammatical lemma, the functor and the values of the number and typgroup grammemes are given.

here instead of cardinal numerals while the cardinals combined with these nouns express the number of single entities (cf. *troje boty* ‘three pairs of shoes’ vs. *tři boty* ‘three shoes’).

When considering how to treat the pair/group meaning within the framework of FGD, the fact was of crucial importance that this meaning is still connected with a list of typical pair/group nouns in Czech but not limited to them: If a set numeral co-occurs, the pair/group meaning can be expressed by most Czech concrete nouns, cf. ex. (21).¹¹

- (21) *Najdeme-li dvoje velké*
 Find.1.pl.fut-if two.sets big.acc.pl.fem
stopy a mezi nimi
 traces.acc.pl.fem and between them.instr.pl.fem

¹¹Unlike the common co-occurrence of set numerals with nouns for which the pair/group meaning is not frequent, for the above mentioned nouns *ruce* ‘hands’, *klíče* etc. the ‘bare’ plural form is commonly interpreted as pair(s)/group(s) and the set numeral is used only if the concrete number of pairs or groups is important.

jedny menší, řekneme si:
one_set smaller.acc.pl.fem, say.1.pl.fut REFL:
rodina na výletě.
 family.nom.sg.fem on trip.loc.sg.inan.
 ‘If we find **two sets of big traces** and **one set of smaller ones** between them, we say: a family on a trip.’

6.2 Pair/group meaning as a grammaticalized feature

This fact has led us to the decision to treat the pair/group meaning as a grammaticalized category, namely as a special grammatical meaning of the plural forms of nouns (besides the simple plural meaning of several single entities), and to include it into the grammatical component of the description. If we had decided to capture the pair/group meaning as a lexical characteristic, it would have implied to split lexicon entries (at least) of the prototypical pair/group nouns into two entries, an entry with a common singular–plural opposition and an entry for cases in which the plural of the noun denotes pair(s) or group(s); the potential compatibility of the pair/group meaning with other nouns, though, would have remained unsolved. The economy of the lexicon seems to be the main advantage that can be achieved when preferring our solution to the lexicalist one in this particular case.

As a (newly established) grammatical meaning, the pair/group meaning has been introduced as a new grammateme *typgroup* in the grammatical module of FGD. For this grammateme three values were distinguished: *single* for noun occurrences denoting a single entity or a simple amount of them, *group* for cases in which pair(s) or group(s) are denoted, and *nr* for unresolvable cases.

The *typgroup* grammateme is closely related to the number grammateme (values *sg*, *pl*, *nr*). The following six combinations of a value of the grammateme number (given at the first position) with a value of the grammateme *typgroup* (at the second position) are possible according to the pilot manual annotation of the pair/group meaning carried out on the tectogrammatically annotated data of PDT 2.0: *sg-single*, *pl-single*, *sg-group*, *pl-group*, *nr-group*, and *nr-nr*, cf. ex. (22) to (27), respectively, and the tectogrammatical tree in Fig. 1.

- (22) *Na stole leží kniha*.sg-single
 On table.loc.sg.inan lies.3.sg.pres **book**.nom.sg.fem.
 ‘A **book** lies on the table.’
- (23) *Na stole leží knihy*.pl-single
 On table.loc.sg.inan lie.3.pl.pres **books**.nom.pl.fem.
 ‘**The books** lie on the table.’
- (24) *Namaloval to vlastníma rukama*.sg-group
 Draw.3.sg.pst.anim it own.instr.pl.fem
hands.instr.pl.fem.
 ‘He draw it by his **hands**.’
- (25) *Děti, zujte si boty*.pl-group!
 Kids.voc.pl.fem, take_off.2.pl.imp REFL
shoes.acc.pl.fem!
 ‘Kids, take off your **shoes**!’
- (26) *Vyčistil si boty*.nr-group
 Cleaned.3.sg.pst.anim REFL **shoes**.acc.pl.fem.
 ‘He has cleaned his **shoes**.’
- (27) *Odnes boty do opravy!*
 Take.2.sg.imp **shoes**.acc.pl.fem to repair.gen.sg.fem!
 ‘Take **the shoes** to a repair!’

7 Conclusions

In the present contribution we tried to illustrate that the balanced interplay between the grammatical and the lexical module of a language description is needed and document it by several concrete examples based on the data from the Czech language. In Sect. 3 the valency was introduced as an issue that must be necessarily included in the lexical entry of particular words; however, the valency is reflected in the grammatical part of the description as well, where the obligatoriness, optionality, surface deletions etc. must be taken into account.

Content clauses as valency slots of a special kind need a special treatment: The verbs governing content clauses classified as PAT or EFF require certain modality in the dependent clause (assertive, imperative, interrogative expressed on the surface by the subordinated conjunctions), only few verbs are compatible with more than one type of modality, e. g. *říci* ‘to tell’. These requirements (sometimes modified by morphological categories of the respective governor) are a part of valency information in the lexicon, while the rules for their realization by the conjunctions *že*, *zda*, *jestli*, *aby*, *ať* are a part of the grammar (Sect. 4).

Selected grammatical diathesis as a type of meanings of the verbal morphological categories are analyzed as to the constraints on their constitution (as a piece of information to be included in the lexicon) as well as to the regular syntactic operations applied on their participants (as a part of grammar; see Sect. 5). The arguments for an introduction of a new morphological grammateme (*typgroup*) connected with the category of the noun number are presented in Sect. 6. This meaning (with values *single* vs. *set*) is considered to be a grammaticalized category rather than a lexical characteristic of typical pair/group nouns.

Our considerations presented in Sect. 3 to 6 must be reflected within the technical apparatus of FGD both in the realization of lexical entries within the lexicon and in the shape of grammatical rules within the corresponding module.

Acknowledgments

The research reported on in the present paper was supported by the grants GA ČR P406/2010/0875, GA ČR 405/09/0278, and MŠMT ČR LC536.

References

- Anne Abeillé – Owen Rambow (eds.). 2000. *Tree Adjoining Grammars: Formalisms, Linguistic Analysis and Processing*. Center for the Study of Language and Information, Stanford.
- Kazimierz Ajdukiewicz. 1935. Die syntaktische Konnexität. *Studia Philosophica* I, 1–27.
- Jurij D. Apresjan – Igor' M. Boguslavskij – Leonid L. Iomdin – Vladimir Z. Sannikov. 2010. *Teoreticheskie problemy russkogo sintaksisa. Vzaimodejstvie grammatiki i slovarja*. Jazyki slavjanskich kul'tur, Moskva.
- Jaroslav Bauer. 1965. Souvětí s větami obsahovými. *Sborník prací filosofické fakulty brněnské university*, A 13:55–66.
- Leonard Bloomfield. 1933. *Language*. Henry Holt, New York.
- Noam Chomsky. 1970. Remarks on Nominalization. In *Readings in English Transformational Grammar*. Ginn and Company, Waltham, Mass., 184–221.
- Steven J. Clancy. 2010. *The Chain of Being and Having in Slavic*. Benjamins, Amsterdam – Philadelphia.
- Greville G. Corbett. 2000. *Number*. Cambridge University Press, Cambridge.
- František Daneš. 1968. Dostal jsem přidáno a podobné pasivní konstrukce. *Naše řeč*, 51:269–290.
- František Daneš. 1985. *Věta a text*. Academia, Praha.
- Markus Giger. 2003. *Resultativa im modernen Tschechischen*. Peter Lang, Bern – Berlin.
- Jan Hajič – Jarmila Panevová – Eva Hajičová – Petr Sgall – Petr Pajas – Jan Štěpánek – Jiří Havelka – Marie Mikulová – Zdeněk Žabokrtský – Magda Ševčíková Razímová. 2006. *Prague Dependency Treebank 2.0*. CD-ROM. Linguistic Data Consortium, Philadelphia.
- Jan Hajič – Jarmila Panevová – Zdeňka Uřešová – Alevtina Bémová – Veronika Kolářová – Petr Pajas. 2003. PDT-VALLEX: Creating a Large-coverage Valency Lexicon for Treebank Annotation. In *Proceedings of the 2nd Workshop on Treebanks and Linguistic Theories*. Vaxjo University Press, Vaxjo, 57–68.
- Václava Kettnerová. 2008. Czech Verbs of Communication with Respect to the Types of Dependent Content Clauses. *Prague Bulletin of Mathematical Linguistics*. 90:83–108.
- Markéta Lopatková – Zdeněk Žabokrtský – Václava Kettnerová et al. 2008. *Valenční slovník českých sloves*. Karolinum, Praha.
- Vilém Mathesius. 1925. Slovesné časy typu perfektního v hovorové češtině. *Naše řeč*, 9:200–202.
- Igor A. Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. University of New York Press, New York.
- Igor A. Mel'čuk. 2006. Explanatory Combinatorial Dictionary. In *Open Problems in Linguistics and Lexicography*. Polimetrica, Monza, 225–355.
- Mira Načeva Marvanová. 2010. *Perfektum v současné češtině*. NLN, Praha.
- Jarmila Panevová. 1974/75. On Verbal Frames in Functional Generative Description. *Prague Bulletin of Mathematical Linguistics*, 22:3–40, 23:17–52.
- Jarmila Panevová. 1994. Valency Frames and the Meaning of the Sentence. In *The Prague School of Structural and Functional Linguistics*. Benjamins, Amsterdam – Philadelphia, 223–243.
- Jarmila Panevová. In press. O rezultativnosti (především) v češtině. In *Zbornik Matice srpske*. Matice srpska, Novi Sad.
- Jarmila Panevová – Magda Ševčíková. 2010. Annotation of Morphological Meanings of Verbs Revisited. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*. ELRA, Paris, 1491–1498.
- Petr Sgall. 1967. *Generativní popis jazyka a česká deklinace*. Academia, Praha.
- Petr Sgall. 1998. Teorie valence a její formální zpracování. *Slovo a slovesnost*, 59:15–29.
- Petr Sgall – Eva Hajičová – Jarmila Panevová. 1986. *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. Reidel – Academia, Dordrecht – Praha.
- Nad' a Svozilová – Hana Prouzová – Anna Jirsová. 1997. *Slovesa pro praxi*. Academia, Praha.

The Word Order of Inner Participants in Czech, Considering the Systemic Ordering of Actor and Patient

Kateřina Rysová

Charles University in Prague,
Faculty of Mathematics and Physics,
Institute of Formal and Applied Linguistics
katerina.rysova@post.cz

Abstract

The paper deals with the unmarked word order (systemic ordering) of inner participants (Actor and Patient) in the focus-part of Czech sentences. The analysis of the sequence of Actor and Patient reveals the criteria that may influence the arrangement of sentence participants as such.

1 The word order in Czech – systemic ordering

The present paper aims at an analysis of one of the basic properties of the sentence structure in Czech, namely the unmarked word order of sentence constituents focusing on inner participants (arguments) Actor and Patient.

Czech is a language with the so-called free word order. However, this does not mean that the word order is arbitrary; rather, it is not grammatically fixed to such an extent as the word order in English. Therefore, the word order in Czech has a great opportunity to copy the topic-focus articulation (functional sentence perspective / sentence information structure).

In the unmarked word order in Czech, the contextually bound sentence elements appear first (in the topic-part of the sentence) followed by the contextually non-bound elements in the focus-part. The last member in the sentence is usually the very “core” of communication (focus proper), i.e. the element carrying the most important information (the greatest degree of communicative dynamism) and also the lowest degree of identifiability from the context (whether linguistic or situational), cf. Petr Sgall, Eva Hajičová and Eva Buráňová (1980, p. 17). It is thus the context that is the strong factor affecting the word order in the Czech sentence (*Mluvnice češtiny 3*, 1987, p. 582).

The elements in the focus-part of the sentence are mostly contextually non-bound. However, their sequence is not arbitrary here. It seems that the order of sentence constituents in the focus is subject to certain principles and is probably influenced to some extent by grammatical factors.

The research on focus-part of the Czech sentences in terms of word order (i.e. research on the so-called systemic ordering) was carried out by Praguian generative linguists Petr Sgall, Eva Hajičová and Eva Buráňová (1980). They have formulated the hypothesis that there exists a canonical ordering of verb participants and circumstantials and the tentative ordering they proposed is as follows (1980, p. 77):

Actor ACT – time (when) TWHEN – since when TSIN – to when TTILL – how often THO – how long THL – location (where) LOC – manner MANN – criterion CRIT – instrument MEANS – direction (which way) DIR2 – addressee ADDR – origin ORIG – direction (from where) DIR1 – patient PAT – direction (to where) DIR3 – effect EFF – condition COND – aim (purpose) AIM – reason (cause) CAUS.

The scale was established on the basis of an empirical study of Czech texts complemented by psycholinguistic tests carried out with native speakers of Czech. The authors assume that it is the kind of sentence participants or circumstantials (rather than the choice by the author) that has the greatest influence on the placement of the sentence element in the scale (P. Sgall et al. 1980, p. 69). At the same time they highlight the fact that the systemic ordering may interfere with other factors as well (not taken into account yet), such as clause or non-clause form of participants (1980, p. 76), so that not all realized sentences in real texts must copy the

established scale in their focus-part. This was confirmed in the research by Šárka Zikánová (2006).

2 Verifying the systemic ordering on data from the Prague Dependency Treebank

The aim of this paper is to verify a part of that scale. Our attention is focused on the order of inner participants (Actor and Patient) with regard to each other (Actor – Patient / Patient – Actor) and also against the other inner participants (Addressee, Origin, Effect) and against the so-called free verbal modifications (such as Cause, Condition, Aim, Locative, Manner etc.) – e.g. Actor – Locative / Locative – Actor.

The research was conducted on data from the Prague Dependency Treebank (PDT) which includes more than 38,000 sentences annotated on tectogrammatical (i.e. underlying syntactic) layer. The corpus consists of journalistic texts, so that the conclusions of the research mainly apply to sentences from the texts of journalistic style.

In the analysis, only positive declarative sentences were collected since we assume that the type of the sentence or the use of negation may influence the results. Moreover, only participants that have not the form of clauses were included into the research (in contrast to the original scale of system ordering that ignored a possible difference in the behaviour of participants expressed by clauses and non-clauses). At the same time, the sentence elements had to be contextually non-bound. To decide whether a participant is or is not contextually bound, the annotation of topic-focus articulation in PDT was used (for the annotation instructions for the assignment of the values of the attribute of topic-focus articulation in PDT see Marie Mikulová et al. 2005, pp. 142ff). The monitored participants also had to be explicitly present in the sentence (in the surface structure). Unexpressed constituents present only implicitly (in the underlying structure of sentences) were not taken into account.

It was then tested, for inner participants Actor and Patient pairwise, which order is more common – whether Actor – Patient or Patient – Actor. In addition, we

examined the common sequence of each inner participant in combination with other inner participants (Addressee, Origin and Effect) and with a free verbal modification (e.g. Condition, Aim, Locative, Manner etc.). The analysis followed the position of Actor and Patient in pairs with all free verbal modifications which the corpus PDT distinguishes (there are almost 40 types of them, see M. Mikulová et al. 2005, pp. 114ff). The number of occurrences of pairs in the two sequences was recorded in a table.

It is natural that some types of sentence participants or circumstantials occurred more frequently in the corpus (e.g. Actor, Patient, Locative) and some others (especially those with more specific semantic characteristics) occur less often (e.g. Heritage, Obstacle). This fact is also reflected in the frequency of the occurrence of some participants in pairs – for some pairs, there were not found any sentences in the corpus where the participants would appear side by side (under the given conditions). The research results include only those pairs that appeared in PDT (under the given conditions) at least in 10 cases (the tables of systemic ordering are, therefore, different in size for Actor and for Patient).

3 Research results

The tables summarizing the results of research reflect the frequency of inner participants Actor and Patient in a particular position in relation to other sentence elements. The first column of each table indicates the type of the participant (its functor); for the abbreviations and characteristics of sentence elements used in PDT see <http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/t-layer/html/ch07.html>.

In the second column, there is the number of co-occurrences of a given inner participant and another type of functor in the order “functor – inner participant” / “inner participant – functor”. The third column contains the probability that the systemic ordering is in the PDT in the sequence “inner participant – functor”. This probability was calculated from samples of different sizes – by small samples the probability has only an informative value and its importance should not be overestimated.

E.g. inner participant Actor (ACT) occurred in the corpus PDT (under the given conditions described above) with the free verbal modification expressing Manner (MANN) in 256 cases. In 213 occurrences ACT and MANN appeared pairwise in the order MANN – ACT. In the order ACT – MANN they occurred in 52 cases. The probability that this pair will appear in order ACT – MANN is 52/265, i.e. 0.20.

Research results are reflected in the following tables¹:

Funcutor (*)	*-ACT / ACT-*	P (ACT-*)
RESTR	23 / 2	0.08
MANN	213 / 52	0.20
THL	34 / 12	0.26
EXT	105 / 41	0.28
THO	30 / 13	0.30
TWHEN	267 / 109	0.30
CRIT	32 / 14	0.30
TSIN	14 / 7	0.33
LOC	241 / 152	0.39
TTILL	8 / 6	0.43
PAT	615 / 486	0.44
DIR1	14 / 13	0.48
DIR2	5 / 5	0.50
TPAR	5 / 5	0.50
DIR3	36 / 38	0.51
ADDR	38 / 49	0.56
COND	9 / 12	0.57
MEANS	23 / 34	0.60
CAUS	12 / 19	0.61
EFF	15 / 24	0.62
ORIG	4 / 7	0.64
AIM	7 / 13	0.65
REG	6 / 11	0.65

¹ ACMP accompaniment; ACT actor; ADDR addressee; AIM purpose; BEN sth is happening for the benefit (or disadvantage) of sb/sth; CAUS cause; COMPL predicative complement; COND condition; CRIT criterion/measure/standard; DIFF difference (between two entities, states etc.); DIR1 direction: from where; DIR2 direction: which way; DIR3 direction: to where; EFF effect; EXT extent; LOC locative: where; MANN manner; MEANS means (of doing sth); ORIG origin; PAT patient; REG with regard to what sth is asserted; RESL result of sth; RESTR exception / restriction; SUBS sb/sth substitutes for sb/sth else; TFHL temporal: for how long; THL temporal: how long / after how long; THO temporal: how often / how many times; TPAR in parallel/simultaneously with what / during what time; TSIN temporal: since when; TTILL temporal: until when.

BEN	11 / 23	0.68
ACMP	15 / 34	0.69
COMPL	12 / 27	0.69
DIFF	0 / 11	1.00

Table 1 Systemic ordering with regard to ACTOR

Funcutor (*)	*-PAT / PAT-*	P (PAT-*)
RESL	16 / 2	0.11
THL	120 / 22	0.15
EXT	282 / 53	0.16
MANN	643 / 125	0.16
RESTR	25 / 8	0.24
TWHEN	465 / 165	0.26
TSIN	34 / 14	0.29
CRIT	55 / 22	0.29
THO	68 / 30	0.31
ADDR	229 / 113	0.33
REG	60 / 35	0.37
LOC	383 / 276	0.42
BEN	77 / 55	0.42
TPAR	11 / 8	0.42
TTILL	29 / 22	0.43
ORIG	51 / 43	0.46
TFHL	10 / 9	0.47
COMPL	62 / 63	0.50
DIR1	45 / 49	0.52
MEANS	87 / 98	0.53
CAUS	42 / 49	0.54
SUBS	5 / 6	0.55
ACT	486 / 615	0.56
ACMP	48 / 73	0.60
DIR3	96 / 145	0.60
COND	19 / 41	0.68
DIFF	14 / 31	0.69
EFF	66 / 160	0.71
AIM	13 / 58	0.82

Table 2 Systemic ordering with regard to PATIENT

The tables reflect a certain degree of probability that a given contextually non-bound sentence element (inner participant or free modification) expressed by non-clause will follow a contextually non-bound inner participant (Actor and Patient) which is expressed also by non-clause form². As noted

² The tables reflect only the probability of particular sentence elements to appear 1. after the Actor 2. after the Patient in the sentence. They do not show the word order of the verbal participants or

above, this probability of the word order “inner participant – other sentence element” concerns the positive declarative sentence from the journalistic text in Czech.

In some cases, it was possible to explore a relatively large sample of sentences (up to several hundred). Such a sample certainly reflects some features of primary word order of sentence components but the results can not be found absolute. The order of inner participants may be affected also by other criteria (for the time being, they are disregarded here, see below).

The results indicate that in some cases, we can actually observe a stronger or weaker tendency to a certain sequence of verbal participants or circumstantials in the focus-part of the sentence (e.g. MANN – ACT; TWHEN – PAT; PAT – EFF; ADDR – PAT). In other cases, it seems that a given pair of participants or circumstantials does not have any preferred word order (such as PAT / COMPL; PAT / DIR1, PAT / MEANS).

At the same time, all pairs report only a certain tendency (of varying degrees) to a canonical (systemic) ordering. However, for all pairs, it is also possible to find grammatical sentences in which their order will not correspond with the systemic ordering.

3.1 Order Actor / Patient

Due to the observed proportions of occurrences of pairs in the two possible sequences, a comparison can be made of systemic ordering of inner participants in the original scale. Interestingly, the original systemic ordering expected Actor in the first place followed by all other inner participants (even free modifications). However, the position PAT – ACT is slightly predominant in the data from the PDT. This finding is quite surprising because Czech is referred to as the language with the basic word order type SVO, which would correspond to the order ACT – PAT.

circumstantials with regard to each other. E.g. the sequence in the table 1 RESTR, MANN, THL only says that these participants or circumstantials appear often before than Actor in the sentence. It does not say that the usual mutual word order of these circumstantials is in the sequence RESTR, MANN and THL.

However, we should look at other possible word order factors (not taken into account yet) that may influence the word order position of Actor³ and Patient⁴ in the sentence.

3.1.1 Actor and Patient in the constructions with the verb *to be*

3.1.1.1 PAT.adjective – ACT.infinitive

The order PAT–ACT often occurs in structures with the copula verb *to be*, where the PAT frequently has the form an adjective and the ACT is in the form of verbal infinitive (like in English structures *it is necessary to expect, it is fair to assume, is good to compare, it is possible to deliver...*) – see (1) and (2). (It should be noted that with all of the examples below, the English translations are often only literal, presented here just to illustrate the intended meaning of the Czech sentence. At the same time we do not use just glosses and try to formulate grammatical sentences in English so that the order of the given participants or circumstantials in English translations do not correspond to their order in Czech; however, we believe that the reader can easily identify such cases by comparing the values of the respective functors.)

(1) *Je nutné.PAT_{focus} přiznat.ACT_{focus}, že nebýt regulace cen tepla, mnozí jeho výrobci by už jistě neexistovali.*

It is necessary.PAT_{focus} to admit.ACT_{focus} that without the regulation of heat prices, many of its producers probably would not already exist.

³ “ACT (Actor) is a functor used primarily for the first argument. In those cases when there is no argument shifting, the modification with the ACT functor refers to the human or non-human originator of the event, the bearer of the event or a quality/property, the experiencer or possessor.” (M. Mikulová et al., 2008)

⁴ “The PAT functor (Patient) is a functor used primarily for the second argument. In those cases when there is no argument shifting, the modification with the PAT functor refers to the affected object (in the broad sense of the word). [...] [However,] the Patient is defined primarily syntactically. [...] The PAT functor is also assigned to nodes representing the nominal part of a verbonominal predicate (e.g. *být hodný.PAT* (= to be good)).” (M. Mikulová et al., 2008)

(2) *Improvizace je dobrá věc, ale je potřebné.PAT_{focus} se zamyslet.ACT_{focus} nad možnými eventualitami a důsledky.*

The improvisation is a good thing, but it is needed.PAT_{focus} to consider.ACT_{focus} the possible eventualities and consequences.

In the PDT, 202 of these structures occur in the order: PAT.adjective – ACT.infinitive. It is interesting to notice that this pair does not occur there in the reverse order (ACT.infinitive – PAT.adjective), or, better to say, it is present (25 occurrences), but the ACT is always contextually bound in such structures (these constructions – see example 3 – are not included in the research). However, this does not mean that the sequence ACT.infinitive – PAT.adjective cannot appear in Czech with both the ACT and the PAT being contextually non-bound.

(3) *(Že úrokové sazby jsou vysoké, je zřejmé.) Proto splatit.ACT_{non-focus} úvěr za čtyři roky je pro většinu nových vlastníků nemožné.PAT_{focus}.*

(That the interest rates are high, it is obvious.) Therefore it is impossible.PAT_{focus} to pay back.ACT_{non-focus} the credit for most new owners in four years.

3.1.1.2 PAT.noun – ACT.noun /

ACT.noun – PAT.noun

In PDT, there is a total of 560 occurrences of the PAT and the ACT in the constructions with the verb *to be*. The vast majority of them is in order PAT – ACT (391 hits) and 169 occurrences in order ACT – PAT. If we leave the last-mentioned structures (PAT.adjective – ACT.infinitive), there are 189 matches in the order PAT – ACT (examples 4 and 5) and 169 occurrences in the order ACT – PAT (examples 6 and 7) so that their proportion is nearly balanced.

(4) *Pro mne je absolutním spisovatelem.PAT_{focus} Shakespeare.ACT_{focus}.*

For me, the absolute writer.PAT_{focus} is Shakespeare.ACT_{focus}

(5) *80procentním podílem je nejfrekventovanějším padělkem.PAT_{focus} stomarková bankovka.ACT_{focus}.*

With 80percent share, a one-hundred-mark bill.ACT_{focus} is the busiest fake.PAT_{focus}.

(6) *V blížících se komunálních volbách je starost.ACT_{focus} o štěstí budoucích generací líbivým politickým gestem.PAT_{focus}.*

In the upcoming municipal elections, the concern.ACT_{focus} for the happiness of future generations is a catchy political gesture.PAT_{focus}.

(7) *Na rozdíl od jiných armád byla služba.ACT_{focus} v bojových jednotkách ozbrojených sil pro Američanky dlouho tabu.PAT_{focus}.*

Unlike other armies, the service.ACT_{focus} in combat units of the armed forces was taboo.PAT_{focus} for American women for a long time.

It seems that in these cases (examples 4 through 7), it is mainly the speaker's communicative intention that decides the order of the ACT and the PAT. He or she puts the more important information more to the right in word order as it is typical for Czech. And since the order of the ACT and the PAT is probably not grammatically fixed in Czech in these cases (as demonstrated above), the speaker has a choice of two (probably grammatically equivalent) options. However, these options are not equivalent in terms of communication.

In the sentence 4 the speaker (or writer) expresses who is his or her absolute writer (he or she chooses one possibility out of the “menu” of writers – e.g. *Beckett, Goethe, Schiller, Shakespeare...*). While in the sentence 8 with a reversed word order, the speaker would testify the fact who is Shakespeare for him or her – if the intonation centre would be at the end of the sentence (he or she would choose from the “menu” of Shakespeare's characteristics – such as *a good man, an interesting person, an average actor...*) – cf. Sgall et al. (1980, p. 82ff). However, in example 8, *Shakespeare* must be probably context bound.

(8) *Pro mne je Shakespeare.ACT_{non-focus} absolutním spisovatelem.PAT_{focus}.*

For me, Shakespeare.ACT_{non-focus} is the absolute writer.PAT_{focus}.

It seems that in some cases, the position $ACT_{focus} / PAT_{focus}$ has only one possible sequence in word order – as in example 4. In this example, the only unmarked position is probably $PAT_{focus} - ACT_{focus}$. Another position would be marked – as in example 8: $ACT_{non-focus} - PAT_{focus}$. Therefore, the position $ACT_{focus} / PAT_{focus}$ depends probably on the concrete lexical expressions of ACT and PAT. This issue must be further examined in details in another research.

3.1.2 Actor and Patient depending on a verb other than the copula *to be*

It is interesting to examine also the constructions with the ACT and the PAT that depend on a verb other than the copula *to be*. Here the order ACT – PAT is more common, attesting the original scale of systemic ordering (317 occurrences of the order ACT – PAT; 224 occurrences of PAT – ACT).

Among them, it is possible to find two types of more frequently occurring structures. The first is the ACT expressed by a noun and the PAT expressed by a verb. The other type is the structure in which the ACT and the PAT are expressed by nouns.

3.1.2.1 PAT.verb – ACT.noun /

ACT.noun – PAT.verb

There are 51 constructions in the order PAT.verb – ACT.noun in the PDT (examples 10 and 11) and 20 constructions in the order ACT.noun – PAT.verb (examples 12 and 13). It seems that the position PAT.verb – ACT.noun is more typical.

(10) *Poměrně velká část poptávky odpadla, když k nám ze zahraničí začali jezdit.PAT_{focus} chudší turisté.ACT_{focus}.*

The relatively large proportion of demand fell down when poorer tourists.ACT_{focus} began to come.PAT_{focus} to us from abroad.

(11) *V cestovním ruchu se rozhodla podnikat.PAT_{focus} i řada.ACT_{focus} živnostníků.*

An array.ACT_{focus} of traders decided to do business PAT_{focus} in the tourism.

(12) *Stále více začínají podnikatelé.ACT_{focus} oceňovat.PAT_{focus}, když v počítači získají také svého daňového a právního poradce.*

Businessmen.ACT_{focus} begin to appreciate.PAT_{focus} more and more when they receive also their tax and legal advisors in computer.

(13) *Pro nadcházející období navrhuje ministr.ACT_{focus} financí přitvrdit.PAT_{focus} výdajovou politiku vlády.*

For the coming period, the Chancellor.ACT_{focus} of the Exchequer proposes to tighten up.PAT_{focus} the expenditure government policy.

In our opinion, the order of the ACT and the PAT is influenced, also here by the communicative point of view – the sentence element carrying the more important information (in the opinion of the speaker or writer) is more to the right. Here we can also observe a related tendency to such order in that the member with a more specific meaning (more meaningful new information) is more to the right (cf. *Mluvnice češtiny* 3, 1987, pp. 608ff).

In examples 12 and 13, the lexical meaning of the PAT is supplemented by the lexical meaning of other sentence elements depending on PAT (and at the same time, all these meanings give an additional piece of information). The semantic importance of the infinitive is thus significantly complemented: e.g. *to appreciate what, to tighten up what* – the elements depending on PAT are in the focus-part of the sentence.

By contrast, in examples 10 and 11, the PAT is informatively poorer. It rather has dependent elements, but they carry “old”, identifiable (i.e. contextually bound) information – the elements depending on PAT are in the topic-part of the sentence. The only “new” information here (except the predicate and the PAT) is carried by the ACT. And the ACT has also the most meaningful information of all the contextually non-bound members.

Probably because of the low “semantic weight” of the end element, the sentence 14 would be unnatural if the ACT and the PAT

were context non-bound. This sentence could be used probably only if all other elements except the last one were contextually bound.

(14) *Poměrně velká část poptávky odpadla, když k nám ze zahraničí začali chudší turisté.ACT_{non-focus} jezdit.PAT_{focus}.*

The relatively large proportion of demand fell when poorer tourists.ACT_{non-focus} began to come.PAT_{focus} to us from abroad.

The infinitive itself carries likely too “little meaning” (little information) in this case to be able to occur in the most communicatively important place of the sentence (if the ACT were context non-bound). However, if we complement it by other (“new”) semantic features, it could be at the end place without any problems (if we understand its “new” dependent complements as a whole with it) – see example 15.

(15) *Poměrně velká část poptávky odpadla, když k nám ze zahraničí začali chudší turisté.ACT_{focus} jezdit.PAT_{focus} za památkami UNESCO.*

The relatively large proportion of demand fell when poorer tourists.ACT_{focus} began to come.PAT_{focus} to us from abroad because of the UNESCO sights.

In most cases, found in the PDT with the order ACT.noun – PAT.verb, the PAT has still another (contextually non-bound) dependent sentence members. In contrast, in the structures PAT.verb – ACT.noun, the PAT has (if any) mostly contextually non-bound dependent members (i.e. known and therefore less informatively important) – see example 16 – or the PAT has also contextually non-bound dependent elements, but in the role of the ACT there is often a semantically richer (and usually a more specified) participant (examples 17 and 18).

(16) *Milionový poplatek.PAT_{non-focus} za vydání osvědčení, které umožňuje vést lékárnu, zakázalo.PRED_{focus} vybírat.PAT_{focus} Ministerstvo.ACT_{focus} pro hospodářskou soutěž.*

The Ministry.ACT_{focus} for Economic Competition banned.PRED_{focus} to collect.PAT_{focus} the million fee.PAT_{non-focus} for

issuing of a certificate which allows having a pharmacy.

(17) *Loupežným přepadením.MEANS_{non-focus} při němž jim byly odcizeny pasy, se v srpnu.TWHEN_{non-focus} snažili.PRED_{focus} hlídce.ADDR_{non-focus} oddělení pohraniční policie vysvětlit.PAT_{focus} ilegální vstup.PAT_{focus} do ČR dva Turci.ACT_{focus} (33, 31 let), kteří žijí a pracují v Německu.*

Two Turks.ACT_{focus} (33, 31 years) who live and work in Germany tried.PRED_{focus} to explain.PAT_{focus} the illegal entry.PAT_{focus} to the CR [Czech Republic] to the patrol.ADDR_{non-focus} of border police department by robbery.MEANS_{non-focus} in which their passports were stolen in August.TWHEN_{non-focus}.

(18) *Po souboji.TWHEN_{non-focus} s Jelínkem zůstal.PRED_{focus} za švédskou brankou.LOC_{non-focus} bezvládně.MANN_{focus} ležet.PAT_{focus} 27letý Mikael Lindman.ACT_{focus}.*

After the battle.TWHEN_{non-focus} with Jelinek, 27-year-old Mikael Lindman.ACT_{focus} remained.PRED_{focus} lying.PAT_{focus} limply.MANN_{focus} behind the Swedish goal.LOC_{non-focus}.

It is grammatically possible to put the ACT on the communicatively most important place despite the fact that the PAT and its dependent members carry many pieces of “new” (contextually non-bound) information (example 19), but these cases are quite rare in PDT. Such constructions sometimes probably better serve for the communicative plan of the speaker (however, we have to notice that here also the ACT is not informatively poor – it also carries a large amount of meaning).

(19) *Američan vytvořil světový rekord 47.02 v roce 1983 a jeho čas se podařilo překonat.PAT_{focus} až o devět let později.TWHEN_{focus} ve finále.LOC_{focus} závodu olympijských her v Barceloně jeho krajanovi Kevinu Youngovi.ACT_{focus} (46.78).*

An American set a world record of 47.02 in 1983 and his compatriot Kevin Young.ACT_{focus} (46.78) managed to overcame.PAT_{focus} his time nine years



later.TWHEN_{focus} in the final.LOC_{focus} of race in the Olympic Games in Barcelona.

On the other hand, if the PAT is semantically richer, it would take place after the ACT (example 20).

(20) *Během ní jí před hotelem stačili zloději.ACT_{focus} ukrást.PAT_{focus} auto.*

During it, the thieves.ACT_{focus} managed to steal.PAT_{focus} a car to in front of the hotel.

The reverse word order (example 21) of ACT and PAT would be unnatural, because the verb *to steal* includes in its semantics that the ACT are *thieves*.

(21) ? *Během ní jí před hotelem stačili ukrást.PAT_{focus} auto zloději.ACT_{focus}.*

During it, the thieves.ACT_{focus} managed to steal.PAT_{focus} a car to in front of the hotel.

However, if we add some “new” (unretrievable) information about the thieves, the word order PAT – ACT is possible (22) as well as the order ACT – PAT (in such case, probably the choice of the speaker, or, as the case may be, his/her communicative plan, would decide which word order will be used).

(22) *Během ní jí před hotelem stačili ukrást.PAT_{focus} auto zloději.ACT_{focus} v zelených bundách.*

During it, the thieves.ACT_{focus} in green jackets managed to steal.PAT_{focus} a car to in front of the hotel.

There are also some formal criteria that affect the word order. Š. Zikánová (2006, p. 43) mentions the well-known tendency of so-called heavy (i.e. long) members to occur rather at the end of the sentence (example 23). However, it is questionable whether the heavy members tend to be at the sentence end because of their form or because of the fact that “more words put together more information” and therefore they have better chance to be placed in the communicatively most important position.

(23) *Právě kvůli němu se rozhodli hráči.ACT_{focus} vstoupit.PAT_{focus} do stávky, v*

jejímž důsledku pak nenastoupili ke třem zápasům na turnaji Seliko Cup' 94 v Přerově a v Olomouci.

Precisely due to him, the players.ACT_{focus} decided to join.PAT_{focus} the strike; in consequence of this they did not attend three matches at the tournament Seliko Cup '94 in Přerov and in Olomouc.

It seems that in Czech the tendency to occupy a final position is mainly observed by members on which another clause depends, but again, it is not a rule (example 24).

(24) *Velkou akci začali tři sokolovští „podnikatelé“.ACT_{focus} z nichž jednoho už v té době stíhala plzeňská policie pro podvod, plánovat.PAT_{focus} v prosinci minulého roku.*

Three “bussinesmen”.ACT_{focus} from Sokolov – one of them had been hunted for fraud by police in Pilsen at that time – started planning.PAT_{focus} the big event in December last year.

Obviously the preference of the end-position in these cases depends also on the fact how long the member is. If the heavy member is not at the end, it should not be “too long”. The listener or reader would have to keep in memory the valency frame of the predicate for a long time and it would make the understanding difficult. If the heavy member is at the end, the listener or reader knows (at least syntactically) all other members of the valency frame before he/she begins to perceive the longest (and most complicated) one.

A similar feature of word order (to put the heavy member to the end) can be found also in German. In German (in contrast with Czech) there is a strong grammatical tendency to put the infinitive at the very end position. However, e.g. if a member of the sentence is further modified by a dependent relative clause, this clause can follow the infinitive (example 25).

(25) *Ich wollte auch Drumsticks haben, die nicht so schnell kaputt gingen.*

I wanted to have also drum sticks that were not easily broken.

The syntactic structures in which the semantically obligatory member is separated

from the verb on which it depends by too many other members may be a source of language comics (example 26 – from Czech comic drama *Posel z Liptákova*).

(26) Při průjezdu Mladou Boleslaví **dostal.PRED** můj spolujezdec kolega Hraběta právě v místech, kde byl na prahu románského kostelíka zavražděn svým bratrem Boleslavem roku 929 nebo 935, o tom jsou doposud spory, kníže Václav **žízeň.PAT**.

While driving through Mladá Boleslav, my fellow passenger colleague Hraběta became.PRED thirsty.PAT right in places where the Prince Wenceslas was murdered on the verge of a Romanesque church by his brother Boleslav in 929 or 935, there are still disputes.

3.1.2.1 ACT.noun – PAT.noun /

PAT.noun – ACT.noun

If both members (ACT and PAT) are expressed by a noun, the word order ACT.noun – PAT.noun is more common (examples 27 and 28): in PDT there were 251 occurrences of such structures (the probability of this sequence in PDT is 0.66). It corresponds with the original scale of systemic ordering.

(27) V prodejně Arxonu najdou **zákazníci.ACT**_{focus} mnozí již stálí, také různé **příručky.PAT**_{focus} pro podnikatele a ekonomy.

*The customers.ACT*_{focus} many already regular, find also the various **guides.PAT**_{focus} for entrepreneurs and economists in the shop Arxon.

(28) Společně se třemi zahraničními deníky vydávají Lidové **noviny.ACT**_{focus} Středoevropské **noviny.PAT**_{focus}.

*Together with three foreign dailies, the People's Newspaper.ACT*_{focus} publishes the Central European Newspaper.PAT_{focus}.

The order PAT.noun – ACT.noun has 131 occurrences in PDT (examples 29, 30).

(29) Na dvojnásobné trati žen vynikajícím závěrečným finišem přesprintovala favorizovanou Jihoafričanku Elanu

*Meyerovou.PAT*_{focus} časem 31.56,97 Yvonne Murrayová.**ACT**_{focus} ze Skotska.

*On the women's double track, Yvonne Murray.ACT*_{focus} of Scotland overtook favored South African Elana Meyer.**PAT**_{focus} by excellent finish with the time 31.56,97.

(30) Ke konci minulého školního roku rozevázalo pracovní **poměr.PAT**_{focus} na 250 **pedagogů.ACT**_{focus}.

*At the end of the last school year, 250 teachers.ACT*_{focus} terminated their **employment.PAT**_{focus}.

Which word order will be chosen by the speaker, is probably determined also by already mentioned reasons – the communicative plan of the speaker, the “fullness of ‘new’ meaning” of both participants and their length. However, there are certainly other reasons also at play – such as idioms (cf. Zikánová, 2006, p. 43) as demonstrated in example 31 (the *rozevázat pracovní poměr* ‘terminate employment’ is a fixed legal multiword expression in Czech) or the grammatical form of the participants (example 29 with the homonymous form *noviny*_{nominative pl.} – *noviny*_{accusative pl.}). They will be observed in further research.

4 Conclusion

The aim of the paper was to put under scrutiny the scale of the original systemic ordering for inner participants Actor and Patient. Our analysis of their sequence if they are the contextually non-bound (i.e. in the focus-part of the sentence) demonstrates that it is quite problematic to establish a single scale. Further research will therefore concentrate on looking for criteria and reasons that may influence a canonical Czech word order.

Acknowledgment

This paper was supported by the grant GA ČR 405/09/0729 “From the structure of a sentence to textual relationships”.

References

Daneš, František; Hlavsa, Zdeněk; Grepl, Miroslav et al. 1987. *Mluvnice češtiny (3). Skladba*. Academia, Prague.

- Mikulová, Marie et al. 2008. *Annotation on the tectogrammatical level in the Prague dependency treebank: annotation manual*. Universitas Carolina Pragensis, Prague. ISBN 978-80-254-3088-0. WWW: <<http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/t-layer/html/ch07.html>>.
- Mikulová, Marie et al. 2005. *Anotace na tektogramatické rovině Pražského závislostního korpusu: anotátorská příručka*. Universitas Carolina Pragensis, Prague. ISBN 80-254-3087-1. WWW: <<http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/cz/t-layer/html/>>.
- Sgall, Petr; Hajičová, Eva; Buráňová, Eva. 1980. *Aktuální členění věty v češtině*. Academia, Prague.
- Zikánová, Šárka. 2006. What do the data in Prague Dependency Treebank say about systemic ordering in Czech? *The Prague Bulletin of Mathematical Linguistics* 86, pp. 39–46. ISSN 0032-6585.
- Cimrman, Jára da; Smoljak, Ladislav; Svěrák, Zdeněk. 2002. *Posel z Liptákova*. Paseka, Prague. ISBN 80-7185-479-4.
- Prague Dependency Treebank*. Version 2.0. Charles University in Prague, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics. WWW: <<http://ufal.mff.cuni.cz/pdt2.0/>>. [29. 4. 2011]

Representation of Zero and Dummy Subject Pronouns within multi-strata dependency framework

Dina El Kassas

Faculty of languages and Translation
Minya University, Minya
Modyco Laboratory
Nanterre University, Paris
delkassas@gmail.com

Abstract

The objective of this paper is to discuss a formal representation of subject pronoun within a multi-strata dependency model. We propose criteria to describe consistently subject pronoun variations, naming subject pronouns that have no meaning and/or no morpho-phonological expression. We will present particular syntactic structures raised from a change of voice category; and will emphasize the problematic representation of Pro-Drop impersonal construction within the multi-strata framework.

1 Introduction

The present study aims to describe the typologically widespread pronoun dropping and the expletive pronoun subject phenomena. The representation is based on the core of the nature of linguistic sign as well as the main communicative function of the pronoun as a grammatical part of speech.

The term Pro-Drop describes a feature of some languages that does not require an obligatory overt actant to be present in the clause. Languages allowing Pro-Drop fall into three categories (Dryer, 2008): those allowing Pro-Drop only in particular context; those allowing Pro-Drop only in subject position; and those allowing both subject and direct object Pro-Drop.

The dropped subject pronoun is commonly identified by Universal Grammar as a null subject and is defined as a linguistic sign that has a meaning but doesn't have a phonetic realization. The result is an independent clause lacking an explicit subject. The verb agreement expresses person, number and/or gender with the referent. We will call it following Mean-

ing-Text Theory (MTT) terminology the zero pronoun. An MTT zero pronoun is exactly a linguistic sign that has a meaning of 'people' or 'element'.

Studies on expletive subject pronoun representation have focused on its semantic emptiness and its non-referential (non-endophoric) status. The construction including an expletive subject pronoun governed by finite verbal clause is commonly identified as impersonal construction. Again, following the terminology used in MTT, we will call it the dummy pronoun.

We propose a formal description of zero and dummy pronouns within the framework of MTT that offers a rigorous exhaustive coverage of linguistic sign and makes explicit its intersection with voice (Mel'čuk, 2006). As in many other dependency frameworks (XDG, FDG ...), MTT model posits multiple strata of representations related by explicit interfaces. The study refers primarily to examples from the Arabic.

The paper is structured as follows:

Section 2 presents: the linguistic sign as described within the MTT framework; a typology covering sentences featuring zero and dummy subjects; and a formal treatment of these constructions within the Meaning-Text dependency syntax framework.

In Section 3, we discuss the grammemes of the Arabic voices. The objective is to shed light on some issues concerning zero and dummy construction representations provoked by the deep-syntactic level.

Section 4 is dedicated to the conclusion and future work.

We take for granted the basic notions of the Meaning-Text dependency syntax (Mel'čuk, 1988), such that the representations are multi-stratal with intermediate interfaces.

The sentence structure at the semantic level is a dependency network. Each node is labeled by a language-specific semantic unit which corresponds to one particular word-sense. The oriented arcs connect the head predicate to its arguments, or semantic actants. Arc labels are consecutively numbered. These numbers distinguish semantic relations of the argument-to-predicate type. Roughly, the numbering follows the needs of semantic decomposition.

The sentence structures at deep and surface syntactic levels are dependency trees with lexemes being represented as nodes and syntactic relations as arcs. At the deep-syntactic level, the syntactic relations presenting actant relations are numbered by I, II, III, etc. and are assumed to be universal. At the surface-syntactic level, the encoded syntactic relations are language-specific functions (e.g. subject, direct object, oblique-object etc.).

2 Linguistic sign in MTT

According to the traditional Saussurean definition, a linguistic sign combines a signifier (a sound image, i.e. *signifiant*) and a signified (a concept, i.e. *signifié*). So, if *x* is a sign, *x* should be a combination of a phonetic realization and a meaning.

To these two components of the linguistic sign entity, a set of properties is added to give necessary syntactic information that specifies the correct combination of the given sign with other.

In MTT, the lexeme assuming the surface-syntactic subject function should be linked to nodes in both deep morpho-phonological and deep-syntactic levels and must have its own syntactics.

When the subjectal role is assumed by a pronoun, it should normally have an endophoric function, i.e. it should refer to another lexeme in the text. We have thus a first distinction: “**endophoric ~ non endophoric [subject pronoun]**” (or a *personal ~ impersonal pronoun*). Additionally, the subject pronoun may or may not have a morpho-phonological realization. Here comes the second distinction: “**overt ~ zero [subject pronoun]**”.

By subject pronoun, we refer only to the third personal pronouns such as English HE, SHE or THEY that assume a referential function but don't have a meaning in opposition with pronouns such as English I, WE or YOU that do have a meaning.

According to these two distinctions, we have four possible combinations in case of subject pronoun:

1) Subject pronoun having a phonetic realization and filling an endophoric function → [full pronoun]

It is off-topic to discuss here full pronoun. At any rate, subjects of type (1) are not relevant for our topic. The pronominalized and communicatively salient subject appears on the surface in Anti-Pro-Drop structures. The indefinite pronouns ON (French) and MAN (German) linked to the semantic collective/generic actant are considered as subject full pronouns.

2) Subject pronoun having no phonetic realization but filling an endophoric function → [zero pronoun]

By zero pronoun, we mean a pronoun that is morpho-phonetically empty. We are aware that the term in MTT terminology refers to zero meaning and not zero physical expression. Yet, we use it for lack of a better term. The subject pronoun appears in the SSyntS as a meaningful zero or empty lexeme and controls the agreement of the verb. Arabic has a wide range of sentences lacking an overt sentence element. For example, the copula KĀNA ‘(to) be’ has a zero present indicative form and governs sentences traditionally called nominal sentences:

(1) Ø_{kāna} 'alqalaqu mubarrarun
V.is N.concern ADJ.justified
‘Concern is justified’

vs. *kāna* 'alqalaqu mubarraran
V.was N.concern ADJ.justified
‘Concern was justified’

Zero radicals are also frequent in Slavic, Romanian and Semitic languages. The zero sign lacks the signifier. The trace of the presence of a zero subject pronoun in the sentence is the feature of its syntactics that is copied on the verb via a rich agreement and is communicatively salient:

(2) Rus *Stučat v dver'* ‘[they] knock at door’
It Fumo ‘[he] smokes’
Sp *Llaman a la puerta* ‘[Someone] is knocking the door’
Ar '*akalū*_{v.3.pl.masc} ‘[they] ate’
Hebrew *axalti*_{v.active.past.1.sg} *tapuax* ‘[I] ate an apple’

In Arabic, the subject pronoun is not realized phonetically and the verb exhibits a full PNG agreement. The Arabic inflected verb agrees with its subject pronoun in person (1, 2, 3), number (singular, dual, plural) and gender

(masculine, feminine). This rich verb agreement allows the suppression, or more precisely the non-realization of the pronominal inaccentuated subject, avoiding thus a grammatical redundancy without giving rise to any ambiguity:

- (3) *'akalū*_{V.eat.active.past.3.masc.pl}
*'akalna*_{V.eat.active.past.3.fem.pl}
*'ukilū*_{V.eat.passive.past.3.masc.pl}

The meaningful subject pronoun with zero form may be compatible with a specific individual who satisfies the description, giving so an existential reading, but it may also imply a generic universal reading. In both cases, the morpho-phonetically zero-subject pronoun denotes an endophoric relation with a full lexeme in the sentence or the text. This pronoun must be distinguished from the dummy-subject one commonly described as an impersonal construction (cf. figure 1).

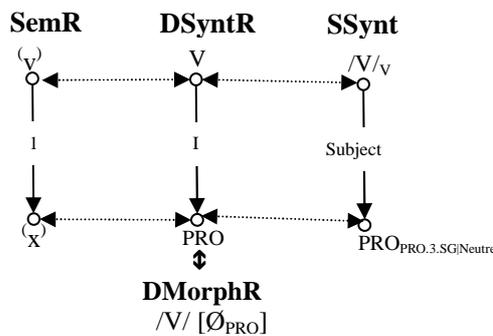


Figure 1: zero-form meaningful subject pronoun

Zero subject sign has to be carefully distinguished from deleted subject. The syntactic operation of deletion or ellipsis consists in removing a sign from a representation, like when we answer a question; while zero sign lacks the overt signifier but is not omitted. There are different types of ellipsis but we are not interested here in the distinction between discourse ellipsis and Pro-Drop phenomenon. Let's just mention that an elided subject can be reconstituted in context, but a zero-form subject pronoun cannot. In the following example, the subject *hathā* is elided.

- (4) - *hal* *hathā* *mumkinun?*
interro. this possible
'Is this possible?'
- *na'am* *hathā* *mumkin dziddan*
yes possible very
'Yes, it is very possible'.

Meteorological expressions present also a case of meaningful subject dropped pronoun. In sentences (5a) and (5b), the verb agreement

presents a case of zero-form semantically full pronoun: the verbs are at the active.present.3.fem.sg inflectional forms and indirectly govern the pronoun *hā*_{3.fem.sg}, referring to *alsamā* 'the sky' which is a feminine noun in Arabic.

- (5) a. *'inna=hā* *tumtir* [Ø_{PRO}]
Particule=PRO_{3.fem.sg} V_{act.pr.3.fem.sg} [she]
assert=she rains
'[she] rains'
b. *tar^cadu* [Ø_{PRO}] | *tabruqu* [Ø_{PRO}]
V_{act.pr.3.fem.sg} [she] | V_{act.pr.3.fem.sg} [she]
thunders | lightens
'[she] thunders' | '[she] lightens'

It is also accurate to assign to meteorological verbs the noun *alsamā* as an explicit SSynt subject, thus the following sentences are correct: '*tumtir alsamā'u*', '*tar^cadu alsamā'u*', etc. This assignation of meteorological verbs to the appropriate nature force is frequent in Arabic:

- (6) a. *tahubbu* *alrijāhu*
V_{act.pr.3.fem.sg} N_{fem.NOM}
blows the winds
'It blows'
b. *jabzugu* *alfadzru*
V_{act.pr.3.masc.sg} N_{fem.NOM}
emerged the dawn
'It dawns'

The corresponding equivalent in Anti-Pro-Drop language like English is generally an impersonal construction with a semantically empty explicit subject pronoun.

3) Subject pronoun having phonetic realization but not filling an endophoric function → [dummy pronoun]

The subject is semantically empty and thus presents a **dummy sign** which is defined as a sign lacking the signified. The dummy subject occurs in impersonal constructions. Indeed, an impersonal construction is defined by the presence of an automatically generated subject pronoun that does not correspond to a deep-syntactic / semantic actant, which means that the pronominal subject is not assuming an endophoric function in the discourse. The term 'impersonal construction' is quite felicitous but it is so entrenched in the linguistic literature that it is impossible to spare. However, we find it more accurate to talk about a **semantically empty non-endophoric subject pronoun** and so, only 3rd singular pronoun may be the subject of an impersonal construction, 1st and 2^{sd} pronouns cannot be the subject of an impersonal construction as they have semantic referents. We have examples of dummy sign in

Anti-Pro-Drop languages: IT (English), IL (French), etc.

- (7) a. Fr. *Il tonne* = ‘It thunders’.
 b. Fr. *Il est arrivé 3 personnes* = ‘It comes 3 persons’.
 c. Fr. *Il a été détruit 3 camions* = ‘It was destroyed three trucks’.

In principle, the dummy construction can be used with all types of verbs (transitive, intransitive, pronominal) and combines with voice grammemes in the language. Figures (2) and (3) present semantic and syntactic patterns of impersonal constructions:

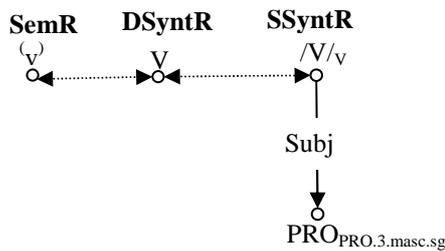


Figure 2: Impersonal construction (7a)

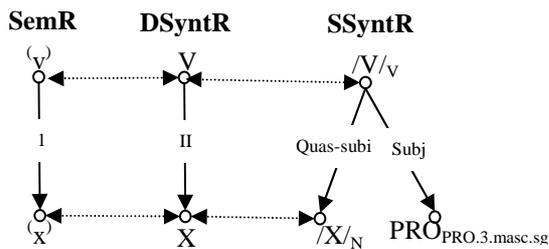


Figure 3: Impersonal construction (7b) and (7c)

4) **Subject pronoun having no phonetic realization and not filling an endophoric function** → [dummy zero pronoun]

The fourth case presents subjectless sentences including those lacking subjects even in the SSyntS. The pronoun represents a sign lacking both the signified and the signifier:

- (8) It. *Piove* ‘Rains’ = ‘It rains’
 Serb. *Grmi* ‘thunders’ = ‘It thunders’

Arabic has a particular zero-subject pronoun featuring an impersonal structure, as in the examples (9a) and (9b) featuring a subjective suppressive voice; the verbs are systematically in the SubjSupp.3.masc.sg inflectional form:

- (9) a. *murra* *bi=hindin*
 $V_{\text{SubjSupp.3.masc.sg}}$ PREP=N_{GEN}
 passed by=Hind
 ‘[Someone] passed by=Hind’

- b. *nīma* *fī* *aldāri*
 $V_{\text{SubjSupp.3.masc.sg}}$ PREP N_{GEN}
 slept in the.house
 ‘[Someone] slept in the house’

We will discuss thoroughly the SubjSupp grammeme in section 3. Let's say here that, on the one hand, the subject pronoun has no physical expression and thus presents a zero pronoun. On the other hand, it will not be accurate to describe it as a dummy zero pronoun because it is not semantically empty: even if the zero-subject pronouns in examples (9a) and (9b) are not linked to specific entities, the sentences still have an existential reading: ‘one or few persons passed by Hind’, ‘one or few persons slept in the house’. The semantic actant in both cases must be a human agent: the subject pronoun of a verb in the subjective suppressive voice could not correspond to a non-human agent. Thus, the sentences *marra alkilābu bi=hindin* ‘the dogs passed by Hind’ could not be transformed to the subjective suppressive. We would rather refer to this structure as an INDEFINITE PERSONAL like in the Russian tradition, or the pronoun ‘ON’ in French.

As we see, the study of zero and dummy subject pronouns is intrinsically related to voice grammemes that it's why the next section will be dedicated to the formal representation of voice grammemes in Arabic.

3 Formal Representation of Voice Category in MTT

In the MTT framework, a diathesis of a word-form is defined as the correspondence between its Semantic and Deep-Syntactic Actants ($\text{SemA} \Leftrightarrow \text{DSyntA}$)¹; voice is a *semantic* inflectional category whose grammemes specify modifications of the basic diathesis of a lexical unit L without affecting the propositional meaning of L². The basic diathesis of L is its lexicographic diathesis. Voices help to construct different messages about the same situation.

As we said above, the semantic actant corresponds to the argument of the predicate and is identified by a distinctive asemantic number. A DSyntA of a lexical unit L is another lexical unit depending syntactically on L and corresponding to a SemA of L or to a Surface-Syntactic Actant [SSyntA] of L³. An important

¹ Mel'čuk, 2006, p. 187.

² *Idem*, p. 191.

³ *Idem*, p. 184.

feature of DSyntAs is their intermediate character as an interface between the SemAs and the SSyntAs of a lexical unit: they are determined either semantically or surface-syntactically. The dummy subject, whether with expletive or no physical expression, does not appear in the DSyntS.

DSyntA are identified by meaningful Roman numbers and ordered following the decreasing obliqueness. Each number corresponds to a family of surface-syntactic constructions brought together because of their similarity. Thus, DSyntA I stands for the syntactic constructions that express the subjectal SSynt-relation; DSyntA II represents, among others, the DirO, the Indirect or Oblique object, and the Agentive complement with the passive form of a transitive verb; DSyntA from III to VI represent more oblique objects/complements.

The diathesis of L can be modified by one of the following three operations: PERMUTATION of the DSyntAs of L with respect to the corresponding SemAs, SUPPRESSION of DSyntAs of L (which means that the SemA involved cannot be manifested syntactically as a direct syntactic dependent of L. This means blocking the slot of this DSyntA), REFERENTIAL IDENTIFICATION of two SemAs, with obligatory suppression of at least one DSyntA.

There are four possible main voices: active, passive, suppressive and reflexive. The passive voice consists of assigning another DSynt-role (II or III) to the expression that fills the DSynt-role I in the active voice. There are five possible passive voices:

- Full, if both DSyntAs of L are affected (I \Rightarrow II and simultaneously II \Rightarrow I).
- Partial, if only one of the DSyntAs of L is affected (I \Rightarrow III, while II remains in place and nothing becomes I).
- Promotional, if the passive promotes the DSyntA II to I and demotes the DSyntA I.
- Demotional, if the passive demotes the DSyntA, without promoting anything.
- Agentless, if the passive does not allow for an Agentive Complement.

According to this formalism, Arabic has the following six voice grammemes:

1) Active voice [Act.]

It is the basic diathesis: the DSyntA I corresponding to the primary semantic argument is linked to the SSynt subject [$x'_1, X_{I,}/X_{/subj}$]:

X	Y
I	II
Subject	Object

The Zero-Subject pronoun in the active voice has a full meaning, a syntactic presence but no physical expression [$x'_1, X_{I,}/\emptyset_{subj-pro}$]. The verb may be transitive (10a) or intransitive (10b), regardless of the verb tense:

- (10) a. *qālat* $\emptyset_{pro.3.fem.sg}$ *alḥaqa*
 $V_{act.past.3.fem.sg}$ $N_{def.ACC}$
 said [she] the truth
 ‘[She] said the truth’
- b. *janaamuun* $\emptyset_{pro.3.masc.pl}$
 $V_{act.pres.3.masc.pl}$
 are.sleeping [they]
 ‘[They] are sleeping’

It will not be accurate to consider the above sentences as elliptical constructions. The omission is not intentional but grammatically required: the pronoun filling the subject function does not have a morpho-phonetic expression. We prefer distinguishing between the grammatical obligatory omission and the discursive or stylistic omission even if the latter one is occasionally obligatory also.

We should also differentiate the zero subject pronoun from the completive clause that may fill the subject function as in the following example:

- (11) *balagha=nī* [*anna=ka satarḥalu*]_{subjClause}
 $V_{act.pr.3.masc.sg}=\text{PRO}_{1.pl}$ [CONJ=PRO V]_{subj}
 was.informed=me [that=you will.leave]
 ‘I was informed that you will leave’.

A Demotional Active voice [DemAct.]

The detransitivization of some verbs may feature an impersonal construction: the subject is a dummy zero subject pronoun ($\emptyset^{\emptyset}_{3.MASC.SG}$), the SemA 1 [X] is demoted to DSyntA II rank, the SemA 2 is omitted and the DSyntA III [Y] keeps its rank:

X	Y
II	III
ObIO	ObIO

Let's take the example of the trivalent verb KAFAA meaning ‘suffice’ (X suffices as Y for Z):

- (12) *takfī=nā*_{DirO} *alsūrātu*_{subj} *šāhidan*_{co-predicate}
 $V=\text{PRO}$ $N_{def.NOM}$ $N_{indef.ACC}$
 suffice=us the.picture a.witness
 ‘The picture suffices for us as a witness’

The verb is in the demotional active present 3 feminine singular form as in agreement with the singular feminine noun *alsūratu* filling the subject function. Its actants are distributed as follow:

X	Y	Z
I	II	III
Subj	DirO	CoPred
<i>alsūratu</i>	<i>nā</i>	<i>šāhidan</i>

The co-predicate may be omitted without affecting the sentence's grammaticality: '*takfī=nā alsūratu*'. The direct object may also be omitted: '*takfī alsūratu*', meaning 'the picture suffices' or 'the picture is enough'. The DSyntA III could be realized as an oblique-object: '*takfī alsūratu ka_{PREP}=šāhidin_{GEN}*'.

The verb may also have a particular government pattern with a demoted DSyntA I as in the following sentence:

- (13) *kafā* [bi=*alsūrati*] *šāhidan*
 V_{DemAct.past.3.masc.sg} [prep=N_{fem.sg.GEN}] N_{ACC}
 is enough [of the picture] witness
 'The picture suffices as a witness'.

The sentence literally means 'It_{subject} makes_sufficient witness_{CoPred} with the picture_{OblO}'. The verb is in the demotional active past 3.masculine.singular form. It will not be inaccurate to use the verb in the present form, yet we don't notice a frequent use of it: '*jakfī bi=alsūrati šāhidan*'. The valency of *kafā_{act.pr.past.3.masc.sg}* is (Ø_{subj}, OblO, CoPred).

We can't follow the Arabic traditional grammar and analyze the prepositional phrase [bi=*alsūrati*] as a subject. We have here a demotional transformation of the DSyntA I from SSynt Subject rank to SSynt Oblique Object rank, the result is an impersonal construction with a subject pronoun featuring no meaning and no morpho-phonetic realization. The verb is systematically in the DemAct.3.MASC.SG inflectional form.

Some verbs govern by default this exceptional construction. In the following sentences the verb is systematically in the DemAct.3.MASC.SG whether the verb is in the past (14a) or the present (14b) form even if the lexemes expressing the SemA I are feminine nouns. These examples express the exclusion: the verb preceded by a negative particle governs an exclusive construction composed of the exclusive particle *'illa* followed by a noun re-

ferring to the SemA I of the verb in the affirmative form.

- (14) a. *mā fāza* 'illa 'anti
 Pa V_{DemAct.past.3.masc.sg} Pa PRO_{2.fem.sg}
 neg won except you
 'Only you have won'
 b. *lā jadhulu almawki^ca* 'illa 'alfatajātu
 Pa V_{DemAct.pr.3.masc.sg} Pa N_{3.fem.pl}
 neg enters the site except the girls
 'Only girls may enter the site'

The pronoun [*'illa 'anti_{2.fem.sg}*] or the noun [*'illa 'alfatajātu_{3.fem.pl}*] could not be considered as the subject of the head verbs for several reasons:

First, the verbs do not agree in gender with these elements.

Second, the verbs are in the negative form or these lexical elements correspond to the SemA I of the verbs in the affirmative form, as it shows the translation.

Third, as we said above, the subject pronoun has no physical expression in Arabic and so the pronoun *'anti* in (14a) cannot fulfill the subject function. This pronoun will disappear for example in the affirmative non-exclusive construction: *fuzti_{Act.past.2.fem.sg}* 'you won'. By analogy, the noun *'alfatajātu* in (14b) is not the subject. The sentences may be literally translated by: 'It won not except you' and 'It enters not the site except the girls'.

For these reasons, in my opinion, it will be pertinent to distinguish between an active and a demotional active voice.

2) Full Promotional Passive voice

In Arabic, as in Semitic languages, the passive voice is originally used only when the agent could not be expressed because it is unknown or the speaker does not want to name it. Therefore, the general rule is that the verb in the passive voice does not govern an agentive complement corresponding to the SemA I. However, even if the full passive voice is not frequent in Arabic, there are a number of prepositions and complex prepositions that are believed to be the equivalent of English agentive *by*. The SemA I is demoted to the DSyntA II rank, and conversely, DSyntA II is promoted:

X	Y
II	I
AgCo	Subject

The most common prepositions and complex prepositions introducing an agentive complement (AgCo) are: /bi/, /li/ /min/ /^cabr/ ‘by’; /bi-sababi/ ‘because of’; /min dzānibi/ ‘from the side of’; /^calā jadi/ajdi/ ‘at the hand/s of’; /min qibali/ /min khilāli/ ‘on the part of’; /biwāsīta/; /^can tarīqi/ ‘by means of’. The agentive complement may denote a human agent (15a) and (15b) or an entity expressing the way or the instrument (15c) and (15d):

- (15) a. *kutibat alriwāyatu biwāsītati zaidin*
was.written the.novel **by** Zayd
‘The novel was written by Zayd’.
- b. *futihāt alrisālatu ^can tarīqi almustaqbili*
was.opened the.message **by** the.receiver
‘The message was opened by the receiver’.
- c. *futiha albarīdu ^can tarīqi mawqī^ci*
was.opened the.mail **by** my.site
‘The mail was opened by my site’.
- d. *muni^ca albaladu ^cabra al’istiftā’i*
was.prevented the.country **by**
the.referendum
‘The country was prevented by the referendum’.

The full passive transformation is strongly due to the impact of translation from European languages in contemporary practice, particularly in journalism and IT fields. In the active voice, the agentive complement is promoted to the subject rank. Examples (16) present the active transformation of the sentences in (15): the agent regains its position as a subject in the nominative form followed by the direct object in the accusative form (ended by the vowel /a/).

- (16) a. *kataba zaidun_{subj} alriwāyata*
b. *fataha almustaqbilu_{subj} alrisālata*
c. *fataha mawqī^ci_{subj} albarīda*
d. *mana^ca al’istiftā’u_{subj} albalada*

3) Agentless passive voice [AgPass]

It is the most frequent passive in Arabic. The passivization of a bivalent verb consists of the suppression of the DSyntA I corresponding to the subject in the basis diathesis and the promotion of the DSyntA II. The agentless passive voice is intrinsically related to the de-transitivization process. In the remainder of the sub-section, we will present three specific cases: first the passivization of verbs governing ‘an/anna’ ‘that’-construction, the decreasing of the valence of bivalent verbs (intransitivization), then the decreasing of the valence of trivalent verbs governing a clausal object.

1) Verbs governing ‘an/anna’-constructions

The government pattern of some verbs categories, mainly verbs of speech includes three actants: a subject, an ‘an/anna’-construction as a direct object, and according to the verb, an indirect object. With the agentless passivization process, the direct object completive clause is promoted to the subject rank:

X	Y
–	I
–	Subject _{an/anna-construction}

No changes occur in the clause and the verb is systematically in the 3.masc.sg inflectional form (17a). We notice that some verbs are more frequently used in the passive form rather than the active one (17b). The most common equivalent in this case is the impersonal construction {IT + to be + ADJ}. Yet, the Arabic construction is not an impersonal one: the head verb governs systematically a completive clause as subject.

- (17) a. *luḥitha* [^canna...]_{subj-Clause}
V_{AgPass.past.3.masc.sg} [CONJ...]_{subj-Clause}
noticed [that...]
‘It was noticed that...’.
- b. *justaḥsanu* [^can taḥdira]_{subj-Clause}
V_{AgPass.past.3.masc.sg} [CONJ...]_{subj-Clause}
is.better [that you.come]
‘It would be better if you come’.

2) Intransitivization

When a bivalent verb undergoes agentless passivization, the direct object is promoted to the subject rank and the Sema 1 is not expressed:

X	Y
–	I
–	Subject

The lexeme filling subject SSynt Relation has generally a vague or a general meaning like [en. MATTER], [fr. AFFAIRE] and [ar. ‘AMR] (18 a-b). We note that the verb agrees in gender with the lexical subject.

- (18) a. *quḍija* [^cal’amru]_{subj}
V_{AgPass.past.3.masc.sg} N_{def.masc.sg.NOM}
was.settled the.matter
‘The matter was settled’
- b. *nuthirat* [^cal=mas’alatu]_{subj}
V_{AgPass.past.3.fem.sg} N_{def.fem.sg.NOM}
was.reviewed the issue
‘The issue was reviewed’

The SSynt subject role may be filled by a **non dummy zero morpho-phonological subject pronoun**: a pronoun, as we said above, having a full meaning, a syntactic presence but no physical expression. In the following examples, the non dummy zero subject pronoun is the 3rd masculine plural personal pronoun in (19a), the 1st singular personal pronoun in (19b) and the 2nd feminine singular pronoun in (19c). The subject identification was allowed by the verb agreement. So, even if the subject pronoun is deprived of a physical expression, it has a full meaning and a syntactic presence.

- (19) a. *qutilū* Ø_{subj-pro} *djamīʿan*
 V_{AgPass.pr.3.masc.pl} N
 have been killed all
 ‘They all have been killed’.
- b. *ʾukaddabu* Ø_{subj-pro} *dāʾiman*
 V_{Ag.Pass.pr.1.sg} ADV
 am accused of lying always
 ‘I am always accused of lying’.
- c. *bulligti* Ø_{subj-pr} *bi=nadjāhu=ki*
 V_{AgPass.pr.2.fem.sg} PREP=masdar=PRO
 was informed by=success=yours
 ‘You was informed of you success’.

3) Decreasing the valence of trivalent verbs

The passivization of trivalent verbs governing a clausal object, e.g. *ʾarā* (X_I, Y_{II}, Z_{III}) ‘X show to Y that Z’, consists of the suppression of DSyntA I, the promotion of the DSyntA III to DSyntA I rank while DSyntA II keeps its rank:

X	Y	Z
–	II	I
–	Obj.Clause	Subject

In the example (20), the verb is in the 2nd singular form. The SSynt subject role is filled by a **non dummy zero morpho-phonological subject pronoun**. There is a particular communicational issue with the verb *ʾarā* in this context: the sentence literally means ‘You are shown what happened’, however its accurate English equivalent is ‘I wonder what happened’. This discernible communicational change is due to the agentless passivization transformation. In the Arabic sentence, even if the subject is a non dummy pronoun, it is not individuated. The subject pronoun does not also support a general reading. The example (20) presents so a syntactic constraint structure. It closest equivalent in English is the sentence ‘I wonder...’

- (20) *turā* Ø_{pro} [*māzā hadath*]_{Obj_clause}
 V_{AgPass.pr.2.sg} [interro. V_{act.pr.3.masc.sg}]
 is.shown [what happened]
 ‘You are shown what happened’.

4) Partial Agentless passive voice [PaAg-Pass]

The partial agentless passivization process concerns verbs governing a completive clause or a free direct/indirect speech. It denotes a detransivization process: the DSyntA I is omitted, the DSyntA II corresponding to the completive clause and the DSyntA III, in case of trivalent verb, are respectively promoted.

X	Y	Z
–	I	II
–	Subj.Clause	ObjCo

The Examples (21) present the passivization of trivalent verb *qāla* (X_I, Y_{II}, Z_{III}) ‘X say to Y that Z’ (21a) and bivalent *junṭazaru* (X_I, Z_{II}) ‘X expecting that Y’ (21b). The verb agrees with the subject clause and is systematically in the 3rd masculine singular inflectional form. The sentences do not present so an impersonal construction even if the English equivalent is.

- (21) a. *qāla* [*la=hu*]_{ObjCo} [*ʾirḥal*]_{subj}
 V_{PaAgPass.3.masc.sg} [PREP=PRO] []_{clause}
 was.said [to=him] [go]
 ‘It was said to him: go’.
- b. *junṭazaru* [*ʾan juthmira ʾamalu=nā*]_{subj}
 V_{PaAgPass.3.masc.sg} []_{clause}
 is expected [that get fruitful our hope]
 ‘It is expected that hope get fruitful results’.

5) Full suppression passive voice [FullSup-Pass]

A distinctive feature of the MTT approach lies in the definition of voice based on a deep syntactic level as an intermediate. Any expression that syntactically depends on L and manifests a SemA of L is a DSyntA of L. Yet, a displacement process can take place: L may govern a DSyntA not corresponding to one of its SemAs. According to MTT: “An added displaced DSyntA is by definition unrelated to any of L’s SemAs and therefore cannot modify the diathesis of L”⁴.

Arabic full suppressive passivization process consists of the raising of an adjunct to the subject rank; the adjunct denotes the nomina-

⁴ Mel’čuk, 2006, p. 192.

tive case mark and triggers verb agreement (only for gender, as it is usual in VSO order). Both DSyntA I and II are suppressed and the DSynt A III is promoted: [DSyntA III, CircCo, Accusatif] ⇒ [DSyntA I, Subject, Nominative].

X	Y	Z
–	–	I
–	–	Subject

In the examples (22), the DSyntAs III corresponding respectively to the SSynt circumstantial of time, place and manner are promoted to the SSynt subject rank by a full suppressive passivization process. The lexemes *laylatun*, *almadīnatu* and *farahun* cannot be analyzed as direct objects because they denote the nominative case mark /u/:

- (22) Time *suhirat* *laylatun* *mumti^catun*
V_{pass.3.fem.sg} N_{fem.NOM} ADJ_{NOM}
was.stayed night funny
‘The night was stayed funnily’
- place *qudijat* *almadīnatu*
V_{pass.3.fem.sg} N_{fem.NOM}
was. spent [time] [in] the city
‘The city was spent time in’.
- manner *furiha* *farahun* *kabīrun*
V_{pass.3.masc.sg} N_{masc.NOM} ADJ
was.rejoiced joy great
‘A great joy happened’

An individuated Agent generally controls the action. Yet, this is not an unrestricted rule. In the following example, the agent is a dog. The meaning of the verb *nubiha* ‘bark’ disallows the individualization of the agent:

- (23) manner *nubiha* *nibāhun* *shadīdun*
V_{pass.3.masc.sg} N_{masc.NOM} ADJ
was barked barking intensive
‘It was barked intensively’

The following figure presents the respective Semantic, DSynt and SSynt representations of the example (23). We note that the subject function is not filled by a semantic actant of the verb, and that the agent is not human.

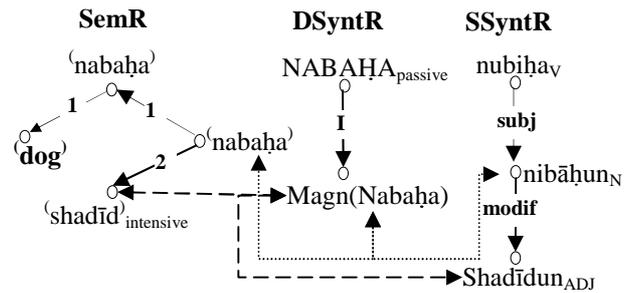


Figure 4: Full suppressive passivization process and non individuated agent

A "circumstantial aspect" may also be promoted to the subject rank. The "circumstantial aspect" or the "accusative of relation" is an indefinite singular noun corresponding to the SemA I of the verb (El Kassar, 2005). A synecdochic relation exists between the subject and the circumstantial aspect. The full suppressive passivization process consists of the suppression of the subject and the promotion of the circumstantial aspect to fill this function. In the following example, it exists a synecdochic relation between the lexeme /alkūbu/ ‘the glass’ and the lexeme /mā’an/ ‘water’. The first one is the subject of the verb /fāda/ ‘overflow’ at the active voice, while the second one is promoted to the subject rank at the full suppressive passive voice.

- (24) *fāda* *alkūbu* *mā’an*
V_{act.past.3.masc.sg} N_{masc.sg.NOM} N_{indef.ACC}
overflowed glass water
‘The glass overflowed of water’

↓

- fīda* *almā’u*
V_{FullSupPass.past.3.masc.sg} N_{masc.sg.NOM}
overflowed water
‘The water overflowed’

In brief, the verb in the full suppressive passive voice governs systematically a lexeme as a subject. We don't think that a pronoun could fill the subject function of the full suppressive passive voice.

6) Subject suppressive voice (SubjSupp)

This voice is commonly called *impersonal passive*. Like Slavic and some Romance languages, Arabic has no physical expression of impersonal pronoun. This analysis follows Teeple (2008), Saad 1982, Mohammad (1999), Fassi Fehri (1982), and Fischer (2002), but contrarily to them, we will not use the term *impersonal passive* that we find inaccurate. We will use rather the term *subject suppressive voice*. This voice occurs with indirect transitive verb: V (subject, oblique object). The DSyntA

⁵ In these examples, the abbreviation 'pass' refers to the full suppressive passive grammeme. We ought to this abbreviation for space reason, but in example (24), we will use the abbreviation 'Full-SupPass'.

I is suppressed while the DSyntA II keeps its SSynt oblique rank. The SSynt subject role is fulfilled by a linguistic sign having no meaning and a zero phonetic realization. The head verb is only in the 3rd masculine singular form.

X	Y
–	II
–	ObIO

We will content ourselves by mentioning that verbs accepting the subject suppressed voice may express, among others, a general or psychological situation (25a), a physiological (25b-c) state, or an action verb (25d). The ObIO in all cases expresses the experiencer. We will not go any further in the semantic classification which will need more details.

- (25) a. *'u'tunija* [bi=hā]_{ObIO}
V_{SubjSupp.3.masc.sg} [PREP=PRO_{fem.sg}]_{ObIO}
was.taken.care [of=him]
‘It was taken care of her’
- b. *'uġmija* [‘alaj=hi]_{ObIO}
V_{SubjSupp.3.masc.sg} [PREP=PRO_{masc.sg}]_{ObIO}
was.fainted [on=him]
‘He fainted’
- c. *ġurrira* [bi=him]_{ObIO}
V_{SubjSupp.3.masc.sg} [PREP=PRO_{masc.pl}]
deceived [of=them]_{ObIO}
‘They was deceived’
- d. *Jī'a* [bi=hindin]_{ObIO}
V_{SubjSupp.3.masc.sg} [PREP=N_{fem.sg.GEN}]
come [with-Hind]_{ObIO}
‘They brought Hind’

The oblique object may also express an action and be expressed by a masdar:

- (26) *sumiħa* [bi=alkhurūdzi]_{ObIO}
V_{SubjSupp} [PREP=N_{masdar}]_{ObIO}
was allowed [to leave]_{ObIO}
‘It was allowed to leave’

The subject suppressive process can lower the SSynt rank of the DirO in a detransitivization process, no internal argument is promoted to the subject rank. For example, in (27) below, the lexeme *almas'alata* ‘the issue’ fills the direct object function in the active voice and denotes the accusative case mark /a/. In the passive voice, the lexeme is promoted to the subject rank, takes the nominative case mark /u/ and governs the head verb agreement; while with the subject suppressive transformation, it is demoted to the oblique object rank and takes the genitive case mark /i/. The verb in this case is in the 3.masc.sg form and the subject is systematically a dummy zero pronoun.

- (27) *naṭara* [X]_{subj} [almas'alata]_{DirO}
V_{act.pr.3.masc.sg} [N_{def.fem.sg.ACC}]
reviewed [X]_{subj} [the issue]_{DirO.ACC}
‘X reviewed the issue’
- nuthirat* [almas'alatu]_{subj}
V_{pass.pr.3.fem.sg} [N_{def.fem.sg.Nom}]
was.reviewed [the issue]_{subj.NOM}
‘The issue was reviewed’
- nuthira* [fī almas'alati]_{ObIO}
V_{SuppPass.pr.3.masc.sg} [PREP N_{def.fem.sg.GEN}]
was.reviewed [in the issue]_{ObIO.GEN}
‘It was reviewed in the issue’

In the traditional Arabic grammar, the prepositional constituent is analyzed as the subject. In my opinion, this analysis is totally inaccurate.

In case of intransitive or monovalent verb, the subject suppressive transformation consists of the omission of all verb' actants. In the following examples, the verbs govern only a circumstantial of place or time. Again the subject is a dummy zero pronoun and the verb systematically in the 3.masc.sg inflectional form:

- (28) *thuhiba* [ilā manzilu=ka]_{Circ}
place was.gone [to house=your]_{Circ}
‘It was gone to your house’.
- dzulisa* [fī alġurfa]_{Circ}
place was.sat [in the.room]_{Circ}
‘It was sat in the room’.
- sufira* [jawmu alsabti]_{Circ}
time was.traveled [day Saturday]_{Circ}
‘It was traveled on Saturday’

As in a pro-drop language, impersonalization in Arabic means that the subject pronoun has no meaning and zero physical expression, which means that the subject function is fulfilled semantically, syntactically and morphologically by an empty actant. The analysis is rigorous yet the introduction of an empty element in this way jeopardizes its acceptability. The only justification of the presence of an empty subject in the sentence is to copy verb agreement. The following figure presents the representations of the sentence *dzulisa [fī alġurfa]* ‘It was sat in the room’. As we see, the subject does exist syntactically while it has no deep-syntactic or morphological existence.

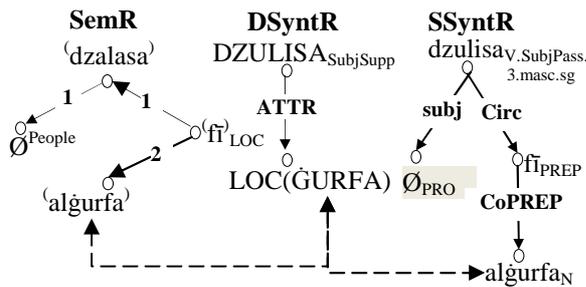


Figure 5: The subject suppressive voice

Describing the above sentence as an impersonal construction will not be accurate considering that there is no occurrence of a physical non endophoric pronoun like English IT; but it will be accurate if we consider that the first syntactic actant of the verb in the passive voice has no meaning: the primary semantic agent is not identified even if it is not empty and implies an individuated agent. The construction may have an existential reading: ‘[a specific person] stays in the room’.

4 Conclusion

In the present paper, we represented four categories of subject pronouns based on its endophoric function and phonetic realization. We described syntactic representation of unfamiliar structures where the subject pronoun exists only surface-syntactically. A particular attention was given to impersonal constructions. We criticize some traditional analysis considering that a prepositional phrase may fill the subject function; and stressed on the fact the impersonal construction is not necessarily translated by an impersonal construction in another language. Further studies may discuss several issues: the representation of this kind of pronoun in other multi-stratal dependency frameworks, its representation within a mono-stratal framework, and its frequency in Pro-Drop languages. It will also be interesting to study thoroughly government patterns and semantic classification of verbs heading no-meaning zero-phonetic subject pronouns in Arabic.

Acknowledgments

I would like to express my gratitude for Igor Mel’čuk and Yasmina Milicevic for accepting reading and commenting this work from its early phase.

References

- Ahmed Kamal El-Din Abdel-Hamid. 1972. *A transfer grammar of English and Arabic*. PhD dissertation. University of Texas, Austin.
- Abdelkader Fassi Fehri. 1982. *Linguistique arabe: forme et interprétation*. Press of the faculty of Letters and Human Sciences of Mohammed V University, Rabat.
- Wolfdietrich Fischer. 2002. *A Grammar of Classical Arabic*. Translated from the German by Jonathan Rodgers. Yale University Press, New Haven and London, UK.
- David Teeple. 2008. *The Arabic Impersonal Passive in OT*. University of California, Santa Cruz.
- Dina El Kassas. 2005. *Etude contrastive du Français et de l’Arabe dans une perspective de génération multilingue*. PhD dissertation, Paris VII University, Paris.
- Matthew S. Dryer. 2008. Expression of Pronominal Subjects. The World Atlas of Language Structures Online, ed. Haspelmath, Martin, Matthew S. Dryer, David Gil, and Bernard Comrie, chp. 101. Max Planck Digital Library, Munich. <http://wals.info/feature/101>. (Accessed on 2011-03-28.)
- Abd-Alkareem Massalha. 2005. *The Agentive Passive Construction in English and Its Translation into Arabic*. M.A. dissertation, University of Haifa, Haifa.
- Igor Mel’čuk. 1988. *Dependency Syntax: Theory and Practice*. The SUNY Press, Albany, N.Y.
- Igor Mel’čuk. 2006. *Aspects of the Theory of Morphology*. Ed. de Gruyter, Berlin - New York.
- A. Mohammad Mohammad. 1999. *Word Order and Pronominalization in Palestinian Arabic. Current Issues in Linguistic Theory*. John Benjamins, Amsterdam / Philadelphia.
- George Nehmeh Saad. 1982. *Transitivity, causation and passivization: A semantic-syntactic study of the verb in Classical Arabic*. Kegan Paul International ed. London.

The ‘Errant’ Scope of Question in Turkish:

A Word Grammar Account

Taiki Yoshimura

Research Institute for World Languages

Osaka University, Japan

taiki_wger@hotmail.com

Abstract

Previous studies dealing with the position of the interrogative clitic in Turkish, such as Besler (2000) and Aygen (2007), seem to be based on the assumption that the position of the interrogative clitic naïvely corresponds to the scope of question. However, Zimmer (1998) and Göksel and Kerslake (2005) point out that there are cases where the interrogative clitic is located in the pre-verbal position and attached to a word which is the dependent of the predicate, but the scope of question is the whole of the proposition rather than its specific part. In this article, I would like to argue that an analysis based on Word Grammar, a kind of dependency-based theories, successfully deals with these types of the ‘errant’ scope of the question, by showing a rich network concerned with semantic structure where some concepts concerned with the speech-act such as a speaker and an addressee are introduced, following Hudson (1990) and Hudson (2010).

1 Introduction

It is a well-known fact that the interrogative clitic (hereafter IC) *mI*¹ in Turkish forms yes-no or alternative questions. Unlike most other Turkic languages, *mI* in Turkish appears in various positions in a sentence so as to focus a particular part of the sentence. Let us first consider example (1) (Uzun, 2000: 301):

- (1) a. Ali kitab-ı Ayşe-ye ver-di mi?
Ali-Nom book-Acc Ayşe-Dat give-Past:3sg Q

¹ Following traditions of Turkish linguistics, variable vowels are shown by the capital letter in this article. For example, *mI* can occur as *mi/mu/mü/mü*.

‘Did Ali give Ayşe the book?’

b. Ali kitab-ı Ayşe’ye mi ver-di?

Ali-Nom book-Acc Ayşe-Dat Q give-Past:3sg

‘Is it to Ayşe that Ali gave the book?’

c. Ali kitab-ı mı Ayşe’ye ver-di?

Ali-Nom book-Acc Q Ayşe-Dat give-Past:3sg

‘Is it the book that Ali gave Ayşe?’

d. Ali mi kitab-ı Ayşe’ye ver-di?

Ali-Nom Q book-Acc Ayşe-Dat give-Past:3sg

‘Is it Ali who gave Ayşe the book?’

From these examples in (1), we can say that IC occurs not only in the sentence-final position but also in the sentence-middle position, in order to focus on the specific part of the sentence. If IC occurs with the verbal complex (i.e. the predicate) of the sentence, the scope of question is the whole of the sentence; on the other hand, when IC appears in sentence-middle and attaches to the specific word, then IC turns only the word immediately preceding itself into question. Taking these facts into consideration, as we shall see later, previous analyses have concentrated on how to predict the proper syntactic position of IC without violating any morpho-syntactic rule.

They have not, however, taken the Zimmer’s (1998) discussion into consideration; in some cases the scope of question is the whole of the proposition but IC at surface occurs in the pre-verbal position, which means that the position of IC does not always correspond to the semantic scope. In this article, therefore, I would like to argue that an analysis based on Word Grammar (hereafter WG) successfully handles the cases where the position of IC is at the pre-verbal position but the scope of question covers the whole of the sentence, by a rich

conceptual network proposed by Hudson (1990, 2007, 2010, among others).

2 A Brief Review of Previous Analyses

Besler (2000) and Aygen (2007) are outstanding studies which account for the appropriate positions of IC (which they call Q-particle). In these literatures, the assumptions about where IC is base-generated and moves afterwards are different from each other. Nevertheless, they both conclude that IC moves in order to focus either the whole of the sentence or the specific element of the sentence.

For all their well-developed analyses, it is worth pointing out that they ignore the fact that there are cases where IC is located in the preverbal position and attached to the word which is the dependent of the predicate, but the scope of question is the whole of the proposition rather than its specific part. In fact, as we shall see below, not only Zimmer (1998) but also Göksel and Kerslake (2005) point out this phenomenon; above all, Zimmer (1998) points out that the “standard accounts”, in which Besler (2000) and Aygen (2007) are thought to be included, fail to deal with the use of IC in certain types involving idiomatic expressions and some other types of sentences. Let us first consider (2), quoted in Zimmer (1998):

- (2) *Dalga mı geç-iyor-sun?*
wave Q pass-Prog-2sg
'Are you (just) wasting time?'

In (2), the noun *dalga* 'wave' and the verbal predicate *geçiyorsun* '(you are) passing' combine with each other, constituting an idiom whose meaning is 'wasting time'. In addition, the sentence (2) is a kind of yes-no questions and IC occurs in the preverbal position. Considering a series of example in (1), we may well predict that the scope of question is limited to the specific part *dalga*, but the scope of the question is actually the whole of the sentence rather than *dalga*. The similar cases are also found in less idiomatic sentences such as (3a) below:

- (3) a. *Nermin okul-a mı git-miş?*
Nermin-Nom school-Dat Q go-Evi-3sg
'Has Nermin gone to school?'
b. *Nermin okul-a git-miş mi?*
Nermin-Nom school-Dat go-Evi.-3sg Q
'Has Nermin gone to school?'

According to Göksel and Kerslake (2005: 294), the two questions exemplified in (3) cannot be used in the same context, although both turn the whole sentence into question. (3a) is used 'when the speaker has an assumption about the situation s/he is asking about, usually because there are non-linguistic clues (visual or perceptible by other senses)' (ibid.). On the other hand, sentences like (3b) are 'out-of-the-blue questions, where the speaker has no assumption about the situation' (ibid.).

It is worth pointing out that Zimmer suggests the pragmatic form for yes-no interrogative questions (which he calls 'focus questions') as in (4) (Zimmer, 1998: 480):

- (4) (X) Y *mI* Predicate (with sentence stress on Y)

In (4), X and Y are variables where Y is substituted by either a candidate for a role, or a state of affairs that the speaker has in mind, and *mI* (naturally enough) stands for IC. His argument seems to be good enough to account for the phenomena in question, but I would like to point out that it is not clear at all where we should place this formulate in the whole of grammar: he argues that it is the pragmatic form, but at once it must be the syntactic form because it consequently mentions word order. In short, it is necessary to propose the whole image of grammar at which the interrogative sentence is located. Additionally, it may be problematic that (4) itself does not explain when Y is substituted by a state of affair rather than a role, although Zimmer (1998) points out that this mismatch is seen in an idiomatic expression and some other expressions. To put it briefly, if we can predict the condition under which the mismatch happens, the analysis becomes more explanatory.

In summary, we have to explain the mismatch between the position of IC and its scope in meaning, to which most of previous studies do not refer. I would like to argue that a WG account successfully explains this mismatch, although Yoshimura (2010), which is based on WG, has also ignored this kind of mismatch. In the following sections, I will introduce the framework of WG (Section 3) and analyse every type of yes-no interrogative sentence marked by IC (Section 4).

3 Word Grammar: An Introduction

WG is a general theory of language structure, which Richard Hudson has been developing since early 1980s. In what follows, I would like to introduce the framework of WG to the extent that it is necessary for the discussion.

3.1 A Conceptual Network

WG treats the language structure as a network where concepts about words are linked in some relations. One of important relations between concepts in WG is the ‘isA’ relation, namely the model-instance relation between a general concept and a specific concept. For example, the English noun *cats* is an instance of a lexeme CAT, and of a plural noun, at the same time. These are described in terms of ‘isA relation’ in WG. As we can see in Figure 1 below, the word *cats* inherits several properties from two higher (and different) concepts.

In addition to the isA relation, most other relations are shown by links with arrows pointing from the word to other concepts. This is based on the following assumptions in WG: language structure consists of innumerable concepts stored (and learnt) in humans’ mind, a word is a kind of concepts, and there are two kinds of concepts, namely ‘entity concepts’ (e.g. ‘cat’, ‘plural noun’ in Figure 1) corresponding to people, things, activities and so on, and ‘relational concepts’ (e.g. ‘sense’, ‘form’ in Figure 1) which link a concept to another. WG also assumes that most concepts are learned (Hudson 2007: 232) to the extent that they are defined in terms of existing concepts a person stores in his/her mind. This is called Recycling Principle in WG, which enables us to make use of a rich semantic network without making semantic structure too complex.

Let us take a small network about a word *cats* for example. WG treats a word and its form as separate concepts, so a ‘form’ relation between CAT: plural at word-level and {cats} (in words, ‘the form of CAT: plural is {cats}’) is recognised. Similarly, there is also a ‘sense’ relation between CAT: plural and its target meaning that can be labelled ‘cat’ (in other words, the sense of CAT: plural is ‘cat’). These relations are shown by a curved arrow with a label written in an ellipse as shown in Figure 1. Note that WG clearly distinguishes words from forms. This is helpful if we account for the formal characteristics of IC. That is, the distinction enables us to show that IC in Turkish

is a syntactically independent element but a part of a larger word-form in morpho-phonology level (Yoshimura 2010). Another point is that the inflectional notion ‘plural’ is thought to be inherited by a noun, accordingly it is an instance of the more general category, ‘word’.

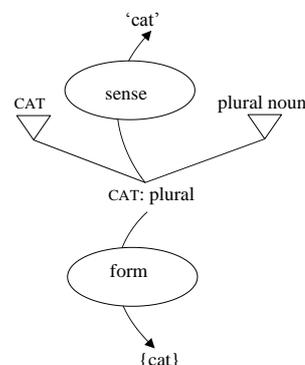


Figure 1. A Small Network of the Word *cat*

In WG, isA relation is represented by a straight line with a triangle, the base of which directs to the category. Taking Figure 1 for example, the word represented CAT: plural is an instance of (i.e. isA) the lexeme CAT. At the same time, it also ‘isA’ plural noun. As I said earlier, WG allows a concept to inherit properties from multiple super-categories.

3.2 A Word’s Properties in WG

According to Hudson (2010), one of the significant difference between WG and other theories is to clearly distinguish word types with tokens. One of the reason to do so is to explain various language-internal structure such as syntax and semantics. In this line of analysis, for example, tokens are kinds of actions, so it is helpful to illustrate tense and aspect in semantic structure, because their utterance-time are deeply relevant to event-time. For example, the time of referent of the past-tense verb is always followed by the time when the word token is uttered.² A token is categorized by being linked to some type, then it can inherit all the properties of this type.

One may well ask what the properties of a word are, or if any, how many properties there are. Notice that, it is pointless to establish a definition of a word; rather, as we have seen so

² Hereafter I shall not make a notational distinction between types and tokens in order to avoid complexity of notation.

far, words are also instances of concepts, thus a word in itself should be a concept where there is a bundle of properties. Hudson (2010: 114-116) introduces a handful of relational properties of a word, such as meaning, a realization (i.e. a form and sounds), a word-class, and so on. For the discussion in this article, the properties ‘speaker’ and ‘addressee’ are important, as we shall see below. Notice that here, too, the distinction between types and tokens is important: some properties belong to tokens, but not to types.

According to Hudson (2010), properties such as a speaker and an addressee of a word belong primarily to word-tokens. In this article, too, I shall follow the idea of the type-token distinction proposed in Hudson (2010), in order to introduce two important concepts for explanation of the semantic structure of the interrogative sentence: the speaker and the addressee of a word.

3.3 Sense, Referent and Semantic Phrasing in WG

In WG semantics, the distinction between ‘referent’ and ‘sense’ is important as in other theories: a word’s sense is some general category, and its referent is typically some particular instance of this category. This distinction is clearly represented in the network diagram. Consider the following simple sentence, whose semantic network is illustrated in Figure 2:

- (5) Bir kedi gel-di.
A cat-Nom come-Past: 3sg
‘A cat came.’

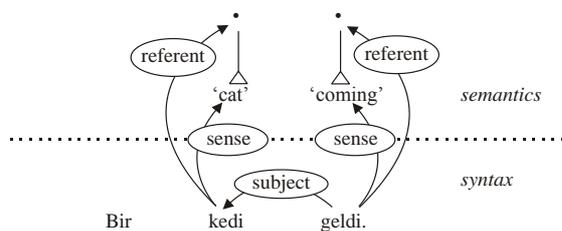


Figure 2. Sense and Referent

Figure 2 above shows this distinction, where the referents of words and their sense are linked by the isA relations. Notice that the dotted nodes are concepts which are difficult to find natural-language names; it may seem to be problematic, but in WG this does not matter because any nodes (or relational links) are simply mnemonics for our own purposes and have no theoretical status (Hudson, 2007:18).

In any case, the sense/referent distinction plays an important role in our purpose; as we shall see below, it is supposed that IC in Turkish shares a referent with any other word. Usually it is the preceding word of IC that shares the referent, but if the word has no referent (or does not refer to any particular concept), then IC shares the referent with its adjacent word, i.e. the referent of the predicate.

Another point that plays a crucial role in our analysis is that a word’s sense is affected by some other words, i.e. dependents. In WG, this is demonstrated by a hierarchical structure which is called semantic phrasing. Hudson (2010: 228) assumes that there are at least four patterns that a word’s meaning is affected by a dependent. Of these, the default pattern (i.e. the dependent’s referent combines with the word’s sense), coreference (i.e. the dependent’s referent merges with the word’s referent), and idioms (i.e. the dependent changes the word’s sense in an irregular way, which is exemplified in (2)) are necessary for the discussion.³ Let us consider these patterns below.

First, we consider the default pattern: combination of the dependent’s referent with the sense of its parent. Taking our stored example *Köpek havladı*. ‘(A/the) dog barked’,⁴ the word token *köpek* ‘dog’ is the subject of the predicate word token *havladı* ‘barked’, so *köpek* modifies the meaning of *havladı* which is inherited from the lexeme HAVLA-. The point is that the sense of HAVLA- is simply ‘barking’, but as we have seen so far, word-tokens has their own senses; in this case, the word token *köpek* changes not the sense of the lexeme HAVLA-, but that of the word token *havladı*. This becomes clearer from examples such as (6):

- (6) köpek havla-dı, fakat daha önce
dog-Nom bark-Past:3sg but more before
öyle bir şey tek bir kez ol-muş-tu.
such a thing only one time be-Rep-Past:3sg
‘The dog barked, but which had only once
happened before.’

In (6), the reading of the sentence should be that there are two incidents of ‘(the) dog bark-

³ The last type of semantic phrasing is predicative pattern, where a word’s sense combines with that of its dependent. See Hudson (2010: 232-233) for more detail.

⁴ This ambiguity depends on the context: there is no obligatory definite determiner in Turkish, although *bir* can be an indefinite determiner.

ing’, so the relation between *havladı* and the demonstrative *öyle* (possibly with the following noun *şey* ‘thing’) must be identity-of-sense anaphora. Accordingly, the subject of the first clause *köpek* modifies the sense of *havladı*, rather than the referent of it. This is of course true of other languages such as English. As Hudson (2010: 229) points out, if we hanged *some dogs* (in English) into *some big dogs*, the dependent *big* changes the sense into ‘big dog’, but does not change the referent set into a big set.

Turning to the meaning of the dependent, it is the dependent’s referent which modifies the parent’s sense. This is clear from the fact that some nouns such as pronouns and proper nouns, which can be the subject of the predicate, have only their referents but do not have any sense. To conclude, the referent of a dependent word, by default, modifies the sense of its parent word. Our stored example *Köpek havladı.* is, therefore, analysed as in Figure 3 below:

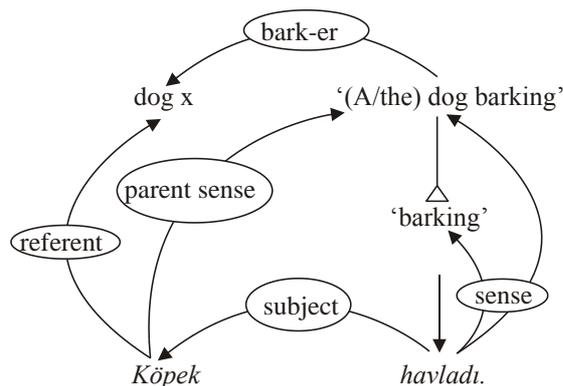


Figure 3. The Small Semantic Network of ‘*Köpek havladı*’ in Turkish

In Figure 3, a new link labelled ‘parent sense’ (Hudson, 2010: 229) is introduced. By this link ‘(A/the) dog barking’ and more basic sense ‘barking’ is successfully distinguished. This link is helpful when we show the details of modification by dependents, because there may be two or more dependents of a word.

The second pattern is coreference, where the two words share the same referent. Taking *a cat* in English for example, both words refer to a single concept: a single countable dog. It may seem that the very similar analysis applies to the translation equivalent *bir kedi* in Turkish. Assuming that *bir* is an instance of pronoun, this word confirms that the referent is again a single entity, and that it is indefinite. Like the analysis proposed in Hudson (2010: 229-230),

the co-reference of these two words, i.e. *bir kedi* (‘a cat’), is reflected in Figure 4, which is, in consequence, a slightly developed illustration of Figure 2:

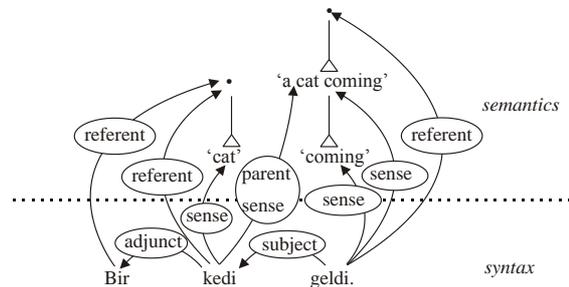


Figure 4. The Small Network of ‘*Bir kedi geldi*’ in Turkish

One may argue that there is no other ‘article’ than the indefinite *bir* in Turkish, but co-reference relation is in fact needed for some cases because there are a handful of ‘determiners’ (including pronouns) which indicates the definiteness or the indefiniteness of their following noun.⁵ According to Göksel and Kerslake (2005), there is a handful of determiners which are thought to function as the articles (i.e. *a/an* and *the*) in English. It seems, therefore, that we can assume that this type of semantic phrasing is also applicable to Turkish.⁶

According to Hudson (2010), coreference is not only found in pronouns, but also in some auxiliary verbs and prepositions. In English, the combination between auxiliary and the main verb such as *will bark* show that their co-referent is a single event, whose time is set in the future, and in another combination between a noun and the preposition such as *a book by Dickens*, the preposition *by* shares the referent of *Dickens*, where it associates the author Dickens with the book as a result (Hudson 2010: 230). As I suggested, this semantic phrasing pattern applies to IC and its pair word in Turkish.

The last type is idiomatic combination, where the effect of the dependent is unpredictable. In English, a very well-known example is

⁵ WG assumes that there is no point to recognize a category ‘determiner’ in English for several reasons (Hudson, 1990); I assume here that the same is true of Turkish. That is, there is no need to recognize the category ‘determiner’ in Turkish. Instead, this category can be recognized as a subcategory of ‘pronoun’.

⁶ For more detail about determiners in Turkish, see Göksel and Kerslake (2005: 201-203).

the combination KICK THE BUCKET, whose meaning is, say, ‘dying’. The analysis of this example in terms of WG is as Figure 5 (Hudson, 2010: 234) shows:

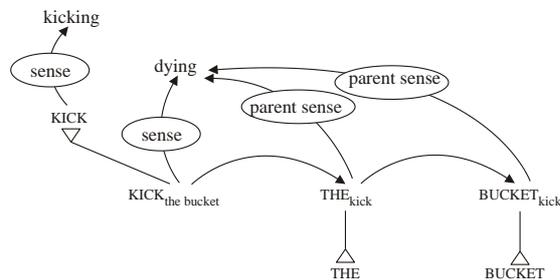


Figure 5. The Idiom ‘KICK THE BUCKET’ (Hudson 2010: 234)

The point of Figure 5 is that each of the words in the idiom is a special sublexeme of an ordinary lexeme. Above all, the sense of *KICK_{the bucket}* is ‘dying’ rather than ‘kicking’, which is possible because *KICK_{the bucket}* is a *KICK* but *KICK_{the bucket}* has its own sense ‘dying’. The solution that WG offers is the process called Default Inheritance: the specific property of the sub-category overrides the default. So in this case, the sense of the sublexeme *KICK_{the bucket}*, ‘dying’, overrides the default sense of the more general lexeme *KICK*.

The analysis is also applicable to examples in Turkish. There are so many idioms in Turkish, where they demonstrate some kind of gradient, in that the senses of some idioms are predictable from individual words, but the others do not. Our concern here is, of course, how to explain idioms whose meaning cannot be deductible from each word. One of our stored examples is *dalga geç-* ‘wasting time’, where *dalga* is the noun whose basic meaning is ‘wave’, and *geç-* is the verb whose meaning is ‘passing’. So this example is clearly an idiom because the whole meaning cannot be predictable from individual lexemes. Taking the analysis of the example from English shown in Figure 5 into consideration, the WG account of the idiom in question will be like Figure 6 below:

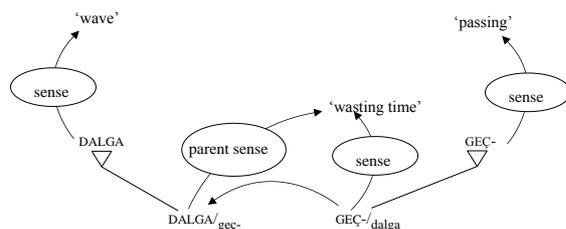


Figure 6. The Idiom DALGA GEÇ-

3.4 Two New Concepts: ‘Factuality’ and Its ‘Knower’

Our concern is the semantic structure of the interrogative sentences in Turkish. The semantic structural difference between the declarative and the interrogative sentence is, if any, crucial in our analysis. Let us begin by discussing the following question sentences.

- (7) a. Siz ne ye-di-niz?
 you(Hon)-Nom what eat-Def. Past-2pl
 ‘What did you eat?’
 b. Siz yemek ye-di-niz mi?
 you(Hon)-Nom meal eat-Def.Past-2plQ
 ‘Did you have a meal?’

Example (7a) is a kind of WH-questions, where there is no IC in the sentence.⁷ In contrast, IC marks the so-called yes-no question as in (7b), and if WH-word and IC co-occur in the sentence, the sentence will not be grammatical unless the sentence is interpreted as an echo-question of the whole sentence.

Taking the data shown above into consideration, WG introduces new relational concepts that are, even though they are somewhat tentative, responsible for their illocutionary force such as declaration, command, and question: ‘knower’ and ‘factuality’ (Hudson 1990). In WH-questions, the speaker does not know some information about the event, such as who does it and what the person does. The speaker therefore asks the addressee a question, assuming that the addressee knows it. In the speaker’s mind, therefore, the addressee is the ‘knower’ of the concept which the speaker wants to know. These are illustrated in Figure 7 below, where the rather plain sentence *Kim geldi?* ‘who came?’ is analysed, and ‘addressee’ links come to the concept which the knower of the person who came, from both *kim* and *geldi*.

⁷ Except for the so-called echo questions, WH pronouns and IC do not co-occur in Turkish.

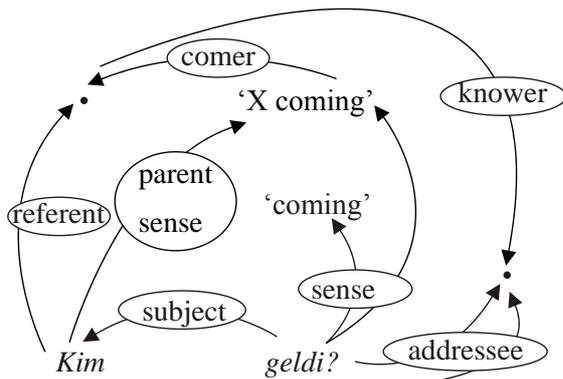


Figure 7. The Analysis of *Kim geldi?* ‘Who came?’ in Turkish

Another concept ‘factuality’ is a relational property whose value is either factual or non-factual (Hudson, 1990: 223), which intelligibly corresponds to the yes-no question. Factuality involves a choice-set (Hudson, 2010): either factual or non-factual, and they are contradictory each other. In the case of yes-no questions too, as well as WH-questions, the speaker assumes that, regardless of whether the speaker’s guess is right or not, the addressee is the knower of the factuality of the referent in question. The analysis of our stored example in (7b), an example of yes-no questions, will be like Figure 8. It should be noted that labels for syntactic relation between words are omitted for the sake of simplicity.⁸

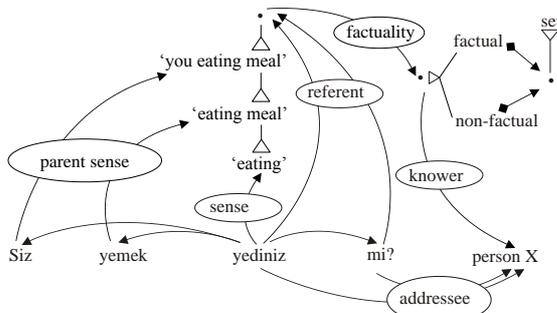


Figure 8. ‘Factuality’ and ‘Knower’ for the Functional Explanation of Yes-no Question

As I said earlier on, I suppose that IC has no sense in itself, but shares the referent of the word to which it attaches: the verbal predicate in this case. This is simply because it is hard to imagine the typical meaning of IC and it does not affect the sense of any word. Instead, I

⁸ At the upper right of Figure 8, there are two straight arrows with a diamond at its base. These arrows show that they are members of a particular set and they are exclusive each other. Such relation is called ‘or-relation’ (Hudson, 2010).

suggest that IC by default has to share the referent of the word immediately before itself, and the concept referred by IC is what the speaker want to ask whether it is factual or non-factual. In other words, I suggest, the reason why IC occurs in various positions in the sentence is to co-refer to the concept which is what the speaker likes to ask.

As we saw in Section 1, IC appears not only in sentence-final, but also in sentence-middle to show the scope of question. I shall deal with the scope of question according in order basically to the position of IC in the next section; at this stage, it is sufficient to confirm that the function of the interrogative sentence can be explained by a rich network provided by semantic (and possibly pragmatic) network in WG.

4 The Analysis

One may doubt whether WG can explain the scope of question which is exhibited by the position of IC in syntax and morphology, because it does not use phrase structure which works well in showing the scope of question by some asymmetry in phrase structure such as c-command. I suggest that, however, it is easy for WG to provide a solution to the problem by displaying the scope of question in the area of semantic networks whose logic is quite similar to the rest of language structure including syntactic dependencies.

We have already seen the cases where IC comes up in the sentence-final position in 3.4. If the assumption that attaching IC to the predicate of the sentence turns the whole clause into a question is right (except for some cases we have pointed out in Section 2), then this statement is easily shown in a WG network as in Figure 8. As we have seen so far, the problem lies in the cases where IC occurs in the preverbal position but the scope of question is different.

4.1 IC in Sentence-middle Focussing on the Specific Part of the Sentence

In the cases of *mi* in a sentence-middle position, the scope of question is restricted to a particular word (or constituent if it is applicable). This is easy to display in the WG network; let us show the network of our earlier example (1b) in Figure 9.

Considering the contrast between the ‘ordinary’ patterns of the scope of question, the most important point in Figure 10 is that there is only one judgement carried out by the speaker: a judgement about whether the event is an instance of ‘wasting time’ or not. This clearly corresponds to what we call the thetic reading, where the speaker simply likes to confirm whether the event itself is factual or non-factual.

The remaining problem is the analysis of cases where the meaning of the predicate is less idiomatic, with IC being located at the immediately preverbal position. We have already seen such cases exemplified in (3a), where IC attaches to the preverbal word *okula*, but the scope of question is the whole of the sentence, showing that the speaker has an assumption that the person in question has gone somewhere or not. In this case, too, the similar explanation to the cases of highly idiomatic expressions is possible. That is to say, the speaker’s judgement is oriented towards the referent of the predicate. The analysis of the example (3a) will be like Figure 11 below, where the referent of the noun *okula* ‘to school’ is not recognized:

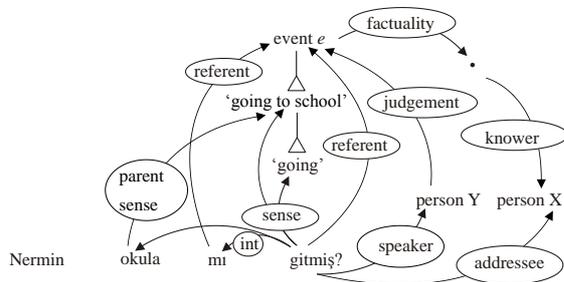


Figure 11. The Analysis of the Sentence *Nermin okula mı gitmiş?* ‘Has Nermin gone to school?’ in Turkish

Finally, we have to show how the grammar permits IC to show the errant scope. The problem is that IC shares the referent with the preceding word in semantics by default, but in the errant case it does not. A solution offered in WG is to apply default inheritance, the logical operation in the theory. That is, we assume that there are several subtypes (i.e. sub-lexemes) of IC including one that has the errant scope of question. By definition, all subtypes including IC in emphatic and subjunctive use isA IC, the more general category, and each subcategories inherit all properties unless they already have their own conflicting properties. The isA hier-

archy of some types of IC in Turkish is illustrated as in Figure 12:

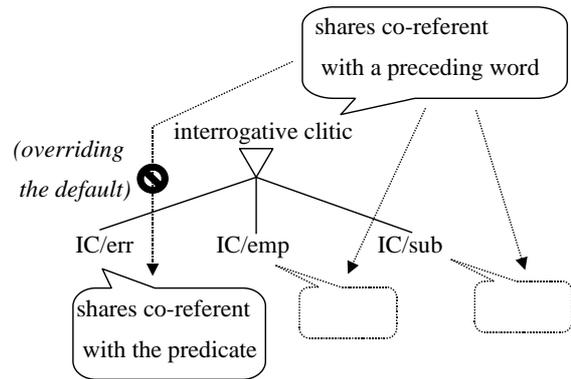


Figure 12. The IsA Hierarchy of IC in Turkish

In figure 12, IC/err (i.e. IC whose scope is errant) has its own property in that it shares the referent with the predicate of the sentence and therefore overrides the default one. The point is that the theory allows exceptions, so even the errant scope of question does not cause any problem in the grammar.

To sum up, the mismatch between the position of IC and its scope of question is purely the matter of semantics and/or pragmatics, as Zimmer (1998) and Göksel and Kerslake (2005) point out. This mismatch may seem difficult to incorporate into grammatical structure at first sight. However, this is easy for an analysis using WG to account for this mismatch, by recognizing a handful of concepts relevant to the speech act. WG provides a rich network of concepts, most of which are open-ended except for a limited number of primitive concepts such as the isA relation. This conceptual network enables us to refer to semantico-pragmatic factors in grammar.

5 Conclusion

In this article, I argued that our analysis in terms of WG covers all the patterns which concern IC and its scope of question. The analysis is applicable regardless of whether IC is in the sentence-final position or the sentence-middle position. Also, it is unnecessary to assume any syntactic movement rule, which is taken for granted in some works within the Generative Grammar framework such as Bessler (2000) and Aygen (2007). What is more important is that there are cases where there is a mismatch between the position of IC and the scope of question. We solved the problem by recognizing a rich network including concepts

relevant to pragmatics, which compensates for some weak points Zimmer (1998) has: relating pragmatic factors to syntactic structure and predicting when the mismatch concerned with IC between semantics and syntax happens.

The analysis offered so far, contrary to Previous analyses such as Besler (2000) and Aygen (2007), dispenses with any syntactic rules such as movement of IC. In this sense, other non-transformational theories seem to handle the mismatch between the position of IC and the scope of question. However, not many non-transformational framework can deal with this mismatch. That is to say, the concepts ‘speaker’ and ‘addressee’ are not available unless a distinction between word-types and word-tokens is made in the theory because ‘speaker’ and ‘addressee’ are typically concerned with word-tokens rather than word-types. As I pointed out in 3.1, they are clearly distinguished in WG. To avoid complexity, I have not shown this distinction in diagrams drawn throughout this article.

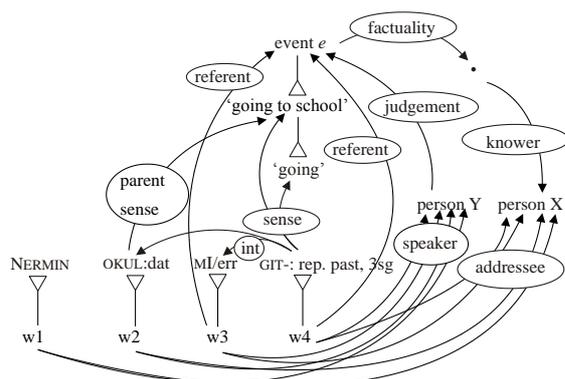


Figure 13. An Elaborated Analysis of Figure 11

As shown in Figure 13, it is easy to demonstrate this distinction in WG: tokens are labelled as ‘w1’, ‘w2’, and so on. Each word token, assumed to be linked to a corresponding word-type, and the relational properties ‘speaker’ and ‘addressee’, therefore comes up from tokens rather than types. If most other theories, as Hudson (2010: 111) points out, pay very little attention to this distinction, then our WG-based analysis is among a few theories which can correctly incorporate the speech-level concepts into the rest of grammar.

Acknowledgments

I would like to thank Prof. Kensei Sugayama for his continuous supports and useful comments for my earlier draft, three anonymous

reviewers who gave comments and suggestion, and an anonymous informant for judging the Turkish data shown in this paper other than those from other literatures. All remaining errors are however entirely mine.

This research is supported by the “Linguo-Cultural Contextual Studies of Ethnic Conflicts of the World” project (LiCCOSEC) of Research Institute for World Languages, Osaka University, JSPS Grant-in-Aid Scientific Research (C) (23520587) organised by K. Sugayama, and JSPS Grant-in-Aid Scientific Research (A) (21251006) organised by Tomoyuki Kubo.

References

- Aygen, Gülşat. 2007. Q-particle. *Journal of Linguistics and Literature* (Mersin University) 4(1): 1-30.
- Besler, Dilek. 2000. *The Question Particle and Movement in Turkish*. Unpublished MA thesis, Boğaziçi University, Istanbul.
- Göksel, Aslı and Kerslake, Celia. 2005. *Turkish: A Comprehensive Grammar*. Routledge, New York.
- Hudson, Richard. 1990. *English Word Grammar*. Blackwell, Oxford.
- Hudson, Richard. 2003. Encyclopedia of English Grammar and Word Grammar. <http://www.phon.ucl.ac.uk/home/dick/enc.htm> (14 May, 2011)
- Hudson, Richard. 2007. *Language Networks: The New Word Grammar*. Oxford University Press, Oxford.
- Hudson, Richard. 2010. *An Introduction to Word Grammar*. Cambridge University Press, Cambridge.
- Kuroda, Sige-Yuki. 1972. The Categorical and the Thetic Judgment: Evidence from Japanese Syntax. *Foundations of Language* 9: 153-185.
- Uzun, N. Engin. 2000. *Evrensel Dilbilgisi ve Türkçe (Universal Grammar and Turkish)*. Multilingual, Istanbul.
- Yoshimura, Taiki. 2010. The position of the interrogative clitic in Turkish: a Word Grammar account. Paper read at the 15th International Conference on Turkish Linguistics.
- Zimmer, Karl. 1998. The case of the errant question marker. In Johanson, Lars (ed.). *The Mainz Meeting: Proceedings of the Seventh International Conference on Turkish Linguistics*. Harrassowitz Verlag, Wiesbaden: 478-481.

Wh-Copying in German as Replacement

Andreas Pankau

Goethe Universität Frankfurt am Main &
Universiteit Utrecht

A.Pankau@lingua.uni-frankfurt.de

Abstract

This paper offers an argument for the necessity of adopting grammatical relations as primitives in syntactic theory. The argument is based on the phenomenon of wh-copying in German. Wh-copying in German poses a problem for approaches based on phrase structure (PS) representations because the construction is governed by two generalizations which a PS approach fails to capture. As soon as a relational perspective on syntactic structures is adopted, however, the generalizations can be captured. I will present an analysis for wh-copying in German within the Arc Pair Grammar framework, which does adopt such a relational view. It will be shown that the operation *Replace* in interaction with other principles of that framework successfully captures the two generalizations of wh-copying in German, and that it eventually even allows one to reduce the two generalizations to a single one.

1 Introduction

In this paper, I will deal with the proper analysis of wh-copying in German and argue that such an analysis can only be arrived at if grammatical relations are adopted as primitives for syntactic theory. The paper is organized as follows. In section 2, I will give a brief overview of the construction in German and argue that it is a regular extraction construction. In section 3, two specific generalizations of wh-copying in German will be established. Section 4 shows that phrase structure approaches fail to express these generalizations. Section 5 presents an analysis for wh-copying within Arc Pair Grammar that succeeds expressing these two generalizations. I will finally deal with a restriction found in German that restricts the shape of both extracted and resuming element simultaneously and show that this variation can be easily explained by the analysis.

2 Wh-Copying in German

Wh-copying is a construction in which an **extracted**¹ **wh-phrase** originating in an embedded clause is taken up by a resuming element in the initial position of the embedded clause, cf. (1).

- (1) **Wen** glaubst du wen sie t² liebt?
who believe you who she loves
'Who do you think she loves?'

In (1), the direct object of the verb *lieben* (to love) is extracted to sentence initial position. The construction is however not confined to direct objects: subjects (cf. (2)) and indirect objects (cf. (3)) are extractable, too, among others.

- (2) Wer glaubst du wer t Maria liebt?
who believe you who Mary loves
'Who do you think loves Mary?'
- (3) Wem denkst du wem sie t geholfen hat?
who think you who she helped has
'Who do you think she helped?'

Wh-copying is arguably a subspecies of regular extraction, that is, it is structurally similar to the more familiar type of extraction as in (4).

- (4) Wen glaubst du dass sie t liebt?
who believe you that she loves
'Who do you think she loves?'

This is backed up by a couple of arguments of which I would like to mention four. First, similar to more regular extraction, wh-copying is in principle unbounded, that is, it can target any embedded clause.

- (5) Wen denkst du wen sie meint wen er t liebt?
who think you who she means who he loves
'Who do you think she believes he loves?'

Second, it is island sensitive, which I have illustrated with the subject island in (6) and the complex NP island in (7).

- (6)*Wen hat [_{SUBJ} wen sie t liebt] alle überrascht?
who has who she loves all surprised
'Who did that she loves surprise everyone?'

¹ 'Extraction' refers to a class of construction and not to an operation where an element is linearly reordered.

² 't' is a mnemonic device indicating the position of the extracted element in the structure without extraction.

- (7) * Wen machte Peter [_{NP} die Behauptung
who made Peter the claim
[_S wen sie t liebt]]?
who she loves

'Who did Peter make the claim that she loves?'

Third, wh-copying is possible only with a handful of verbs in the matrix clause, viz. only with so called bridge verbs, to which *fragen* (to ask) does not belong.

- (8) * Wen fragst du wen sie t liebt?
who ask you who she loves
(9) * Wen fragst du dass sie t liebt?
who ask you who she loves
'Who did you ask she loves?'

Fourth, wh-copying shows connectivity effects (Jacobson 1984), by which one refers to the fact that an extracted element has to satisfy restrictions imposed on it by the selecting element. For example, the German predicate *'sich sicher sein'* (to be sure of) selects a genitive marked NP.

- (10) Sie ist sich dessen sicher.
she is self that sure
'She is sure of that.'

If this element is extracted, the case is retained.

- (11) Wessen glaubst du wessen sie sich sicher ist?
whose believe you whose she self sure is
'What do you think she is sure of?'

The last two points are not trivial, because they indicate that wh-copying is not a subtype of the 'scope marking' construction, illustrated in (12), which is often treated on a par with wh-copying (Höhle 2000)³, and for which an analysis similar to regular extraction is very problematic, at least for German (cf. Klepp (2002)).

- (12) Was glaubst du wen sie t liebt?
what believe you who she loves
'Who do you think she loves?'

The set of bridge verbs for scope marking is different from the one for wh-copying and regular extraction. On the one hand, it excludes raising predicates and volitional verbs, both of which are possible for wh-copying and regular extraction. I have illustrated this for raising predicates.

- (13) *Was scheint es wen Hans t geschlagen hat?
whom seems it that Hans beaten has
(14) Wen scheint es wen Hans t geschlagen hat?
whom seems it whom Hans beaten has

- (15) Wen scheint es dass Hans t geschlagen hat?
whom seems it that Hans beaten has
'Who does it seem that Hans hit?'

On the other hand, it includes verbs that are impossible as bridge verbs for wh-copying and regular extraction, such as *vermuten* (engl. to suppose) or *befürchten* (engl. to fear); it is illustrated only for the first verb.

- (16) Was vermutest du wem sie t hilft?
what suppose you who she helps
(17) * Wem vermutest du wem sie t hilft?
who suppose you who she helps
(18) * Wem vermutest du dass sie t hilft?
who suppose you that she helps
'Who do you suppose she helps?'

Regarding connectivity effects, they do not hold in scope marking, in which the extracted element nearly always surfaces as *was* (what), which is not genitive marked, as shown in (20).

- (19) Was glaubst du wessen sie sich t sicher ist?
what believe you whose she herself sure is
'What do you think she is sure of?'
(20) * Was ist sie sich t sicher?
what is she self sure
'What is she sure of?'

3 Two Generalizations about Wh-Copying

Wh-copying in German is characterized by two specific generalizations concerning extracted and resuming element, which I will describe now.

3.1 Generalization I: Agreement

Many speakers license PPs in wh-copying.

- (21) Mit wem meinst du mit wem sie t tanzt?
with whom mean you with whom she dances
'Who do you think she dances with?'

In this case, the extracted and the resuming element have to agree in category. The non-agreeing forms of (21) are all ungrammatical.

- (22) * Wem meinst du mit wem sie t tanzt?
who mean you with whom she dances
(23) * Mit wem meinst du wem sie t tanzt?
with whom mean you whom she dances
'Who do you think she dances with?'

Crucially, this is not a connectivity effect because this agreement requirement extends to cases where extracted and resuming element do not agree, but satisfy connectivity. Consider the verb *schreiben* (to write). The indirect object either surfaces as a PP or as a dative-marked NP.

- (24) Sie schreibt (ihm) einen Brief (an ihn).
she writes him a letter on him
'She writes (him) a letter (to him).'

³ The proper analysis of this construction is hotly debated (Fanselow 2006). For some, the *was* in (12) is a dummy element indicating directly the scope of the real wh-phrase *wen* in clause initial position. For others, the *was* is an extracted sentential expletive of the matrix verb, and the scope of the real wh-phrase *wen* comes about indirectly such that the embedded clause defines the relevant restriction for the beliefs the speaker asks for. If true, a more adequate translation for (12) is 'What do you think? Who does she love?'

Now consider the following examples.

- (25) * Wem denkst du an wen sie t schreibt?
 (26) *An wen denkst du wem sie t schreibt?
 (27) Wem denkst du wem sie t schreibt?
 (28) An wen denkst du an wen sie t schreibt?
 (on) whom think you (on) whom she writes
 ‘Who do you think she writes to?’

If only connectivity were at work in wh-copying, all sentences in (25)-(28) should be grammatical because in all cases both extracted and resuming element are compatible with the verb (cf. (24)), and should therefore be correctly connected. But only (27) and (28) are grammatical. The reason is that only in these sentences, extracted and resuming element agree, viz. in their categorial status. Agreement shows up in other contexts as well. The indirect object NP of the verb *lehren* (to teach) bears either accusative or dative.

- (29) Ich lehre ihm/ihn Latein.
 I teach him.dat/him.acc Latin
 ‘I teach him Latin.’

In wh-copying, the following pattern emerges.

- (30) * Wem denkst du wen er t Latein lehrt?
 (31) * Wen denkst du wem er t Latein lehrt?
 (32) Wem denkst du wem er t Latein lehrt?
 (33) Wen denkst du wen er t Latein lehrt?
 who think you who he Latin teaches
 ‘Who do you think he teaches Latin?’

Again, only those sentences are grammatical in which extracted and resuming element agree, this time for case. Finally, agreement extends to NPs as complements of prepositions. For example, the German verb *sich verlieben* (to fall in love) requires a PP headed by *in*, whose complement NP can bear any gender marking.

- (34) Er verliebte sich in den/das.
 He fell.in.love self in that_{masc}/that_{neut}
 ‘He has fallen in love with him/that.’

If extracted, the gender marking has to remain constant on both extracted and resuming element, that is, they have to agree for this feature.

- (35)* In wen denkst du in was er sich t verliebte?
 (36)* In was denkst du in wen er sich t verliebte?
 (37) In wen denkst du in wen er sich t verliebte?
 (38) In was denkst du in was er sich t verliebte?
 in wh. think you in wh. he self fell.in.love
 ‘Who/what do you think he fell in love with?’

The agreement requirement for NPs as complements to PPs holds for other morphological features, too, such as number and case, which due to lack of space I have not illustrated.

To sum up, agreement between extracted and resuming element extends to cases not covered by connectivity so that consequently it must be treated as a separate generalization.

3.2 Generalization II: Proforms Only

Wh-copying in German is subject to the curious restriction that complex wh-phrases, that is, wh-phrases consisting of a determiner and a restriction, are excluded (McDaniel 1986; Höhle 2000).

- (39) * Welchen Mann glaubst du
 which man believe you
 welchen Mann sie t liebt?
 which man she loves
 ‘Which man do you think she loves?’

In the literature on wh-copying, this is often interpreted as a constraint licensing only wh-pronouns in the construction, while barring complex wh-phrases from it in general (Felser 2004, Höhle 2000, Nunes 2004). This view however is too simplistic in light of data that are almost never taken into account. First of all, it is not the case that only wh-pronouns appear as resuming elements. Already McDaniel (1986) noted that some speakers license d-pronouns.

- (40) Wen glaubst du den sie t liebt?
 who believe you who she loves
 ‘Who do you think she loves?’

She also noted that this extends PPs, that is, in case a PP is extracted, the speakers also license d-pronouns as complements to a preposition.

- (41) Mit wem denkst du mit dem er t spricht?
 with whom think you with whom he speaks
 ‘With whom do you think he talks?’

Second, it is equally not true that complex wh-phrases are generally excluded. Anyadi & Tamrazian (1993) reported that some speakers license structures such as (42) and (43).

- (42) Welchem Mann glaubst du wem sie das
 which man believe you who she the
 Buch t gegeben hat?
 book given has

‘Which man do you think she gave the book to?’

- (43) Mit welchem Werkzeug glaubst womit Ede
 with which tool believe you with.what Ede
 das Auto t repariert?
 the car fixes

‘With which tool do you think Ede fixes the car?’

Although not all speakers license such sentences, they are robustly attested. In a data collection carried out recently, I was able to find five speakers⁴ licensing them. As such structures were not investigated before, their properties were unclear. The aim of the data collection was to fill this gap. Eventually, four results could be established. First, only a specific set of pronouns

⁴ Three came from the Lower Rhine area, one from Saxony, one from Bavaria. This is in line with the observation that wh-copying is not a dialectal phenomenon (Höhle 2000).

is available as resuming elements. Personal pronouns, for example, are excluded altogether.

(44) * *Wen glaubst du ihn sie t liebt?*
whom believe you him she loves
'Who do you think she loves?'

(45) * *Mit wem glaubst du mit ihm sie t tanzt?*
with wh. believe you with wh. she dances
'With whom do you think she dances?'

Second, if a speaker licenses d-pronouns as resuming elements, then he will also license them as free relative pronouns, that is, as elements introducing free relative clauses. In other words, the same speakers accepting (40) and (41) also accepted the sentences (46) and (47).

(46) *Ich lade ein den alle t mögen.*
I invite who everyone likes
'I invite who everyone likes.'

(47) *Ich treffe mich mit dem sie t getanzt hat.*
I meet with whom she danced has
'I met up with whom she danced.'

Third, if speakers license complex wh-phrases in wh-copying, then they only license them as extracted elements. Sentences such as (48) and (49) were uniformly rejected.

(48) * *Wem glaubst du welchem Mann sie das Buch t gegeben hat?*
whom believe you which man she the book given has
'Which man do you think she gave the book to?'

(49) * *Mit wem glaubst du mit welchem Mann sie t getanzt hat?*
with whom believe you with which man she dances has
'Which man do you think she has danced with?'

Fourth, speakers licensing complex wh-phrases as extracted elements also only license wh- or d-pronouns as resuming elements (note that d-pronouns were only available in these structures if they were also available in structures with simple wh-phrases as extracted elements, as in (40) and (41)).

(50) *Welchen Mann glaubst du wen sie t liebt?*
Welchen Mann glaubst du den sie t liebt?
which man believe you who she loves
'Which man do you think she loves?'

(51) *Mit welchem Mann glaubst du mit wem sie Mit welchem Mann glaubst du mit dem sie t tanzt?*
with which man believe you with whom she dances
'With which man do you think she dances?'

Full NPs as resuming elements on the other hand were never judged grammatical by any speaker.

(52) * *Welchem Mann glaubst du dem Mann sie which man believe you the man she das Buch t gegeben hat?*
the book given has

'Which man do you think she gave the book to?'

(53) * *Mit welchem Mann glaubst du mit dem Mann sie t getanzt hat?*
with which man believe you with the man she danced has

'With which man do you think she dances?'

What all four results have in common is that they restrict the set of resuming elements. This leads to the question whether they can be subsumed under a single generalization; and in fact they can, as shown in (54).

(54) If x is licensed as a resuming element then x is also licensed as a free relative proform

Before I turn to the use of “proform” in this statement, let me briefly explain how this generalization covers all four results. The first result is covered because personal pronouns are not licensed as free relative pronouns.

(55) * *Ich lade ein ihn alle t mögen.*
I invite him everyone likes
'I invite who everyone likes.'

(56) * *Ich treffe mich mit ihm sie t getanzt hat.*
I meet with whom she danced has
'I met up with whom she danced.'

The second result follows from the generalization without further explication as it is nearly identical to it. The third result is subsumed because the elements appearing as resuming elements in (48) and (49) are not pronouns but full NPs. For the same reason, the fourth result is covered too: the resuming elements in (52) and (53) are full NPs, too, and not pronouns. Note that the generalization in (54) is silent on what categorial and morphological features the resuming element has to bear. However, this is no problem. For this is taken care of by the first generalization, according to which extracted and resuming element have to agree. Let me finally turn to the use of the term “proform” in (54). As the discussion in this section has shown, not only pronouns are licensed as resuming elements, but also PPs containing pronouns which are in themselves not pronouns, but rather “pro-PPs”. In order to capture this, I preferred using the word “proform” instead of “pronoun” in (54). The advantage of the term “proform” is that it doesn't imply a category for the element it refers to, which the term “pronoun” does, as it implies that the element is nominal.

4 A Problem with PS Analyses

As shown in the previous section, wh-copying is characterized by two generalizations that constrain the relation between extracted and resuming element. In this section, I would like to show that PS approaches cannot express the generalizations in a uniform way: NPs and PPs are equally subject to the generalizations but either of them requires a separate analysis. I will first sketch the analyses, and then discuss why having two analyses would be a problem at all.

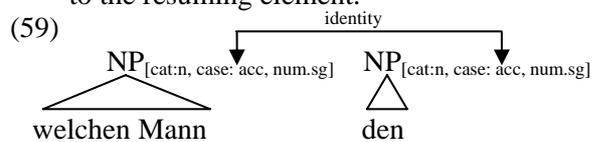
4.1 The PS Analyses

Consider the sentence in (57).

- (57) Welchen Mann glaubst du den sie t liebt?
 which man believe you who she loves
 ‘Which man do you think she loves?’

The extracted element in (57) is an accusative marked NP. That an accusative marked pronoun appears as a resuming element can be accounted for quite easily: all that is required is the operation given in (58), and illustrated in (59)⁵.

- (58) Establish identity for syntactic features between the node corresponding to the extracted element and the node corresponding to the resuming element.



The node corresponding to the extracted element is labeled ‘NP’ and specifies both category and morphological features, among others. The shape of the resuming element then follows because (58) requires the syntactic features of the extracted element to be identical to the syntactic features of the resuming element. As (58) requires identity only for syntactic but not for semantic features, it also follows that a pronoun will appear, as only they are semantically vacuous. (58) is attractive because it reduces the two generalizations to a single requirement, viz. one of agreement for syntactic features between two nodes. Unfortunately, (58) doesn’t work for PPs; consider the sentence in (60).

- (60) An welchen Mann meint er an den Jo denkt?
 on which man means he on whom Jo thinks
 ‘Which man does he believe Jo thinks of?’

⁵ (58) – and also (63) – is compatible with transformational (for example, GB) and non-transformational PS approaches (for example, HPSG); the difference is only whether the identity for syntactic features is analyzed as feature sharing or as a copying transformation. This difference is irrelevant, though, because either analysis is defined for PS trees.

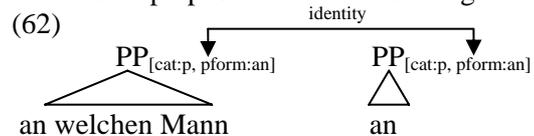
The extracted element *an welchen Mann* is taken up by the resuming element *an den*. If (58) were to hold for PPs, we expect the sentence in (60) to be ungrammatical; instead, the sentence in (61) should be grammatical, contrary to fact.

- (61) * An welchen Mann meint er an Jo denkt?

on which man means he on Jo thinks

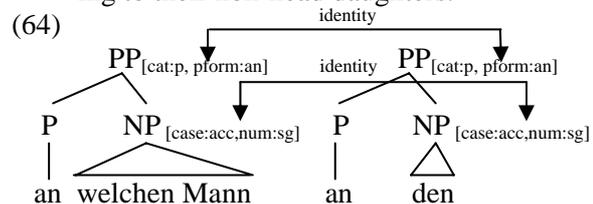
‘Which man does he believe Jo thinks of?’

The reason is that the PP node is specified only for features of its head but not for morphological features of its complement NP. If according to (58) identity between this node and the node for the resuming element is established, one ends up with a bare preposition as the resuming element.



To obtain the right result for PPs, one needs a separate statement requiring a dependency between pairs of nodes, as described in (63) and illustrated in (64).

- (63) Establish identity for category between the nodes corresponding to the extracted and the resuming element, and for morphological features between the nodes corresponding to their non-head daughters.



4.2 The Problem

Although both (58) and (63) give correct results, a problem arises. The problem is that by having one analysis for NPs and another one for PPs, one fails to express the uniform behavior of NPs and PPs in wh-copying. For each category requires a separate rule that incorporates the generalizations in a different way. In other words, the two generalizations cannot be uniformly expressed in a PS approach. What this means in the end though is that they are in fact lost in such an approach. No connection can be established between the two analyses because each analysis defines a requirement that is completely different from the requirement of the other analysis. Eventually, one also fails to express the fact that both analyses exist simultaneously in a language.

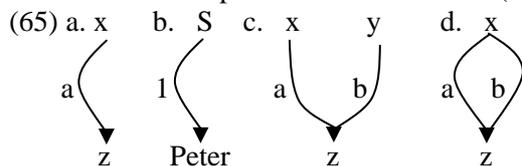
In sum, PS approaches cannot provide a tool for capturing in a descriptively adequate manner the two generalizations governing wh-copying.

5 A Relational Analysis

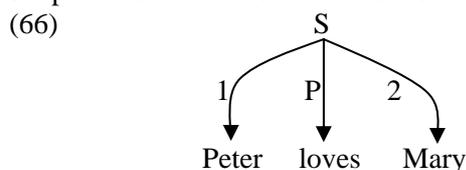
In this section, I will present an analysis of wh-copying within the framework of Arc Pair Grammar (Johnson & Postal 1980; Postal 2010), henceforth *APG*. I will show that due to its relational orientation, APG is not only capable of covering the two generalizations, it even allows unification of them into a single one. I will start by giving a brief overview of the characteristics of APG, then introduce APG's analyses of proforms, agreement, PPs, and extraction, and will then show how these assumptions provide the relevant tools for capturing the two generalizations of wh-copying in German.

5.1 Brief Overview of APG

APG is a descendent of Relational Grammar (cf. Perlmutter & Postal (1983) for an overview). APG differs from PS grammars in three ways. First, it assumes that grammatical relations – such as subject, object, indirect object – are primitive theoretical notions and that syntactic generalization need to be stated in terms of such relations. Formally, these relations are expressed via labeled, directed arcs. Second, APG allows what is called multidominance in a PS grammar, that is, a node can have more than one mother node. Both assumptions are illustrated in (65).

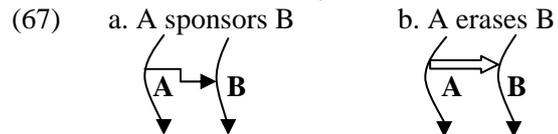


The representation in (65a) is called an *arc* and has to be read as: the node labeled 'z', the *head* node, bears the grammatical function 'a' to the node labeled 'x', the *tail* node. As an example, (65b) says that the node labeled 'Peter' bears the grammatical relation 'subject' – indicated as '1' – to the node labeled 'S', which is meant to indicate the sentence node. (65c) and (65d) give examples for multidominance, which is called *overlapping* in APG. Sentences are analyzed as graphs of a specific type⁶ that are 'composed' of arcs. As an example, consider the highly oversimplified structure for 'Peter loves Mary'.

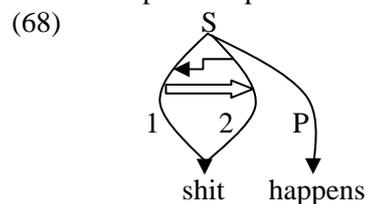


⁶ Cf. Johnson & Postal (1980), p. 51.

'P' indicates the predicate-of relation, '2' the direct-object relation. Third, APG also assumes primitive relations holding between arcs, in total two, viz. *Sponsor* and *Erase*. Saying that an arc A sponsors another arc B means that A is a necessary condition for the presence of B. And saying that an arc A erases another arc B means that the presence of A is a sufficient condition for nonpresence of B in surface structure. These relations, both of which are binary, are represented in the following way (bold capital letters are used to indicate arcs, not labels).



If an arc A bears such a relation to an arc B with which it shares the tail node, then the relation is called *local*, otherwise *foreign*. Sponsor and Erase are relevant for dealing with surface and non-surface aspects of sentence structure. In a nutshell, the set of non-sponsored arcs (called *initial arcs*) represents the initial structure of a sentence, and it – and only it – is therefore relevant for semantic interpretation; the set of non-erased arcs is irrelevant for semantic concerns and only represents the surface structure of a sentence. The sentence 'Shit happens' might serve as an example for Sponsor and Erase.



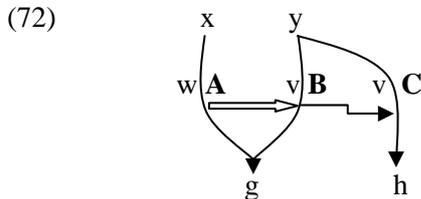
Happen belongs to the set of unaccusative predicates, which initially take direct objects that surface as subjects though. This property is represented through Sponsor and Erase in (68): the direct-object arc sponsors a subject arc which in turn erases the direct-object arc. As only the direct-object arc and the predicate arc are initial arcs, only they will be relevant for semantics. And since the subject and the predicate arc are the only non-erased arc, only they will surface.

5.2 Proforms

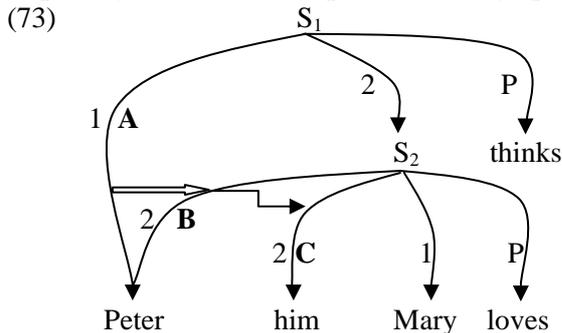
Proforms in APG are analyzed as elements heading non-initial arcs, that is, as arcs that are not relevant for semantics concerns. More specifically, they are analyzed as elements detaching, that is, *replacing* an initial, overlapping arc. The relevant definitions for *Replace* are given in (69)-(71).

- (69) An arc A (pronominally) detaches an arc B iff there is an arc C such that C replaces B, A seconds C, and A is not a branch of C
- (70) Replace/Second: an arc C replaces an arc B iff C and B are equivalent colimbs, B sponsors C, and there exists an arc A distinct from C that erases B. In this case, A is said to second C.
- (71) The Second Condition: If an arc C replaces an arc B and an arc A seconds C, then A overlaps B⁷.

‘Equivalent’ in definition (70) means that the label on the arcs C and B are identical; ‘colimbs’ mean that the two arcs C and B share the same tail node. Taken together, the definitions license a partial graph of the form given in (72); the letters for the arcs in (72) are held constant with respect to the ones in the definitions.



In order to understand the form of the graph, consider the definitions in (69)-(71). As required by (70), C and B bear the same label, viz. ‘v’, are colimbs (they share the same tail node, viz. ‘y’), B sponsors C, and a distinct arc A erases B. Accordingly, A seconds C. That A overlaps B follows from (71): since A seconds C, A is required to overlap B. Finally, (69) guarantees that this type of Replace will be one of pronominal detachment because A is not a branch of C. The idea behind this approach becomes clearer by inspecting a concrete example for such a graph⁸.



In this example, the element ‘Peter’ bears two grammatical relations: the subject relation within, and therefore to, S₁, and the direct-object relation to S₂. Due to the erasure of the direct-object B arc heading ‘Peter’, Replace inserts the equivalent arc C headed by the proform ‘him’. The

⁷ All definitions are taken from Postal (2010, ch. 1).

⁸ Linear order is generally *not* represented in the structures.

equivalence is taken care of by the requirement that the replacer arc has to have the same label as the replaced arc. Crucially, although Replace eventually constrains which elements can head a replacer arc, Replace substitutes arcs for arcs, not the elements heading them.

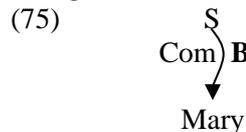
5.3 Agreement

Agreement between two elements is established via the *Lateral Feature Passing Law* in (74), adapted from Aissen (1990), p. 286.

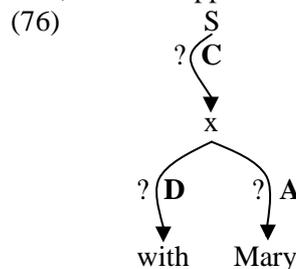
- (74) If a and b head nominal arcs, such that neither a nor b is a dependent of the other then, if a passes its morphological features to b, then the arc headed by b is equivalent to, and sponsored by, the arc headed by a
- ‘Dependent’ means that neither is ‘b’ the tail of the arc headed by ‘a’, nor is ‘a’ the tail of the arc headed by ‘b’. Applied to (73), ‘Peter’ corresponds to head ‘a’ and ‘him’ to head ‘b’, and both head nominal arcs. Transmission of the morphological features of ‘Peter’ to ‘him’ is licit because the arc headed by ‘him’ is equivalent to, and sponsored by, the arc headed by ‘Peter’.

5.4 Prepositional Phrases

APG adopts a relational view on sentence structure. Similar to proforms, categorial information such as being a PP represents only a surface aspect of sentence structure. In other words, prepositions are not analyzed as *bearing* a grammatical relation, but as elements *indicating* a grammatical relation, called *flags*. Consequently, the PP ‘with Mary’ is initially not a PP, but a nominal heading an arc that bears the label ‘Com’, indicating the comitative relation.



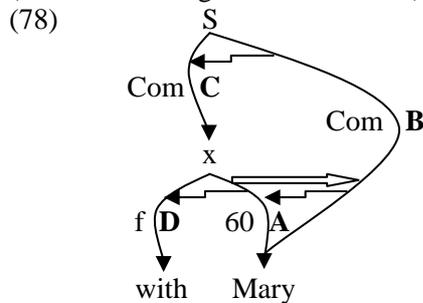
The question then arises is how to turn this initial structure into the structure appearing on the surface, which is approximately of the form in (76).



The answer given by APG is that the structures in (75) and (76) are connected via Sponsor and Erase. The relevant condition establishing this connection is the *flagging condition* in (77).

- (77) Iff an arc B is an output arc and not attached to a flag arc, then (i) B foreign sponsors a 60 arc A overlapping B, (ii) B is replaced by an arc C, (iii) A is a branch of C iff B's label is one of {Com, ...}, and (iv) A locally sponsors an arc D

This definition will license the following graph (the letters are again held constant).



In order to understand how the definition (77) licenses the graph in (78), one needs the definition for output arc.

- (79) An arc B is an output arc iff B is a domestic arc and has no local eraser.

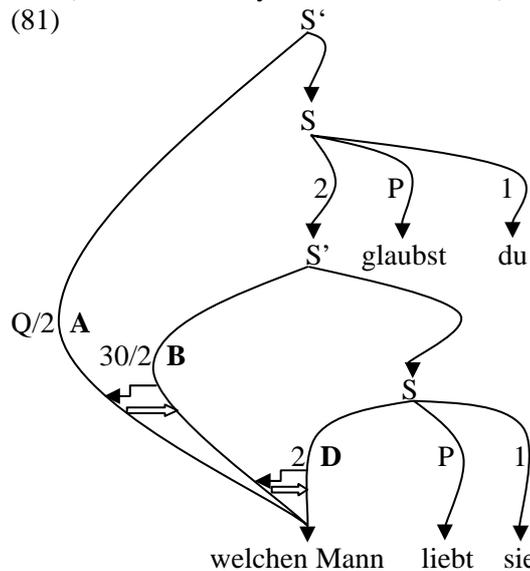
The definition for domestic arc is given in (80).

- (80) An arc B is a domestic arc iff any sponsor of B is a colimb of B.

In other words, an output arc is an arc that is (i) either unsponsored or locally sponsored and (ii) if erased, then not locally erased. Turning back to the graph in (78), let me explicate how (77) licenses it. First of all, B is an output arc: it is unsponsored and not locally erased. Second, B is not attached to flag arc: no arc bearing the 'f' relation is connected to either the tail or the head node of B. Therefore, B foreign sponsors the 60 arc A overlapping B. Then, B is replaced by the arc C such that the 60 arc A is branch of C; that is, A's head node is C's tail node. Finally, the 60 arc locally sponsors the flag arc D. Note that although C replaces B, this replace relation is not one of pronominal detachment because in this case, A is a branch of C; but pronominal detachment forbids A to be a branch of C. That B has to be flagged in the first place is due its label 'Com', which appears in the set specifying those relations that need to be labeled. Which relations this set contains is ultimately subject to language particular rules: whereas the comitative relation requires the prepositional flag 'with' in English, it doesn't in Hungarian (instead, the case suffix '-vel/-val' is added). Finally, that C itself is not subject to flagging also follows from (77). C is an output arc already attached to a flag arc; if it were attached to another flag arc, the condition in (77) would be violated, due to its formulation as a biconditional.

5.5 Extraction

The APG analysis of extraction has three ingredients. First, it is modeled via multidominance, which means that the extracted element will appear as the head of two overlapping arcs. One arc will indicate the initial relation of the element, for example direct-object. The other arc will indicate the relevant extraction, for example question-extraction, the label for which will be 'Q'. Second, extraction proceeds through positions that correspond neither to the initial nor to the final position of the extracted element. More specifically, I assume that extraction proceeds through every clause peripheral position between initial and final position of the extracted element. The arc that the element heads in this position will be labeled for convenience by '30'. Third, the labels of the relevant extraction arcs have to conserve the initial relation of the extracted element; this is expressed by simply adding the initial label to the label of both the 30- and the Q-arc⁹. This analysis gives the following structure for the sentence *Welchen Mann glaubst du liebt sie?* (Which man do you think she loves?).



Welchen Mann is the direct object of the embedded clause and the extracted element of the main clause. This is expressed by letting the Q/2-arc overlap the 2-arc. As the extraction targets a position outside the clause the 2-arc originates in, a 30/2-arc appearing in the clause peripheral position of the embedded clause is required. Finally, D sponsors B, B sponsors A, and A erases B, and B erases D.

⁹ Cf. Postal (2004), pp. 61-68, for a detailed discussion of the mechanism accomplishing this.

nally shows up as a PP must have an independent reason. This reason is the flagging condition: C is an output Com-arc not attached to a flag arc and must be replaced by a Com-arc attached to a flag arc. The identity for labels between C and B, which is due to Replace, guarantees that C will be attached to the same flag as B, viz. to *mit*, which gives agreement for category.

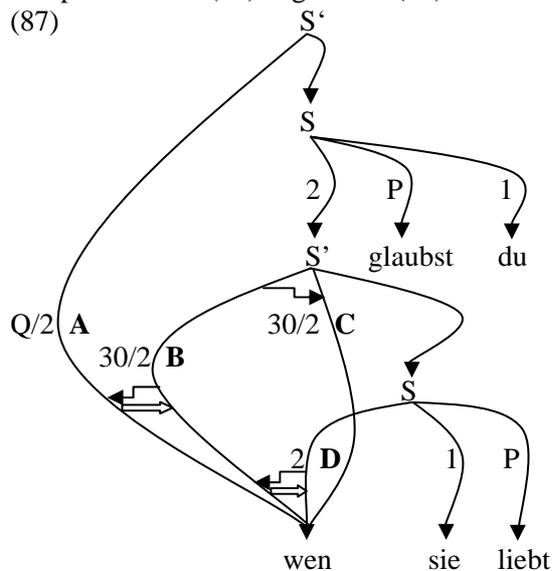
In sum, wh-copying always involves replacer arcs. As such, they can only be proform, must agree, and – depending on their label – sometimes require flagging, and sometimes not.

5.7 Outlook: Variation within German

Many speakers of German do not allow complex wh-phrases as extracted elements nor d-pronouns as resuming elements: only wh-proforms are licensed. The characteristic of d-pronouns is that they are not question words, whereas the characteristic of complex wh-phrases is that they are not proforms. This suggests that the constraint in (86) is at work for these speakers.

(86) The replacer arc overlaps the replaced arc

To satisfy this constraint, the replaced element has to be a proform because a replacer arc can only head a proform. This excludes complex wh-phrases as extracted elements, as the extracted element always overlaps the replaced arc. As the replaced arc can only head a proform that is available as a question word, only wh-proforms are licensed as replacers. It follows then that only wh-proforms can appear in general. A structure compatible with (86) is given in (87).



6 Conclusion

Due to its relational nature, APG allows one to give a uniform characterization of the resuming

element as a specific type of arc, viz. as a replacer arc. From this, the two generalizations governing the resuming element in wh-copying simply reduce to independently motivated constraints on well-formed arcs in general.

References

- Judith Aissen. 1990. Towards a Theory of Agreement Controllers. In: Paul M. Postal, Brian Joseph (eds.), *Studies in Relational Grammar 3*, (pp. 279-321). The University of Chicago Press, Chicago.
- Stefanie Anyadi, Armine Tamrazian. 1993. Wh-movement in Armenian and Ruhr German. *UCL Working Papers in Linguistics*, 5:1-22.
- Gisbert Fanselow. 2006. Partial Wh-Movement. In: Martin Everaert, Henk van Riemsdijk (eds.), *The Blackwell Companion to Syntax*, volume 3, (pp. 437-492). Blackwell Publishing.
- Claudia Felser. 2004. Wh-copying, phases, and successive cyclicity. *Lingua*, 114:543-574.
- Tilman Höhle. 2000. The W...W-Construction: Appositive or Scope Indicating? In: Uli Lutz, Gereon Müller, Arnim von Stechow (eds.), *Wh-Scope Marking*, (pp. 249-270). John Benjamins Publishing, Amsterdam.
- Pauline Jacobson. 1984. Connectivity in phrase structure grammar. *Natural Language and Linguistic Theory*, 1:535-581.
- David Johnson, Paul M. Postal. 1980. *Arc Pair Grammar*. Princeton University Press, Princeton, New Jersey.
- Melanie Klepp. 2002. Much ado about was: Why German directly depends on indirect dependency. In: Marjo van Koppen, Joanna Sio, Mark de Vos (eds.), *Proceedings of ConSOLE X* (pp. 111-125).
- Dana McDaniel. 1986. *Conditions on Wh-chains* (Unpublished doctoral dissertation). City University of New York. New York.
- Jairo Nunes. 2004. *Linearization of Chains and Sideward Movement*. The MIT Press, Cambridge, Massachusetts.
- David Perlmutter, Paul M. Postal. 1983. Some Proposed Laws of Basic Clause Structure. In: David Perlmutter (ed.), *Studies in Relational Grammar 1*, (pp. 81-129). The University of Chicago Press, Chicago.
- Paul M. Postal. 2004. *Skeptical Linguistic Essays*. Oxford University Press, Oxford, UK.
- Paul M. Postal. 2010. *Edge-Based Clausal Syntax*. The MIT Press, Cambridge, Massachusetts.

Why *kono akai hana* and *akai kono hana* Are Both Possible in Japanese: A Word Grammar Account

Kensei Sugayama
Kyoto Prefectural University
Sakyo-ku, Kyoto, Japan
ken@kpu.ac.jp

Abstract

It is an interesting fact that *kono akai hana* and *akai kono hana* are both possible in Japanese, while in English, only *this red flower*, the structure corresponding to the former, is possible. How do we explain this fact? To my knowledge, there has been no satisfactory answer so far to this old but not easy question in the literature on Japanese linguistics. In this paper I shall try to solve this problem within the framework of Word Grammar (henceforth abbreviated as WG; Hudson 1990, 2007b, 2010a, 2010b, 2010c).

The problem this article addresses is why one can say *this red flower* and *red this flower* in Japanese but not in English. The answer given is very simple: it depends on an analysis of the relevant English form as a DP with the *as the* head, while the relevant Japanese form is an NP with *flower* as the head. This, together with a precedence concord principle from Word Grammar, is supposed to account for the contrast between Japanese and English.

The central mystery to be explained is thus the free relative ordering of determiner and adjective in Japanese and the restricted order in English (*this+red+flower*). The explanation requires default dependent-to-governor linear ordering with a no-crossing constraint on "edges" plus the assumption that in English determiners are the heads/governors of the nouns that accompany them, but in Japanese determiners are dependent on the noun.

1. Introduction

It is a gripping fact that *kono akai hana* and *akai kono hana* are both possible in Japanese, while in English, only *this red flower*, the structure corresponding to the former, is possible. How we explain this fact is an intriguing and challenging problem. To my knowledge, there has been no satisfactory answer so far to this old but not easy question in the literature on Japanese linguistics. In this paper I shall try to

solve this problem within the framework of Word Grammar (henceforth abbreviated as WG, Hudson 1990, 2007b, 2010a, 2010b, 2010c).

An analysis is offered in terms of WG to explain these contrastive facts in Japanese and English. These facts are accounted for by the Precedence Concord Principle, the Promotion (Demotion) Principle and the Extra Dependency.

I will begin with a discussion of the key concepts in Word Grammar for this study of the internal structure of noun phrases containing Japanese and English determiners.¹ Next, I will present data from Japanese and English. Then, I will demonstrate how Japanese and English data are handled quite neatly in WG using the Precedence Concord Principle.

2. Word Grammar in a Nutshell

Word Grammar is a theory of language structure which Richard Hudson of University College London has been building since the early 1980's. It is still changing in detail, but the main ideas are the same. These ideas themselves developed out of two other theories that he had proposed: Systemic Grammar (now known as Systemic Functional Grammar), due to Michael Halliday, and then Daughter-Dependency Grammar, his own invention.

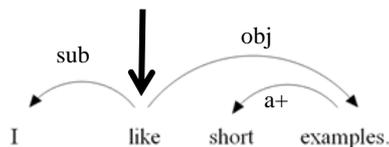
Here are the main ideas, most of which come from the latest version of the WG encyclopedia and WG Homepage (2010b), together with an indication of where they originally came from.

- It (i.e. WG) is monostratal - only one structure per sentence, no transformations. (From Systemic Grammar)

¹ As will be discussed later in the paper, there is syntactic evidence against categorizing these Japanese demonstratives as determiners.

- It uses word-word dependencies - e.g. a noun is the subject of a verb. (From John Anderson and other users of Dependency Grammar, via Daughter Dependency Grammar; a reaction against Systemic Grammar where word-word dependencies are mediated by the features of the mother phrase.)
- It does not use phrase structure - e.g. it does not recognise a noun phrase as the subject of a clause, though these phrases are implicit in the dependency structure. (This is the main difference between Daughter Dependency Grammar and Word Grammar.)
- It shows grammatical relations/functions by explicit labels - e.g. 'subject' and 'object' as shown in (1). (From Systemic Grammar)

(1)



- It uses features only for inflectional contrasts - e.g. tense, number but not transitivity. (A reaction against excessive use of features in both Systemic Grammar and current Transformational Grammar.)
- It uses default inheritance, as a very general way of capturing the contrast between 'basic' or 'underlying' patterns and 'exceptions' or 'transformations' - e.g. by default, English words follow the word they depend on, but exceptionally subjects precede it; particular cases 'inherit' the default pattern unless it is explicitly overridden by a contradictory rule. (From Artificial Intelligence)
- It views concepts as prototypes rather than 'classical' categories that can be defined by necessary and sufficient conditions. All characteristics (i.e. all links in the network) have equal status, though some may for pragmatic reasons be harder to override than others. (From Lakoff and early Cognitive Linguistics, supported by work in sociolinguistics)
- It presents language as a network of knowledge, linking concepts about words, their meanings, etc. - e.g. *kerb* is linked to the meaning 'kerb', to the form /ke:b/, to the word-class 'noun', etc. (From Lamb's

Stratificational Grammar, now known as Neurocognitive Linguistics)

- In this network there are no clear boundaries between different areas of knowledge - e.g. between 'lexicon' and 'grammar', or between 'linguistic meaning' and 'encyclopedic knowledge'. (From early Cognitive Linguistics)
- In particular, there is no clear boundary between 'internal' and 'external' facts about words, so a grammar should be able to incorporate sociolinguistic facts - e.g. the speaker of *jazzed* is an American. (From sociolinguistics)

In this theory, word-word dependency is a key concept, upon which the syntax and semantics of a sentence build. Dependents of a word are subcategorised into two types, i.e. complements and adjuncts. These two types of dependents play a significant role in this theory of grammar.

Let me give you a flavour of the syntax and semantics in WG, as shown in Figure 1.²

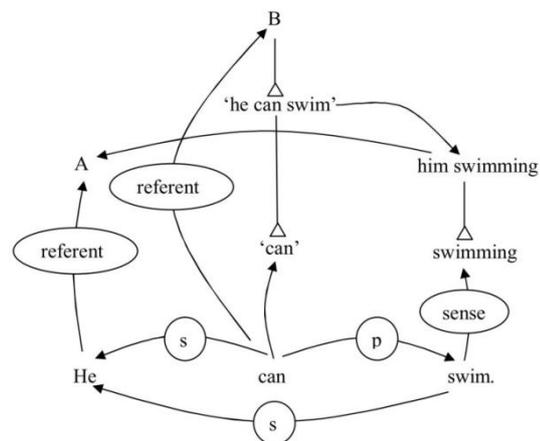


Figure 1: Syntax and Semantics in WG

3. The Data from Japanese and the Analysis

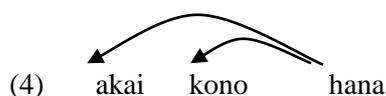
What is the structure of *akai kono hana*? Consider the following data first:

(2) *akai kono hana*

² A letter above or below the dependency arrow represents a grammatical function: 's' stands for subject, 'o' for object, 'c' for complement, 'a<' for pre-adjunct. '>a' for post-adjunct, etc. The vertical arrow shows the root (head, parent) of the sentence.

red this flower
 (3) kono akai hana

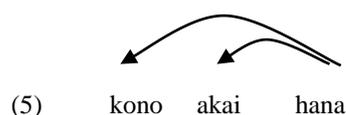
If we take the WG view that in a word combination like *akai hana*, the word which provides the basic meaning is the head (also called as parent in the paper) of the combination, and the other is its dependent (i.e. depends on it), then *hana* is the head of the whole structure *akai hana*. In general, the word which is expanded is the head and the words which expand it are said to depend on it. The structure for *akai kono hana* is shown in (4), where the arrow points from the head to its dependent.



The head has a certain number of characteristics; in a pair of words related by dependency, the head is the one

- from which the other word takes its position
- which controls the inflected form (if any) of the other
- which controls the possibility of occurrence of the other
- which denotes the more general case of which Head + Dependent is an instance (i.e. Head + Dependent is a kind of Head, not vice versa). (Hudson 1984)

Because there is no tangling in dependency relation in (4) and the No-tangling Principle requires that there should be no tangling in dependency lines when the order of two dependents is reversed, it is predicted that *kono akai hana* is also grammatical, which is in fact the case as in (5).³



³ The No-tangling Principle states that in surface structure no dependencies should tangle with each other - i.e. dependency lines must not cross except when required to do so for coordination or clitics.

The same dependency pattern is also found in similar structures in (6) - (9), where a different determiner *ano* is used with the common noun as the head in the NP.

- (6) ano subarasii asa
 that beautiful morning
 (7) subarasii ano asa
 (8) ano ameno hi
 that rainy day
 (9) ameno ano hi

Much the same is true of a relative clause modifier used with a determiner as shown in (10) and (11).

- (10) ano karega katta hon
 that he-Sub bought book
 (11) karega katta ano hon

Thus, these data imply that nouns like *hana* etc are actually heads and the determiners, like other adjectives, are dependents in these nominal structures in Japanese.⁴

A further piece of evidence that the noun is a head in the so-called NPs in Japanese comes from the behaviour of two determiners and one adjective in the structure at issue. Consider further data in (12) - (17). Phrases like the following are all possible in Japanese.

- (12) John-no kono akai hon
 John-Gen this red book
 (13) John-no akai kono hon
 (14) kono John-noakai hon
 (15) kono akai John-nohon
 (16) akai John-nokono hon
 (17) akai kono John-nohon

Since all the permutations in order of the words *John-no*, *kono* and *akai* are allowed before the common noun and since *akai* is obviously an adjective and therefore a dependent of *hon*, we have to assume that the same is true of *John-no* and *kono* given the No-tangling Principle, although they both translate as determiners in English.

⁴ I notice that it is a matter for discussion whether or not Japanese demonstratives such as *kono*, *sono*, and *ano* are proper determiners in the X-bar tradition. For the sake of simplicity, let us suppose throughout the paper that these demonstratives can be called as determiners.

Morphologically, demonstratives such as *kono*, *sono*, and *ano* seem to be composed of *ko*, *so*, *a*, and a dependent particle *-no*, which explains why *kono*, etc. behave exactly the same as other adjectives, since *-no* changes the base pronoun into a kind of adjective, which attach quite freely with nouns as long as the derived structures are semantically interpreted.

As to the internal structure of *kono*, *kono* is considered to have the structure in which *ko*, deictic pronoun referring to a place 'more or less near the speaker, is attached to the genitive-case marker *-no*.

4. The Data from English and the Analysis

In contrast, English involves a situation where only the structure *this red flower* corresponding to (3) is possible and **red this flower* corresponding to (2) is not allowed. Before getting into a detailed analysis of why this structure is ungrammatical in English, let us consider engrossing facts about the word order in English and try to find a way in which to explain the facts.

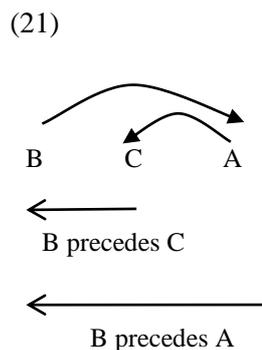
- (18) I teach bilingual students.
- (19) *I bilingual teach students. (* because the arc *bilingual* ← *students* crosses the arc *I* ← *teach*)

This fact is accounted for by the Precedence Concord Principle (PCP), as formulated in (20).

(20) Precedence Concord Principle :

A word must have the same precedence relation as its head to the latter's head. (Hudson 2010b)

Precedence concord is the very general principle that makes phrases hang together -- i.e. that in general bans discontinuous phrases. It assumes that words have 'precedence relations' to one another, showing which comes before which. Precedence concord means that two words have to have the same precedence relationship to some other word: so if A follows B, and C has precedence concord with A, then C must also follow B. The diagram in (21) represents the basic rationale of the Precedence Concord Principle (now known as 'Order Concord' in Hudson (2010b)).

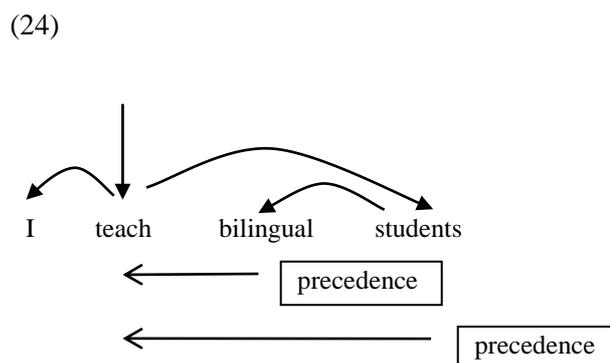


Thus, in (21) the word C has the same precedence concord as the word A to the word B. Put alternatively in terms of dependency, the principle states that if C depends on A and A on B, then C must have the same precedence (before/after) relations as A to B, as in (21).

To see how well it works to give an elegant elucidation for discontinuous phrases in English, let us come back to earlier examples (18) and (19), repeated as (22) and (23) below.

- (22) I teach bilingual students.
- (23) *I bilingual teach students. (* because the arc *bilingual* ← *students* crosses the arc *I* ← *teach*)

The diagram in (24) corresponds to the structure of (22). The precedence is displayed by arrows below the structure in (24) where *bilingual* has the same precedence as *students* to *teach*.

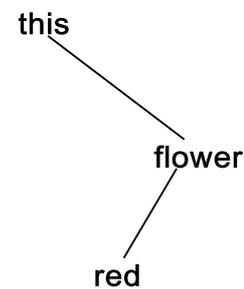
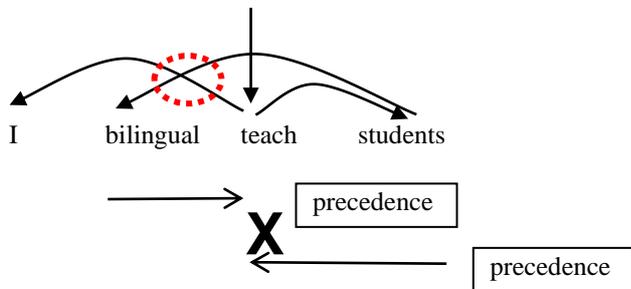


As shown in (24), if *students* in *I teach bilingual students* depends on *teach*, and follows *teach*, then any other word which depends on *students* must also follow *teach*. This is exactly the reason why *bilingual* has to come between *teach* and *students* in (24).

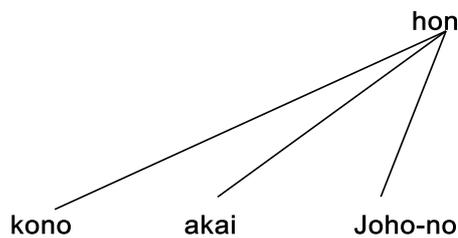
Let us consider then the case with (23), the structure of which is shown in (25). In (25) tangle of dependency line is represented by a

circle in broken line. The principle rules out **I bilingual teach students*, on the grounds that *bilingual* depends on *students* but does not share its precedence relation to *teach*, as in (25), where clash in precedence is indicated by two arrows pointing at each other below the diagram. (27)

(25)



What the PCP predicts will be much more transparent in the Tesnière-style stemma. The structure below represents *kono akai John-no hon* in stemma. (26)



The PCP predicts that the order of three words headed by *hon* will be free as long as they appear before their governing head *hon*.

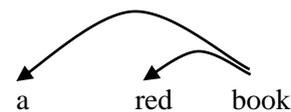
On the other hand, an order restriction on the three words in *this red flower* in English will be easily explained by the dependency relation between the words and their linear order (precedence in order). The principle predicts that *red* has to appear between *this* and *flower* because *flower*, dependent of *this*, appears after *this*, and *red* is a dependent of *flower*. See the structure in (27).

5. Why **red a book* Is not Possible

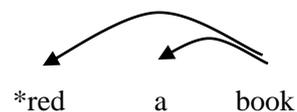
In this section, I will show within the framework of WG why we have to take a determiner as head in the NP in English. My claim is that the Precedence Concord Principle blocks the word order adjective > determiner > noun such as **red this flower* if determiners is defined as head in English.⁵

If we take a noun as head of the phrase, then what the No-tangling Principle will predict based on the dependency structure in (28) is that (29) should be grammatical, which on the contrary is not the case.

(28)



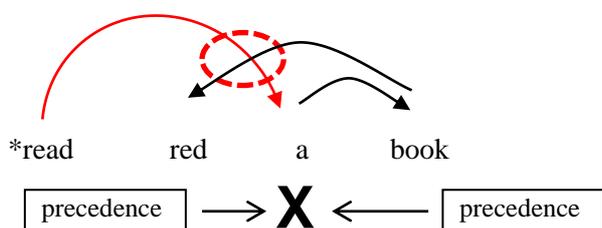
(29)



In contrast, to take a determiner rather than a noun as head, together with the Precedence Concord Principle will produce the correct result, which is attested by (30).

⁵ ' $\beta > \alpha$ ' indicates that β precedes α .

(30)



Although *red* is a dependent of *book* in (30), it does not have the same precedence as *book* to *a*, which is automatically predicted to be ungrammatical by the Precedence Concord Principle.

Additionally, there is also semantic evidence supporting that determiners are actually heads in English. They must be heads because *this flower* is an expansion of *this*, which can always occur without *flower*; it cannot be an expansion of *flower*, because this is impossible without *this*. Therefore *flower* must depend on *this*, not the other way round. In Japanese, on the other hand, *kore* (corresponding to *this*) cannot be expanded, but *hana* can, which implies that *hana* is a head in (2) or (3).

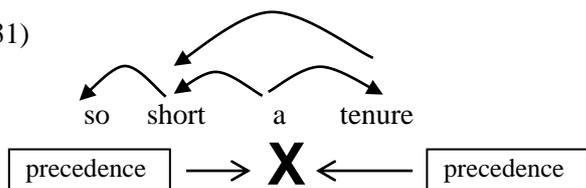
To sum up, fitting the required order of determiner > adjective > noun in English NPs into the grammar of English necessarily involves a rule which takes a determiner as head. This view is shared with the so-called DP analysis (Abney 1987).

6. Apparent Counter-examples

As Murata (2003) points out, however, there are structures which the Precedence Concord Principle seems to predict to be ungrammatical, yet do exist in English. Below are some such apparent counter-examples.

In (31) there is obviously violation of the Precedence Concord Principle with the word *short* (and also violation of the No-tangling Principle), which has to come to the right of *a*.

(31)



In English noun phrases the determiner canonically precedes the pronominal adjectives, both the lexical and the phrasal ones.

(32) a. a big house

b. a very big house

(33) a. *big a house

b. *very big a house

A notable exception are the adjectival phrases which are introduced by *as*, *so*, *too*, *how*, *this* and *that*. When they occur in a nominal which contains the indefinite article, they precede the determiner (Huddleston and Pullum 2002, 435).

(34) a. It's so good a bargain I can't resist buying it. (*ibid.*)

b. How serious a problem is it? (*ibid.*)

c. *They're so good bargains I can't resist buying them. (van Eynde 2007)

d. *How serious problems are they? (*ibid.*)

(35) a. *It's a so good bargain I can't resist buying it. (*ibid.*)

b. *A how serious problem is it? (*ibid.*)

This construction, for which Berman (1974) coined the term Big Mess Construction, only occurs in nominals with an indefinite article. It does not occur in nominal with another kind of determiner, as in (36a), nor in nominals without determiner, as in (36b).

(36) a. *How serious some problem is it? (van Eynde 2007)

b. *They are so good bargains I can't resist buying them. (*ibid.*)

c. How serious a problem is it? (Huddleston and Pullum 2002, 435)

d. *How serious problems are they? (*ibid.*)

A further complication is provided by the APs which are introduced by *more* or *less*. They can either occur in the canonical position or in the exceptional one (Huddleston and Pullum 2002, 435).

(37) a. This is a more serious problem than the other. (*ibid.*)

b. This is more serious a problem than the other. (*ibid.*)

Also here, the exceptional position is only possible in combination with the indefinite article.

What makes the Big Mess Construction interesting is not only its idiosyncrasy and the descriptive challenges which it raises, but also the light which its treatment sheds on the issue of the trade-off between lexicalism and constructivism in formal grammar.

To pave the way for the treatment I first present my analysis of the internal structure of the relevant BMCs. It deals with the canonical order autonomously. The exceptional order, as exemplified by (38) to and (40), is modeled later in this section.

This alleged violation of the Precedence Concord Principle seen in the BMC, however, can be saved by introducing the idea of dividing the sentence's dependencies into two, i.e. the 'surface' dependencies and 'other' (*alias* 'extra') dependencies.

In general, the dependency structures in the surface structure are drawn above the words of the sentence - i.e. literally on the sentence's 'surface'. Other dependencies (called 'extra dependencies') are drawn below the sentence-words. This idea is a fairly recent addition to WG theory (which used to rely on the Adjacency Principle - Hudson 1984, 113-120). The basic idea is that not all dependencies are relevant to word order (i.e. visible to the Precedence Concord Principle), so we pick out a sub-set which are visible to the principle and show them separately from the others. This sub-set is the 'surface structure'. The diagram in (38) on the right column shows the surface structure above the words and the rest of the dependency structure below them.

The surface structure is a domain for the No-dangling Principle, which requires every word to have one surface parent. It also used to be a domain for the No-tangling Principle, but this has now been replaced by order concord as the means for keeping the words in a phrase together. In current theory, the surface structure contains all the dependencies that define landmarks. The term 'landmark' is borrowed from Langacker's Cognitive Grammar, where it is used for talking about meaning.⁶ This

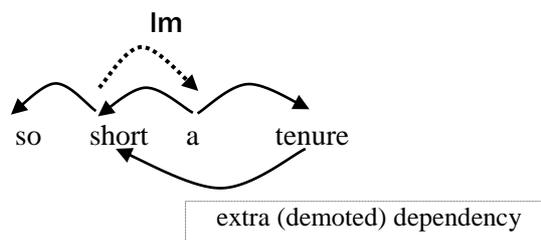
⁶ For example, a spatial preposition defines a place by its relation to the 'landmark'; so in the phrase *in Rylestone*, Rylestone is the landmark and the place is somewhere within the City of Rylestone, and in *at the door* the door is the landmark. The landmark is the fixed identifiable reference-point for the relationship.

term is very useful in semantics, but it can be extended to syntax as well (Hudson 2010b). The idea is that almost every word is positioned in relation to some other word in syntax.

Here let us assume that where the word has more than one parents, only one is relevant - for example, in raising, only the higher parent is relevant; this is the basic idea behind the earlier notion of surface structure.

A word's landmark is typically one of its parents; but which one should count as the landmark? In (31) above *short* has two parents (i.e. *a* and *tenure*). In most cases the choice is forced by the special restriction called the Promotion Principle which favours the higher of two parents with the effect of 'promoting' the word to the highest possible rank in the sentence. In practice, though, the landmark arrows are almost redundant once you have marked all the dependencies because most words have only one parent, and that is their landmark. The only cases where words whose landmarks are worth distinguishing from parents are those which have more than one parents; and we already have a notation for making the relevant distinctions. This is the surface-structure notation, which demotes the dependencies that do not provide landmarks by drawing their arrows below the words. This allows the very much simpler notation below as in (38) where the extra dependency from *tenure* to *short* is now demoted as extra to become invisible to the Precedence Concord Principle, providing a favourable solution to the data, which would otherwise be predicted to be ungrammatical.

(38)



- If a word has more than one landmark, only the nearest landmark is visible to dependency analysis. The dependencies that do not provide landmarks are demoted. (cf. Hudson 2010b)

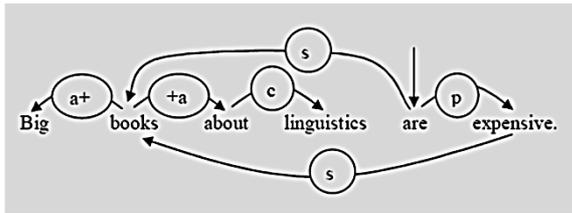
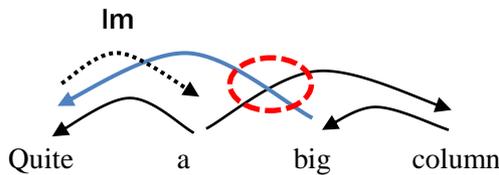


Figure 2: Landmarks shadow dependencies

This idea extends further to another case of seeming counter-examples as in (39). In (39) *quite* has two parents (i.e. *a* and *big*) at the surface, and a dependency relation from *big* to *quite* crosses the one between *a* and *column*, violating the No-tangling Principle.⁷

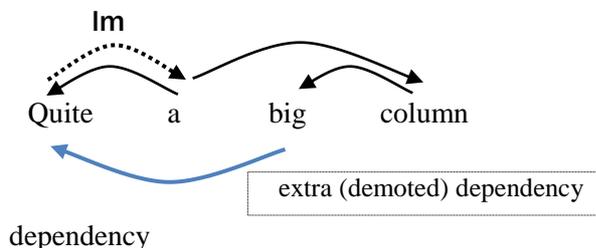
(39)



In (39), as *big* is a remoter parent to *quite*, which demotes the dependency from *big* to *quite* to create a new structure in (40).

(40)

So far, we have seen that alleged counter-examples can in fact be predicted to be grammatical in WG by taking only the surface dependency into account and excluding the extra



⁷ 'Im' represents the relation 'landmark'.

7. Conclusion

In summary, I have shown why *kono* and *akai* are reversible in the structure *kono akai hana* in Japanese, while English allows only one corresponding structure. My arguments are based on the difference in grammatical category of *kono* and *this* in each language. From the dependency analysis above, the conclusion, then, is that the determiner is the head (parent) of a NP, and that the common noun is a complement in English NPs.

Acknowledgments

I am grateful to the four anonymous referees for the invaluable comments and suggestions.

References

- Steven P. Abney. 1987. *The English Noun Phrase in Its Sentential Aspect*. PhD thesis, MIT.
- Arlene Berman. 1974. *Adjective and Adjective Complement Constructions in English*. PhD thesis, Harvard University.
- R. Huddleston and G.K. Pullum. 2002. *The Cambridge Grammar of the English Language*. Cambridge University Press, Cambridge.
- Richard A. Hudson. 1984. *Word Grammar*. Blackwell, Oxford.
- Richard A. Hudson. 1990. *English Word Grammar*. Blackwell, Oxford.
<http://www.phon.ucl.ac.uk/home/dick/wg.htm>.
- Richard A. Hudson. 2003. The Psychological Reality of Syntactic Dependency Relations. Paper read at MTT 2003, Paris, 16-18 juin 2003.
- Richard A. Hudson. 2004. Are Determiners Heads? *Functions of Language*, 11(1): 7 - 42.
- Richard A. Hudson. A. 2007a. Word Grammar. In Dirk Geeraerts and Hubert Cuyckens. (eds.). *The Oxford Handbook of Cognitive Linguistics*, 509-542. Oxford University Press, Oxford.
- Richard A. Hudson. 2007b. *Language Networks: The New Word Grammar*. Oxford University Press, Oxford.
- Richard A. Hudson. 2010a. *An Introduction to Word Grammar*. Cambridge University Press, Cambridge.
- Richard A. Hudson. 2010b. An Encyclopaedia of Word Grammar and English Grammar. (Version: 17 June 2010).
<http://www.phon.ucl.ac.uk/home/dick/wg.htm>.
- Richard A. Hudson. 2010c. Word Grammar Homepage. (Version: 29 July 2010).
<http://www.phon.ucl.ac.uk/home/dick/wg.htm>.
- Kim Jong-Bok and Peter Sells. 2011. The Big Mess Construction: Interactions between the Lexicon and Constructions. *English Language and Linguistics*, 15(2): 335-362.

- P. Kay and Ivan A. Sag. 2008. Not as Hard a Problem to Solve as You Might Have Thought. Handout for a paper read at Construction Grammar Conference. University of Texas, Austin.
- Ronald W. Langacker. 1991a. *Foundations of Cognitive Grammar*. Vol. 1. Stanford Univ. Press, Stanford.
- Ronald W. Langacker. 1991b. *Foundations of Cognitive Grammar*. Vol. 2. Stanford Univ. Press, Stanford.
- Jun'ichi Murata. 2003. Where Are They Headed. In Kensei Sugayama. (ed.). *Studies in Word Grammar*, 135-142. Research Institute of Foreign Studies, Kobe City University of Foreign Studies, Kobe.
- Carl Pollard and Ivan A. Sag. 1987. *Information-Based Syntax and Semantics*. CSLI, Stanford.
- Paul Schachter. 1978. Review of *Arguments for a Non-Transformational Grammar*. *Language*, 54(2): 348-376.
- Kensei Sugayama. 2003a. Introduction. In Kensei Sugayama. (ed.). 2003. *Studies in Word Grammar*, 3-6. Research Institute of Foreign Studies, Kobe City University of Foreign Studies, Kobe.
- Kensei Sugayama. 2003b. The Grammar of *Be To*: From a Word Grammar Point of View. In Kensei Sugayama. (ed.). 2003. *Studies in Word Grammar*, 97-111. Research Institute of Foreign Studies, Kobe City University of Foreign Studies, Kobe.
- Kensei Sugayama. (ed.). 2003. *Studies in Word Grammar*. Research Institute for Foreign Studies, Kobe City University of Foreign Studies, Kobe.
- Kensei Sugayama. 2007. A Word-Grammatic Account of *Eat* and Its Japanese Equivalent *Taberu*. In Viktoria Eschbach-Szabo, André Włodarczyk, and Yoshi-hiko Ikegami (eds.). *Japanese Linguistics European Chapter*, 285-300. Kurosio Pub, Tokyo.
- Kensei Sugayama and Richard A. Hudson. 2005. *Word Grammar: New Perspectives on a Theory of Language Structure*. Continuum, London.
- Lucien Tesnière. 1976. *Éléments de Syntaxe Structurale*. Klincksieck, Paris.
- Natsuko Tsujimura. 1996, 2007². *An Introduction to Japanese Linguistics*. Blackwell, Oxford.
- Frank van Eynde. 2007. The Big Mess Construction. In Stefan Müller. (ed.). *The Proceedings of the 14th International Conference on Head-Driven Phrase Structure Grammar*, 415-433. CSLI Publications, Stanford.

Sentence Structure and Discourse Structure: Possible Parallels

Pavína Jínová, Lucie Mladová, Jiří Mirovský

Charles University in Prague, Faculty of Mathematics and Physics

Institute of Formal and Applied Linguistics (UFAL MFF CUNI)

Malostranské nám. 25, 118 00 Prague 1, Czech Republic

{jinova,mladova,mirovsky}@ufal.mff.cuni.cz

Abstract

The present contribution represents the first step in comparing the nature of syntactico-semantic relations present in the sentence structure to their equivalents in the discourse structure. The study is carried out on the basis of a Czech manually annotated material collected in the Prague Dependency Treebank (PDT). According to the semantic analysis of the underlying syntactic structure of a sentence (tectogrammatrics) in the PDT, we distinguish various types of relations that can be expressed both within a single sentence (i.e. in a tree) and in a larger text, beyond the sentence boundary (between trees). We suggest that, on the one hand, each type of these relations preserves its semantic nature both within a sentence and in a larger text (i.e. a causal relation remains a causal relation) but, on the other hand, according to the semantic properties of the relations, their distribution in a sentence or between sentences is very diverse. In this study, this observation is analyzed for two cases (relations of condition and specification) and further supported by similar behaviour of the English data from the Penn Discourse Treebank.

1 Motivation and Background

Although the annotation in the Prague Dependency Treebank 2.0 (PDT, Hajič et al., 2006; Mikulová et al., 2005) in principle does not surpass the sentence boundaries, i.e. each sentence is represented by a single dependency tree structure, to a certain extent, the information about the context has always been one of its concerns. First, the context of every sentence is reflected in one attribute of the nodes in the syntactico-semantic (tectogrammatric) structure: the information structure of the sentence (Topic-Focus Articulation, TFA, cf. Sgall, Hajičová and Panevová, 1986; Hajičová,

Partee and Sgall, 1998), second, some basic coreference relations are marked (especially the grammatical coreference and some types of the textual coreference). In recent years, the interest in analyzing the structure of discourse in a more complex way has increased, and also the PDT is being enriched with this type of information. After having annotated the anaphoric chains and also the so-called bridging relations (or the association anaphora, see Nedoluzhko et al., 2009), the annotation of semantic relations between text spans indicated by certain discourse markers is now in progress. This annotation has two linguistic resources: besides the Prague (syntactico-semantic) approach it is inspired also by the Penn Discourse Treebank 2.0 approach based on identifying discourse connectives and their arguments (Prasad et al., 2007 and 2008).

One of the benefits of annotating discourse semantic relations on tectogrammatric trees is a possibility to exploit the syntactico-semantic information already captured in the corpus. This fact also enables us to compare the nature of relations expressed both within a single sentence (in a single tree) and in a larger text (between trees). Since the discourse annotation of the PDT is still a work in progress, it is premature to make some final conclusions in this respect. On the other hand, a majority of the corpus has already been processed and some tendencies are evident. In the present contribution we therefore want to introduce some observations about the nature of these corresponding relations and support them with our data.

The contribution is divided into three main parts. In Section 2, we describe some basic aspects of the Praguian approach to the syntactic structure (tectogrammatrics); criteria according to which some relations from the tectogrammatrics are considered to be discourse relations are introduced in Section 3; and in Section 4 a comparison of intra-sentential and inter-sen-

tential (discourse) relations is carried out on an example of two semantic relations from our manually annotated data.

2 Basic Aspects of the Underlying Syntactic Structure in the PDT Relevant for Discourse

There are three basic aspects of the syntactic structure already captured on the tectogrammatical layer in the PDT (see also Mladová et al., 2008) that are relevant for the discourse structure analysis: (i) the dependency edge between nodes filled with finite verbs (i.e. the relation between a subordinate clause and its governing clause), (ii) the coordination connecting finite-verbal nodes (i.e. the relation between coordinate clauses), and (iii) nodes with the label “reference to PREceding Context” (PREC) (i.e. the label for such expressions as *however*, *hence* and *so forth*). The subordinate and coordinate structures are classified according to their syntactico-semantic values and some of these values can be directly transferred to the discourse annotation (e.g. the semantic label of a subordinate clause “cause” corresponds in the vast majority of cases with its discourse counterpart). However, in other cases, the set of semantic values of the edges is not subcategorized enough for the discourse annotation and it needed to be classified in a more detailed way (e.g. the only semantic label for adversative meaning on the tectogrammatical layer was for the purpose of the discourse annotation divided into finer types of contrast, such as opposition, restrictive opposition and correction, cf. Zikánová, 2007). Moreover, one special type of relation – apposition – and the meaning indicated by expressions with the label PREC were not semantically interpreted at all on the tectogrammatical layer. The notion of apposition is descriptive, it stands for a syntactic structure with one syntactic position filled by two formally independent nominal or verbal phrases that are referentially at least partially identical (e.g. *he has only one obsession: he spends at least two hours a day playing computer games*). It follows that the notion of apposition is semantically too abstract for the purposes of the discourse annotation and so it was also subcategorized and re-labeled (see Figure 1 below in Section 4.2).

3 Discourse Annotation

3.1 Discourse Relevance of Intra-sentential Relations

From our point of view, there is a necessary condition for each syntactico-semantic relation (taken from the tectogrammatical analysis, Mikulová et al., 2005) to be considered a discourse relation: its possibility to relate two syntactically independent sentences. In other words, it must be possible in a natural language to relate two independent text spans with semantically exactly the same meaning, as there is on the syntactic level (often more plausibly) between the governing verbal node and its complement, dependent node; or, in a compound sentence, between the coordinate (verbal) clauses¹.

Another, milder requirement concerns the connective means of each relation. Whereas the transparency of the sentence semantics depends on the presence of subordinating connectives, which anchor the meaning (Mladová, 2009), we prefer to treat a syntactico-semantic relation as discourse-applicable, if we can find a corresponding means to the subordinating expression on the discourse level. In some cases, this is quite easy, such as in (1)²: in (1a), the discourse-semantic relation occurs between a subordinate clause and its governing clause, whereas in (1b) it relates two independent sentences.

(1)

(a) [*Arg1: **Protože** slovenská elita byla zklamána politickou volbou Slovenska,*]
[*Arg2: většina kvalitních odborníků zůstala v Praze.*]

[*Arg1: **Because** Slovak political elite was disappointed by the political choice of Slovakia,*]

[*Arg2: the majority of skilled professionals remained in Prague.*]

(b) [*Arg1: Slovenská elita byla zklamána politickou volbou Slovenska.*]

[*Arg2: **Proto** většina kvalitních odborníků zůstala v Praze.*]

¹ For the first phase of the discourse annotation, only clauses headed by a finite verb were taken to be discourse-level units. Nominalizations and other clause-like phenomena are to be explored for their discourse functions in the next phases of the project.

² Abbreviations Arg1 and Arg2 are used in examples for indication of the two text spans between which the discourse semantic relation occurs. Connectives are in bold.

[Arg1: Slovak political elite was disappointed by the political choice of Slovakia.]

[Arg2: **Therefore**, the majority of skilled professionals remained in Prague.]

As for coordinated clauses, the situation is very simple. Coordinated clauses in a compound sentence always play the role of discourse arguments and their conjunction is a discourse-level connective.³ This applies not only for structures connected by connectives such as *therefore*, *but*, or *etc.* but also when the coordinating connective is represented by a “mere” punctuation mark like a dash (see (2)) or a colon (see (3)). According to their semantics, these structures can be reformulated as two independent sentences (two trees) either by adjacency without any connective (the case of (2)) or by independent sentences linked with an explicit connective. In the case of (3), the connective *totiž* (in this context without any appropriate English equivalent, perhaps it can be roughly translated as “that is to say” or “as a matter of fact”, depending on the context) can be used in the part after the colon.

Example (2) demonstrates a discourse semantic relation expressed (a) by a coordinative structure with a dash and (b) by two independent sentences.

(2)

(a) [Arg1: Sparta přenechává volné pole konkurenci]

[Arg2: – Látal odešel do Schalke 04, Hogen se Šmejkalem jsou ve Slavii, Poborský září na Žižkově.]

[Arg1: FC Sparta leaves the field open to competition]

[Arg2: – Látal left to Schalke 04, Hogen and Šmejkal are in Slavia, Poborský shines in FC Žižkov.]

(b) [Arg1: Sparta přenechává volné pole konkurenci.]

[Arg2: Látal odešel do Schalke 04, Hogen se Šmejkalem jsou ve Slavii, Poborský září na Žižkově.]

[Arg1: FC Sparta leaves the field open to competition.]

[Arg2: Látal left to Schalke 04, Hogen and Šmejkal are in Slavia, Poborský shines in FC Žižkov.]

³ Coordinative connectives often connect also text spans larger than one sentence.

Example (3) illustrates the discourse semantic relation expressed (a) in a coordinative structure with a colon and (b) by two independent sentences:

(3)

(a) [Arg1: Zdá se, že to byl šťastný krok]

[Arg2: : provinční rumunský časopis se vyhranil jako médium autorů kvalitní literatury z celé Evropy.]

[Arg1: This step seems to have been lucky]

[Arg2: : the provincial Romanian magazine crystallized into a platform of high quality literature from the whole Europe.]

(b) [Arg1: Zdá se, že to byl šťastný krok.]

[Arg2: **(totiž)** Provinční rumunský časopis se vyhranil jako médium autorů kvalitní literatury z celé Evropy.]

[Arg1: This step seems to have been lucky.]

[Arg2: The provincial Romanian magazine crystallized into a platform of high quality literature from the whole Europe.]

Moreover, it turned out that this “punctuating” type of connecting discourse units is preferable in certain types of relations, see Section 4.2 below.

Third, in some cases, such a reformulation is not possible without a loss of the original meaning (as pointed out in Mladová et al., 2009) so that the syntactico-semantic relation does not hold inter-sententially.⁴ Hence, subordinate clauses which can be expressed as independent pieces of discourse without having changed their meaning (and, as mentioned, also coordinate clauses) are considered discourse-level units connected with a discourse relation, others are not.

3.2 Basic Aspects of Discourse Annotation

In our approach to discourse we decided in the first phase to annotate only semantic relations between units (text spans) containing a finite

⁴ Consider for example the following sentence (A) from Mladová et al., 2009. The syntactic form of the construction does not allow to express this type of relation by independent sentences (B).

(A) *The older the wine, the better it is. (Čím je víno starší, tím je lepší.)*

(B) **The older is the wine. The better it is. (*Čím je víno starší. Tím je lepší.)*

verb and indicated by an explicit connective.⁵ The hierarchy of discourse sense labels was established on the basis of the tectogrammatical labels (see Mikulová et al., 2005) and the Penn hierarchy of sense tags (Miltsakaki et al., 2008). The original Penn division of the sense tags to four major categories is preserved: we differentiate temporal, contingency, contrast (comparison) and expansion relations.

In the following section, we show tendencies in the behaviour of two particular discourse relations observed during the annotation process in the PDT.

4 Two Semantic Relations Expressed both in a Sentence and in a Text

We have now at our disposal approximately 33,000 sentences of Czech texts annotated both for the underlying syntax (tectogrammatrics) and for the discourse structure. We believe this builds a solid base for looking for certain tendencies in the behaviour of individual semantic relations. In the course of the development of the data annotation, we have built a hypothesis that there is a certain scale (though we do not yet present claims about determining its end points) that determines to what extent a language prefers a semantic relation to be expressed more likely within a single sentence or between sentences. In the following sections, we give examples of two relations that act very differently in this respect – condition and specification. These two relations, in our opinion, demonstrate two poles of the scale.

4.1 The Case of Condition

Mladová et al. (2009) demonstrated that the semantic relation of condition, often expressed intra-sententially, can be easily projected into an inter-sentential relation by using different language means (e.g. *if* + subordinate clause → *but* + modal verb in the second sentence), for clarification purposes we cite the example sentences below under (4):

- (4)
- (a) [Arg 1: *I will cook pancakes.*]
[Arg 2: **if** you buy eggs.]
- (b) [Arg 1: *I will cook pancakes.*]
[Arg 2: **But** you must buy eggs first.]

⁵ The only exception is the relation between a text span introducing a list structure (so-called hyper-theme) and the items of the list structure – (i) in our approach, they can be annotated also without any explicit connective, (ii) the hyper-theme needs not to be a verbal clause.

Nonetheless, our annotation indicates that in reality this type of a semantic relation strongly tends to be expressed within a sentence, as a relation between the main verb and its conditional modifier – a subordinate clause. The formulation of a conditional meaning in a language⁶ seems to be closely associated with the occurrence of a (subordinating) connective such as *if* or *when* – in Czech mainly *pokud*, *zda*, *jestli(že)*. The overview of all possible syntactic forms of condition with their distribution in the 33 thousand sentences from the PDT is presented in Table 1:

Sentence/ Discourse	Syntactic form of condition	Number of occurrences in the PDT sample ⁷
within one sentence (tree)	non-clausal modifier of the main predicate verb ⁸	651
	dependent clause (clausal (= verbal) modifier of the main predicate verb) ⁹	963
between sentences (trees)	between adjacent sentences ¹⁰	7
	long-distant relation	0

Table 1. Distribution of various types of expressing conditional meaning in the PDT

Table 1 indicates that the usage of the inter-sentential relation of condition is quite rare.

⁶ at least in languages like English or Czech

⁷ 33,000 sentences of Czech journalistic texts

⁸ Example (expression of condition in bold): *Kupující, který je získal za tisíc korun, je tedy např. může další den v **případě zájmu** prodat za 1 100 Kč.* (A buyer who got them for 1000 Czech crowns can **in case of interest** sell them the next day for 1,100 Czech crowns.)

⁹ Example (expression of condition in bold): ***Pokud** pracovník nemůže závazku z vážných důvodů dostát, omluví se včas a navrhne jiné řešení.* (If an employee for serious reasons cannot meet the obligations, he must apologize and suggest in good time a different solution.)

¹⁰ Example (expression of condition in bold): ***Posluchač musí přistoupit na pozici, že vše je dovoleno.** Potom se pobaví a také pochopí, že drama znázorňuje ztrátu reálné komunikace.* (The listener has to accept the position that everything is permitted. Then he enjoys the drama and also understands that it symbolizes the loss of a real-life communication.)

Moreover, the cases we found of such a use occur strictly between adjacent sentences, which means, the relation of condition applies neither for long distances nor between larger text units. All the cases of inter-sententially expressed conditional relations have a similar structure like the example in Table 1 (see footnote 10) – with the connective *potom/pak* (*then*) in the second argument. These findings imply that the nature of the given condition is very closely bound to the information in the second text span (the result of the condition). The best setting for relating these two pieces of information in communication is a clear delimitation of a sentence. Thus, we can state that in the repertoire of discourse-semantic relations, the condition relation tends to be one of the most condensed, the most syntax-bound.

To find out more about this matter, we compared the acquired numbers for Czech with those that were measured over the English data of the Penn Discourse Treebank (Prasad et al., 2007)¹¹. The situation is quite similar – the absolute majority of the conditional relations was assigned to discourse connectives like *if* or *when* and their modifications (e.g. *at least when*, *especially if*, *even if*, *if and when*, *only when*, *particularly if*, *until*, *unless* etc.), which are all subordinate.¹² Hence, also for English

holds that the conditional meaning tends to be expressed within a single sentence. Having discovered this symmetry, there arises an assumption that must be first verified in the course of a more detailed research, that, to a certain extent, this phenomenon is language-independent.

4.2 The Case of Specification

The semantic relation of specification occurs between two text spans when the second one describes something already expressed in the first one but in more detail. This relation corresponds on the sentential level in the PDT to the notion of apposition – the piece of information in the second span is not a new one, it only completes the information in the preceding context. In other words, when a specification relation is to be expressed intra-sententially, it fills a single syntactical position twice (see Figure 1) – first with a piece of information to some extent general, second with its details.

This relation has not been described in traditional Czech grammars¹³ and therefore many instances of the specification relation are interpreted also as conjunction in the PDT. Specification applied intra-sententially is exemplified by (5)¹⁴ (and also by Figure 1), an inter-sentential one is displayed in (6).

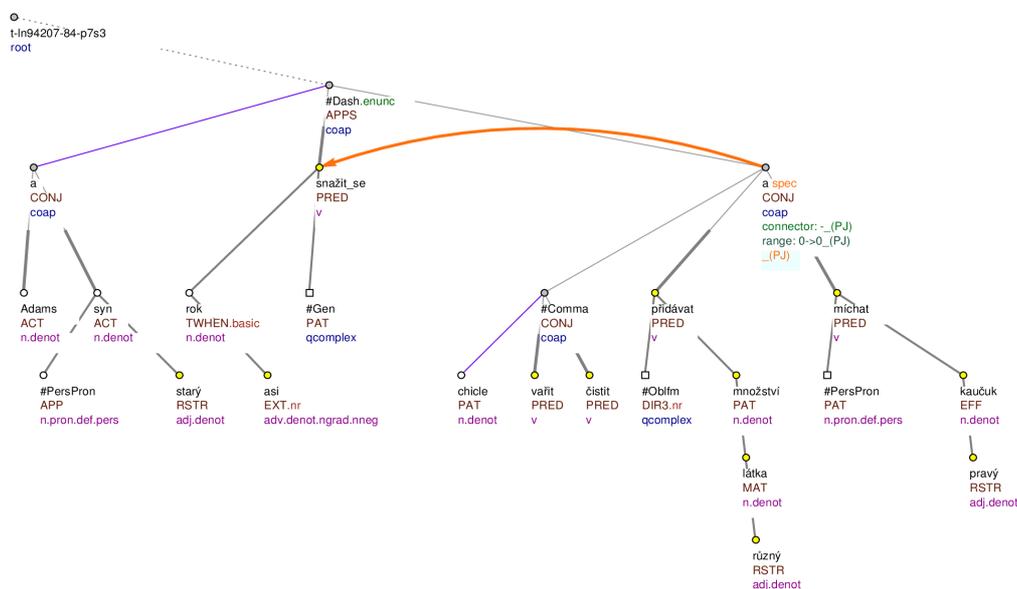


Figure 1. Apposition of two verbal nodes with the predicate function. (At the same time an example of intra-sentential specification (the bold arrow with the label *spec*). For the example sentence and its English translation see (5)).

¹¹ Approx. 49,000 sentences annotated for discourse structure.

¹² Exact distribution numbers for each connective see in Prasad et al. (2007).

¹³ as they concern primarily the issues of sentence syntax and semantics in deeper insight

¹⁴ Some necessary context is given in brackets.

(5)

[Arg1: *Asi rok se Adams a jeho nejstarší syn snažili*]

[Arg2: – *chicle vařili, čistili, přidávali množství různých látek a míchali s pravým kaučukem.*]

[Arg1: *Adams and his eldest son struggled about a year*]

[Arg2: – *they cooked chicle, cleaned it, added to it a number of different substances and mixed it with genuine rubber.*]

In (6) the semantic relation of specification occurs inter-sententially. The second sentence describes the details of the scoring and the bad start mentioned in the first sentence. This specification is indicated by the connective *totiž*, which does not have any English equivalent in this context (it can be perhaps loosely translated by “as a matter of fact”).

(6)

[Arg1: *Po dvou porážkách ve Frýdku-Místku a v Příbrami konečně zabral Havířov, ačkoliv premiéru na vlastním hřišti nezačal dobře.*]

[Arg2: *Pardubice se **totiž** ujaly vedení Pločkovou bombou ve 26. minutě, ale domácí otočili skóre třemi góly v rozpětí dvaceti minut na přelomu prvního a druhého poločasu.*]

[Arg1: *Havířov finally scored after two defeats in Frýdek-Místek and in Příbram, although the premiere at its own field did not start well.*]

[Arg2: *Pardubice (**totiž**) took lead in the 26th minute by Pločk’s bomb but players from Havířov turned the score by three goals within twenty minutes at the turn of the first and the second halves.*]

The current part of the PDT annotated for discourse structure contains 339 occurrences of the specification relation. 244 of them are expressed within one tree, only 95 between trees (moreover, 60 cases from these 95 occurrences represent the relation between a hyper-theme and a list structure and as such they either stand without connectives (36 occurrences) or are indicated by a colon (24 occurrences)). The most common connectives are punctuation marks: a colon (151 occurrences) and a dash (57 occurrences). Not only there is just one “non-punctuating” connective associated primarily with this relation – the compound connective *a to* (*and that*), but its occurrence is also restricted to special structures with an

elided verb. Other “non-punctuating” connectives associated with specification are rather typical for other relations (for results summary see Table 2). We have not found any subordinate structure to express the specification relation.

Sentence/ Discourse		Specification indicated by	Number of occurrences in PDT sample ¹⁵
within one sentence (tree)		„non-punctuating“ connective	78
		punctuation mark	166
between sentences (trees)	list structure	punctuation mark	24
		no surface connective	36
	other structure	punctuation mark	8
		„non-punctuating“ connective	27

Table 2. The distribution of the specification relation in the PDT

The decision to annotate in the first phase only relations indicated by explicit connectives limited especially the number of captured inter-sentential specifications. However, the fact that specification is the second most frequent relation with an implicit connective in the Penn Discourse Treebank (PDTB, 2,471 occurrences (Prasad et al., 2007: 90)) but it has a very low frequency when represented by explicit connectives (108 occurrences, Prasad et al., 2007: 75) supports our observation that, also in the PDT, this relation is expressed very often without any explicit connective. And this comparison enables us to go even further. If we take into account the fact that punctuation marks are supposed to be implicit connectives in the PDTB (and therefore we can only include 105 occurrences of specification in the PDT for the purpose of the comparison), we can claim that the semantic relation of specification strongly tends to be expressed inter-sententially. Only

¹⁵ 33,000 sentences of Czech journalistic texts

inter-sententially expressed specifications indicated by no surface connective can explain the evident discrepancy between our and the PDTB data (see also Table 3).

PDT sample		PDTB	
Specification indicated by	Number of occurrences	Specification indicated by	Number of occurrences
“non-punctuating” connective	105	explicit connective	108
punctuation mark	198	implicit connective	2,471
no surface connective (list structure)	36		
no surface connective in other structures	not included into annotation		

Table 3. Comparison of the distribution of the specification relation in the PDT and in the PDTB

To sum up, the specification relation is indicated preferably by punctuation marks or by the pure adjacency of sentences and the only means of its expression in one sentence is a coordinate structure. The comparison with the PDTB data supports our observation that this semantic relation is expressed primarily inter-sententially. These findings result, in our opinion, from the semantic nature of specification – the information in the second text span is not very closely bound to the information in the first text span, it only supplements the information that has already been given. Therefore, we can claim that the nature of specification is connected with the discourse structure rather than with the sentence structure.

5 Conclusion

We have demonstrated on two examples of discourse-semantic relations – condition and specification – that there are great differences in the nature of these relations, namely in their distribution in the discourse structure. Whereas the conditional meaning is expressed primarily within a single sentence and it is in an absolute majority of cases bound by a subordinate form

of expression and a usage of hypotactic language means, for the meaning of specification it is rather the opposite: it prefers to be expressed between sentences, via adjacency and with no discourse connectives at all or just with punctuation marks as a colon or a dash.

The aim of this study was to demonstrate that semantic relations between discourse units are not on the same level, but, on the contrary, their nature is quite different according to their semantic properties. In this regard, we consider the two analyzed relations to represent two poles of a scale leading from the language means used in the sentential syntax to those used in the discourse composition.

Second, the analysis of Czech and English language data processed on the basis of a similar theoretical background indicates that the findings about the nature of these semantic relations are in both languages identical, and this analysis further leads to the assumption that this phenomenon might be, at least to a certain extent, language independent.

For further enhancement of our findings, studies in three directions would be ideal to follow: (i) an analysis of the distribution of other discourse-semantic relations, for instance those from the contrast group (as we assume they might stay somewhere in between), (ii) an analysis of the distribution of discourse semantic relations in various genres (our findings are based on journalistic texts), and (iii) a comparison with data from a third, preferably typologically different language.

Acknowledgements

The research reported in this contribution has been carried out under the grant projects of the Grant Agency of the Czech Republic (GA405/09/0729), the Center for Computational Linguistics (LC536) and the Grant Agency of Charles University in Prague (GAUK 103609). We would like to thank anonymous reviewers for their detailed and insightful comments. All mistakes are ours.

References

- Hajič, Jan, Eva Hajičová, Jarmila Panevová, Petr Sgall, Jan Štěpánek, Jiří Havelka and Marie Mikulová. 2006. *Prague Dependency Treebank 2.0*. Philadelphia: Linguistic Data Consortium.
- Hajičová, Eva, Barbara H. Partee and Petr Sgall. 1998. *Topic, focus articulation, tripartite struc-*

- tures and semantic content*. Dordrecht: Kluwer Academic Press.
- Mikulová, Marie et al. 2005. *Annotation on the Tectogrammatical Layer in the Prague Dependency Treebank: Annotation Manual*. Prague: Universitas Carolina Pragensis.
- Miltsakaki, Eleni, Livio Robaldo, Alan Lee and Aravind Joshi. 2008. *Sense annotation in the Penn Discourse Treebank*. In *Computational Linguistics and Intelligent Text Processing, Lecture Notes in Computer Science*, vol. 4919: pp. 275–286.
- Mladová, Lucie, Šárka Zikánová and Eva Hajičová. 2008. *From Sentence to Discourse: Building an Annotation Scheme for Discourse Based on Prague Dependency Treebank*. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, CD-ROM.
- Mladová, Lucie, Šárka Zikánová, Zuzanna Bedřichová and Eva Hajičová. 2009. *Towards a Discourse Corpus of Czech*. In *Proceedings of the fifth Corpus Linguistics Conference*, Liverpool, UK, in press.
- Mladová, Lucie. 2009. *Annotation of Discourse Connectives for the PDT*. In *WDS'09 Proceedings of Contributed Papers*. Praha, Czechia.
- Nedoluzhko, Anja, Jiří Mirovský and Petr Pajas. 2009. *The Coding Scheme for Annotating Extended Nominal Coreference and Bridging Anaphora in the Prague Dependency Treebank*. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing*, Suntec, Singapore.
- Prasad, Rashmi, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi and Bonnie Webber. 2008. *The Penn Discourse Treebank 2.0*. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, CD-ROM.
- Prasad, Rashmi et al. 2007. *The Penn Discourse TreeBank 2.0 Annotation Manual*, available at: <http://www.seas.upenn.edu/~pdtb/PDTBAPI/pdtb-annotation-manual.pdf>.
- Sgall, Petr, Eva Hajičová and Jarmila Panevová. 1986. *The Meaning of the Sentence and its Semantic and Pragmatic Aspects*. Praha: Academia.
- Zikánová, Šárka. 2007. *Possibilities of Discourse Annotation in Prague Dependency Treebank (Based on the Penn Discourse Treebank Annotation)*. Technical report. Institute of Formal and Applied Linguistics, Charles University, Prague.
- Zikánová, Šárka, Lucie Mladová, Jiří Mirovský and Pavlína Jínová. 2010. *Typical Cases of Annotators' Disagreement in Discourse Annotations in Prague Dependency Treebank*. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta. Pages range: 2002–2006.

Dependencies over prosodic boundary tones in Spontaneous Spoken Hebrew

Vered Silber-Varod^{1,2}

¹The Open University of Israel;

²ACLP – Afeka Center for Language Processing, Afeka Tel Aviv Academic College of Engineering

vereds@openu.ac.il

Abstract

The aim of the present study is to investigate two aspects of speech: suprasegmental characteristics and syntagmatic relations. More specifically, it focused on the segmentation role of prosody and its interface with the syntagmatic sequence. While certain prosodic boundary tones seem to break speech into coherent syntactic structures, it was found that excessive elongated words are indeed prosodic breaks of various "strong" dependencies. Such a break is not due only to prosody or phonological rules, but can be attributed to the strength of syntactic relations (i.e. dependencies) between the elongated word and the word that precedes it, and between the elongated word and the following word. The findings suggest an encompassing approach to prosody-syntax interface which says that through the elongated boundaries phenomenon, speakers and listeners are exposed to the tension between the prosodic strata and the syntactic strata of language, i.e., between a prosodic *break* and syntactic *continuity*. This tension occurs about 10%-18% of spontaneous Israeli Hebrew boundary tones.

1 Introduction

The hypothesis underlying the study was that prosody and syntax are different levels of speech and therefore prosodic units do not necessarily correspond to syntactic structures. Moreover, while prosodic unit refers in the present research to the intonation unit (IU) (see (Izre'el 2010) for the role of intonation unit in spoken Israeli Hebrew), the term syntactic structures refers to "units or building blocks of different sizes, not just individual words and their [morphological] endings" (Carter and McCarthy 2006, 2). The aim was therefore to find a mechanism that regulates these two levels of human language. The research premise defines prosody as the primary linguistic tool of speech segmentation. Thus, in order to find

the regularities underlying the prosody-syntax interface, IU segmentation was carried according to a binary division between perceptually terminal and non-terminal (i.e. continuous) IU boundaries. The present research concentrates on the continuous (C)-boundary inventory in a corpus of spontaneous Israeli Hebrew. The importance of the communicative value of the C-boundary tone is in its linkage function, which will be analyzed according to the syntactic relations between the word preceding and following each of the C-boundaries.

The research will be presented as follows: In §2 I present the theoretical framework. §3 is dedicated to the methodology taken: The boundary tones in spontaneous Israeli Hebrew are introduced in §3.1, and a complex n-gram analysis is explained in §3.2. In §4 I refer to the Israeli Hebrew (IH) corpus and to the issue of word order in IH (§4.1). The research questions are presented in §5. Section 6 presents the results as a mapping between dependency relations and prosodic boundaries. In §7 I discuss the connection between form (prosodic boundary tones) and function, using Dependency Grammar (DG) terminology of *head* and *dependent*; while in §8 I present the [+dependency] feature in order to explain the results.

2 Theoretical framework

The segmentation of speech into intonation units allegedly encompasses several types of prosodic units in the prosodic hierarchy which are above the PrWd level: the phonological phrase, the intonational phrase and the utterance (Selkirk 1995), which are "to a large extent ... similar to syntactic structures" (Anderson 2005, 68). Steedman (2001) claims that "surface structure and information structure coincide, the latter simply consisting in the interpretation associated with a constituent analysis of the sentence. Intonation in turn coincides with surface structure (and hence information structure) in the sense that all in-

tonational boundaries coincide with syntactic boundaries..." (Steedman 2001, 652).

The analysis below attempts to answer the following question: How can evidence of continuous boundary tones, which are actually prosodic breaks, *within* syntactic units such as clause or phrase, be explained linguistically? The answer will use the notion of heads in grammatical theory: "the idea that one word may dominate another – that a subordinate word depends on a head word – is the central insight of traditional dependency grammar and its more recent offspring" (Fraser, Corbett, and McGlashan 1993, 3). Yet, the syntactic priority adopted in most of the theoretical approaches is increasingly disclaimed as evidence for the grammaticalization of spontaneous speech phenomena, like hesitations, self-repair or false-starts emerge in Hebrew and in other languages (Fox, Maschler, and Uhmman 2006). Thus, the main concern is to find a syntactic approach that can deal with "hesitations" or what is called here *excessive elongation* phenomenon and that is able to classify syntactically the elongated POSs, mainly function words. This goes hand in hand with Selkirk (1995), who claims that "the question of how many levels of phrasing there are in the universal Prosodic Hierarchy turns out not to be relevant to the prosodic analysis of function words." (Selkirk 1995, 5).

DG (inter alia Hudson 1993; 1996) seems to be adequate since its main concern is relations between words, or a pair of elements on the same level in a sentence, such as the relation of the subject to the predicate or of a modifier to a common noun. Moreover, the syntax-prosody interface was already studied in Mertens (2011) "sur la notions de dependance, ..." (Mertens 2011, p. 20). To this end, the terms *head* and *dependent* as well as the notion of *dependency* between words will be used.

The main relevant notions in DG to the present study are the following: DG is a grammar in which syntactic structure is expressed primarily in terms of dependency relations. One of the elements depends morphologically, syntactically, or semantically on the other. Dependency relations contrast with constituency relations which hold between elements on different levels of a sentence (Fraser 1996, 71). In DG, the syntactic structures "are represented by dependency trees or sets of nodes whose inter connections specify structural relations, i.e., a governor controls its de-

pendents by dependency rules which specify the correct structural relations for each class of unit" (Brown and Miller 1996, 397; illustrated in Fraser 1996, 72). According to Brown and Miller (1996), "in contrast with constituent structures, functional structures focuses on, not arrangements of constituents, but the relationships between constituents" (1996, xiii). Schneider (1998) notes that of the models that take the functional relations as primary, "the most syntactic" is DG, in which relations such as 'head' and 'modifier' are primary. One of the principles that he mentions concerns the syntactic duality that exists in a single word: "What is important in DG is the ability to analyze words at both levels, structural and linear: dependency is a grammar in which individual words both act as terminal nodes and as non-terminal nodes. They are terminal because they directly access the lexicon, because in its purest form, dependency only knows words; and they are non-terminal because they "require", they "subcategorize for" other words, so-called dependents." (Schneider 1998, 7).

3 Method

3.1 Prosodic annotation and distribution

As mentioned above, the present study is concerned with syntactic relations over continuous prosodic boundary tones. A boundary tone was perceptually annotated as Continuous (C) whenever the final tone of the intonation unit signaled "more to come". This annotation is primarily based on perception of the author and according to the prosodic segmentation rules described in Izre'el and Mettouchi (forthcoming: 11-19). Yet, over 15% of the corpus were similarly annotated and proofed by other researchers in several other studies (inter alia Izre'el 2005).

Continuous boundary tones were further divided into five sub-sets, and their manual annotation was carried using acoustic cues. The five C-boundaries are: Continuous Rising (C↑) tone (14% of C-boundaries), Continuous-Falling (C↓) tone (5%), Continuous Rising-Falling (C↑↓) tone (6%), Continuous Neutral (C→) tone (33%), and Continuous Elongated (C:) tone (42%). This last C: boundary tone was defined phonetically and phonologically in Silber-Varod (2010). It should be mentioned that C-boundaries are only 29% of prosodic boundaries in the corpus. Terminal boundaries

consist of 66% and truncated IUs consist of 5%.

3.2 Linear (n-gram) analysis

The present research uses linear analysis called *n*-gram. An *n*-gram model considers the probability of *n* items occurring in sequence, i.e., it is a type of probabilistic model for predicting the next item in a sequence. The probability calculation was performed on trigrams (a sequence of 3 items). The items analyzed were trigram of ApB sequences, where A and B are Parts-of-Speech and p is a C-boundary type (one of the five C-boundaries introduced in §3.1). The annotations included 36 Parts of Speech (syntax) and five C-boundaries (prosody). All annotations were manually performed on the words that precede and follow each C-boundary. Conditional probability processing was performed by AntConc software (Anthony 2007).

For example, in the string in (1) (first line is SAMPA for Hebrew transcription; the second is the translation), which includes two C-boundaries, only the underlined sequences were annotated and calculated.¹

- (1) az amaRti la Se C: etmol halaXti le Xatuna
 C↑↓ az keilu ... [D631]
 'so I told her that C: yesterday I went to a
 wedding C↑↓ so like ...'

Thus, two trigrams were extracted from (1) to the trigram inventory:

COMP C: ADV

N C↑↓ DM

where COMP is the subordinate particle [Se] 'that'; ADV for adverbs, such as [etmol] 'yesterday'; N for nouns, [Xatuna] 'wedding' in the second underlined sequence in (1); and DM for discourse markers, [az] 'so'.

It should be noted that an automatic dependency parser of Israeli Hebrew was developed by Goldberg (In progress. See also Goldberg and Elhadad 2010). Goldberg's (In progress) Easy-First parser process sentences written in Hebrew orthography, and was trained on a daily Israeli newspaper. In the present study, the analysis and annotation were carried directly over the *transcriptions* of *spontaneous* speech.

Part-of-Speech tagging in this study is based on the list of standard abbreviations in the

Leipzig Glossing Rules. Yet, additional ad hoc tags were used in the present study, such as PREP-DEF which represents the definite article /ha/ 'the' which is morphologically attached to two possible prepositions /be/ 'in, at' and /le/ 'to'. This combination of the two lexemes creates two monosyllabic CV structures, with the first consonant of the preposition and the [a] vowel of the definite article: /ba/ 'in the' and /la/ 'to the', respectively.

4 Data

The corpus used in this research contains 19 audio segments from 19 recordings that were selected from CoSIH – Corpus of Spoken Israeli Hebrew. The recordings, which were made during 2001-2002, are of authentic Israeli Hebrew everyday conversations. Each dialogue consists of conversations between one core speaker and various interlocutors with whom the speaker interacted on that day. The research corpus consists of 31,760 word-tokens (over 6 hours of speech) of which 4,289 are word-types. All recordings were manually transcribed according to SAMPA (Speech Assessment Methods Phonetic Alphabet).

The prosodic boundary tone inventory consists of 9,400 annotated boundary tones. The present research focus on the 2,775 C-boundaries (see §3.1 above).

4.1 Israeli Hebrew word order

Among the 'basic orders' found in languages of the world, Hebrew is said to prefer a SVO word order. Nevertheless, Israeli Hebrew word order is relatively free and all possible alternatives can appear in specific contexts, e.g. literature and poetry.

Several standard issues are mentioned with respect to IH word order: Adjectives always follow the nouns and numerals they modify, with exception of the numeral 'one' that always precedes it. Definite nouns are preceded by the definite article [ha] 'the', which also appears in the modifying adjective [ha-banana ha-tsehuba] (lit. the-banana the-yellow) 'the yellow banana'. Prepositions also appear at the head of the phrase. The conjunctive marker [ve] 'and' appears before the last element in the list and the subordination marker [Se] 'that' appears before the subordinate clause. Question words such as [mi] 'who', [ma] 'what',

¹ In several defined cases, the sequences were wider.

[mataj] 'when', [efo] 'where', appear at the beginning of the phrase, in standard Hebrew.

Like other Semitic languages, the isomorphic connection between phonology, morphology, syntax and semantics is much more overt when compared with the Indo-European languages. The vast majority of the words of the language can be analyzed into consonantal roots signaling broad semantic fields. These roots are combined with fixed morphophonemic patterns for what is traditionally called nominal, verbal, and adjectival forms. Nouns in IH exhibit prosodic and vocalic restrictions called *mishkal* ('weight').

In the verb system, Israeli Hebrew morphology is characterized by the non-concatenative Semitic type structure. A verb must belong to one of the five to eight morphological classes called *binyanim* ('constructions'). Verbs are also accompanied by affixes indicating tense, person, number, and gender. Rosén (1977) suggested considering the preposition as forming one constituent together with the verb: "The preposition constitutes the government properties of the verb" (Rosén 1977, 169-170). Rosén presented an example of the prepositions /le/ 'to', /be/ 'in' and /al/ 'on', and noted that, with the occurrence of certain verbs, these prepositions have no substitution, and function as cases (such as the accusative case marker [et] 'Acc.').

Nevertheless, Hebrew, as a "non-strict word-order" language, does not allow clitics and affixes at the phrase final position. Thus, the preposition stranding phenomenon does not occur in Hebrew. This characteristic of Hebrew means that we will not find prepositions in clause final position or in phrase final position (although this syntactic constraint is overruled in case of few coined idioms).

5 Research questions

The research seeks to determine what are the most probable POSs at each of the C-boundaries environment, and to see if there is a difference in the dependency distribution among C-boundaries. For example: Is a C-boundary, notably C_i, a repetition domain or a repair domain, thus finding the same POS before and after the C-boundary might serve as a clue, or is it a prosodic "bridge", in which case we would expect to find dependent POS, and C-boundaries occurring within a clause? Alternately, do we find clues to the ends of clauses

before C-boundaries, so that we can assume that C-boundaries are only minor prosodic breaks between clauses? And, of course, is there an inherent difference between the different C-boundaries, as implied by example (1) above?

6 Results

In this section, the results of both the preceding and the following POS attachments to C-boundaries will be described, in order to examine whether any relations exist between the POSs on the two sides of the C-boundary. These a-priori relations are called *dependencies* in this research, since it is assumed that C-boundaries connect dependent words (e.g., a head and its dependent(s)).

The first stage was to find regularities. This was achieved by analyzing trigrams (see §3.2) in terms of the number of repetitions and probability. After a clean-up procedure, which excluded unintelligible words, and "isolated" disfluencies, i.e. disfluencies between pauses, 2,517 sequences of "POS C-boundary POS" trigrams were examined. Of these, 962 are trigram types, of which 502 (52%) are singleton (unique) trigrams.

Table 1 shows three parameters of analysis: occurrence; conditional probability (of the first POS in the trigram sequence, given the two following items: C-boundary and the following POS);² and (assumed) syntactic dependency. The table is arranged according to probability (descending order) and it shows the 13 most probable and most frequent trigrams (The next most probable trigrams are with less than 10 occurrences).

The primary tendencies shown in Table 1 are the following: In terms of prosody, it is evident that C_i boundaries show more regularity than other C-boundaries – 9 cases vs. 3 cases of C_→ and a single case of C_↑, while the two other C-boundaries are not even in the list. It is also evident that the two *level* boundary tones, C_i and C_→, are substantial in terms of regularity.

In terms of POS *preceding* the C-boundaries, it is evident that all 9 cases of C_i have a POS of the closed class, e.g. definite article, preposition, personal pronoun. The preceding POS to C_→ and C_↑ are of the open

² I would like to thank Yoav Goldberg for his assistance with the probability calculations.

class, i.e., adjectives and nouns (lines 4, 10, 13). Although a single case of preceding POS to C→ is a pronoun (line 6), which belongs to the closed class, it was found that its "dependency" type is inherently different than the all dependencies over C: (see example in Table 1 line 6).

In terms of POS *following* the C-boundaries in Table 1, it is evident that only conjunctions follow C→ and C↑ (lines 4, 6, 10, and 13). POSs that follow C: are mostly of the open class (lines 1-3, 7, and 9), and the rest four are of the closed class.

This closed vs. open class categorization was found useful for the generalization attempts to find regularities of dependencies over C-boundaries, as is demonstrated in the five dependency types found:

1. Five cases of dependencies are *within coordination structure*. These dependencies are of two types, syndetic coordination or simply juxtaposed (a-syndetic coordination). In the present study, the syndetic coordination is either when the CONJ follows the C-boundary (lines 4, 6, 10 and 13), or when the CONJ precedes the C-boundary, mainly C: (line 12). A-syndetic construction appeared *within enumeration* dependency (not one of the most probable cases presented in Table 1).
2. The dependency in line 10 reflects the most frequent trigram in the corpus "N C→ CONJ", which occurs 52 times.
3. Three cases are assumed dependencies *within phrases*: the dependencies within NPs are of a definite article and a noun (lines 1, 2); the dependencies within PPs are of a preposition and a noun (line 3).
4. Two are dependencies *between a subject and a predicate* (lines 7 and 9).
5. Two cases show *no* dependency but an assumed repetition (lines 5 and 8).
6. A single case shows *no* dependency but an elongated discourse marker (line 11) with a following personal pronoun, which is assumed to be the subject in a new clause.

The results, partly presented in Table 1, demonstrate how *preceding* POSs can be predicted with respect to C-boundaries. This reflects the fact that a rather restricted group of closed set POSs appears before C: boundary

tone, compared to a rather varied, open class, group of POSs with each of the four other C-boundary tones.

Considering the *following* POSs, the results demonstrate, again, the similarity, in terms of POS attachment, between these four C-boundaries – C↑, C→, C↑↓, C↓ – and the unique case of the C: boundary. Only three POSs were found after the four C-boundaries: PRP, CONJ and DM. On the other hand, N was the POS most likely to appear after C:

To sum up the results, the dependencies can be scaled according to their "strengths":

1. *No* dependency: The weakest dependency is when a C-boundary does not split a syntactic dependency. This occurs when a new start begins after the C-boundary and is common to all 4 boundaries: – C↑, C→, C↑↓, C↓. The other type is when a C-boundary follows discourse markers – this is typical of C: boundaries.
2. *Within coordination construction*: A stronger dependency occurs when a coordination structure is observed. This dependency is divided into two types which affect the C-boundary distribution. When the conjunction follows the C-boundary, it is more likely that C↑, C→, C↑↓, C↓ will occur. When the conjunction precedes the C-boundary, it is most probable that the C: boundary will occur. This case can also be considered as discourse marker case, i.e., *no* dependency case, since the most frequent conjunctions [ve] 'and' and [aval] 'but' function as such in Israeli Hebrew (Fox, Maschler, and Uhmman 2006).
3. *Between a subject and a predicate*: C: boundaries are more likely to occur within this dependency.
4. *Within phrases*: This is the "strongest" dependency that C-boundaries break, and it is most likely that C: boundaries will occur here.

Although only ~30% of the prosodic boundaries in the corpus are C-boundaries, (see §3.1), the results suggests they seem to play a significant role in spoken IH, while C: boundary tone is a marked continuous boundary tone, since it regularly "breaks" grammatical dependencies.

	Preceding POS	C-boundary tone	Following POS	Occurrences	Probability	Assumed dependency and a typical example
1	definite article	C:	noun	43	0.413	Within the nominal chunk of NP
	lehavin et ha C: # et ha C: # tiskul [C412] to understand ACC the C: # ACC the C: # frustration 'to understand the the frustration'					
2	PREP-DEF	C:	noun	16	0.333	Within the nominal of NP (within PP)
	... Se ze b-a C: sefeR [G711] ... that it is in-the C: book '(it didn't seem to be) in the book'					
3	preposition	C:	noun	26	0.224	Within PP
	bXina be C: # histoRja Sel naSim [C412] exam in C: # history of women 'an exam on the history of women'					
4	adjective	C→	conjunction	17	0.157	Within coordination structure
	ze lo holeX lihjot maSehu mesubaX C→ ki hem lo holXim lehaSkia joteR midaj be maSkaot [D341] 'it is not going to be too complicated C→ since they will not invest too much in alcohol'					
5	preposition	C:	preposition	17	0.147	Repetition
	meaSeR li-Xjot be C: be hitnagSut kol ha zaman [C1111] than to-live in C: in conflict all the time 'than living in conflict all the time'					
6	pronoun	C→	conjunction	12	0.138	Within coordination structure
	halaX hisgiR-ø et atsmo C→ ve jaSav-ø mamaS ktsat zaman [C1621] go.PST.3SG. turn_in.PST-3SG ACC himself C→ and sit.PST-3SG really little time '(he) turned himself in and was imprisoned for a short time'					
7	personal pronoun	C:	verb	18	0.129	Between subject and predicate
	ve hi C: amR-a li Se hi holeXet li-Son [C514] and she C: tell.PST-3SG.F me that she go.PTCP.SG.F to-sleep 'and she told me that she was going to sleep'					
8	conjunction	C:	conjunction	21	0.124	Assumed repetition
	aval e C: # imm miSehu ja-XziR [D741] but eh C: # if someone 3SG.M-FUT.return 'but eh if someone will return [something]'					
9	personal pronoun	C:	participle	17	0.122	Between subject and predicate
	az hem C: mizdahn-im tl- neXlaSim ve noflim T [OCh] so they C: old.PTCP-PL.M @- weak.PTCP.PL.M and fall.PTCP.PL.M 'so they are getting old @- getting weak and falling down'					
10	noun	C→	conjunction	52	0.118	Within coordination structure
	kol jom medabRot ba telefon C→ ve nifgaSot ve hakol [G1241] 'every day (they) talk on the phone C→ and meet and everything'					
11	discourse marker	C:	personal pronoun	21	0.114	None
	zot omeRet e C: at pogeSet kaXa anaSim [C413] I mean eh C: you.2SG.F meet.PTCP.SG.F this_way people 'I mean eh you meet people this way'					
12	conjunction	C:	personal pronoun	19	0.112	Within coordination structure
	aval e C: hem amRu [G313] but eh C: they tell.PST.3PL.M 'but eh they told (me to write down the details)'					
13	adjective	C↑	conjunction	12	0.111	Within coordination structure
	ani mamaS gea C↑ # aval ha C: XaveRim ba avoda ... I am really proud.F C↑ # but the C: friends at work ... I am really proud but friends at work ...					

Table 1: Probabilities of the most probable trigrams with 10+ occurrences

7 Head and Dependant in the light of the results

The relevance of DG to the present research is the possibility of linking the evidence presented, specifically the evidence concerning elongated words and dependency rules. To be more specific, should a link between form (C: boundary tone) and function (either head or dependent) be established, as in the following hypothetical rules:

- Should elongated *personal pronouns* be considered dependents of *verb* heads or any other predicates?

At least in IH analysis, it is helpful to remember that there may be confusion when using morphological dependency as a criterion for defining syntactical dependency. As Schneider (1998) notes, "Many linguists ... point out that the direction of the dependency is often unclear...[but] this is only one more confusion between syntactic and morphological dependency. E.g. the main verb and the grammatical subject can be said to mutually depend on each other" (ibid., 26). Or, in other words, "the subject determines the verb morphologically, while the subject depends on the verb syntactically" (ibid., 41).

- Should elongated *articles*, e.g. [ha] 'the', be considered dependents of *noun* heads?

According to Schneider, "For this construction it seems to be hardest to determine a head and no clear answer seems to emerge yet" (1998, 48). On the other hand, Hudson (1990, 268-276) suggests the determiner as head.

- Should elongated *prepositions*, e.g., [be] 'in', be dependents of *noun* heads.
- Should elongated *subordinate conjunction* [Se] 'that' be considered dependent of a more complex unit, the subordinate clause.

These last two points are not straightforward. In verbal clauses, both *P+NP* (prepositional phrase) and *COMP+S* (subordinate clause) are verb complements, i.e. selected by the verb valence. While P assigns Case to NP or COMP assigns [+/-finite] to S, NP and S depend on the verb. Therefore, P and COMP can be parts of the nucleus. It can be said, therefore, that although dependency relations evolved from Tesnière's (1959) notion of verb valency, today valence is even attributed to

lexicalized prepositions exactly the same way Tesnière treats functional words (Schneider 1998, 52). For example, in the sequence [halaX le tel aviv] 'went to Tel Aviv', the transitive verb [halaX-ø] 'go.PST-3SG.M' and the PP [le tel-aviv] 'to Tel Aviv' are analyzed as head [halaX le] 'went to' and dependent [tel aviv] 'Tel Aviv'.

- Should elongated *conjunctions*, e.g. [ve] 'and', be considered dependents of the two structures (i.e., conjuncts) they are coordinating.

This last point is problematic, since there is *no coordination in dependency*. "In pure dependency, coordination cannot be expressed. A dependency system will have to employ a constituency element like Tesnière's *junctions*" (Schneider 1998, 90). Therefore, for current dependency theories, coordination remains a very serious problem.

Following the above hypotheses and restrictions, to determine what is head and what is dependent remains an open question as the identical prosodic form does not suggest a similar cohesion in terms of dependency functions: For constructions like *subject+verb*, *AUX+V* and *DEF+N*, perhaps even *COMP+S* and *P+NP*, "it is questionable ... if a clear dependent should be established, as both elements usually require each other. It is justifiable to think of them in terms of ... *concomitance* or to think of the first element in these constructions as a functional marker or head" (Schneider 1998, 53).

7.1 Function words as heads in IH

Since DG begins with the notion of the verb as the head, I will take a closer look at verbs in IH. Verbs are heads of items that saturate their valence, i.e. their arguments. Since elongated verbs were also found in the present research (as in (2)), a question emerges about the functional element within the verb that goes through elongation.

- (2) asi-nu e C: et ha tavoR [C614]
do.PST-1PL eh C: ACC the Tabor
'we tour eh Mount Tabor'

Indeed, the morphology of Hebrew verb structures (*binyanim*) has prefix and suffix conjugations that mark the person, and indicate gender and number (singular or plural) that are found in nouns. For example, the verb [asi-nu] 'do.PST-1PL', in (2) above, occurs in the cor-

pus three times before a C: boundary. The suffix [-nu] '1PL' has the semantic meaning of the person and number (i.e., 'we'), which means that the elongated part is the subject. It was found to be elongated in separate structures (Table 1 lines 7 and 9), and can definitely be interpreted as a dependent of V. Thus, the elongated part, when a morpheme, can be considered the functional element of the word as opposed to the substantive core element.

Another example is the gerund form in Hebrew (gerunds also have rich morphology, which is based on a root+template system). As applied to Hebrew, the term "gerund" refers either to the verb's action noun (*Shem Pe'ula*), or to the part of the infinitive following the infinitival prefix /le/ 'to'. Cases of elongated infinitival *prefixes*, shown in (3a)-(3h), also demonstrate the tendency of elongated elements to be part of functional (prefixes) vs. substantive elements (gerund):

(3) Infinitive prefixes, /le/ 'to', preceding C:

- a. at jodaat le- C: le-Sapets oto ktsat ve ze [D341]
'you know (how) to- C: to-renovate it a little and this'
- b. holeX li- C: kRot [C714]
going to- C: happen.INF
'is it going to happen'
- c. ze mamaS # jaXol la- C: le-halhiv otXa meod [G711]
it really # can to- C: to-excite ACC.2SG.M very
'it really can excite you very much'
- d. hu tsaRiX le- C: le-hotsi Xultsot CN [G831]
he need to- C: to-get out shirts CN
'he needs to to get the shirts out'
- e. ani holeXet aXSav le- C: sadeR [G831]
'I am going now to- C: tide.INF'
- f. ve holeXet l- la- C: haSlim et kol Sot ha Sena [D341]
'and going t- to- C: refill.INF all the missing sleeping hours'
- g. ve laS- la- C: asot RoSem kaze [OCh]
'and @- to- C: make.INF such an impression'
- h. az hu nivXaR me ha C: SliXim be kanada le C: le-jatseg et ha C: ... [C612]
'so he was chosen from the C: diplomats in Canada to C: to-represent the C: ...'

Although these relatively few cases can be considered coincidental, I view them as evidence of function words that sometimes cling to the preceding words, and thus together

create phonological words. This may be due to the speaker's (unconscious?) wish to utter his ideas unambiguously. Since the relations between the verb and its arguments determine the precise lexical meaning of the verb, or the several meanings of a specific verb (Stern 1994, 16-17), the meaning of that verb will be unambiguous only when an increment that neutralizes a possible ambiguity is uttered. For example, [holeXet le] 'going.F to' is an unambiguous verb as opposed to [holeXet] 'goes.F', which has an intransitive meaning as well. Such an explanation should also be relevant to the prosodic separation between the two parts of the infinitive, the infinitival prefix [le] 'to' and the gerund [sadeR] 'arrange', in the case of (3e).

The examples above suggest that the elongated category is a function element, which can be a word, a clitic, and even an affix, and that it can be interpreted as a dependent. However, viewing the elongated function elements as dependents is only one option for analysis, which suggests that the head element is uttered in a separate following IU, while C: boundaries are the most probable prosodic breaks within syntactic dependencies.

This mapping demonstrated that two main syntactic structures – phrases and clauses – were challenged by C: boundaries, while the four other boundary tones (C→, C↑, C↑↓, C↓) usually occur between phrases and clauses.

7.2 The second element of the dependency

I have attempted to explain the phenomenon of C: boundaries from the point of view of the "preceding" POS, i.e., the elongated POS, and to show that regularity exists in terms of word class (function words) and that the prosodic pattern of elongation can be explained in terms of form and function, i.e. head-dependent relations. Yet, another aspect of C-boundaries is the *following* POS, or more generally – the following syntactic structure. When Hudson (1993) compares constituency theory and dependency theory with respect to the load on working memory, he argues that dependency theory allows us to count the number of active dependencies, defining a dependency as active if either the head or the dependent are still awaited. An active dependency is satisfied as soon as the word concerned is encountered (Hudson, 1993, 275-279). At that point, the

burden on the working memory decreases and more space remains for continuous processing of information. Thus, the C: boundary tone phenomenon can be explained by Hudson's "active dependency" as a working memory load that is about to be satisfied.

8 The dependency feature

According to the dependency approach adopted here, what is common to all elongated words is the fact that they imply continuity, regardless of whether they are dependents of heads or heads of dependents. What should be stressed here is that they share a [+dependency] syntactic feature. It can be said that what is actually elongated is not the word itself (or a syllable of the word), but the syntactic *feature* itself.

For example, the results of the present research show a noun to be defined with a [-dependency] feature, since they do not tend to be elongated and since nouns tend to occur in phrase-final position rather than in phrase-initial position; a preposition, on the other hand, can be defined with [+dependency]; an intransitive verb with [-dependency], e.g. [halaX-ø] 'walk.PST-3SG.M', but a transitive verb with [+dependency], e.g. [halaX] 'go.PST-3SG.M', as in [halaX le tel aviv] 'went to Tel Aviv'. Thus, the [+dependency] feature shows that "there is more to come", and to mark the communicative intentions of the speaker. It allows the speaker to think, either the head or the dependent are still awaited, by elongating structures. In my view, what is common to elongated grammatical elements is the [+dependency] feature. I will refer to these elongated increments as *leads*.

Lead will be used here as a generic term for a variety of syntactical increments that have the [+dependency] feature and that are to be followed by another syntactical increment. In the context of the present research, *leads* are sometimes marked prosodically by the C: boundary tone. I present the term *lead* since, as was demonstrated, the term *head* cannot always be attributed to the elongated POSs in the present study (e.g., elongated personal pronouns).

To sum up, one characteristic can be said to apply to the findings of the present research on spontaneous spoken Hebrew, that of "syntactic planning coming before lexical planning" (Blanche-Benveniste 2007, 61). Blanche-

Benveniste (2007) stated that recent studies "have ... given more grammatical and semantic importance to dysfluencies.... Determiners and subjects signal the nature of the phrase-to-come, without any lexical inside. I suggest an explanation: speakers would give first the syntactic frame, with no lexical fillers, and they would only give the whole phrase, syntax and lexicon together, in a second time.... That is why getting rid of such phenomena is a linguistic mutilation." (Blanche-Benveniste 2007, 61-62). In this respect, excessive elongations are prosodic morphemes which also have a pronominal nature. This is to say that speakers first utter the syntactic frame – the lead with its [+dependency] feature, which is carried by the C: boundary tone with its pronominal nature. The lead is expected to be followed by a *syntactic increment* or a *target word*.

9 Summary

The present research attempted to describe and explain the phenomenon of excessive elongated forms by promoting prosody and prosodic patterns before the syntactic structures. The findings demonstrate a high measure of regularity of the C-boundary annotation, which can also be interpreted as regularity in spontaneous speech processing, in general, and in spontaneous spoken Hebrew, in particular.

The analysis was performed on results that showed different types of dependencies between POSs, or words, across C-boundaries. I tried to explain the dependencies through DG – a syntactic theory that can refer to prosody (inter alia, Mertens 2011), and used the terms *head* and *dependent* to find a common feature of POSs that carry the C: boundary tone (i.e., the (pre-) elongated word). In this respect, the present research brought a new perspective of the prosodic form and function relation, which encompass *all* parts of linguistic increments.

Following these results, an explanation of the role of the C-boundary tones in general, and the phenomenon of continuous elongation in particular, is offered, suggesting that the C: boundary phenomenon can be explained as a tension between the prosodic and syntactic strata of language. More specifically, the tension occurs between a prosodic break (two intonation units: one that ends with the C: tone and the following intonation unit) and the syntactic continuity, and is what enables both the

speaker and the listener to process (spontaneous) speech. The prosody-syntax interface described above clarifies the structural role of prosody in speech, that focuses on the *chaining* of prosodic units to one another (and, through this, subsequently chaining dependent syntactic units); rather than on the hierarchical nature of prosodic units.

References

- Anderson, S. R. 2005. *Aspects of the theory of clitics*. Oxford: Oxford University Press.
- Anthony, L. 2007. Antconc version 3.2.1w. Center for English Language Education, Waseda University. Available from <http://www.antlab.sci.waseda.ac.jp/index.html>.
- Blanche-Benveniste, C. 2007. Linguistic analysis of spoken language: The case of French language. In *Spoken language corpus and linguistic informatics: Contributions of linguistics, applied linguistics, computer sciences*, edited by Y. Kawaguchi, S. Zaima and T. Takagaki. Tokyo: Tokyo University of Foreign Studies.
- Brown, K. E., and J. E. Miller, eds. 1996. *Concise encyclopedia of syntactic theories*. Oxford: Elsevier.
- Carter, R., and M. McCarthy. 2006. *Cambridge grammar of English: A comprehensive guide: Spoken and written English grammar and usage*. Cambridge: Cambridge University Press.
- CoSIH – The Corpus of Spoken Israeli Hebrew. Available from <http://www.tau.ac.il/humanities/semitic/cosih.html>.
- Fraser, N. M. 1996. Dependency grammar. In *Concise encyclopedia of syntactic theories*, edited by K. E. Brown and J. E. Miller, 71-75. Oxford: Pergamon.
- Fraser, N. M., G. G. Corbett, and S. McGlashan. 1993. Introduction. In *Heads in grammatical theory*, edited by G. G. Corbett, N. M. Fraser and S. McGlashan, 1-10. Cambridge: Cambridge University Press.
- Fox, B., Y. Maschler, and S. Uhmman. 2006. A cross-linguistic study of self-repair in English, German, and Hebrew. Paper read at ICCA 2006, May 12, at Helsinki.
- Goldberg, Y. in progress. *Automatic Syntactic Processing of Modern Hebrew, PhD dissertation*. Department of Computer Science, Ben Gurion University of the Negev, Beer Sheva.
- Goldberg, Y. and M. Elhadad. 2010. Easy First dependency parsing of Modern Hebrew. *SPMRL-2010 – a NAACL/HLT workshop on Statistical Parsing of Morphologically Rich Languages*.
- Hudson, R. A. 1993. Do we have heads in our minds? In *Heads in grammatical theory*, edited by G. G. Corbett, N. M. Fraser and S. McGlashan, 266-291. Cambridge: Cambridge University Press.
- Hudson, R. A. 1996. Word grammar. In *Concise encyclopedia of syntactic theories*, edited by K. E. Brown and J. E. Miller, 368-372. Oxford: Pergamon.
- Izre'el, S. 2005. Intonation units and the structure of spontaneous spoken language: A view from Hebrew. Paper read at IDP05 – International Symposium on "Towards modeling the relations between prosody and spontaneous spoken discourse", at Aix-en-Provence, France.
- Izre'el, S. 2010. The basic unit of language: A view from spoken Israeli Hebrew. In *Proceedings of the international workshop on Afroasiatic languages*, 55-89. Japan: Tsukuba University.
- Izre'el, S. and A. Mettouchi. Forthcoming. Representation of Speech in CorpAfroAs: Transcriptional Strategies and Prosodic Units. publication of *CorpAfroAs: The Corpus of AfroAsiatic Languages* (<http://corpafroas.tge-adonis.fr/Home.html>).
- Ladd, D. R. 1996. *Intonational phonology*. Cambridge: Cambridge University Press.
- Mertens, P. 2011. Prosodie, syntaxe, discours: autour d'une approche prédictive. in: Yoo, H-Y and Delais-Roussarie, E. (eds.), *Actes d'IDP 2009, Paris, Septembre 2009*, ISSN 2114-7612, pp. 19-32.
- Rosén, H. B. 1977. *Contemporary Hebrew*. The Hague: Mouton.
- Schneider, G. 1998. A linguistic comparison of constituency, dependency and link grammar. Master's thesis, University of Zurich.
- Selkirk, E. 1995. The prosodic structure of function words. Available from <http://people.umass.edu/selkirk/pdf/PSFWUMOP%27%20copy.pdf>.
- Silber-Varod, V. 2010. Phonological aspects of hesitation disfluencies. *Speech Prosody 2010*, May 14-19, Chicago, USA.
- Steedman, M. 2001. Information structure and the syntax-phonology interface. *Linguistic Inquiry* 31(4): 649-685.
- Stern, N. 1994. *Dictionary of Hebrew verbs: The valence and distribution of the verb in contemporary Hebrew*. Ramat Gan: Bar Ilan University Press (in Hebrew).
- Tesnière, T. 1959. *Éléments de syntaxe structurale*. Klincksieck, Paris.

Statistical Language Generation from Semantic Structures

Bernd Bohnet¹, Simon Mille², Leo Wanner^{2,3}

¹ Institut für maschinelle Sprachverarbeitung (IMS)

Universität Stuttgart, {first-name.last-name}@ims.uni-stuttgart.de

² Departament de Tecnologies de la Informació i les Comunicacions

Universitat Pompeu Fabra

³Institució Catalana de Recerca i Estudis Avançats (ICREA)

{first-name.last-name}@upf.edu

Abstract

Semantic stochastic sentence realization is still in its fledgling stage. Most of the available stochastic realizers start from syntactic structures or shallow semantic input structures which still contain numerous syntactic features. This is unsatisfactory since sentence generation traditionally starts from abstract semantic or conceptual structures. However, a change of this state of affairs requires first a change of the annotation of available corpora: even multilevel annotated corpora of the CoNLL competitions contain syntax-influenced semantic structures. We address both tasks—the amendment of an existing annotation with the purpose to make it more adequate for generation and the development of a semantic stochastic realizer. We work with the English CoNLL 2009 corpus, which we map onto an abstract semantic (predicate-argument) annotation and into which we introduce a novel “deep-syntactic” annotation, which serves as intermediate structure between semantics and (surface-)syntax. Our realizer consists of a chain of decoders for mappings between adjacent levels of annotation: semantic \rightarrow deep-syntactic \rightarrow syntactic \rightarrow linearized \rightarrow morphological.

1 Introduction

Deep, or semantic, stochastic sentence generation is still in its fledgling stage. Only a few stochastic generators start from real semantic input structures; see, for instance, (Wong and Mooney, 2007; Mairesse et al., 2010), who use higher order predicate logic structures as input. Most are either restricted to syntactic generation (Bangalore and Rambow, 2000; Langkilde-Geary, 2002; Filippova and Strube, 2008) or imply a symbolic submodule that operates on semantic structures to derive syntactic structures that

are then used by the stochastic submodule (Knight and Hatzivassiloglou, 1995; Langkilde and Knight, 1998).

Walker et al. (2002) and Stent et al. (2004) start from *deep-syntactic structures* (DSyntSs) as introduced in the Meaning-Text Theory (MTT) (Mel’čuk, 1988), which they consider to be semantic. However, as argued by numerous authors, DSyntSs are, in fact, genuine syntactic structures, although they reflect the valency of the lexemes.

Bohnet et al. (2010) use CoNLL 2009 shared task corpora (Hajič, 2009) annotated in accordance with the PropBank/NomBank annotation guidelines (Palmer et al., 2005; Meyers et al., 2004), which they preprocess to adapt for dependency-based generation: non-connected adjectival modifiers are annotated as predicates with their syntactic heads as arguments, detached verbal arguments are connected with their head, etc. However, the result of this preprocessing stage is still not a genuine semantic structure: it contains all nodes of a (surface-) syntactic structure (auxiliaries, governed prepositions, determiners, etc.), including the nodes of functional words, and the part of speech tags of the individual nodes. Furthermore, it maintains the syntactic traces of the PropBank annotation such as the orientation of modifier relations and annotation of control and relative constructions.

All these types of information cannot be counted upon in most applications of natural language generation (NLG), which start from numeric time series or conceptual or semantic structures. In order to ensure a high quality linguistic generation, sentence realizers must be able to take as input abstract semantic structures derived from numeric time series or conceptual structures. In this paper, we present a deep sentence realizer that achieves this goal. Similar to (Bohnet et al., 2010), we start from



a CoNLL 2009 shared task corpus. However, unlike (Bohnet et al., 2010), we extend the CoNLL 2009 annotation in two respects: (i) we map the original CoNLL 2009 annotation onto a more abstract semantic annotation, and (ii) we introduce a deep-syntactic annotation in the sense of MTT (and as has already been used by Walker et al. (2002) and Stent et al. (2004)), which provides intermediate linguistic structures that do not contain any superficial functional nodes, but rather only the grammatical function structures. The introduction of the semantic annotation allows us to get close to the predicate-argument structures in general considered in generation as input structures of acceptable abstraction (Mellish et al., 2006); the introduction of the deep-syntactic annotation helps ensure high quality output in that it bridges the gap between the abstract semantic structures and concrete linguistic structures as the “surface-syntactic” structures are. So far, we carried out experiments only on the generation of English, but, in principle, our proposal is language-independent, as Bohnet et al. (2010)’s is.¹

In the next section, we introduce the two new levels of annotation of the CoNLL 2009 corpus: the semantic and deep-syntactic annotations, and describe how we obtain them. In Section 3, we present the setup of the realizer. Section 4 outlines the individual stages of sentence realization: semantics \rightarrow deep-syntax \rightarrow (surface-)syntax \rightarrow linearized structure \rightarrow chain of inflected wordforms. Section 5 describes the setup of the experiments for the evaluation of the realizer and discusses the results of the evaluation. Section 7, finally, summarizes the most important features of the realizer and compares it to other recent approaches in the field.

2 Adjusting the CoNLL Annotation

As mentioned above, it is common in NLG to start from abstract input representations: conceptual or semantic structures derived from ontologies or even from numeric time series. Since it is not feasible to map such input structures to the linguistic surface in one shot without sacrificing the entire potential of linguistic variation, most generators draw on

¹Obviously, the derivation of the semantic structure, which draws upon the available syntactic features remains language-specific.

models that foresee a number of intermediate representations. Common are: 1) conceptual or semantic representation that is close to the abstraction of the knowledge in ontologies; 2) syntactic representation that captures the sentence structure; 3) a linearized morphological representation that spells out the inflection and orders the words in the sentence; see (Mellish et al., 2006) for an overview.

In order to get close to this ideal picture, we not only ensure, as Bohnet et al. (2010) do, that the starting semantic structure, i.e., the PropBank annotation, is a connected graph, but, furthermore, make it truly semantic. Furthermore, we introduce the MTT’s DSyntS as an intermediate structure. DSyntS links to the semantic structure (SemS) in that it does not contain any function words, and, at the same time, to the CoNLL syntactic structure (SyntS) in that it contains the grammatical functions of the content words. DSyntS thus facilitates a two-step semantics-syntax projection, allowing for higher quality generation. For an evaluation of the quality of our annotations on a manually annotated gold standard, see (Wanner et al., submitted).

2.1 Deriving the Semantic Annotation

In order to turn a PropBank/NomBank-annotation, which, when visualized as a tree, looks as illustrated in Figure 1,² into a genuine semantic input annotation that can serve as departure for stochastic sentence generation, we 1) exclude the functional nodes from the annotation, 2) substitute syntactically motivated arcs by semantic arcs, 3) introduce missing semantic nodes, minimal information structure, and 4) ensure connectivity of the semantic annotation.

1. Removal of functional nodes and syntactic edges: The following functional nodes and syntactic edges are removed from the PropBank annota-

²Ai (i = 1,2,3,...) denotes the i-th argument of a predicative word according to this word’s frame (\approx valency) structure; A0 denotes “the external argument” of a predicative word; and AM-X denotes a modifier of type X (X = TMP (temporal), LOC(ation), DIR(ection), MNR (manner), etc.). In the course of this section, we also refer to R-Ai, C-Ai, NMOD, etc.: R-Ai (i = 1,2,3,...) stands for “i-th argument in a relative clause”; and C-Ai (i = 1,2,3,...) for “i-th argument in a control construction”. For further details, see, e.g., (Palmer et al., 2005; Meyers et al., 2004) and references therein. NMOD, PMOD, VBD, etc. are Penn TreeBank tags.

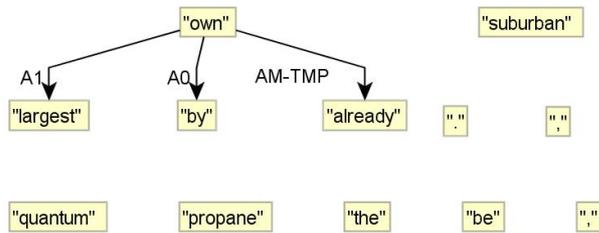


Figure 1: PropBank/NomBank annotation of the sentence *The largest, Suburban Propane, was already owned by Quantum.*

tion: (i) governed prepositions (i.e., prepositions annotated as predicate arguments A1, A2, ...); (ii) relative pronouns (i.e., nodes connected to the governing verb by an “R-Ax” edge); (iii) determiners and analytical auxiliaries (identified as such in the Penn TreeBank and PropBank annotations.);³ (iv) control construction C-Ax edges since they stand for a syntactic dependency between a semantically controlled element and a verbal predicate.

2. Substitution of syntactically motivated edges:

“Modifier” construction edges in PropBank AM-DIR, AM-LOC, AM-MNR, AM-TMP, AM-EXT, AM-PRD, AM-PNC, AM-CAU, AM-ADV, and AM-NEG are in their nature syntactic edges in that they go from the modified to the modifier. However, from the semantic view, the “modifiers” (or, better, “attributes” since we talk about semantic structure) are, in fact, predicative semantemes that take as argument the node that governs them in the syntactic structure. As a consequence, for these nodes we invert the original arc and label it with A1 in most cases. In the case of semantic prepositions and adverbs with two arguments, the second actant is linked to the preposition/adverb in question by an A2-edge.

3. Introduce missing semantic information: The PropBank annotation does not encode number and tense information, except for verbs with an analytical tense auxiliary. Since we remove aux-

³Interrogative pronouns are annotated the same way as relative pronouns in PB, but they are not removed since their removal would imply a loss of meaning; instead, we invert the R-Ax edge and relabel it with an arc “A1”: an interrogative pronoun is also a semantic predicate having as argument what is being questioned.

iliaries, we add a tense feature to every predicate which has tense; similarly, we add a number feature to every noun:⁴ TENSE: “past” for the PoS-tags VBD, VDD, VHD, VVD and “pres(ent)” for the PoS-tags VBP|VBZ, VDP|VDZ, VHP|VHZ, VVP|VVZ;⁵ NUMBER: “singular” for the PoS-tags NN and NNP and “plural” for the PoS-tags NNS and NNPS.

4. Introduce minimal information structure:

In order to be able to map the semantic structure onto a syntactic tree, a minimal information (or *communicative* in terms of Mel’čuk (2001)) structure that captures theme/rheme and given/new is needed. We add the THEMATICITY and GIVENNESS features: “THEMATICITY = theme” is assigned to the element which acts as subject in the syntactic structure and “THEMATICITY = rheme” to the main verb, the objects and close verb modifiers; “DEFINITENESS = 1” is assigned to elements with an indefinite determiner in the syntactic structure, and “DEFINITENESS = 2|3” to elements with a definite|demonstrative determiner.

5. Ensure connectivity of the semantic structure

As Bohnet et al. (2010), we ensure that the resulting semantic structure is a connected graph in that we traverse the syntactic dependency tree (i.e., the Penn Treebank annotation) d_{s_i} of each sentence x_i in the corpus breadth first and examine for each of d_{s_i} ’s nodes n whether (i) it has a correspondence node n' in d_{s_i} ’s semantic structure s_i obtained from the original shallow semantic graph in stages 1–4 sketched above, and (ii) n' is connected to the node that is n ’s semantic correspondence node. If not, we introduce a new arc between them. However, unlike Bohnet et al. (2010), who use a look-up table to read out the direction and labels of the introduced arcs, we implemented a rule-based procedure. This procedure makes use of PoS tags, syntactic arc labels, and the linearization information contained in the syntactic tree. Figure 2 shows a sample SemS as obtained applying Algorithm 2.⁶

⁴By doing so, we follow the newly announced Surface Generation Challenge <http://www.nltg.brighton.ac.uk/research/genchal11>.

⁵In case of analytical constructions (e.g., *has built*), the tense-feature is not directly on the verb, but derived from the syntactic construction.

⁶The passive of *own* is captured in the semantic annotation by the communicative feature “THEMATICITY = theme” as-

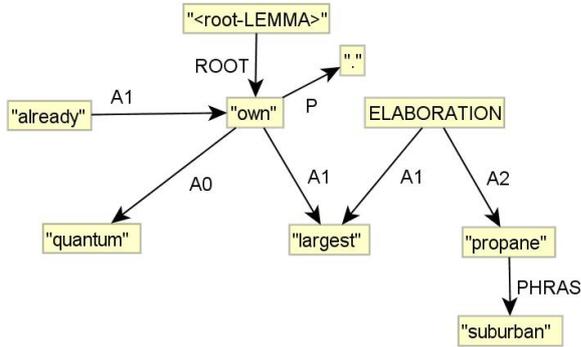


Figure 2: Semantic annotation of the sentence *The largest, Suburban Propane, was already owned by Quantum.* (the features assigned to each node are not shown)

2.2 Deriving the Deep-Syntactic Annotation

As pointed out above, DsyntS is meant to facilitate the mapping between the abstract semantic structure obtained as described above and the CoNLL syntactic structure. It contains only content nodes, i.e., nodes of the semantic structure (function words are removed, and some nodes such as "QUANTITY" or "ELABORATION" are inserted into the semantic and deep-syntactic structures), and, at the same time, syntactic relations since the deep syntactic structure shows explicitly the structure of the sentence. That is, the governors and dependents are not organized based on predicate/argument relations, but rather on the notion of syntactic governor. The syntactic governor of a lexeme is the one that imposes syntactic constraints on its dependents: linearization and agreement constraints, case or governed preposition assignments, etc. Hence, like the syntactic structure, the deep-syntactic structure representation is a tree, not a graph. Every node at this level contains part-of-speech tags. Figure 3 shows a sample dsynts.

3 Setup of the Realizer

Our sentence realizer performs the following mappings to generate a sentence for a given semantic input graph:

1. *Semantic graph* \rightarrow *Deep-syntactic tree*
2. *Deep-syntactic tree* \rightarrow *Syntactic tree*
3. *Syntactic tree* \rightarrow *Linearized structure*
4. *Linearized structure* \rightarrow *Surface*

signed to *largest*.

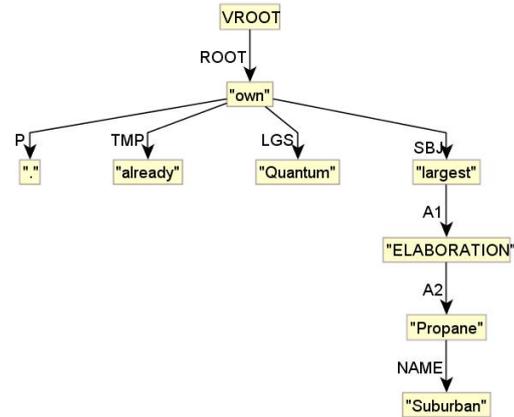


Figure 3: Deep-syntactic annotation of the sentence *The largest, Suburban Propane, was already owned by Quantum.* (the features assigned to each node are not shown)

Each of the steps is carried out by a decoder that uses a classifier to select the appropriate operations.

As already Bohnet et al. (2010), we use MIRA (Margin Infused Relaxed Algorithm) (Crammer et al., 2006) for the realization of the classifiers. MIRA has been successfully applied to structured learning tasks such as dependency parsing and semantic role labeling.⁷

We have to perform similar tasks for generation. The goal is to obtain a function that separates correct realizations (or items) by a decoder from the incorrect realizations. The items are characterised by features provided by feature extractors. The features are used to obtain a weight vector that separates the correct and incorrect items. The features are represented as a vector $\phi(x_i)$, which can be multiplied with the weight vector w in order to obtain a score.

The weight vector w can be obtained by an on-line learning algorithm. Online training considers a training example in each iteration of the training procedure. This has the advantage that we can process one example at a time, keeping only this example in the memory.

Algorithm 1 shows the outline of the training algorithm. The algorithm iterates I times over all training examples $\tau(x_i, y_i)_{i=1}^n$. A passive-

⁷The difference between MIRA and the perceptron algorithm is the use of a loss function by MIRA during the training procedure that measures the regret or cost for a wrong classification y' compared to the correct one y .

aggressive weight vector update strategy updates at the beginning of the training procedure the weights more aggressively. To what extent is determined by the factor β .

The weight vector v accumulates all weights, which are *averaged* at the end of the algorithm to avoid overfitting (Collins, 2002).

Algorithm 1: Online learning

Input: $\tau = \{(x_i, y_i)\}_{i=1}^n$
 $w^{(0)} = 0; v = 0; i = 0;$
 $\beta = I * N$
for $n = 1$ **to** I // Training iterations
 for $n = 1$ **to** N // Training instances
 $w^{(i+1)} = \text{update } w^{(i)}$ according to (x_i, y_i)
 $v = v + \beta w^{i+1}$
 $i = i + 1$
 $\beta = \beta - 1$
 $w = v / (I * N)$

4 Sentence Generation

Sentence generation consists in the application of the previously trained decoders in the sequence outlined in the previous section.

4.1 Semantic Generation

Our approach to semantic generation, which consists of the derivation of the deep-syntactic tree from an input semantic graph, is analogous to graph-based parsing (Eisner, 1996; McDonald and Pereira, 2006).

The derivation is defined as search for the highest scoring tree y from all possible trees given an input graph x :

$$F(x) = \text{argmax } \text{Score}(y), \text{ where } y \in \text{MAP}(x)$$

(with $\text{MAP}(x)$ as the set of all trees spanning over the nodes of the semantic graph x).

As already proposed by Bohnet et al. (2010), the search is a beam search which creates a maximum spanning tree.⁸ Unlike Bohnet et al. (2010), however, we use “early update” as introduced for parsing by Collins and Roark (2004): when the correct beam element drops out of the beam, we stop and update the model using the best partial solution. The idea

⁸The maximum spanning tree algorithm can be applied here thanks to the introduction of the deep-syntactic structure.

behind this is that when all items in the current beam are incorrect, further processing is obsolete since the correct solution cannot be reached extending any elements of the beam. When we reach a final state, i.e. a tree spanning over all words and the correct solution is in the beam, but not ranked first, we perform an update as well since the correct element should have ranked first in the beam.

Algorithm 2 displays the algorithm for the generation of the deep-syntactic structure from the semantic structure. *extend-trees* is the central function of the algorithm. It expands a tree or a set of trees by one edge, selecting each time the highest scoring edge. Attachment point for an outgoing edge is any node; for an incoming edge only the top node of the built tree.

Algorithm 2: Semantic generation

// (x_i, y_i) semantic graph and the deep syntactic tree
// beam-size $\leftarrow 80$
// build an initial tree
for all $n_1 \in x_i$ **do**
 trees $\leftarrow \{\}$ // empty list of partial trees
 for all $n_2 \in x_i$ **do**
 if $n_1 \neq n_2$ **then**
 for all $l \in \text{edge-labels}$ **do**
 trees = trees $\cup \{(\text{synt}(n_1), \text{synt}(n_2), l)\}$
 trees $\leftarrow \text{sort-trees-descending-to-score}(\text{trees})$
 trees $\leftarrow \text{subset}(0, \text{beam-size}, \text{trees})$
// extend the initial trees consisting of one edge
while rest $\neq \emptyset$ **do**
 trees $\leftarrow \text{extend-trees}(\text{trees})$
 trees $\leftarrow \text{sort-trees-descending-to-score}(\text{trees})$
 trees $\leftarrow \text{subset}(0, \text{beam-size}, \text{trees})$
 // training: **if** gold tree is not in the beam
 // **then** update weight vector and continue with next
return first element of trees

For score calculation, we use structured features composed of the following elements: (i) the lemmata, (ii) the **distance** between the starting node s and the target node t , (iii) the **direction** of the path (if the path has a direction), (iv) the sorted **bag** of incoming edges labels without repetition, (v) the **path** of edge labels between source and target node. The templates of the composed structured features are listed in Table 1. We obtain about 2.6 Million features in total. The features have binary values, meaning that a structure has a specific feature or it does

not.

feature templates
label+dist(s, t)+dir
label+dist(s, t)+lemma $_s$ +dir
label+dist(s, t)+lemma $_t$ +dir
label+dist(s, t)+lemma $_s$ +lemma $_t$ +dir
label+dist(s, t)+bag $_s$ +dir
label+dist(s, t)+bag $_t$ +dir
label+path(s, t)+dir

Table 1: Feature templates for the semantic \rightarrow deep-syntactic mapping ('s' means "source node" and 't' "target node" of an edge)

4.2 Deep-Syntactic Generation

Since the DSyntStr contains by definition only content words, function words such as governed prepositions, auxiliaries, and determiners must be introduced during the deep-syntactic–surface-syntactic generation passage in order to obtain a fully spelled out syntactic tree.

Tree transducers are best suited for this task because of their capability to rewrite trees. Top down tree transducers have been independently introduced by Rounds (1970) and Thatcher (1970) as extensions of finite state transducers. Tree Transducers have been already successfully applied in NLP—for instance, in machine translation (Knight and Graehl, 2005). Tree transducers traverse the input trees from the root to the leaves and rewrite the tree using rewriting rules.

For DSynt-generation, we use around 280 rules derived automatically by comparing a gold standard set of deep-syntactic structures and surface-syntactic dependency trees. The rules are of the following three types:

1. Rules introducing an edge and a node:
 $X \Rightarrow X \text{ label}_s \rightarrow Y$,
 Example: $X \Rightarrow X \text{ NMOD} \rightarrow \text{'the'}$
2. Rules introducing a new node and edges between two nodes:
 $X \text{ label}_a \rightarrow Y \Rightarrow X \text{ label}_s^1 \rightarrow N \text{ label}_s^2 \rightarrow Y$
 Example: $X \text{ OPRD} \rightarrow Y \Rightarrow X \text{ OPRD} \rightarrow \text{'to' } IM \rightarrow Y$
3. Rules introducing a new node label:
 $X \Rightarrow N$
 Example: 'LOCATION' \Rightarrow 'on'

The restricted number of rules and rule types suggests the use of classifiers to select applicable rules

in each stage of the DSynt-generation and thus consider more contextual information for the decision.

We train discriminative classifiers for each of three rule types that either selects a specific rule or NONE (i.e., that no rule is to be applied). Some parts do not need any changes. Therefore, on this parts there is no need to apply and the classifier has to select NONE. The Algorithm 3 displays the algorithm for the generation of the surface-syntactic structure from the deep-syntactic structure. The algorithm uses for score calculation features listed in Table 2.

Algorithm 3: Deep Syntactic Generation

```

//( $x_i, y_i^g$ ) the deep syntactic tree
// and gold surface syntactic tree for training case only
//  $R$  set of rules
// travers the tree top down depth first
 $y_i \leftarrow \text{clone}(x_i)$ 
node-queue  $\leftarrow \text{root}(x_i)$ 
while node-queue  $\neq \emptyset$  do
  //depth first traversal
  node  $\leftarrow \text{remove-first-element}(\text{node-queue})$ 
  node-queue  $\leftarrow \text{children}(\text{node}, x_i) \cup \text{node-queue}$ 
  // select the rules, which insert a leaf node
  leaf-insert-rules  $\leftarrow \text{select-leaf-rules}(\text{next-node}, x_i, R)$ 
   $y_i \leftarrow \text{apply}(\text{leaf-insert-rules}, y_i)$ 
  // in the training, we update here the weight vector
  // if the rules are not equal to the gold rules
  //
  // select the rules, which insert a node in the tree
  // or a new node label
  node-insert-rules  $\leftarrow \text{select-node-rules}(\text{node}, x_i, R)$ 
  // in the training, we update here the weight vector
   $y_i \leftarrow \text{apply}(\text{edge-insert-rules}, y_i)$ 

```

Table 3 shows the confusion matrix of the DSynt \rightarrow SSynt transducer rules. The first column contains the number of the gold rule that should have been applied; the second the gold rule itself and the third the actually applied rule. 'ie:' is the prefix of "insert-edge" rules, and 'in:' the prefix of "insert-node" rules.⁹

As we see, confusions occur, first of all, in the selection of the correct preposition in <nominal modifier>–<prepositional modifier> sequences in

⁹We hope that the Penn TreeBank tags 'NMOD', 'PMOD', 'DIR', 'OBJ', etc. are intuitive enough to allow for the understanding of the semantics of the rules.

feature template
pos(node)
pos(head(node))
pos(head(head(node)))
pos(node)+pos(head((node)))
pos(node) + pos(head(node))+ edge-label(node)
feature-1(node)
feature-2(node)
feature-3(node)
feature-1(node)+feature-2(node)
lemma(node)
lemma(head(node))
lemma(node)+lemma(head(node))
bag-of-children-pos(node)
sorted-bag-of-children-pos(node)
sorted-bag-of-children-labels(node)

Table 2: *pos* are coarse-grained Part-of-Speech tags, *feature* are the features attached to the nodes, *lemma* are node labels, *edge-label* labels of edges; *feature-1* stands for “definite=yes”, *feature-2* for “num=sg”, and *feature-3* for “tense=past”

# rule	gold rule	wrongly applied rule
65	ie:NMOD:for:PMOD	ie:NMOD:of:PMOD
40	ie:LOC:in:PMOD	ie:NMOD:of:PMOD
34	ie:NMOD:to:PMOD	ie:NMOD:of:PMOD
23	ie:NMOD:on:PMOD	ie:NMOD:of:PMOD
26	ie:NMOD:with:PMOD	ie:NMOD:of:PMOD
18	ie:NMOD:from:PMOD	ie:NMOD:of:PMOD
16	ie:DIR:to:PMOD	ie:ADV:to:PMOD
12	ie:DIR:from:PMOD	ie:DIR:to:PMOD
11	in:NMOD:to	
11	ie:NMOD:of:PMOD	
10	ie:NMOD:of:PMOD	ie:LOC:in:PMOD
9	ie:ADV:at:PMOD	ie:ADV:for:PMOD
9	ie:DIR:from:PMOD	ie:ADV:from:PMOD
6	ie:PMOD:to:PMOD	
8	ie:OBJ:that:SUB	
8	ie:OPRD:to:IM	
8	ie:LOC:at:PMOD	ie:NMOD:with:PMOD

Table 3: Confusion matrix of the dsynt \rightarrow synt rules

edge inserting rules. A possible solution to this problem that needs to be further explored is the inclusion of a larger context or/and consideration of semantic features.

4.3 Linearization and Morphologization

There is already a body of work available in statistical text generation on linearization and morphological realization. Therefore, these subtasks did not form the focus of our work. In the current version of the realizer, we use Bohnet et al. (2010)’s implementations. The linearization is a beam search for an optimal linearization according to a local and global score functions.

The morphological realization algorithm selects the edit script based on the minimal string edit distance (Levenshtein, 1966) in accordance with the highest score for each lemma of a sentence obtained during training and applies then the scripts to obtain the wordforms.

5 Experiments

To evaluate the proposed realizer, we carried out a number of experiments, whose setup and results are presented in what follows.

5.1 Setup of the Experiments

In our experiments, we use the PropBank/NomBank corpus of the CoNLL shared task 2009, which we preprocess as described in Section 2 to obtain the semantic structure from which we start. We follow the usual training, development and test data split (Langkilde-Geary, 2002; Ringger et al., 2004; Bohnet et al., 2010). Table 4 provides an overview of the used data.¹⁰

set	section	# sentences
training	2 - 21	39218
development	24	1334
test	23	2400

Table 4: Data split of the used data in the WSJ Corpus

In order to measure the accuracy of the isolated components and of the realizer as a whole and to be able to compare their performance with previous works, we use measures already used before, for instance, in (Ringger et al., 2004; Bohnet et al., 2010). Thus, for the semantics \rightarrow deep-syntax mapping, we use the unlabeled and labeled attachment score, as it is also commonly used in dependency parsing. The unlabeled attachment score (ULA) is the percentage of correctly identified heads. The labeled attachment score (LAS) is the percentage of correctly identified heads that are in addition correctly labeled by syntactic functions. For the assessment of the deep-syntax \rightarrow syntax mapping, we use the F-score of correctly/wrongly introduced nodes. For the evaluation of the sentence realizer as a whole, we use

¹⁰The raw PropBank/NomBank corpus of the CoNLL shared task 2009 is the WSJ corpus, such that the section numbers refer to sections in the WSJ corpus.

the BLEU metric on a gold standard compiled from our corpus.

Since we use Bohnet et al. (2010)’s implementations of the linearization and morphological realization, we use their metrics as well. To assess linearization, three metrics are used: (i) per-phrase/per-clause accuracy (*acc snt.*):

$$acc = \frac{\text{correct constituents}}{\text{all constituents}},$$

(ii) edit distance metrics:

$$di = 1 - \frac{m}{\text{total number of words}}$$

with m as the minimum number of deletions combined with insertions to obtain the correct order (Ringger et al., 2004); and (iii) the BLEU-score.

For the assessment of the morphological realization, the accuracy score (the ratio between correctly generated word forms and the entire set of generated word forms) is used.

5.2 Results of the Experiments

Table 5 displays the figures obtained for both the isolated stages of the semantic sentence realization and the generation as a whole—with reference to some of the recent works on statistical generation, and, in particular to (Bohnet et al., 2010), which is most similar to our proposal.¹¹ We include the performance of (Bohnet et al., 2010) in two stages that differ from our semantics→syntax, and syntax→topology (or linearized structure), and its overall performance. (Filippova and Strube, 2009) and (Ringger et al., 2004) are, in fact, not fully comparable with our proposal since the data are different. Furthermore, Filippova and Strube (2009) linearize only English sentences that do not contain phrases that exceed 20,000 linearization options—which means that they filter out about 1% of the phrases. We include them because these are reference works with which any new work on statistical generation has to compete.

5.3 Discussion

The overall performance of our semantic realizer is comparable (although somewhat lower) to the performance of (Bohnet et al., 2010). This is

¹¹We do not compare here to (Wong and Mooney, 2007) and (Mairesse et al., 2010) because the tasks of both are rather different from ours: both explore phrase-based generation.

Mapping	Value
Semantics→Deep-Syntax (ULA/LAS)	93.8/87.3
Deep-Syntax→Syntax (correct)	97.5
Syntax→Topology (BLEU)	0.89
All stages (BLEU)	0.64
All stages (BLEU) (Bohnet et al., 2010)	0.659
Semantics→Syntax (ULA/LAS) (Bohnet et al., 2010)	94.77/89.76
Syntax→Topology (di/acc) (Bohnet et al., 2010)	0.91/74.96
(Filippova and Strube, 2009)	0.88/67
(Ringger et al., 2004) (BLEU)	0.836

Table 5: Performance of the individual stages of semantic sentence realization and of the realization as a whole

remarkable given that we start from a considerably more abstract semantic structure that does not contain any function words and that encodes some of the information (for instance, information structure features) in terms of node attributes instead of nodes/arcs. The performance of the semantics→deep-syntax projection is slightly lower than the semantics→syntax projection of (Bohnet et al., 2010). However, the quality of our deep-syntax→syntax projection is rather high—despite the fact that during this projection new nodes are introduced into the target structure (i.e., the projection is by far not isomorphic). A more detailed analysis of this projection shows that the precision of correctly introduced nodes is 0.79 and the recall is 0.74. As a result, we obtain an F-score of 0.765. However, the introduction of nodes affects only a relatively small part of the syntactic structure. Before we apply the rules, the (gold) deep-syntactic tree has about 92% correct nodes and correctly attached edges of the (surface) syntactic tree. After the rule application this value improves to about 97.6%. Our performance during the syntax→topology stage is slightly lower than in (Bohnet et al., 2010). This is the effect of the (imperfect) introduction of function words (such as determiners and prepositions) into the syntactic structure at the preceding stage. But it is still higher than the performance of the reference realizers such as (Ringger et al., 2004) and (Filippova and Strube, 2009) for this task.

6 Related Work

Most of the widely cited works on statistical generation which use intermediate syntactic representations, as, for instance, Knight and Hatzivassiloglou (1995), Langkilde and Knight (1998) or Ringger et al. (2004), do not handle statistically the first stage of generation. Rather, they use rule-based components to build syntactic trees—even though some of them actually tackle the issue of statistical lexicalization, which we do not. Many recent works focalize on surface realization only, i.e., linearization and morphologization of syntactic representations; see, for instance, (Bangalore and Rambow, 2000; Filippova and Strube, 2008).

Mairesse et al. (2010) describe a statistical language generator, which uses dynamic Bayesian networks to assign a semantic part directly to a phrase. The representation is based on stacks which contain the semantic information for a sentence decomposed into phrases. The Bayesian networks are used to order the phrases and to align semantic parts with phrases. The model generalizes to some degree since it contains lexicalized backoff features that reduce the needed semantic coverage. For instance, the probability $P(r = \text{centre of town} \mid s = \text{reject}(\text{area}(\text{centre})))$ is backed off by $P(r = \text{centre of town} \mid h = \text{centre})$.

Wong and Mooney (2007) present a generator based on an inverted semantic parser. The input is a partially ordered meaning representation. The process is similar to the one described in (Mairesse et al., 2010) in that they do not use any intermediate structure. Their statistical system, trained on very few sentences (880) produces concurrent output sentences. To choose the best candidate, they use n -gram models, as Knight and Hatzivassiloglou (1995), Bangalore and Rambow (2000) and Langkilde-Geary (2002). Walker et al. (2002) and Stent et al. (2004) describe a trainable sentence planner for dialog systems. The system uses MTT's DSyntSs as intermediate representations. In this respect, their approach is similar to ours. However, unlike us, they consider the DSyntSs predicate-argument structures, mapping fragments of text plans onto them by a set of operations in a bottom-up, left-to-right fashion. Starting from DSyntSs,

they then use the rule-based RealPro generator to generate the sentences (Lavoie and Rambow, 1997).

7 Conclusions

We presented a decoder-based statistical semantic sentence realizer, which goes significantly beyond the works in this area, while showing a similar or, in some aspects, even better performance. The main difference of our proposal, to the statistical realizers of Ringger et al. (2004; He et al. (2009) is that we start with the generation from a truly semantic (predicate-argument) graph. An important extension compared to (Langkilde and Knight, 1998; Bohnet et al., 2010) is the mapping from the semantic graph to the DSyntS that forms an intermediate structure between the semantic structure and the (surface-) syntactic structure. In analogy to the semantic structure, the DSyntS contains no function words, and in analogy to the syntactic structure, it contains grammatical functions of the words that are present. This is motivated by the fact that we can easily build first a syntactic structure and then, in the next step, introduce function words based on the syntactic properties. We see this approach as the most promising direction for the derivation of a highly accurate syntactic tree and also in accordance with a holistic linguistic theory, namely MTT.

Unlike many of the previous works, we do not use at any stage components that are based on manually crafted rules. The abstract nature of the semantic structure and the availability of the DSyntS is an important add-on when compared to Bohnet et al. (2010)'s proposal, which starts from a semantic graph that already contains all words. The other works on statistical generation we know of that draw upon DSyntSs, namely (Walker et al., 2002; Stent et al., 2004), seem to overestimate the semantic nature of DSyntSs in that they consider them as (semantic) predicate-argument structures, which they are not: after all, DSyntSs are and remain syntactic structures, even if abstract ones.

Although we applied our approach so far only to English, the proposed realizer is language-independent—as the one proposed by Bohnet et al. (2010). In the months to come, we will apply it to other languages. This work will be accompanied by an effort to reach truly semantic corpus anno-

tations. The mapping of the PropBank/NomBank annotation to such an annotation demonstrated that CoNLL corpora are a good starting point for such an effort. As pointed out by one of the reviewers, LFG f-structures and MTT DSyntStrs also have a lot in common—which suggests experiments on deriving DSyntStr annotated corpora from LFG corpora.

Acknowledgements

The work described in this paper has been partially funded by the European Commission under the contract number FP7-ICT-248594. We would like to thank the four anonymous reviewers for their detailed and very useful comments.

References

- S. Bangalore and O. Rambow. 2000. Exploiting a Probabilistic Hierarchical Model for Generation. In *Proceedings of COLING '00*, pages 42–48.
- B. Bohnet, L. Wanner, S. Mille, and A. Burga. 2010. Broad coverage multilingual deep sentence generation with a stochastic multi-level realizer. In *Proceedings of COLING '10*, pages 98–106.
- M. Collins and B. Roark. 2004. Incremental parsing with the perceptron algorithm. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.
- M. Collins. 2002. Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In *Proceedings of the EMNLP Conference*.
- K. Crammer, O. Dekel, S. Shalev-Shwartz, and Y. Singer. 2006. Online Passive-Aggressive Algorithms. *Journal of Machine Learning Research*, 7:551–585.
- J. Eisner. 1996. Three New Probabilistic Models for Dependency Parsing: An Exploration. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, pages 340–345, Copenhagen.
- K. Filippova and M. Strube. 2008. Sentence fusion via dependency graph compression. In *Proceedings of the EMNLP Conference*.
- K. Filippova and M. Strube. 2009. Tree linearization in English: Improving language model based approaches. In *Proceedings of the NAACL '09 and HLT, Short Papers*, pages 225–228.
- J. Hajič. 2009. The CoNLL-2009 Shared Task: Syntactic and Semantic Dependencies in Multiple Languages. In *Proceedings of the CoNLL*.
- W. He, H. Wang, Y. Guo, and T. Liu. 2009. Dependency based chinese sentence realization. In *Proceedings of the ACL and of the IJCNLP of the AFNLP*, pages 809–816.
- K. Knight and J. Graehl. 2005. An overview of probabilistic tree transducers for natural language processing. In *Sixth International Conference on Intelligent Text Processing and Computational Linguistics*. Lecture Notes in Computer Science.
- K. Knight and V. Hatzivassiloglou. 1995. Two-level, many paths generation. In *Proceedings of the ACL*.
- I. Langkilde and K. Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *Proceedings of the COLING/ACL*, pages 704–710.
- I. Langkilde-Geary. 2002. An empirical verification of coverage and correctness for a general-purpose sentence generator. In *Proceedings of the Second INLG Conference*, pages 17–28.
- B. Lavoie and O. Rambow. 1997. A fast and portable realizer for text generation systems. In *Proceedings of the 5th Conference on ANLP*.
- V.I. Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics*, 10:707–710.
- F. Mairesse, M. Gašić, F. Juričić, S. Keizer, B. Thomson, K. Yu, and S. Young. 2010. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the ACL*.
- R. McDonald and F. Pereira. 2006. Online Learning of Approximate Dependency Parsing Algorithms. In *In Proc. of EACL*, pages 81–88.
- C. Mellish, D. Scott, L. Cahill, D. Paiva, R. Evans, and M. Reape. 2006. A reference architecture for natural language generation systems. *Natural Language Engineering*, 12(1):1–34.
- I.A. Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press, Albany.
- I.A. Mel'čuk. 2001. *Communicative Organization in Natural Language : The Semantic-Communicative Structure of Sentences*. John Benjamins Publishing, Philadelphia.
- A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman. 2004. Annotating Noun Argument Structure for NomBank. In *Proceedings of LREC-2004*, Lisbon, Portugal.
- M. Palmer, D. Gildea, and P. Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–105.
- E. Ringger, M. Gamon, R.C. Moore, D. Rojas, M. Smets, and S. Corston-Oliver. 2004. Linguistically informed statistical models of constituent structure for ordering in sentence realization. In *Proceedings of COLING*, pages 673–679.

- W. Rounds. 1970. Mappings and Grammars on Trees. *Mathematical Systems Theory*.
- A. Stent, R. Prasad, and M. Walker. 2004. Trainable sentence planning for complex information presentation in spoken dialog systems. In *Proceedings of the 42th Annual Meeting of the ACL*.
- J.W. Thatcher. 1970. Generalized Sequential Machine Maps. *Computer Systems*.
- M.A. Walker, O.C. Rambow, and M. Rogati. 2002. Training a sentence planner for spoken dialogue using boosting. *Computer Speech and Language*, 16:409–433.
- L. Wanner, S. Mille, and B. Bohnet. submitted. Do we need new semantic corpus annotation policies for deep statistical generation?
- Y.W. Wong and R.J. Mooney. 2007. Generation by inverting a semantic parser that uses statistical machine translation. In *Proceedings of the HLT Conference of the NAACL*, pages 172–179.

Categorial Dependency Grammars: from Theory to Large Scale Grammars

Alexander Dikovsky
LINA CNRS UMR 6241, Université de Nantes
Alexandre.Dikovsky@univ-nantes.fr

Abstract

Categorial Dependency Grammars (CDG) generate unlimited projective and non-projective dependency structures, are completely lexicalized and analyzed in polynomial time.

We present an extension of the CDG, also analyzed in polynomial time and dedicated for large scale dependency grammars. We define for the extended CDG a specific method of “Structural Bootstrapping” consisting in incremental construction of extended CDG from representative samples of dependency structures. We also outline a wide coverage dependency grammar of French developed using this method.

1 Introduction

Categorial Dependency Grammars (CDG) were introduced in (Dikovsky, 2004). Since then, they were intensively studied (e.g., see (Béchet et al., 2004; Dekhtyar and Dikovsky, 2008; Dekhtyar et al., 2010)). CDG is very expressive. In particular, simple CDG generate such non-CF languages as $L^{(m)} = \{a_1^n a_2^n \dots a_m^n \mid n \geq 1\}$ for all $m > 0$ and $MIX = \{w \in \{a, b, c\}^+ \mid |w|_a = |w|_b = |w|_c\}$. At the same time, CDG are recognized in polynomial time. CDG have interesting mathematical properties: an extension of CDG defines an Abstract Family of Languages (AFL) (Dekhtyar and Dikovsky, 2008; Dekhtyar et al., 2010)¹, they are equivalent to real time pushdown automata with independent counters (Karlov, 2008), interesting sufficient conditions of learning CDG in the limit were recently found (Béchet et al., 2004; Béchet et al., 2010; Béchet et al., 2011).

¹CDG-languages are closed under all AFL operations, but iteration.

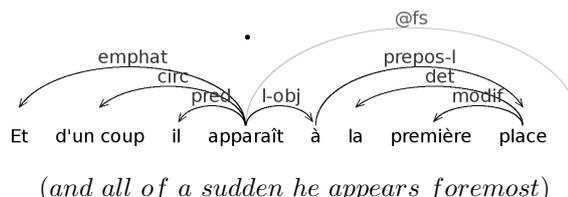
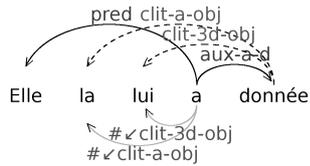


Figure 1: Projective DS

At the same time, the exact relationship between the weak generative power of the CDG and that of the so called mildly context sensitive grammars (see e.g. (Joshi et al., 1991; Shanker and Weir, 1994)) is not known.

CDG have important advantages which make them a convenient and natural means of definition of wide-coverage dependency grammars. First, they are completely lexicalized, as it is the case of all categorial, and more generally, type logical grammars (Bar-Hillel et al., 1960; Lambek, 1961; Lambek, 1999; Steedman, 1996). The second advantage of CDG is that they naturally and directly express unlimited dependency structures (DS). Basically, CDG define DS in terms of valencies of words, i.e. in terms very close to those of the traditional linguistic theories of syntax. Of course, as all dependency grammars (e.g. (Gaifman, 1961; Maruyama, 1990; Sleator and Temperly, 1993; Debusmann et al., 2001)), they express the **projective** DS, i.e. those in which the dependencies do not cross (as the one in Fig. 1). But they express as well the **non-projective** DS, in which they may cross (as in the DS shown in Fig. 2). Non-projective DS are a challenge for dependency grammars. Generally, the grammars expressing them are untractable (cf. (Debusmann et al., 2001)) or need some constraints on the DS in order to be polynomially analyzed (cf. (Ka-



(*she it[fem.] to him has given)

Figure 2: Non-projective DS

hane et al., 1998; Bröker, 2000)). As to the CDG, they are analyzed in a reasonable polynomial time using a rather standard tabular dynamic programming algorithm (see (Dekhtyar and Dikovskiy, 2008)), and this is their third advantage. Fourth, an extension of CDG by regular type expressions (RTE) specially designed for large scale grammars was proposed in (Dikovskiy, 2009). We outline this extension below. Importantly, the extended CDG are also analyzed in polynomial time.

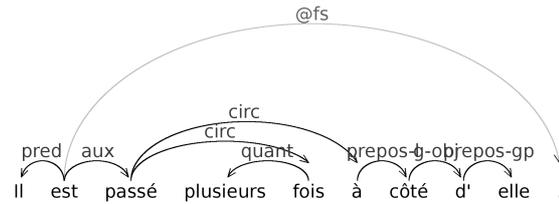
In this paper we define a simple and practical **Structural Bootstrapping Method** of incremental development of large scale extended CDG from representative samples of DS. Using this method and a toolkit specially designed for CDG (Alfared et al., 2011), we have developed in a short space of time a rather complete dependency grammar of French, briefly described below.

The plan of this paper is as follows. The next Section introduces the CDG. Section 3 presents their extension by RTE. In Section 4 is defined and illustrated the Method of Structural Bootstrapping of extended CDG. Finally, a wide scope extended CDG of French developed by this method is outlined in Section 5.

2 Categorical Dependency Grammars

CDG originate from a straightforward encoding of DS in terms of dependency relation valencies of words. Basically, they are classical categorial grammars with subtypes interpreted as dependency valencies and with categories extended by potentials defining non-projective dependencies. Valencies of projective and of non-projective dependencies are encoded differently.

When in a DS D there is an arc $w_1 \xrightarrow{d} w_2$, we say that d is a dependency between w_1 and w_2 , w_1 is the **governor** and w_2 is **subordinate**



(he passed by her side several times)

Figure 3: Repetitive dependency *circ*

to w_1 through d . E.g., in Fig. 1, *il* (*he*) is subordinate to *apparaît* (*appears*) through *pred* and the locative case preposition *à* governs *place* through dependency *prepos-l*.

The valency of a governor of w through dependency d is encoded by d itself. In particular, the no-governor valency of the root word is encoded by S (a special symbol called **axiom**).

The valency of a left subordinate of w through projective dependency l is encoded by $\backslash l$ and that of a right subordinate through projective dependency r is encoded by $/r$. The set of all left valencies of a word is encoded by the concatenation of their codes. So, for instance, the valencies of projective dependencies of the root word *apparaît* in the DS in Fig. 1 is encoded by the expression $pred \backslash circ \backslash emphat \backslash S / @fs / l - obj$.

The repetitive dependencies are a special case. A dependency d is **repetitive** (see (Mel'čuk, 1988)) if a word may have more than one subordinates through d . The valency of left repetitive dependency d is encoded by d^* (the right one is encoded by $/d^*$). So, e.g. in the DS in Fig. 3, the valencies of projective dependencies of the word *passé* (*passed*) are encoded by the expression $aux / circ^*$.

In CDG, the non-projective dependency valencies of a word w are **polarized**. They are of four kinds: $\swarrow v$, $\searrow v$ (**negative**) and $\nwarrow v$, $\nearrow v$ (**positive**). E.g., when a word w has valency $\swarrow v$, it intuitively means that its governor through dependency v must occur somewhere on the right. Two polarized valencies with the same valency name v and orientation, but with the opposite signs are **dual**. Together they define non-projective dependency v . The (possibly empty) set of all non-projective dependencies of a word w is encoded by the concatenation of the correspond-

$$\begin{array}{c}
\frac{\frac{[\#(\sphericalangle \textit{clit} - a - \textit{obj})]^{\sphericalangle \textit{clit} - a - \textit{obj}} \quad \frac{[\#(\sphericalangle \textit{clit} - 3d - \textit{obj})]^{\sphericalangle \textit{clit} - 3d - \textit{obj}} [\#(\sphericalangle \textit{clit} - a - \textit{obj}) \setminus \#(\sphericalangle \textit{clit} - a - \textit{obj}) \setminus \textit{pred} \setminus S / \textit{aux} - a - d]}{[\#(\sphericalangle \textit{clit} - a - \textit{obj}) \setminus \textit{pred} \setminus S / \textit{aux} - a - d]^{\sphericalangle \textit{clit} - 3d - \textit{obj}}} (\mathbf{L}^1)}{[\textit{pred}]} (\mathbf{L}^1)} \\
\frac{[\textit{pred} \setminus S / \textit{aux} - a - d]^{\sphericalangle \textit{clit} - a - \textit{obj} \setminus \textit{clit} - 3d - \textit{obj}} \quad \frac{[\#(\sphericalangle \textit{clit} - a - \textit{obj}) \setminus \textit{pred} \setminus S / \textit{aux} - a - d]^{\sphericalangle \textit{clit} - 3d - \textit{obj}}}{[\textit{aux} - a - d]^{\sphericalangle \textit{clit} - 3d - \textit{obj} \setminus \textit{clit} - a - \textit{obj}}} (\mathbf{L}^1)}{[S / \textit{aux} - a - d]^{\sphericalangle \textit{clit} - a - \textit{obj} \setminus \textit{clit} - 3d - \textit{obj}}} (\mathbf{L}^1)} \\
\frac{[S]^{\sphericalangle \textit{clit} - a - \textit{obj} \setminus \textit{clit} - 3d - \textit{obj} \setminus \textit{clit} - 3d - \textit{obj} \setminus \textit{clit} - a - \textit{obj}}}{S} (\mathbf{D}^1 \times 2)
\end{array}$$

Figure 4: Dependency structure correctness proof

ing polarized valencies called **potential** of w .² E.g., in the DS in Fig. 2, the participle *donnée* has potential $\sphericalangle \textit{clit} - a - \textit{obj} \setminus \textit{clit} - 3d - \textit{obj}$, which means that it needs, somewhere on its left, a word subordinate through dependency $\textit{clit} - a - \textit{obj}$ and also another word subordinate through dependency $\textit{clit} - 3d - \textit{obj}$. At the same time, the accusative case clitic *la* (*it[fem.]*) has potential $\sphericalangle \textit{clit} - a - \textit{obj}$ and the dative case clitic *lui* (*to him*) has potential $\sphericalangle \textit{clit} - 3d - \textit{obj}$. The proper pairing of these dual valencies with those of the participle defines two non-projective dependencies between the participle and its cliticized complements.

The expression

$$t = [l_m \setminus \dots l_1 \setminus h / r_1 \dots / r_n]^P$$

($m, n \geq 0$) is called a **type** of a word w if:

(i) $l_m \setminus \dots l_1 \setminus$ and $/r_1 \dots /r_n$ encode respectively left and right projective dependency valencies of w ,

(ii) h is a governor (no-governor) valency and

(iii) P , the potential of w , encodes its valencies of non-projective dependencies.

l_m, \dots, l_1 are **left subtypes** of t , r_1, \dots, r_n are its **right subtypes** and h is its **head subtype**.

Below we use CDG with non-empty head subtypes. When a type $[\alpha \setminus d / \beta]^P$ has a negative valency in its potential P , say $P = \sphericalangle v P'$, the word w with this type has two governors: one through v , the other through d . In such cases we use special head subtypes $d = \#(A)$, called **anchors**, to express the adjacency of w to a **host** word w_0 . The anchor dependencies are displayed below the sentence for a better readability. E.g., the DS in Fig. 2 is defined by the following assignment of types to words: $\textit{elle} \mapsto [\textit{pred}]$, $\textit{la} \mapsto [\#(\sphericalangle \textit{clit} - a - \textit{obj})]^{\sphericalangle \textit{clit} - a - \textit{obj}}$, $\textit{lui} \mapsto [\#(\sphericalangle \textit{clit} - 3d - \textit{obj})]^{\sphericalangle \textit{clit} - 3d - \textit{obj}}$, $\textit{donnée} \mapsto [\textit{aux} - a - d]^{\sphericalangle \textit{clit} - a - \textit{obj} \setminus \textit{clit} - 3d - \textit{obj}}$, $a \mapsto [\#(\sphericalangle \textit{clit} - 3d - \textit{obj}) \setminus \#(\sphericalangle \textit{clit} - a - \textit{obj}) \setminus \textit{pred} \setminus S / \textit{aux} - a - d]$.

Due to the anchor subtypes $\#(\sphericalangle \textit{clit} - 3d - \textit{obj})$,

$\#(\sphericalangle \textit{clit} - a - \textit{obj})$ in the type of the auxiliary verb a (*has*), it serves as the host verb for both clitics and also defines their precedence order. Derivability of DS in CDG is formalized through the following calculus³ (with C being a dependency, H being a dependency or an anchor and V being a polarized valency):

$$\mathbf{L}^1. H^{P_1} [H \setminus \beta]^{P_2} \vdash [\beta]^{P_1 P_2}$$

$$\mathbf{I}^1. C^{P_1} [C^* \setminus \beta]^{P_2} \vdash [C^* \setminus \beta]^{P_1 P_2}$$

$$\mathbf{\Omega}^1. [C^* \setminus \beta]^P \vdash [\beta]^P$$

$\mathbf{D}^1. \alpha^{P_1 (\sphericalangle V)^P (\sphericalangle V)^{P_2}} \vdash \alpha^{P_1 P P_2}$, if the potential $(\sphericalangle V)^P (\sphericalangle V)$ satisfies the following pairing rule **FA** (first available):⁴

FA : P has no occurrences of $\sphericalangle V, \sphericalangle V$.

\mathbf{L}^1 is the classical elimination rule. Eliminating the argument subtype $H \neq \#(\alpha)$ it constructs the (projective) dependency H and concatenates the potentials. $H = \#(\alpha)$ creates the **anchor dependency**. \mathbf{I}^1 derives $k > 0$ instances of C . $\mathbf{\Omega}^1$ serves for the case $k = 0$. \mathbf{D}^1 creates **non-projective dependencies**. It pairs and eliminates dual valencies with name V satisfying the rule FA to create the non-projective dependency V .

In Fig. 4 we show a proof of correctness of DS in Fig. 2 with respect to the type assignment shown above.

A CDG G is defined by its dictionary W and its **lexicon** λ , an assignment of finite sets of types to words in W . G defines a DS D of a sentence $x = w_1 \dots w_n$ and x is generated by G (denoted $D \in \Delta(G)$ and $x \in L(G)$) if it is possible to assign through λ a type t_i to every word w_i so that the obtained type string $t_1 \dots t_n$ were reducible to the axiom S . $L(G)$ is the **language** and $\Delta(G)$ is the **structure language** generated by G .

²Their order is irrelevant (so one may choose a standard lexicographic order).

³We show left-oriented rules. The right-oriented rules are symmetrical.

⁴Cf. a different pairing rule in (Dikovskiy, 2007).

$Vt(F = fin, C = a) \mapsto \{pred?, neg?, vocative?, \#(\surd explet)?, circ^*\}[\{lpar?, \#(\surd coref)?, \#(\surd select)?\} \setminus (\#(\surd compos-neg)|\#(\surd restr-neg)|\#(\surd compos-neg)|\#(\surd restr-neg))?\setminus interrog?\emphat?\setminus S/(\#(\surd fs)|\#(\surd qu)|\#(\surd xl))/coordv^*/(a-obj|claus|pre-inf|inf)?/\{rpar?, \#(\surd modif)?, \#(\surd attr)?, \#(\surd appos)?, \#(\surd dist-rel)?, \#(\surd aggr)?\}$

Figure 5: A RTE for transitive French verbs

3 Extended CDG

CDG is a theoretical model not adapted for wide coverage grammars. The main problem with wide coverage is the excessive sharing of subtypes in types. For lexicons running to hundreds of thousands of lexical units it results in a combinatorial explosion of spurious ambiguity and in a significant parsing slowdown. Wide coverage grammars face many hard problems, e.g. those of compound lexical entries including complex numbers, compound terms, proper names, etc. and also that of flexible precedence order. An extension of CDG well adapted for wide coverage grammars is proposed in (Dikovsky, 2009).

The extended CDG use classes of words in the place of words and use restricted regular expressions defining sets of types in the place of types. I.e., the dictionary W is covered by classes: $W = \bigcup_{i \in I} C_i$ and the lexicon λ assigns

sets of regular expressions to classes. At that:

- all words in a class C share the types defined by the expressions assigned to C ,
- every word has all types of the classes to which it belongs.

The **regular type expressions (RTE)** we describe below are flat (i.e. bounded depth). In these expressions, C, C_i are dependency names or anchors, B is a **primitive type**, i.e. a dependency name, or an anchor or an iterated or optional type, and H is a choice.

Choice: $(C_1 | \dots | C_k)$; $(C) =_{df} C$

Optional choice: $(C_1 | \dots | C_k)?$; $(C)? =_{df} C?$

Iteration: $(C_1 | \dots | C_k)^*$; $(C)^* =_{df} C^*$

Dispersed subtypes expressing flexible order.

Left: $[\{\alpha_1, B, \alpha_2\} \setminus \alpha \setminus H / \beta]^P$

Right: $[\alpha \setminus H / \beta / \{\alpha_1, B, \alpha_2\}]^P$

Two-way: $\{\alpha_1, B, \alpha_2\}[\alpha \setminus H / \beta]^P$

Here is a fragment of the extended calculus:

1. Choice rules:

LC¹. $C^{P_1}[(\alpha_1 | C | \alpha_2) \setminus \beta]^{P_2} \vdash [\beta]^{P_1 P_2}$

IC¹. $C^{P_1}[(\alpha_1 | C | \alpha_2)^* \setminus \beta]^{P_2} \vdash [(\alpha_1 | C | \alpha_2)^* \setminus \beta]^{P_1 P_2}$

ΩC¹. $[(\alpha_1 | C | \alpha_2)^* \setminus \beta]^P \vdash [\beta]^P$

(**DC¹** is as **D¹** in the CDG calculus).

2. Dispersed subtypes rules:

LD¹. $H^{P_1}[\{\alpha\} \setminus H \setminus \beta / \{\gamma\}]^{P_2} \vdash [\{\alpha\} \setminus \beta / \{\gamma\}]^{P_1 P_2}$

ID¹. $C^{P_1}[\{\alpha_1, C^*, \alpha_2\} \setminus \beta / \{\gamma\}]^{P_2} \vdash$
 $[\{\alpha_1, C^*, \alpha_2\} \setminus \beta / \{\gamma\}]^{P_1 P_2}$

ΩD¹. $[\{\alpha_1, C^*, \alpha_2\} \setminus \beta / \{\gamma\}]^P \vdash [\{\alpha_1, \alpha_2\} \setminus \beta / \{\gamma\}]^P$

(**DD¹** as **D¹** in the CDG calculus).

E.g., the rule **ID¹** intuitively says that the dispersed iterated subordinates through C may be found in any position at the left of the governor with type $[\{\alpha_1, C^*, \alpha_2\} \setminus \beta / \{\gamma\}]^{P_2}$.

Fig. 5 shows an example of one of RTE assigned to the class $Vt(F = fin, C = a)$ of French transitive verbs in finite forms. It defines the simplest case where the complement is neither fronted nor cliticized. E.g., it states that the subject (subordinate through *pred*) may occur at the left or at the right of the verb, whereas the (exactly defined) position of the direct object (subordinate through *a-obj*) is at its right, and in the same position may be found a subordinate clause and a prepositional or preposition-less infinitive phrase.

The RTE and the classes do not extend the expressive power of CDG. At the same time, they dramatically reduce the grammar size. Sure, the unfolding of an extended CDG may exponentially blow up its size. However, due to the extended type calculus the polynomial time parsing algorithm of (Dekhtyar and Dikovsky, 2008) can be adapted to parse them directly, without unfolding. So the RTE are well adapted for large scale grammars. But still more, they are also ment for incremental bootstrapping of extended CDG from DS.

4 Structural Bootstrapping

In (Béchet et al., 2010; Béchet et al., 2011), it is proved that, in contrast to the constituent-structure grammars, even the projective CDG assigning one type per word cannot be learned from the DS they generate. This means that CDG cannot be automatically computed from dependency treebanks. The reason is that they express repeatable dependencies through iteration (and not through recursion). In these



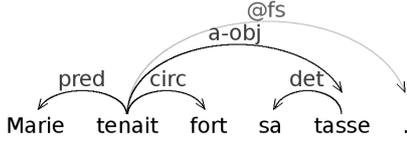


Figure 6: DS of (s_1)

papers are also defined and proved algorithms which learn from DS some subclasses of extended CDG, under reasonable conditions on the use of iteration. These partial solutions are still far from being practical. Below we present an intuitive heuristic method of construction of extended CDG from DS.

This method, we call **structural bootstrapping**, consists in that an extended CDG is incrementally constructed from a sample of DS, element by element. Ideally, the sample should be representative with respect to the surface syntax of the language. We suppose that the extended CDG is defined as $G = (W, \mathbf{D}, \mathbf{V}, S, \lambda)$, where W is its dictionary with a classification $W = \bigcup_{i \in I} C_i$, \mathbf{D} is the set of dependency names, \mathbf{V} is the set of valency names, S is the axiom and λ is the lexicon.

The method is based on a genericity partial order $(\text{PO}) \preceq$ on extended CDG, compatible with the inclusion of DS-languages: $G \preceq G' \Rightarrow \Delta(G) \subseteq \Delta(G')$. \preceq is the closure by reflexivity, transitivity and by type construction of the following basic PO (below t is a subtype, X is a list of alternatives and $(t) =_{df} t$):

1. $t \lesssim (t|X)$
2. $(t|X) \lesssim (t|X)?$
3. $(t|X)? \lesssim (t|X)^*$
4. $\{\gamma\}[\{\gamma_1\} \setminus t \setminus \beta]^P \lesssim \{\gamma\}[\{t, \gamma_1\} \setminus \beta]^P$
5. $\{\gamma\}[\{t, \gamma_1\} \setminus \beta]^P \lesssim \{t, \gamma\}[\{\gamma_1\} \setminus \beta]^P$
(similar for right subtypes)
6. $\{\gamma\}[\alpha / \{t, \gamma_1\}]^P \lesssim \{t, \gamma\}[\alpha / \{\gamma_1\}]^P$.

Basically, the Structural Bootstrapping Method consists in extracting from the sample DS the vicinities of words and in merging them into minimally generalized RTE of the preceding grammar. By **vicinity** of a word w in a DS D we mean the maximal subgraph $V(w, D)$ of D with the nodes $\{w, w_1, \dots, w_m\}$, w_1, \dots, w_m being the subordinates of w in D . Here is a schematic description of the method.

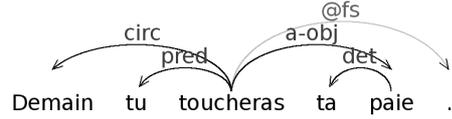


Figure 7: DS of (s_2)

Structural Bootstrapping Method:

Input: Extended CDG G_{in} ;

DS D_x of a sentence x //next DS.

Output: Extended CDG G_{out} generating D_x .

let $G_{in} = (W, \mathbf{C}, \mathbf{V}, S, \lambda)$ where $W = \bigcup_{i \in I} C_i$;

if $(D_x \in \Delta(G_{in}))$

then $G_{out} = G_{in}$

else

for every word $w \in x$

if $(w \in W)$

then select a class C such that $w \in C$;

else select a class C and add w to C

end;

find the vicinity $V(w, D)$;

if $(V(w, D)$ is generated by a RTE

$t \in \lambda(C)$)

then $\lambda'(C) = \lambda(C)$

else select RTE $t \in \lambda(C)$;

find *minimal* RTE $t' \succ t$

generating $V(w, D)$;

set $\lambda'(C) = (\lambda(C) - \{t\}) \cup \{t'\}$,

$\lambda'(C_1) = \lambda(C_1)$ for every $C_1 \neq C$

end

until $D_x \in \Delta((W, \mathbf{C}, \mathbf{V}, S, \lambda'))$

end;

return $G_{out} = (W, \mathbf{C}, \mathbf{V}, S, \lambda')$

Let us see how may evolve RTE of transitive verbs. Suppose that the class $Vt(F = fin, C = a)$ contains the verbs *tenait* (*took*), *toucheras* (*will get*, when applied to wages) and *mettrait* (*might put*). This is how the Structural Bootstrapping Method might change this class when applied to the following sample of sentences:

(s_1) *Marie tenait fort sa tasse.* (*Mary held tight her cup.*)

(s_2) *Demain tu toucheras ta paie.* (*Tomorrow you will get your wage.*)

(s_3) *Où mettrait-elle la clé?* (*Where might she put the key?*)

From the DS of (s_1) in Fig. 6 we have:

$Vt(F = fin, C = a) \mapsto$

$[pred \setminus S / @fs / a-obj / circ].$

The DS of (s_2) in Fig. 7 induces the following generalization:



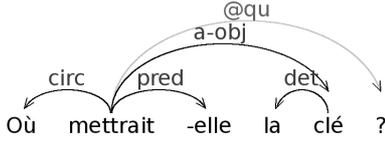


Figure 8: DS of (s_3)

$$Vt(F = fin, C = a) \mapsto \{circ^*\}[pred \setminus S / @fs/a-obj].$$

Finally, from the DS of (s_3) in Fig. 8 we obtain a still more general RTE:

$$Vt(F = fin, C = a) \mapsto \{pred, circ^*\}[S / (@fs|@qu)/a-obj].$$

In practice, the RTE generalization effected by the main operation:

find minimal RTE $t' \succ t$ generating $V(w, D)$ carries over to all other RTE $t'' \in \lambda(C)$ representing the same syntactic function as t in a compatible local context. E.g., the subject inversion as in (s_3) may also be applied to the RTE defining the coordinated clauses, but not to that defining the parenthetical clauses.

To see that this method is incremental, we should extend the partial order of generalization \preceq to the extended CDG:

1. $\tau \preceq \tau'$ for sets of RTE τ, τ' , if either:
 - (i) $\tau' = \tau \cup \{t\}$ for a RTE $t \notin \tau$ or
 - (ii) $\tau = \tau_0 \cup \{t'\}$ and $\tau' = \tau_0 \cup \{t''\}$ for a set of RTE τ_0 and some RTE t', t'' such that $t' \preceq t''$.
2. $\lambda \preceq \lambda'$ for two RTE assignments λ and λ' , if $\lambda(C') \preceq \lambda'(C')$ for a class C' and $\lambda(C) = \lambda'(C)$ for all classes $C \neq C'$.
3. \preceq_{gener} is the **genericity** PO which is the reflexive-transitive closure of the PO \preceq .
4. For CDG G_1 with lexicon λ and G_2 with lexicon λ' , $G_1 \preceq_{gener} G_2$ if $\lambda \preceq \lambda'$.

Now, it is not difficult to see the incrementality of this method in the sense that, if $G_1 \preceq_{gener} G_2$, then $\Delta(G_1) \subseteq \Delta(G_2)$.

Application of the Structural Bootstrapping Method in practice needs several resources. First of all, being applied directly as it is defined above, the method will always give grammars with a lexicon limited to that of the sample of representative sentences. So one should choose a morpho-syntactically annotated dictionary of the language (**MS-dictionary**) and to integrate it into the grammar establishing a

correspondence between its categories and the grammar's classes. Besides this, it is needed an efficient parser complete with respect to the class of all extended CDG.

5 Bootstrapping of a Wide Coverage CDG of French

The Structural Bootstrapping Method was applied to develop a wide coverage extended CDG of French. Its kernel part (Version 1) was bootstrapped from about 400 French sentences during half a year. In this phase, the method was applied completely incrementally. Then, after two months' long joint work with two colleagues, this grammar was integrated with the freely available MS-dictionary of French Lefff 3.0 (Sagot, 2010) containing 536,375 entries corresponding to 110,477 lemmas. The transition to this integrated Version 2 was non-monotone because the initial lexical classification was to be adapted to Lefff 3.0 and also because of a reorganization of prepositional dependencies. In Version 1 we more or less followed the so called "pronominal approach" (see (van den Eynde and Mertens, 2003)), but finally we have passed to a system of pronominal and prepositional dependencies based on the case of pronouns. The Version 2 incrementally evolved to Version 3 into which were introduced various more peripheral "small syntax" constructions extracted from DS of about 200 more French sentences. The last two non-monotone updates of the grammar gave the Versions 3.1, 3.2. They were due to a reorganization of verbal RTE, leading to a simple and symmetrical system of negation dependencies and of parenthetical clauses. Basically, the bootstrapping process has stabilized already on Version 2. Till then the grammar keeps the main body of its RTE.

Version 3.2 of the CDG of French covers the major part of French syntax including:

- negation, the main binary negation: *ne...pas* | *jamais* | *plus*, ... and the ternary restrictive negation: *ne...que* as in *Eve n'a donné a Adam qu'une pomme* (Eve gave to Adam only one apple);
- reflexives and clitics: *Les loups ne se dévorent pas entre eux* (The wolfs do not eat up one another), see also the DS in Fig. 2;
- topicalized complements: *À ces départs s'ajoutent trois autres* (To these departures are added three more);
- clefting: *C' est très amicalement qu'Alain nous*

Examples	Classes			Regular Expressions	Dependencies	
	total	verbal	nominal		projective	non-projective
~ 600	185	46	7	~ 3120	84(9 <i>par</i>)	20(3 <i>par</i>)

(where $n(m\ par)$ means n of which m are parametrized)
 Tab. 1. Parameters of the CDG for French constructed by bootstrapping

a reçu (It is very friendly that Alain has received us);

- subordinate and relative clauses: *Maintenant, tous les soirs, quand il l'avait ramenée chez elle, il fallait qu'il entrât* (Now, every evening, when he accompanied her home, he was obliged to enter);

- interrogative clauses, order inversion: *Qui cela aurait – il pu être?* (*Who this would it be?);

- light verbs, e.g. *Le laisser faire mal à ma soeur était ma première erreur* (To let him cause damage to my sister was my first error);

- partial extraction from a complement: *Il m'en reste une très facile* (I have one [*fem.*] more resting, a very simple one);

- comparatives: *Il est deux fois plus grand qu'elle* (He is twice as great as she);

- vocatives and co-reference: *Ce truc, restons – en là, Adam!* (This matter, let us let it alone, Adam!);

- expletives: *Un voleur, de temps en temps, ça se repose* (A thief, from time to time, it takes a rest);

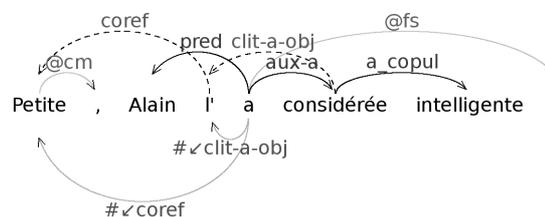
- aggregation: *Adam en a six ou sept rouges* (Adam has six or seven of them red);

- coordination with ellipsis: *J'ai reçu cette notice, et lui non.* (I have received this notice, and he not);

- extracted post-position modifiers: *Le gros chat tigré guette la souris, immobile et silencieux* (The fat stripy cat watched for a mouse, immovable and silent).

Table 1 shows some parameters of this grammar. It has 185 classes 46 of which are verbal, 7 are nominal, 9 are adjectival, 14 are adverbial and in the rest there are the multiple classes of prepositions, pronouns, numerals, determiners, conjunctions, particles, collocations and punctuation markers. The grammar uses 104 dependencies 84 of which are projective and 20 are non-projective. In fact, their number is greater because many of them are parametrized. E.g., there are 6 non-projective clitic dependencies $\surd\textit{clit-C-obj}$, in which C is the person-case parameter (cf. $\surd\textit{clit-3d-obj}$). Let us see the grammar in more detail.

Main principles. 1. This grammar is intended for analysis (not for generation) of correct sentences. So only the oppositions distinguishing between the dependency relations and the order are taken into account. E.g.,



(Small [*fem.*], Alain considered her intelligent [*fem.*])

Figure 9: Consecutive discontinuities

the agreement in number and in gender do not count, whereas the agreement in person is partially used to define the order of clitics. At the same time, the principle of minimality of the set of oppositions (see (Mel'čuk, 1988; Mel'čuk and Iordanskaja, 2000)) is abandoned in favour of a better distributed dependencies' system and lexicon classification. E.g., in the sentences like *Petite, Alain la considérée intelligente* (see Fig. 9) is used the (rather frequent) coreference dependency *coref* and not the minimally opposed, but rare dependency *object – copredicative* (from *considérée* to *Petite*) used in (Mel'čuk and Iordanskaja, 2000).

2. The grammar is rather intended for the development of French dependency treebanks, so the completeness criterion is prevailing over those of lower ambiguity and of more efficient parsing.

3. Basically, the grammar respects the fundamental principle of the dependency grammars (Kunze property): “words subordinate through the same dependency and belonging to the same grammatical category are substitutable”⁵, but in the place of grammatical categories are considered the lexicon classes.

4. To reduce the ambiguity, a number of values are propagated through dependencies. For instance, some dependencies are parametrized by case. In French, only prepositions and pronouns mark for case. So we define the case

⁵See (Mel'čuk, 1988) and (Mel'čuk and Iordanskaja, 2000) for a weaker version.

VERBAL DEPENDENCIES			
Group	Governor (<i>G</i>)	Subordinate (<i>D</i>)	Relation
<i>PRED</i>	main verb	subject	predicative
<i>AUX</i>	auxiliary verb	past participle	auxiliary
<i>COPUL</i>	main verb	noun / adjective / circumstantial	copular
<i>OBJ</i>	verb / noun / adjective	complement	objectival
<i>AGENT</i>	past participle	preposition (e.g. <i>par</i>)	agentive
<i>CLIT</i>	verb	pre-position clitic	clitic
<i>NEG</i>	main verb	<i>ne</i>	negative
<i>NEG</i>	<i>ne</i>	<i>pas, plus, etc.</i>	composite negative
<i>NEG</i>	<i>ne</i>	restrictive <i>que</i>	restrictive negative
<i>CIRC</i>	verb	e.g., adverbs	circumstantial
<i>COORD</i>	verb	verb	verb coordination
<i>CLAUS</i>	rel. pronoun / verb	verb	clausal
NOMINAL DEPENDENCIES			
<i>DET</i>	noun / adjective	determiner	determinative
<i>MODIF</i>	noun	adjective / past participle	modifier
<i>ATTR</i>	noun	preposition	attributive
<i>QUANT</i>	noun	numeral	quantitative
<i>REL</i>	noun	pronoun	relative
<i>COMPAR</i>	noun	junction / adj.	comparative
<i>COREF</i>	pronoun	noun	co-referential
<i>RESTRICT</i>	noun	<i>que</i> / adv.	restrictive
<i>APPOS</i>	noun	noun / adj.	appositive

Tab. 2. A sample of verbal and nominal dependency relations

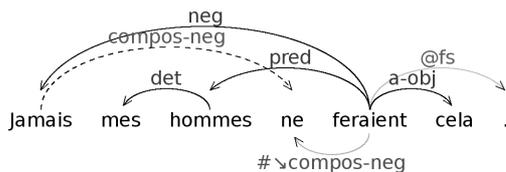
indirectly: a noun has case *C* if it can be replaced (in the same position) by a pronoun in case *C* without affecting the syntactic well-formedness. We distinguish five cases: *a* (accusative), *d* (dative), *g* (genitive), *l* (locative) and *o* (oblique, that of non-cliticizable complements). Respectively, we parametrize the objective dependency by the case of the subordinate complement, e.g. *a-obj* (direct object), *d-obj* (indirect object), etc. Moreover, when a word (e.g. an auxiliary verb) *w* serves as the host word for a pronoun in case *C*₁, the dependency from *w* to a subordinate word (e.g. a participle) is parametrized by *C*₁. The propagated parameters are used to prohibit to the subordinate to have the same case complements in their standard position (e.g. being subordinate through *aux-a-d*, the participle *donnée* in Fig. 2 cannot have complements).

Lexicon classes. As explained above, every class is defined, on the one hand, by a list of forms belonging to the class (the correspondence between the CDG classes and the Leff categories is external with respect to the grammar), and on the other hand, by a set of RTE. Each RTE defines a set of CDG types possible for the lexical units in the list. In all, the French CDG, version 3.2 has about 3120 RTE. E.g., the RTE in Fig 5 is one of 32 RTE defining the class $Vt(F = fin, C = a)$.

The grammar's lexicon includes four fam-

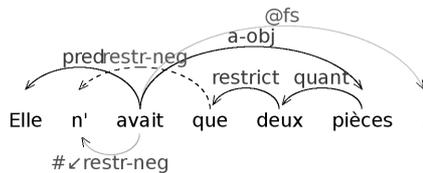
ilies of verbal classes: auxiliary verbs *Vaux* (*avoir, être*), copulas *Vcopul* (e.g. *être, devenir*), light verbs *Vlight* (e.g. *faire, laisser*) and significant verbs *V*. Every family has four subfamilies corresponding to verb forms: $F = fin$ (finite formes), $F = pz, T = pres$ (present participle), $F = pz, T = past$ (past participle) and $F = inf$ (infinitive). Finally, the significant verbs are classified by their **government patterns**, i.e. by the number of their complements and by the complements' case (e.g. $V2t(F = fin, C1 = a, C2 = d)$). Among the nominal and adjectival classes there are also those with genitive and dative arguments. The prepositional classes are opposed by the syntactic function of the prepositional phrase (e.g., *complement* (infinitival or not), *circumstantial, attribute*), and by case/role (e.g. *agent*). Finally, there is a complex class *UT* of unknown lexical units.

Inventory of dependencies. The main advantage of the dependency syntax is that it is very close to a semantic representation of sentences. At that, dependency relations are numerous. The dependency relations used in the grammar are broken down into 39 groups: 15 verbal, 14 nominal, 4 prepositional and several others: of aggregation, expletive, emphatic, junction/punctuation and deictic. Some of them are shown in Table 2.



(Never my people would do this)

Figure 10: Negation in pre-position



(It [fem.] had only two rooms)

Figure 11: Negation in post-position

Some dependency grammars and parsers flatten and distort DS because they cannot express non-projective dependencies. Such dependencies being not an obstacle for CDG, the grammar Version 3.2 uses numerous non-projective dependencies. Let us see the example of negative dependencies (group *NEG*).

The negation in French consists of two parts: the (main) **catégoriematic** part (*pas*, *plus*, *jamais*, *que*, *aucun* etc.) and the **syncatégoriematic** part *ne*. We distinguish between the **restrictive** verbal negation with the catégoriematic part *que*, *aucun*, etc. and the binary verbal negation with the catégoriematic part other than *que*, *aucun*, etc. because the latter is related through dependencies only with the negated verb, whereas the former is related not only with the verb, but also with one of its complements. For both kinds of negation, the catégoriematic part may be found in pre- and post-position with respect to the verb (cf. DS in Fig. 10 and 11).

Evaluation. The French CDG Version 3.2 was used to create an experimental dependency treebank (DTB). Actually, this DTB contains about 1500 DS. It was created within three months with the help of the toolkit CDG Lab (Alfared et al., 2011). The analyzed sentences originated from heterogeneous sources: French grammar (Grevisse, 1993), literary prose (E.Zola, M.Proust, La Rochefoucauld), scientific prose, periodical press (cor-

pus Le Monde (Abeillé et al., 2003)), blogs, publicity, spoken language. These sentences vary from very short and simple to extremely long and complex. Nearly 200 of them needed application of the bootstrapping procedure in order to complete the grammar (lexically or syntactically or both). 42.8% of DS in the constructed corpus are non-projective. Among the non-projective dependencies used in these DS the most frequent are not only the negative dependencies, but also the reflexive and the clitic dependencies, as well as some nominal non-projective dependencies (e.g. *coref* and *dist-rel*). Other non-projective dependencies are less frequent but are used in regular constructions, e.g. *C-obj*, *modif*, *attr* (of topicalized complements, modifiers and attributes), *expletive* (of parenthetical phrases) and many other.

Parser. The CDG of French is parsed with a special CDG-complete polynomial time symbolic parser rather adapted to the parallel development of the CDG of French and of DS corpora. It computes, after every grammar update, the scores of correctness of the grammar with respect to a DS corpus and also supports a semi-automatic analysis by consecutive approximations (see (Alfared et al., 2011) for more details). A higher-performing autonomous mixed stochastic-symbolic parser is under design.

Conclusion

The extended CGD prove to be well adapted for practical development of wide scope dependency grammars and of dependency treebanks. Due to their formalization through the extended type calculus, they allow to express voluminous sets of types using well-structured and succinct restricted regular type expressions, and at the same time are analyzed in a reasonable polynomial time. A specific Structural Bootstrapping Method based on a genericity order on RTE and supported by a set of appropriate and efficient tools allows to incrementally develop in a relatively short space of time large scale dependency grammars and dependency treebanks provably correct with respect to them.

References

- A. Abeillé, L. Clément, and F. Toussenet. 2003. Building a treebank for french. In A. Abeillé, editor, *Treebanks*.
- Ramadan Alfaref, Denis Béchet, and Alexander Dikovsky. 2011. “C DG Lab”: a Toolbox for Dependency Grammars and Dependency Treebanks Development. In *Proc. of the Int. Conf. on Dependency Linguistics (Depling’2011)*, Barcelona, Spain.
- Y. Bar-Hillel, H. Gaifman, and E. Shamir. 1960. On categorial and phrase structure grammars. *Bull. Res. Council Israel*, 9F:1–16.
- Denis Béchet, Alexander Dikovsky, Annie Foret, and Erwan Moreau. 2004. On learning discontinuous dependencies from positive data. In *Proc. of the 9th Intern. Conf. “Formal Grammar 2004” (FG 2004)*, pages 1–16, Nancy, France.
- Denis Béchet, Alexander Dikovsky, and Annie Foret. 2010. Two models of learning iterated dependencies. In *Proc. of the 15th Conference on Formal Grammar (FG 2010)*, LNCS, to appear, Copenhagen, Denmark. [online] http://www.angl.hu-berlin.de/FG10/fg10_list_of_papers.
- Denis Béchet, Alexander Dikovsky, and Annie Foret. 2011. On dispersed and choice iteration in incrementally learnable dependency types. In *Proc. of the 6th Int. Conf. “Logical Aspects of Computational Linguistics” (LACL’2011)*, LNAI 6736, pages 80–95.
- Norbert Bröker. 2000. Unordered and non-projective dependency grammars. *Traitement Automatique des Langues (TAL)*, 41(1):245–272.
- Ralf Debusmann, Denis Duchier, and Geert-Jan. M. Kruijff. 2001. Extensible dependency grammar: A new methodology. In *Proc. of the COLING 2004 Workshop on Recent Advances in Dependency Grammar*, Geneva.
- Michael Dekhtyar and Alexander Dikovsky. 2008. Generalized categorial dependency grammars. In *Trakhtenbrot/Festschrift*, LNCS 4800, pages 230–255. Springer.
- Michael Dekhtyar, Alexander Dikovsky, and Boris Karlov. 2010. Iterated dependencies and kleene iteration. In *Proc. of the 15th Conference on Formal Grammar (FG 2010)*, LNCS, to appear, Copenhagen, Denmark. [online] http://www.angl.hu-berlin.de/FG10/fg10_list_of_papers.
- Alexander Dikovsky. 2004. Dependencies as categories. In “Recent Advances in Dependency Grammars”. *COLING’04 Workshop*, pages 90–97.
- Alexander Dikovsky. 2007. Multimodal categorial dependency grammars. In *Proc. of the 12th Conference on Formal Grammar*, pages 1–12, Dublin, Ireland.
- Alexander Dikovsky. 2009. Towards wide coverage categorial dependency grammars. In *Proc. of the ESSLLI’2009 Workshop on Parsing with Categorial Grammars*. Book of Abstracts, Bordeaux, France.
- Haïm Gaifman. 1961. Dependency systems and phrase structure systems. Report p-2315, RAND Corp. Santa Monica (CA). Published in: *Information and Control*, 1965, v. 8, n 3, pp. 304–337.
- Maurice Grevisse. 1993. *Le bon usage. Grammaire française*. Duculot.
- Aravind K. Joshi, Vijay K. Shanker, and David J. Weir. 1991. The convergence of mildly context-sensitive grammar formalisms. In *Foundational issues in natural language processing*, pages 31–81, Cambridge, MA.
- Sylvain Kahane, Alexis Nasr, and Owen Rambow. 1998. Pseudo-projectivity : A polynomially parsable non-projective dependency grammar. In *Proc. COLING-ACL*, pages 646–652, Montreal.
- Boris N. Karlov. 2008. Normal forms and automata for categorial dependency grammars. *Vestnik Tverskogo Gosudarstvennogo Universiteta (Annals of Tver State University)*. Series: Applied Mathematics, 35 (95):23–43. (in Russ.).
- J. Lambek. 1961. On the calculus of syntactic types. In Roman Jakobson, editor, *Structure of languages and its mathematical aspects*, pages 166–178. American Mathematical Society, Providence RI.
- J. Lambek. 1999. Type grammars revisited. In Alain Lecomte, François Lamarche, and Guy Perrier, editors, *Logical aspects of computational linguistics: Second International Conference, LACL ’97, Nancy, France, September 22–24, 1997; selected papers*, volume 1582. Springer-Verlag.
- Hiroshi Maruyama. 1990. Structural disambiguation with constraint propagation. In *Proc. of 28th ACL Annual Meeting*, pages 31–38.
- I. Mel’čuk and L. Iordanskaja. 2000. The notion of surface-syntactic relation revisited (valence-controlled surface-syntactic relations in french). In L. L. Iomdin and L. P. Krysin, editors, *Slovo v tekste i v slovare. Sbornik statej k semidesjatiletiju Ju. D. Apresjana*, pages 391–433. *Jazyki russkoj kultury*, Moskva.
- I. Mel’čuk. 1988. *Dependency Syntax*. SUNY Press, Albany, NY.
- B. Sagot. 2010. The lefff, a freely available and large-coverage morphological and syntactic lexicon for french. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC’10)*.
- Vijay K. Shanker and David J. Weir. 1994. The equivalence of four extensions of context-free grammars. *Mathematical Systems Theory*, 27:511–545.
- D. Sleator and D. Temperly. 1993. Parsing English with a Link Grammar. In *Proc. IWPT’93*, pages 277–291.
- Mark Steedman. 1996. *Surface structure and interpretation*. MIT Press, Cambridge, Massachusetts.
- Karel van den Eynde and Piet Mertens. 2003. La valence: l’approche pronominale et son application au lexique verbal. *Journal of French Language Studies*, 13:63–104.

“CDG LAB”: a Toolbox for Dependency Grammars and Dependency Treebanks Development

Ramadan Alfared, Denis Béchet and Alexander Dikovsky

LINA CNRS – UMR 6241 – University of Nantes

Ramadan.Alfared@etu.univ-nantes.fr

Denis.Bechet@univ-nantes.fr

Alexandre.Dikovsky@univ-nantes.fr

Abstract

We present “CDG LAB”, a toolkit for development of dependency grammars and treebanks. It uses the Categorical Dependency Grammars (CDG) as a formal model of dependency grammars. CDG are very expressive. They generate unlimited dependency structures, are analyzed in polynomial time and are conservatively extendable by regular type expressions without loss of parsing efficiency. Due to these features, they are well adapted to definition of large scale grammars. CDG LAB supports the analysis of correctness of treebanks developed in parallel with evolving grammars.

1 Introduction

There are two main technologies of automatic syntactic analysis of natural language:

1. *grammatical parsing* i.e (symbolic or statistical or mixed) parsing of a hand-crafted grammar belonging to a family of formal grammars disposing of a general purpose parser;
2. *data-driven parsing*, i.e. parsing with statistical parsers trained over annotated data.

Both technologies need a large amount of expensive expert linguistic data. The hand-crafted wide coverage grammars are notoriously expensive and only very few of them have been successfully realized and applied to unrestricted material (cf. (Bouma et al., 2000; Riezler et al., 2002)). Besides this, they are prone to explosion of spurious ambiguity when parsed with general purpose parsers. On the other hand, training of statistical parsers needs voluminous high quality treebanks such as the Penn Treebank (Marcus et al., 1993). Training data of this size and quality are in fact as expensive as the hand-crafted grammars and also need a long-term hand work. Even if

the results obtained in the statistical parsing during the last fifteen years are very encouraging, their quality and adequacy depends on those of the hand-crafted annotated data. This is a vital issue for dependency grammars which suffer from the shortage of high quality training data. The several existing dependency treebanks (DTB) such as the Prague Dependency Treebank of Czech (Hajicova et al., 1998), the TIGER treebank of German (Brants and Hansen, 2002) or the Russian treebank (Boguslavsky et al., 2000) only partially solve the problem. First of all, they serve for particular languages. Secondly, even for these languages, the DTB use a particular inventory of dependency relations. At the same time, there is no consensus on such inventories. So the DTB are dependent on the choice of underlying syntactic theories, which makes their reuse problematic. The translation technologies (cf. (Hockenmaier and Steedman, 2007)) consisting in acquisition of dependency structures from high quality constituent structure treebanks also do not resolve the problem because, for technical reasons, they often flatten the genuine dependency structures and introduce into them multiple distortions. For all these reasons, there is a need in efficient and inexpensive methods and tools of development of wide coverage grammars and of training corpora.

Below we present “CDG LAB”, a toolkit supporting parallel development of wide coverage dependency grammars and of DTB. It uses Categorical Dependency Grammars (CDG) as a formal model of dependency grammars.

The CDG, a class of first-order type categorical grammars generating unlimited dependency structures (DS), were introduced in (Dikovsky, 2004). Since then, they were intensively studied (e.g., see (Dekhtyar and Dikovsky, 2004; Béchet et al., 2004; Dekhtyar and Dikovsky, 2008; Dekhtyar et al., 2010; Béchet et al., 2010)). CDG are very expressive. In particular, very simple

CDG generate such non-CF languages as $L^{(m)} = \{a_1^n a_2^n \dots a_m^n \mid n \geq 1\}$ for all $m > 0$ and $MIX = \{w \in \{a, b, c\}^+ \mid |w|_a = |w|_b = |w|_c\}$. They are equivalent to real time pushdown automata with independent counters (Karlof, 2008). Interesting sufficient conditions of learning CDG in the limit were found recently (Béchet et al., 2004; Béchet et al., 2010; Béchet et al., 2011).

CDG also have important advantages which make them a convenient and natural means of definition of wide coverage dependency grammars. First, they are completely lexicalized, as it is the case of all categorial, and more generally, type logical grammars (Bar-Hillel et al., 1960; Lambek, 1961; Lambek, 1999; Steedman, 1996). Second, the CDG types directly encode DS with repeatable and unlimited non-projective dependencies (see below). Third, they are parsed in a polynomial time (Dekhtyar and Dikovsky, 2004; Dekhtyar and Dikovsky, 2008). Fourth, an extension of CDG by regular type expressions (RTE) specially designed for large scale grammars is defined (Dikovsky, 2009; Dikovsky, 2011) and is implemented in the CDG parser presented below. Moreover, for this extension there is a supported by CDG LAB method of incremental bootstrapping of large scale grammars from dependency structures (Dikovsky, 2011).

The plan of this paper is as follows. Section 2 presents the basics of CDG and of their extension by RTE. Then the architecture and the main functionalities of CDG LAB are described in Section 3.

2 Categorial Dependency Grammars

CDG define projective DS (as in Fig. 1) i.e. DS in which dependencies do not cross, and also non-projective DS, as in Fig. 2, in which they may cross. In these graphs, the nodes correspond to the words of the sentence (their precedence order in the sentence is important) and the arcs represent the dependencies: named binary relations on words. Formally, a DS of a sentence x is a linearly ordered cycle-free graph with labelled arcs and the words of x as nodes. We consider connected DS with the root node. When in a DS D there is an arc $w_1 \xrightarrow{d} w_2$, we say that d is a dependency between w_1 and w_2 , w_1 is the governor and w_2 is subordinate to w_1 through d . E.g., *in* is subordinate to *was* in Fig. 1 and *donnée* governs *la* through *clit-a-obj* and *lui* through *clit-3d-obj*.

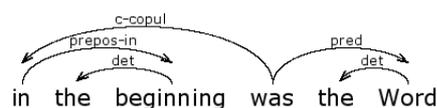


Figure 1: Projective DS

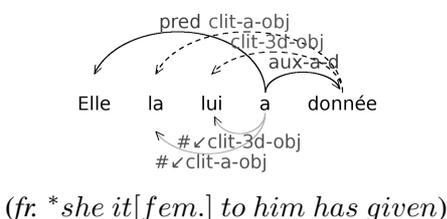


Figure 2: Non-projective DS

As all categorial grammars, the CDG are completely lexicalized and may be seen as assignments of types to words in a dictionary W . CDG types are expressions of the form

$$t = [l_1 \setminus l_2 \setminus \dots \setminus H / \dots / r_2 / r_1]^P.$$

A type assigned to a word $w \in W$ defines its dependencies in a rather straightforward way: its subtypes $H, l_1, l_2, \dots, \dots, r_2, r_1$ represent the claims for w to be related to other words through projective dependencies and P , called potential of t , defines all non-projective dependencies of w . The head subtype H , claims that w should be subordinate to a word through dependency H . When w should be the root of a DS, $H = S$ (S is a special symbol called axiom). The left subtypes l_1, l_2, \dots define the left projective dependencies of w (i.e. the dependencies through which w governs the words occurring in the sentence on its left). The right subtypes \dots, r_2, r_1 define the right projective dependencies of w . For instance, the projective DS in Fig. 1 is uniquely defined by the type assignment:

$in \mapsto [c-copul/prepos-in]$, $the \mapsto [det]$,
 $Word \mapsto [det \setminus pred]$, $beginning \mapsto [det \setminus prepos-in]$, $was \mapsto [c-copul \setminus S / pred]$.

Left and right subtypes may also be iterated. The iterated subtypes define repeatable dependencies. E.g., $l_i = d^*$ means that w may have on it left 0, 1, 2, ... occurrences of words subordinate to it through dependency d . We also use optional subtypes $l_i = d?$. Assignment of type $[d? \setminus \alpha]$ is equivalent to assignment of $[d \setminus \alpha]$ and $[\alpha]$. E.g., the DS in Fig. 3 is defined by the type assignment:
 $she \mapsto [pred]$, $was \mapsto [pred \setminus S / a-copul^*]$,

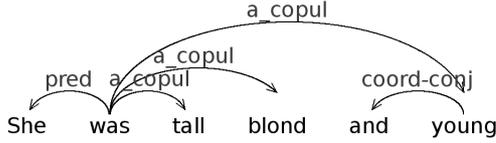


Figure 3: Iterated dependency

tall, blond, young \mapsto $[coord-conj? \setminus a_copul]$,
and \mapsto $[coord-conj]$.

By the way, the coordination scheme used in this analysis is more general than the traditional one, where the coordinated phrase is subordinate to the conjunction. The new scheme applies to the traditional recursive definition of coordination as well as to the iterative one, as in this sentence.

The potential P is the concatenation of so called polarized valencies of w . The polarized valencies are of four kinds: $\swarrow v$, $\searrow v$ (negative) and $\nwarrow v$, $\nearrow v$ positive. E.g., if a word w has valency $\swarrow v$, this intuitively means that its governor through dependency v must occur **somewhere** on the right. Two polarized valencies with the same valency name v and orientation, but with the opposite signs are dual. Together they define non-projective dependency v . The order of polarized valencies in the potential P is irrelevant (so one may choose a standard lexicographic order). For instance, in the DS in Fig. 2, the potential $\nwarrow clit-a-obj \nwarrow clit-3d-obj$ of the participle *donnée* means that it needs somewhere on its left a word subordinate through dependency $clit-a-obj$ and also another word subordinate through dependency $clit-3d-obj$. At the same time, the accusative case clitic *la* (*it[fem.]*) has potential $\swarrow clit-a-obj$ and the dative case clitic *lui* (*to him*) has potential $\swarrow clit-3d-obj$. The proper pairing of their dual valencies with those of the participle defines two non-projective dependencies between the participle and its cliticized complements.

We use CDG with non-empty head subtypes. When a type $[\alpha \setminus d / \beta]^P$ has a negative valency in its potential P , say $P = \swarrow v P'$, the word w with this type has two governors: one through v , the other through d . For such cases we use special head subtypes $d = \#(A)$, called **anchors**, to express the adjacency of w to a host word w_0 . The anchor dependencies are displayed below the sentence for a better readability. E.g., the DS in Fig. 2 is defined by the following types assignment:

elle \mapsto $[pred]$,
la \mapsto $[\#(\swarrow clit-a-obj)]^{\swarrow clit-a-obj}$,
lui \mapsto $[\#(\swarrow clit-3d-obj)]^{\swarrow clit-3d-obj}$,
donnée \mapsto $[aux-a-d]^{\nwarrow clit-a-obj \nwarrow clit-3d-obj}$,
a \mapsto $[\#(\swarrow clit-3d-obj) \setminus \#(\swarrow clit-a-obj) \setminus pred \setminus S/aux-a-d]$.

Due to the anchor subtypes $\#(\swarrow clit-3d-obj)$, $\#(\swarrow clit-a-obj)$ in the type of the auxiliary verb *a* (*has*), it serves as the host verb for both clitics and also defines their precedence order.

Derivability of DS in CDG is formalized through the following calculus¹ (with C being a dependency, H being a dependency or an anchor and V being a polarized valency):

- \mathbf{L}^1 . $H^{P_1} [H \setminus \beta]^{P_2} \vdash [\beta]^{P_1 P_2}$
- \mathbf{I}^1 . $C^{P_1} [C^* \setminus \beta]^{P_2} \vdash [C^* \setminus \beta]^{P_1 P_2}$
- $\mathbf{\Omega}^1$. $[C^* \setminus \beta]^P \vdash [\beta]^P$
- \mathbf{D}^1 . $\alpha^{P_1 (\swarrow V) P (\nwarrow V) P_2} \vdash \alpha^{P_1 P P_2}$, if the potential $(\swarrow V) P (\nwarrow V)$ satisfies the following pairing rule **FA** (*first available*):²

FA : P has no occurrences of $\swarrow V, \nwarrow V$.

\mathbf{L}^1 is the classical elimination rule. Eliminating the argument subtype $H \neq \#(\alpha)$ it constructs the (projective) dependency H and concatenates the potentials. $H = \#(\alpha)$ creates the anchor dependency. \mathbf{I}^1 derives $k > 0$ instances of C . $\mathbf{\Omega}^1$ serves for the case $k = 0$. \mathbf{D}^1 creates non-projective dependencies. It pairs and eliminates dual valencies with name V satisfying the rule **FA** to create the non-projective dependency V .

Let us see how DS in Fig. 2 can be derived using the type assignment shown above.

First, we may eliminate the left anchor subtype $\#(\swarrow clit-3d-obj)$ in the type of the auxiliary verb *a* using the type of the clitic *lui*. As a result, we generate the anchor dependency $\#(\swarrow clit-3d-obj)$ from *a* to *lui* and the derived type of the string *lui a* becomes

$[\#(\swarrow clit-a-obj) \setminus pred \setminus S/aux-a-d]^{\swarrow clit-3d-obj}$.

In this type, we may eliminate the anchor subtype $\#(\swarrow clit-a-obj)$ using the type of the clitic *la*. This will generate the anchor dependency $\#(\swarrow clit-a-obj)$ from *a* to *la*. The derived type of the sequence *la lui a* is $[pred \setminus S/aux-a-d]^{\swarrow clit-a-obj \swarrow clit-3d-obj}$. Now we may eliminate the left subtype *pred* of the derived type using the type of the subject *elle*. This generates the projective dependency from *a* to *elle* and

¹We show left-oriented rules. The right-oriented rules are symmetrical.

²Cf. a different pairing rule mentioned below.

ically reduce the grammar size. Due to the extended type calculus they can be parsed directly, without unfolding. In fact, the polynomial time parsing algorithm of (Dekhtyar and Dikovskiy, 2008) can be adapted to the extended CDG.

3 CDG LAB

CDG LAB is a kit of tools supporting parsing with extended CDG, development and maintainance of dependency treebanks (DTB) and development and test of large scale extended CDG. The core element of CDG LAB is the parser of the extended CDG implemented in Steel Bank Common Lisp. Recently was issued its version 3.1 (below we will call this parser `Parser-3.1`).

All input and output data of `Parser-3.1` are XML-structures. It may analyse sentences, text corpora and DTB either with an extended CDG integrated with an external morpho-syntactic dictionary (lexical base grammar) or only with the internal grammar lexicon (`text grammar`). For instance, for French is developed a large scale extended CDG (Dikovskiy, 2011). Its version 3.2 (called below “French Text CDG”) is integrated with the open MS-dictionary of French Lefff 3.0 (Sagot, 2010) containing 536,375 lexical units (LU). Lefff is kept in object-relational database PostgreSQL and a correspondence between the classes of the French Text CDG and the categories of Lefff is implemented through several hundreds of SQL queries. The integral grammar is called below “French LB CDG”.

Modes of Analysis. `Parser-3.1` is multi-purpose. It is used for semi-automatic analysis by consecutive approximations and also starting from head type/class selection, for estimation of compatibility of analyses with an updated grammar and for expert annotation of DS needed for this estimation, for automatic re-analysis of sentences when the annotation is changed and also for autonomous syntactic analysis. Respectively, the parser is used in different modes:

- analysis by head selection,
- analysis by approximations,
- DS analysis,
- autonomous analysis.

Fig. 4 shows a scheme of functioning of `Parser-3.1` in these modes. Sentences are introduced through the input form (see Fig. 5). Through this form, the User may set various parameters, e.g. the maximal parsing time, the maximal number of DS to re-

Figure 5: Query Form

turn, a graphical representation of DS, a language register (corresponding to specific choices of non-projective dependencies common to official documents or to scientific or literary prose, to periodicals or to the spoken language), etc. The input sentence is lexically analysed. Composite forms are decomposed into separate tokens, as in the case of *l'homme* (the man), which is segmented into three tokens: *l*, *'* and *homme*, for which are found in the lexicon all possible lemmas. In this example, the association is ambiguous: *l'* may be a clitic or a determiner. All possible variants of composite LU are detected (in particular, complex numbers and names recognized through regular expressions, multi-word LU, such as *à la* (of the kind), *à travers* (through) etc.) and unknown terms are identified. The transitions to and from head selection form are followed only in the mode of Analysis by head selection. Functioning in the mode of Analysis by approximations is iterative. It goes round result reporting and passes from annotation interface directly to lexicon loading. Other transitions are common for all modes. So we comment them in the head selection mode.

Analysis by Head Selection. In this mode a selection form is proposed (see Fig. 6), in which the User may select the proper composite LU (if and when several possibilities are detected) and for every LU, to select one of possible classes and one of possible dependency relation groups

Ève la lui a donnée.

Tokens: "Ève la lui a donnée ."
Loading the lexicon.

Select classes (at the top) and/or head types (at the bottom) and press Ok.

<input type="checkbox"/> N	<input type="checkbox"/> Det <input type="checkbox"/> N <input type="checkbox"/> PN	<input type="checkbox"/> PN	<input checked="" type="checkbox"/> Vaux <input type="checkbox"/> Vlight <input type="checkbox"/> Vt	<input type="checkbox"/> Adj <input type="checkbox"/> N <input checked="" type="checkbox"/> V2t <input type="checkbox"/> Vlight <input type="checkbox"/> Vt	<input type="checkbox"/> FullStop
<input checked="" type="radio"/> All classes	<input checked="" type="radio"/> All classes	<input checked="" type="radio"/> All classes	<input checked="" type="radio"/> All classes	<input checked="" type="radio"/> All classes	<input checked="" type="radio"/> All classes
<input type="radio"/> None	<input type="radio"/> None	<input type="radio"/> None	<input type="radio"/> None	<input type="radio"/> None	<input type="radio"/> None
Ève	la	lui	a	donnée	.
<input checked="" type="radio"/> All types	<input checked="" type="radio"/> All types	<input checked="" type="radio"/> All types	<input checked="" type="radio"/> All types	<input checked="" type="radio"/> All types	<input checked="" type="radio"/> All types
<input type="checkbox"/> COMPAR <input type="checkbox"/> COPUL <input type="checkbox"/> OBJ <input type="checkbox"/> PRED <input type="checkbox"/> PREPOS	<input type="checkbox"/> AGGR <input type="checkbox"/> APPOS <input type="checkbox"/> CLIT <input checked="" type="checkbox"/> #∕clit-a-obj <input type="checkbox"/> clit-a-obj <input type="checkbox"/> COMPAR <input type="checkbox"/> COORDN <input type="checkbox"/> COPUL <input type="checkbox"/> COREF <input type="checkbox"/> CORREL <input type="checkbox"/> DET <input type="checkbox"/> OBJ	<input type="checkbox"/> AGGR <input type="checkbox"/> APPOS <input type="checkbox"/> CLIT <input checked="" type="checkbox"/> #∕clit-3d-obj <input type="checkbox"/> clit-3d-obj <input type="checkbox"/> COPUL <input type="checkbox"/> COREF <input type="checkbox"/> OBJ <input type="checkbox"/> PRED <input type="checkbox"/> PREPOS <input type="checkbox"/> VOCATIVE	<input type="checkbox"/> CLAUS <input type="checkbox"/> COORDV <input checked="" type="checkbox"/> SENT	<input type="checkbox"/> AGGR <input type="checkbox"/> APPOS <input type="checkbox"/> AUX <input type="checkbox"/> aux <input type="checkbox"/> aux-A <input type="checkbox"/> aux-a <input type="checkbox"/> aux-a-A <input checked="" type="checkbox"/> aux-a-d <input type="checkbox"/> aux-a-g <input type="checkbox"/> aux-a-l <input type="checkbox"/> aux-a-o <input type="checkbox"/> aux-d	<input type="checkbox"/> PUNCT

Figure 6: Selection Form

(or of their elements). The selected dependency is nothing but the head subtype of the possible types of the LU. The latter selection is in fact decisive. It corresponds to the strong constraint that the LU is subordinate through the selected dependency (or limits the choice of such dependencies to the elements of the selected group).

This selection drastically limits the search-space. As it concerns the Parser-3.1 and the French LB CDG, it reduces the number of possible analyses by two-three orders of magnitude, i.e. in the place of a thousand of possible DS only about ten are found, most often differing between them in positions of repeatable dependencies (such as *modif* (modifier), *attr* (attribute) or *circ* (circumstantial)). E.g., in this example the genuine DS will be immediately found due to this selection (see Fig. 7).

Then is created the sentence's workspace (WS), an XML-structure representing the subgrammar corresponding to all detected LU to which are affected the classes, the types and the features' values compatible with the pre-selection. After this follow the steps common to all modes. First, all possible projective (and anchor) dependencies are computed and registered in the triangular matrix . This computation is done by an CKY-like al-

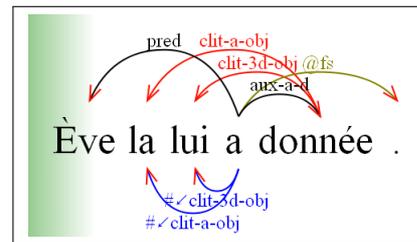


Figure 7: Resulting DS

gorithm adapted to extended CDG. Then all possible pairings of dual valencies providing non-projective dependencies are independently computed and registered in the resulting matrix (more precisely, in the submatrix in which the axiom S can be proved). It should be noted that the pairing principle may be chosen for a CDG and for a particular non-projective dependency. By default, this is the **FA** rule. But in rare exceptional situations, such as that of unlimited cross-serial dependencies in subordinate clauses in Dutch, one may use a different rule **FC** (first cross) defined in (Dikovskiy, 2007). This independence of computations of projective and non-projective dependencies is founded on the fundamental projection independence property of CDG proved for the rule **FA** in (Dekhtyar and Dikovskiy, 2004; Dekhtyar and Dikovskiy, 2008) and for rule **FC** in

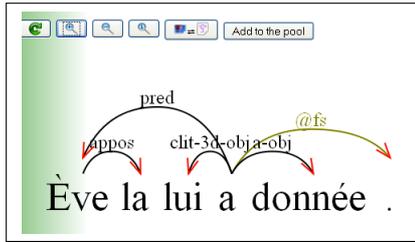


Figure 8: Incorrect DS

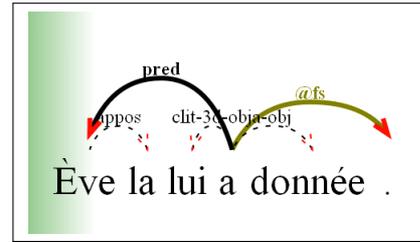


Figure 9: First annotated DS

(Dikovskiy, 2007). Till the end of this step the parsing algorithm is polynomial. The resulting triangular matrix is in fact a packed chart from which it is possible to enumerate all possible DS of the sentence. Given that the number of these DS may be exponential with respect to the size of the matrix, the next step is exponential in space in the worst case. In this step, the DS are generated from the matrix in a certain order and the feature values are assigned to LU in every generated DS. Finally, the parser generates the HTML report page, which includes various useful statistics. An XML structure representation of every DS including all necessary information, in particular the CDG classes and the feature values is also generated and saved to be used by other programs.

Analysis by Approximations. This important mode represents another User-guided strategy of parsing. It allows to find the needed DS starting from any obtained DS by consecutive approximations computed from User's annotations in the DS. There are three possible annotations of dependency relations: positive, negative and neutral. The positively annotated dependencies are those adequate. They will be kept during the whole sequence of approximations (if not discarded). The neutrally annotated dependencies are kept till they are compatible with the positively annotated ones. The negatively annotated dependencies are to be eliminated from the DS. When used in this mode, the Parser computes for every DS the total number of positively annotated dependencies and that of negatively annotated dependencies. The obtained DS are sorted first by the negative annotations' weight (the less negative annotations the better) then by the positive annotations' weight (the more positive annotations the better).

Suppose, that the approximations start from the (partially incorrect) DS of the sentence *Ève la lui a donnée*

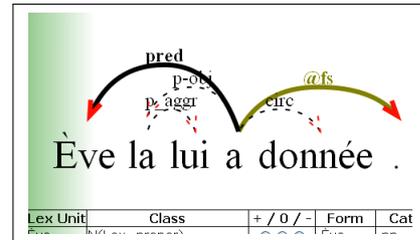


Figure 10: Next approximation

(*Eve it[fem.] to him has given*) shown in Fig. 8. There are only two correct dependencies in this DS: the predicative one: *pred* and the punctuation dependency *@fs*. We annotate both positively (this annotation being displayed by boldface arcs). The other three dependencies *appos*, *clit-3d-obj* and *a-obj* are erroneous. We annotate them as negative (which is displayed by broken arcs). So we obtain the first annotated DS shown in Fig. 9.

From this annotated DS, the Parser computes the next approximation shown in Fig. 10, which is also incorrect. It has the same two correct dependencies and three other incorrect: *p-obj*, *p-aggr* and *circ*. We annotate the three as negative, as it is shown in Fig. 10.

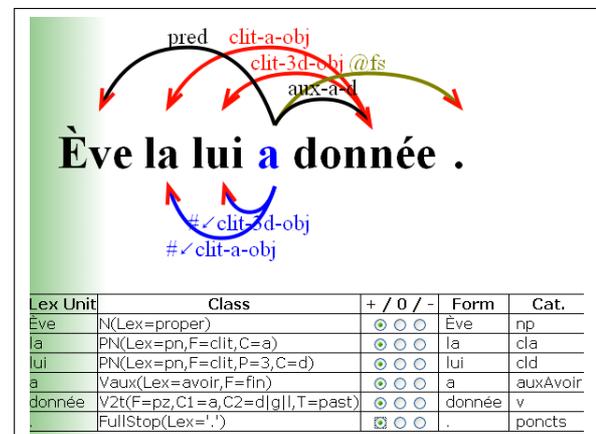


Figure 11: Final approximation

From this annotated DS, the parser finally computes the right one shown in Fig. 11. Not only this final approximation is correct, but it is also annotated as such. This difference is very important for the other mode of use of Parser-3.1, that of DS analysis.

DS, DTB and Grammar Analysis. In fact, Parser-3.1 considers every DS as annotated. The case where there are no annotations is considered as that with weight 0. Moreover, not only the dependencies, but also the LU may be annotated. The LU may have only two annotations: *positive* and *neutral*. Annotating a LU w positively is equivalent to positively annotate all dependencies in the sub-structure with the root w . This is seen in Fig 11, where the positive annotation of the root (displayed in a contrasting color) implies the positive annotations of dependencies (displayed in boldface). More than that, the class and the feature values assigned to every LU in DS may also be annotated as *positive*, *negative* or *neutral*. In the fragment of the class/feature table shown in Fig 11, one may see that not only the dependencies, but also the class/feature assignments for its LU are all annotated as positive. So this analysis is 100% correct. It is using this integral annotation weight, that Parser-3.1 evaluates the DS. Now, for two DS of the same sentence, it is possible to measure the difference of their weights. This simple measure turns out to be an efficient means of analysis of DTB and of the CDG used while their development. Every sentence processed by Parser-3.1 using French LB CDG obtains its *status*. The status includes the analysis result ('NO', when there are no parses, 'YES' otherwise) and for a parsable sentence, also the maximal number of returned DS and the difference (in percents) of annotation weights (of dependencies and of LU) between the best obtained DS and the one present in the DTB (if any) before this sentence processing. When the grammar is updated, the DS of sentences in the DTB become potentially irrelevant. For this case, Parser-3.1 has a special function of *re-parsing* of a DTB, which computes the difference between the DS before and after update, comparing their statuses. The User may choose between keeping or not the same head subtypes while re-parsing. Using this function, one may easily find all sentences to be revised.

Another way round, this test applies to the

Parse group 829...	0.409	DEFAULT	YES	≥ 1	100.0%
Parse group 830...	0.113	OK	YES	≥ 1	100.0%
Parse group 831...	0.136	OK	YES	≥ 1	100.0%
Parse group 832...	0.095	OK	YES	≥ 1	100.0%
Parse group 833...	0.205	OK	YES	≥ 1	100.0%
Parse group 834...	31.341	OK	YES	≥ 1	100.0%
Parse group 835...	0.226	OK	YES	2	75.0%
Parse group 836...	0.234	OK	YES	4	62.5%
Parse group 837...	0.207	OK	YES	1	0.0%
Parse group 838...	0.225	OK	YES	1	0.0%
Parse group 839...	0.488	OK	YES	≥ 5	77.3%

Figure 12: Re-parse results

grammar itself. The French Text CDG was created using the Structural Bootstrapping Method (Dikovskiy, 2011), a method specific to the extended CDG and consisting in an incremental transformation of DS of a sample of sentences σ into an extended CDG $G(\sigma)$ generating these sentences. The incrementality is interpreted in the strong sense: $\Delta(G(\sigma)) \subseteq \Delta(G(\sigma \cup \{s\}))$ for every new sentence s . So such transformations represent monotone grammar updates. Basically, the bootstrapping of French Text CDG was incremental in this sense, except three important revisions which were not. Taking in mind the size and the complexity of this grammar (it consists of more than 3120 RTE distributed between 185 lexicon classes, has 84 projective and 20 non-projective dependencies), it was a very hard task to find all sentences in the sample wrongly analysed using the updated grammar. Indeed, to find them, it was necessary to look through thousands of DS of hundreds of sentences in order to find linguistically adequate DS (the simple existence of a generated DS is of course not sufficient). The situation has completely changed after the implementation of Parser-3.1. Indeed, now all sentences in the sample are initially annotated. The procedure of re-parsing of the sample finds all inconsistencies. When a DS is compatible with the grammar before update (i.e. has 100 % correct annotations) and becomes incompatible after the update, then the dependencies with changed weight of annotation correctness indicate (together with the word classes) the RTE of the grammar to be updated.

In Fig. 12 we show a fragment of the table representing the results of re-parsing applied to a DTB. In this table:

- the first column is the reference to the DS of a sentence,
- the second column shows the (folded) characteristics of parsing complexity,
- in the third column, 'OK' means that all LU of

the sentence are present in the grammar lexicon and ‘DEFAULT’ means that there is at least one LU absent in the lexicon and replaced by the default unit,

- in the fourth column, ‘YES’ means that the re-parsing was successful in the sense that a successful analysis was found and ‘NO’ means the contrary,

- in the fifth column is given the maximal number of DS requested for re-parsing ($\geq k$ means that there are at least k requested DS),

- the sixth column shows the part (in percents) of annotation status coincidence of the DS before and after the update (so it is 0% for new sentences).

Autonomous Analysis. This is the mode of non-User-guided analysis. Parser-3.1 may be used as a general purpose parser for the extended CDG. In CDG LAB, there is a possibility to upload one’s own grammar or to introduce it through the Sandbox. Parser -3.1 is not well adapted to parsing with the grammars of such size and as ambiguous as French LB CDG. For instance, it passed 2373 seconds when applied to a test set of 559 French sentences of various complexity, representing the majority of French syntactic constructions. In so doing, it failed on 157 sentences exceeding the time limit of 10 seconds and successfully analysed the other 402 sentences. To compare: in the re-parsing mode, Parser -3.1 successfully analysed in 175 seconds 1442 DS of sentences, some of which are extremely long (up to 73 words). Another problem with Parser-3.1 is that it generates the DS not in the order of their adequacy. With ambiguous CDG, such as French LB, it generates hundreds of spurious structures per sentence. So for very long and complex sentences, it is practically impossible to know whether an adequate DS was computed. This is why we consider Parser-3.1 as a tool of development of DTB using head subtype selection, approximations and re-parsing. In these modes it performs very well and doesn’t impose any length limits on sentences. A higher-performing autonomous mixed stochastic-symbolic parser of extended CDG is under design.

DTB Development. The annotation based development of DTB in CDG LAB leads to a notable change in the point of view on the quality of treebanks. It is now the grammar, implementing a set of linguistic subjective expert knowledge, which will serve as the “gold standard”. As to the

DTB, they should all be correct with respect to the grammar and should be tested for correctness after every non-monotone grammar update. By definition, the monotone grammar updates preserve correctness of DS.

Besides the means based on DS annotation, CDG LAB also has rather standard means for creation and updates of DTB and for search of DS by projective and non-projective dependency names and by LU in the sentences.

Grammar Development. Besides the described above general purpose means supporting non-monotone grammar updates, CDG LAB has some means specific for CDG of French integrated with Lefff 3.0. In particular, it has several functions for completion of the lexicon of these CDG. Basically, there are two problems: the first is to automatically complete the lexicon by all forms of a missing word (this concerns mainly the verbs), the second is to compute the government pattern of a missing word from that of a present word. For the former problem, CDG LAB has several functions based on updates of the lexicon of French Text CDG. The latter problem mainly concerns the deverbals. The work on completions of this kind is in progress.

4 Conclusion

CDG LAB combines several means of incremental parallel development of wide coverage dependency grammars and of dependency treebanks provably correct with respect to the grammars. These means were successfully tested in the course of development of a wide coverage categorial dependency grammar of French and of an experimental dependency treebank. Some of these means are general purpose. E.g., the annotation weight difference test applies to any kind of structural incremental development based on expert annotations. Some other, such as head subtype selection and consecutive approximations, may be used with other classes of dependency grammars and may be implemented in tabular dependency grammar parsers. Some means are specific to the Parser-3.1 and to the French CDG integrated with Lefff. Several important means of CDG LAB are still under construction, but even this experimental version has proved its high efficiency.

References

- Y. Bar-Hillel, H. Gaifman, and E. Shamir. 1960. On categorial and phrase structure grammars. *Bull. Res. Council Israel*, 9F:1–16.
- Denis Béchet, Alexander Dikovsky, Annie Foret, and Erwan Moreau. 2004. On learning discontinuous dependencies from positive data. In *Proc. of the 9th Intern. Conf. “Formal Grammar 2004” (FG 2004)*, pages 1–16, Nancy, France.
- Denis Béchet, Alexander Dikovsky, and Annie Foret. 2010. Two models of learning iterated dependencies. In *Proc. of the 15th Conference on Formal Grammar (FG 2010)*, LNCS, to appear, Copenhagen, Denmark. [online] http://www.angl.hu-berlin.de/FG10/fg10_list_of_papers.
- Denis Béchet, Alexander Dikovsky, and Annie Foret. 2011. On dispersed and choice iteration in incrementally learnable dependency types. In *Proc. of the 6th Int. Conf. “Logical Aspects of Computational Linguistics” (LACL’2011)*, LNAI 6736, pages 80–95.
- I. Boguslavsky, S. Grigorieva, N. Grigoriev, L. Kreidlin, and N. Frid. 2000. Dependency treebank for russian: Concept, tools, types of information. In *Proc. of the 18th Int. Conf. on Comput. Ling. (COLING’2000)*.
- G. Bouma, G. van Noord, and R. Malouf. 2000. Alpino: Wide-coverage computational analysis of dutch. In *Proc. of the Conf. Computational Linguistics in the Netherlands*, pages 45–59.
- S. Brants and S. Hansen. 2002. Developments in the tiger annotation scheme and their realization in the corpus. In *Proc. of the Second Int. Conf. on Language Ressources & Evaluation (LREC’02)*.
- Michael Dekhtyar and Alexander Dikovsky. 2004. Categorial dependency grammars. In *Proc. of Intern. Conf. on Categorial Grammars*, pages 76–91, Montpellier.
- Michael Dekhtyar and Alexander Dikovsky. 2008. Generalized categorial dependency grammars. In *Trakhtenbrot/Festschrift*, LNCS 4800, pages 230–255. Springer.
- Michael Dekhtyar, Alexander Dikovsky, and Boris Karlov. 2010. Iterated dependencies and kleene iteration. In *Proc. of the 15th Conference on Formal Grammar (FG 2010)*, LNCS, to appear, Copenhagen, Denmark. [online] http://www.angl.hu-berlin.de/FG10/fg10_list_of_papers.
- Alexander Dikovsky. 2004. Dependencies as categories. In *“Recent Advances in Dependency Grammars”*. *COLING’04 Workshop*, pages 90–97.
- Alexander Dikovsky. 2007. Multimodal categorial dependency grammars. In *Proc. of the 12th Conference on Formal Grammar*, pages 1–12, Dublin, Ireland.
- Alexander Dikovsky. 2009. Towards wide coverage categorial dependency grammars. In *Proc. of the ESSLLI’2009 Workshop on Parsing with Categorial Grammars. Book of Abstracts*, Bordeaux, France.
- Alexander Dikovsky. 2011. Categorial Dependency Grammars: from Theory to Large Scale Grammars. Submitted paper.
- E. Hajicova, J. Panevova, and P. Sgall. 1998. Language resources need annotations to make them really reusable. In *Proc. of First Int. Conf. on Language Ressources & Evaluation (LREC)*, pages 713–718.
- J. Hockenmaier and M. Steedman. 2007. Ccgbank: A corpus of ccg derivations and dependency structures extracted from the penn treebank. *Computational Linguistics*, 33(3):355–396.
- Boris N. Karlov. 2008. Normal forms and automata for categorial dependency grammars. *Vestnik Tverskogo Gosudarstvennogo Universiteta (Annals of Tver State University). Series: Applied Mathematics*, 35 (95):23–43. (in Russ.).
- J. Lambek. 1961. On the calculus of syntactic types. In Roman Jakobson, editor, *Structure of languages and its mathematical aspects*, pages 166–178. American Mathematical Society, Providence RI.
- J. Lambek. 1999. Type grammars revisited. In Alain Lecomte, François Lamarche, and Guy Perrier, editors, *Logical aspects of computational linguistics: Second International Conference, LACL ’97, Nancy, France, September 22–24, 1997; selected papers*, volume 1582. Springer-Verlag.
- M. Marcus, B. Santorini, and M.A. Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19:313–330.
- S. Riezler, T. King, R. Kaplan, R. Crouch, J. Maxwell, and M. Johnson. 2002. Parsing the wall street journal using a lexical-functional grammar and discriminative estimation techniques. In *Proc. of the 40th Ann Conf. of the ACL*, pages 271–278.
- B. Sagot. 2010. The lefff, a freely available and large-coverage morphological and syntactic lexicon for french. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC’10)*.
- Mark Steedman. 1996. *Surface structure and interpretation*. MIT Press, Cambridge,Massachusetts.

Comparing Advanced Graph-based and Transition-based Dependency Parsers

Bernd Bohnet

University of Stuttgart
Institut für Maschinelle Sprachverarbeitung
bohnet@ims.uni-stuttgart.de

Abstract

In this paper, we compare a higher order graph-based parser and a transition-based parser with beam search. These parsers provide a higher accuracy than a second order MST parser and a deterministic transition-based parser. We apply and compare the output on languages, which have not been in the research focus of Shared Tasks. The parser are implemented in a uniform framework. The transition-based parser was newly implemented and we revised the graph-based parser. The graph-based parser has to our knowledge the highest published scores for French and Czech with 90.40 and 81.43 labeled accuracy score.

1 Introduction

The two main approaches to data-driven dependency parsing are transition-based dependency parsing (Yamada and M., 2003; Nivre, 2003; Nivre et al., 2004; Titov and Henderson, 2007) and maximum spanning tree based dependency parsing (Eisner, 1996; Eisner, 2000; McDonald and Pereira, 2006; Carreras, 2007; Johansson and Nugues, 2008).

The transition-based approach, might not provide the highest score for instance for English. Nevertheless, it can be justified to improve one of the approaches on its own because for some languages such as Catalan and Spanish, it had higher scores in the CoNLL shared task 2009, cf. (Gesmundo et al., 2009). The transition-based approach has a lower complexity and it is easier to

implement. For stacking experiments with both approaches, each has to be optimized separately towards speed and accuracy.

A statistical **transition-based parser** learns which actions to perform for building a dependency graph while scanning a sentence. The parser builds the dependency trees by going left-to-right (or right-to-left) through the words of a sentence. At each step, a classifier selects the appropriate parsing action for the current state based on a set of features. Transition-based parsers typically have a linear or quadratic complexity (Nivre et al., 2004; Attardi, 2006). Nivre (2009) introduces a transition-based non-projective parsing algorithm that has a worst case quadratic complexity and an expected linear parsing time. Titov and Henderson (2007) combine a transition-based parsing algorithm that uses a beam search with a latent variable machine learning technique. The latest update of this parser provided the best accuracy for transition-based dependency parsing in the CoNLL shared task 2009 (Gesmundo et al., 2009).

Graph-based dependency parsers start with a completely connected graph whose edges are weighted according to a statistical model. They then try to find the spanning tree that covers all nodes in the graph (the words) and at the same time maximizes the sum of the edges belonging to the spanning tree. The original non-projective formulation by McDonald et al. (2005) had a complexity of $O(n^2)$ but was not capable of taking second-order features into account (making the choice for an edge depending on already chosen edges). Second order MST parsing was shown to significantly improve results compared to first-

order parsing (McDonald et al., 2006; Carreras, 2007) but at the cost of a higher complexity (McDonald and Satta, 2007). Carreras (2007) also fully integrated edge labels into the parsing procedure by adding an additional loop over the set of edge labels (L), thus raising performance as well as theoretical complexity ($O(n^4L)$). Johansson and Nugues (2008) reduced the number of loops over the edge labels by looking only at those edges that existed in the training corpus for a distinct head and child part-of-speech tag combination. Recently, Koo and Collins (2010) introduced an efficient third-order dependency parsing algorithm. The algorithm considers substructures containing three dependencies, and is called efficient because it requires only $O(n^4)$ time. The parsing algorithm can utilize both features with sibling and grandchild information.

We apply a discriminative training method that employs a hash kernel to transition-based dependency parsing. Results show state-of-the-art unlabeled and labeled accuracy scores and fast parsing times. We illustrate that negative features can improve the accuracy of transition-based dependency parsers. Zhang and Clark (2008) as well as Gesmundo et al. (2009) applied a beam search to improve the accuracy of transition-based parsers.

2 Transition-based Parsing

We define a deterministic transition-based edge eager parser formally $T_e = \langle \sigma, \beta, \Omega, \epsilon, L, \Pi \rangle$ consisting of a list σ (stack), the list β (input), a set of operations $\Omega = \{\text{shift, left-arc, right-arc, reduce}\}$, a set of edges ϵ and a set of states Π . A state $\pi_i = \{\sigma_i, \beta_i, \epsilon_i\}$, $\pi_i \in \Pi$ consists of a list σ_i , an input buffer β_i and a set of edges ϵ_i .

(1) The initial state π_1 has an empty list σ_1 , the input buffer β_1 contains the words of a sentence, and the edge set ϵ is empty. (2) A transition $\tau_i(\pi_i, \omega) : \Pi \times \Omega \rightarrow \Pi$ is a binary function that maps a state and an operation to a new state π_{i+1} . We write a transition as $\pi_i \xrightarrow{\omega} \pi_{i+1}$. (3) The final state π_f has an empty input buffer β_f and therefore, no further operations are applicable.

The history of the (partial) parse h is a list of operations. We can define the operations and preconditions for the operations as follows:

The **shift** transition $\tau_s : \pi_i \xrightarrow{\text{shift}} \pi_{i+1}$ removes the first element of the input buffer $w_n \in \beta_i$ where: $\beta_i = \{w_n, \dots\}$ and adds the word to the end of the list $\sigma_{i+1} \leftarrow \sigma_i \cup \{w_n\}$. We obtain the

next state with $\pi_{i+1} = \{\sigma_{i+1}, \beta_{i+1}, \epsilon_i\}$. **Precondition:** $\beta_i \neq \emptyset$

The **left-arc** transition $\tau_l : \pi_i \xrightarrow{\text{left-arc}} \pi_{i+1}$ adds the last element $[+ s_l]$ of the list σ_i and the first element $[+ b_1]$ of β_i : $\epsilon_{i+1} \leftarrow \epsilon_i \cup \{(b_1, \text{label}, s_l)\}$; the element s_l is removed from σ_i : $\sigma_{i+1} \leftarrow \sigma_i - \{s_l\}$ and the first element of the input buffer is added as the last element to the list σ_i : $\sigma_{i+1} \leftarrow \sigma_i \cup \{b_1\}$: $\pi_{i+1} = \{\sigma_{i+1}, \beta_{i+1}, \epsilon_{i+1}\}$ **Precondition:** $\beta_i \neq \emptyset$ and $\sigma_i \neq \emptyset$ and not has-head(s_l)

The **right-arc** transition $\tau_r : \pi_i \xrightarrow{\text{right-arc}} \pi_{i+1}$ adds an edge between the last element $s_e \in \sigma_i$ and the first element $b_o \in \beta_i$: $\sigma_{i+1} \leftarrow \sigma \cup \{b_o\}$ and $\beta_{i+1} \leftarrow \beta_i - \{b_o\}$ $\epsilon_{i+1} \leftarrow \epsilon_i \cup \{(s_l, \text{label}, b_1)\}$ **Precondition:** $\beta_i \neq \emptyset$, & $\sigma_i \neq \emptyset$ and has-head(s_r)

The **reduce** transition $\tau_r : \pi_i \xrightarrow{\text{reduce}} \pi_{i+1}$ removes the last element of $s_l \in \sigma_i$: $\sigma_{i+1} \leftarrow \sigma - \{s_l\}$ **Precondition:** $\sigma_i \neq \emptyset$

Applying this definition, we define the transition-based dependency parser with beam search in Algorithm 1. We score a transition

Algorithm 1: Transition-based parser with beam search

```

//  $x_c$  is a input sentence
 $\sigma_0 = \emptyset, \beta_0 = x_c, \epsilon_0 = \emptyset, h = \emptyset$ 
 $\pi_0 \leftarrow \{\sigma_0, \beta_0, \epsilon_0, h_0\}$  // initial parts of a state
 $\text{beam}_0 \leftarrow \{\pi_0\}$  // create initial state
 $n \leftarrow 0$  // iteration
repeat
  for all  $\pi_j \in \text{beam}_n$  do
    operations  $\leftarrow$  possible-applicable-operation ( $\pi_j$ )
    // if no operation is applicable keep state  $\pi_j$ 
    if operations =  $\emptyset$  then  $\text{beam}_n \leftarrow \text{beam}_n \cup \{\pi_j\}$ 
    else for all  $\omega_i \in$  operations do
       $\pi \leftarrow \tau(\pi_j, \omega_i)$  // apply the operation i to state j
       $\text{beam}_n \leftarrow \text{beam}_n \cup \{\pi\}$ 
    // end for
  // end for
  // the score function is defined in the next section
  sort  $\text{beam}_n$  due to the score( $\pi_j$ )
   $\text{beam}_n \leftarrow$  sublist ( $\text{beam}_n, 0, \text{beam-size}$ )
   $n \leftarrow n + 1$ 
until  $\text{beam}_{n-1} = \text{beam}_n$  // stop when the beam is not changed

```

sequence h as the sum of the scores for the individual transitions $w_i \in h$:

$$\text{score}(\pi) = \sum_{i=0}^{|h|} F(\pi_i, \omega_i)$$

Note that a state π_i contains the stack σ_i , input buffer β_i , and set of edges ϵ_i . These elements are taken into account to create the feature set. The feature set is the input for the support vector machine; it provides the score due to the features.

3 Hash Kernel

We use a linear support vector machine with a Hash Kernel as classifier for our dependency parser, cf. (Shi et al., 2009; Bohnet, 2010). The Hash Kernel uses instead of a table to map the features to the indexes in the weight vector a random function. Therefore, the Hash Kernel can quickly process large numbers of features and hence we can use “negative” features. In most parsing approaches features are collected prior to the training phase which are derived from the gold trees and in the training phase, the feature set is not extended further. However, the decoder creates wrong structures and the features derived from predicted trees are often not found since they do not occur in the gold trees. We counted 9 times more negative features than positive ones. A Hash Kernel for structured data uses the hash function $h : J \rightarrow \{1..n\}$ to index ϕ . ϕ maps π_i to a feature space. We define $\phi(\pi_i)$ as the numeric feature representation indexed by J . Let $\bar{\phi}_k(x, y) = \phi_j(x, y)$ the hash based feature–index mapping, where $h(j) = k$. The scoring function of the Hash Kernel is

$$F(\pi_i) = \bar{w} * \bar{\phi}(\sigma_i, \beta_i, \epsilon_i, \omega_i)$$

where \bar{w} is the weight vector and the size of \bar{w} is n .

Algorithm 2: Update of the Hash Kernel

```

// $\pi_i$  is the state of a predicted state and
// $\pi_g$  the gold state including the transition
//sequences  $h_g$  and  $h_i$  for the gold and predicted state
// update( $\bar{w}$ ,  $\bar{v}$ ,  $\pi_i$ ,  $\pi_g$ ,  $\gamma$ )
err =  $\Delta(h_g, h_i)$  // number of wrong transitions
if err > 0 then
     $\bar{u} \leftarrow (\bar{\phi}(\pi_i) - \bar{\phi}(\pi_g))$ 
     $\nu = \frac{err - (F(\pi_i) - F(\pi_g))}{\|\bar{u}\|^2}$ 
     $\bar{w} \leftarrow \bar{w} - \nu * \bar{u}$ 
     $\bar{v} \leftarrow \bar{v} - \gamma * \nu * \bar{u}$ 
return  $\bar{w}$ ,  $\bar{v}$ 

```

Algorithm 2 illustrates the update function of the Hash Kernel. The update function is similar to that of (Crammer et al., 2006). The parameters of the function are the weight vectors \bar{w} and \bar{v} , the predicted state π_i , the gold state π_g , which should have been built by the parsing algorithm so far, and the update weight γ . The function Δ calculates the number of wrong transitions. The update function updates the weight vectors, when a transition is wrong. It calculates the difference \bar{u} of the feature vectors of the gold dependency structure $\bar{\phi}(\pi_i)$ and the predicted transition $\bar{\phi}(\pi_g)$.

The hash function f_h maps the features to integer numbers between 1 and $|\bar{w}|$. After that the update function calculates the margin ν and updates \bar{w} and \bar{v} respectively. The second weight vector is used for averaging in order to avoid overfitting and collects the weight of all training rounds with a passive-aggressive update.

4 Feature Selection

Transition-based dependency parsers are most frequently used with polynomial kernels of degree two since it is very convenient to specify features, cf. (Hall et al., 2006; Nivre et al., 2007; Nivre, 2009). SVMs of degree two use automatically derived combinations of at most two simple features. On the other hand, linear support vector machines provide faster training and classification times. Linear SVMs require a higher manual effort to select the features and combination of simple features because that involves many experiments where each time a parser has to be trained in order to find good combinations of simple features. Therefore, we had to perform a feature selection considering feature and combination. The feature templates are shown in Table 1.

5 Efficient Implementation

We want to emphasize similar to Goldberg and Elhadad (2010) that the parsing time is to a large degree determined by the feature extraction, the score calculation and the implementation.

We use a rich feature set and negative features. Nevertheless the parser is still fast with 47 sentences per second. This is because of the efficiency of the Hash Kernel, which is about four times faster than our implementation of the perceptron algorithm. With our baseline perceptron algorithm, we use about about 6 million features. The hash kernel uses about 50 million features including negative features. Our algorithms provides labeled trees, which distinguishes it from (Zhang and Clark, 2008) and (Goldberg and Elhadad, 2010). Some further optimizations are: (1) For the implementation of the beam, we store and reuse the calculated scores. We use a two step approach. We extract and store the values of the features that do not contain structural elements or elements of the stack σ except the last elements. This part of the weight is only calculated once and stored. Goldberg and Elhadad (2010) use a similar technique. We have to calculate the second part

Standard Features

$L,t,x,y : x \in \{sF,sP\} \& y \in \{bF,bP\}$	$L,t,sP,bP,x : x \in \{s-1P,s+1P,s+2P,s-2P\}$
$L,t,x,y,z : x \in \{s-1P,s+1P\} \& y \in \{b-1P,b+1P,b-2P,b+2P\} \& z \in \{sP,bP\}$	
$L,t,x,y,z : x \in \{s-1F,s+1F\} \& y \in \{b-1F,b+1F,b-2F,b+2F\} \& z \in \{sP,bP\}$	
$L,t,x,y,z : x \in \{s-1F,s+1F\} \& y \in \{b-1P,b+1P,b-2P,b+2P\} \& z \in \{sP,bP\}$	
$L,t,s-1F,s-2F,bP ; L,t,s-2F,s-3F,bP ; L,t,s+1F,s+2F,bP$	$L,t,s+2F,s+3F,bP$
$L,t,b-1F,b-2F,sP ; L,t,b-2F,b-3F,sP ; L,t,b+1F,b+2F,sP$	$L,t,b+1F,b+2F,sP$
$L,t,s-1P,s-2P,bP ; L,t,s-2P,s-3P,bP ; L,t,s+1P,s+2P,bP$	$L,t,s+2P,s+3P,bP$
$L,t,b-1P,b-2P,sP ; L,t,b-2P,b-3P,sP ; L,t,b+1P,b+2P,sP$	$L,t,b+2P,b+3P,sP$
$L,t,sP,x,y : x \in \{s+1F,s+2F,s+3F\} \& y \in \{b+1P,b+2P,b+3P\}$	
$L,t,sP,bP,x,y : x \in \{s-1P,s+1P\} \& y \in \{b-1P,b+1P\}$	

Structural Features

$L,t,x,bP,h(s) : x \in \{s+1F,s+1F\}$	$L,t,x,sP,h(s) : x \in \{b+1F,b+1P\}$
$L,t,sP,bP,x : x \in \{\text{leftsib}(s)L,\text{head}(s_1)L,\text{leftsib}(s_1)L,\text{head}(s_2)L\}$	L,t,s_1P,s_2P
$L,t,sP,bP,x : x \in \{\text{leftsib}(s)P,\text{head}(s_1)P,\text{leftsib}(s_1)P,\text{head}(s_2)P\}$	
$L,t,bP,x,y : x \in \{b+1F,b+2F,b+3F\} \& y \in \{s+1P,s+2P,s+3P\}$	

Table 1: t represents a transition type, s the last word of σ , b the first word of the input β . P represents the part-of-speech-tag, F the form and L the label. $-1,+1,+2$, etc. denote the location one or two word before or after an element. h denotes the head and *leftsib* the the leftmost sibling and *rightsib* rightmost sibling. s_1, s_2, b_1 , etc. are the last but one of σ , etc.

	wrong edges		total number of edges	% of correct edges	
	G	T		G	T
PMOD	315	334	5593	94.36	93.84
VC	15	14	1771	99.15	99.2
SBAR	56	57	1195	95.31	95.23
SUB	145	161	4108	96.47	96.08
PRD	47	56	832	94.35	93.26
P	875	931	7301	88.01	87.24
AMOD	339	339	2072	83.63	83.63
OBJ	142	155	1960	92.75	92.09
ROOT	92	121	2416	96.19	94.99
NMOD	1206	1235	21002	94.25	94.11
VMOD	938	997	8175	88.52	87.8
DEP	95	122	259	63.32	52.89

Table 2: Labeled accuracy scores of the graph-based (G) and transition-based (T) parse for distinct edge types using the Penn2Malt conversion.

of the score each time anew since this depends on structural parts (e.g left-most sibling of s_l) and the elements of σ . The space complexity is $O(n^2L)$ for the feature caching. (2) Furthermore, the calculation of each score is optimized: We calculate for each location determined by the last element $s_l \in \sigma_i$ and the first element of $b_0 \in \beta_i$ a numeric feature representation. This is kept fix and we add only the numeric value for each of the edge labels plus a value for the operation left-arc or right-arc. In this way, we create the features incrementally. (3) Further, we applied edge filtering as it is used in graph-based dependency parsing, cf. (Johansson and Nugues, 2008), i.e., we calculate the edge weights only for the labels that were found for the part-of-speech combination of the head and dependent in the training data.

6 Graph-based Parser

The basis for the graph-based parser is a higher order graph-based dependency parser developed by Bohnet (2010). We contribute two parts to this parser, which will become publicly available. A revision of the feature set and a new random function for the hash kernel that allows to create features incrementally. Based on the incremental features creation, we can provide a faster feature extraction. For the higher order dependency parser, the feature extraction iterates for all edges over all possible labels since the labels are part of the feature. Johansson and Nugues (2008) introduced the concept of edge filters. Edge filter constrain the possible edges labels to the labels, which occur in the training set for the part-of-speech tag combination of the head and its dependent.

Features are extracted for each possible edge label due to the edge constrains. However, the parser



System	Czech	French	English
Malt		87.32/89.73 ²	
MST		88.24/90.91 ²	
Merlo	80.38/-		
transition-based	77.75/84.58 ^{1,2)}	88.12/90.93 ²⁾	89.22/91.82 ^{1,2)}
graph-based	81.43/88.01 ^{1,2)}	90.4/92.81 ²⁾	90.48/92.58 ^{1,2)}

Table 3: Labeled dependency scores / unlabeled dependency scores for top scoring transition-based dependency parsers. ¹ including punctuation, ² predicted POS-tags

does not need to build always the complete features for each of the edge labels. It can extract once the features for an edge and add later the part for the edge label. The same strategy again is possible with the parts of the features of a head and the set of dependents. The parser extracts once the properties of the head and iterates over the possible dependents and adds to the feature part of the head a part for each of the dependents. The same strategy is possible for the sibling and grandchild features.

We could save 81% of the feature creation time and improve the speed of the parser by 25%. For instance, for French the parsing time went down from 0.079 seconds/sentence to 0.059¹

The features consists usually of several components. For instance, a standard second order features consists of the part-of-speech tag of a head, a dependent and a grandchild. These parts describe properties of a edge. There are in addition functional parts of a feature, which are the type of a feature and the edge label. The feature type is used to distinguish features for instance, a sibling features from a grandchild feature. Both types might have the same parts (equal number of part-of-speech tags) but they have of cause a different meaning.

The feature creation function composes parts to features. This can be done by different operations. A standard operator is the bit shift operator. For instance, a tag set might have 52 different tags. Therefore, 6 bits are needed to encode part-of-speech-tags. In order to encode several part-of-speech tags as a long value, we add the integer value of a part-of-speech tag and shift it by 6 bits. This procedure is repeated until all parts are encoded. This method wastes some of the encoding space since the 6 bit space could enumerate 64 values. Therefore, we use to encode the values the multiplication operator and multiply the value

¹We used a computer with 12 cores, Intel Westmere and 3.33 Ghz.

by the number of elements in the set, we want to encode.

The revised feature set combines systematically each part-of-speech tag, word form, lemma, distance features of the governor, dependent, sibling and grandchild. We used instead of a features for each words between the head and the lemmata, a single features that is a sorted bag of part-of-speech tags. The accuracy improved because of this for Czech and slightly for English as Table 3 shows.

7 Experiments

We trained the parser on English dependency trees as provided by the CoNLL shared task 2009 and on dependency trees converted with Penn2Malt using the head-finding rules of (Yamada and Matsumoto, 2003). Table 4 gives an overview of the data used with the these head-finding rules. The training data was 10-fold jackknifed with the tagger included in the Mate-Tools².

	Section	Sentences	PoS Acc.
Training	2-21	39.832	97.08
Dev	24	1.394	97.18
Test	23	2.416	97.30

Table 4: Overview of the training, development and test data split converted to dependency graphs with head-finding rules of (Yamada and Matsumoto, 2003). The last row shows the accuracy of Part-of-Speech tags.

We optimized our parser on section 24 and used section 23 of Penn Treebank for evaluation, which was the test set in the CoNLL shared task.

Table 6 summarizes the results and compares the result with Zhang and Clark (2008) as well as Goldberg and Elhadad (2010). We have taken the results for the Malt parser from Goldberg and Elhadad (2010).

²<http://code.google.com/p/mate-tools/>

System	LAS/UAS	Speed (sent./sec.)
Merlo	88.79/-	
Clear	89.15/91.18	430
this work	89.22/91.82	30

Table 5: **CoNLL Shared Task 2009 Data:** Labeled and unlabeled dependency scores of (Gesmundo et al., 2009) (Merlo), Choi and Palmer (2011) (Clear) and the parser introduced here.¹ including punctuation,² predicted POS-tags as provided in Shared Task.

System	UAS	Speed (sent./sec)
	including punctuation	
Malt	88.36	
NonDir	89.70	40
this work	91.81	47
	excluding punctuation	
Z&C08	91.4	50
Z&N11	92.90	29
this work	92.60	47

Table 6: **Penn2Malt, Train 2-21, Test 23,** predicted POS-tags: Unlabeled dependency scores of transition-based dependency parsers Zhang and Clark (2008) (Z&C08), Zhang and Nivre (2011) (Z&N11), Malt, NonDir (Goldberg and Elhadad, 2010).

In Table 5, we compare the scores of the our transition-based dependency parser with other transition-based parsers. The top score in the CoNLL Shared task 2009 was obtained by the parser of Gesmundo et al. (2009). This parser was ranked first in average for all languages and third for English, which was the best score of a transition-based parser for English. The labeled accuracy score of the dependency parser with Hash Kernel using the CoNLL data is about 0.4 percentage points higher than that of Gesmundo et al. (2009) and only slightly higher than the transition-based parser of Choi and Palmer (2011).

Table 6 shows results for the same data set but converted with Penn2Malt. The first three rows compare the result with other papers that included punctuation in their evaluation. The Malt and NonDir parser do not employ a beam search, which is probably the reason for the lower accuracy scores. The parser of Zhang and Clark (2008) is similar to our parser except that we use the Hash Kernel, which uses negative features in addition. The 2011 version (Zhang and Nivre, 2011) was published in the revision phase of this paper. Their parser uses a richer feature set and obtains 0.3 higher unlabeled accuracy scores. Remarkable is

that our parser as well as the parser of Zhang get close to the results of the second order and third order graph-based dependency parser that carries out an exhaustive search and obtains 93.04 UAS on the test set (Koo and Collins, 2010). Our parser is fast with 47 sentences/second and a beam size of 80 on a MacBook Pro (2.8 Ghz). Gesmundo et al. (2009) uses a beam size of 80 as well and Zhang and Clark (2008) of 64. We use 25 training rounds.

System	English
this work (transition)	92.60/91.48
this work (graph)	93.06/91.96
Z&N11 (transition)	92.9/91.8
KC10	93.04
CCK08	93.50
SICC09	93.79

Table 7: Results obtained by graph-based dependency parser compared with selected transition-based parsers: Z&N11 (Zhang and Nivre, 2011), SICC09 (Suzuki et al., 2009), KC10 (Koo and Collins, 2010), and KCC08 (Koo et al., 2008)

In Table 7, we compare results of transition-based and graph-based parsers. The upper part of the table shows results obtained by parsing systems that do not exploit additional resources. Our updated second order graph-based parser obtain competitive results with 93.06 UAS. Table 2 shows a more detailed analysis on the level of edge labels. Both parsers are similar good on majority of the dependency edges. The transition-based parser has still a bit lower accuracy for the attachment of the root node (ROOT), punctuation marks, and verb modifiers (VMOD). Reviewing the errors in dependence to the distance, we could only observe a very slight tendency that long distance relations are more worse in the case of transition-based parsers.³ An advantage of the transition-based parser is that it can observe some third order features, which the parser has already build, and also some subcategorization features.

Table 3 shows results of the graph-based and transition-based parser for Czech and English on the data of the CoNLL shared task 2009. For French, we use the data of Candito et al. (2010) as well as the same training, development and test data split. We obtain in line to English higher scores for the graph-based parser but the

³The graph-based parser has only 15% error rate on dependency spanning over more than 7 words in contrast to transition-based parser that has a error rate of 16.8%.

difference between the graph-based parser and transition-based parser for instance for Czech is still much higher. We think that the reason for this are the higher portion of non-projective edges.

8 Conclusion and Future Work

We have presented a fast transition-based dependency parser with competitive labeled and unlabeled scores. We have shown that a transition-based parser can benefit from a support vector machine with Hash Kernel that enables the use of negative features, which improve the accuracy.

Our transition-based and graph-based parser performance quite different on the two English data sets. The graph-based parser has a higher accuracy than the transition-based parser with about 1.2 percentage point for English and 3.7 for Czech on the data of the CoNLL Shared Task 2009. The difference between the conversion of the CoNLL and conversion obtained with the Yamada and Matsumoto (2003) head finding rules is high. We observed a difference of 1.2/0.7 LAS/UAS on the CoNLL data and only 0.4/0.48 LAS/UAS with the Yamada and Matsumoto (2003) rules. The cause of this is probably the larger number of edge labels and the non-projective edges contained in the CoNLL data.

References

- G. Attardi. 2006. Experiments with a Multilanguage Non-Projective Dependency Parser. In *Proceedings of CoNLL*, pages 166–170.
- B. Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China, August. Coling 2010 Organizing Committee.
- M. Candito, B. Crabb, and P. Denis. 2010. Statistical french dependency parsing: Treebank conversion and first results. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).
- X. Carreras. 2007. Experiments with a Higher-order Projective Dependency Parser. In *EMNLP/CoNLL*.
- J. D. Choi and Martha Palmer. 2011. Getting the most out of transition-based dependency parsing. In *ACL (Short Papers)*, pages 687–692.
- K. Crammer, O. Dekel, S. Shalev-Shwartz, and Y. Singer. 2006. Online Passive-Aggressive Algorithms. *Journal of Machine Learning Research*, 7:551–585.
- J. Eisner. 1996. Three New Probabilistic Models for Dependency Parsing: An Exploration. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, pages 340–345, Copenhagen.
- J. Eisner, 2000. *Bilexical Grammars and their Cubic-time Parsing Algorithms*, pages 29–62. Kluwer Academic Publishers.
- A. Gesmundo, J. Henderson, P. Merlo, and I. Titov. 2009. A Latent Variable Model of Synchronous Syntactic-Semantic Parsing for Multiple Languages. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL-2009)*, Boulder, Colorado, USA., June 4-5.
- Y. Goldberg and M. Elhadad. 2010. An efficient algorithm for easy-first non-directional dependency parsing. In *HLT-NAACL*, pages 742–750.
- J. Hall, J. Nivre, and J. Nilsson. 2006. Discriminative classifiers for deterministic dependency parsing. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 316–323, Sydney, Australia, July. Association for Computational Linguistics.
- R. Johansson and P. Nugues. 2008. Dependency-based Syntactic–Semantic Analysis with PropBank and NomBank. In *Proceedings of the Shared Task Session of CoNLL-2008*, Manchester, UK.
- T. Koo and M. Collins. 2010. Efficient third-order dependency parsers. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1–11, Uppsala, Sweden, July. Association for Computational Linguistics.
- T. Koo, X. Carreras, and M. Collins. 2008. Simple semi-supervised dependency parsing. In *ACL*, pages 595–603.
- R. McDonald and F. Pereira. 2006. Online Learning of Approximate Dependency Parsing Algorithms. In *In Proc. of EACL*, pages 81–88.
- R. McDonald and G. Satta. 2007. On the Complexity of Non-projective Data-driven Dependency Parsing. In *IWPT '07: Proceedings of the 10th International Conference on Parsing Technologies*, pages 121–132, Morristown, NJ, USA.
- R. McDonald, K. Crammer, and F. Pereira. 2005. Online Large-margin Training of Dependency Parsers. In *Proc. ACL*, pages 91–98.
- R. McDonald, K. Lerman, K. Crammer, and F. Pereira. 2006. Multilingual Dependency Parsing with a Two-Stage Discriminative Parser. In *Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 91–98.

- J. Nivre, J. Hall, and J. Nilsson. 2004. Memory-Based Dependency Parsing. In *Proceedings of the 8th CoNLL*, pages 49–56, Boston, Massachusetts.
- J. Nivre, J. Hall, S. Kübler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. 2007. The conll 2007 shared task on dependency parsing. In *Proc. of the CoNLL 2007 Shared Task. Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, June.
- J. Nivre. 2003. An Efficient Algorithm for Projective Dependency Parsing. In *8th International Workshop on Parsing Technologies*, pages 149–160, Nancy, France.
- J. Nivre. 2009. Non-Projective Dependency Parsing in Expected Linear Time. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 351–359, Suntec, Singapore.
- Q. Shi, J. Petterson, G. Dror, J. Langford, A. Smola, and S.V.N. Vishwanathan. 2009. Hash Kernels for Structured Data. In *Journal of Machine Learning*.
- J. Suzuki, H. Isozaki, X. Carreras, and M Collins. 2009. An empirical study of semi-supervised structured conditional models for dependency parsing. In *EMNLP*, pages 551–560.
- I. Titov and J. Henderson. 2007. A Latent Variable Model for Generative Dependency Parsing. In *Proceedings of IWPT*, pages 144–155.
- H. Yamada and Yuji M. 2003. Statistical dependency analysis with support vector machines. In *In Proceedings of IWPT*, pages 195–206.
- H. Yamada and Y. Matsumoto. 2003. Statistical Dependency Analysis with Support Vector Machines. In *Proceedings of IWPT*, pages 195–206.
- Y. Zhang and S. Clark. 2008. A tale of two parsers: investigating and combining graph-based and transition-based dependency parsing using beam-search. In *Proceedings of EMNLP, Hawaii, USA*.
- Y. Zhang and J. Nivre. 2011. Transition-based dependency parsing with rich non-local features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 188–193, Portland, Oregon, USA, June. Association for Computational Linguistics.

Incremental Parsing and the Evaluation of Partial Dependency Analyses

Niels Beuck, Arne Köhn and Wolfgang Menzel

Fachbereich Informatik

Universität Hamburg

{beuck, 5koehn, menzel}@informatik.uni-hamburg.de

Abstract

In this paper we discuss options for producing structural descriptions for an input sentence which is not yet completely available. Two existing dependency parsers have been modified to generate sequences of output hypotheses in an incremental manner. The parsing results can be characterized with respect to different criteria like the amount of predicted information, its quality, monotonicity, delay, inclusiveness and connectedness. We propose an evaluation scheme able to capture these properties and apply it to the parsers in different configurations.

1 Motivation

Incremental language processing does not consume its input at once but in a word-by-word manner. A sequence of incomplete, but successively more complete interpretations is generated for an utterance. Such a processing mode is particularly interesting in scenarios where language input evolves over time, like in human-computer or human-robot interaction. Since the input can be processed while it is still incomplete, production time is available as processing time. Moreover it also becomes possible to immediately respond to partial input, either by providing non-verbal feedback to the speaker, taking a turn, or starting an action while a command is still being spoken. Such a behavior requires a system which is able to produce an analysis for partial input. These intermediate results are provided for internal and external use. Internally they can guide the processing of the next input increment. External use includes feedback to previous processing modules and incremental input to subsequent processing modules.

To our knowledge no dependency parser is available so far which is able to generate fully connected intermediate results. Existing incremental dependency parsers wait at least until both words to be connected are available. This renders intermediate structures unconnected in most cases. The integration of new words is often delayed further due to lookahead.

The initial question to be answered in this paper is, therefore, how partial dependency analyses should look like and what information they should contain. This question is addressed in Section 2. In Section 3, possible metrics for the evaluation of partial dependency analyses are discussed. Section 4 shows how partial dependency analyses can be produced with a constraint dependency parser or a shift-reduce parser. In Section 5 the parser output is evaluated, before conclusions are drawn in Section 6.

2 Definitions

A dependency analysis is an directed acyclic graph where the words correspond to nodes and dependencies to edges. Exactly one head, also called regent, and one dependency type, also called label, is assigned to every word in a sentence. Candidates for a regent are the other words from the sentence or a special root node.

The dependency analysis for an incomplete sentence prefix will be called **partial dependency analysis** (PDA) throughout this paper. If only a prefix of the sentence is known, assigning a dependency structure to it is not trivial. The first problem is temporary ambiguity, i.e. the decision about the correct assignment for a word might depend on how the sentence continues. In such cases we cannot determine the correct analysis before the continuation of the sentence becomes available.

This uncertainty aggravates the general problem of global ambiguity, which is omnipresent even in complete sentences.

A second problem is introduced, since the words already known are usually not sufficient to represent the correct analysis. This becomes obvious if the structure of a complete sentence is cut off at an arbitrary position. Then we can distinguish four kinds of dependencies: those with both nodes in the already known prefix, those with an unknown dependent, those with an unknown regent and those lying completely outside the prefix. The most problematic class here is the one with unknown regent, as the dependent is part of the prefix but cannot be assigned correctly to one of the possible regents as defined above, i.e. a known word from the prefix or the root node. There are two possibilities to deal with this problem: delaying the assignment, i.e. not including the respective word into the PDA, or predicting hypothetical nodes for the not yet seen input.

A PDA that assigns a regent to every word in the prefix will be called **inclusive**. An analysis that contains nodes in addition to the ones corresponding to words in the prefix or the root node will be called **predictive**. Correct PDAs have to be predictive to be also inclusive: if the correct regent is not available in the prefix, either a placeholder for it has to be provided, e.g. by prediction, or no regent can be assigned. The minimal extension that is necessary to guarantee inclusion will be called **minimal prediction**. It consists of a single node added to the list of permissible regents. This extra node is predictive in the sense that it does not correspond to a known word from the prefix and it is maximally unspecified, i.e., its surface word form, lemma, part-of-speech and its position beyond the fact that is to the right of the other words are unknown. Even its identity is unspecified, i.e., it could stand for an arbitrary number of words and two dependencies meeting at the predicted node do not necessarily meet in the complete dependency analysis.

Also the new node can only serve as regent, but not as dependent of any other node. This kind of predictive node was previously proposed by Daum (2004) and is called *nonspec* due to its unspecified nature. Assigning *nonspec* as regent is more informative compared to not including the respective dependent at all: Firstly, a dependency label can be assigned to the attachment and secondly,

it can be taken for granted that *nonspec* is neither one of the known words nor the root node. While delaying the attachment reflects the uncertainty about the correct regent, attachment to *nonspec* expresses the certainty that the word will not be attached to one of the already known words.

Although minimal prediction facilitates inclusion, it is not sufficient to guarantee the connectedness of a dependency structure. A (partial) dependency analysis is called **connected**, if there is a path of dependencies, ignoring direction, between every two words of the sentence (prefix).

Since *nonspec* itself does not have a regent, the words assigned to *nonspec* are not connected to the other nodes of the dependency graph. Such unconnected words cannot be easily related to the rest of the prefix in a semantic interpretation. We will therefore further extend the number of regents to allow **structural prediction**. Now predictive nodes themselves can be assigned to a regent. These so called **virtual nodes** differ from *nonspec* in that there can be more than one of them, that they require a regent themselves and that each virtual node represents exactly one word from the unknown suffix of the sentence. Features of virtual nodes like their part-of-speech or their order can be specified. Edges between virtual nodes are also possible.

3 Quality Metrics

In the previous section we presented two kinds of prediction which can be used to augment partial dependency analyses. To be able to compare them, we need to quantify and measure their quality.

3.1 Attachment Score

For dependency analyses of complete sentences, quality is usually measured by the attachment score (AS). It is defined as the ratio of words in the sentence that have been assigned to the same regent as in a gold standard annotation of the same sentence (UAS) and, optionally, with the same label (LAS).

There are several difficulties in applying these measures to partial dependency analyses, especially to those including predictions: First of all, there are no gold standard annotations available for sentence prefixes, only for complete sentences. This problem can be addressed in two different ways: either by annotating a corpus of sentence prefixes, or by generating prefix annotations from

existing full sentence annotations. As there are multiple prefixes for every sentence, annotating prefixes requires considerable effort compared to annotating complete sentences. In addition, temporary ambiguity might result in multiple plausible annotations for the same prefix. The same problem occurs if prefix annotations are extracted from a corpus of complete annotations. Two sentences might share a prefix but assign a different syntactic structure to the words in the prefix.¹ The resulting corpus would then contain two "correct" analyses for the same sequence of words. Additionally, the interpretation for the complete sentence might not be the most plausible interpretation for each of its prefixes. However, a corpus large enough might level out these implausible prefix annotations by a greater number of plausible annotations for similar syntactic constructions. A parser that always chooses the more common structure for a prefix would then obtain a better score. In this paper we will use existing dependency annotations.

Complete dependency analyses can be rated with only one score instead of two values like precision and recall because the number of nodes is fixed for a given sentence in contrast to the number of nodes in a phrase structure graph. For PDAs, however, the number of words in the structures to be compared might vary, depending on how many of them have been included or predicted. Therefore, the accuracy measure for PDAs has either to be split into a precision and a recall part, or only inclusive PDAs with no prediction beyond minimal prediction can be considered.

Structurally predictive PDAs can be reduced to minimal predictive ones by interpreting all attachments to predicted regents as minimal predictive attachments and ignoring all attachments of predicted dependents. For cases of non-inclusiveness, where words only remain unattached as long their heads are not yet available, the missing attachments can be interpreted as nonspec attachments².

¹Among 15000 sentences from the Negra corpus more than 5000 share a prefix with another one, which is annotated differently. 81% of these shared prefixes are of length one, 15% of length two and 4% longer than two.

²However, no dependency label can be assigned in such a case. Also, incremental parsers can use lookahead to incorporate features of later words into the decision for a word. This interferes with the re-interpretation of missing attachments as attachments to nonspec, since it would be wrongly assumed that all the words in the lookahead window should be attached to nonspec. Therefore, such results are better dealt with by specifying the fixed lookahead size or providing a separate

This way we can guarantee a fixed number of dependents for a prefix.

Let A be an annotation consisting of dependency arcs (*dependent, label, regent*). Then the annotation A_P for a prefix P is:

$$A_P = \{(d, l, r) \in A \mid d \in P \wedge r \in P \cup \{\mathbf{root}\}\} \cup \{(d, l, \mathbf{nonspec}) \mid d \in P \wedge \exists x((d, l, x) \in A \wedge \neg x \in P \cup \{\mathbf{root}\})\}$$

A general PDA B for a prefix P can be reduced to a minimal predictive and inclusive PDA:

$$\hat{B} = \{(d, l, r) \in B \mid d \in P \wedge r \in P \cup \{\mathbf{root}\}\} \cup \{(d, \mathbf{nolabel}, \mathbf{nonspec}) \mid d \in P \wedge \neg \exists x.(d, -, x) \in B\} \cup \{(d, l, \mathbf{nonspec}) \mid d \in P \wedge \exists v.(d, l, v) \in B \wedge \mathit{virt}(v)\}$$

With this normalization we can assign an attachment score to a PDA as usual. This measurement does not reward prediction beyond the minimal prediction, but provides a common ground for all inclusive partial dependency analyses.

For the Prefix "John" of the sentence "John buys a book" the annotation $\{A = (John, SUBJ, buys), (buys, S, root)...\}$ and the predictive analysis $B_1 = \{(John, SUBJ, [virt]), ([virt], S, root), \dots\}$ would both be normalized to $\{(John, SUBJ, nonspec)\}$, resulting in an AS of 100%. The non-inclusive and empty analysis $B_2 = \{\}$ would be normalized to $\{(John, nolabel, nonspec)\}$, resulting in a UAS of 100%, but a LAS of 0%.

3.2 Accumulating prefix scores

Incremental parsing does not produce a single prefix analysis, but sequences of them. Simply accumulating the accuracies for all the words in all prefixes would introduce a strong bias in favor of the earlier tokens: a word appearing early in a sentence will have a greater influence on the overall score than a later one, giving it more weight in the accumulated score.

Therefore, we apply a sliding window to the sequences of PDAs. For every word the attachment status is determined not only for the prefix it first appears in (and the final result), but also for a fixed number of prefixes in the vicinity of the first appearance. This allows us to investigate the temporal evolution of a word's attachment as well as to give all words the same weight.³

recall value.

³modulo effects at the start and end of the sentence, depending on the window size

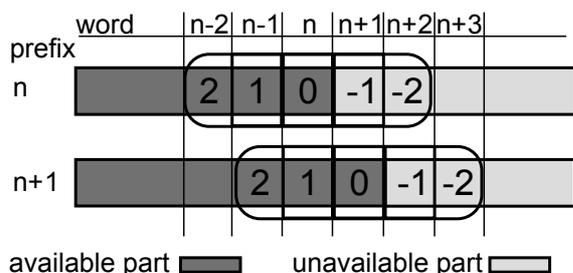


Figure 1: Sliding window for prefix n and $n + 1$ with relative word indices

Note, however, that for incremental systems using lookahead there might not be a single PDA for every input increment. As the lookahead window must first be filled up before any output can be generated, no explicit output will be produced for the first n input increments. As the final input increment fills the lookahead window of the last $n + 1$ words, there are $n + 1$ output increments for it. This can be compensated by adding n empty output increments at the beginning and by keeping only the final analysis for the last input increment.

We can determine three different attributes for an attachment: its correctness (including the label or not), its status (i.e., whether it is included, not included, a minimal or a structural prediction) and its stability (i.e., whether it differs from a later PDA). There might be a difference in the degree to which the predictive attachments are specified. Here, we distinguish between minimal prediction without dependency label, with dependency label and structural prediction.⁴

For every PDA the window is centered at the rightmost input word, so that the previous words are assigned to slots with ascending numbers, as illustrated in Figure 1. The correctness C_{minP} for a given slot for a given prefix is determined as discussed above for the attachment score where non-spec matches any non-included word. Precision P_{minP} and recall R_{minP} can then be calculated by dividing by the number of included words in the PDAs or the total number of encountered words respectively.⁵ Both can be averaged for the window (cf. Figure 2).

With respect to structural prediction two addi-

⁴Even a more comprehensive prediction including the lexical features of a word or its surface form would be possible. This has not been considered here as the choice of the correct lexical reading is also not covered by the usual definition of the attachment score.

⁵This number depends on the slot number: all words are encountered by slot 0, but slot n will not encounter the last n words of a sentence.

tional aspects have to be considered. First of all, the predicted words have to be mapped to words from the gold standard in a one-to-one correspondence. While different mappings might be possible, the mapping bm with the best overall accuracy is chosen. This optimum might depend on whether accuracy is measured labeled or unlabeled, we use unlabeled attachment. Given that a virtual node can partake in more than one dependency edge, i.e. it has one regent and an arbitrary number of dependents, a virtual node can possibly be mapped even if some of these attachments are incorrect. Mappings that share no edge with the gold standard annotation are not considered. Therefore, some virtual nodes might remain unmapped. As the optimal mapping might change as the prefix of a sentence grows, the same virtual node can be mapped to different words for consecutive PDAs. Per definition, all predictions still left in the final result are incorrect.

Secondly, with structural predictions words can be assigned to a regent before they become part of the available prefix. This corresponds to a negative slot index in the window. The sliding window, however, is applied to the positions of the words in the gold standard annotation, not in the PDA, as virtual nodes are not located at a specific position. Thus, they only have the potential to be captured by the sliding window if being successfully mapped. If mapping fails, the virtual node has no well-defined position. Therefore, only the recall of structural R_{strP} prediction, but not the precision P_{strP} , can be determined by means of a sliding window alone. For words with a negative slot number, we calculate the correctness $C_{strP,V}$ independently of slots for all predicted words and combine it with the correctness of the non-negative slots to obtain an accumulated precision, as defined in Figure 2

3.3 Stability

The stability score for a given slot is calculated like precision, but the PDA is compared to the final annotation found by the parser instead of the gold standard annotation.

3.4 Connectedness

Early semantic interpretation requires to integrate incoming words into a connected structure immediately. We quantify the degree of connectedness of a PDA by means of its average **fragmentation**,

$$\begin{aligned}
R_{minP} &:= \frac{\sum_{p \in P_s} \sum_{i \in W} C_{minP}(i, pda_p, |p|)}{\sum_{i \in W} (|s| - |i|)} \\
P_{minP} &:= \frac{\sum_{p \in P_s} \sum_{i \in W} C_{minP}(i, pda_p, |p|)}{\sum_{p \in P_s} |pda_p \cap W|} \\
R_{strP} &:= \frac{\sum_{p \in P_s} \sum_{i \in W} C_{strP}(i, pda_p, |p|, bm(pda_p))}{\sum_{i \in W} (|s| - |i|)} \\
P_{strP} &:= \frac{\sum_{p \in P_s} (\sum_{i \in W_+} C_{strP}(i, pda_p, |p|, bm(pda_p)) + \sum_{v \in V_{pda_p}} C_{strP,V}(v, pda_p, bm(pda_p)))}{\sum_{p \in P_s} (|pda_p \cap W_+| + |V_{pda_p}|)} \\
C_{minP}(sID, pda, n) &:= \begin{cases} 1 & \text{if } reg_{pda}(n - sID) = reg_{gold}(n - sID) \\ 1 & \text{if } reg_{pda}(n - sID) = nonspec \wedge reg_{gold}(n - sID) > n \\ 0 & \text{else} \end{cases} \\
C_{strP}(sID, pda, n, map) &:= \begin{cases} 1 & \text{if } map(reg_{pda}(n - sID)) = reg_{gold}(map(n - sID)) \\ 0 & \text{else} \end{cases} \\
C_{strP,V}(vn, pda, map) &:= \begin{cases} 1 & \text{if } map(reg_{pda}(vn)) = reg_{gold}(map(vn)) \\ 0 & \text{else} \end{cases}
\end{aligned}$$

Figure 2: Definitions for precision and recall for a single sentence, where P_s is a the set of all prefixes of a sentence s , W the slot-ids of the used window, W_+ the non-negative ones, pda_p the analysis for a prefix p , V_{pda} the virtual words used in pda and $|pda \cap W|$ the amount of arcs in an analysis covered by the window, i.e. the number of included words. $bm(pda)$ is the best mapping as defined in Section 3.2.

defined as the average number of tree fragments in addition to the first one.

This is an indication of how many attachments have to be changed at least, to produce a connected tree. As minimal predictive attachments do not predict whether they attach to the same word, each such attachment has to be counted as a potential root of an additional tree fragment. Punctuation marks are never integrated into the dependency graph, and therefore not be considered. The average fragmentation number can then be compared to the fragmentation of the gold standard annotation.

4 Implementation

In this section we will present two approaches for incremental parsing which produce partial dependency analyses that are both inclusive and predictive. We modified two existing parsing systems WCDG⁶ and MaltParser⁷ to generate incremental output.

4.1 WCDG

Weighted Constraint Dependency Grammar (WCDG) is a framework, which maps depen-

⁶<https://nats-www.informatik.uni-hamburg.de/view/CDG/>

⁷<http://maltparser.org>

dependency parsing to the problem of constraint optimization (Schröder, 2002). Menzel (2009) proposed an incremental parsing strategy for WCDG based on the repair based algorithm frobbing (Foth, 2006). It tries to improve an initial structure through a sequence of conflict driven transformation steps. To perform incremental parsing this algorithm can be applied to the prefix of a sentence, and the generated structure (plus an arbitrary attachment for the new words) is used as a starting point for the analysis of the extended prefix. This approach is non-monotonic, as the previous PDA provides only a starting point for the next search step, but the resulting PDA does not need to include all the arcs of its predecessor. WCDG is able to profit from information contributed by external modules (Foth, 2006). We used only use the most essential ones: a PoS tagger and a PP attacher, for our experiments.

Nonspec

As frobbing produces inclusive dependency analyses where a regent is assigned to every word, the system has to assign a regent even in the absence of the intended one (c.f. Section 2). A suitable attachment point has to be made available. A minimal prediction with a nonspec node serves exactly this purpose.

With nonspec, the grammar has to be changed in two regards. First, a constraint is added to slightly penalize nonspec attachments. It guarantees that a nonspec attachment is only chosen if no suitable real regent is available.

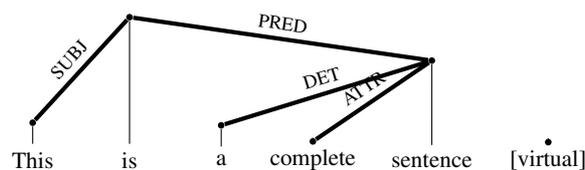
A second change adds guards to the constraints to prevent non-existing attributes of the regent from being accessed. These guards are specified in a way that they replace the non-specified feature with an optimistic estimation. For example a query for a comma between the two ends of a dependency edge would return *true* for constraints demanding a comma, while the same query would return *false* in a constraint that forbids a comma (unless there is already a comma in the known part of the prefix).

Virtual Nodes

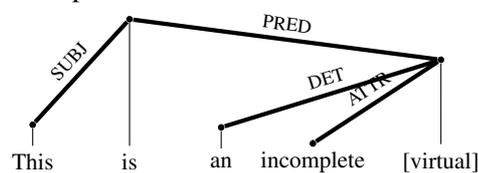
While this implementation of incremental dependency parsing accomplishes minimal prediction, it does not exhaust the potential for syntactic prediction of a given grammar. Constraints demanding the existence of certain words or their lexical features are either prevented from accessing those features, or alternatively their violation, like an unsatisfied verb valency, is simply accepted because no less penalized alternative is available. In particular, no proof is required that and how a predicted regent itself could be integrated into the rest of the dependency structure without violating additional constraints. To extend the range of prediction in dependency analyses, the concept of virtual nodes as defined in Section 2 is used.

Since the frobbing search algorithm is not able to add or remove words to or from the constraint problem, a maximal set of potentially useful predictive nodes has to be introduced prior to search. As with nonspec, attachments to and from virtual nodes are penalized slightly. Virtual nodes which are not integrated into an analysis stay unconnected and are considered unused. As a result, used virtual nodes are per definition always attached to another node. Unused virtual nodes are assigned to the root node with the empty label as dependency type. They may not be assigned as regents to other words. This is enforced by a hard constraint in the grammar. They are not considered part of the sentence and can safely be removed from an analysis without altering its meaning.

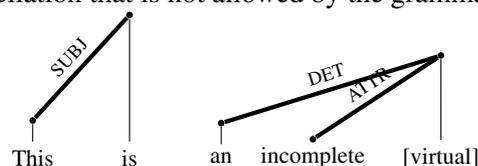
Example for an unused virtual node:



Example for a used virtual node:



Example for a partially used virtual node, a constellation that is not allowed by the grammar:



Virtual nodes, once added to the constraint problem technically behave like other words. Their predictive nature is not visible to the search algorithm, as the topology of the search space remains the same. All restrictions mentioned above are enforced via constraints in the grammar. To be able to distinguish between virtual and non-virtual nodes in a constraint, a new attribute *virtual* is defined. A corresponding predicate can be invoked by a constraint definition. With this approach, prediction, i.e. the inclusion of virtual nodes into the dependency structure, is purely constraint driven.

We can distinguish between two kinds of prediction, bottom-up and top-down. In bottom-up prediction, the inclusion of a predictive node is driven by an unconnected word, for which every other integration would result in constraint violations. Top-down prediction is conflict driven in that a specific constraint violation indicates the need for an additional dependent of an existing word, as it is the case for verb valencies. By providing the search algorithm with a set of predictive nodes for potential use, predictive partial results can be generated without further adaption of the algorithm. All that is needed is adding candidates for dependency arcs for the virtual nodes to the search space and extend the grammar, as discussed above.

Replacement of virtual nodes

After a prediction has been included into an incremental parsing step, it has to be replaced later on with a word from the input, once a fitting word be-

comes available. This can in principle be achieved by the search algorithm, but many transformation steps might be needed to properly integrate it into the existing structure. An alternative consists in checking for each new word prior to search, whether it can fill one of the used virtual nodes, instead of adding it as a separate word to the structure.

To determine whether a replacement is successful, the ratio of penalties assigned by the grammar before and after replacement is compared to a threshold. If at least one successful replacement has been found, the one with the highest score is used as starting point for the next search step. Otherwise the word is appended as usual.

4.2 MaltParser

MaltParser (Nivre et al., 2007) is a dependency parser which provides an incremental algorithm but no incremental output in the sense of intermediate analyses for prefixes. We, therefore, had to modify the parser in a way that allows us to extract partial dependency analyses from its hypothesis space. For that purpose the set of already submitted dependency arcs is recorded immediately before the next word in the input buffer is read, while yet unattached words are considered to be attached to nonspec. This allows us to recover the PDAs for every increment.

There are several algorithms available for MaltParser. The best choice depends on coverage of non-projectivity, eager arc attachment and explicit root handling. As the evaluation is done for German, a language with a comparably high degree of non-projective constructions, it is mandatory to use a version which is able to deal with non-projectivity.

In general, the shift-reduce approach used by MaltParser does not guarantee an arc to be built as soon as both nodes are available. As the attachment reduces the token from the stack rendering it unavailable to further attachments, dependents to the right of their head cannot be attached before all their dependents have been included into the structure. Nivre (2003) proposed a so called arc-eager approach, which splits the *right-reduce* action into a *right-arc* and a *reduce* action. This modification allows an immediate attachment, once head and dependent are available.

There are two ways to deal with root attachment, either as an explicit attachment via an arc

building action or by waiting until the sentence has been completely parsed and attaching all words still left unattached to the root. For our purpose we need the explicit root attachment approach to be able to distinguish temporarily unattached words (interpreted as nonspec attachment) from root attachments. The explicit root attachment implemented by MaltParser has proven not to be exhaustive. Especially punctuation tokens are always left unattached. As those are always attached to the root, it is easy to deal with them separately. For other words that are left unattached despite explicit root handling there is no way to detect whether they will stay unattached until the end of the sentence. The impact of this problem on accuracy, however, is minimal, as for these words the root attachment would often be incorrect as well.

From the algorithms fulfilling these requirements we choose the 2-planar algorithm (Gómez-Rodríguez and Nivre, 2010), as it provides the best performance for German and does not require post-processing to recover non-projective links. It uses an approach with two stacks and an additional parsing action to *switch* between them. Although this does not allow it to parse general non-projective structures, all 2-planar non-projective structures can be dealt with, which covers most non-projectivities in most natural languages, e.g., more than 98% for German. For more details see Gómez-Rodríguez and Nivre (2010).

4.3 Differences between WCDG and MaltParser

The biggest difference is that MaltParser is trained on a tree-bank while WCDG uses a manually generated dependency grammar together with trained external components.

Both parsers apply an incremental algorithm in the sense that information from a previous analysis are used to calculate the analysis for the extended prefix, but apply different strategies to deal with temporary ambiguity as defined in Beuck et al. (2011). While WCDG applies reanalysis, resulting in timely but non-monotonic output, MaltParser applies lookahead, resulting in monotonic but delayed output.

5 Evaluation

5.1 Setup and data

In this section we will compare different configurations of MaltParser and WCDG by evaluating

them with the metrics proposed in Section 3. Evaluation has been carried out on 500 German sentences from the Negra corpus converted to a dependency structure. MaltParser was trained on 15000 different sentences from the same corpus.

Also, the parsers are compatible with different strategies of incremental PoS tagging. While MaltParser is restricted to taggers with best guess or lookahead strategies, WCDG is able to integrate multi-tagging and non-monotonic tagger output. Based on the evaluation of incremental PoS taggers in Beuck et al. (2011), we chose the TnT tagger⁸ with multi-tagging and re-tagging of each prefix for WCDG, while MaltParser is combined with SVMTool⁹ using a lookahead of one or zero, depending on the configuration.¹⁰ As MaltParser accuracy is often reported on gold tagged input, we also provide these numbers for a comparison with already published non-incremental results.

We evaluated MaltParser configurations with a total lookahead between zero and four, as well as incremental WCDG configurations with nonspec (NS), virtual nodes (VN), and both mechanisms activated (VN+NS). All these configurations are evaluated with the scheme for minimal prediction (P_{minP} and R_{minP}). In addition, the WCDG-VN configuration is evaluated in terms of structural prediction (P_{strP} and R_{strP}).

5.2 Discussion

Table 1 contains final accuracy, as well as precision and recall values integrated over a window of size 9. Figure 3 shows the temporal evolution of accuracy and stability within the window for different parser configurations. In these figures the different behavior of MaltParser and WCDG become apparent.

Due to the monotonic nature of MaltParser, the only possible change in subsequent output increments is the replacement of minimally predictive attachments by fully specified ones. Thus, the average accuracy of attachment decreases after the initial appearance of a word. In contrast to this, the accuracy can even rise over time if reanalysis is allowed as in WCDG. Here, the stability of initial attachments is only 70%, i.e., 30% of the

attachments have been changed later on.¹¹

A noteworthy observation is that WCDG proposes significantly more erroneous initial attachments to available words, where a predictive attachment would be a better choice. Obviously, it is too eager to attach words to available regents, but is able to recover in many cases by means of reanalysis.

The delayed output of the configurations with lookahead leads to a smaller number of slots¹² in the window having received any attachments. This is reflected in a reduced recall.

Structural prediction with WCDG leads to an increased recall score, 48.6% compared to the best recall without structural prediction of 46.2% for MaltParser) and 44.3% for WCDG. The precision, however, is reduced, but it should also be noted that a structural predictive attachment contains more information. If we ignore this additional information and interpret the virtual nodes only in a minimal predictive sense, precision and recall are higher than in the configuration with nonspec. The real benefit of structural prediction can be seen in the significantly reduced fragmentation, as indeed the PDAs are connected to a similar degree as the gold standard annotations for the full sentences (which has a fragmentation of 0.17%).

6 Related Work

To our knowledge, partial dependency analyses have not been investigated previously in detail. Work on incremental dependency parsing like Nivre (2004) was focused on the incrementality of the algorithm, not on providing an incremental interface. Therefore, the output of intermediate results was not a primary goal. In other cases, like Menzel (2009), the evolution of partial analyses has been studied, but no broad scale evaluation has been carried out.

Regarding connected partial analyses in other grammar formalisms, Demberg and Keller (2008) presented a variant of the tree adjoining grammar (TAG) formalism that is able to incrementally produce fully connected prefix analyses. In this approach prediction plays a strong role, too. Top-

¹¹In fact there is also a kind of non-monotonicity in MaltParser, if we interpret the unattached words as "to be attached to a not yet available word". These are reinterpreted as being attached to root in the final result, leading to stability reduction of 5%

¹²The few assignments in slots 0-3 in Figure 3d are due to an end-of-sentence effect, where the lookahead window is filled preliminarily.

⁸<http://www.coli.uni-saarland.de/thorsten/tnt/>

⁹<http://www.lsi.upc.edu/nlp/SVMTool/>

¹⁰Another reason for not using SVMT for WCDG, besides performance in different incremental tagging modes, is that SVMT is not able to provide tag percentages, which are needed as constraint weights in WCDG.

parser	configuration	final AS		Average Fragment.	Sliding Window		Precision of struct. pred.
		unlabeled	labeled		Precision	Recall	
Minimal Prediction:							
WCDG	NS	87.02%	84.95%	1.00	78.22%	43.28%	-
	VNs	86.74%	84.57%	1.00	79.98%	44.25%	-
	VNs + NS	86.58%	84.43%	1.00	79.95%	44.23%	-
Malt	LA 0+0	82.28%	78.99%	1.45	83.57%	46.24%	-
	LA 1+0	84.27%	80.65%	1.36	85.27%	38.59%	-
	LA 2+0	84.62%	80.98%	1.26	86.00%	30.75%	-
	LA 2+1	85.04%	81.74%	1.10	86.81%	23.28%	-
	LA 3+1	85.06%	81.66%	0.98	87.11%	16.06%	-
Structural Prediction:							
WCDG	VNs	86.74%	84.57%	0.16	77.05%	48.55%	63.46%
Gold Tagged:							
Malt	gold tags	88.76%	86.25%	-	-	-	-

Table 1: Evaluation of incremental WCDG with different configurations regarding prediction; Lookahead numbers are given as "parser LA + tagger LA"

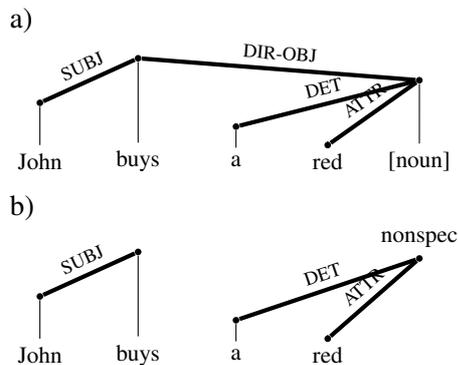


Figure 4: A connected (a) and an unconnected (b) PDA

down prediction is facilitated through substitution nodes in lexical entries, e.g. verbal valencies. Bottom up prediction is achieved by means of connection paths, i.e. the need for additional nodes to connect a subtree to the rest of the structure. A comparison with our results by applying the proposed metrics on derivation trees of TAGS is beyond the scope of this paper but a promising topic for further research.

The metrics in this paper only capture syntactic similarity, but not the utility of an analysis for an application task. Eventually, a more semantically oriented measure would be desirable, which reflect the amount of semantic information conveyed by a structure. The sentence prefix "John buys a red", for example, contains the information $buys(John, X)$ and $color(X, red)$. Since such an information can be more easily extracted from a

predictive dependency analysis like the one in Figure 4 a) compared to the not connected analysis (4 b)), it would be desirable to assign a higher recall value to a). An application oriented measure for prefix analyses is defined by Schlangen et al. (2009) where several variants for incremental reference resolution are discussed. They are, however, only applicable for utterances with a single reference.

7 Conclusions

In this work we presented a definition of partial dependency analyses that allows us to derive fully connected structures by introducing predicted nodes into the dependency graph. It was discussed how the attachment score metric can be extended to also cover such prefix analyses. In addition, a windowing approach was adopted to analyse the temporal evolution of incremental output sequences in more detail.

Using these measures, two existing dependency parsers have been compared. Obviously, there is still a large number of parameters left unexplored, especially for the instantiation of virtual nodes. Eventually, it will be interesting to study possible similarities between psycholinguistic findings about garden path or reanalysis phenomena and the behaviour of the presented architectures.

This work was funded by the Deutsche Forschungsgemeinschaft (DFG) as part of the International Graduate Research Group CINACS.

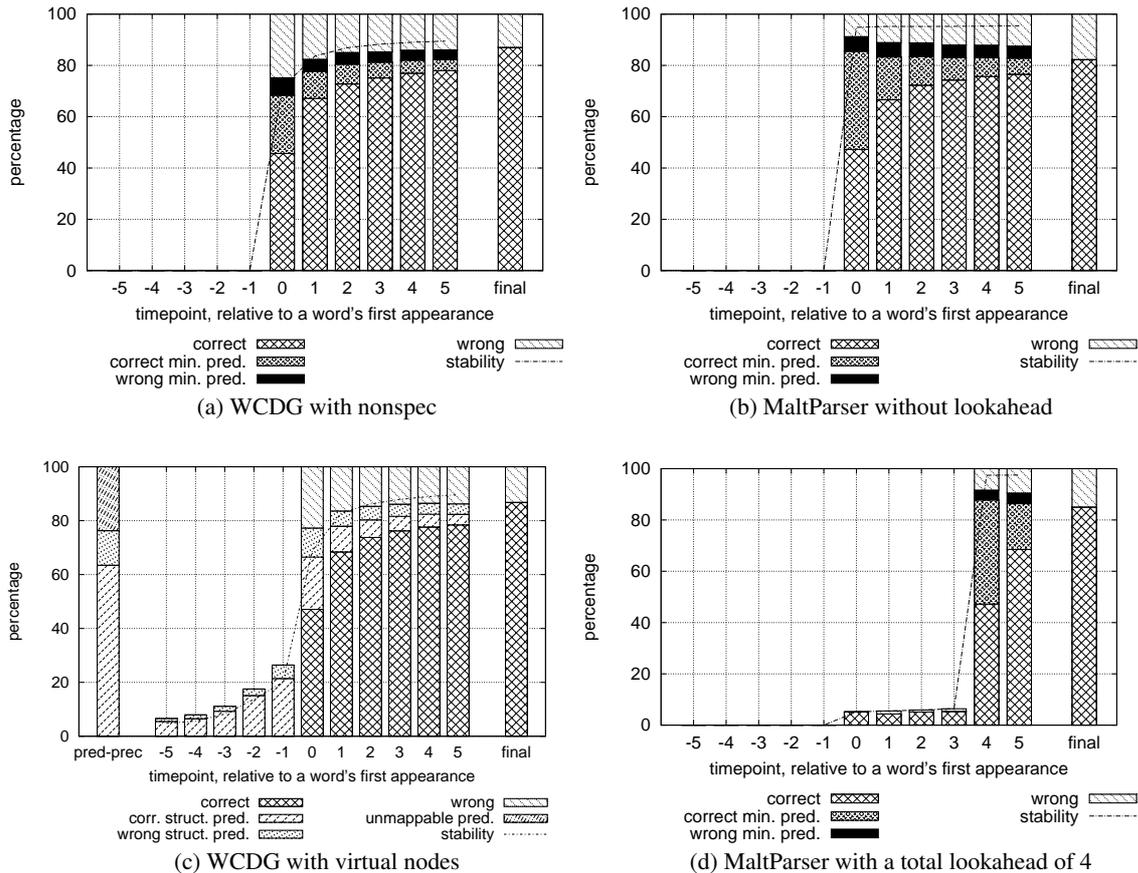


Figure 3: Scores for a sliding window with 9 slots; slot 0 holds a words first appearance in the input; the earlier slots to the left are only filled for the configuration with structural prediction; the leftmost bar is the precision of the attachment of virtual nodes (pred-prec), the rightmost one is the AS for the complete sentence; all scores are unlabeled

References

- Niels Beuck, Arne Köhn, and Wolfgang Menzel. 2011. Decision Strategies in Incremental PoS Tagging. In *Proceedings of NODALIDA 2011*.
- Michael Daum. 2004. Dynamic dependency parsing. In *In Proceedings of the ACL 2004 Workshop on Incremental Parsing*.
- Vera Demberg and Frank Keller. 2008. A psycholinguistically motivated version of tag. In *Proceedings of the ninth international workshop on tree adjoining grammars and related formalisms*.
- Kilian A. Foth. 2006. *Hybrid Methods of Natural Language Analysis*. Ph.D. thesis, Uni Hamburg.
- Carlos Gómez-Rodríguez and Joakim Nivre. 2010. A transition-based parser for 2-planar dependency structures. In *Proceedings of ACL 2010*.
- Wolfgang Menzel, 2009. *Recent Advances in Natural Language Processing V*, chapter Towards radically incremental parsing of natural language, pages 41–56. Number 309 in Current Issues in Linguistic Theory. John Benjamin’s Publisher.
- J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, and E. Marsi. 2007. Malt-parser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13:95–135.
- Joakim Nivre. 2003. An Efficient Algorithm for Projective Dependency Parsing. In *Proceedings of IWPT 03*.
- Joakim Nivre. 2004. Incrementality in deterministic dependency parsing. In *Incremental Parsing: Bringing Engineering and Cognition Together, Workshop at ACL 2004*.
- David Schlangen, Timo Baumann, and Michaela Atterer. 2009. Incremental reference resolution: the task, metrics for evaluation, and a bayesian filtering model that is sensitive to disfluencies. In *Proceedings of SIGDIAL 2009*.
- Ingo Schröder. 2002. *Natural Language Parsing with Graded Constraints*. Ph.D. thesis, Uni Hamburg.

Improving Dependency Label Accuracy using Statistical Post-editing: A Cross-Framework Study

Özlem Çetinöglu Anton Bryl Jennifer Foster Josef van Genabith
NCLT/CNGL, School of Computing, Dublin City University, Ireland
{ocetinoglu, abryl, jfoster, josef}@computing.dcu.ie

Abstract

We present a statistical post-editing method for modifying the dependency labels in a dependency analysis. We test the method using two English datasets, three parsing systems and three labelled dependency schemes. We demonstrate how it can be used both to improve label accuracy in parser output and highlight problems with and differences between constituency-to-dependency converters.

1 Introduction

The quality of dependency analyses produced by automatic parsing is usually evaluated using both *attachment accuracy* and *label accuracy*. A parsing system's attachment accuracy reflects its ability to recover structure correctly, i.e. dependencies between heads and dependents. Label accuracy, on the other hand, reflects the system's ability to correctly determine the nature of these dependencies. In order to ascertain *who* did *what* to *whom*, the dependency labels are crucial since they allow us to distinguish between grammatical roles (subjects versus objects, indirect objects versus adverbial modifiers, etc.). In this paper we focus on dependency labels and present a simple post-editing method for boosting label accuracy.

The idea behind the method is to automatically capture systematic error patterns characterised by local features. A set of parser output dependency analyses is compared to a set of gold standard analyses and a label revision model is learned which can then be applied to new dependency analyses. We ex-

periment with two feature sets to condition the probability of a label. The first makes use of lexical information and the second includes more structural context. We find that both feature sets are effective on their own but are more so when we backoff to the non-lexicalised feature set in the event that the lexicalised feature set does not make a prediction.

The method is designed to fix labelling errors rather than attachment errors, and in that it differs from the tree revision rules of Attardi and Ciaramita (2007). Label and attachment post-editing can be viewed as complementary techniques and in practice may potentially be combined within one system. To our knowledge, this is the first post-editing method to target dependency label accuracy.

In order to fully demonstrate the strengths and weaknesses of the post-editing method, we apply it to two datasets, three parsers and three labelled dependency schemes. In theory, the method is language-independent, although, in this study, we concentrate on English. Our two main datasets are the Wall Street Journal Section of the Penn Treebank (Marcus et al., 1994) and QuestionBank (Judge et al., 2006). We employ two dependency parsers and one constituency parser. The dependency parsers are trained directly on dependency trees produced by applying constituency-to-dependency conversion to Penn Treebank constituency trees. The constituency parser, on the other hand, is trained on the Penn Treebank constituency trees and its output is converted to dependency trees using the same conversion procedure. The dependency parsers we employ are MaltParser (Nivre et al., 2006) and MSTParser (McDonald et al., 2005), and the constituency parser

is the two-stage Charniak and Johnson reranking parser (Charniak and Johnson, 2005). The use of more than one labelled dependency scheme is desirable not only because there is no one standard dependency scheme for English but also because it allows us to highlight some of the differences between the various schemes. The three schemes we employ are LTH (Johansson and Nugues, 2007), Stanford (de Marneffe et al., 2006; de Marneffe and Manning, 2008) and LFGDEP (Çetinoğlu et al., 2010).

From our experiments with the post-editing method we can conclude the following:

Constituency Parser Results The post-editing method results in improved labelled attachment scores for the Charniak and Johnson parser and the three dependency schemes. For two of the schemes, the improvements are statistically significant (89.82 → 91.12 for LTH and 90.67 → 90.88 for LFGDEP).

Dependency Parser Results The method does not work well for the two dependency parsers. Our initial explanation for this failure was the relatively low attachment accuracy of the dependency parsers in comparison to the constituency parser — because the Charniak and Johnson parser has higher unlabelled attachment accuracy than MaltParser and MSTParser, it might be able to benefit more from the method since label modifications can only be learned from correctly attached dependencies. However, this cannot be the main reason as the method also works well for the first-stage Charniak parser (Charniak, 2000) which has unlabelled attachment accuracy at a similar level to MSTParser.

The importance of function labels and null elements The difference between the Stanford scheme and the LTH and LFGDEP schemes is that the Stanford scheme has been designed to be applied to constituency trees which do not contain function labels or null elements.¹ The other two converters work better when applied to trees containing this information and so there is an inherent mismatch between gold constituency trees, which contain function labels and null elements, and constituency parser output, which doesn't (since function labels and null elements are generally stripped

¹Traces, null complementisers, etc. See Bies et al. (1995, Chapter 4).

from the gold trees before training constituency parsers). The dependency parsers are trained on the gold constituency trees with this information intact. We show, for the constituency parser experiments, that the post-editing method can be used to recover some of the information from function labels by comparing the use of the method on raw constituency parser output to its use on trees which have been passed through an automatic function labeller. We show that it can also be used to recover information from null elements by comparing the use of the method on dependency parser output to its use on dependency parser output which has been produced by training a dependency parser on gold constituent trees *with null elements removed*: the latter is the only scenario where the post-editing method works for dependency parsing. To sum up, the post-editing method is able to recover the kind of information that is encoded in constituency trees via function labels and null elements.

Training material for the post-editor We find that the post-editor works when trained on the same data on which the parser was trained. This is an encouraging practical result since it demonstrates that improvements may be achieved at no additional annotation cost.

The paper is organised as follows: we begin by discussing related work in Section 2; Our datasets, parsing systems and labelled dependency schemes are described in Section 3, and the post-editing method itself is described in Section 4. Our experiments with the post-editing method are presented and discussed in Section 5. Finally, Section 6 contains some suggestions for future work.

2 Related Work

Attardi and Ciaramita (2007), Keith and Novak (2005; 2011) and Anguiano and Candito (2011) present techniques for automatic correction of dependency trees. The basic idea behind these approaches and the approach described here is the same — correction rules are learned from training data consisting of parser output for which gold standard analyses are available. The difference is that previous techniques learn how to modify the structure of the dependency tree, whereas our technique learns how to modify the labels on individual depen-

dependency arcs. The more general idea of statistical post-editing has also been applied to machine translation output (Simard et al., 2007).

Dickinson (2008; 2010) has explored the use of automated techniques to signpost potential anomalies in parse trees by identifying atypical cases in both attachments and labelling. Similarly, Goldberg and Elhadad (2010) present a method to learn the systematic attachment biases of particular dependency parsing algorithms. Our method, though originally designed for post-editing, can be also applied for error analysis purposes. That is, the relabelling technique can be used, not only as a post-editing correction step, but also as a type of diagnostic to signal differences between two sets of dependency trees, and hence, potential problems with either parser output or gold standards.

Bryl et al. (2009) presented a way of restoring the missing dependency labels in LFG-based statistical machine translation output. Atomic features of LFG f-structures, such as case, number, etc., were used as features for a Naive Bayes classifier. Though the problem is similar to ours, the approach is not readily reusable for our purpose, because such atomic features (many of which are highly relevant for guessing the correct label) are often not used in the kind of parsers we explore in our work.

3 Data and Tools

3.1 Datasets

We employ two datasets in this work, the Wall Street Journal Section of the Penn Treebank (Marcus et al., 1994) and QuestionBank (Judge et al., 2006), a set of 4,000 manually parse-annotated questions from a TREC question answering task.² Both datasets contain constituency trees which have been produced by an automatic parser and then corrected by hand. It is important to note that the trees in the *WSJ* dataset contain more information than the trees in *QuestionBank*, namely null elements and function labels on nonterminal categories.

We use *WSJ22* as our post-editing training/development set and *WSJ23* as our test set. We use sentences 2001-3000 from *QuestionBank* as our post-editing training/development set and sentences

²Questions occur relatively infrequently in the *WSJ* dataset (Clark et al., 2004).

3001-4000 as our test set. For the remainder of the paper, we use the term *QuestionDev* to refer to this development set and the term *QuestionTest* to refer to the test set.

3.2 Parsing Systems

We evaluate the post-editing method using one constituency parser and two dependency parsers, both trained on Sections 2-21 of the *WSJ* section of the Penn Treebank (Marcus et al., 1994). Our constituency parser is the Charniak and Johnson parser, and the dependency parsers are MaltParser and MSTParser, which exemplify the two main approaches to statistical dependency parsing, namely, transition-based dependency parsing and maximum-spanning-tree dependency parsing.

The Charniak and Johnson parser (C&J) The Charniak parser (Charniak, 2000) is a generative constituency parser which uses a head-lexicalised smoothed PCFG which is conditioned on the parse history and whose probability model is fine-tuned for English. We mainly experiment with the reranking version in which the n-best list returned by the first-stage generative parser is re-ordered using a discriminative reranker trained on features extracted from the complete trees (Charniak and Johnson, 2005), although we also test the method with the first-stage parser.

MaltParser is a multi-lingual transition-based dependency parsing system (Nivre et al., 2006). During training, a classifier learns to predict a parsing action at a particular parsing configuration using information from the parse history and the remaining input string. During parsing, the classifier is used to deterministically construct a dependency tree. For our experiments, we use the *stacklazy* parsing algorithm, which can handle non-projective structures (Nivre et al., 2009). Following Attardi and Ciaramita (2007) and Zhang and Clark (2008), we train a linear classifier which models interactions between features using feature conjunctions.

MSTParser Instead of predicting parsing actions, MSTParser (McDonald et al., 2005) comes from the family of dependency parsers which learn to predict entire dependency trees. The parser finds the maximum spanning tree in a multi-digraph using one of

several algorithms described in McDonald (2006). For our experiments, we use the second-order approximate non-projective parsing model introduced in McDonald and Pereira (2006). Labels are predicted using an atomic maximum entropy model as in Nivre et al. (2010).

Both MaltParser and MSTParser expects POS-tagged input — we use SVMTool (Gimenez and Marquez, 2004) to perform POS tagging.

3.3 Labelled Dependency Schemes

General statistics on the three labelled dependency schemes are provided in Table 1.

Stanford The Stanford dependency scheme represents parser output as labelled bilexical dependencies, and it has been designed with real-world applications in mind (de Marneffe et al., 2006; de Marneffe and Manning, 2008). Stanford dependencies can produce dependencies in different formats. We focus on *basic* dependencies, because we want to be able to compare with two other representations both of which assume that representations are trees that include all tokens. Stanford dependencies do not use null elements and function labels during the conversion and the resulting trees are projective.

LTH In contrast to the Stanford conversion tool, the LTH tool (Johansson and Nugues, 2007) relies on the function tag and trace information in constituency trees. The resulting dependencies – which were used in the CoNLL 2007 dependency parsing shared task (Nivre et al., 2007) – are designed to be useful in downstream semantic processing. The LTH dependency scheme has the richest set of labels of the representations used in this study and, because it tries to take trace information into account, has a higher proportion of non-projective dependencies.

LFGDEP Çetinoğlu et al. (2010) introduce a dependency scheme that takes as a basis a linguistically motivated Lexical Functional Grammar (LFG) f-structure and changes it so that it is a dependency tree. It uses the LFG Annotation Algorithm (AA) which generates LFG f-structures from Penn Treebank style trees (Cahill et al., 2008). This dependency scheme has a lower number of labels than the Stanford and LTH dependencies. The trees can be non-projective but the proportion of non-projectivity

	Stanford	LTH	LFGDEP
# sent	39832	39832	39171
# dep types	49	67	25
non-proj. deps	0%	0.41%	0.29%
non-proj. sents	0%	7.75%	5.62%
head left of modifier	51.6%	60%	53%

Table 1: WSJ sections 02-21 conversion statistics

is not as high as LTH (see Table 1).

4 Dependency Label Post-Editing

The new dependency label for the i th arc in a dependency structure, $l_{i,new}$, is predicted as follows:

$$l_{i,new} = \arg \max_{l_{i,gold}} \hat{P}(l_{i,gold} | f_{i,1}, f_{i,2}, \dots)$$

where $l_{i,gold}$ is the gold (correct) dependency label of the i th dependency arc in the structure; $f_{i,1}, f_{i,2}$, etc. are features extracted from the parser output; and \hat{P} is the approximation of the given probability calculated on a training dataset for which gold standard parses are available. If several labels receive equal probability estimates, the “do not change” outcome is given priority. With our present method, we make no assumption about feature independence, and instead approximate the probability directly:

$$\hat{P}(l_{i,gold} | f_{i,1}, f_{i,2}, \dots) = \frac{\text{count}(l_{i,gold}, f_{i,1}, f_{i,2}, \dots)}{\text{count}(f_{i,1}, f_{i,2}, \dots)}$$

Only correctly attached (in accordance with the gold standard) dependency arcs are used for training. We additionally request that the denominator of the above fraction is not less than 2; in other words, that a decision is made on the basis of at least two relevant samples in the training data. It means, that for some cases no decision is made. This allows us to combine several post-editing transformations in a queue. If, for the given case, a post-editing transformation with a longer feature list refuses to make a decision, another post-editing transformation with a shorter feature list may be given a chance.

In the experiments presented in this paper we employ a combination of two post-editing transformations, with feature sets as follows (all features are taken from the parser output; so, for example, “the dependency label of the arc in question” is the piece

of data which might be replaced as a result of the transformation).³

1. **Lexicalised feature set:** the label of the arc in question, the POS tag of its dependent word, and the surface form of its dependent and head words (see left tree in Figure 1)
2. **Non-lexicalised feature set:** the label of the arc in question, the POS tag of its dependent word and the label of its parent arc (see right tree in Figure 1)

In our preliminary experiments, Naive Bayes was also tried on the same features (as well as some other features, described later in the section) and produced very discouraging results. Together with some correct modifications this method made a huge amount of wrong ones, signalling that the Naive Bayesian assumption is too strong for these features and leads to over-generalisation. With Naive Bayes, we also tried to use concatenations of feature pairs as additional features (to make the independence assumption slightly less strict); this modification proved insufficient and did not improve the situation. In the same way, we also tried multi-class SVM⁴ with a linear kernel, also with a clearly negative outcome. Given these results, the methods similar to these two (e.g. perceptron-based or maximum entropy) are rather unpromising here.⁵

The following additional features were used in the experiments with the described method and/or the alternative ones (Naive Bayes and SVM) but were not included in the final configuration as they failed to noticeably improve the situation: the POS tag of the parent; the POS tags of the previous and next words in the sentence; direction to the parent in the surface word order (parent to the left vs. parent to the right); presence/absence of siblings.

³We settle on these two feature sets after experimenting on our development sets.

⁴SVM^{multiclass} (<http://svmlight.joachims.org>) was used, which is a variant of SVM^{light} (Joachims, 1999)

⁵One possibility that could be of interest in future work is to develop a combined approach: that is, to limit the search for strict matches in the training data (which is our current method) to only a subset of features, and in this form to use it as a training data preselection method for Naive Bayes, SVM or some other general-purpose classifier.

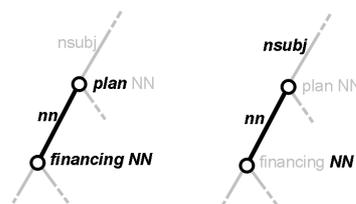


Figure 1: Lexicalised and unlexicalised features sets

5 Experiments

For both WSJ and QuestionBank, the method is evaluated on both development and test data. For evaluation on a development set (*QuestionDev* or *WSJ22*), a leave-one-out approach is used, i.e. each tree in the development set is corrected with the posteditor trained on the rest of the same development set. For evaluation on a test set (*WSJ23* and *QuestionTest*), the posteditor is trained on the corresponding development set. For WSJ, we also experiment with using the full parser training data to train the post-editor. For some experiments, we apply an automatic function labeller, FunTag (Chrupała et al., 2007), to the output of C&J, and to the QuestionBank gold trees (which have not been labelled with function tags).⁶ We use the CoNLL evaluation metrics of labelled attachment score (LAS) and unlabelled attachment score (UAS).

5.1 WSJ Results

The results for the WSJ dataset are shown in Tables 2-4. For each parser type, the baseline scores are provided first, followed by the post-editing scores obtained when using *WSJ22* for training (when *WSJ22* is stated as both training and test set, it means that leave-one-out evaluation took place). The post-editor results when the training set is *WSJ2-21* are given in the third row. The scores are provided both for *WSJ22* and for *WSJ23*. The number of correct modifications minus the number of wrong modifications are provided beside the labelled attachment scores.

Concentrating first on the C&J results, we can see from Tables 2-4 that LTH benefits the most from

⁶In the task of node function labelling, FunTag achieves an f-score of 91.47% when evaluated using a dataset consisting of correctly parsed *WSJ23* constituents.

post-editing. It is followed by LFGDEP and then Stanford. The reason for these large differences in correction balances between the conversion schemes is their design decisions. The parser outputs do not contain function labels and LTH suffers from the lack of this information. LFGDEP is less dependent on them and Stanford is almost insensitive. This explanation is confirmed by using FunTag. When function labels are provided by FunTag, the order of balances remains the same, but the correction balance drops dramatically for LFGDEP and even more for LTH, while the already small correction balances decreases slightly for Stanford dependencies.

For the Stanford scheme, the most successful post-editing rule is the one in which generic `dep` relations are converted to more informative `npadvmod`⁷ relations. Using FunTag eliminates the problem almost without a need for post-editing. Training the post-editing tool with a larger data set does not affect the results.

For LTH, relations incorrectly labelled as `VMOD` are converted to various other relations including `ADV`, `SUBJ` and `OBJ`. The correction type breakdown is different for C&J and C&J FT. The `VMOD` corrections appear to cease altogether with FunTag, but actually FunTag only transforms `VMOD` into `DEP` in most of the cases. It still needs to be corrected and it is successfully handled by the post-editing tool. In most frequent sub-cases of `VMOD => SBJ/OBJ` conversions, the post-editing tool converts them to the correct label without using FunTag. When the post-editing tool is trained on *WSJ2-21* instead of *WSJ22*, it makes fewer modifications — the number of incorrect modifications in particular drops, and this explains the increase in correction balance. The type of the corrections is almost the same, but how they are corrected differs. When the post-editor is trained on *WSJ22*, the non-lexicalised feature set is used in modifications. The same modifications are carried out based on the lexicalised feature set when the size of the training data increases. On *WSJ23*, correct modifications increase, and, more importantly, incorrect modifications drop dramatically. As a result the balance increases by 0.5 % absolute, a statistically significant improvement.

Looking at the breakdown of results in Table 4,

⁷noun phrase adverbial modifier

we see that, for the LFGDEP dependency scheme, the post-editing rules succeed in correctly converting adjuncts to obliques and complements to adjuncts. Very few instances of these corrections remain after using FunTag. Post-editing corrects only `topicrel => subj` in the C&J FT configuration. This covers sentences with a relative pronoun which acts both as a subject and a relative topic. Due to design decisions (there is only one head of a dependent and a grammatical function has a higher priority than a discourse function), LFGDEP prefers to keep the `subj` relation. Gold trees have the subject information due to traces and coindexation, so LFGDEP correctly picks the `subj` relation. Parse trees lack this information hence, only `topicrel` can be assigned. The other remaining correction is `subj => adjunct`, which highlights a systematic error made by LFGDEP. Using a larger training data does not change the type of modifications and slightly increases the correction balance.

Post-editing does not help the dependency parsers for any of the conversion schemes. A closer look reveals that the kind of errors made by the dependency parsers appear to be unsystematic. One exception to this is if the dependent word is a preposition, in which case, additional experiments suggest that it is worth including in the feature set the surface form of the dependent. The failure of the method to work for the two dependency parsers does not appear to be related to the lower unlabelled attachment accuracy of Malt and MST in comparison to C&J because the method also works well for the first-stage Charniak parser which has a UAS close to that of MST. Interestingly, when null elements are removed from the gold training constituency trees before conversion to dependencies for dependency parser training, the method achieves more promising results. This suggests that the kind of information that is supplied by the post-editing method is already available in the dependency parsers' training data.

5.2 QuestionBank Results

The QuestionBank results in Table 5 are interesting because they highlight the different ways the post-editing method can be used. The method works better for QuestionBank than for the WSJ dataset because, for all three parsers, it succeeds in transforming the parser output so that it more closely resem-

Parser	WSJ 22		WSJ 23	
	UAS	LAS	UAS	LAS
C&J	94.18	91.52	94.21	91.76
C&J post-editor-WSJ22	94.18	91.82 (128 - 26 = 102)	94.21	91.94(20 - 9 = 11)
C&J post-editor-WSJ2-21	94.18	91.80 (118 - 21 = 97)	94.21	91.98(20 - 7 = 13)
C&J FT	94.18	91.94	94.21	92.03
C&J FT post-editor-WSJ22	94.18	91.99 (31 - 14 = 17)	94.21	92.06(109 - 20 = 89)
C&J FT post-editor-WSJ2-21	94.18	91.95 (11 - 10 = 1)	94.21	92.06(129 - 17 = 112)
Malt	90.61	87.98	90.28	87.68
Malt post-editor-WSJ22	90.61	87.93 (11 - 26 = -15)	90.28	87.67(15 - 23 = -8)
Malt post-editor-WSJ2-21	90.61	87.95 (12 - 16 = -4)	90.28	87.68(11 - 8 = 3)
MST	91.33	88.76	90.74	88.36
MST post-editor-WSJ22	91.33	88.74 (14 - 26 = -12)	90.74	88.35(22 - 27 = -5)
MST post-editor-WSJ2-21	91.33	88.73 (9 - 16 = -7)	90.74	88.35(7 - 10 = -3)

Table 2: Parser accuracy scores for WSJ 22 and WSJ 23 when Stanford Dep. is used

Parser	WSJ 22		WSJ 23	
	UAS	LAS	UAS	LAS
C&J	92.21	65.32	91.91	64.31
C&J post-editor-WSJ22	92.21	82.57 (6313 - 25 = 6288)	91.91	81.52(8803 - 18 = 8785)
C&J post-editor-WSJ2-21	92.21	84.54 (7112 - 95 = 7017)	91.91	84.46(10377 - 32 = 10345)
C&J FT	93.99	89.66	93.86	89.82
C&J FT post-editor-WSJ22	93.99	90.87 (530 - 92 = 438)	93.86	90.68(659 - 233 = 426)
C&J FT post-editor-WSJ2-21	93.99	90.89 (483 - 26 = 457)	93.86	91.12(710 - 31 = 679)
Malt	90.84	87.18	90.80	87.58
Malt post-editor-WSJ22	90.84	87.22 (87 - 96 = -9)	90.80	87.31(46 - 209 = -163)
Malt post-editor-WSJ2-21	90.84	87.17 (21 - 24 = -3)	90.80	87.61(32 - 15 = 17)
MST	92.24	88.8	91.89	88.9
MST post-editor-WSJ22	92.24	88.81 (78 - 78 = 0)	91.89	88.7(40 - 146 = -106)
MST post-editor-WSJ2-21	92.24	88.77 (8 - 19 = -11)	91.89	88.91(9 - 6 = 3)

Table 3: Parser accuracy scores for WSJ 22 and WSJ 23 when LTH is used

bles the gold standard. However, we have to be careful here since the QuestionBank gold dependencies are even less “gold” than the WSJ gold dependencies for three reasons: 1) QuestionBank constituency trees have undergone not one but two automatic procedures, function labelling (recall that the QuestionBank constituency trees do not contain functional labels) and constituency-to-dependency conversion, 2) the constituency trees which are converted to dependencies do not contain null elements, and 3) the three constituency-to-dependency converters and the function labeller have been developed using PTB trees and so they are not expected to perform as well on questions. Examining the QuestionBank results in more detail we find problems with the individual converters as well as problems with parser output.

The LTH converter particularly suffers when applied to *QuestionDev*. The most common “correct”

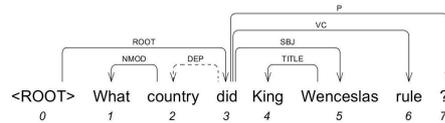


Figure 2: The incorrect gold dependency tree converted by the LTH scheme

relabelling rules for the two dependency parsers involve a label being converted to the generic DEP label. In order to investigate these suspicious relabelling rules, we inspect the gold standard LTH *QuestionDev* dependency trees and find that these dependency trees are in fact incorrect (see, for example, the tree in Figure 2). It is interesting that we discover this problem by looking at the dependency parser relabellings — in this case, the post-editing method is making the dependency parser output

Parser	WSJ 22		WSJ 23	
	UAS	LAS	UAS	LAS
C&J	92.22	87.35	91.67	87.61
C&J post-editor-WSJ22	92.22	88.77 (678 - 104 = 574)	91.67	88.48 (691 - 196 = 495)
C&J post-editor-WSJ2-21	92.22	89.44 (978 - 148 = 830)	91.67	89.33 (1190 - 235 = 955)
C&J FT	92.85	90.83	92.49	90.67
C&J FT post-editor-WSJ22	92.85	90.99 (97 - 23 = 74)	92.49	90.71 (87 - 53 = 34)
C&J FT post-editor-WSJ2-21	92.85	91.02 (108 - 14 = 94)	92.49	90.88 (145 - 20 = 125)
Malt	89.20	87.19	89.42	87.55
Malt post-editor-WSJ22	89.20	87.18 (26 - 29 = 3)	89.42	87.45 (14 - 62 = -48)
Malt post-editor-WSJ2-21	89.20	87.19 (15 - 15 = 0)	89.42	87.56 (15 - 11 = 4)
MST	91.02	89.12	90.75	88.94
MST post-editor-WSJ22	91.02	89.11 (20 - 21 = -1)	90.75	88.86 (9 - 56 = -47)
MST post-editor-WSJ2-21	91.02	89.11 (2 - 5 = -3)	90.75	88.94 (4 - 3 = -1)

Table 4: Parser accuracy scores for WSJ 22 and WSJ 23 when LFGDEP is used

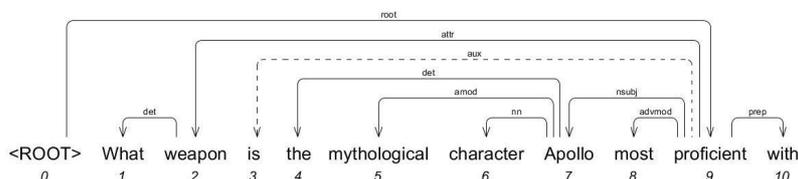


Figure 3: The incorrect gold dependency tree converted by Stanford dependencies

worse and this could be because the dependency parsers are trained on dependency trees which were produced from constituency trees containing null elements and so their output is more accurate than the QuestionBank gold standard. The experiment (described in Section 5.1) which shows that the post-editor only works for the dependency parsers when null elements are removed before training, suggests that this is indeed what is happening. Examination of the post-editing results highlights a similar (albeit much smaller) problem with the Stanford converter: the correct `cop` dependency label for the copular verb in a question such as *Which X is Y?* is replaced by the incorrect `aux` dependency label because the gold Stanford dependency trees are themselves incorrect (see Figure 3 for an example).

There are also many instances in which the gold data is correct and the post-editing method succeeds in correcting labelling errors in parser output. For example, the Stanford relabelling rules manage to correct the mislabelled dependency between the expletive *there* and the main verb in questions such as *How many James Bond novels are there?* from `advmod` to `expl`. An inspection of the LFGDEP

rules show that many correct relabellings are from `subj` to `xcomp` and vice versa in questions of the form *What are/is X?*. We have tracked these parser errors back to the question annotation strategy in the Penn Treebank. According to the Penn Treebank bracketing guidelines (Bies et al., 1995), copular verbs are annotated differently to other main verbs in questions in that they do not introduce a VP node (see Figure 4). Judge et al. (2006) comment that this distinction is difficult for parsers to learn. The fact that the relabelling occurs for the dependency parsers (where the conversion is applied to the gold constituency trees before parser training) as well as the constituency parser (where the conversion is applied to the parser output) suggests that this is not a parser-specific problem and that the gold standard PTB questions contain some noise.⁸

6 Conclusion

We have presented a technique for modifying the labels in a dependency tree and shown that it has con-

⁸ An example is the following tree in *WSJ02*:
 ((S_{BARQ} (“ “) (WHNP-305 (WP What)) (SQ (NP-SBJ (-NONE- *T*-305)) (VP (VBZ is) (NP-PRD (NP (DT the) (NN way)) (ADVP (RB forward))))) (. ?)))

Parser	QuestionDev		QuestionTest	
	UAS	LAS	UAS	LAS
C&J	82.58	78.40	83.62	79.22
C&J post-editor-QDev	82.58	78.72 (41 - 12 = 29)	83.62	79.47(41 - 16 = 25)
C&J FT	82.58	78.41	83.62	79.26
C&J FT post-editor-QDev	82.58	78.73 (41 - 11 = 30)	83.62	79.5(41 - 16 = 25)
Malt	72.59	67.39	74.10	69
Malt post-editor-QDev	72.59	67.65 (56 - 26 = 30)	74.10	69.45(62 - 17 = 45)
MST	74.75	68.9	76.42	70.59
MST post-editor-QDev	74.75	69.62 (99 - 18 = 81)	76.42	71.17(86 - 25 = 61)

(a) Stanford Dependencies

C&J	90.66	68.47	90.99	69.27
C&J post-editor-QDev	90.66	81.34 (1212 - 5 = 1207)	90.99	81.51(1152 - 3 = 1149)
C&J FT	90.78	84.08	91.21	86.9
C&J FT post-editor-QDev	90.78	86.33 (227 - 22 = 205)	91.21	84.81(223 - 30 = 193)
Malt	85.39	66.96	87.08	68.54
Malt post-editor-QDev	85.39	79.37 (1219 - 88 = 1131)	87.08	80.68(1209 - 89 = 1120)
MST	85.29	68.09	87.03	69.64
MST post-editor-QDev	85.29	79.23 (1133 - 113 = 1020)	87.03	67.63(790 - 1043 = -253)

(b) LTH Conversion

C&J	88.47	72.1	88.70	72.46
C&J post-editor-QDev	88.47	81.38 (1017 - 152 = 865)	88.70	81.83 (1041 - 161 = 880)
C&J FT	90.00	82.7	90.43	83.54
C&J FT post-editor-QDev	90.00	85.73 (383 - 109 = 274)	90.43	86.51 (394 - 119 = 275)
Malt	84.89	71.75	85.56	72.61
Malt post-editor-QDev	84.89	78.95 (809 - 155 = 654)	85.56	79.73 (836 - 172 = 664)
MST	85.16	73.06	85.94	74.35
MST post-editor-QDev	85.16	79.52 (751 - 116 = 635)	85.94	71.9 (71 - 297 = -226)

(c) LFGDEP

Table 5: Parser accuracy scores for QuestionDev and QuestionTest

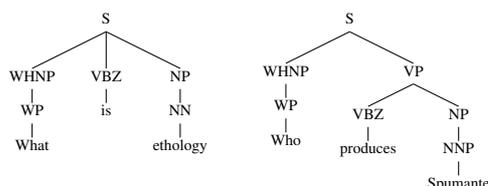


Figure 4: Question Annotation according to PTB Bracketing Guidelines

siderably more success on the Charniak and Johnson reranking parser (for which it brought about statistically significant improvements in accuracy) than on MaltParser and MSTParser. We have also demonstrated how the technique can be used to pinpoint problems in automatic constituency-to-dependency converters. The latter use of the technique is important given the absence of a truly gold dependency test set for English.

In the future we intend to explore the use of the

label post-editing after attachment post-editing. We also intend to explore the extent to which the method can be improved by taking into account label hierarchies and by imposing global constraints.

Acknowledgments

This research has been supported by the Science Foundation Ireland (Grant 07/CE/ I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Dublin City University, School of Computing. We thank Markus Dickinson and the anonymous reviewers for their valuable comments.

References

- Enrique Henestroza Anguiano and Marie Candito. 2011. Resolving difficult syntactic attachments with parse correction. In *Proceedings of EMNLP*.
- Giuseppe Attardi and Massimiliano Ciaramita. 2007.

- Tree revision learning for dependency parsing. In *Proceedings of NAACL-HLT*.
- Ann Bies, Mark Ferguson, Karen Katz, and Robert MacIntyre. 1995. Bracketing guidelines for treebank ii style, Penn Treebank project. Technical Report Tech Report MS-CIS-95-06, University of Pennsylvania, Philadelphia, PA.
- Anton Bryl, Josef van Genabith, and Yvette Graham. 2009. Guessing the grammatical function of a non-root f-structure in LFG. In *Proceedings of IWPT'09*.
- Aoife Cahill, Michael Burke, Ruth O'Donovan, Stefan Riezler, Josef van Genabith, and Andy Way. 2008. Wide-coverage deep statistical parsing using automatic dependency structure annotation. *Computational Linguistics*, 34(1):81–124.
- Özlem Çetinoğlu, Jennifer Foster, Joakim Nivre, Deirdre Hogan, Aoife Cahill, and Josef van Genabith. 2010. LFG without c-structures. In *Proceedings of TLT9*.
- Eugene Charniak and Mark Johnson. 2005. Course-to-fine n-best-parsing and maxent discriminative reranking. In *Proceedings of the 43rd ACL*.
- Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of NAACL*.
- Grzegorz Chrupała, Nicolas Stroppa, Josef van Genabith, and Georgiana Dinu. 2007. Better training for function labeling. In *RANLP 2007*, pages 133–138, Bulgaria.
- Stephen Clark, Mark Steedman, and James R. Curran. 2004. Object extraction and question parsing using ccg. In *Proceedings of EMNLP*.
- Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The stanford typed dependencies representation. In *Proceedings of the COLING Workshop on Cross-Framework and Cross-Domain Parser Evaluation*, August.
- Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*.
- Markus Dickinson. 2008. Ad hoc treebank structures. In *Proceedings of the 46th ACL*, June.
- Markus Dickinson. 2010. Detecting errors in automatically-parsed dependency relations. In *Proceedings of the 48th ACL*, June.
- Jesus Gimenez and Llus Marquez. 2004. Svmtool: A general pos tagger generator based on support vector machines. In *Proceedings of LREC*.
- Yoav Goldberg and Michael Elhadad. 2010. Inspecting the structural biases of dependency parsing algorithms. In *Proceedings of CoNLL*.
- Keith Hall and Václav Novák. 2005. Corrective modeling for non-projective dependency parsing. In *Proceedings of IWPT*, pages 42–52.
- Keith Hall and Vaclav Novak, 2011. *Trends in Parsing Technology*, volume 43 of *Text, Speech and Language Technology*, chapter Corrective Dependency Parsing, pages 151–167. Springer.
- Thorsten Joachims. 1999. Making large-scale svm learning practical. In *Advances in Kernel Methods - Support Vector Learning*. MIT-Press.
- Richard Johansson and Pierre Nugues. 2007. Extended constituent-to-dependency conversion for english. In Joakim Nivre, Heiki-Jaan Kaalep, Kadri Muischnek, and Mare Koit, editors, *Proceedings of NODALIDA*.
- John Judge, Aoife Cahill, and Josef van Genabith. 2006. Questionbank: Creating a corpus of parse-annotated questions. In *Proceedings of the 21st COLING/44th ACL*.
- Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the 1994 ARPA Speech and Natural Language Workshop*.
- Ryan McDonald and Fernando Pereira. 2006. Online learning of approximate dependency parsing algorithms. In *Proceedings of EACL*.
- Ryan McDonald, Koby Crammer, and Fernando Pereira. 2005. Online large-margin training of dependency parsers. In *Proceedings of the 43rd ACL*.
- Ryan McDonald. 2006. *Discriminative Learning and Spanning Tree Algorithms for Dependency Parsing*. Ph.D. thesis, University of Pennsylvania.
- Joakim Nivre, Johan Hall, and Jens Nilsson. 2006. Malt-parser: A data-driven parser-generator for dependency parsing. In *Proceedings of LREC*.
- Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007. The conll 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague, Czech Republic, June.
- Joakim Nivre, Marco Kuhlmann, and Johan Hall. 2009. An improved oracle for dependency parsing with online reordering. In *Proceedings of IWPT'09*.
- Joakim Nivre, Laura Rimmell, Ryan McDonald, and Carlos Gomez-Rodriguez. 2010. Evaluation of dependency parsers on unbounded dependencies. In *Proceedings of COLING*.
- Michel Simard, Cyril Goutte, and Pierre Isabelle. 2007. Statistical phrase-based post-editing. In *Proceedings of HLT-NAACL*.
- Yue Zhang and Stephen Clark. 2008. A tale of two parsers: investigating and combining graph-based and transition-based dependency parsing using beam-search. In *Proceedings of EMNLP*.

Comparing Rule-Based and Data-Driven Dependency Parsing of Learner Language

Julia Krivanek Detmar Meurers

Seminar für Sprachwissenschaft

Universität Tübingen

{krivanek, dm}@sfs.uni-tuebingen.de

Abstract

We explore the performance of two dependency parsing approaches, the rule-based WCDG approach (Foth and Menzel 2006) and the data-driven dependency parser MaltParser (Nivre et al. 2007) on texts written by language learners.

We show that WCDG outperforms MaltParser in identifying the main functor-argument relations, whereas MaltParser is more successful than WCDG in establishing optional, adjunct dependency relations. This can be interpreted as a tradeoff between the rich, hand-crafted lexical resources capturing obligatory argument relations in WCDG and the ability of a data-driven parser to identify optional, adjunct relations based on the linguistic and world knowledge encoded in the gold-standard training corpora.

1 Introduction

Texts written by language learners provide an interesting test case for parsing. They include significant well-formed and ill-formed variation in forms highlighting the robustness of the syntactic analysis performed by different parsing approaches and the resources they use. Dependency parsing is an attractive option in this context, given its focus on the lexical dependency structure serving as interface to interpretation, which avoids further commitments inherent in elaborate constituency-based representations. Parsing learner language is a foundation for any kind of deeper analysis of learner language, as, e.g., needed for automatic content-assessment (Meurers et al. 2011).

Ott and Ziai (2010) describe a dependency parsing experiment based on texts written by American college students learning German. To obtain a

gold standard test set, Ott and Ziai (2010) manually annotated this learner corpus using the German dependency annotation scheme developed by Foth (2006) using multiple annotators. For the parsing experiment, they used the data-driven MaltParser (Nivre et al. 2007). They trained this parser on the fifth release of the TüBa-D/Z treebank (Telljohann et al. 2004), after converting it into a dependency treebank format in the style of Foth (2006) using the conversion procedure described in Versley (2005). The TüBa-D/Z treebank consists of newspaper articles, so that there is a significant difference between the training and the test corpus they used. Despite this difference, Ott and Ziai (2010) report that the MaltParser as one of the best current data-driven dependency parsing approaches reliably identified the main functor-argument relation types with a relatively high precision and recall in the 80-90%.

While this is an encouraging result for tasks requiring dependency analysis of learner language, it made us wonder about the impact of the parsing method. In this paper, we therefore explore how parsing of learner language with the data-driven MaltParser compares to parsing with a dependency parser using hand-written rules, for which we make use of the German WCDG parser (Foth and Menzel 2006). Grammar-based parsing with WCDG is based on an information-rich, hand-crafted lexicon (Foth 2006, ch. 2.2). This lead us to hypothesize that the subcategorization requirements hand-coded in the lexicon will contribute to a high-quality coverage of the specific argument requirements of a lexical item. In terms of dependency parsing, this would predict that the rule-based approach in comparison to the data-driven one will fare better in detecting the core functor-argument relations, such as subject and object dependencies. On the other hand, for the subtle distributional differences of adjunct relations, for which relatively few specific constraints are im-



posed by theoretical linguistic models, a statistical approach trained on corpora – which by their nature encode a combination of language competence, use, and world knowledge – may well fare better.

2 Parsing experiments: The setup

Learner language test corpus We base our parsing experiments on the learner corpus of Ott and Ziai (2010). It is a sub-corpus of the Corpus of Reading Comprehension Exercises in German (CREG, Meurers et al. 2010), which we will refer to as CREG-109. It consists of 109 sentences representing answers to reading comprehension exercises written by US college students at the beginner and intermediate levels of German programs. An example for a learner answer (LA) from the CREG-109 corpus is shown below, where we also show the reading comprehension question (Q) and the teacher’s target answer (TA) (but for space reasons not the reading text itself).

Q: *Warum sollte er nicht lachen?*

Why should he not laugh?

TA: *Er sollte nicht lachen, weil das Kind schläft.*

He should not laugh because the baby is sleeping.

LA: *Er sollte nicht lachen für das schlafende Baby.*

He should not laugh for the sleeping baby.

Ott and Ziai (2010) semi-automatically annotated the corpus with STTS part-of-speech tags (Thielen et al. 1999) by running TreeTagger (Schmid 1994) followed by a manual correction phase. On this basis, they manually annotated the corpus according to the dependency annotation scheme devised by Foth (2006), relying on three annotators for each sentence and adjudication for any disagreement to ensure a high quality annotation.

Training MaltParser For data-driven parsing, we essentially followed the setup of Ott and Ziai (2010). We used MaltParser, a system for transition-based dependency parsing (Nivre et al. 2007). The system supports inducing a parsing model from a corpus which has been annotated with dependencies and to parse previously unseen data using the induced model. For **training MaltParser**, we used 90% of the dependency treebank version of the TüBa-D/Z treebank (Telljohann et al. 2004), a corpus consisting of German

news texts for which dependency representations in the style of Foth (2006) were obtained with the help of the conversion procedure described in Versley (2005). Training was performed using the LIBSVM learning algorithm and 2-Planar Arc-Eager transition system (Gómez-Rodríguez and Nivre 2010), a linear-time algorithm which is capable of handling limited non-projectivity. The resulting parsing model was used for all MaltParser results reported on in this paper.

Native language test corpus We used sentences from the remaining 10% of the TüBa-D/Z dependency treebank as a **benchmark test corpus** to be able to identify the effect of text type and the impact of parsing learner language in contrast to native language – in line with the well-known fact that parser performance is text type dependent and some text types are more difficult to parse than other ones (Versley 2005). To ensure effective parsing with WCDG, we removed 8% of the sentences, which had character set encoding problems and lexical coverage issues, resulting in a test set of 4142 sentences which we will refer to as TüBa-D/Z test corpus.

The WCDG parser integrates a statistical POS tagger (TnT, Brants 2000) and cannot easily be provided with input including gold standard tags. To make the input to both parsers identical, we thus ran the TnT tagger using the STTS tagset on the CREG-109 and the TüBa-D/Z test corpora and used these automatically tagged version as input for parsing with the MaltParser.

WCDG Parser The WCDG parser representing rule-based dependency parsing in our experiments is an implementation of weighted constraint dependency parsing for German (Foth and Menzel 2006). The WCDG parser allows constraints to express any formalizable property of a dependency tree and the weights for constraints were assigned manually. Parsing with such a WCDG is NP-complete and thus can result in non-termination and efficiency problems. Instead of a full search (*netsearch*) we thus selected *frobbing* as a heuristic search option. Efficiency still remains an issue, so that for our experiments we used the hybrid version of the WCDG parser (Foth and Menzel 2006), which together with a rule-based dependency grammar makes use of a chunker, a supertagger, and a probabilistic shift-reduce parser for labeled dependency trees as stochastic

predictor components. While the overall WCDG system successfully tackles parsing of the learner language and the native language test corpus and provides some interesting results, which we now turn to, efficiency clearly is not competitive with statistical dependency parsing, with parse times of several minutes for CREG-109 and several days for the TüBa-DZ test set.

3 Results

3.1 Quantitative evaluation

For the quantitative evaluation we used the *eval.pl* tool from the CoNLL-X shared task on dependency parsing (Buchholz and Marsi 2006). Table 1 sums up the labeled (LAS) and unlabeled (UAS) attachment scores obtained for parsing the CREG-109 and TüBa-DZ test sets with the MaltParser, and Table 2 shows the WCDG parser results.

	LAS	UAS	$\delta(\text{LAS}, \text{UAS})$
TüBa-D/Z	84.04%	87.25%	3.21%
CREG-109	78.12%	84.56%	6.44%
			= 3.23% diff.

Table 1: MaltParser results

	LAS	UAS	$\delta(\text{LAS}, \text{UAS})$
TüBa-D/Z	81.42%	85.71%	4.29%
CREG-109	79.28%	86.36%	7.08%
			= 2.79% diff.

Table 2: WCDG results

Looking at the results for the learner language test corpus CREG-109, we find that both parsers achieve similar overall results. The WCDG results for parsing the native TüBa-D/Z test corpus for the labeled case are slightly better than for the CREG-109 learner corpus (2.14%), whereas for the unlabeled case the performance for the learner corpus is slightly better (0.65%), probably due to the more complex nature of the TüBa-D/Z news sentences. The linguistic generalizations manually encoded in the WCDG grammar thus appear to be surprisingly applicable to the learner language properties, resulting in a robust parsing performance. MaltParser, on the other hand, with a drop of 5.92% in labeled and 3.31% in unlabeled dependency results between native and learner data shows more clearly that it was trained on the native language news corpus TüBa-D/Z and thereby learned specifics of language and text type which

do not generalize that well to reading comprehension answers written by language learners.

An interesting issue arises when one investigates the clear drop between the labeled (LAS) and the unlabeled attachment (UAS) results which arises for both parsers. This drop is significantly larger for the CREG-109 learner corpus than for the native TüBa-DZ corpus, which was also observed by Ott and Ziai (2010) in their CREG-109 parsing experiments with the MaltParser. They hypothesize that this gap may result from the presence of ungrammatical sentences in the corpus.

We investigated this hypothesis for the WCDG parsed CREG-109 corpus by manually inspecting all the relations which were correctly detected but assigned false labels. In other words, we inspected the 53 cases where the parser assigned correct relations but false labels, causing the 7.08% difference between the LAS (79,28%) and UAS (86,36%) results in WCDG parsing CREG-109. We found that 21 of these relations received false labels as the result of an ungrammaticality related to that dependency. We tested this by parsing a corrected version of the sentence with WCDG and observing that the parser then assigned the proper label for the dependency in question. Out of the 53 correctly identified relations with false labels (7.08% of all errors), the 21 cases correspond to 2.8% of all errors which are the result of ungrammaticality. This corresponds exactly to the 2.79% difference in $\delta(\text{LAS}, \text{UAS})$ between native and learner corpora results of WCDG, fully confirming the hypothesis.

As an example, consider the sentence in (1) and its WCDG parse in Figure 1.

- (1) *Sein Eltern hat BA geholfen.*
his parents has BA helped

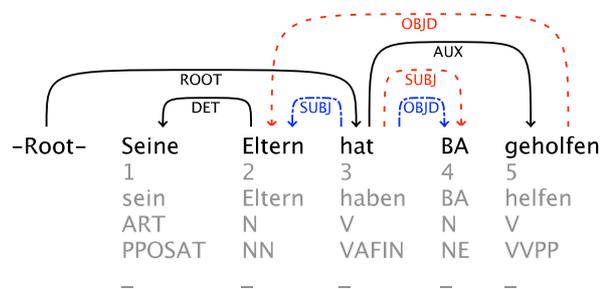


Figure 1: WCDG parser fails to identify subject and object due to subject-verb agreement error

The black solid lines represent correct dependencies identified by the parser, the red dashed de-

dependencies are dependencies incorrectly posited by the parser, and the blue dash-dotted lines are correct dependencies which were not identified by the parser. The learner used a third person singular verb form *hat* (*has*), which causes the parser to reject the plural *Eltern* (*parents*) as subject (despite the singular article *sein* (*his*)) and to label the dependency with *BA* as the subject instead of as a dative object.

Results for different dependency types To investigate the hypothesis formulated in the introduction that the different parsing approaches will show differences in the way they handle argument compared to the way they handle adjunct relations, we need to take a closer look at the two sets of labeled dependency types. The dependency annotation scheme of Foth (2006) distinguishes a range of argument relations. Given the small size of the CREG-109 corpus, we here focus on the most common ones, for which we have over 10 instances each: SUBJ (subject), OBJA (accusative object), PRED (predicate), and AUX (argument of auxiliary verb). Among the adjuncts, the most common ones are ADV (adverbial modifier) and PP (prepositional adjunct). Note that Foth (2006) uses the labels PP and ADV for grammatical *functions* (adjunct, modifier), different from the typical usage of those labels for grammatical *categories*.

Table 3 shows the results by dependency type for both parsers in percentage figures for precision and recall. The numbers in bold are the best results for a given dependency type.

Label	#	MaltParser		WCDG	
		Recall	Prec.	Recall	Prec.
Argument relations					
SUBJ	95	84.21	80.00	87.37	86.46
OBJA	52	65.38	70.83	75.00	75.00
PRED	26	61.54	69.57	57.69	83.33
AUX	23	60.87	87.50	73.91	94.44
Modifier relations					
ADV	44	65.91	56.86	65.91	48.33
PP	32	75.00	55.81	71.88	43.40
Coordination relations					
KON	49	63.27	67.39	67.35	76.74
CJ	39	82.05	86.49	89.74	92.11

Table 3: CREG-109 results for the most common argument and adjunct dependency types

We see that in line with our hypothesis, the WCDG parser performs better for each of the lex-

ically subcategorized arguments, the *subject*, *accusative object* and *predicative complements*, and *auxiliary verbal complements* dependencies. The data-driven MaltParser, on the other hand, performs better in identifying *adverbial modifiers* and *prepositional adjuncts*.

The two coordination relations CJ (conjunct) and KON (non-final coordination conjunct) are a special case, because coordinated elements can function as adjuncts or as arguments. We thus manually inspected the coordination relations in the CREG-109 corpus and found that only 3 (about 6 %) of the KON relations in this corpus involve adjuncts. The fact that WCDG parser here outperforms the MaltParser on KON thus also confirms the hypothesis that WCDG is better in detecting argument relations. In the same vein, for the CJ relation the only case where WCDG performed worse than MaltParser is an adjunct case.

The CREG-109 corpus is very small, though, and the small number of instances for each dependency type (shown in the # column) should caution us against overinterpreting these results. On the other hand, we can take a closer look at the parsing results for the larger, native TüBa-D/Z test corpus to see whether we can obtain further support for the interpretation. Table 4 shows the results of parsing the TüBa-D/Z test corpus.

Label	#	MaltParser		WCDG	
		Recall	Prec.	Recall	Prec.
Argument relations					
SUBJ	5408	83.54	87.05	89.00	89.64
OBJA	2658	75.43	72.96	79.83	82.15
PRED	1044	66.48	71.77	60.82	76.51
AUX	2236	85.73	89.41	91.77	96.11
Modifier relations					
ADV	5115	78.92	77.78	69.72	64.13
PP	5562	71.88	72.26	69.67	62.92
Coordination relations					
KON	2531	76.37	71.70	62.90	71.42
CJ	2164	90.48	91.41	86.18	83.74

Table 4: TüBa-D/Z results for the most common argument and adjunct dependency types

The table confirms the picture we found for the learner corpus. Again, the WCDG parser obtained the better precision and recall figures for the argument relations (with one exception, the recall of the PRED relation). This is particularly remarkable since we trained the MaltParser on (the 90% development subset of) the TüBa-D/Z cor-

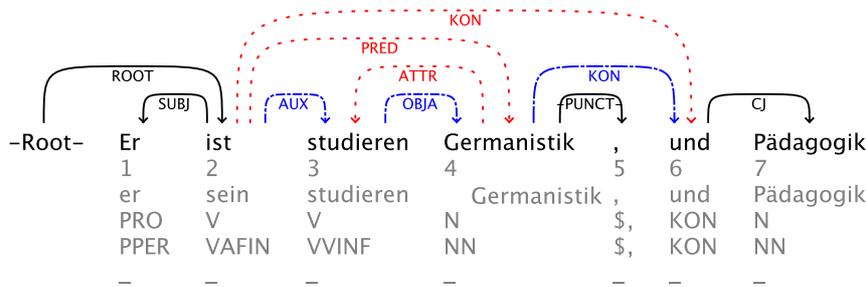


Figure 2: MaltParser: wrong analysis of ill-formed auxiliary verb dependency

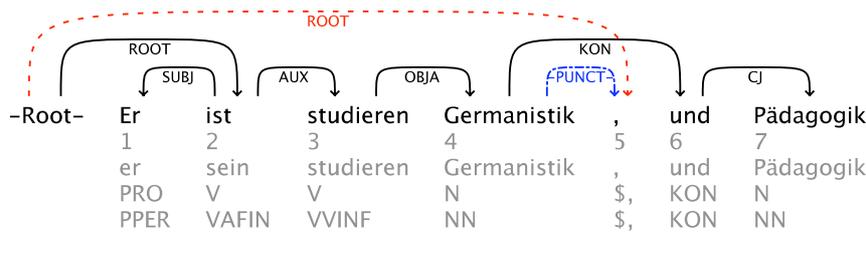


Figure 3: WCDG parser: correct analysis of ill-formed auxiliary verb dependency

pus, which should give it an advantage when parsing the TüBa-D/Z test set. It does indeed improve the results compared to those for the CREG-109, but not enough to overtake the WCDG parser, where the hand-specified lexical subcategorization information apparently is sufficient to maintain an edge. For the modifier relations, on the other hand, the MaltParser significantly outperformed the WCDG parser as expected, confirming the hypothesis that a data-driven parser is better at capturing the characteristics of optional, adjunct relations from those observed in the training data.

The TüBa-D/Z results for the coordination relations KON and CJ, however, are the inverse of the ones we obtained for the CREG-109 corpus. This could be due to the proportion of arguments and adjuncts which are coordinated in the TüBa-D/Z corpus, where the number of coordinated adjuncts is predicted to be higher than in the learner corpus.

3.2 Aspects of a qualitative analysis

Complementing the quantitative analysis, we performed a qualitative inspection of the results obtained for the CREG-109 learner corpus to gain a better understanding of the problems which arise in parsing learner language and how the two parsers differ in this respect.

WCDG: robust parsing of ill-formed AUX An interesting aspect of the results in Tables 3 and 4 is that the scores for identifying arguments of aux-

iliary verbs are particularly high for the WCDG parser compared to MaltParser, which raises the question why this is the case.

Example (2) illustrates a case where the ungrammatical combination of *ist* (*is*) with *studieren* (*study*) prevented the MaltParser from identifying an AUX relation, as shown in Figure 2.

- (2) *Er ist studieren Germanistik und Pädagogik.*
 he is study German and Pedagogy

The target form of the learner most likely was the English progressive *is studying*, which does not exist as such in German; alternatively, if the learner targeted a perfect tense construction, he chose the wrong auxiliary for this verb and the wrong form for the verbal complement.

Figure 3 shows that the WCDG parser did identify an AUX dependency. Inspection of the WCDG grammar showed that this happens because the WCDG grammar licenses a particular type of passive where the auxiliary *ist* combines with a *zu*-infinitive. For the example (2), the WCDG parser penalized the absence of the particle *zu*, but still this (incorrect) passive analysis achieved the highest weight so that the relation between *ist* and *studieren* was labeled AUX, the most meaningful way to connect these two verbs.

WCDG: robust parsing of subjectless sentences Another interesting issue arises around the analysis of subjects. Example (3) shows an ungrammat-

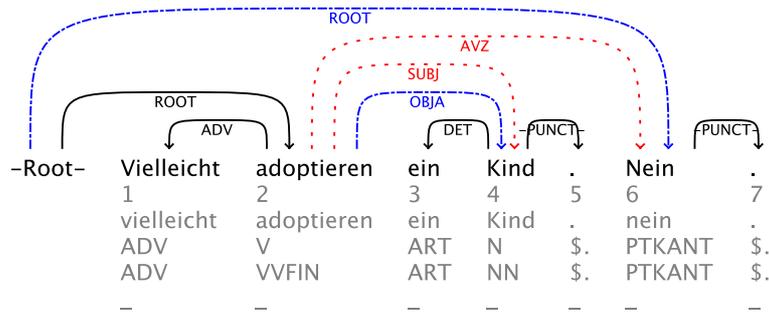


Figure 4: MaltParser: wrong analysis of a sentence missing the subject

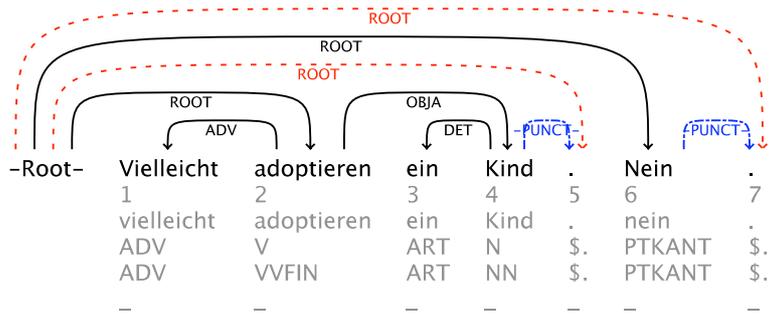


Figure 5: WCDG parser: correct analysis of a sentence missing the subject

ical learner sentence in which the subject is missing (which, different from English, is possible in German in specific cases).

- (3) *Vielleicht adoptieren ein Kind. Nein.*
 perhaps adopt_{plur} a child no

The analysis of the MaltParser is shown in Figure 4. It posits a subject relation between the finite verb *adoptieren* (*adopt*) and *Kind* (*child*), despite the fact that this relation violates subject-verb agreement between the singular *Kind* and the plural *adoptieren*. Essentially, MaltParser always attempts to identify a subject in a sentence.

Figure 5 shows the WCDG analysis for this sentence. In the WCDG grammar there are restrictions on the cases when a subject can be ordered to the right of a predicate. The sentence under discussion is not among those cases. Thus *Kind* is correctly identified as a object in an overall analysis of a sentence missing a subject.

Naturally, the generalizations captured in the WCDG grammar do not always succeed in balancing the evidence and regularities appropriately for learner language. Example (4), another sentence with a missing subject, is a case in point.

- (4) *Rockmusik hören und Mundharmonika spielen*
 rock music hear and mouth harp play

Figure 6 shows that in contrast to the sentence we saw in Figure 5, the WCDG parser did not recognize that the subject is missing here and just like the MaltParser labeled the relations between the verbs and the corresponding nouns as subject instead of as objects. The reason for this is a word order rule in the grammar, where a subject in front of a predicate is weighted higher than an object.

4 Conclusion

The aim of this paper was to investigate and compare the performance of a data-driven and a rule-based dependency parsing approach for a learner corpus and a native control corpus. In pursuit of this goal, we reported on parsing experiments with the MaltParser and the WCDG parser for the CREG-109 and a subset of the TüBa-DZ and reported overall results as well as an analysis in terms of the main argument and adjunct relations.

The results highlighted the different strengths of the two parsing approaches. The rich lexical resources of the WCDG parser apparently provide an advantage for identifying the main functor-argument relations, whereas the capability of the supervised machine learning approach of the MaltParser to identify subtle statistical differences in the training data seems to give it an edge in the analysis of optional, adjunct relations.

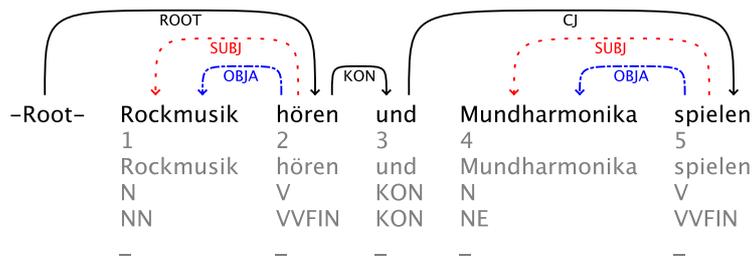


Figure 6: Wrong analysis of both parsers due to word order

In the broader context, this insight essentially lends support to the pursuit of hybrid approaches and parser combinations (e.g., Khmylko et al. 2009; Øvrelid et al. 2009). The learner language domain poses additional challenges to the prioritization of different sources of information is important given that certain language properties are known not to be reliably realized by language learners. While in this paper we have focused on comparing data-driven and rule-based dependency parsing of learner language, an underlying issue which requires more attention is what exactly a dependency analysis of learner language should look like, which has started to receive some attention (Dickinson and Ragheb 2009; Rosén and Smedt 2010; Hirschmann et al. 2010). As far as we see, the criteria crucially depend on the purpose of the analysis, so different types (or multiple layers) of dependency analysis will be needed. On the one hand, a robust dependency analysis glossing over any learner language specifics is needed as a step towards robustly building meaning representations and related processes in applications. On the other hand, detailed dependency analyses based on the various types of evidence that are available when interpreting learner data (morphological, syntactic, and semantic evidence in the data itself, and information about the learner and the task for which the language was produced) could be particularly useful for identifying specific learner language aspects as part of research investigating second language acquisition.

In terms of outlook, while the analysis of the parsing results for the small CREG-109 learner corpus is fully supported by the results obtained for the larger TüBa-DZ test corpus, we would like to extend the analysis to more argument and adjunct relations, for which a larger learner corpus is needed. A larger release of CREG data will become available so that we plan to tackle an extended evaluation based on that larger data set.

Acknowledgments

We would like to thank Yannick Versley, Niels Ott, Ramon Ziai and the anonymous Depling reviewers for their helpful suggestions and discussion.

References

- Thorsten Brants, 2000. TnT – A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*. Association for Computational Linguistics, pp. 224–231.
- Sabine Buchholz and Erwin Marsi, 2006. CoNLL-X shared task on multilingual dependency parsing. In *CoNLL-X '06: Proceedings of the Tenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, Morristown, NJ, USA, pp. 149–164.
- Markus Dickinson and Marwa Ragheb, 2009. Dependency Annotation for Learner Corpora. In *Proceedings of the Eighth Workshop on Treebanks and Linguistic Theories (TLT-8)*. Milan, Italy.
- Kilian Foth, 2006. *Eine umfassende Constraint-Dependenz-Grammatik des Deutschen*. Technical report, Universität Hamburg.
- Kilian A. Foth and Wolfgang Menzel, 2006. Hybrid parsing: using probabilistic models as predictors for a symbolic parser. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACL-44, pp. 321–328.
- Carlos Gómez-Rodríguez and Joakim Nivre, 2010. A Transition-Based Parser for 2-Planar Dependency Structures. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. pp. 1492–1501.

- Hagen Hirschmann, Anke Lüdeling, Ines Rehbein, Marc Reznicek and Amir Zeldes, 2010. Syntactic Overuse and Underuse: A Study of a Parsed Learner Corpus and its Target Hypothesis. Invited Plenary Talk at the Treebanks and Linguistic Theory Workshop.
- Lidia Khmylko, Kilian A. Foth and Wolfgang Menzel, 2009. Co-Parsing with Competitive Models. In *Proceedings of the 11th International Conference on Parsing Technologies (IWPT'09)*. Association for Computational Linguistics, Paris, France, pp. 99–107.
- Detmar Meurers, Niels Ott and Ramon Ziai, 2010. Compiling a Task-Based Corpus for the Analysis of Learner Language in Context. In *Proceedings of Linguistic Evidence*. Tübingen, pp. 214–217.
- Detmar Meurers, Ramon Ziai, Niels Ott and Stacey Bailey, 2011. Integrating Parallel Analysis Modules to Evaluate the Meaning of Answers to Reading Comprehension Questions. *IJCELL. Special Issue on Automatic Free-text Evaluation*, 21(4):355–369.
- Joakim Nivre, Jens Nilsson, Johan Hall, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov and Erwin Marsi, 2007. Malt-Parser: A Language-Independent System for Data-Driven Dependency Parsing. *Natural Language Engineering*, 13(1):1–41.
- Niels Ott and Ramon Ziai, 2010. Evaluating Dependency Parsing Performance on German Learner Language. In Markus Dickinson, Kaili Müürisepp and Marco Passarotti (eds.), *Proceedings of the Ninth International Workshop on Treebanks and Linguistic Theories*. volume 9 of *NEALT Proceeding Series*, pp. 175–186.
- Victoria Rosén and Koenraad De Smedt, 2010. Syntactic Annotation of Learner Corpora. In Hilde Johansen, Anne Golden, Jon Erik Hagen and Ann-Kristin Helland (eds.), *Systematisk, variert, men ikke tilfeldig. Antologi om norsk som andrespråk i anledning Kari Tenfjords 60-årsdag [Systematic, varied, but not arbitrary. Anthology about Norwegian as a second language on the occasion of Kari Tenfjord's 60th birthday]*, Novus forlag, Oslo, pp. 120–132.
- Helmut Schmid, 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*. Manchester, UK, pp. 44–49.
- Heike Telljohann, Erhard Hinrichs and Sandra Kübler, 2004. The TüBa-D/Z Treebank: Annotating German with a Context-Free Backbone. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*. Lissabon.
- Christine Thielen, Anne Schiller, Simone Teufel and Christine Stöckert, 1999. *Guidelines für das Tagging deutscher Textkorpora mit STTS*. Technical report, Institut für Maschinelle Sprachverarbeitung Stuttgart and Seminar für Sprachwissenschaft Tübingen.
- Yannick Versley, 2005. Parser Evaluation across Text Types. In *Proceedings of the Fourth Workshop on Treebanks and Linguistic Theories (TLT)*. Barcelona, Spain.
- Lilja Øvrelid, Jonas Kuhn and Kathrin Spreyer, 2009. Improving data-driven dependency parsing using large-scale LFG grammars. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACLShort '09, pp. 37–40.

Rule-Based Dependency Parser Refined by Empirical and Corpus Statistics

Igor Boguslavsky

Institute for Information Transmission
Problems, Russian Academy of
Sciences, Moscow / Universidad
Politécnica de Madrid

igor.m.boguslavsky@gmail.com

Victor Sizov

Institute for Information Transmission
Problems, Russian Academy of
Sciences, Moscow

sizov@iitp.ru

Leonid Iomdin

Institute for Information Transmission
Problems, Russian Academy of
Sciences, Moscow

iomdin@iitp.ru

Leonid Tsinman

Institute for Information Transmission
Problems, Russian Academy of
Sciences, Moscow

cinman@iitp.ru

Vadim Petrochenkov

Institute for Information
Transmission Problems, Russian
Academy of Sciences, Moscow

vadim.petrochenkov@gmail.com

Abstract

The paper presents a large-coverage rule-based dependency parser for Russian, ETAP-3, and results of its evaluation according to several criteria.

The parser takes a morphological structure of a sentence processed as input and builds a dependency tree for this sentence using a set of syntactic rules. Each rule establishes one labeled and directed link between two words of a sentence that form a specific syntactic construction. The parser makes use of about 65 different syntactic links. The rules are applied by an algorithm that at first builds all possible hypothetical links and then uses a variety of filters to delete excessive links so that the remaining ones form a dependency tree. Several types of data collected either empirically or from a syntactically tagged corpus of Russian, SynTagRus, are used at this filtering stage to refine the parser performance.

The parser utilizes a highly structured 120,000-strong Russian dictionary, whose entries contain detailed descriptions of syntactic, semantic and other properties of words. A notable proportion of the links in the output trees are non-projective.

An important feature of the parser is its ability to produce multiple parses for the same sentence. In a special mode of

operation, the parser may be instructed to produce more parsing outputs in addition to the first one. This can be done automatically or interactively.

In the evaluation, SynTagRus is viewed as a gold standard. Evaluation results show the figures of 0.900 for unlabelled attachment score, 0.860 for labeled attachment score, and 0.492 for unlabeled structure correctness.

1 Introductory Remarks

The syntactic parser, developed by a research team of the Institute for Information Transmission Problems in Moscow for a multipurpose linguistic processor, ETAP-3 (see e.g. Apresjan *et al.* 2003) is in many respects based on the general linguistic framework of the Meaning \Leftrightarrow Text theory, proposed by Igor Mel'čuk (e.g. Mel'čuk 1974) – especially the syntactic component of this theory. The parser is fully operational for two languages: English and Russian.

In this paper, the Russian option of the parser will be considered. Within the ETAP-3 linguistic processor, it is used in a number of applications, including Russian-to-English machine translation and the tagger for the syntactic annotation of a Russian text corpus SynTagRus.



2 Morphological Analyzer

During text analysis, the parser proper operates after the **morphological analyzer** has processed the text sentence by sentence and produced a **morphological structure** (MorphS) for each sentence. MorphS is the ordered sequence of all words of a sentence, each one represented by a lemma name, a POS attribute and a set of morphological features. The morphological analyzer works, essentially, with individual words, with a relatively few number of cases where a collocation (like *vse ravno* ‘all the same’) or a compound preposition (like *so storony* ‘on the part of’) are viewed as indivisible words. If a word form is lexically and/or morphologically ambiguous, it appears in the MorphS as a set of objects, somewhat loosely called **homonyms**, each consisting again of a lemma name, a POS attribute and a set of morphological features.

To give an example, the sentence

(1) *Inostrannye rabočie často ploxo znajut russkij jazyk* (lit. *foreign workers often badly know Russian language*) ‘Foreign workers often have a poor knowledge of Russian’

will yield the following MorphS:

1.1	INOSTRANNYJ	A,NOM,PL
1.2	INOSTRANNYJ	A,ACC,INANIM,PL
2.1	RABOČIJ1	A,NOM,PL
2.2	RABOČIJ1	A,ACC,INANIM,PL
2.3	RABOČIJ2	N,NOM,PL,MASC,ANIM
3.1	ČASTYJ	A,SG,SHORT,NEUT
3.2	ČASTO	ADV
4.1	PLOXOJ	A,SG,SHORT,NEUT
4.2	PLOXO	ADV
5.1	ZNAT’	V,NONPAST,NONPERF,PL,3P
6.1	RUSKIJ1	A,NOM,SG,MASC
6.2	RUSKIJ1	A,ACC,INANIM,SG,MASC
6.3	RUSKIJ2	N,NOM,SG,MASC,INANIM
7.1	JAZYK1	N,NOM,SG,MASC,INANIM
7.2	JAZYK1	N,ACC,SG,MASC,INANIM
7.3	JAZYK2	N,NOM,SG,MASC,INANIM
7.4	JAZYK2	N,ACC,SG,MASC,INANIM
7.5	JAZYK3	N,NOM,SG,MASC,ANIM

Here, A, ADV, N, and V denote, respectively, the adjective, adverb, noun and verb; NOM and ACC stand for the nominative and the accusative cases; SG and PL mark the singular and plural numbers. MASC and NEUT denote the masculine and the neutral gender. SHORT represents the short form of the adjective. ANIM and INANIM represent

the animateness/inanimateness of adjectives and nouns. NONPAST, NONPERF and 3P show the present tense, the imperfective aspect and the third person of the verb.

As it happens, all words of (1) except word 5 (the verb ‘know’) are ambiguous. In particular, word 6 is lexically ambiguous between adjective ‘Russian’ and noun ‘the Russian’, both varying in case marking; words 3 and 4 may both be interpreted as adverbs (‘often’, ‘badly’) or adjectives (‘frequent’, ‘bad’), whilst word 7 has three lexical readings corresponding to ‘language’, ‘tongue’, and ‘prisoner’, of which the former two, being inanimate, have the same forms for the nominative and the accusative case.

Accordingly, (1) consisting of 7 words has a MorphS that has as many as 18 homonyms.

The morphological analyzer is based on a comprehensive morphological dictionary of Russian that counts about 130,000 entries (over 4 million word forms).

ETAP-3 parser does not have a separate POS tagger; however, there is a small post-morphological module that partially resolves lexical and morphological ambiguity taking account of near linear context. In the case of sentence (1), this module will only delete 2 homonyms and reduce the strength of one more. On average, the module purges about 20% of homonyms.

3 The Parser

3.1 Parser Essentials

The syntactic analyzer takes a MorphS of a sentence processed as input and builds a dependency tree for this sentence using a set of syntactic rules, or **syntagms**. Each syntagm is a rule designed to establish one labeled and directed link between two words of a sentence that form a specific syntactic construction: in other words, any syntagm produces a minimal subtree that consists of two words and a link between them. There are 65 different syntactic links; e.g. the predicative link marks the domination by a finite verb [X] of its subject [Y], as in *John* [Y] *sees* [X]; the 1st completive link represents the relation between a predicate word as head and a word instantiating its 2nd valency as daughter, as in *sees* [X] *light* [Y] or *aware* [X] *of* [Y] (*my presence*), etc. Syntagms are used by the parsing algorithm that starts by building all possible hypothetical links and then uses a

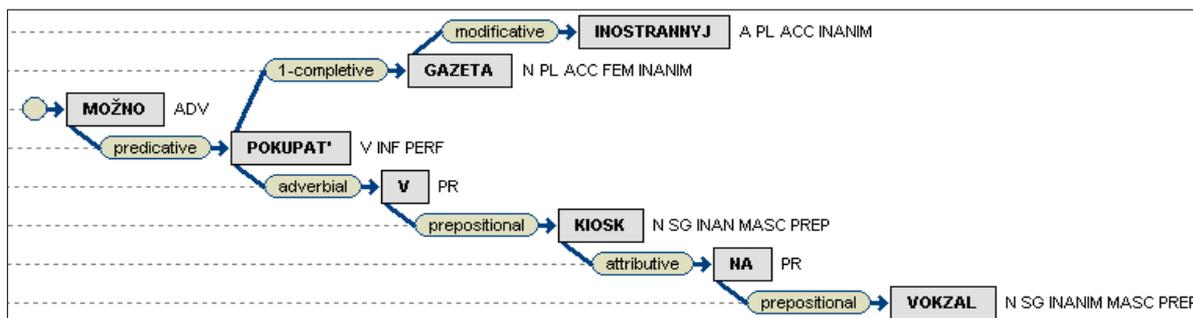


Figure 1: The dependency tree for sentence (2)

variety of filters to delete excessive links so that the remaining ones form a dependency tree.

These filters are of diverse nature and may involve data on agreement or government, repeatability/non-repeatability of specific syntactic relations (e.g. a verb may have several adverbial modifiers attached by the adverbial relation but only one subject or one direct object attached by the predicative or 1st completive relation¹), data on link projectivity (by default, any link is projective unless a set of specific conditions are met²).

Fig. 1 above shows the dependency tree of the sentence

(2) *Inostrannye gazety možno kupit v kioske na vokzale* ‘One can buy foreign newspapers at a news-stand in the railway station’.

We can see that the 1st completive link going from the verb *pokupat* ‘buy’ to the noun *gazeta* ‘newspaper’ is non-projective as it crosses the projection of *možno* ‘one can’, which is the absolute head of the tree.

The parser makes use of a highly structured 120,000-strong Russian dictionary, whose entries contain detailed descriptions of syntactic, semantic and combinatorial properties of words.

An important feature of the parser is its ability to produce multiple parses for the same sentence. While every effort is made to ensure that the first parse obtained adequately reflects the structure of the sentence, this is not always the case. In the supervised mode of operation, the parser may be instructed to produce more parses in addition to the first one if it is

¹ In case of subject/object coordination, only one predicative or 1st completive relation is established between the predicate and the head of the coordination string (the leftmost member of this string).

² It turns out that even though a notable proportion of the links in dependency trees are non-projective (averagely, about 10% of processed sentences contain at least one non-projective link), the share of such links in the total amount of produced links is less than 1%.

unsatisfactory (the first parse may be outright wrong or, in the case of a genuinely ambiguous sentence, it may correspond to a different interpretation than that expected for the text processed) This can be done automatically or interactively, with a targeted choice of word and/or link interpretations (cf. Boguslavsky *et al.* 2005).

The parser operates in a sufficiently robust way: in the worst case, if no adequate tree can be obtained for a sentence, some of its words are linked by a soft-fail **fictitious** syntactic relation. Words that could not be found in the dictionary receive a special POS attribute **NID** (non-identified word).

Normally, each node in the resulting tree corresponds to one word of the sentence parsed. Exceptions are cases where a word is a composite not assigned a dictionary entry (such as *vos'mitomnyj* ‘eight-volume), for which the parser produces two (or more) nodes in the dependency tree.

3.2 Empirical Refinement: Intersynt Duplicates of Syntagms

At the filtering stage of the analysis algorithm, two different modules can be additionally involved in order to improve the performance of the parser.

The first module (cf. Tsinman and Druzhkin 2008) is based on close empirical observation of parsed linguistic material. Basically, it implements the idea that close links between the words, especially those responsible for the core, or “skeleton”, structure of the sentence, have a noticeably higher occurrence than the respective long-distance links.

In order to account for this fact, the most important syntagms (around 60 of the total number of over 200) were replicated in simpler rules that do not establish any links but increase or diminish the priority of links already established. These new rules, called Intersynt rules, work after the syntagms have

been applied and check the bulk of the conditions specified in syntagms (disregarding some of the most subtle ones) but, unlike syntagms, operate within a very narrow space of the sentence (usually at a distance of no more than 4 words). As a result, many close links are reliably confirmed and remain in the sentence tree.

3.3 Data-Driven Statistical Refinement: Statistics of the Tagged Corpus of Texts

The data-driven statistical module (cf. Petrochenkov and Sizov 2010) collects the statistics of links from the syntactically annotated corpus of Russian texts, to be described in more detail in Section 5 below. Statistical data represent the distribution of syntactic links in the treebank that takes account of the following three factors: 1) the distance between the words, 2) the direction of the link (from left to right or vice versa), and 3) the number of the word sense of the word involved in the link (normally, lexical meanings of polysemantic words are ordered in the dictionary in such a way that the more general and more frequently used meanings have smaller numbers than the peripheral meanings). The statistical module intervenes at the moment when the parser chooses among the established competing syntactic hypotheses for an undecided syntactic daughter and prompts the algorithm to select the link occurring in the tagged corpus in similar environment with the maximum frequency.

In different modes of operation, the parser may use either of the two modules, both of them, or neither. The use of the empirical module turned out to provide a noticeable improvement to parser performance as compared to the “bare” parser.

4 The Corpus

The ETAP-3 parser is used to construct the first Russian dependency treebank, SynTagRus (Boguslavsky *et al.* 2000, 2009; Apresjan *et al.* 2006). Currently the treebank counts over 45,000 sentences (650,000 words) belonging to texts from a variety of genres (contemporary fiction, popular science, newspaper, magazine and journal articles dated between 1960 and 2011, texts of online news, etc.) and is steadily growing.

Since Russian, as other Slavic languages, has a relatively free word order, SynTagRus adopted a dependency-based annotation scheme, in some respects parallel to the Prague Dependency Treebank (Hajič *et al.*, 2001). Syntactic tagging makes use of the full list of the 65 syntactic relations active in the parser (plus one or two specially introduced relations that cannot be handled automatically). All sentences are supplied by a complete tree structure, even if the parser cannot build one. The fictitious link mentioned above is not allowed.

The corpus is built semi-automatically: first, each sentence is processed by the ETAP-3 parser, then it is manually edited by expert linguists, who correct errors made by the parser and handle cases of ambiguity that cannot be reliably resolved without extralinguistic knowledge.

During the manual stage of corpus creation, certain improvements are introduced into the dependency tree annotation that cannot be achieved automatically. In particular, hard cases of ellipsis are made explicit by introducing additional nodes into the annotation. A sentence like (3) *Ja priexal iz Moskvy, a on iz Madrida* ‘I came from Moscow and he from Madrid’ will receive a resulting tree with another instance of the verb *priexal* ‘came’ so that the syntactic links that form the tree have a more natural look. This additional node is marked with a special phantom label.

In this study, SynTagRus is used as gold standard for parsing evaluation. As a matter of fact, the corpus has already been used for a number of linguistic research and development tasks. In particular, it was used as benchmark in regression tests designed to ensure stable performance of the ETAP-3 Russian parser in the course of its development (see e.g. Boguslavsky *et al.* 2008) and as a source for the creation, by machine learning methods, of a successful statistical parser for Russian (Nivre *et al.*, 2008).

5 Evaluation Metrics

We use two types of evaluation: a general evaluation and a penalty-based one. Both will be briefly characterized below.

5.1 General evaluation

Lexico-Grammatical Score (LG)

As discussed in Section 3 above, ETAP-3 does not have a separate POS-tagging stage. Disambiguation of lexico-grammatical features is carried out in parallel with establishing dependency links. However, it is useful to evaluate the POS attribution accuracy separately. It is calculated as follows. For each identified word L its lexico-grammatical coefficient $KL = n_1/n_2$ is determined, where n_1 is the number of correctly identified features of L , and n_2 is the number of all its features. Lexico-grammatical score is defined as the sum of all lexico-grammatical coefficients divided by the number of words.

Word-oriented syntactic scores

- **Head Score:** proportion of words for which the head (or the absence of a head) has been assigned correctly (= Unlabelled Attachment Score; Nivre and Scholz 2004, Eisner 1996).
- **Link Score:** proportion of words for which a name of subordinating link (or the absence of the head) has been assigned correctly. This link may depart from a wrong head.
- **Head and Link Score:** proportion of words for which both the head and the label for subordinating link have been identified correctly (= Labelled Attachment Score; Lin 1998, Nivre and Scholz 2004).
- **Link Audit:** for each link type, its precision, recall and F-score are calculated.

Sentence-oriented syntactic scores

These scores can be computed either for the whole corpus, or for sentences of certain length, e.g. for sentences with less than 10 words, with 10-20 words, with 20-30 words, etc.

- **Root Score:** proportion of sentences for which the root has been identified correctly.
- **Unlabeled Structure Correctness Score:** proportion of sentences for which all the links have been identified correctly – with no regard to link labels (= Complete Rate of Yamada and Matsumoto 2003).
- **Strict Structure Correctness Score:** proportion of sentences for which all

links and their labels have been identified correctly.

- **Gold Standard Achievability:** proportion of sentences for which the gold standard (GS) structure is achieved within the first N alternatives in the stack.

GS achievability is an important feature of the parser. As mentioned in 4.1, the ETAP-3 parser can produce all alternative parses compatible with the grammar. The order in which these alternatives are presented depends on the rank which the parser assigns to them. Sometimes the parser is able to obtain GS but this parse is not on the top of the stack of alternatives. It is useful to know how many GS parses the parser can produce, even if not as the first alternative. This score shows what proportion of incorrect parses is due to grammar flaws as opposed to defects that could be eliminated by means of a better ordering of alternatives. The GS achievability score provides information on the proportion of sentences which achieved GS within the first N alternatives and some other types of supplementary information.

5.2 Penalty-based evaluation

This type of evaluation is based on a detailed list of possible types of deviation of a parse (P) from the gold standard (GS). These types are as follows.

Tokenization deviations

- GS contains a phantom node which has no match in P (see Section 5 above).
- A string of characters in the sentence is differently segmented into tokens in P and GS. This happens when a multiword expression is treated as one word by the corpus annotator but not by the parser dictionary. Here two cases can be distinguished: (a) the difference is recoverable, i.e. one can automatically match nodes in P and GS, and (b) it is unrecoverable. Example of case (a): *antiterrorizm* is represented by one node in GS, but corresponds to two nodes in P (*anti* and *terrorizm*). The parser did not find *antiterrorizm* in the dictionary but decomposed it into two parts and connected them with a composite link. In this case, *antiterrorizm* in GS matches with *terrorizm* in P for further comparison. It is easy since both items have the same list of features. Example

of case (b): the annotator decides that a multiword expression should be represented in the corpus as one indivisible word, which is not present in the dictionary. In this case, the GS may contain e.g. an adverb like *po krajnej mere* ('at least') which is hardly possible to match with the sequence of preposition *po*, adjective *krajnej* and noun *mere* present in P.

Lexico-grammatical deviations between nodes in P and GS with identical tokens

- The word is not recognized in P. It is absent from the dictionary and cannot be decomposed derivationally.
- Nodes in P and GS have different parts of speech, e.g. *čto* can be a pronoun 'what' or a conjunction 'that'.
- Nodes in P and GS have different features within the same part of speech (for examples, see Section 3).
- Nodes in P and GS have different lemmas within the same part of speech, e.g. *naxodit'sja* can be interpreted either as the verb meaning 'be located (somewhere)' or as the passive of the verb *naxodit* 'find'.

Syntactic deviations between nodes in P and GS with identical tokens

All syntactic deviations are mutually exclusive.

- The node in P is connected to another node by a **fictitious** link.
- The node is the root in P but not in GS, or vice versa.
- The node in GS is connected by a link which is absent in the list of links supported by the parser. This may happen, since SynTagRus contains some specific constructions annotated manually.
- In GS the node is linked to node Z with relation R, and in P it is also linked to Z, but with a relation different from R.
- In GS the node is linked to node Z with relation R, and in P it is also linked with R, but to a node different from Z.
- In GS the node is linked to node Z with relation R, and in P it is neither linked to node Z, nor with relation R.

Each deviation type is assigned a penalty. Accordingly, we can calculate penalties of nodes, parses of sentences and parses of corpora. Two types of evaluation can be used.

Non-normalized evaluation is very simple and convenient for comparing results obtained on the same corpus at different times. It consists in summing up all penalties assigned in parsing the corpus. **Normalized evaluation** permits to compare the results obtained on different corpora. It is calculated as follows. For each node, its penalties are summed up and divided by the maximum penalty a node can get. For a sentence, node evaluations are summed up and divided by the number of nodes composing the sentence. For a corpus, sentence evaluations are summed up and divided by a number of sentences in the corpus.

Besides generating the general penalty for a node, sentence or corpus, one can identify a number of specific errors, which helps parser developers to assess the processing accuracy for certain syntactic phenomena. Among them, failures can be detected in:

- finding actants of finite verbs, non-finite verbs, nouns, adjectives and adverbs,
- finding the subject of a verbless sentence,
- finding non-actant dependents of verbs,
- establishing various types of auxiliary links,
- identifying coordination chains.

The syntactic model underlying the parser includes several weakly contrasting dependency types, e.g. different types of attributes and modifiers. One could think of merging them into one hyper-dependency type so as to increase the accuracy of the model. The evaluation software provides a convenient tool to assess the effect of such a merge without the need to previously introduce complex changes to the rules. Specifically, the program can be instructed to disregard certain types of syntactic deviations. For example, one can evaluate the parse of the corpus under the condition that relations R1 and R2 are identical.

6 ETAP-3 Parser Evaluation

Below, some general evaluation data obtained on a fragment of the SynTagRus corpus are presented. This fragment is selected relatively randomly: it represents complete data introduced in the corpus in 2007. The fragment contains 66401 words in 4676 sentences. We will give the results of two types of evaluation: **strict evaluation**, which involves

straightforward calculation of the parameters listed in Section 6, and **relaxed evaluation**, which ignores certain deviations between the gold standard and the evaluated performance of the parser.

The evaluation is largely based on the data from the same syntactically tagged Russian corpus which is used for parser refinement. Methodologically, this is in our opinion quite acceptable, since the version that we evaluated does not include any machine learning.

6.1 Strict evaluation

As was mentioned in section 4, the parser has several modes of operation, including (a) the default mode using the empirical module (EM), (b) the mode that incorporates a data-driven statistical component (DD mode) and (c) the mode that uses neither of the two. The DD component was trained on a different fragment of SynTagRus than that used for evaluation: it includes the data introduced in 2009 (7379 sentences with 103694 words).

Our experiments show that mode (a) where the EM component is used yields the best quality: we will treat it as the default mode. The DD mode yields a slightly worse parsing quality but operates substantially faster. The results obtained in the default mode are given below.

Lexico-Gram. Score	0.977
Head Score (UAS)	0.900
Link Score	0.887
Head&Link Score (LAS)	0.860
Unlabeled Struct. Correctness	0.492
Strict Struct. Correctness	0.352
GS Achievability (stack of 5)	0.511

Table 1. Strict evaluation for the default mode

The Gold Standard Achievability within the first 5 trees in the stack reaches 0.512. This figure is worth comparing with the Strict Structure Correctness score. While the first tree in the stack coincides with the Gold Standard in 35.2% of cases, the Gold Standard tree is found among the first 5 trees in the stack in 51.1% of cases.

The table below presents the link audit calculated on the first alternative basis.

LINK NAME	RECALL	PREC.	F-SC.
1- complement	0.895	0.900	0.897
2- complement	0.815	0.747	0.780
3- complement	0.738	0.629	0.679
4- complement	0.667	0.267	0.380
Appositive	0.855	0.820	0.838
Attributive	0.713	0.631	0.670
Parenthetical	0.832	0.903	0.866
Durative	0.638	0.620	0.673
Infinitive-conjunctive	0.955	0.984	0.969
Quasiagentive	0.927	0.877	0.901
Quantitative	0.938	0.956	0.947
Nonactant-completive	0.741	0.642	0.688
Circumstantial	0.732	0.881	0.800
Restrictive	0.936	0.872	0.903
Modificative	0.966	0.984	0.975
Passive-analytical	0.986	0.973	0.979
Subordinative-conjunctive	0.863	0.867	0.865
Predicative	0.906	0.941	0.923
Prepositional	0.985	0.990	0.988
Copulative	0.858	0.895	0.876
Proleptic	0.475	0.848	0.609
Explicative	0.744	0.668	0.703
Relative	0.830	0.899	0.863
Sentential-coordinative	0.724	0.601	0.657
Coordinative-conjunctive	0.877	0.909	0.893
Coordinative	0.864	0.875	0.869
Comparative-conjunctive	0.809	0.793	0.800
Comparative	0.859	0.751	0.801
Expletive	0.841	0.860	0.850
Elective	0.868	0.951	0.908

Table 4: Link audit

6.2 Relaxed evaluation

In this type of evaluation, the comparison criteria for the parser and the gold standard were weakened as follows.

1) Certain poorly distinguishable syntactic links that the annotators failed to treat in a consistent way throughout the corpus were considered as one link. This was e.g. the case with three types of links that represent different kinds of apposition (the appositive, the nominative-appositive, and the numerative-appositive links: prototypical examples are, respectively, Russian equivalents of phrases like *President Medvedev*, *Novel 'Gone with the wind'* and *Group Three*. Other link clusters included (a) the parenthetical and the restrictive relation for cases like *In particular, they refused to obey* vs. *They refused to obey, in particular John*, and (b) agentive and 2nd

completive relations for cases such as *On byl ubit otravlennoj streloj* ‘He was killed by/with a poisoned arrow’: in one interpretation, the arrow is the agent whilst in the other it is the tool.

2) Certain differences between the parses were ignored if the correct choice required deep semantic knowledge. Primarily, this was the case with different PP attachment in sentences like *He saw a girl with a telescope*.

The following data are the results of relaxed evaluation for the default parsing mode.

	Relaxed eval. default	Wrt Strict Eval.
Lexico-Gram. Score	0.978	+0.001
Head Score (UAS)	0.918	+0.018
Link Score	0.904	+0.017
Head&Link Score (LAS)	0.885	+0.025
Unlabeled Str. Correct.	0.582	+0.090
Strict Struct. Correctness	0.439	+0.087
GS Achievability (stack of 5)	0.560	+0.049

Table 5: Relaxed evaluation for the default mode

The most notable distinction from the strict evaluation is the increase of the Head & Link Score by 2.5%, as well as the increase of the Unlabeled structure and the Strict Structure Correctness by 9.0% and 8.7%, respectively. GS Achievability also grew by 4.9%.

6.3 Comparison with related work

To the best of our knowledge, there are no data on Russian parsers with which we could compare our results. The only exception is the data-driven MaltParser by J. Nivre trained on the SynTagRus corpus (Nivre *et al.* 2008). That is, both parsers strive to come to exactly the same structures, which provides favorable conditions for comparison. However, direct collation of the ETAP-3 parser performance with the results obtained by J. Nivre would hardly be correct, since the input of these parsers is significantly different. The ETAP-3 parser processes the raw, unprepared text. The MaltParser begins with the POS-tagger output. Since no such tagger for Russian was available for the experiments, the input was taken directly from the GS. This means that all tokenization and lexico-grammatical deviations between the sentence and GS (cf. 6.2 above) have been rid of in advance. It is

difficult to accurately assess the impact of these deviations on the ETAP-3 performance. This being said, one can compare two scores available for both parsers. They are largely similar: Head Score – 0.900 (ETAP-3, strict evaluation mode) vs. 0.891 (MaltParser), Head and Link Score – 0.860 (ETAP-3, strict evaluation mode) vs. 0.823 (MaltParser).

As for the related work on dependency parsers for other languages, we can compare the Unlabeled Structure Correctness of ETAP with the English data in Collins 1997, Charniak 2000, Yamata and Matsumoto 2003 and in Nivre and Scholz 2004:

Charniak	0.452
Collins	0.433
Yamada & Matsumoto	0.384
Nivre & Scholz	0.304
ETAP-3 (strict evaluation)	0.492

Table 6: Unlabeled Structure Correctness Score

Additionally, our Head & Link score (both for strict and relaxed evaluation) proves visibly higher than the average figure for this parameter (0.8253) given for several languages in Nivre and McDonald 2008.

7 Error Analysis

As seen from Table 4, of 30 dependency relations represented in the corpus, there are 8 whose F-score exceeds 0.9, and 8 stay below 0.7. We will illustrate both groups of relations with short examples (the head of the construction will be denoted in the gloss as X and the subordinate as Y).

High accuracy relations: infinitival-conjunctive (*čtoby vstretit* ‘in-order-to [X] meet [Y]’), restrictive (*ne byl* ‘was not’, lit. ‘not [Y] was [X]’), quantitative (*pjat’ dnej* ‘five [Y=Nom] days [X=Gen]’), modificative (*tri opytnyx rabotnika* ‘three experienced [Y=Pl] workers [X=Sg]’), passive-analytical (*byl isključen* ‘was [X] expelled [Y]’), predicative (*solnce svetit* ‘the sun [X] shines [Y]’), prepositional (*v dlennom spiske* ‘in [X] the long list [Y]’), elective (*samaja interesnaja iz knig* ‘the most interesting [X] of [Y] the books’).

Low-accuracy relations: 3-rd completive (*oprobovat’ preparat na myshax* ‘to test [X] the medication on [Y] mice’); 4-th completive (*arendovat’ na tri goda* ‘rent [X] for [Y] three years’, *perevozit’ počtu samoletom* ‘to

transport [X] mail by [Y] airplane'), attributive (*dom za uglom* 'a house [X] round [Y] the corner'), durative (*on spit po pjat' chasov v sutki* 'he sleeps [X] five hours [Y] a day'), nonactant-completive (*prishel ko mne v kabinet* lit. 'came [X] to [Y] me into my study' proleptic (*sommenija, oni dolžny byt'* 'doubts [X], they [Y] should exist'), explicative (*my kupili vse – xleb, syr, moloko* 'we bought everything [X] – bread [Y], cheese, milk'; sentential-coordinative (*Oni ne pridut, i my ostanemsja odni* 'they will [X] not come, and [Y] we will be alone').

A detailed error analysis cannot be done within a short paper. By way of example, we will only comment on the attributive link which connects a noun with its non-argument modifier if they do not agree in case, number and gender. This is a notoriously difficult link to establish, due to the absence of formal features and the abundance of possible heads. Most of the situations in which an attributive link is established erroneously are the following:

- (a) it is established instead of a circumstantial link leading from a verb,
- (b) it is established instead of an attributive link leading from a more distant noun,
- (c) it is established instead of an appositive link, if the subordinate node is a non-identified (NID) proper noun absent in the dictionary.

The latter case deserves a special comment. Existence of NIDs significantly decreases the recall of the attributive and the precision of appositive links. This may be improved by including a Named Entity Recognizer at the preprocessing stage. Another direction of improvement is connected with augmenting the performance of the guessing rules which should identify the morphological form of the word even if it is absent from the dictionary.

One more notable source of parser failure is inconsistent dictionary coverage. In many cases, ignorance is better than half-truth: it is better to leave a whole family of lexical units outside the dictionary than to introduce it fragmentarily. Consider a typical situation where a Russian name of a town, like *Krasnojarsk*, is present in the dictionary but the corresponding adjective, *krasnojarskij*, is not. Due to a specific intersection of paradigms of such words (they have coinciding word forms in structurally different cases: the instrumental case of the noun coincides with

the locative case of the adjective), sentences like

(3) *On rabotaet na krasnojarskom zavode* 'He works at a Krasnojarsk plant'

will not be parsed sensibly because the adjective will be treated as a stray noun in the instrumental case. It would be counterintuitive to instruct the parser to treat a word form found in the dictionary on a par with a non-identified word. Accordingly, the parse would be more acceptable if the whole family of words remained unlisted: in this case, there will be a local parsing mistake, whereas in the opposite case the parser will simply play havoc with the structure.

8 Conclusion

We have presented ETAP-3 parser, a rule-based system for dependency parsing which makes part of a multifunctional linguistic processor. It was developed for Russian and English, but evaluated only for Russian by means of a SynTagRus dependency treebank. The characteristic feature of the parser is a fine-grained dependency type set which includes 65 types. Some of them are rather rare: in the fragment of the treebank used for evaluation, only 30 types are represented. We use various types of metrics, some of them better suited for intrinsic evaluation (penalty-based), while others (general) are convenient for comparison with other systems.

The main directions of future research are: 1) improvement of rules for low score dependency types, 2) development of rules for treating ellipsis, 3) upgrading the algorithm for producing alternative parses, 4) experiments on developing a hybrid rule-based/data-driven parser.

9 Acknowledgements

This study has been partially funded by the Russian Foundation for Basic Research (grants No. 10-07-90001-Bel_a and 11-06-00405-a, and a 2011 Grant on Corpus Linguistics from the Russian Academy of Sciences. The authors express their appreciation to the Foundation and the Academy.

References

- Jurij Apresian, Igor Boguslavsky, Leonid Iomdin, Alexander Lazursky, Vladimir Sannikov, Victor Sizov, Leonid Tsinman. 2003. ETAP-3 Linguistic Processor: a Full-Fledged NLP

- Implementation of the MTT. // MTT 2003, First International Conference on Meaning – Text Theory. Paris, École Normale Supérieure, Paris, pp. 279-288.
- Juri Apresjan, Igor Boguslavsky, Boris Iomdin, Leonid Iomdin, Andrei Sannikov, Victor Sizov. (2006). A Syntactically and Semantically Tagged Corpus of Russian: State of the Art and Prospects. // Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006). Genoa, p. 1378-1381.
- Igor Boguslavsky, Svetlana Grigorieva, Nikolai Grigoriev, Leonid Kreidlin, Nadezhda Frid. 2000. Dependency Treebank for Russian: Concept, Tools, Types of Information // Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000), p. 987-991)
- Igor M. Boguslavsky, Leonid L. Iomdin et al. (2005). Interactive Resolution of Intrinsic and Translational Ambiguity in a Machine Translation System. // CICLing 2005. Lecture notes in computer science. A. Gelbukh (ed.), Springer-Verlag Berlin Heidelberg 2005, pp. 383 – 394.
- Igor M. Boguslavsky, Leonid L. Iomdin, Svetlana P. Timoshenko, Tatyana I. Frolova (2009). Development of the Russian Tagged Corpus with Lexical and Functional Annotation // Metalanguage and Encoding Scheme Design for Digital Lexicography. MONDILEX Third Open Workshop. Proceedings. Bratislava, Slovakia. April 15-16, 2009. Bratislava, 2009. P. 83-90. ISBN 978-80-7399-745-8.
- Charniak, E. 2000. A maximum-entropy-inspired parser. In Proceedings of NAACL.
- Collins, M. 1997. Three generative, lexicalized models for statistical parsing. In Proceedings of ACL, pp. 16-23, Madrid.
- Eisner, J.M. 1996. Three new probabilistic models for dependency parsing: An exploration. In Proceedings of COLING, Copenhagen, Denmark.
- Hajič et al., 2001: Hajič, J., Vidova Hladka, B., Panevová, J. E. Hajičová, P. Sgall, and P. Pajas. 2001. Prague Dependency Treebank 1.0. LDC, 2001T10.
- Lin, D. 1998. A dependency-based method for evaluating broad coverage parsers. Natural Language Engineering 4, 97-114.
- Mel'čuk I.A. 1974. Opyt teorii lingvisticheskix modelej klassa "Smysl - Tekst". [The theory of linguistic models of the Meaning – Text Type]. Moscow, Nauka. (In Russian).
- Nivre, J. and Scholz, M. 2004. Deterministic dependency parsing of English text. // Proceedings of the 20th International Conference on Computational Linguistics (COLING), pp. 64-70.
- Nivre, J. and McDonald, R. 2008. Integrating Graph-Based and Transition-Based Dependency Parsers. In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT), pp. 950-958.
- Petrochenkov V.V. and Sizov V.G. Ispol'zovanie statističeskoj informacii o konkurirujuščix sintaksičeskix svjazjax v sintaksičeskom analizatore ETAP-3 dlja polučenija naibolee verojatnoj sintaksičeskoj struktury frazy // Informacionnye tehnologii i sistemy (ITiS'10). Sbornik trudov 33 Konferencii molodyx učenyx i specialistov IPPI RAN. Gelendžhik, 18-26 sentjabrja 2010 g. Moskva: IPPI, 2010. P. 299-305.
- Tsinman, L. and Druzhkin K. (2008). Sintaksičeskij analizator lingvističeskogo processora ETAP-3: eksperimenty po ranžirovaniju sintaksičeskix gipotez // Kompjuternaja lingvistika i intellektal'nye tehnologii (Dialog 2008). Trudy meždunarodnoj konferencii. Bekasovo, 4-8 ijunja 2008 g. Moskva, RGGU, 2008. Vyp. 7(14). P. 147-153. ISBN 978-5-7281-1022-4.
- Yamada H., Matsumotu Y. 2003. Statistical dependency analysis with support vector machines. // Proceedings of the 8th International Workshop on Parsing Technologies (IWPT), pp.195-206, Nancy, France.