M A T E M A T I C K O - F Y Z I K Á L N Í F A K U L T A

P R A H A

CROSS-LANGUAGE STUDY ON INFLUENCE OF COORDINATION STYLE ON DEPENDENCY PARSING PERFORMANCE

David Mareček, Martin Popel, Loganathan Ramasamy,
Jan Štěpánek, Daniel Zeman, Zdeněk Žabokrtský, Jan Hajič

ÚFAL Technical Report
TR-2013-49

U N I V E R S I T A S C A R O L I N A P R A G E N S I S

Copies of ÚFAL Technical Reports can be ordered from:

Institute of Formal and Applied Linguistics (ÚFAL MFF UK)
Faculty of Mathematics and Physics, Charles University
Malostranské nám. 25, CZ-11800 Prague 1
Czech Republic

or can be obtained via the Web: http://ufal.mff.cuni.cz/techrep

Cross-language Study on Influence of Coordination Style on Dependency Parsing Performance

David Mareček, Martin Popel, Loganathan Ramasamy, Jan Štěpánek,
Daniel Zeman, Zdeněk Žabokrtský, Jan Hajič

Abstract

In this report we explore alternative representations of coordination structures within dependency trees and study the impact of particular solutions on performance of two selected state-of-the-art dependency parsers across a typologically diverse range of 25 languages.

Contents

Chapter 1

Introduction

Dependency parsing has received continuously growing attention in the last decade. One of the reasons is growing availability of dependency treebanks, be they resulting from genuine dependency annotation projects or converted automatically from already existing phrase-structure treebanks.

In both ways, large number of decisions have to be made during the construction of a dependency treebank. Even if the traditional notion of dependency might look clear at the first sight (an attribute modifies a noun, an object is an argument of a verb, etc.), it does not provide unique clues in many situations, for example when it comes to attaching functional words. Even worse, this notion comes absolutely short when it is to represent paratactic linguistic phenomena such as coordination, whose nature is symmetric (two or more conjuncts play the same role), as opposed to head-modifier asymmetry of dependency relations.

The dominating solution is to introduce artificial rules for encoding coordination structures (CS) within dependency trees, by the same means that serve for expressing dependencies, i.e. by presence of edges and by labeling of nodes or edges. Obviously, any tree-shaped representation of coordination structures must be perceived only as a "shortcut", since relations present in coordination structures form an undirected cycle, as illustrated already in [Tesnière, 1959]. For example, if a noun is modified by two coordinated adjectives, there is a (symmetric) coordination relation between the two conjuncts and two (asymmetric) dependency relations between the conjuncts and the noun.

However, as there is no obvious linguistic intuition on which tree-shaped CS encoding is better and as the degree of freedom has several dimensions (variations both in topology and labeling are possible), one can find a number of distinct conventions introduced in particular dependency treebanks. So the first goal of this report is to give a systematic survey of possible solutions.

Naturally, the intricate interplay of dependency and coordination relations within a single tree structure leads to parsing issues.[1] Unlike dependency relation, coordination structure typically comprises at least three tokens: coordination conjunction and two (or more) conjuncts, which implies that independence assumptions often put on tree edges are inadequate. One can find in the literature several strategies to tackle this problem:

- The fact that there are two different types of relations mixed in the same tree is not reflected at all in the internal parser structure, and it is hoped to be overcome just by using large set of features – this is by far the most frequent ap-

[1] To our experience, coordination structures belong to the most frequent sources of parsing errors, not only in terms of attachment accuracy. Their impact on quality of dependency-based machine translation can be also substantial. As documented on an English-to-Czech dependency-based translation system [Popel and Žabokrtský, 2009], 39 % of serious translation errors which are caused by wrong parsing have to do with coordination.

proach. Feature templates targeted at coordination are designed occasionally, e.g. in [Novák and Žabokrtský, 2007].

- Coordination structures are subject to specialized pre-processing or post-processing, for instance reparsing of coordination structures. For example, intraclausal coordination candidates are detected prior to the main parsing step in [Marinčič et al., 2007].
- The parser has separate models for dependency and for coordination, as in [Zeman, 2004].
- Several possible representations of coordination structures are compared in terms of parsing feasibility and the one which fits best to the chosen parser (in terms of parsing accuracy) is used; unless the best fitting convention is the same which was used for the original treebank, this approach implies that transformations from the desired style to the best fitting style and back (inverse transformations) must be available.

We adhere to the last strategy in this report. The second goal of the report is to find out which tree-shaped representation of coordination structures fits best with two state-of-the-art parsers.

Attempts at comparing formal feasibility of different representations of coordination for dependency parsing go back to [Lombardo and Lesmo, 1998], and a number of empirical studies focused on performance of data-driven dependency followed later. What is novel about our work is a systematic multidimensional exploration of possible coordination styles and typologically very diverse (probably the widest published) set of languages under study. Even if the drawn conclusions are not the ultimate answers, the consistency across the range of languages adds to their importance.[2]

The rest of the report is structured as follows. Section 2 gives a survey of previous approaches to dependency tree transformations. Section 3 summarizes possible "styles" (topological and labeling variations) for representing coordination structures. Section 4 describes our efforts on collecting and homogenizing dependency trees from as many as 25 languages. Section 5 presents our experimental settings, final results and discussion. Section 6 concludes.

[2]Our present study is part of a broader project in which we compare different annotation styles of various other phenomena such as preposition-noun configuration, relative clauses, modal and complex verb forms etc. Preliminary results indicate that coordination structures are the most interesting phenomenon with respect to the impact on parsing.

Chapter 2

Related work

Let us recall the basic well-known characteristics of CSs first.

In the simplest case of CS, a coordination conjunction joins two (usually syntactically and semantically compatible) sentence elements called conjuncts. Even this simplest case is difficult to represent within a dependency tree, because, in words of [Lombardo and Lesmo, 1998]:

> Dependency paradigms exhibit obvious difficulties with coordination because, differently from most linguistic structures, it is not possible to characterize the coordination construct with a general schema involving a head and some modifiers of it.

Proper formal representation of CSs is further complexified by the following facts:

- CSs with more than two conjuncts are possible (and frequent).
- Besides private modifiers of individual conjuncts, there can be shared modifiers belonging to all conjuncts, such as in *"Mary came and cried"*. Shared modifiers can appear alongside with private modifiers of particular conjuncts.
- Shared modifiers can be coordinated too: *"big and cheap apples and oranges"*.
- Embedded (nested) coordinations are possible, such as in *"John and Mary or Peter and Lisa"*. For estimate frequencies of nested CSs across the 25 languages, see the last column of Table 4.1.
- Punctuation (commas, semicolons, three dots) is frequently used in CSs, mostly with multi-conjunct coordination.
- In many languages, comma or other punctuation mark can play the role of the main coordinating conjunction.
- Coordinating conjunction itself can be a multiword expression (*"as well as"*).
- Deficient CSs with a single conjunct exist.
- Abbreviations like *"etc."*, *"atd."* (Czech) and *"usw."* (German) comprise both conjunction and last conjunct.
- Coordination combined with ellipsis forms an intricate structure. For example, a conjunct can be elided while its arguments remain in the sentence, such as in the following traditional example: *"I gave the books to Mary and the records to Sue."*
- The border between paratactic and hypotactic surface means for expressing coordination relations is fuzzy. Some languages can use enclitics instead of conjunctions/prepositions, e.g. Latin *"Senatus Populus*que *Romanus"*. Purely hypotactic surface means such as the preposition in *"John with Mary"* occur too.

- Careful semantic analysis of CSs discloses additional complications: if a node has a CS as its child, it might happen that it is the node itself (and not its modifiers) what should be semantically considered as a conjunct. Note the difference between *"red and white wine"* (which is synonymous to *"red wine and white wine"*) and *"red and white flag of Poland"*. Similarly, *"five dogs and cats"* has different meaning than *"five dogs and five cats"*.

Some of these issues were recognized already in [Tesnière, 1959]. In his solution, conjuncts are connected by vertical edges directly to the head, as well as by horizontal edges to the conjunction (which leads to a cycle in every CS).

Many different models have been proposed since, out of which the following are probably the most frequently used ones:

- Mel'čuk style (MS) used in the Meaning-Text Theory (MTT), in which the first conjunct is the root of the CS, with the second conjunct attached below the first one, third conjunct below the second one, etc. Coordinating conjunction is attached below the penultimate conjunct, and the last conjunct is attached below the conjunction [Mel'čuk, 1988],

- Prague Dependency Treebank style (PS), in which all conjuncts are attached below coordination conjunction (as well as shared modifiers, which are distinguished by a special attribute) [Hajič et al., 2006],

- Stanford style (SS),[1] in which the first conjunct is head and the remaining conjuncts as well as the conjunctions are attached below it.

One can find various arguments supporting the particular choices. MTT possesses a complex set of linguistic criteria for identifying governor of a relation (see also [Mazziotta, 2011] for an overview), leading to MS. MS is preferred in a rule-based dependency parsing system of [Lombardo and Lesmo, 1998]. PS is advocated in [Štěpánek, 2006] by the claim that it can represent shared modifiers using a single additional binary attribute, while MS would require a more complex coindexing attribute for that. An argumentation of [Tratz and Hovy, 2011] follows a similar direction: *We would like to change our [MS] handling of coordinating conjunctions to treat the coordinating conjunction as the head [PS] because this has fewer ambiguities than [MS]...*

In the era of statistical data-driven approaches, the question of choosing an optimal representation for a phenomenon which does not provide enough intuition is often governed by pragmatic concerns, which helps to escape from potentially controversial formal linguistic arguments. In the case of coordination, maximizing parsers' performance seems to be a reasonable pragmatic criterion.[2] Such experiments have typically the following scenario summarized in [Bengoetxea and Gojenola, 2009]:

1. apply the transformation to the training data,

2. train a parser on the transformed data,

3. parse the test set, and

4. apply the inverse transformation to the parse output, so that the final evaluation is carried over the original tree representations.

[1] We use the already established MS-PS-SS distinction to facilitate literature overview; as shown in Section 3, the space of possible coordination styles is much richer and can be structured along several dimensions.

[2] This is certainly not specific for dependency parsing, problems related to various possible representations are often addressed also in the world of constituency parsing.

One can find a number of such experiments aimed at comparing parser performance for different coordination styles in the literature, for example:

- [Tsarfaty et al., 2011] compare performance of two parsers on three different coordination styles applied on English; their conclusion is that if the resulting parses are converted into a common more abstract representation (called functional trees, resembling constituency trees), then the dramatic gaps observed when comparing parsing results obtained in isolation decrease or dissolve completely;
- three different dependency parsers developed and tested with respect to two treebanks for the Italian language are compared in [Bosco et al., 2010];[3]
- [Bengoetxea and Gojenola, 2009] shows that PS performs worse than MS, which performs worse than SS for Basque;
- the conjecture that MS outperforms PS is confirmed also in [Nilsson et al., 2006], this time on the PDT data,
- PS performs as the worst also in [McDonald and Nivre, 2007], in which 11 treebanks are used.

Besides maximizing parsers' performance, transformations between different coordination styles is often needed also when parsers trained on different data are to be compared (cross-experimental evaluation), or when dependency trees are projected from one language to another.

We find it natural to consider resolution of coordination structures as a subtask of parsing. However, it is not the only option. For instance, [Ogren, 2010] developed a system for resolving coordination structures using language models, independently of any parser.

We conclude that the influence of the choice of coordination style is a well known problem in dependency parsing. Nevertheless, all published works focus only on a very narrow set of traditional coordination styles. Moreover, the experiments are conducted using a single language or just a few languages in most cases.

[3]This is the very rare case in which one can find a pair of treebanks for the same language originally annotated with different coordination styles and does not have to transform the data first. However, it does not make the situation simpler, as the treebanks are likely to differ also in many other aspects.

Chapter 3

Variations in representing coordination structures

3.1 Assumptions

We assume that each sentence is represented by one dependecy tree, in which each node corresponds to one token (word or punctuation mark). Apart from these usual conventions, we deliberately limit ourselves to CS representations that have shapes of connected subgraphs of dependency trees. Moreover, we disregard CS styles which systematically generate non-projective edges.

We limit our repertory of means for expressing CSs within dependency trees to:

- tree topology (presence or absence of a directed edge between two nodes),
- node labeling (additional attributes attached to nodes),[1]

Further, we expect that the set of possible variations can be structured along several dimensions, each of which corresponds to a certain simple characteristic (such as picking the CS root on the right-hand side, or attaching shared modifiers below the nearest conjunct). Even if it does not make sense to create full Cartesian product of all dimensions because some values cannot be combined, it allows to explore the space of possible CS styles in a relatively systematic fashion and to study the influence of individual factors in isolation.

One can find CS representations in the literature that do not fit into these limitations, such as CS representation using additional secondary (tree-crossing) edges in the Tiger Treebank [Brants et al., 2002], or bubble trees suggested for Mel'čuk style in [Kahane, 1997] (bubbles are objects representing embeddable clusters of nodes). We exclude such means from our experiments because these constructs are not supported by the contemporary state-of-the-art parsers and would require deep redesign of the underlying parsing algorithms.

3.2 Topological variations

For each particular CS, it would be easy to generate an exhaustive set of possible trees spanning over its participants. However, it would be extremely difficult to pick variants belonging to the same coordination style across the whole data. Therefore we prefer to generate topological variations by hand-crafted transformations along several pre-defined dimensions, even if it does not guarantee that all possible variations are explored.

[1] Edge labeling can be trivially converted to node labeling in tree structures.

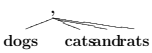
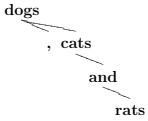
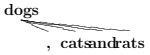
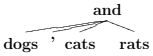
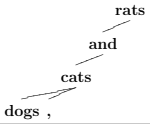
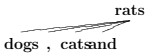
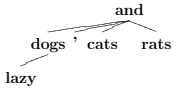
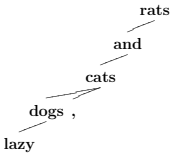
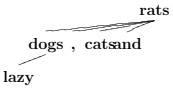
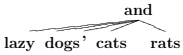
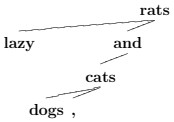
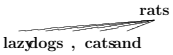
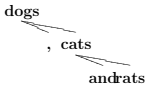
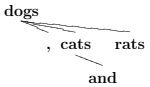
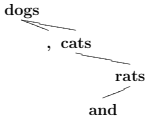
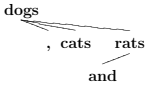
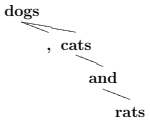
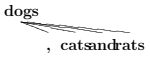
Main family	Prague family (code fP) [13 treebanks]	Moscow family (code fM) [5 treebanks]	Stanford family (code fS) [6 treebanks]
Choice of head			
Head on left (code hL) [11 treebanks]			
Head on right (code hR) [13 treebanks]			
Mixed head (code hM)	This style is a mixture of hL and hR – for each CS, we choose the head which is closer to the parent of the whole CS. We are not aware of any treebank using this style.		
Attachment of shared modifiers			
Shared modifier below the nearest conjunct (code sN)			
Shared modifier below head (code sH) [7 treebanks]			
Attachment of coordination conjunction			
Coord. conjunction below previous conjunct (code cP) [2 treebanks]	—		
Coord. conjunction below following conjunct (code cF) [1 treebank]	—		
Coord. conjunction between two conjuncts (code cB) [8 treebanks]	—		
Coord. conjunction below as the head (code cH) is the only applicable style for Prague family [13 treebanks]		—	—
Placement of punctuation			
values pP [7 treebanks], pF [1 treebank] and pB [14 treebanks] are analogous to cP, cF and cB (but applicable also to Prague family)			

Table 3.1: Different coordination styles, variations in tree topology. Example phrase: *"lazy dogs, cats and rats"*. Style codes are described in Section 3.2.

We distinguish the following dimensions of topological variations of CSs (see Figure 3.1):

Family – configuration of conjuncts We divide the topological variations into three main groups, labeled as Prague (fP), Moscow (fM), and Stanford (fS) families (names of the cities are chosen purely as a mnemonic device, so that Prague Dependency Treebank belongs to the Prague family, Mel'čukian style belongs to the Moscow family, and Stanford parser style belongs to the Stanford family). This first dimension distinguishes the configuration of conjuncts: in Prague family all the conjuncts are siblings governed by one of the conjunctions (or punctuation); in Moscow family the conjuncts form a chain where each node in the chain depends on the previous (resp. following) node; in Stanford family the conjuncts are siblings except for the first (resp. last) conjunct which is the head.[2]

Choice of head – leftmost or rightmost In Prague family, the head can be either the leftmost[3] conjunction or punctuation (hL) or the rightmost (hR). Similarly, in Moscow and Stanford families the head can be either the leftmost (hL) or the rightmost (hR) conjunct. We introduce a third option called *mixed* (hM), where for each CS, we choose the head which is closer to the parent of the whole CS. So in hM, some CSs look like hL and some like hR. The motivation behind this option is to make the edge between CS head and its parent shorter, which may improve the parser training.

Attachment of shared modifiers Shared modifiers can appear before the first conjunct or after the last conjunct. Therefore, it seems reasonable to attach shared modifiers either to the CS head (sH), or to the nearest (i.e. first or last) conjunct sN.

Attachment of coordinating conjunctions In Moscow family, conjunctions can be either part of the chain of conjuncts (cB), or they may be put aside the chain and attached to the previous (cP) or following (cF) conjunct. In Stanford family, conjunctions can be either attached to the CS head (and therefore *between* conjuncts) (cB), or they may be attached to the previous (cP) or following (cF) conjunct. The cB option, in both Moscow and Stanford, treats conjunctions in the same way as conjuncts (as for the topology only, of course). In Prague family, there is just one option available (cH) – one of the conjunctions is the CS head, while the others are attached to it.

Attachment of punctuation Punctuation separating conjuncts (commas, semicolons etc.) in CSs could be treated in the same way as conjunctions. However, in most treebanks it is treated differently and we follow the practice by allowing to choose different option for conjunctions and for punctuation. Values pP, pF and pB are analogous to cP, cF and cB except that punctuation can be attached also to conjunction in case of pP and pF (otherwise, a comma before conjunction would be non-projectively attached to the member following the conjunction).

The three established styles mentioned in Section 2 can be defined in terms of the newly introduced abbreviations: PS = fPhRsHcHpB, MS = fMhLsNcBp?, and SS = fShLsNcBp? (the question marks indicate that the original Mel'čuk and Stanford styles ignore punctuation).

[2]Note that for CSs with just two conjuncts (which is the most common case), fM and fS may look exactly the same (depending on the attachment of conjunctions and punctuation as described below).

[3]For simplicity, we use the terms left and right even if their meaning is reversed for languages with right-to-left writing system such as Arabic.

3.3 Labeling variations

Most state-of-the-art dependency parsers can produce labeled edges. However, the parsers produce only one label per edge. To fully capture CSs, we need more than one label, because there are several aspects involved (see 3.1): We need to identify the coordinating conjunction (morphological information might not be enough), conjuncts, shared modifiers, and punctuation separating conjuncts. Besides that, there should be a label classifying the dependency relation between the CS and its parent.

Some of the information can be retrieved from the topology and the "main label", but not everything. The additional information can be, of course, concatenated into just one label, but such an approach leads to sparser data and thus makes the parser results worse.

Different types of labeling are equivalent (to some extent) and their switching might be regarded as a type of a transformation (see Section 3.5).

In **Prague family**, there are two possible ways to *label a conjunction and conjuncts:*

- Code `dU` ("**d**ependency labeled at the **u**pper level of the CS"). The dependency relation of the whole CS to its parent is represented by the label of the conjunction, while the conjuncts are labeled with a special label for conjuncts. This style was used e.g. in the Hyderabad Dependency Treebank (conjuncts are marked with the label `ccof`).

- Code `dL` ("**l**ower level"). The CS is represented by a coordinating conjunction (or punctuation if there is no conjunction) with a special label. Subsequently, each conjunct has its own label that reflects the dependency relation towards the parent of the whole CS (therefore, conjuncts of the same CS can have different labels). This style was used e.g. in PDT (the label for coordinating conjunctions is `Coord`).

To represent *shared modifiers* in Prague family, there are again several possibilities. Each child of a coordinating conjunction has to belong to one of the three sets: conjuncts, shared modifiers, and punctuation or additional conjunctions. In PDT, only conjuncts are labeled (by the `is_member = 1` attribute), whereas the other two sets can be distinguished according to the labels (`AuxX`, `AuxY`, and `AuxG` can never be shared modifiers). It is not possible, though, to tell conjuncts and shared modifiers apart according to their labels (`Sb` is used for Subject both in *"Peter sleeps."* and *"Peter sleeps and snores."* Therefore, members of one of the two sets must be labeled.

In **Stanford and Moscow families**, one of the conjuncts is taken as the representative. In practice, it is never labeled as a conjunct because its conjunctness can be deduced from the fact there are conjuncts among its children. The other conjuncts are labeled as conjuncts and coordinating conjunctions and punctuation also have a special label. This type of labeling will be marked `dX`. Alternatively (found in Turkish treebank), all conjuncts in the Moscow chain bear the dependency label and their conjunctness follows from the `COORDINATION` labels of the conjunction and punctuation nodes between them (marked `dA`).

To represent shared modifiers in the latter styles, some additional label is needed again to distinguish between private and shared modifiers, since they cannot be distinguished topologically. Moreover, if embedded CSs are used, the label cannot be just binary (i.e. "shared" versus "private"), because it also has to indicate what conjuncts the shared modifier belongs to. (This is not needed in Prague family where shared modifiers are attached to the conjunction, provided each shared modifier is shared by conjuncts that form a full subtree together with their coordinating conjunctions; no exceptions to this assumption were found during the annotation process of the PDT.) See also Section 3.4.

Codes: binary flags: `m1` = conjuncts labeled; `m2` = shared modifiers labeled (therefore, `m3` would mean "both labeled").

3.4 Expressive power

Particular styles (Prague, Moscow and Stanford) do not capture the same information, or, in other words, the sets of CSs they can render are not isomorphic.

It is not possible to represent embedded CSs (see Section 2) in Moscow and Stanford styles without significantly changing the number of possible labels (Mel'čuk uses "grouping" to nest CSs, but this approach was not used in any of the researched treebanks. To combine grouping with shared modifiers, each group in a tree should have a different number or identifier).

The Prague family can represent coordination of different relations. This is again not possible in the other styles without adding a special "prefix" denoting the relations.

We can see that the Prague family has greater expressive power than the other two: it can represents complicated CSs with just one additional binary label. Shared modifiers and conjuncts can be distinguished only using such a label; similar additional label is needed in the other styles to distinguish between shared and private modifiers.

The possible impact of each style is discussed in Sections 5.4 and 6.

3.5 Style convertibility

Because of the different expressive power (see Section 3.4), converting a CS from one style to another can lose information. For example, there is no way how to represent shared modifiers in the Moscow style without additional labels, therefore converting a Prague style CS with shared modifiers makes them private. When converting back, there can be some heuristics to handle the most obvious cases, but sometimes the modifiers will stay private (very often, the nature of a modifier depends on context or is debatable even for humans, e.g. *"Young boys and girls"*).

3.6 Transformation algorithm

The algorithm we used to transform one CS style to another consists of two subtasks: detecting CSs (including classification of CS participants), and the very transformation procedure, which transforms one CS at a time.

We change the trees in-place by a depth-first traversal. Each node is classified either as a CS participant or as a node not participating in a CS. CS participants are further classified as: coordinating conjunction, conjunct, shared modifier, or punctuation separating conjuncts. If a node is classified as CS participant, but its parent is not, we can be sure that we have reached the topmost node of a CS (so we have already gathered all the participants of the CS) and we apply the transformation procedure on the participants. One of the most difficult steps is to handle correctly embedded coordinations.

The transformation procedure is quite straightforward – once we have detected all the CS participants, we reattach them according to the desired output coordination style. The transformation procedure must return the new CS head, because it may be a conjunct of an outer CS in case of embedded CSs.

3.7 Need for empirical evaluation

In this report we compare feasibility of individual CS styles on a purely empirical basis. We believe that it would be difficult (if not impossible) for a human to hypothesize about parser-optimal CS styles and to correctly identify the fundamental causes of superiority of one CS style above the others, even with perfect knowledge of parser internals. The reason

is that the eventual parser performance is influenced (among others) by several pairs of mechanisms pushing its learning algorithm in opposite directions. We give two examples:

- Keeping all conjuncts in a chain without interrupting it by a conjunction (e.g. fM cP) if beneficial for features that model coordinability – at least we would expect it from the linguistic point of view, since the presence of coordination is hard to predict if the second conjunct is not accessible. On the other hand, this style leads to longer edges (compared to the style with interleaved conjunctions), which makes the observations generally sparser.

- On one hand, PDT style leads to less scattered distributions of node fertility for word classes other than conjunctions, and it also requires less complex labeling if shared modifiers need to be properly resolved in embedded CSs (compared both to Mel'čuk and Stanford styles). But on the other hand, the PDT style implies that conjuncts do not "see" their dependency governors directly, which reduces the discriminative potential of first-order edge models.

Only the experiments can show to what degree which of these intuitions prevails with real parsers applied on real data.

Chapter 4

Data preparation

4.1 Data resources

Our goal was to compare as many different languages and annotation styles as possible. Without any claim of completeness, we were able to identify approx. 30 languages for which treebanks exist and are available for research.[1] Treebanks released during dependency parsing shared task campaignes proved to be the most fruitful data source. We used:

- 6 languages from CoNLL-2006 [Buchholz and Marsi, 2006],
- 6 languages from CoNLL-2007 [Nivre et al., 2007a],
- 3 languages from CoNLL-2009 [Hajič et al., 2009],
- 3 languages from ICON-2010 [Husain et al., 2010].

We added a few others freely available on the Web. Whenever possible, we used the CoNLL format of the data. Dealing with fewer input formats and using similar data as in related work are the obvious advantages; on the other hand we risk that the original formats of the treebanks contained additional information, lost in the CoNLL conversion process.

Many treebanks are natively dependency-based but some were originally based on constituents and their conversion to CoNLL included a head-selection procedure. For instace, the Spanish phrase-structure trees were converted to dependencies using a procedure described in [Civit et al., 2006].

For some languages (Estonian, Hebrew, Icelandic) we found constituent treebanks only. We originally experimented with our own simple head-selection procedure for Estonian. Unfortunately we were not able to come up with reasonable results; the treebank is also very small and it contains both text and speech data, so we decided to exclude it from our current experimentation. We have not attempted to process Hebrew and Icelandic.

We work with the following treebanks (note the ISO 639 codes after the language names—we use these codes to refer to the languages elsewhere in the article):

- Arabic (ar): Prague Arabic Dependency Treebank 1.0 / CoNLL 2007 [Smrž et al., 2008][2]
- Basque (eu): Basque Dependency Treebank (larger version than CoNLL 2007 generously provided by IXA Group) [Aduriz et al., 2003]

[1] Most of the datasets can either be acquired free of charge or they are included in the Linguistic Data Consortium membership fee.

[2] `http://padt-online.blogspot.com/2007/01/conll-shared-task-2007.html`

- Bulgarian (bg): BulTreeBank [Simov and Osenova, 2005][3]
- Czech (cs): Prague Dependency Treebank 2.0 / CoNLL 2009 [Hajič et al., 2006][4]
- Danish (da): Danish Dependency Treebank / CoNLL 2006 [Kromann et al., 2004], now part of the Copenhagen Dependency Treebank[5]
- Dutch (nl): Alpino Treebank / CoNLL 2006 [van der Beek et al., 2002][6]
- English (en): Penn TreeBank 2 / CoNLL 2009 [Surdeanu et al., 2008][7]
- Finnish (fi): Turku Dependency Treebank [Haverinen et al., 2010][8]
- German (de): Tiger Treebank / CoNLL 2009 [Brants et al., 2002][9]
- Greek (modern) (el): Greek Dependency Treebank [Prokopidis et al., 2005]
- Greek (ancient) (grc): Ancient Greek Dependency Treebank [Bamman and Crane, 2011][10]
- Hindi (hi), Bengali (bn) and Telugu (te): Hyderabad Dependency Treebank / ICON 2010 [Husain et al., 2010]
- Hungarian (hu): Szeged Treebank [Csendes et al., 2005][11]
- Italian (it): Italian Syntactic-Semantic Treebank / CoNLL 2007 [Montemagni et al., 2003][12]
- Latin (la): Latin Dependency Treebank [Bamman and Crane, 2011][13]
- Portuguese (pt): Floresta sintá(c)tica [Afonso et al., 2002][14]
- Romanian (ro): Romanian Dependency Treebank [Călăcean, 2008][15]
- Russian (ru): Syntagrus [Boguslavsky et al., 2000]
- Slovene (sl): Slovene Dependency Treebank / CoNLL 2006 [Džeroski et al., 2006][16]
- Spanish (es): AnCora [Taulé et al., 2008]
- Swedish (sv): Talbanken05 [Nilsson et al., 2005][17]
- Tamil (ta): TamilTB [Ramasamy and Žabokrtský, 2011][18]
- Turkish (tr): METU-Sabanci Turkish Treebank [Atalay et al., 2003][19]

[3] http://www.bultreebank.org/indexBTB.html
[4] http://ufal.mff.cuni.cz/pdt2.0/
[5] http://code.google.com/p/copenhagen-dependency-treebank/
[6] http://odur.let.rug.nl/~vannoord/trees/
[7] http://www.cis.upenn.edu/~treebank/
[8] http://bionlp.utu.fi/fintreebank.html
[9] http://www.ims.uni-stuttgart.de/projekte/TIGER/TIGERCorpus/
[10] http://nlp.perseus.tufts.edu/syntax/treebank/greek.html
[11] http://www.inf.u-szeged.hu/projectdirs/hlt/index_en.html
[12] http://medialab.di.unipi.it/isst/
[13] http://nlp.perseus.tufts.edu/syntax/treebank/latin.html
[14] http://www.linguateca.pt/floresta/info_floresta_English.html
[15] http://www.phobos.ro/roric/texts/xml/
[16] http://nl.ijs.si/sdt/
[17] http://www.msi.vxu.se/users/nivre/research/Talbanken05.html
[18] http://ufal.mff.cuni.cz/~ramasamy/tamiltb/0.1/
[19] http://www.ii.metu.edu.tr/content/treebank

4.2 Train/test division

For CoNLL treebanks we used the CoNLL-defined train/test data split. Whenever we had to split the treebank ourselves we tried to keep the test size similar to the majority of CoNLL 2006/2007 test datasets, i.e. roughly 5000 tokens.

4.3 Dependency tree style unification

So many treebanks inevitably adhere to many different annotation styles. Ideally, we would like to 1. identify all differences in annotation styles; 2. unify (normalize) the datasets, i.e. convert all to one annotation style; 3. for each phenomenon that is captured in at least two different ways in the original data, try each annotation approach one-by-one (by transforming occurrences of that phenomenon in the normalized data) and study its impact on parsing.

While we limit our present experiments on coordination structures only, we still strive to normalize all the differences we are able to identify, be they coordination-related or not.

We decided to derive our normalized form from the annotation style of the Prague Dependency Treebank. There are couple of reasons for this choice. In the area of coordination structures, almost half of the treebanks already use a PDT-like approach or are close to it; the PDT-like annotation of coordination is also the strongest in terms of expressive power, which is important in order not to loose information contained in the original data. Also, PDT is the largest manually annotated dependency treebank with very detailed annotation guidelines.[21]

The normalization procedure involves both structural transformation and dependency relation relabeling. While we try to design the structural transformations as reversible as possible, we do not attempt to save all the information stored in the labels. The DEPREL tagsets are *very* different across the treebanks, ranging from simple statements such as "this is a noun phrase modifying something" over standard *subject, object* etc. relations, to deep-level functions of Pāṇinian grammar such as *karma* and *karta*. It does not seem possible to unify these tagsets without manual relabeling of the whole treebanks. We use a lossy scheme that maps the DEPREL tags on the moderately-sized tagset of PDT analytical functions (more or less the same as the DEPREL tags in CoNLL Czech data). However, the only really important tags in our experiments are those that describe coordination. That is why we also use the unlabeled attachment score (UAS) as our main evaluation metric.

Occasionally the original structure and dependency labels are not enough to determine the normalized output, as we also need to consider the part-of-speech, the word form or even the values of morphological features. Since the POS/morphological tagsets also vary greatly across treebanks, we use the Interset approach described by [Zeman, 2008] to access all the morphological information. As a by-product, many of the normalized treebanks provide Interset-unified morphology, too.

4.3.1 Transformations related to coordination

Coordination-related transformations are described in detail in Section 3, and native styles for particular treebanks are listed in the *Orig. CS style* column of Table 4.1. Normalization thus means converting the original CS style to the PDT style. CS styles of most treebanks are easily classifiable using the codes introduced in Section 3, plus a few additional codes:

`p0` – punctuation was removed from the treebank;

[20] The terms *left* and *right* may be misleading for Arabic which is written right-to-left. Please note that `hL` is to be interpreted as "head closer to the beginning of the sentence" rather than "head on the left".

[21] Only part of PDT was included in CoNLL 2009 which we use in our experiments.

Language	Primary data source	Prim. tree type	Used data source	Sents.	Toks.	Train /test div. [%]	Orig. CS style	CSs / 100 toks.	CJs / CS	SMs / CS	embed. CS [%]
1: ar Arabic	Prague Ar. DT	dep	CoNLL 2007	3043	116793	96 / 4	fP hL^{20} sH cH pB dL m0	3.76	2.42	0.13	10.6
2: bg Bulgarian	BulTreeB.	phr	CoNLL 2006	13221	196151	97 / 3	fS hL sX cB pB dX m1	2.99	2.19	0.00	0.0
3: bn Bengali	Hyderab. DT	dep	ICON 2010	1129	7252	89 / 11	fP hR sH cH pP dU m3	4.87	1.71	0.05	24.1
4: cs Czech	Prague DT	dep	CoNLL 2007	25650	437020	99 / 1	fP hR sH cH pB dL m3	4.09	2.16	0.20	14.6
5: da Danish	Danish DT	dep	CoNLL 2006	5512	100238	94 / 6	fS$_1$ hL sX cP! pB dX m1	3.68	1.93	0.13	7.5
6: de German	Tiger TB	phr	CoNLL 2009	38020	680710	95 / 5	fM hL sX cP pP dX m1	2.79	2.09	0.01	0.0
7: el M. Greek	Greek DT	dep	CoNLL 2007	2902	70223	93 / 7	fP hR sH cH pB dL m3	3.25	2.48	0.18	7.2
8: en English	Penn TB	phr	CoNLL 2009	40613	991535	97 / 3	fM hL sX cB pP dX m1!	2.07	2.33	0.05	6.3
9: es Spanish	AnCora corpus	phr	CoNLL 2009	15984	477810	89 / 11	fS hL sX cB pB dX m1	2.79	1.98	0.14	12.7
10: eu Basque	Basque DT	dep	primary source	11225	151593	91 / 9	fP hR sX cH pP dU m0!	3.37	2.09	0.03	5.1
11: fi Finnish	Turku DT	dep	primary source	4307	58576	91 / 9	fS hL sX cB pB dX m1	4.06	2.41	0.00	6.4
12: grc A. Greek	A. Greek DT	dep	primary source	31316	461782	99 / 1	fP hR sH cH pB dL m3	6.54	2.17	0.16	10.3
13: hi Hindi	Hyderab. DT	dep	ICON 2010	3515	77068	84 / 16	fP hR sH cH pP dU m3	2.45	1.97	0.04	10.3
14: hu Hungarian	Szeged TB	phr	CoNLL 2007	6424	139143	95 / 5	fT h0 sX cX pX dA m0	2.37	1.90	0.01	2.2
15: it Italian	Italian SST	dep	CoNLL 2007	3359	76295	93 / 7	fS hL sX cB pB dX m1	3.32	2.02	0.03	3.8
16: la Latin	Latin DT	dep	primary source	3473	53143	91 / 9	fP hR sH cH pB dL m3	6.74	2.24	0.41	12.3
17: nl Dutch	Alpino TB	phr	CoNLL 2006	13735	200654	97 / 3	fP hR sX cH pP dU m1	2.06	2.17	0.05	3.3
18: pt Portuguese	Floresta Sint.	phr	CoNLL 2006	9359	212545	97 / 3	fS hL sX cB pB dX m1	2.51	1.95	0.26	11.1
19: ro Romanian	Romanian DT	dep	primary source	4042	36150	93 / 7	fP$_1$ hR sX cH p0 dU m1	1.80	2.00	0.00	0.0
20: ru Russian	Syntagrus	dep	primary source	34895	497465	99 / 1	fM hL sX cB p0 dX m1!	4.02	2.02	0.07	3.9
21: sl Slovene	Slovene DT	dep	CoNLL 2006	1936	35140	82 / 18	fP hR sH cH pB dL m0	4.31	2.49	0.00	10.8
22: sv Swedish	Talbanken 05	phr	CoNLL 2006	11431	197123	97 / 3	fM hL sX cF pF dX m1	3.94	2.19	0.13	0.7
23: ta Tamil	TamilTB	dep	primary source	600	9581	79 / 21	fP hR sH cH pB dL m3	1.66	2.46	0.22	3.8
24: te Telugu	Hyderab. DT	dep	ICON 2010	1450	5722	90 / 10	fP hR sH cH pP dU m3	3.48	1.59	0.06	5.0
25: tr Turkish	Turkish TB	dep	CoNLL 2007	5935	69695	94 / 6	fM hR sX cB pB dA m1	3.81	2.04	0.00	34.3

Table 4.1: Overview of used data resources. SM stands for shared modifier; CJ stands for conjunct. The last column shows what portion of CSs is participating in embedded CSs (both as the inner and outer CS).

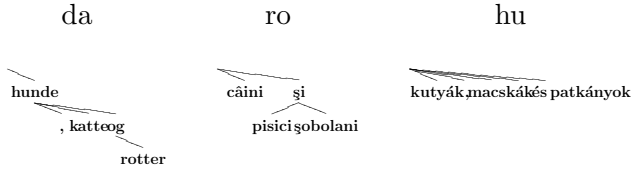


Figure 4.1: Annotation styles of a few treebanks do not fit well into the multidimensional space defined in Section 3.2.

sX – shared modifiers were not distinguished from the "private" modifiers (in most cases this results in the sN topology but the sX code makes explicit that no additional labeling was used to distinguish shared modifiers);

fT – Hungarian Szeged TB uses "Tesnière family": disconnected graphs for CSs (conjuncts are attached directly to the parent of CS; see Figure 4.1_{hu}).

There are a few points to emphasize. The sX class contains all non-Prague-style treebanks because they have no explicit notion of shared modifiers (these are attached to the head conjunct but they cannot be distinguished from the private modifiers of that conjunct). Our normalization procedure cannot recover the missing distinction reliably. We apply a few heuristics but in most cases the modifiers of the head conjunct will remain private modifiers after normalization. Danish employs a mixture of the Stanford and Moscow styles, where the last conjunct is attached indirectly via the conjunction (see Figure 4.1_{da}). The Romanian and Russian treebanks omit punctuation tokens (these do not have corresponding nodes in the tree); in the case of Romanian, this means that coordinations of more than two conjuncts get split (see Figure 4.1_{ro}).

4.3.2 Transformations not related to coordination

Besides coordination, numerous other phenomena can be normalized in treebanks. Here is a selection of those that we have observed and, to various degrees for various languages, included in our normalization scenario. (Language codes in parentheses give examples of treebanks where the particular approach is employed.)

- Prepositions (or postpositions) can either govern their noun phrase [cs, sl, en, ...] or they can be attached to the head of the NP [hi]. When they govern the NP, other modifiers of the main noun are usually attached to the noun but they can also be attached to the preposition [de]. The label of the relation of the PP to its parent can be found at the prepositional head [de, en, nl], or the preposition, despite serving as head, gets an auxiliary label (such as `AuxP` in PDT) and the real label is found at the NP head [cs, sl, ar, el, la, grc].

- Roots (predicates) of relative clauses are usually attached to the noun they modify (example: in *"the man that came yesterday"*, *"came"* would be attached to *"man"* and *"that"* would be attached to *"came"* as its subject). Some clauses use a subordinating conjunction (complementizer; e.g. *"that, dass, que, che"* if not used as a relative pronoun/determiner, example: *"the man said that he came yesterday"*). The conjunction can either be attached to the predicate of the embedded clause [es, ca, pt, de, ro] or it can lie between the clause and the main predicate it modifies [cs, en, hi, it, ru, sl]. In the latter case the label of the relation of the clause to its parent can be assigned to the conjunction [en, it, hi] or to the clausal predicate [cs, sl]. The comma before the conjunction is attached either to the conjunction or to the predicate of the clause. The Romanian treebank is segmented to clauses instead of sentences, so every clause has its own tree and inter-clausal relations are not annotated.

- Various sorts of verbal groups include analytical verb forms (such as auxiliary + participle), modal verbs with infinitives and similar constructions. Dependency relations, both internal (between group elements) and external (leading to parent on one side and verb modifiers on the other side), may be defined according to various criteria: content verb vs. auxiliary, finite form vs. infinitive, subject-verb agreement (typically holds for finite verbs and participles but not for infinitives). Participles often govern auxiliaries [es, ca, it, ro], elsewhere the finite verb is the head [pt, de, nl, en, sv] or both approaches are possible based on semantic criteria [cs]. In [hi], the content verb (which could be a participle or a bare verb stem) is the head and auxiliaries (finite or participles) are attached to it. The head typically bears the label describing the relation of the group to its parent. As for child nodes, subject and negative particle (if any) are often attached to the head, especially if it is the finite element [de, en] while the arguments (objects) are attached to the content element whose valency slot they fill (often participle or infinitive). Sometimes even the subject [nl] or the negative particle [pt] can be attached to the non-head content element. Various infinitive-marking particles (English *"to"*, Swedish *"att"*, Bulgarian *"da"*) can be treated similarly to subordinating conjunctions, can govern the infinitive [en, bg] or be attached to it. In [pt], prepositions used between main verb and the infinitive (*"estão a usufruir"*) are attached to the infinitive. In [bg], all modifiers of the verb including the subject are attached to *"da"* instead of the verb below.

- The Danish treebank is probably the most extraordinary one. Nouns often depend on determiners, numerals etc.: the opposite of what the rest of the world is doing.

- Paired punctuation (quotation marks, brackets, parenthesizing commas) is typically attached to the head of the segment between the marks. Occasionally it is attached one level higher, to the parent of the enclosed segment, which may break projectivity [pt]. Non-coordinating unpaired punctuation symbols are usually attached to a neighboring symbol or its parent. In [it], left paired marks are attached to the next token and all the others to the previous token.

- Sentence-final punctuation is attached to the artificial root node [cs, ar, sl, grc, ta], to the main predicate [bg, ca, da, de, en, es, et, fi, hu, pt, sv], to the predicate of the last clause [hi], to the previous token [eu, it, ja, nl]. In [la] there is no final punctuation. In [bn, te] it is rare but when present, it can govern a few previous tokens! In [tr], it is attached to the artificial root node but instead of being sibling of the main predicate, the punctuation governs the predicate.

Chapter 5

Experiments and Results

We evaluate the parser performance on the PDT style normalized treebanks and on the various CS-related transformations (due to space limitations, we have selected just few of them to be presented in Table 5.1). For contrast we also provide scores of the original unnormalized treebanks, although these numbers are comparable with results in the literature rather than with our normalized and transformed treebanks (see below why). Our central focus is on how various CS transformations affect the parsing accuracy when compared against the normalized PDT style treebank. The division of training and testing data for various language treebanks has already been mentioned in Table 4.1.

Our current results are preliminary because they do not yet include the inverse transformation suggested in Section 2 (i.e., a parser trained on transformed corpus is now evaluated against transformed test data, which in some cases makes the parsing task easier). Complete results with the inverse transformations will be available for the final version of the article.

5.1 Evaluation metric

Our main evaluation metric is the Unlabeled Attachment Score (UAS). We ignore the labels in order to reduce the impact of one of important factors that make treebank annotation schemes different.[1] Strictly speaking, our attachment score is slightly less "unlabeled" than is usual in the related work. Together with correct parent links, we also evaluate correctness of link types specific to CSs, namely `is_conjunct` and `is_shared_modifier`. We encode these binary attributes in the DEPREL labels when we train the parsers, and we extract them from parser-assigned DEPREL labels before dropping the labels.

For this reason our evaluation of the normalized and transformed treebanks is not directly comparable to the unnormalized treebanks, which contain only the original DEPREL tags without the possibility of encoding the two binary attributes.

Finally, we use confidence measures to address the significance of score differences between the transformations and the normalized PDT style treebank.

5.2 Used parsers and their settings

In our experiments we employed representatives of two contemporarily dominating families of dependency parsers, namely a graph-based parser and a transition-based parser.

In graph-based parsing, we learn a model for scoring graph edges, and we search for the highest-scoring tree composed of the graph's edges. We used Maximum Spanning

[1] Changes in edge labeling lead not only to different labeled attachment scores; they can influence also the UAS because transition-based parsers may use previously assigned labels as features for the following decisions.

Tree parser [McDonald and Pereira, 2006] which is capable of incorporating second order features (`MST` for short). We used MST parser in its version 0.4.3b (downloaded from `http://sourceforge.net/projects/mstparser/`) with second-order and non-projective setting (`order:2 decode-type:non-proj`).

Transition-based parsers utilize the shift-reduce algorithm. Input words are put into a queue and consumed by shift-reduce actions, while the output parser is gradually built. Unlike graph-based parsers, transition-based parsers have linear time complexity and allow straightforward application of non-local features. We used Malt parser (`MALT`) introduced in [Nivre et al., 2007b]. We used the Malt parser in its version 1.5 (downloaded from `http://maltparser.org/`), nivreeager algorithm, liblinear learner, and the default features (`-a nivreeager -l liblinear`).

5.3 Results

The results are summarized in Table 5.1. Transformations selected for the evaluation are described using the codes defined in Section 3.

5.4 Discussion

The current results do not show any widespread and consistent tendency. Some of the Moscow-family transformations gather multiple significant improvements cross-lingually and some languages seem to be affected more than others, possibly due to a bad baseline result. Statistical significance seems to be impacted by test data size (larger datasets yield significant results more often).

The main weakness of the current results is that the reverse transformation to the original annotation style has not been applied; unfortunately, one can expect that with the reverse transformation the improvement will be even less convincing (because reverse transformation can be lossy).

We have not investigated all possible graphs over the CS participants. We have not evaluated extra-dependency means of representing coordination. We deliberately limited ourselves to representations that suit well existing parsers but perhaps it would be better to adapt parser architecture to more specialized representations.

The PDT style, despite being the most expressive one among those used in treebanks, still falls short of representing CSs expressed using suffixes or otherwise lacking coordinating conjunction.

One possible (and probable) source of problems is the gigantic diversity among treebank annotation approaches. We have shown a sample of this Universe in Section 4.3; however, our current implementation of the unifying procedures is insufficient, many phenomena are tackled only approximatively by heuristics. Further refinement of the normalization steps could lead to more reliable results of the transformations and at the minimum it should reduce the drop in accuracy the normalized data show now.

Another source of low significance of the results could be low proportion of CS participants to other tree nodes. Separate evaluation of CS nodes is thus also of interest. Table 5.2 shows such partial evaluation of Malt parser output.

Lang.	orig	fP (PDT) hR sH cH pB	fM hL sN cB pP	fM hM sN cB pP	fM hR sH cB pP	fM hR sN cB pP	fP hR sN cH pB	fS hL sN cB pP	fS hM sN cB pP	fS hR sH cB pP	fS hR sN cB pP
ar	72.50 72.20	69.30±1.50 72.20±1.80	**2.00** 0.80	1.00 0.20	0.90 0.60	**1.60** 0.60	-0.10 - 0.40	1.30 - 0.70	**1.60** - 0.70	1.10 1.20	1.20 1.20
bg	88.10 86.30	80.50±1.30 79.00±1.90	-0.50 0.50	-0.80 - 0.10	0.20 0.20	0.40 0.30	0.10 - 0.40	-1.10 0.20	-2.00 - 0.20	-0.40 0.90	-0.30 0.90
bn	80.30 81.60	78.50±2.80 81.10±2.80	-0.40 - 0.60	0.20 - 0.50	0.50 0.10	-0.60 - 0.50	1.50 - 0.90	1.00 - 0.40	-0.20 - 0.40	-0.80 - 0.10	0.30 0.50
cs	75.40 68.60	75.10±1.90 68.90±2.70	-1.20 1.70	-2.40 0.90	-1.00 - 0.40	0.00 0.30	-2.40 - 2.20	-3.80 - 0.10	-4.50 - 0.20	-2.40 0.40	-1.70 1.10
da	88.10 84.30	81.40±1.50 75.70±1.40	-0.60 1.40	-1.30 0.40	-0.60 - 1.00	0.00 - 0.90	-1.30 - 1.40	-2.30 - 0.20	-2.90 - 0.50	-1.70 - 0.40	-1.10 - 0.20
de	88.50 81.50	82.90±0.70 74.90±0.80	-0.30 **1.20**	-0.90 0.50	-0.30 0.30	0.00 0.40	-0.40 - 0.40	-0.90 0.50	-1.50 - 0.10	-0.60 0.40	-0.70 0.50
el	73.60 72.80	74.10±1.80 72.40±1.40	0.40 1.20	-0.10 0.10	0.40 0.20	0.40 0.70	-1.50 - 0.70	-1.60 - 0.60	-1.80 - 0.80	-1.10 0.50	-0.50 0.70
en	90.90 86.20	85.80±0.90 79.40±1.00	-0.90 0.30	-1.40 - 0.20	-1.00 - 0.20	-1.00 - 0.20	-1.20 - 1.00	-1.80 - 0.10	-2.20 - 0.50	-1.50 0.40	-1.30 0.40
es	88.00 83.90	84.20±0.80 79.10±1.00	-0.70 0.70	-1.20 0.40	-0.80 - 0.30	-0.80 - 0.30	-1.00 - 1.50	-2.10 - 0.40	-2.40 - 0.70	-1.30 0.10	-1.20 0.00
eu	76.20 71.80	66.00±1.40 60.10±1.60	-1.30 0.50	-2.50 - 0.70	-1.70 0.30	-1.80 0.30	-2.60 - 2.70	-2.70 - 0.80	-3.50 - 1.50	-3.00 - 0.50	-3.30 - 0.70
fi	72.20 70.00	69.00±1.20 64.80±1.80	-1.70 **2.60**	-2.50 1.40	-2.00 0.20	-1.90 0.00	0.40 - 0.20	-3.80 1.30	-4.50 0.70	-2.40 0.50	-3.50 0.50
grc	56.20 42.50	55.10±1.60 43.40±1.80	-1.60 **2.40**	-0.90 **2.00**	-0.60 **2.70**	-0.60 **2.60**	-1.10 - 1.20	-1.70 **2.10**	-2.00 **2.60**	-1.20 1.60	-1.20 **2.10**
hi	76.90 86.60	71.40±1.60 81.90±1.90	1.10 0.20	**1.70** 0.00	0.90 0.10	0.60 0.00	0.00 - 1.00	1.40 - 0.20	**1.70** - 0.10	0.80 - 0.30	0.60 - 0.40
hu	80.40 76.10	76.10±1.90 71.50±1.90	-1.50 0.00	-2.00 - 0.50	-1.80 - 0.70	-1.70 - 0.70	-1.70 - 0.70	-2.40 - 0.70	-2.50 - 0.60	-1.80 - 0.50	-1.70 - 0.40
it	85.00 83.20	79.60±2.40 76.30±2.20	-1.20 0.30	-1.50 - 0.20	-1.40 - 0.40	-1.20 - 0.20	-1.60 - 0.80	-2.40 - 0.10	-2.40 - 0.50	-1.60 0.30	-1.90 0.30
la	56.30 44.90	54.80±2.30 42.40±1.70	1.60 **5.30**	0.70 **2.60**	1.50 **3.30**	**2.90** **5.50**	-0.60 - 0.70	-0.50 **3.80**	-1.20 **3.10**	-0.60 **2.00**	0.50 **2.40**
nl	83.80 75.10	78.60±1.50 70.00±2.00	-0.90 0.50	-1.60 - 0.70	-0.60 - 0.40	-0.30 - 0.30	-1.30 - 1.70	-3.80 - 1.40	-3.90 - 2.20	-1.60 - 0.60	-1.20 - 0.70
pt	87.80 85.40	82.00±1.40 77.80±2.10	-0.30 0.10	-0.50 - 0.20	0.00 - 0.20	-0.30 - 0.40	-0.90 - 1.60	-1.00 - 0.40	-1.10 - 0.40	-0.20 - 0.20	-0.40 - 0.40
ro	88.30 86.20	88.80±1.60 86.50±1.70	-0.90 - 0.10	-0.10 0.00	-0.20 - 0.50	-0.20 - 0.50	0.00 - 0.20	-1.20 - 0.50	-1.60 - 1.00	0.00 0.00	0.00 0.00
ru	? 58.90	78.10±1.60 84.40±1.80	**1.80** 0.50	0.40 - 0.10	0.60 - 0.30	0.60 - 0.30	0.70 - 0.20	1.40 - 0.30	**1.70** 0.50	1.00 - 0.10	1.30 0.00
sl	75.30 71.50	74.10±1.50 68.60±1.60	-0.20 **2.10**	-0.70 **1.70**	-0.80 1.20	-0.30 1.50	-0.60 - 0.80	-2.30 0.60	-2.70 0.20	-1.70 0.70	-1.60 0.90
sv	87.10 88.20	78.50±1.70 76.60±1.50	-0.10 0.20	-0.80 - 0.60	-0.90 - 1.00	-0.70 - 0.70	-1.50 - 2.40	-2.20 - 1.60	-2.80 - 1.70	-1.90 - 0.90	-1.70 - 1.20
ta	69.40 71.40	71.60±2.00 72.80±2.70	0.40 1.10	0.30 1.20	-0.60 0.40	0.30 1.00	0.00 - 0.10	-0.90 0.20	-0.40 1.30	0.70 1.10	0.30 1.60
te	86.90 87.30	87.20±3.70 88.00±3.50	1.20 2.60	-0.90 2.30	-1.20 0.40	-0.90 0.50	-0.10 0.00	0.10 2.30	-0.40 1.60	-0.80 0.50	-1.60 0.70
tr	78.30 72.70	76.30±1.90 72.10±2.10	-1.70 0.10	-1.00 - 0.30	-1.00 - 0.30	-0.90 - 0.30	-1.00 - 0.40	-1.40 0.00	-1.30 - 0.40	-1.80 - 0.60	-1.90 - 0.50
Aver.	76.22 75.57	75.96 72.80	-0.30 1.02	-0.75 0.38	-0.46 0.17	-0.26 0.34	-0.73 -0.94	-1.39 0.10	-1.71 -0.10	-0.99 0.26	-0.90 0.37
Significantly positive change			2 5	1 3	? 2	2 2	? ?	? 2	3 2	? 1	? 2
Insignificant change			21 20	16 22	22 23	20 23	21 22	13 22	8 21	16 24	20 23
Significantly negative change			2 ?	8 ?	3 ?	3 ?	4 3	12 1	14 2	9 ?	5 ?

Table 5.1: Parsing accuracy (UAS). Accuracy by MST in the upper part and accuracy by MALT in the lower part of each cell. The third column shows confidence intervals for each treebank. The columns of the other transformations indicate score differences rather than absolute numbers; statistically significant positive changes are typeset in boldface.

trans	ar	bg	bn	cs	da	de	el	en	es	eu	fi	grc	hi	hu
pdtstyle	32.8	49.8	86.9	37.6	52.9	50.7	29.4	47.6	43.7	54.4	41.6	28.2	60.4	47.9
fMpPcBhLsN	3.9	6.9	-11.6	8.7	4.5	12.9	6.0	5.9	8.5	0.5	5.7	-1.3	-2.4	7.3
fMpPcBhMsN	-2.2	1.1	-10.1		-1.8	4.2	0.6	-1.4	4.9	-8.6	-3.4	-4.0	-6.1	1.9
fMpPcBhRsH	3.2		-10.1	1.0	-7.5		4.4	0.0	-6.1	-2.0	-2.8		-2.6	1.4
fMpPcBhRsN	2.1	4.1	-10.1	1.8	-6.8	7.4	5.9	-0.0	-6.1	-3.2	-2.7	-2.0	-11.9	0.9
fPpBcHhRsH	-1.7	-2.9	-10.1	-7.3	-11.7	2.2	-6.2	-6.5	-11.7	-13.2	-2.1	-8.4	-7.6	-7.5
fPpBcHhRsN		-4.1	-7.2	-9.2	-10.9	1.0	-7.0	-6.1	-12.1	-13.2	-1.7	-10.3	-7.5	-7.5
fSpPcBhLsN	-8.6	-2.0	-7.2	-2.9	-8.5	4.9	-6.7	-3.2	-2.0	-9.4	-5.5	-2.9	-7.6	1.7
fSpPcBhMsN	-8.6	-5.2	-11.6	-5.4	-10.3	-2.9	-10.0		-6.1	-13.4		-2.0	-7.0	-6.8
fSpPcBhRsH	8.2	7.0	-10.1	-1.7	-0.7	6.2	4.7	5.8	1.6	-5.2	1.1	-7.1	-5.1	1.2
fSpPcBhRsN	7.1	6.9	-7.2	0.2	-1.2	7.0	2.3	5.8	1.3	-7.5	0.9	-8.2	-5.6	0.7

trans	it	la	nl	pt	ro	ru	sl	sv	ta	te	tr	better	worse	average
pdtstyle	39.6	23.1	52.2	54.0	71.6	55.6	36.0	51.4	40.2	69.8	48.8			48.93
fMpPcBhLsN	-1.3	6.6	5.9	-4.9	-0.6	1.7	3.8	1.8	10.7	13.2	0.5	19	7	3.52
fMpPcBhMsN	-6.4	0.6	-2.3	-13.2	-0.6	-2.3	-0.5	-2.0	6.8	11.3	-3.5	8	17	-1.54
fMpPcBhRsH	-4.5		1.0	-8.4	-5.4	-4.7	1.3	-0.4	4.9		-3.2	7	13	-1.99
fMpPcBhRsN	-3.2	12.1	0.3		-5.4	-2.3	0.2	1.3	5.8	-3.7	-3.5	11	13	-0.80
fPpBcHhRsH	-4.0	-2.7	-3.7	-17.2	-3.3	-1.1	-9.4	-8.1	-0.9	1.8	-5.5	2	23	-5.77
fPpBcHhRsN	-4.5	-4.9	-3.7	-20.5	-3.3	-1.7	-8.3	-7.9	-1.9	-3.7	-5.2	1	23	-6.51
fSpPcBhLsN		3.0	-9.3	-14.1	-10.8	1.4	-9.1	-10.6	-3.9	11.3	-2.0	5	20	-4.46
fSpPcBhMsN	-10.8		-14.6	-12.5	-13.5	-2.3	-12.1	-13.3	9.8	5.6	-1.4	2	21	-6.98
fSpPcBhRsH	0.0	1.9	-1.1	-10.3	2.7	2.9	-5.1	-3.6	10.7	0.0	-1.4	13	11	0.15
fSpPcBhRsN	0.3	0.4	-1.3	-12.5	2.7	3.8	-4.4	-4.1	10.7	0.0	-1.4	14	10	-0.12

Table 5.2: Accuracy measured on gold-standard CS participants (conjuncts, delimiters and shared modifiers) only. Nodes that are not part of a gold-standard CS are ignored. Unlike with overall scores, here the first fM transformation shows consistent improvement in many languages.

Chapter 6

Conclusion

We have conducted a systematic comparison of annotation and parsing of coordination structures within dependency treebanks of 25 languages – a broader and more comprehensive study than any other previously published work we are aware of.

Even though our current results are preliminary and the experiments can (and should) be more elaborated in future, the observed tendency is unconvincing and not very promising. In this sense, our observation is in line with that of [Tsarfaty et al., 2011].

On the other hand, the collection of normalized multilingual treebanks, which we are creating, is a unique resource that will be valuable for further research; while we cannot distribute the original treebanks, most of them are easily obtainable for the research community, and our conversion software is available for anyone interested.

Bibliography

[Aduriz et al., 2003] Aduriz, I., Aranzabe, M. J., Arriola, J. M., Atutxa, A., Díaz de Ilarraza, A., Garmendia, A., and Oronoz, M. (2003). Construction of a Basque dependency treebank. In *Proceedings of the 2nd Workshop on Treebanks and Linguistic Theories*.

[Afonso et al., 2002] Afonso, S., Bick, E., Haber, R., and Santos, D. (2002). "Floresta sintá(c)tica": a treebank for Portuguese. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC)*, pages 1968–1703.

[Atalay et al., 2003] Atalay, N. B., Oflazer, K., Say, B., and Inst, I. (2003). The annotation process in the Turkish treebank. In *In Proc. of the 4th Intern. Workshop on Linguistically Interpreteted Corpora (LINC*.

[Bamman and Crane, 2011] Bamman, D. and Crane, G. (2011). The Ancient Greek and Latin dependency treebanks. In Sporleder, C., Bosch, A., and Zervanou, K., editors, *Language Technology for Cultural Heritage*, Theory and Applications of Natural Language Processing, pages 79–98. Springer Berlin Heidelberg.

[Bengoetxea and Gojenola, 2009] Bengoetxea, K. and Gojenola, K. (2009). Exploring treebank transformations in dependency parsing. In *Proceedings of the International Conference RANLP-2009*, pages 33–38, Borovets, Bulgaria. Association for Computational Linguistics.

[Boguslavsky et al., 2000] Boguslavsky, I., Grigorieva, S., Grigoriev, N., Kreidlin, L., and Frid, N. (2000). Dependency treebank for Russian: Concept, tools, types of information. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*, pages 987–991. Association for Computational Linguistics Morristown, NJ, USA.

[Bosco et al., 2010] Bosco, C., Montemagni, S., Mazzei, A., Lombardo, V., Lenci, A., Lesmo, L., Attardi, G., Simi, M., Lavelli, A., Hall, J., Nilsson, J., and Nivre, J. (2010). Comparing the influence of different treebank annotations on dependency parsing.

[Brants et al., 2002] Brants, S., Dipper, S., Hansen, S., Lezius, W., and Smith, G. (2002). The TIGER treebank. In *Proceedings of the Workshop on Treebanks and Linguistic Theories*, Sozopol.

[Buchholz and Marsi, 2006] Buchholz, S. and Marsi, E. (2006). CoNLL-X shared task on multilingual dependency parsing. In *In Proc. of CoNLL*, pages 149–164.

[Civit et al., 2006] Civit, M., Martí, M. A., and Bufí, N. (2006). Cat3LB and Cast3LB: From constituents to dependencies. In Salakoski, T., Ginter, F., Pyysalo, S., and Pahikkala, T., editors, *FinTAL*, volume 4139 of *Lecture Notes in Computer Science*, pages 141–152. Springer.

[Csendes et al., 2005] Csendes, D., Csirik, J., Gyimóthy, T., and Kocsor, A. (2005). The Szeged treebank. In Matousek, V., Mautner, P., and Pavelka, T., editors, *TSD*, volume 3658 of *Lecture Notes in Computer Science*, pages 123–131. Springer.

[Călăcean, 2008] Călăcean, M. (2008). Data-driven dependency parsing for Romanian. Master's thesis, Uppsala University.

[Džeroski et al., 2006] Džeroski, S., Erjavec, T., Ledinek, N., Pajas, P., Žabokrtský, Z., and Žele, A. (2006). Towards a Slovene dependency treebank. In *Proceedings of the Fifth International Language Resources and Evaluation Conference, LREC 2006*, pages 1388–1391, Genova, Italy. European Language Resources Association (ELRA).

[Hajič et al., 2006] Hajič, J., Panevová, J., Hajičová, E., Sgall, P., Pajas, P., Štěpánek, J., Havelka, J., Mikulová, M., Žabokrtský, Z., and Ševčíková-Razímová, M. (2006). Prague Dependency Treebank 2.0. CD-ROM, Linguistic Data Consortium, LDC Catalog No.: LDC2006T01, Philadelphia.

[Hajič et al., 2009] Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N., and Zhang, Y. (2009). The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL-2009), June 4-5*, Boulder, Colorado, USA.

[Haverinen et al., 2010] Haverinen, K., Viljanen, T., Laippala, V., Kohonen, S., Ginter, F., and Salakoski, T. (2010). Treebanking Finnish. In Dickinson, M., Müürisep, K., and Passarotti, M., editors, *Proceedings of the Ninth International Workshop on Treebanks and Linguistic Theories (TLT9)*, pages 79–90.

[Husain et al., 2010] Husain, S., Mannem, P., Ambati, B., and Gadde, P. (2010). The ICON-2010 tools contest on Indian language dependency parsing. In *Proceedings of ICON-2010 Tools Contest on Indian Language Dependency Parsing*, Kharagpur, India.

[Kahane, 1997] Kahane, S. (1997). Bubble trees and syntactic representations. In *In Proceedings of the 5th Meeting of the Mathematics of the Language, DFKI, Saarbrucken.*

[Kromann et al., 2004] Kromann, M. T., Mikkelsen, L., and Lynge, S. K. (2004). Danish dependency treebank.

[Lombardo and Lesmo, 1998] Lombardo, V. and Lesmo, L. (1998). Unit coordination and gapping in dependency theory. In Kahane, S. and Polguère, A., editors, *Processing of Dependency-Based Grammars; proceedings of the workshop. COLING-ACL*, Montreal.

[Marinčič et al., 2007] Marinčič, D., Gams, M., and Žabokrtský, Z. (2007). Parsing aided by intra-clausal coordination detection. In Smedt, K. D., Hajič, J., and Kübler, S., editors, *Proceedings of the 6th International Workshop on Treebanks and Linguistic Theories (TLT 2007)*, volume 1 of *NEALT Proceedings Series*, pages 79–84, Bergen, Norway. North European Association for Language Technology.

[Mazziotta, 2011] Mazziotta, N. (2011). Coordination of verbal dependents in Old French: Coordination as a specified juxtaposition or apposition. In *Proceedings of International Conference on Dependency Linguistics (DepLing 2011.*

[McDonald and Nivre, 2007] McDonald, R. and Nivre, J. (2007). Characterizing the errors of data-driven dependency parsing models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 122–131, Prague, Czech Republic. Association for Computational Linguistics.

[McDonald and Pereira, 2006] McDonald, R. and Pereira, F. (2006). Online learning of approximate dependency parsing algorithms. In *Proceedings of EACL*, pages 81–88.

[Mel'čuk, 1988] Mel'čuk, I. A. (1988). *Dependency Syntax: Theory and Practice*. State University of New York Press.

[Montemagni et al., 2003] Montemagni, S., Barsotti, F., Battista, M., Calzolari, N., Corazzari, O., Lenci, A., Zampolli, A., Fanciulli, F., Massetani, M., Raffaelli, R., Basili, R., Pazienza, M. T., Saracino, D., Zanzotto, F., Mana, N., Pianesi, F., and Delmonte, R. (2003). Building the Italian syntactic-semantic treebank. In Abeillé, A., editor, *Building and using Parsed Corpora*, Language and Speech series, pages 189–210, Dordrecht. Kluwer.

[Nilsson et al., 2005] Nilsson, J., Hall, J., and Nivre, J. (2005). MAMBA meets TIGER: Reconstructing a Swedish treebank from antiquity. In *Proceedings of the NODALIDA Special Session on Treebanks*.

[Nilsson et al., 2006] Nilsson, J., Nivre, J., and Hall, J. (2006). Graph transformations in data-driven dependency parsing. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 257–264. Association for Computational Linguistics.

[Nivre et al., 2007a] Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. (2007a). The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL 2007 Shared Task. Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

[Nivre et al., 2007b] Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryiğit, G., Kübler, S., Marinov, S., and Marsi, E. (2007b). MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

[Novák and Žabokrtský, 2007] Novák, V. and Žabokrtský, Z. (2007). Feature Engineering in Maximum Spanning Tree Dependency Parser. In Matoušek, V. and Mautner, P., editors, *Lecture Notes in Artificial Intelligence, Proceedings of the 10th International Conference on Text, Speech and Dialogue*, volume 4629 of *Lecture Notes in Computer Science*, pages 92–98, Pilsen, Czech Republic. Springer Science+Business Media Deutschland GmbH.

[Ogren, 2010] Ogren, P. V. (2010). Improving syntactic coordination resolution using language modeling. In *Proceedings of the NAACL HLT 2010 Student Research Workshop*, HLT-SRWS '10, pages 1–6, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Popel and Žabokrtský, 2009] Popel, M. and Žabokrtský, Z. (2009). Improving English-Czech Tectogrammatical MT. *The Prague Bulletin of Mathematical Linguistics*, (92):1–20.

[Prokopidis et al., 2005] Prokopidis, P., Desipri, E., Koutsombogera, M., Papageorgiou, H., and Piperidis, S. (2005). Theoretical and practical issues in the construction of a Greek dependency treebank. In *In Proc. of the 4th Workshop on Treebanks and Linguistic Theories (TLT*, pages 149–160.

[Ramasamy and Žabokrtský, 2011] Ramasamy, L. and Žabokrtský, Z. (2011). Tamiltb.v0.1: A syntactically annotated corpora for Tamil.

[Simov and Osenova, 2005] Simov, K. and Osenova, P. (2005). Extending the annotation of BulTreeBank: Phase 2. In *The Fourth Workshop on Treebanks and Linguistic Theories (TLT 2005)*, pages 173–184, Barcelona.

[Smrž et al., 2008] Smrž, O., Bielický, V., Kouřilová, I., Kráčmar, J., Hajič, J., and Zemánek, P. (2008). Prague Arabic dependency treebank: A word on the million words. In *Proceedings of the Workshop on Arabic and Local Languages (LREC 2008)*, pages 16–23, Marrakech, Morocco. European Language Resources Association.

[Štěpánek, 2006] Štěpánek, J. (2006). *Závislostní zachycení větné struktury v anotovaném syntaktickém korpusu (nástroje pro zajištění konzistence dat) [Capturing a Sentence Structure by a Dependency Relation in an Annotated Syntactical Corpus (Tools Guaranteeing Data Consistence)]*. PhD thesis, Charles University in Prague, Faculty of Mathematics and Physics, Prague, Czech Rep.

[Surdeanu et al., 2008] Surdeanu, M., Johansson, R., Meyers, A., Màrquez, L., and Nivre, J. (2008). The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of CoNLL*.

[Taulé et al., 2008] Taulé, M., Martí, M. A., and Recasens, M. (2008). AnCora: Multilevel annotated corpora for Catalan and Spanish. In *LREC*. European Language Resources Association.

[Tesnière, 1959] Tesnière, L. (1959). *Eléments de syntaxe structurale*. Paris.

[Tratz and Hovy, 2011] Tratz, S. and Hovy, E. (2011). A fast, accurate, non-projective, semantically-enriched parser. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1257–1268, Edinburgh, Scotland, UK. Association for Computational Linguistics.

[Tsarfaty et al., 2011] Tsarfaty, R., Nivre, J., and Andersson, E. (2011). Evaluating dependency parsing: Robust and heuristics-free cross-annotation evaluation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 385–396, Edinburgh, Scotland, UK. Association for Computational Linguistics.

[van der Beek et al., 2002] van der Beek, L., Bouma, G., Daciuk, J., Gaustad, T., Malouf, R., van Noord, G., Prins, R., and Villada, B. (2002). Chapter 5. the Alpino dependency treebank. In *Algorithms for Linguistic Processing NWO PIONIER Progress Report*, Groningen, The Netherlands.

[Zeman, 2004] Zeman, D. (2004). *Parsing with a Statistical Dependency Model*. PhD thesis, Univerzita Karlova v Praze.

[Zeman, 2008] Zeman, D. (2008). Reusable tagset conversion using tagset drivers. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odjik, J., Piperidis, S., and Tapias, D., editors, *Proceedings of the Sixth International Language Resources and Evaluation Conference, LREC 2008*, pages 28–30, Marrakech, Morocco. European Language Resources Association (ELRA).

The ÚFAL/CKL Technical Report Series

ÚFAL

ÚFAL (Ústav formální a aplikované lingvistiky; http://ufal.mff.cuni.cz) is the Institute of Formal and Applied linguistics, at the Faculty of Mathematics and Physics of Charles University, Prague, Czech Republic. The Institute was established in 1990 after the political changes as a continuation of the research work and teaching carried out by the former Laboratory of Algebraic Linguistics since the early 60s at the Faculty of Philosophy and later the Faculty of Mathematics and Physics. Together with the "sister" Institute of Theoretical and Computational Linguistics (Faculty of Arts) we aim at the development of teaching programs and research in the domain of theoretical and computational linguistics at the respective Faculties, collaborating closely with other departments such as the Institute of the Czech National Corpus at the Faculty of Philosophy and the Department of Computer Science at the Faculty of Mathematics and Physics.

CKL

As of 1 June 2000 the Center for Computational Linguistics (Centrum komputační lingvistiky; http://ckl.mff.cuni.cz) was established as one of the centers of excellence within the governmental program for support of research in the Czech Republic. The center is attached to the Faculty of Mathematics and Physics of Charles University in Prague.

Technical Reports

The ÚFAL/CKL technical report series has been established with the aim of disseminate topical results of research currently pursued by members, cooperators, or visitors of the Institute. The technical reports published in this Series are results of the research carried out in the research projects supported by the Grant Agency of the Czech Republic, GAČR 405/96/K214 ("Komplexní program"), GAČR 405/96/0198 (Treebank project), grant of the Ministry of Education of the Czech Republic VS 96151, and project of the Ministry of Education of the Czech Republic LN00A063 (Center for Computational Linguistics). Since November 1996, the following reports have been published.

ÚFAL TR-1996-01 Eva Hajičová, *The Past and Present of Computational Linguistics at Charles University*
Jan Hajič and Barbora Hladká, *Probabilistic and Rule-Based Tagging of an Inflective Language – A Comparison*

ÚFAL TR-1997-02 Vladislav Kuboň, Tomáš Holan and Martin Plátek, *A Grammar-Checker for Czech*

ÚFAL TR-1997-03 Alla Bémová at al., Anotace na analytické rovině, *Návod pro anotátory (in Czech)*

ÚFAL TR-1997-04 Jan Hajič and Barbora Hladká, *Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structural Tagset*

ÚFAL TR-1998-05 Geert-Jan M. Kruijff, *Basic Dependency-Based Logical Grammar*

ÚFAL TR-1999-06 Vladislav Kuboň, *A Robust Parser for Czech*

ÚFAL TR-1999-07 Eva Hajičová, Jarmila Panevová and Petr Sgall, *Manuál pro tektogramatické značkování (in Czech)*

ÚFAL TR-2000-08 Tomáš Holan, Vladislav Kuboň, Karel Oliva, Martin Plátek, *On Complexity of Word Order*

ÚFAL/CKL TR-2000-09 Eva Hajičová, Jarmila Panevová and Petr Sgall, *A Manual for Tectogrammatical Tagging of the Prague Dependency Treebank*

ÚFAL/CKL TR-2001-10 Zdeněk Žabokrtský, *Automatic Functor Assignment in the Prague Dependency Treebank*

ÚFAL/CKL TR-2001-11 Markéta Straňáková, *Homonymie předložkových skupin v češtině a možnost jejich automatického zpracování*

ÚFAL/CKL TR-2001-12 Eva Hajičová, Jarmila Panevová and Petr Sgall, *Manuál pro tektogramatické značkování (III. verze)*

ÚFAL/CKL TR-2002-13 Pavel Pecina and Martin Holub, *Sémanticky signifikantní kolokace*

ÚFAL/CKL TR-2002-14 Jiří Hana, Hana Hanová, *Manual for Morphological Annotation*

ÚFAL/CKL TR-2002-15 Markéta Lopatková, Zdeněk Žabokrtský, Karolína Skwarská and Vendula Benešová, *Tektogramaticky anotovaný valenční slovník českých sloves*

ÚFAL/CKL TR-2002-16 Radu Gramatovici and Martin Plátek, *D-trivial Dependency Grammars with Global Word-Order Restrictions*

ÚFAL/CKL TR-2003-17 Pavel Květoň, *Language for Grammatical Rules*

ÚFAL/CKL TR-2003-18 Markéta Lopatková, Zdeněk Žabokrtský, Karolina Skwarska, Václava Benešová, *Valency Lexicon of Czech Verbs VALLEX 1.0*

ÚFAL/CKL TR-2003-19 Lucie Kučová, Veronika Kolářová, Zdeněk Žabokrtský, Petr Pajas, Oliver Čulo, *Anotování koreference v Pražském závislostním korpusu*

ÚFAL/CKL TR-2003-20 Kateřina Veselá, Jiří Havelka, *Anotování aktuálního členění věty v Pražském závislostním korpusu*

ÚFAL/CKL TR-2004-21 Silvie Cinková, *Manuál pro tektogramatickou anotaci angličtiny*

ÚFAL/CKL TR-2004-22 Daniel Zeman, *Neprojektivity v Pražském závislostním korpusu (PDT)*

ÚFAL/CKL TR-2004-23 Jan Hajič a kol., *Anotace na analytické rovině, návod pro anotátory*

ÚFAL/CKL TR-2004-24 Jan Hajič, Zdeňka Urešová, Alevtina Bémová, Marie Kaplanová, *Anotace na tektogramatické rovině (úroveň 3)*

ÚFAL/CKL TR-2004-25 Jan Hajič, Zdeňka Urešová, Alevtina Bémová, Marie Kaplanová, *The Prague Dependency Treebank, Annotation on tectogrammatical level*

ÚFAL/CKL TR-2004-26 Martin Holub, Jiří Diviš, Jan Pávek, Pavel Pecina, Jiří Semecký, *Topics of Texts. Annotation, Automatic Searching and Indexing*

ÚFAL/CKL TR-2005-27 Jiří Hana, Daniel Zeman, *Manual for Morphological Annotation (Revision for PDT 2.0)*

ÚFAL/CKL TR-2005-28 Marie Mikulová a kol., *Pražský závislostní korpus (The Prague Dependency Treebank) Anotace na tektogramatické rovině (úroveň 3)*

ÚFAL/CKL TR-2005-29 Petr Pajas, Jan Štěpánek, *A Generic XML-Based Format for Structured Linguistic Annotation and Its application to the Prague Dependency Treebank 2.0*

ÚFAL/CKL TR-2006-30 Marie Mikulová, Alevtina Bémová, Jan Hajič, Eva Hajičová, Jiří Havelka, Veronika Kolařová, Lucie Kučová, Markéta Lopatková, Petr Pajas, Jarmila Panevová, Magda Razímová, Petr Sgall, Jan Štěpánek, Zdeňka Urešová, Kateřina Veselá, Zdeněk Žabokrtský, *Annotation on the tectogrammatical level in the Prague Dependency Treebank (Annotation manual)*

ÚFAL/CKL TR-2006-31 Marie Mikulová, Alevtina Bémová, Jan Hajič, Eva Hajičová, Jiří Havelka, Veronika Kolařová, Lucie Kučová, Markéta Lopatková, Petr Pajas, Jarmila Panevová, Petr Sgall, Magda Ševčíková, Jan Štěpánek, Zdeňka Urešová, Kateřina Veselá, Zdeněk Žabokrtský, *Anotace na tektogramatické rovině Pražského závislostního korpusu (Referenční příručka)*

ÚFAL/CKL TR-2006-32 Marie Mikulová, Alevtina Bémová, Jan Hajič, Eva Hajičová, Jiří Havelka, Veronika Kolařová, Lucie Kučová, Markéta Lopatková, Petr Pajas, Jarmila Panevová, Petr Sgall,Magda Ševčíková, Jan Štěpánek, Zdeňka Urešová, Kateřina Veselá, Zdeněk Žabokrtský, *Annotation on the tectogrammatical level in the Prague Dependency Treebank (Reference book)*

ÚFAL/CKL TR-2006-33 Jan Hajič, Marie Mikulová, Martina Otradovcová, Petr Pajas, Petr Podveský, Zdeňka Urešová, *Pražský závislostní korpus mluvené češtiny. Rekonstrukce standardizovaného textu z mluvené řeči*

ÚFAL/CKL TR-2006-34 Markéta Lopatková, Zdeněk Žabokrtský, Václava Benešová (in cooperation with Karolína Skwarska, Klára Hrstková, Michaela Nová, Eduard Bejček, Miroslav Tichý) *Valency Lexicon of Czech Verbs. VALLEX 2.0*

ÚFAL/CKL TR-2006-35 Silvie Cinková, Jan Hajič, Marie Mikulová, Lucie Mladová, Anja Nedolužko, Petr Pajas, Jarmila Panevová, Jiří Semecký, Jana Šindlerová, Josef Toman, Zdeňka Urešová, Zdeněk Žabokrtský, *Annotation of English on the tectogrammatical level*

ÚFAL/CKL TR-2007-36 Magda Ševčíková, Zdeněk Žabokrtský, Oldřich Krůza, *Zpracování pojmenovaných entit v českých textech*

ÚFAL/CKL TR-2008-37 Silvie Cinková, Marie Mikulová, *Spontaneous speech reconstruction for the syntactic and semantic analysis of the NAP corpus*

ÚFAL/CKL TR-2008-38 Marie Mikulová, *Rekonstrukce standardizovaného textu z mluvené řeči v Pražském závislostním korpusu mluvené češtiny. Manuál pro anotátory*

ÚFAL/CKL TR-2008-39 Zdeněk Žabokrtský, Ondřej Bojar, *TectoMT, Developer's Guide*

ÚFAL/CKL TR-2008-40 Lucie Mladová, *Diskurzní vztahy v češtině a jejich zachycení v Pražském závislostním korpusu 2.0*

ÚFAL/CKL TR-2009-41 Marie Mikulová, *Pokyny k překladu určené překladatelům, revizorům a korektorům textů z Wall Street Journal pro projekt PCEDT*

ÚFAL/CKL TR-2011-42 Loganathan Ramasamy, Zdeněk Žabokrtský, *Tamil Dependency Treebank (TamilTB) – 0.1 Annotation Manual*

ÚFAL/CKL TR-2011-43 Nguy Giang Linh, Michal Novák, Anna Nedoluzhko, *Coreference Resolution in the Prague Dependency Treebank*

ÚFAL/CKL TR-2011-44 Anna Nedoluzhko, Jiří Mírovský, *Annotating Extended Textual Coreference and Bridging Relations in the Prague Dependency Treebank*

ÚFAL/CKL TR-2011-45 David Mareček, Zdeněk Žabokrtský, *Unsupervised Dependency Parsing*

ÚFAL/CKL TR-2011-46 Martin Majliš, Zdeněk Žabokrtský, *W2C – Large Multilingual Corpus*

ÚFAL TR-2012-47 Lucie Poláková, Pavlína Jínová, Šárka Zikánová, Zuzanna Bedřichová, Jiří Mírovský, Magdaléna Rysová, Jana Zdeňková, Veronika Pavlíková, Eva Hajičová, *Manual for annotation of discourse relations in the Prague Dependency Treebank*

ÚFAL TR-2012-48 Nathan Green, Zdeněk Žabokrtský, *Ensemble Parsing and its Effect on Machine Translation*

ÚFAL TR-2013-49 David Mareček, Martin Popel, Loganathan Ramasamy, Jan Štěpánek, Daniel Zemana, Zdeněk Žabokrtský, Jan Hajič *Cross-language Study on Influence of Coordination Style on Dependency Parsing Performance*