HamleDT:
Harmonized Multi-Language Dependency Treebank

Daniel Zeman, Ondřej Dušek, David Mareček, Martin Popel, Loganathan Ramasamy, Jan Štěpánek, Zdeněk Žabokrtský and Jan Hajič
Charles University in Prague, Faculty of Mathematics and Physics, ÚFAL

February 11, 2014

Abstract. We present HamleDT – a *HArmonized Multi-LanguagE Dependency Treebank*. HamleDT is a compilation of existing dependency treebanks (or dependency conversions of other treebanks), transformed so that they all conform to the same annotation style. In the present article, we provide a thorough investigation and discussion of a number of phenomena that are comparable across languages, though their annotation in treebanks often differs. We claim that transformation procedures can be designed to automatically identify most such phenomena and convert them to a unified annotation style. This unification is beneficial both to comparative corpus linguistics and to machine learning of syntactic parsing.

Keywords: dependency treebank, annotation scheme, harmonization

1. Introduction

Growing interest in dependency parsing is accompanied (and inspired) by the availability of new treebanks for various languages. Shared tasks such as CoNLL 2006–2009 (Buchholz and Marsi, 2006; Nivre et al., 2007; Surdeanu et al., 2008; Hajič et al., 2009) have promoted parser evaluation in multilingual settings. However, differences in parsing accuracy in different languages cannot be always attributed to language differences. They are often caused by variation in domains, sizes and annotation styles of the treebanks. The impact of data size can be estimated by learning curve experiments, but normalizing the annotation style is difficult. We present a method to transform the treebanks into a common style, including a software that implements the method. We have studied treebanks of 29 languages and collected a long list of variations.[1] We propose one common style (called HamleDT v1.5 style) and provide a transformation from original annotations to this style for almost all[2] the phenomena we identified. In addition to dependency

[1] The initial version has been described in Zeman et al. (2012).

[2] HamleDT v1.5 does not include the harmonization of verbal groups (see Section 5.4).

The final publication is available at link.springer.com
`http://link.springer.com/article/10.1007/s10579-014-9275-2`

tree structure normalization, we also unify the tagsets of both the part-of-speech/morphological tags and the dependency relation tags.

The motivation for harmonizing the annotation conventions used for different treebanks was already described in literature, e.g., by McDonald et al. (2013). Clearly, a unified representation of language data is supposed to facilitate the development of multilingual technologies. The harmonized set of treebanks should improve the interpretability and comparability of parsing accuracy results, and thus help to drive the development of dependency parsers towards multilingual robustness. For instance, the range of unlabeled attachment scores reached by a typical state-of-the-art supervised dependency parser in different languages spans an interval of around 10 percent points (given training data of a comparable size) and is even bigger for unsupervised parsers, as documented, e.g., by Mareček and Žabokrtský (2012). It is not entirely clear whether and to what extent this variance can be attributed to the peculiarities of the individual languages, or merely to the choice of annotation conventions used for the language. Using HamleDT should make it possible to separate these two sources of variance. Besides supervised and unsupervised multilingual parsing, homogeneity of the data is also essential for experiments on cross-lingual transfer of syntactic structures, be it based on projecting trees (Hwa et al., 2005) or on transferring delexicalized models (McDonald et al., 2011a).

The common style defined in HamleDT v1.5 serves as a reference point: the ability to say "our results are based on HamleDT v1.5 transformations of treebank XY" will facilitate the comparability of future results published in all these subfields.

The purpose of HamleDT is not to find a single choice of annotation conventions that ideally suits all possible tasks concerning syntactic structures, as this is hardly to be expected doable. However, assuming a different annotation convention fits a particular task better, it is much simpler to transform all the treebanks to the desired shape after they have been collected and unified in HamleDT.

Last but not least, we believe that the unified representation of linguistic content may be advantageous for linguists, enabling them to compare languages based on treebank material without the need to study multiple annotation guidelines.

2. Related Work

There have been a few attempts recently to address the same problem, namely:

- Schwartz et al. (2012) define two measures of syntactic *learnability* and evaluate them using five different parsers on varying annotation styles of six phenomena (coordination, infinitives, noun phrases, noun sequences, prepositional phrases and verb groups). They work only with English; they generate varying annotations during the conversion of the Penn TreeBank WSJ corpus (Marcus et al., 1993) constituency annotation to dependencies.

- Tsarfaty et al. (2011) compare the performance of two parsers on different constituency-to-dependency conversions of the (English) Penn Treebank. They do not see the solution in data transformations; instead, they develop an evaluation technique that is robust with respect to some[3] annotation styles.

- McDonald et al. (2011b) experiment with cross-language parser training, relying on a rather small universal set of part-of-speech tags. They do not transform syntactic structures, however. They note that different annotation schemes across treebanks are responsible for the fact that some language pairs work better together than others. They use English as the source language and Danish, Dutch, German, Greek, Italian, Portuguese, Spanish, and Swedish as target languages.

- Seginer (2007) discusses possible annotation schemes for coordination structures and relative clauses in relation to his *common cover link* representation.

- Bosco et al. (2010) compare three different dependency parsers developed and tested with respect to two Italian treebanks.

- Bengoetxea and Gojenola (2009) evaluate three types of transformations on Basque: transformation of subordinate sentences, coordinations and projectivization. An important difference between their approach and ours is that their transformations can change tokenization.

- Nilsson et al. (2006) show that transformations of coordination and verb groups improve parsing of Czech.

[3] The transformations are not robust to coordination styles.

3. Data

We identified over 30 languages for which treebanks exist and are available for research purposes. Most of them can either be acquired free of charge or are included in the Linguistic Data Consortium[4] membership fee.

Most of the treebanks are natively based on dependencies, but some were originally based on constituents and transformed via a head-selection procedure. For instance, Spanish phrase-structure trees were converted to dependencies using the method of Civit et al. (2006).

HamleDT v1.5 currently covers 29 treebanks, with several others to be added soon. Table I lists the treebanks along with their data sizes. In the following, we use ISO 639 language codes in square brackets to refer to the treebanks of these languages, so e.g. [en] refers to the English treebank. A list of all 29 treebanks with references is included in Appendix A.

Many treebanks (especially those used in CoNLL shared tasks) define a train/test data split. This is important for the comparability of experiments with automated parsing and part-of-speech tagging. We preserve the original data division and define test subsets for the remaining treebanks as well. In doing so, we try to keep the test size similar to the majority of CoNLL 2006/2007 test sets, i.e., roughly 5,000 tokens.

Throughout this article, a *dependency tree* is an abstract structure of *nodes* and *dependencies* that capture syntactic relations in a sentence. Nodes correspond to the *tokens* of the sentence, i.e. to words, numbers, punctuation and other symbols (see Section 5.7 for more on tokenization). Besides the actual word form, the node typically holds additional attributes of the token, such as its lemma and part of speech. Dependencies are directed arcs between nodes. Every node *is attached to (depends on)* exactly one other node, called its *parent.* We draw the dependency as an arrow going from the parent to the *child.* Thus every node has one incoming dependency and any number of outgoing dependencies. There is one exception: an artificial *root node* that does not correspond to any real token and has only outgoing dependencies. Dependencies have *labels* that mark the type of the relation.

Most diagrams in this article (Figure 1 and onwards) depict just a snippet of the sentence, i.e. a *subtree.* Selected tokens (word forms) are shown in a sequence respecting the word order, with dependencies drawn as labeled arrows between two tokens (nodes). The artificial root of the whole sentence is never shown; the root token of the subtree

[4] `http://www.ldc.upenn.edu/`

has one incoming dependency going straight down (from an invisible parent). The relation between the subtree and its invisible parent is labeled X (it does not make sense to show the real relation type without the parent).

Language	Prim. tree type	Used data source	Sents.	Tokens	Train / test [% sents]	Avg. sent. length	Nonprj. deps. [%]
Arabic (ar)	dep	C2007	3,043	116,793	96 / 4	38.38	0.37
Basque (eu)	dep	prim	11,226	151,604	90 / 10	13.50	1.27
Bengali (bn)	dep	I2010	1,129	7,252	87 / 13	6.42	1.08
Bulgarian (bg)	phr	C2006	13,221	196,151	97 / 3	14.84	0.38
Catalan (ca)	phr	C2009	14,924	443,317	88 / 12	29.70	0.00
Czech (cs)	dep	C2007	25,650	437,020	99 / 1	17.04	1.91
Danish (da)	dep	C2006	5,512	100,238	94 / 6	18.19	0.99
Dutch (nl)	phr	C2006	13,735	200,654	97 / 3	14.61	5.41
English (en)	phr	C2007	18,577	446,573	99 / 1	24.03	0.33
Estonian (et)	phr	prim	1,315	9,491	90 / 10	7.22	0.07
Finnish (fi)	dep	prim	4,307	58,576	90 / 10	13.60	0.51
German (de)	phr	C2009	38,020	680,710	95 / 5	17.90	2.33
Greek (el)	dep	C2007	2,902	70,223	93 / 7	24.20	1.17
Greek (grc)	dep	prim	21,160	308,882	98 / 2	14.60	19.58
Hindi (hi)	dep	I2010	3,515	77,068	85 / 15	21.93	1.12
Hungarian (hu)	phr	C2007	6,424	139,143	94 / 6	21.66	2.90
Italian (it)	dep	C2007	3,359	76,295	93 / 7	22.71	0.46
Japanese (ja)	dep	C2006	17,753	157,172	96 / 4	8.85	1.10
Latin (la)	dep	prim	3,473	53,143	91 / 9	15.30	7.61
Persian (fa)	dep	prim	12,455	189,572	97 / 3	15.22	1.77
Portuguese (pt)	phr	C2006	9,359	212,545	97 / 3	22.71	1.31
Romanian (ro)	dep	prim	4,042	36,150	93 / 7	8.94	0.00
Russian (ru)	dep	prim	34,895	497,465	99 / 1	14.26	0.83
Slovene (sl)	dep	C2006	1,936	35,140	79 / 21	18.15	1.92
Spanish (es)	phr	C2009	15,984	477,810	90 / 10	29.89	0.00
Swedish (sv)	phr	C2006	11,431	197,123	97 / 3	17.24	0.98
Tamil (ta)	dep	prim	600	95,81	80 / 20	15.97	0.16
Telugu (te)	dep	I2010	1,450	5,722	90 / 10	3.95	0.23
Turkish (tr)	dep	C2007	5,935	69,695	95 / 5	11.74	5.33

Table I.: Overview of data resources included in HamleDT v1.5. The average sentence length is the number of tokens divided by the number of sentences. Varying tokenization schemes obviously influence the numbers; see Section 5.7 for details on the individual languages. The *C* code in the fourth column means “CoNLL shared task”, *I* means “ICON” and *prim* means primary (non-shared-task) source. The last column gives the percentage of nodes attached non-projectively.

4. Harmonization

Our effort aims at identifying all syntactic constructions that are annotated differently in different treebanks. Once a particular construction is identified, we can typically find all its instances in the treebank using existing syntactic and morphological tags, i.e., with little or no lexical knowledge. Thanks to this fact, we were able to design algorithms to normalize the annotations of many linguistic phenomena to a single style, which we refer to as the HamleDT v1.5 style.

The HamleDT v1.5 style is mostly derived from the annotation style of the Prague Dependency Treebank (PDT, Hajič et al. (2006)).[5] This is a matter of convenience, to a large extent: This is the scheme with which the authors feel most at home, and many of the included treebanks already use a style similar to PDT. We do not want to claim that the HamleDT v1.5 style is objectively better than other styles. (Please note, however, that in case of coordination, the HamleDT v1.5 style provides a more expressive power than the other options, as described in Section 5.1.)

The normalization procedure involves both structural transformations and changes to dependency relation labels. While we strive to design the structural transformations to be as reversible as possible, we do not attempt to save all information stored in the dependency labels. The original[6] labels vary widely across treebanks, ranging from very simple, e.g., NMOD "generic noun modifier" in [en], over standard *subject, object*, etc. relations, to deep-level functions of Pāṇinian grammar such as *karta* and *karma* (k1 and k2) in [hi, bn, te].[7] It does not seem possible to unify these tagsets without relabeling whole treebanks manually.

[5] So far, there are only two differences between the PDT style (used in [cs]) and the HamleDT v1.5 style: handling of appositions (see Table III) and marking of conjuncts (in HamleDT, the root of a conjunct subtree is marked as conjunct even if it is a preposition or subordinating conjunction; in PDT, only content words are marked as conjuncts). By conjunct, we mean a member of coordination (unlike Quirk et al. (1985)). By content word, we mean autosemantic word, i.e. a word with a full lexical meaning, as contrasted with auxiliary. Note that PDT also has a more abstract layer of annotation (called *tectogrammatical*), but in this work, we only use the shallow dependencies (called *analytical* layer in PDT).

[6] Unless we explicitly say otherwise, we mean by "original" the data source indicated in Table I. It may actually differ from the *really original* treebank. For instance, some of the CoNLL data underwent a conversion procedure to the CoNLL format from other formats, and some information may have been lost in the process.

[7] In the Pāṇinian tradition, *karta* is the agent, doer of the action, and *karma* is the "deed" or patient. See Bharati et al. (1994).

We use a lossy scheme that maps the dependency labels on the moderately-sized tagset of PDT analytical functions[8] – see Table II.

Language	Atr	Adv	Obj	AuxP	Sb	Pred	Coord	AuxV	AuxC	*rest*
Arabic (ar)	36.5	6.4	9.1	14.2	6.3	3.1	4.0	0.0	2.3	18.2
Basque (eu)	19.6	24.0	8.7	0.0	7.2	5.7	3.4	8.3	1.0	22.2
Bengali (bn)	18.2	22.7	17.9	0.0	16.6	16.7	4.9	0.0	0.0	3.0
Bulgarian (bg)	23.3	8.8	12.8	14.6	7.7	7.3	3.1	0.8	3.3	18.4
Catalan (ca)	22.4	16.7	5.2	9.9	7.4	8.1	2.9	9.3	1.8	16.4
Czech (cs)	28.5	10.4	8.1	9.9	7.1	6.0	4.1	1.2	1.7	23.1
Danish (da)	23.8	12.2	12.1	10.7	9.8	5.3	3.4	0.0	3.4	19.3
Dutch (nl)	14.1	24.7	6.8	10.3	8.5	7.4	2.1	5.2	3.7	17.2
English (en)	30.0	12.0	5.7	9.8	7.9	4.3	2.2	4.0	1.8	22.2
Estonian (et)	12.8	25.7	6.6	5.9	13.0	14.1	1.3	2.6	0.6	17.4
Finnish (fi)	29.7	18.2	7.8	1.5	9.4	8.3	4.1	1.6	1.2	18.2
German (de)	31.2	11.8	10.4	10.1	7.9	5.3	2.8	0.5	1.2	18.7
Greek (grc)	15.4	13.0	14.2	3.8	7.7	8.6	6.5	0.0	1.4	29.4
Greek (el)	39.8	9.9	7.5	8.3	7.1	4.5	3.2	4.0	1.6	14.0
Hindi (hi)	26.8	13.4	9.6	21.1	6.8	5.3	2.4	6.3	1.6	6.8
Hungarian (hu)	30.4	13.9	5.2	1.6	5.9	8.3	2.4	1.3	1.6	29.2
Italian (it)	22.2	12.4	4.9	14.7	5.2	4.8	3.3	2.8	1.1	28.5
Japanese (ja)	11.5	16.6	0.6	5.8	3.4	7.3	0.3	0.0	0.0	54.6
Latin (la)	17.9	13.7	15.9	5.3	10.6	8.8	6.6	1.1	3.1	17.2
Persian (fa)	25.3	8.8	10.0	14.0	6.4	7.7	4.1	0.1	2.7	20.8
Portuguese (pt)	24.6	24.0	7.1	11.4	6.0	4.3	2.4	0.0	1.0	19.0
Romanian (ro)	27.7	13.3	7.2	17.6	8.5	11.2	1.8	7.7	0.0	5.0
Russian (ru)	30.4	16.9	16.3	12.3	10.4	6.2	4.0	0.0	1.6	1.9
Slovene (sl)	15.0	10.9	8.1	7.3	5.9	7.2	4.3	9.4	3.7	28.1
Spanish (es)	22.8	16.9	5.1	9.0	7.8	8.7	2.8	8.0	2.0	17.0
Swedish (sv)	19.3	19.5	6.9	9.3	10.8	6.4	3.9	2.5	2.7	18.8
Tamil (ta)	27.7	0.0	9.7	3.0	7.3	6.0	1.6	6.3	2.8	35.6
Telugu (te)	7.3	21.3	19.5	0.0	19.2	25.6	3.5	0.1	0.0	3.6
Turkish (tr)	38.5	8.0	10.8	1.9	6.9	9.5	3.8	0.0	1.4	19.2
Average	26.2	13.9	8.9	10.3	7.6	6.3	3.3	2.8	1.8	18.8

Table II.: Selected types of dependency relations and their relative frequency in the harmonized treebanks. One can see repeated patterns in the table such as the dominance of adverbials and attributes, or the relatively stable proportion of subjects. However, the numbers are still biased by imperfections in the conversion procedures (e.g., unrecognized AuxV in certain languages). The abbreviations are inherited from the PDT: Atr = attribute, Adv = adverbial, Obj = object, AuxP = preposition, Sb = subject, Pred = predicate, Coord = coordinating conjunction, AuxV = auxiliary verb, AuxC = subordinating conjunction.

[8] They are approximately the same as the dependency relation labels in the Czech CoNLL data set. To illustrate the mapping, more details on [bn] and [en] conversion are presented in Tables IV and V in Appendix B.

Occasionally the original structure and dependency labels are not enough to determine the normalized output. For instance, the German label RC is assigned to all dependencies that attach a subordinate clause to its parent. The set of HamleDT v1.5 labels distinguishes clauses that act as nominal attributes (`Atr`) from those that substitute adverbial modifiers (`Adv`). We look at the part of speech of the parent: if it is a noun, we label the dependency `Atr`; if it is a verb, we label it `Adv`.[9] Thus we also consider the part of speech, the word form, or even further morphological properties. Since the morphological (part-of-speech) tagsets also vary greatly across treebanks, we use the Interset approach described by Zeman (2008) to access all morphological information. Interset is a kind of interlingua for parts of speech and morphosyntactic features. Its aim is to provide a unified representation for as many feature values in existing tagsets as possible. We created converters ("drivers") to Interset from all treebank tagsets for which it had not already been available. The normalized treebanks thus provide Interset-unified morphology as well.

In a typical scenario, the harmonization steps are ordered as follows:

1. file format conversion (from various proprietary formats to a common-schema XML) and character encoding conversion (to UTF-8),

2. conversion of morphological tags to the Interset tagset,

3. conversion of dependency relation labels to the set of HamleDT labels,

4. conversion of coordination structures into the HamleDT style (i.e., distinguishing members of coordination and shared modifiers, and attaching them to the main coordination conjunction),

5. other changes in the tree structure (i.e., rehanging nodes to make the dependent-governor relations comply with the HamleDT conventions, including relation orientation) and possibly further refinements of the dependency labels.

The last two points (tree transformations) represent the main focus of the present study; many detailed examples are provided in Section 5.

The implementation of file format converters is relatively straightforward, even though reverse engineering is sometimes needed due to missing technical documentation.

[9] Ideally we would also want to distinguish objects (`Obj`) from adverbials. Unfortunately, this particular source annotation does not provide enough information to make such a distinction.

When implementing the Interset converters, around 200–500 lines of Perl code are typically needed; the code is usually not very challenging from the algorithmic point of view, but requires a very good insight into the annotation guidelines of the respective resource.

Mapping of dependency labels is usually relatively simple to implement too: sometimes it is enough just to recode the original label (e.g. `Subj` to `Sb`), sometimes the decision must be conditioned by the POS value of the node or of its parent, sometimes the rules are conditioned lexically or by certain structural properties of the tree. However, it all can be done relatively reliably.

More or less the same holds for rehanging the nodes in the fifth step. Typically, there are just a few dozens of transformation rules needed for the third and fifth step (i.e., around 200 lines of Perl code).

The algorithmically most complex step in the harmonization is typically a proper treatment of coordination structures because resolving a coordination structure affects at least three nodes in most cases, coordinations can be nested, and they can combine with almost any dependency relation type. In addition, there are multiple different encodings of coordination structures used in treebanks (17 in HamleDT v1.5), as analyzed in depth by Popel et al. (2013).

Performing the normalization of coordination structures before the normalization of other relations brings about an important advantage: in step 5, it is possible to work with dependent-governor pairs of nodes in the sense of dependency (not just with child-parent node pairs as stored in the trees), disregarding whether the former or the latter (or both) are coordinated. Without this abstraction, even simple operations, such as swapping the relation orientation between nouns and prepositions, would become quite cumbersome, as one would have to keep all possible combinations in mind, e.g. “with A and B”, “with A and with B”, “with A and B or with C and D”, “with or without A”, etc. For more details, please refer to concrete examples in Section 5.

5. Annotation Styles for Various Phenomena

In this section, we present a selection of phenomena that we observed and, to various degrees for various languages, included in our normalization scenario. Language codes in brackets give examples of treebanks where the particular approach is employed. The ☞ symbol in figure captions marks artificial examples. Figures not marked with ☞ contain genuine examples found in real data, though some of them have been shortened.

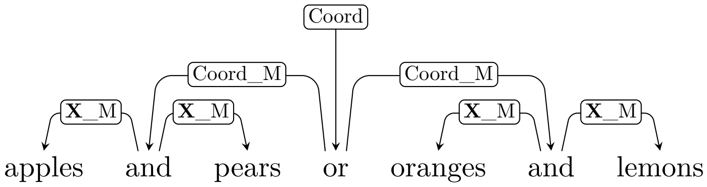

Figure 1. ☞ Nested coordination in the Prague style. `X` represents the relation of the whole structure to its parent. `_M` denotes *members* of coordination, i.e., conjuncts.

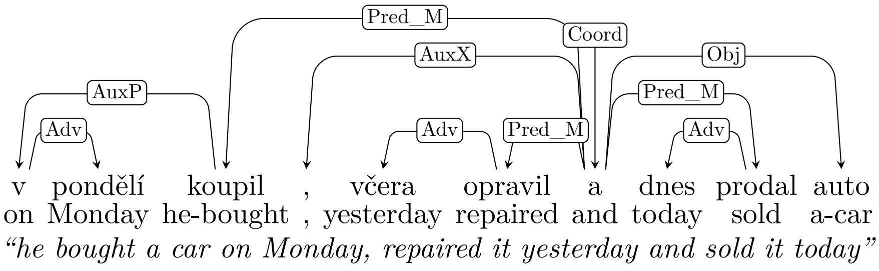
Pred_M
Coord
AuxX
Obj
AuxP
Pred_M
Adv
Adv
Pred_M
Adv
v pondělí koupil , včera opravil a dnes prodal auto
on Monday he-bought , yesterday repaired and today sold a-car
"he bought a car on Monday, repaired it yesterday and sold it today"

Figure 2. ☞ Shared and private modifiers in the Prague style [cs]: *Car (auto)* is an object shared by all three verbs while the adverbials *(on Monday, yesterday, today)* are private. The whole structure is in the predicate relation to its parent (which is probably the sentence root), so using the notation of Figure 1: `X = Pred`.

Dependency relation labels from the original treebanks that appear in figures are briefly explained in Appendix C.

5.1. Coordination

Capturing coordination in a dependency framework has been repeatedly described as difficult for both treebank designers and parsers (and it is generally regarded as an inherent difficulty of dependency syntax as such). Our analysis revealed four families of approaches, which may further vary in the attachment of punctuation, shared modifiers, etc.:

- *Prague* (Figures 1, 2 and 8). All conjuncts are headed by the conjunction. Used in [ar, bn, cs, el, en, eu, grc, hi, la, nl, sl, ta, te] (Hajič et al., 2006).

- *Mel'čukian* (Figure 3). The first/last conjunct is the head, others are organized in a chain. Used in [de, ja, ru, sv, tr] (Mel'čuk, 1988).

- *Stanford* (Figure 4). The first/last conjunct is the head, others are attached directly to it. Used in [bg, ca, es, fi, it, pt] (de Marneffe and Manning, 2008). And

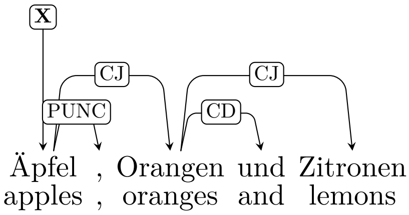

Figure 3. ☞ Coordination in the Mel’čukian style as seen in [de].

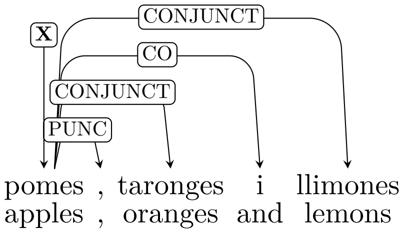

Figure 4. ☞ Coordination in the Stanford style as seen in [ca].

– *Tesnièrian* (Figure 5). There is no common head, all conjuncts are attached directly to the node modified by the coordination structure. Used in [hu] (Tesnière, 1959).

Furthermore, the Prague style provides for nested coordinations, as in *“apples and pears or oranges and lemons”* (see Figure 1). The asymmetric treatment of conjuncts in the other styles makes nested coordination difficult to read or even impossible to capture in some situations. The Prague style also distinguishes between shared modifiers, such as the subject in *“Mary came and cried”*, from private modifiers of the conjuncts, as in *“John came and Mary cried”* (see Figure 2). Because this distinction is missing in non-Prague-style treebanks, we cannot recover it reliably. We apply several heuristics, but in most cases, the modifiers of the head conjunct are classified as private modifiers.

Danish (Figure 6) employs a mixture of the Stanford and Mel’čukian styles where the last conjunct is attached indirectly via the conjunction. The Romanian and Russian treebanks omit punctuation tokens (they do not have corresponding nodes in the trees); in the case of Romanian, this means that coordinations of more than two conjuncts are disconnected (Figure 7).

Given the advantages described above, we decided to use the Prague style (in its [cs] flavor) in our harmonized data. There is just one drawback that we are aware of: Occasionally, there may be no node suitable for the coordination head. Most asyndetic constructions do not pose

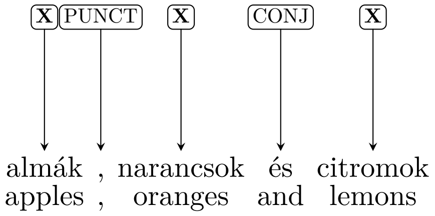

Figure 5. ☞ Coordination in the Tesnièrian style as seen in [hu]. All participating nodes are attached directly to the parent of the coordination.

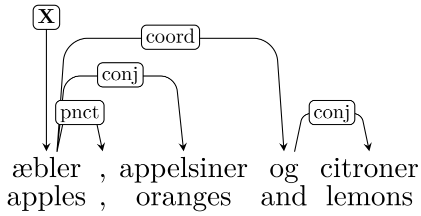

Figure 6. ☞ Danish mixture of Stanford and Mel'čukian coordination styles.

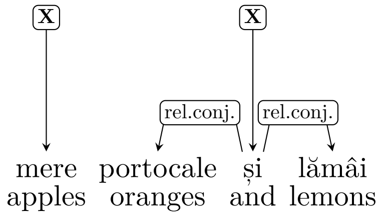

Figure 7. ☞ [ro] uses Prague coordination style mixed with Tesnièrian because punctuation is missing from data.

a problem because there are commas or other punctuation. Without punctuation, the Prague style would need an extra node—that solution has been adopted by the authors of the [ta] treebank (see Figure 8). Note that one-half of our treebanks already use the Prague style as their native approach, thus they always have a coordination head. In the other half, a fraction of coordinate structures cannot be fully converted (unless we add a new node, which we do not in the current version of HamleDT). For example, 14 out of the 5,988 coordinate structures in [bg] (0.23 %) lack any conjunction or punctuation that could be made the head. In these cases we currently use the first conjunct instead, effectively backing off to the Stanford style.

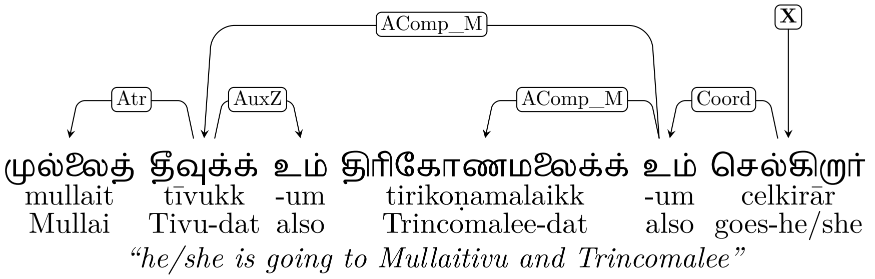
AComp_M
X
Atr
AuxZ
AComp_M
Coord
முல்லைத் தீவுக்க் உம் திரிகோணமலைக்க் உம் செல்கிறார்
mullait tīvukk -um tirikoṇamalaikk -um celkirār
Mullai Tivu-dat also Trincomalee-dat also goes-he/she
“he/she is going to Mullaitivu and Trincomalee”

Figure 8. Coordination in [ta]: The coordinating function is performed by the two morphological suffixes *-um*. They had to be made separate nodes during tokenization because [ta] uses the Prague style and no other coordination head was available except these morphological indicators.

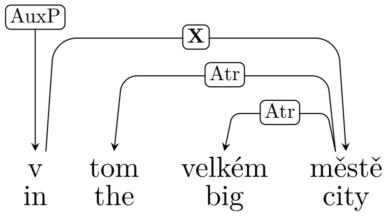

Figure 9. ☞ A prepositional phrase in [cs].

5.2. Prepositions

Prepositions (or postpositions; Figures 9–11) can either govern their noun phrase (NP) [cs, en, sl, ...] or modify the head of its NP [hi]. When they govern the NP, other modifiers of the main noun are attached either to the noun (in most cases) or to the preposition [de]. The label of the relation of the prepositional phrase to its parent is sometimes found at the preposition [de, en, nl]. Elsewhere, the preposition gets an auxiliary label (such as `AuxP` in PDT) despite serving as head, and the real label is found at the NP head [cs, sl, ar, el, la, grc].

In HamleDT v1.5 style, prepositions govern their noun phrase because 1. they may govern the form of the noun phrase (e.g. [cs, ru, sl, de]) and 2. this is the approach taken in most of the treebanks we studied. Other modifiers inside the prepositional phrase, such as determiners and adjectives, should depend on the embedded noun phrase. The preposition is labeled with the auxiliary tag `AuxP` and the real relation between the prepositional phrase and its parent is labeled at the NP head.

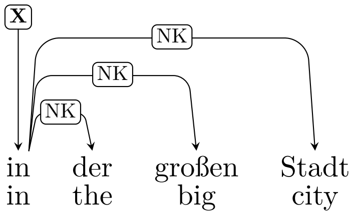

Figure 10. ☞ A prepositional phrase in [de].

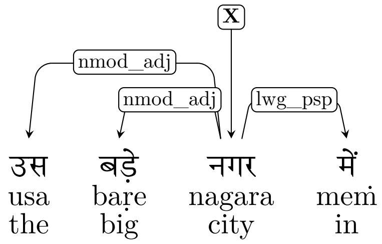

Figure 11. ☞ A postpositional phrase in [hi].

5.3. Subordinate Clauses

There are three main types of subordinate clauses:

- *Relative clauses.* They modify noun phrases. Typically they are marked by relative pronouns that represent the modified noun and its function within the relative clause. Example: *"The man who came yesterday."*

- *Complement clauses.* They serve as arguments of predicates, typically verbs. They are marked by subordinating conjunctions. Example: *"The man said that he came yesterday."*

- *Adverbial clauses.* They modify predicates in the same way as adverbs; but they are not selected as arguments. Example: *"If the man comes today he will say more."*

Roots (predicates) of relative clauses are usually attached to the noun they modify, e.g., in *"the man who came yesterday"*, *"came"* would be attached to *"man"* and *"who"* would be attached to *"came"* as its subject.

The predicate-modifying clauses use a subordinating conjunction (complementizer, adverbializer) to express their relation to the governing predicate. In treebanks, the conjunction is either attached to the predicate of the subordinate clause [es, ca, pt, de, ro] (Figure 12) or it lies between the embedded clause and the main predicate it modifies

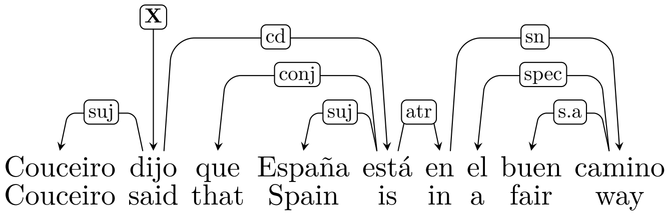
X
cd
sn
conj
spec
suj
suj
atr
s.a
Couceiro dijo que España está en el buen camino
Couceiro said that Spain is in a fair way

Figure 12. Subordinate clause in [es].

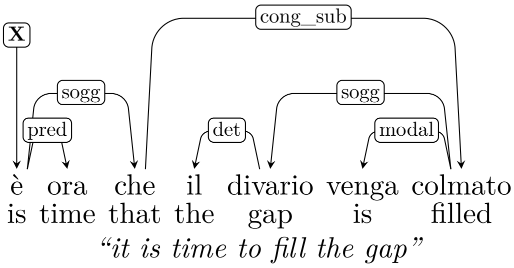
cong_sub
X
sogg
sogg
pred
det
modal
è ora che il divario venga colmato
is time that the gap is filled
"it is time to fill the gap"

Figure 13. Subordinate clause in [it].

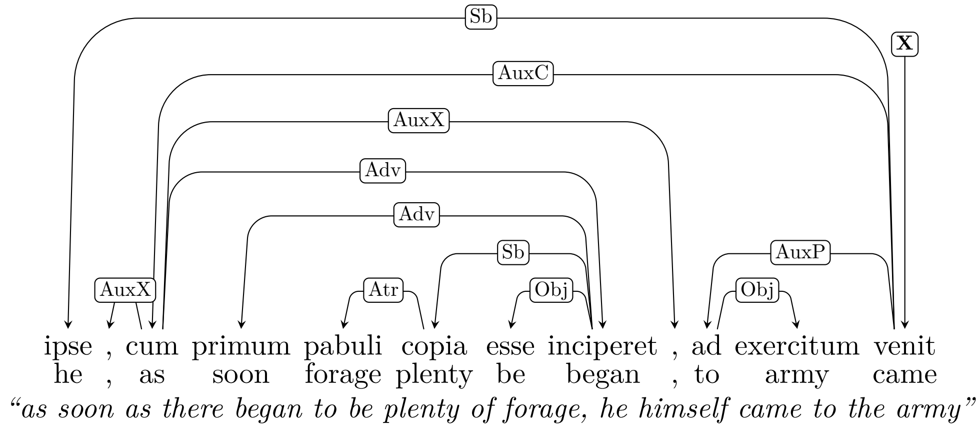
Sb
X
AuxC
AuxX
Adv
Adv
Sb
AuxP
AuxX
Atr
Obj
Obj
ipse , cum primum pabuli copia esse inciperet , ad exercitum venit
he , as soon forage plenty be began , to army came
"as soon as there began to be plenty of forage, he himself came to the army"

Figure 14. Subordinate clause in [la].

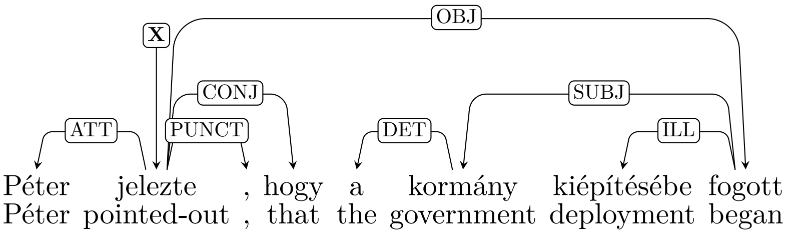
OBJ
X
CONJ
SUBJ
ATT
PUNCT
DET
ILL
Péter jelezte , hogy a kormány kiépítésébe fogott
Péter pointed-out , that the government deployment began

Figure 15. Subordinate clause in [hu].

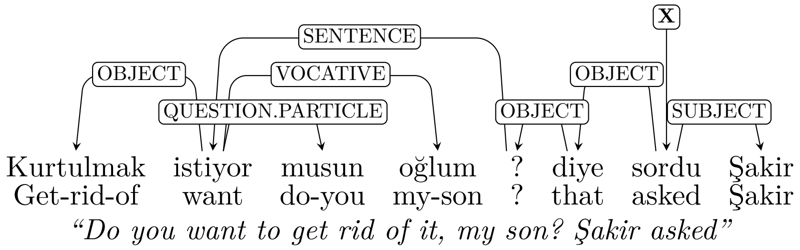
X
SENTENCE
OBJECT
VOCATIVE
OBJECT
QUESTION.PARTICLE
OBJECT
SUBJECT
Kurtulmak istiyor musun oğlum ? diye sordu Şakir
Get-rid-of want do-you my-son ? that asked Şakir
“Do you want to get rid of it, my son? Şakir asked”

Figure 16. Subordinate clause (direct speech) in [tr]: Here, the question mark serves as the head of the direct question.

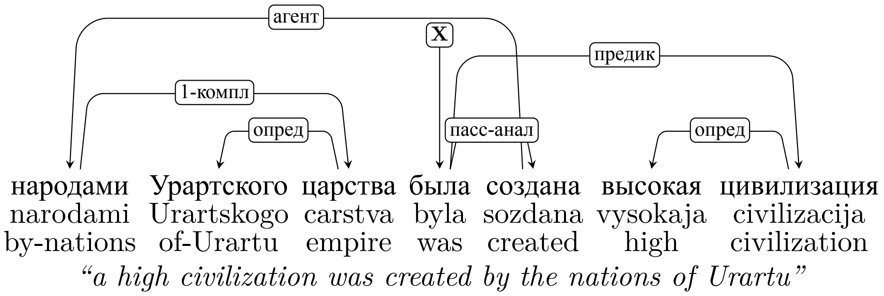
агент
X
предик
1-компл
опред
пасс-анал
опред
народами Урартского царства была создана высокая цивилизация
narodami Urartskogo carstva byla sozdana vysokaja civilizacija
by-nations of-Urartu empire was created high civilization
“a high civilization was created by the nations of Urartu”

Figure 17. Passive construction in [ru]: Finite auxiliary verb *(была)* is the head, passive participle *(создана)* depends on it. As a result, the agent *(народами)* is attached non-projectively to the participle *(создана)*.

[cs, en, hi, it, la, ru, sl] (Figure 13). In the latter case, the label of the relation of the subordinate clause to its parent can be assigned to the conjunction [en, hi, it] or to the clausal predicate [cs, la, sl] (Figure 14). The comma before the conjunction is attached either to the conjunction or to the subordinate predicate.

The subordinating conjunction may also be attached as a sibling of the subordinate clause [hu], an analogy to the Tesnièrian coordination style (Figure 15). In Figure 16, a direct question in [tr] is rooted by the question mark, which is attached to a subordinating postposition.

The Romanian treebank is segmented into clauses instead of sentences, so every clause has its own tree, and inter-clausal relations are not annotated.

HamleDT v1.5 style follows the [cs, sl, la] approach to subordinate clauses (see Figures 14 and 19).

5.4. Verb Groups

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 ele-

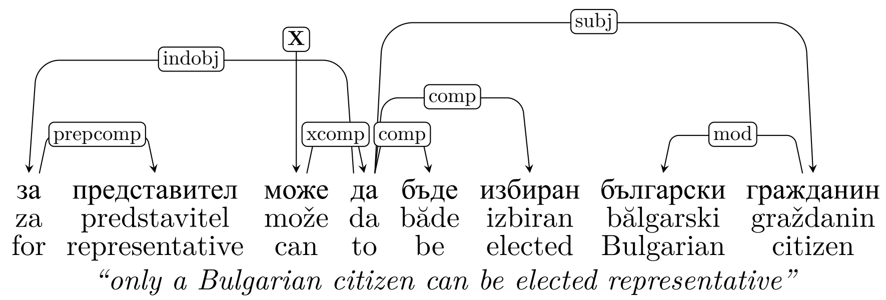
subj
indobj
X
comp
prepcomp
xcomp
comp
mod
за представител може да бъде избиран български гражданин
za predstavitel može da băde izbiran bălgarski graždanin
for representative can to be elected Bulgarian citizen
“only a Bulgarian citizen can be elected representative”

Figure 18. Modal passive construction in [bg]: The finite modal verb *(може)* is the head, the infinitive particle *(да)* is the second-level head. The infinitive auxiliary *(бъде)* is attached to *да*, as is the passive participle of the content verb *(избиран)* and the two arguments of the content verb, one of them *(за представител)* non-projectively.

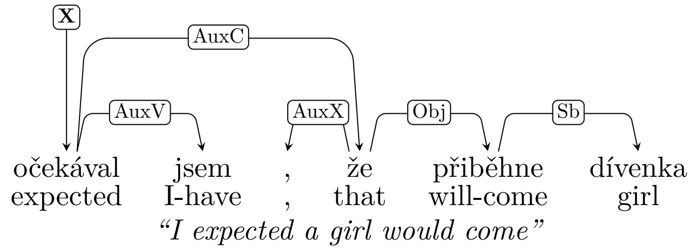
X
AuxC
AuxV
AuxX
Obj
Sb
očekával jsem , že přiběhne dívenka
expected I-have , that will-come girl
“I expected a girl would come”

Figure 19. Past tense in [cs]: The participle of the content verb *(očekával)* governs the finite form of the auxiliary *(jsem)*. Making the auxiliary the head would cause problems because it is not always present, e.g., omitting it in this sentence would just shift the sentence to the 3rd person meaning *(“He expected a girl would come.”)*

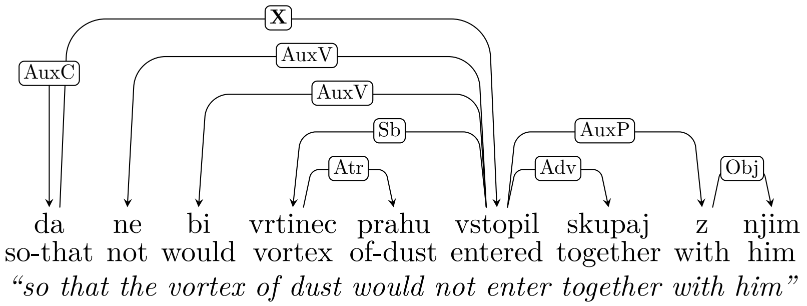
X
AuxV
AuxC
AuxV
Sb
AuxP
Atr
Adv
Obj
da ne bi vrtinec prahu vstopil skupaj z njim
so-that not would vortex of-dust entered together with him
“so that the vortex of dust would not enter together with him”

Figure 20. Negated conditional construction in [sl]. The past participle of the content verb *(vstopil)* is the head, the negative particle *(ne)* and the auxiliary *(bi)* depend on it.

ments) and external (leading to the parent on the one side and verb modifiers on the other side), may be defined according to various criteria: content verb vs. auxiliary, finite form vs. infinitive, or subject-

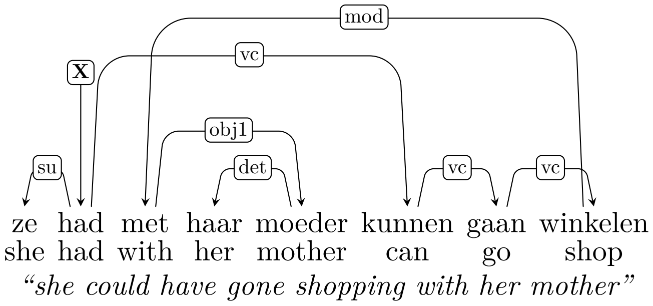
mod
vc
X
obj1
su
det
vc
vc
ze had met haar moeder kunnen gaan winkelen
she had with her mother can go shop
"she could have gone shopping with her mother"

Figure 21. Past modal construction in [nl]. The finite auxiliary verb *(had)* is the head. The subject *(ze)* is attached to the finite verb *(had)* while the modifier *(met haar moeder)* is attached non-projectively to the content verb *(winkelen)*.

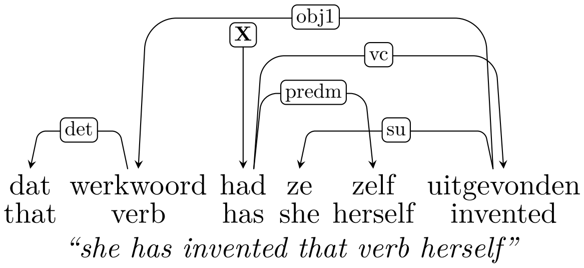
obj1
X
vc
predm
det
su
dat werkwoord had ze zelf uitgevonden
that verb has she herself invented
"she has invented that verb herself"

Figure 22. Another example from [nl]. Unlike in other treebanks, even the subject *(ze)* is attached to the non-head participle *(uitgevonden)*.

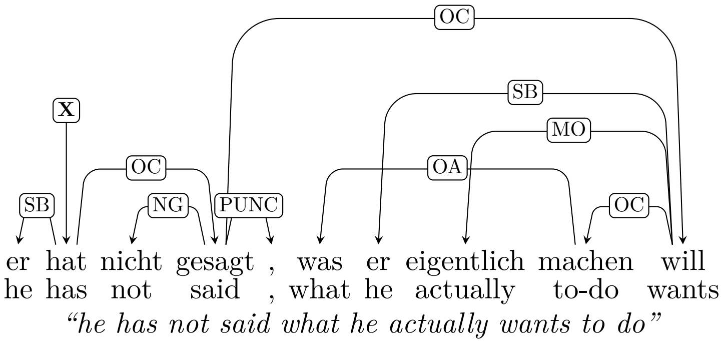
OC
SB
X
MO
OC
OA
SB
NG
PUNC
OC
er hat nicht gesagt , was er eigentlich machen will
he has not said , what he actually to-do wants
"he has not said what he actually wants to do"

Figure 23. A combination of perfect tense, modal verb, and infinitive in [de]. Infinitives are attached to modals as their objects in many treebanks, including [de]. The finite auxiliary verb *(hat)* is the head of the perfect tense, the participle *(gesagt)* depends on it. The subject *(er)* is attached to the finite verb *(hat)* while the object clause *(was er eigentlich machen will)* is attached to the content verb *(gesagt)*.

verb agreement, which typically holds for finite verbs, sometimes for participles but not for infinitives.

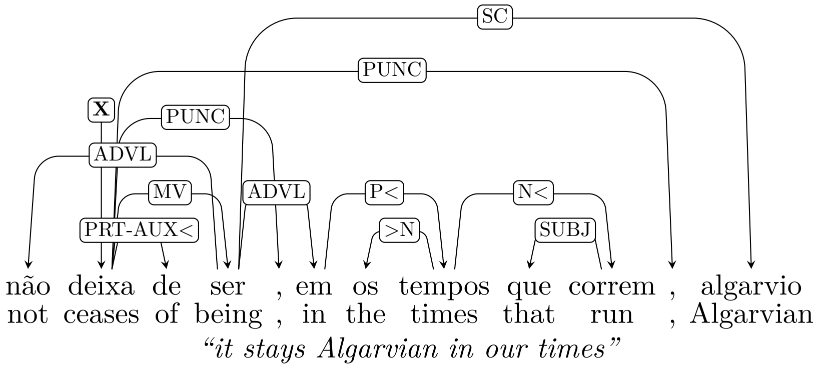
SC
PUNC
X
PUNC
ADVL
MV
ADVL
P<
N<
PRT-AUX<
>N
SUBJ
não deixa de ser , em os tempos que correm , algarvio
not ceases of being , in the times that run , Algarvian
"it stays Algarvian in our times"

Figure 24. Infinitive with preposition in [pt]: Preposition *(de)* is not attached between the phase verb *(deixa)* and the infinitive *(ser)*. The negative particle *(não)* is attached non-projectively to the non-head verb *(ser)*. Moreover, the commas around the parenthetical *(em os tempos que correm)* are also non-projective.

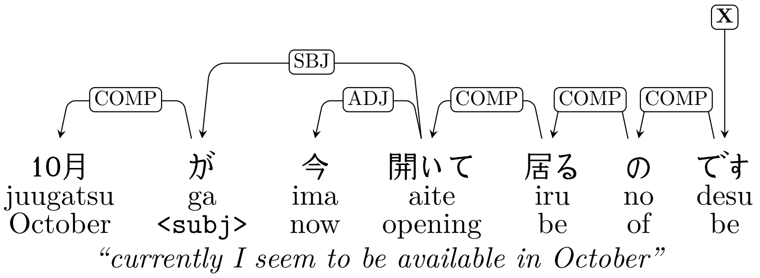
X
SBJ
COMP
ADJ
COMP
COMP
COMP
10月 が 今 開いて 居る の です
juugatsu ga ima aite iru no desu
October <subj> now opening be of be
"currently I seem to be available in October"

Figure 25. [ja] *Desu* is the polite copula. *Aite* is the conjunctive form of *aku* = "to open". The auxiliary *iru* with conjunctive of content verb together form the progressive tense. Japanese is an SOV language and left-branching structures are much preferred.

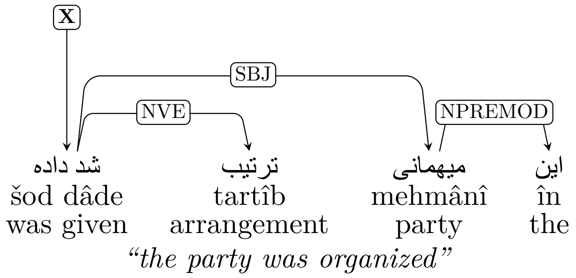
X
SBJ
NVE
NPREMOD
شد داده ترتیب میهمانی این
šod dâde tartîb mehmânî în
was given arrangement party the
"the party was organized"

Figure 26. [fa] Note that the dependency tree of the sentence *(În mehmânî tartîb šod dâde.)* is ordered right-to-left, the way Persian is written. The analytical passive *šod dâde* is represented by a single node (token).

Participles often govern auxiliaries [es, ca, it, ro, sl] (Figures 19 and 20); elsewhere the finite verb is the head [pt, de, nl, en, sv, ru]

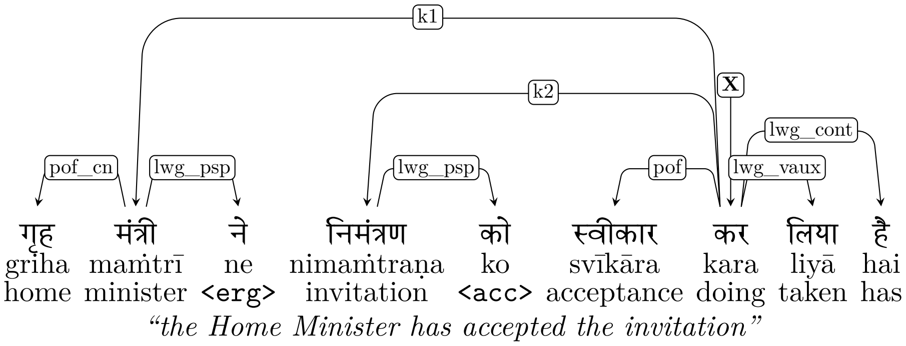
k1
k2
X
lwg_cont
pof_cn
lwg_psp
lwg_psp
pof
lwg_vaux
गृह मंत्री ने निमंत्रण को स्वीकार कर लिया है
griha maṁtrī ne nimaṁtraṇa ko svīkāra kara liyā hai
home minister <erg> invitation <acc> acceptance doing taken has
“the Home Minister has accepted the invitation”

Figure 27. [hi] *Kara* is a light verb stem, *svīkāra karanā* means “to accept”. *Liyā*, the perfect participle of *lenā* “to take”, is another light verb, specifying the direction of the result of the action. *Hai* is the auxiliary verb “to be” in finite form. Content verbs govern verbal groups in the [hi] treebank; as the main verb in this case is a compound verb *(svīkāra kara),* the head node of the two *(kara)* governs the whole group, even though the real content lies in the nominal element *(svīkāra).*

(Figures 17, 22 and 23), and finally, [cs] mixes both approaches based on semantic criteria. In [hi, ta], 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 (Figure 27).

The head typically holds the label describing the relation of the whole verbal group to its parent. As for child nodes, subjects and negative particles 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 (in [nl]) or the negative particle (in [pt]) can be attached to the non-head content element (Figure 22). Various infinitive-marking particles (English *“to”*, Swedish *“att”*, Bulgarian *“да”*) are usually treated similarly to subordinating conjunctions, i.e., they either govern the infinitive [en, da, bg] or are attached to it [de, sv]. In [pt], prepositions used between the main verb and the infinitive (*“estão a usufruir”*=*“are enjoying”*) are attached to the finite verb (Figure 24). In [bg], all modifiers of the verb including the subject are attached to the infinitive particle *“да”* instead of the verb below it (Figure 18).

We intend to unify verbal groups under a common approach, but the current version 1.5 of HamleDT does not do so yet. This part is more language-dependent than the others and a further analysis is needed.

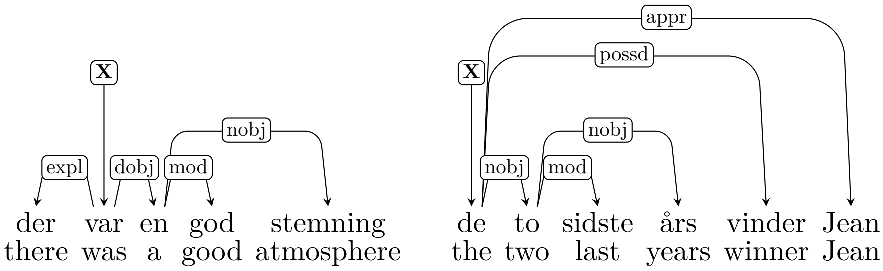
X
expl
dobj
mod
nobj
der var en god stemning
there was a good atmosphere
X
appr
possd
nobj
mod
nobj
de to sidste års vinder Jean
the two last years winner Jean

Figure 28. Two fragments from [da] show determiners and numerals governing noun phrases.

5.5. Determiner Heads

The Danish treebank is probably the most extraordinary one. Nouns often depend on determiners, numerals, etc. (see Figure 28). This approach is very rare in dependency treebanks, although it has its advocates among linguists (Hudson, 2004; Hudson, 2010).[10]

In HamleDT v1.5, we attach articles as well as other determiners to their nouns and numerals to the counted nouns.[11]

5.6. Punctuation

Table III presents an overview of punctuation treatment in the treebanks. Details and exceptions are discussed below. The *type codes* at paragraph beginnings refer to the columns of the table.

Pair/Pcom: Paired punctuation marks (quotation marks, brackets, parenthesizing commas *(Pcom)* or dashes) are typically attached to the head of the segment between them. Occasionally, they are attached one level higher, to the parent of the enclosed segment, or even higher, if the parent is member of a verbal group. Attaching punctuation to higher levels may break projectivity, as in Figure 24. The [pt] approach attaches paired punctuation to the parent of the interior segment (i.e. to the parent of the head of the segment, not to the head), unless the parent is the root or there are tokens outside the punctuation that depend on the head inside. In this latter case, the punctuation is attached to the inner head. In [tr], the *Pcom* column does not necessarily refer to *paired* punctuation; some commas are just attached to the root, which may result in non-projectivity.

[10] In Chomskian (constituency-based) approaches, it is the standard analysis that determiners function as the head of a noun phrase.

[11] Note however that numerals governing nouns are not restricted to [da]. Czech has a complex set of rules for numerals (motivated by the morphological agreement), which may result under some circumstances in the numeral serving as the head.

Rcom: Similarly, commas before and after a relative clause are typically attached either to the root of the relative clause (be it verb or conjunction) or to its parent. In [la], the clause is sometimes headed by a subordinating conjunction, but the comma is attached to the verb below. Note, however, that a comma terminating a clause may have multiple functions: it may at the same time delimit several nested clauses, a parenthetical phrase, and/or a conjunct.

In several languages, commas (in [fa]) or all punctuation symbols (in [eu, it, nl]) are systematically attached to neighboring tokens.

Coord: Commas, semicolons, or dashes can also substitute coordinating conjunctions, which is important especially if the Prague style of coordination is used (see Section 5.1). In [te], this is the sole function of commas (see Figure 29). In [da], which does not follow the Prague approach to coordination, we observed two adjectives modifying the same noun, separated by a comma; the comma was attached to the first "conjunct". We list the case in the *Coord* column although the structure was not formally tagged as coordination. In [hu], coordinating commas are normally attached to the parent of the coordination. Parents that are roots of the tree are an exception: in such cases, the comma is used as the head of the coordination.

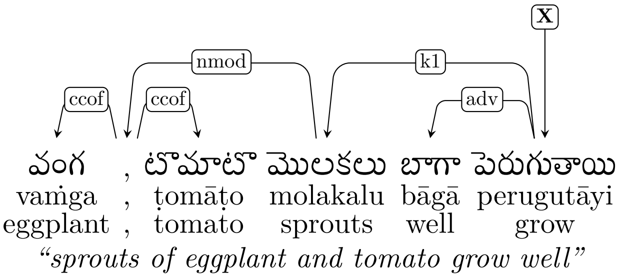
X
nmod
k1
ccof
ccof
adv
వంగ , టొమాటొ మొలకలు బాగా పెరుగుతాయి
vaṁga , ṭomāṭo molakalu bāgā perugutāyi
eggplant , tomato sprouts well grow
"sprouts of eggplant and tomato grow well"

Figure 29. Coordinating comma in [te].

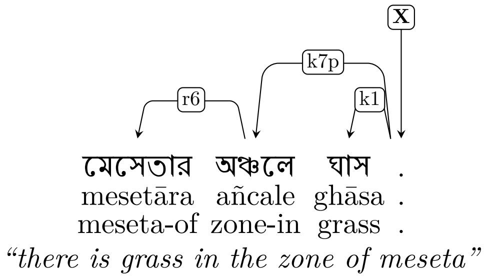
X
k7p
r6
k1
মেসেতার অঞ্চলে ঘাস .
mesetāra añcale ghāsa .
meseta-of zone-in grass .
"there is grass in the zone of meseta"

Figure 30. NULL-like usage of period in [bn]. The node with the period represents a dropped copula. Elsewhere in the treebank, such nodes are labeled by the pseudo-word-form "NULL".

Language	Fin	Pair	Pcom	Rcom	Coord	Coor1	Apos
Arabic (ar)	RN	SH	SH		HD	(PT)	
Basque (eu)	PT	PT	PT	PT		PT	PT
Bengali (bn)	(MP*)				HD		
Bulgarian (bg)	MP	SH	SH	SH	SH	SH	SH
Catalan (ca)	MP	SH	SH	SH	SH	SH	SH
Czech (cs)	RN	SH	SH	SH	HD	SH	HD
Danish (da)	MP	SH		SP/SH	PT*	SH	
Dutch (nl)	PW	PW	PW	PW		PW	PW
English (en)	MP	SH?	SP	SP	HD	SH	SP
Estonian (et)	MP	SH|SP	SP	SH|SP	HD	SH	SP
Finnish (fi)	MP	SH	SH	SH	SH	SH	
German (de)	MP	SH?	SP	SP	SH	PC	
Greek (el)	RN	SH			HD	SH	HD
Greek (grc)	RN	(SH)	SH	SH	HD	SH	HD
Hindi (hi)	MP	SH		(SP)		PC|PT	
Hungarian (hu)	MP	SH|SP	SH|SP	SP	HD|SP*	SP	
Italian (it)	PT	NT/PT	PT	PT		PT	PT
Japanese (ja)	MP*						
Latin (la)			SH	SH*	HD	SH	HD
Persian (fa)	MP	SH	PT	PT		PT	PT
Portuguese (pt)	MP	SP*	SP	SP	SH	SH	SP
Romanian (ro)	*no punctuation*						
Russian (ru)	*no punctuation*						
Slovene (sl)	RN	SH	SH	SH	HD	SH	HD
Spanish (es)	MP	SH	SH	SH	SH	SH	SH
Swedish (sv)	MP	NT/PT	SP	SP	PC	PC	SP
Tamil (ta)	RN	SH	SP	SP	HD	SH	
Telugu (te)	(MP)				HD		
Turkish (tr)	RR		RN*	CH	CH	CH	
HamleDT v1.5	RN	SH	SH	SH	HD	SH	SH

Table III.: Punctuation styles overview. RN = attached to the artificial root node; RR = attached to the artificial root and serving as root for the rest of the sentence, i.e., heading the main predicate; MP = attached to the main predicate; NT = attached to the next token; PT = attached to the previous token; PW = attached to the previous word (i.e., non-punctuation token); PC = attached to the previous conjunct; SH = attached to the head of the rel. clause / subtree inside paired punc. / coordination / second appos. member; SP = attached to the (grand)parent node of the rel. clause / subtree inside paired punc.; or to the first appos. member; CH = chain: attached to parent, and the head of the clause attached to the comma; for *Coord*, previous conjunct attached to comma, comma attached to next conjunct; HD = serving as head of coordination; *(X)* = rare in this treebank, based on very few observations; *X/Y* = initial *X*, final *Y*; *X*|*Y* = both observed; *X?* = unexplained exceptions observed; *X** = see text for more details; *empty cell* = not observed.

Coor1: Multi-conjunct coordination often involves one conjunction and one or more commas. Even within the same coordination family, multiple attachment schemes are possible for the commas (the previous conjunct, the head of the coordination, etc.) Additional commas are rare in [ar], where repeated conjunctions are more common.

Apos: Constructions in which two phrases describe the same object are called *appositions*. These are mostly but not solely noun phrases separated by a comma, dash, bracket, etc. as in *"Nicoletta Calzolari, the chief editor"*. Appositions are treated in the same way as parenthesis in most treebanks – the second phrase is attached to the first. Other treebanks regard appositions as coordinations – the punctuation serves as the head, with both phrases attached symmetrically.

Fin: Sentence-final punctuation (period, question mark, exclamation mark, three dots, semicolon, or colon) 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], or to the previous token [eu, it, ja, nl].[12] In [la, ro, ru], there is no final punctuation. It is also extremely rare in [bn, te]; however, there are a few punctuation nodes in [bn] that govern other nodes in the sentence. In fact, these nodes actually should have been labeled `NULL` to represent a copula or other constituents missing from the surface (Figure 30). Such `NULL` nodes appear elsewhere in [bn]. Punctuation is attached to the artificial root node in [tr] but instead of being a sibling of the main predicate, it governs the predicate. Note that some languages (e.g. Czech) may require final quotation marks (if present) to appear after the final period, but in [cs], it is not treated as final punctuation (unlike the period). Such quotation marks may end up attached non-projectively to the main verb.

A few treebanks [bg, cs, la, sl] use separate nodes for periods that mark abbreviations and ordinal numbers. These nodes are attached to the previous node (i.e., the abbreviation). In [cs], this rule has a higher priority even in cases where a period serves as an abbreviation marker and a sentence terminator at the same time. Most other treebanks are tokenized so that the period shares a node with the abbreviation (see also Section 5.7).

In HamleDT v1.5, we treat apposition as parenthesis, we attach paired punctuation to the root of the subtree inside and sentence-final punctuation to the artificial root node, mostly for consistency reasons. For the other punctuation types, a further analysis is needed.

[12] In [ja], the previous token essentially means the main predicate, but if it is followed by a question particle then the punctuation node is attached to the particle.

5.7. Tokenization and Sentence Segmentation

The only aspect that remains unchanged in HamleDT is tokenization and segmentation. Our harmonized trees always have the same number of nodes and sentences as the original annotation, despite some variability in the approaches we observe in the original treebanks.

Multi-word expressions and missing tokens
Some treebanks collapse multi-word expressions into single nodes [ca, da, es, eu, fa, hu, it, nl, pt, ro, ru]. Collapsing is restricted to personal names in [hu] and to named entities in [ro]. In [fa], it is used for analytical verb forms. The word form of the node is composed of all the participating words, joined by underscore characters or even by spaces [fa].

In [bn, te], dependencies are annotated between chunks instead of words (Figure 31). Therefore, one node may represent a whole noun phrase with modifiers and postpositions. The treebank only shows chunk headwords, which means we cannot reconstruct the original sentence. On a similar note, punctuation tokens have been deleted from two treebanks ([ro, ru]; see also Section 5.6).

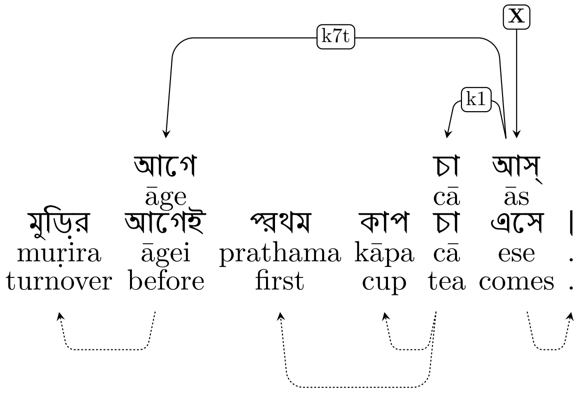

Figure 31. "The first cup of tea comes before the turnover." [bn] captures dependencies between chunks, not between tokens. Every sentence has been chunked and chunk headwords serve as nodes of the tree (their word forms are replaced by lemmas). The dotted dependencies below the sentence indicate which tokens belong to which chunk. Neither these dependencies nor the chunk-dependent words are visible in the treebank. The original sentences cannot be reconstructed from the trees.

Split tokens
On the other hand, orthographic words may be split into syntactically autonomous parts in some treebanks [ar, fa]. For instance, the Arabic

word وبالفلوجة (*wabiālfālūjah* = “and in al-Falujah”) is separated into *wa*/CONJ + *bi*/PREP + *AlfAlwjp*/NOUN_PROP. In [ta], the suffix *-um* indicating a coordination is treated as a separate token (see Section 5.1 and Figure 8).

Artificial nodes
Occasionally [bn, hi, te, ru], we see an inserted NULL node, which mostly stands for participants deleted on the surface, e.g., copulas [bn, ru] or conjuncts as in the Hindi example in Figure 32.

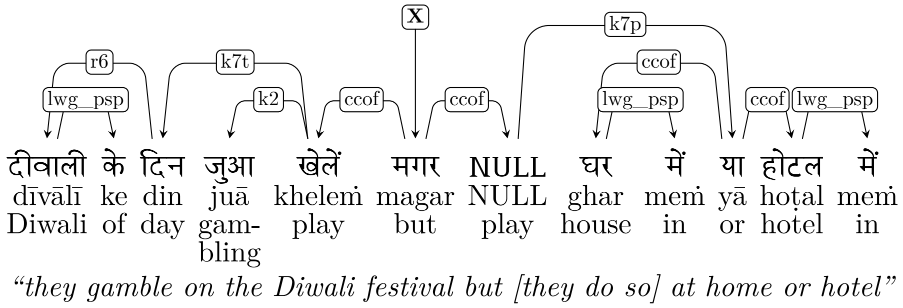
X
r6
k7t
k7p
ccof
lwg_psp
k2
ccof
ccof
lwg_psp
ccof
lwg_psp
दीवाली के दिन जुआ खेलें मगर NULL घर में या होटल में
dīvālī ke din juā khelem̐ magar NULL ghar mem̐ yā hoṭal mem̐
Diwali of day gambling play but play house in or hotel in
“they gamble on the Diwali festival but [they do so] at home or hotel”

Figure 32. A NULL node for a deleted verb (serving as head of conjunct) in [hi].

Along the same lines, some treebanks of pro-drop languages [ca, es] use empty nodes (with artificial word “_”) representing missing subjects, as in the following Spanish sentence: *“_ Afirmó que _ sigue el criterio europeo y que _ trata de incentivar el mercado donde no lo hay.”* = “He said he follows the European standard and encourages the market where there is none.” All the underscores mark subjects of the following verbs and could be translated as “he”.

Underscore/NULL nodes also appear in [tr], where they encode additional information related to morphological derivation.

Sentence segmentation
Similarly to tokenization, we also treat sentence segmentation as fixed, despite some less usual solutions: in [ar], sentence-level units are paragraphs rather than sentences, which explains the high average sentence length in Table I. In contrast, [ro] annotates every clause as a separate tree.

6. Obtaining HamleDT

Twelve harmonized treebanks from HamleDT v1.5 [ar, cs, da, fa, fi, grc, la, nl, pt, ro, sv, ta] are directly available for download from our web site:

http://ufal.mff.cuni.cz/hamledt

The license terms of the rest of the treebanks prevent us from redistributing them directly (in their original or normalized form), but most of them are easily acquirable for research purposes, under the links given in Appendix A). We provide the software that can be used to normalize and display the data after obtaining them from the original provider.

All the normalizations are implemented in Treex (formerly TectoMT) (Popel and Žabokrtský, 2010), a modular open-source framework for structured language processing, written in Perl.[13] In addition to normalization scripts for each treebank, Treex contains also other transformations, so for example, coordinations in any treebank can be converted from Prague to Stanford style.

The tree editor TrEd[14] can open Treex files and display original and normalized trees side-by-side on multiple platforms.

7. Conclusion

We provide a thorough analysis and discussion of varying annotation approaches to a number of syntactic phenomena, as they appear in publicly available treebanks, for many languages.

We propose a method for automatic normalization of the discussed annotation styles. The method applies transformation rules conditioned on the original structural annotation, dependency labels and morphosyntactic tags. We also propose unification of the tag sets for parts of speech, morphosyntactic features, and dependency relation labels. We take care to make the structural transformations and the morphosyntactic tagset unification as reversible as possible.[15]

We provide an implementation of the transformations in the Treex NLP framework. Treex can also be used for transforming the data to other annotation styles besides the one we propose (cf. Popel et al. (2013)). The resulting collection of harmonized treebanks, called HamleDT v1.5, is available to the research community according to the original licenses. A subset of the treebanks whose license terms permit redistribution is available directly from us. For the rest, users need to acquire the original data and apply our transformation tool.

Several future directions of our work are possible. Besides deepening the current level of harmonization (especially for verbal groups),

[13] http://ufal.mff.cuni.cz/treex/

[14] http://ufal.mff.cuni.cz/tred/ with EasyTreex extension

[15] We do not attempt at reversibility when unifying dependency relations.

we plan on adding new treebanks and languages, for which resources exist (e.g., French, Hebrew, Chinese, Icelandic, Ukrainian or Georgian). We also want to run parsing experiments and evaluate the various annotation styles from the point of view of learnability by parsers.

Acknowledgements

The authors wish to express their gratitude to all the creators and providers of the respective corpora.

The work on this project was supported by the Czech Science Foundation grant no. P406/11/1499 and no. P406/14/06548P, by the European Union Seventh Framework Programme under grant agreement FP7-ICT-2013-10-610516 (QTLeap), and by research resources of the Charles University in Prague (PRVOUK). This work has been using language resources developed and/or stored and/or distributed by the LINDAT/CLARIN project of the Ministry of Education of the Czech Republic (project LM2010013).

Finally, we are very grateful for the numerous valuable comments provided by the anonymous reviewers.

References

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

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

Atalay, N. B., K. Oflazer, B. Say, and I. Inst: 2003, 'The Annotation Process in the Turkish Treebank'. In: *In Proc. of the 4th Intern. Workshop on Linguistically Interpreteted Corpora (LINC)*.

Bamman, D. and G. Crane: 2011, 'The Ancient Greek and Latin Dependency Treebanks'. In: C. Sporleder, A. Bosch, and K. Zervanou (eds.): *Language Technology for Cultural Heritage*, Theory and Applications of Natural Language Processing. Springer Berlin Heidelberg, pp. 79–98.

Bengoetxea, K. and K. Gojenola: 2009, 'Exploring Treebank Transformations in Dependency Parsing'. In: *Proceedings of the International Conference RANLP-2009*. Borovets, Bulgaria, pp. 33–38, Association for Computational Linguistics.

Bharati, A., V. Chaitanya, and R. Sangal: 1994, *Natural Language Processing: A Paninian Perspective*. New Delhi: Prentice-Hall of India.

Bick, E., H. Uibo, and K. Müürisep: 2004, 'Arborest – a VISL-Style Treebank Derived from an Estonian Constraint Grammar Corpus'. In: *Proceedings of Treebanks and Linguistic Theories*.

Boguslavsky, I., S. Grigorieva, N. Grigoriev, L. Kreidlin, and N. Frid: 2000, 'Dependency treebank for Russian: Concept, tools, types of information'. In: *Proceedings of the 18th conference on Computational linguistics-Volume 2*. pp. 987–991.

Bosco, C., S. Montemagni, A. Mazzei, V. Lombardo, A. Lenci, L. Lesmo, G. Attardi, M. Simi, A. Lavelli, J. Hall, J. Nilsson, and J. Nivre: 2010, 'Comparing the Influence of Different Treebank Annotations on Dependency Parsing'.

Brants, S., S. Dipper, P. Eisenberg, S. Hansen, E. König, W. Lezius, C. Rohrer, G. Smith, and H. Uszkoreit: 2004, 'TIGER: Linguistic Interpretation of a German Corpus'. *Journal of Language and Computation* **2**(4), 597–620. Special Issue.

Buchholz, S. and E. Marsi: 2006, 'CoNLL-X shared task on multilingual dependency parsing'. In: *In Proc. of CoNLL*. pp. 149–164.

Civit, M., M. A. Martí, and N. Bufí: 2006, 'Cat3LB and Cast3LB: From Constituents to Dependencies.'. In: T. Salakoski, F. Ginter, S. Pyysalo, and T. Pahikkala (eds.): *FinTAL*, Vol. 4139 of *Lecture Notes in Computer Science*. pp. 141–152, Springer.

Csendes, D., J. Csirik, T. Gyimóthy, and A. Kocsor: 2005, 'The Szeged Treebank'. In: V. Matoušek, P. Mautner, and T. Pavelka (eds.): *TSD*, Vol. 3658 of *Lecture Notes in Computer Science*. pp. 123–131, Springer.

Călăcean, M.: 2008, 'Data-driven Dependency Parsing for Romanian'. Master's thesis, Uppsala University.

de Marneffe, M.-C. and C. D. Manning: 2008, 'Stanford typed dependencies manual'.

Džeroski, S., T. Erjavec, N. Ledinek, P. Pajas, Z. Žabokrtský, and A. Žele: 2006, 'Towards a Slovene Dependency Treebank'. In: *Proceedings of the Fifth International Language Resources and Evaluation Conference, LREC 2006*. Genova, Italy, pp. 1388–1391, European Language Resources Association (ELRA).

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

Hajič, J., M. Ciaramita, R. Johansson, D. Kawahara, M. A. Martí, L. Màrquez, A. Meyers, J. Nivre, S. Padó, J. Štěpánek, P. Straňák, M. Surdeanu, N. Xue, and Y. Zhang: 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, K., T. Viljanen, V. Laippala, S. Kohonen, F. Ginter, and T. Salakoski: 2010, 'Treebanking Finnish'. In: M. Dickinson, K. Müürisep, and M. Passarotti (eds.): *Proceedings of the Ninth International Workshop on Treebanks and Linguistic Theories (TLT9)*. pp. 79–90.

Hudson, R.: 2004, 'Are Determiners Heads?'. *Functions of Language* **11**(1).

Hudson, R.: 2010, *An Encyclopedia of Word Grammar and English Grammar*. London, UK: University College London, http://tinyurl.com/wg-encyc.

Husain, S., P. Mannem, B. Ambati, and P. Gadde: 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.

Hwa, R., P. Resnik, A. Weinberg, C. I. Cabezas, and O. Kolak: 2005, 'Bootstrapping parsers via syntactic projection across parallel texts.'. *Natural Language Engineering* **11**(3), 311–325.

Kawata, Y. and J. Bartels: 2000, 'Stylebook for the Japanese Treebank in Verbmobil'. In: *Report 240*. Tübingen, Germany.

Kromann, M. T., L. Mikkelsen, and S. K. Lynge: 2004, 'Danish Dependency Treebank'.

Marcus, M. P., B. Santorini, and M. A. Marcinkiewicz: 1993, 'Building a Large Annotated Corpus of English: The Penn Treebank'. *Computational Linguistics* **19**(2), 313–330.

Mareček, D. and Z. Žabokrtský: 2012, 'Exploiting Reducibility in Unsupervised Dependency Parsing'. In: *Proceedings of EMNLP-CoNLL'12*. pp. 297–307.

McDonald, R., J. Nivre, Y. Quirmbach-Brundage, Y. Goldberg, D. Das, K. Ganchev, K. Hall, S. Petrov, H. Zhang, O. Tčkström, C. Bedini, N. B. Castelló, and J. Lee: 2013, 'Universal Dependency Annotation for Multilingual Parsing'. In: *Proceedings of the ACL 2013*. Association for Computational Linguistics.

McDonald, R., S. Petrov, and K. Hall: 2011a, 'Multi-source Transfer of Delexicalized Dependency Parsers'. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA, pp. 62–72, Association for Computational Linguistics.

McDonald, R., S. Petrov, and K. Hall: 2011b, 'Multi-Source Transfer of Delexicalized Dependency Parsers'. In: *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, Scotland, UK., pp. 62–72, Association for Computational Linguistics.

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

Montemagni, S., F. Barsotti, M. Battista, N. Calzolari, O. Corazzari, A. Lenci, A. Zampolli, F. Fanciulli, M. Massetani, R. Raffaelli, R. Basili, M. T. Pazienza, D. Saracino, F. Zanzotto, N. Mana, F. Pianesi, and R. Delmonte: 2003, 'Building the Italian Syntactic-Semantic Treebank'. In: A. Abeillé (ed.): *Building and using Parsed Corpora*. Dordrecht, pp. 189–210, Kluwer.

Nilsson, J., J. Hall, and J. Nivre: 2005, 'MAMBA Meets TIGER: Reconstructing a Swedish Treebank from Antiquity'. In: *Proceedings of the NODALIDA Special Session on Treebanks*.

Nilsson, J., J. Nivre, and J. Hall: 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*. pp. 257–264.

Nivre, J., J. Hall, S. Kübler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret: 2007, '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)*.

Popel, M., D. Mareček, J. Štěpánek, D. Zeman, and Z. Žabokrtský: 2013, 'Coordination Structures in Dependency Treebanks'. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, pp. 517–527, Association for Computational Linguistics.

Popel, M. and Z. Žabokrtský: 2010, 'TectoMT: modular NLP framework'. *Advances in Natural Language Processing* pp. 293–304.

Prokopidis, P., E. Desipri, M. Koutsombogera, H. Papageorgiou, and S. Piperidis: 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)*. pp. 149–160.

Quirk, R., S. Greenbaum, G. Leech, and J. Svartvik: 1985, *A Comprehensive Grammar of the English Language*. London: Longman.

Ramasamy, L. and Z. Žabokrtský: 2012, 'Prague Dependency Style Treebank for Tamil'. In: *Proceedings of LREC 2012*. İstanbul, Turkey.

Rasooli, M. S., A. Moloodi, M. Kouhestani, and B. Minaei-Bidgoli: 2011, 'A Syntactic Valency Lexicon for Persian Verbs: The First Steps towards Persian

Dependency Treebank'. In: *5th Language & Technology Conference (LTC): Human Language Technologies as a Challenge for Computer Science and Linguistics.* Poznań, Poland, pp. 227–231.

Schwartz, R., O. Abend, and A. Rappoport: 2012, 'Learnability-based Syntactic Annotation Design'. In: *Proceedings of COLING 2012: Technical Papers.* Mumbai, India, pp. 2405–2422.

Seginer, Y.: 2007, 'Learning Syntactic Structure'. Ph.d. thesis, University of Amsterdam.

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

Smrž, O., V. Bielický, I. Kouřilová, J. Kráčmar, J. Hajič, and P. Zemánek: 2008, 'Prague Arabic Dependency Treebank: A Word on the Million Words'. In: *Proceedings of the Workshop on Arabic and Local Languages (LREC 2008).* Marrakech, Morocco, pp. 16–23, European Language Resources Association.

Surdeanu, M., R. Johansson, A. Meyers, L. Màrquez, and J. Nivre: 2008, 'The CoNLL-2008 Shared Task on Joint Parsing of Syntactic and Semantic Dependencies'. In: *Proceedings of CoNLL.*

Taulé, M., M. A. Martí, and M. Recasens: 2008, 'AnCora: Multilevel Annotated Corpora for Catalan and Spanish'. In: *LREC.* European Language Resources Association.

Tesnière, L.: 1959, *Éléments de syntaxe structurale.* Paris: Klincksieck.

Tsarfaty, R., J. Nivre, and E. Andersson: 2011, 'Evaluating Dependency Parsing: Robust and Heuristics-Free Cross-Annotation Evaluation'. In: *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing.* Edinburgh, Scotland, UK., pp. 385–396, Association for Computational Linguistics.

van der Beek, L., G. Bouma, J. Daciuk, T. Gaustad, R. Malouf, G. van Noord, R. Prins, and B. Villada: 2002, 'Chapter 5. The Alpino Dependency Treebank'. In: *Algorithms for Linguistic Processing NWO PIONIER Progress Report.* Groningen, The Netherlands.

Zeman, D.: 2008, 'Reusable Tagset Conversion Using Tagset Drivers'. In: N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, and D. Tapias (eds.): *Proceedings of the Sixth International Language Resources and Evaluation Conference, LREC 2008.* Marrakech, Morocco, pp. 28–30, European Language Resources Association (ELRA).

Zeman, D., D. Mareček, M. Popel, L. Ramasamy, J. Štěpánek, Z. Žabokrtský, and J. Hajič: 2012, 'HamleDT: To Parse or Not to Parse?'. In: N. Calzolari, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, J. Odijk, and S. Piperidis (eds.): *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12).* İstanbul, Turkey, European Language Resources Association (ELRA).

Appendix

A. List of included languages and treebanks

- Arabic [ar]: Prague Arabic Dependency Treebank 1.0 / CoNLL 2007 (Smrž et al., 2008)
  `http://padt-online.blogspot.com/2007/01/conll-shared-task-2007.html`
- Basque [eu]: Basque Dependency Treebank, a larger version than the one included in CoNLL 2007, generously provided by IXA Group (Aduriz et al., 2003)
  `http://hdl.handle.net/10230/17098`
- Bengali [bn], Hindi [hi] and Telugu [te]: Hyderabad Dependency Treebank / ICON 2010 (Husain et al., 2010)
  `http://ltrc.iiit.ac.in/icon/2010/nlptools/`
- Bulgarian [bg]: BulTreeBank (Simov and Osenova, 2005)
  `http://www.bultreebank.org/indexBTB.html`
- Catalan [ca] and Spanish [es]: AnCora (Taulé et al., 2008)
  `http://clic.ub.edu/corpus/en/ancora-descarregues`
- Czech [cs]: Prague Dependency Treebank 2.0 / CoNLL 2009 (Hajič et al., 2006)
  `http://ufal.mff.cuni.cz/pdt2.0/`
- Danish [da]: Danish Dependency Treebank / CoNLL 2006 (Kromann et al., 2004), now part of the Copenhagen Dependency Treebank
  `http://code.google.com/p/copenhagen-dependency-treebank/`
- Dutch [nl]: Alpino Treebank / CoNLL 2006 (van der Beek et al., 2002)
  `http://odur.let.rug.nl/~vannoord/trees/`
- English [en]: Penn TreeBank 3 / CoNLL 2007 (Marcus et al., 1993)
  `http://www.cis.upenn.edu/~treebank/`
- Estonian [et]: Eesti keele puudepank / Arborest (Bick et al., 2004)
  `http://www.cs.ut.ee/~kaili/Korpus/puud/`
- Finnish [fi]: Turku Dependency Treebank (Haverinen et al., 2010)
  `http://bionlp.utu.fi/fintreebank.html`
- German [de]: Tiger Treebank / CoNLL 2009 (Brants et al., 2004)
  `http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger.html`
- Greek (modern) [el]: Greek Dependency Treebank (Prokopidis et al., 2005)
  `http://gdt.ilsp.gr/`

- Greek (ancient) [grc] and Latin [la]: Ancient Greek and Latin Dependency Treebanks (Bamman and Crane, 2011)
  `http://nlp.perseus.tufts.edu/syntax/treebank/greek.html`,
  `http://nlp.perseus.tufts.edu/syntax/treebank/latin.html`
- Hindi [hi]: *see Bengali*
- Hungarian [hu]: Szeged Treebank (Csendes et al., 2005)
  `http://www.inf.u-szeged.hu/projectdirs/hlt/index_en.html`
- Italian [it]: Italian Syntactic-Semantic Treebank / CoNLL 2007 (Montemagni et al., 2003)
  `http://medialab.di.unipi.it/isst/`
- Japanese [ja]: Verbmobil (Kawata and Bartels, 2000)
  `http://www.sfs.uni-tuebingen.de/en/tuebajs.shtml`
- Latin [la]: *see Greek (ancient)*
- Persian [fa]: Persian Dependency Treebank (Rasooli et al., 2011)
  `http://dadegan.ir/en/persiandependencytreebank`
- Portuguese [pt]: Floresta sintá(c)tica (Afonso et al., 2002)
  `http://www.linguateca.pt/floresta/info_floresta_English.html`
- Romanian [ro]: Romanian Dependency Treebank (Călăcean, 2008)
  `http://www.phobos.ro/roric/texts/xml/`
- Russian [ru]: Syntagrus (Boguslavsky et al., 2000)
  `http://ruscorpora.ru/en/`
- Slovene [sl]: Slovene Dependency Treebank / CoNLL 2006 (Džeroski et al., 2006)
  `http://nl.ijs.si/sdt/`
- Spanish [es]: *see Catalan*
- Swedish [sv]: Talbanken05 (Nilsson et al., 2005)
  `http://www.msi.vxu.se/users/nivre/research/Talbanken05.html`
- Tamil [ta]: TamilTB (Ramasamy and Žabokrtský, 2012)
  `http://ufal.mff.cuni.cz/~ramasamy/tamiltb/0.1/`
- Telugu [te]: *see Bengali*
- Turkish [tr]: METU-Sabanci Turkish Treebank (Atalay et al., 2003)
  `http://ii.metu.edu.tr/corpus/`

B. Examples of harmonization of dependency relations

orig. label	tokens	distribution of HamleDT v1.5 labels
k1	1,168	Sb=98% Coord=2%
main	1,130	Pred=85% Coord=14%
r6	790	Atr=98% Coord=2%
k2	788	Obj=95% Coord=5%
ccof	602	Pred=40% Atr=23% Obj=12% Coord=8% Adv=8% Sb=7% Pnom=2%
vmod	583	Adv=98% Coord=2%
pof	421	Obj=100%
k7p	325	Adv=98% Coord=2%
k7t	303	Adv=100%
nmod	233	Atr=91% Coord=9%
k1s	202	Pnom=98% Coord=2%
k7	152	Adv=97% Coord=3%
other	470	Atr=41% Adv=37% Obj=11% Coord=4% Sb=4% Atv=2% *rest*<0.5%

Table IV.: The Bengali treebank [bn] uses 42 dependency labels, but we show only 12 most frequent ones. The remaining 30 labels are summarized in the last line. Bengali dependency labels are explained in `http://ltrc.iiit.ac.in/nlptools2010/files/documents/dep-tagset.pdf` and their mapping to HamleDT v1.5 dependency labels is relatively straightforward, except for coordinations, where the Bengali treebank marks the conjunction with the dependency relation, while in HamleDT v1.5, the conjuncts are marked with the dependency relation and the conjunction is marked with *Coord*). For example, k7p is *location in space*, k7t *location in time* and k7 *location elsewhere*; all three labels are basically mapped to *Adv* (adverbial).

orig. label	tokens	distribution of HamleDT v1.5 labels
NMOD	155,951	Atr=62% AuxA=22% AuxP=13% Coord=1% AuxV=1% *rest*=1%
P	52,051	AuxX=44% AuxK=36% AuxG=20% *rest*<0.5%
PMOD	45,207	Adv=50% Atr=40% Coord=6% AuxP=2% NR=1% Obj=1% *rest*<0.5%
SBJ	35,446	Sb=94% Coord=2% NR=1% Atr=1% Obj=1% Adv=1% *rest*<0.5%
ADV	32,202	AuxP=56% Adv=30% AuxC=5% NR=3% Atr=2% AuxV=2% Obj=1% Coord=1% *rest*<0.5%
OBJ	30,507	Obj=55% Adv=29% AuxV=8% Coord=5% Atr=2% *rest*=1%
COORD	22,865	Atr=34% Adv=21% Obj=12% Pred=11% Sb=8% AuxV=5% Pnom=3% AuxP=2% NR=2% Coord=1% AuxA=1% *rest*=1%
VMOD	21,053	AuxV=30% AuxC=24% Pnom=20% Neg=10% Adv=6% Atr=3% Coord=2% Obj=2% NR=2% AuxP=1% Sb=1% *rest*<0.5%
ROOT	18,791	Pred=69% AuxV=18% Coord=8% ExD=4% *rest*<0.5%
AMOD	15,269	Atr=52% Adv=19% AuxP=14% NR=9% AuxC=3% AuxV=1% *rest*=1%
VC	13,745	Pred=29% Adv=25% Obj=23% AuxV=10% Atr=9% Coord=2% NR=1% *rest*<0.5%
IOBJ	1,883	Obj=92% Adv=3% Coord=2% Atr=1% Sb=1% *rest*<0.5%
CC	1,336	NR=98% Neg=2%
PRT	1,268	AuxV=95% AuxC=2% Adv=2% *rest*<0.5%
PRN	1,259	Atr=43% Adv=27% AuxP=8% NR=7% Obj=6% Coord=5% AuxV=1% AuxC=1% *rest*<0.5%
LGS	1,211	AuxP=99% AuxC=1% *rest*<0.5%
DEP	892	AuxP=46% Atr=23% Adv=10% NR=9% Neg=4% AuxC=3% AuxA=2% Coord=1% *rest*=1%
GAP	272	Atr=47% AuxP=38% Adv=10% NR=2% Coord=1% AuxC=1% Neg=1%
EXP	219	Adv=84% AuxV=11% Coord=5%
TMP	149	Atr=97% NR=3%

Table V.: The English treebank [en] (from CoNLL 2007) uses 20 dependency labels, but their mapping to HamleDT v1.5 labels is not straightforward. In practice, we found the English CoNLL 2007 labels not helpful, and we based the conversion only on dependency structure and morphological tags.

C. List of dependency relation labels in figures

Language	Label	Description	Example
	X	Our meta-label that represents the unknown relation of the depicted subtree to its unshown parent.	
bg	comp	Complement, i.e. argument of non-verbal head, non-finite verbal head, copula.	Figure 18
bg	indobj	Child is indirect object of parent.	Figure 18
bg	mod	Child is modifier, e.g. of a noun phrase, or a negative particle modifying a verb etc.	Figure 18
bg	prepcomp	Child is noun phrase, parent is preposition.	Figure 18
bg	subj	Child is subject of parent.	Figure 18
bg	xcomp	Child is clausal complement; this includes complements of modal verbs.	Figure 18
ca	CO	Child is coordinating conjunction, parent is the first conjunct.	Figure 4
ca	CONJUNCT	Parent is the first conjunct, child is one of the other conjuncts.	Figure 4
ca	PUNC	Child is punctuation symbol.	Figure 4
cs, sl, la, ta	Adv	Child is adverbial modifier of parent.	Figure 2
cs, sl, la, ta	Atr	Parent is noun, child is its attribute.	Figure 9
cs, sl, la, ta	AuxC	Child is subordinating conjunction, parent is governing predicate. The relation of the subordinate clause to the parent is labeled at the grandchild.	Figure 19
cs, sl, la, ta	AuxP	Child is preposition. The relation of the prepositional phrase to the parent is labeled at the grandchild.	Figure 2
cs, sl, la, ta	AuxV	Child is auxiliary verb or negative particle, parent is content verb.	Figure 19
cs, sl, la, ta	AuxX	Child is comma and does not serve as coordination root.	Figure 2
cs, sl, la, ta	AuxZ	Emphasizing word.	Figure 8
cs, sl, la, ta	Coord	Child serves as root of a coordinate structure.	Figure 1
cs, sl, la, ta	Obj	Child is object of parent.	Figure 2
cs, sl, la, ta	Pred	Child is predicate of a main clause.	Figure 2
cs, sl, la, ta	Sb	Child is subject of parent.	Figure 19
cs, ta	_M	Suffix to a label, saying that the child is a conjunct. The main label tags its relation to the parent of the coordinate structure.	Figure 1
da	appr	Restrictive apposition (no comma).	Figure 28
da	conj	Child is conjunct, parent is first conjunct or coordinating conjunction.	Figure 6
da	coord	Parent is conjunct, child is coordinating conjunction.	Figure 6
da	dobj	Child is direct object of parent.	Figure 28

da	expl	Child is expletive subject of parent.	Figure 28
da	mod	Modifier, e.g. attribute of noun, adverbial modifier of verb, adjective attached to determiner etc.	Figure 28
da	nobj	Child is noun phrase or infinitive, parent is e.g. determiner, numeral, preposition etc.	Figure 28
da	pnct	Child is punctuation symbol.	Figure 6
da	possd	Child is argument of possessive parent, i.e. child is the thing possessed.	Figure 28
de	CD	Child is coordinating conjunction, parent is one conjunct and right sibling is the other conjunct.	Figure 3
de	CJ	Parent and child are conjuncts.	Figure 3
de	MO	Modifier. In NPs only focus particles are annotated as modifiers.	Figure 23
de	NG	Child is negative particle, parent is negated verb.	Figure 23
de	NK	Noun Kernel. Child attached within a noun phrase or a prepositional phrase.	Figure 10
de	OA	Child is accusative object of parent.	Figure 23
de	OC	Clausal object. Also verb tokens building a complex verbal form and modal constructions.	Figure 23
de	PUNC	Child is punctuation symbol.	Figure 3
de	SB	Child is subject of parent.	Figure 23
es	atr	Attribute. E.g. child is adverbial/prepositional phrase, parent is verb.	Figure 12
es	cd	Child is direct object of parent.	Figure 12
es	conj	Child is subordinating conjunction.	Figure 12
es	s.a	Child is adjectival phrase, parent is not verb.	Figure 12
es	sn	Child is noun phrase. Parent may be e.g. preposition.	Figure 12
es	spec	Specifier. E.g. child is determiner and parent is noun.	Figure 12
es	suj	Child is subject of parent.	Figure 12
fa	NPREMOD	Child is premodifier of parent noun.	Figure 26
fa	NVE	Child is non-verbal element of compound verb. Parent is verbal element.	Figure 26
fa	SBJ	Child is subject of parent.	Figure 26
hi	lwg_cont	Child is additional node of a complex expression; child and parent together perform certain function.	Figure 27
hi	lwg_psp	Child is postposition and modifies a noun.	Figure 11
hi	lwg_vaux	Child is auxiliary verb, parent is content verb.	Figure 27
hi	pof	Part of relation, e.g. part of conjunct verb.	Figure 27
hi	pof_cn	Part of relation.	Figure 27
hi, bn, te	adv	Child is adverbial modifier (only adverbs of manner) of parent.	Figure 29

hi, bn, te	ccof	Child is conjunct, parent is coordinating conjunction or comma.	Figure 29
hi, bn, te	k1	Child is karta (doer / agent / subject) of parent predicate.	Figure 27
hi, bn, te	k2	Child is karma (pacient / object) of parent predicate.	Figure 27
hi, bn, te	k7p	Child is deshadhikarana (location in space) of the parent predicate.	Figure 30
hi, bn, te	k7t	Child is kaalaadhikarana (location in time) of the parent predicate.	Figure 31
hi, bn, te	nmod	Parent is noun, child is its attribute.	Figure 29
hi, bn, te	nmod_adj	Child is adjective and modifies a noun.	Figure 11
hi, bn, te	r6	Shashthi (possessive). Child is possessor in genitive, parent is the possessed noun.	Figure 30
hu	ATT	Attribute.	Figure 15
hu	CONJ	Child is conjunction (coordinating or subordinating).	Figure 5
hu	DET	Child is determiner, parent is noun.	Figure 15
hu	ILL	Child is verbal argument in illative case.	Figure 15
hu	OBJ	Child is object of parent.	Figure 15
hu	PUNCT	Child is punctuation symbol.	Figure 5
hu	SUBJ	Child is subject of parent.	Figure 15
it	cong_sub	Parent is subordinating conjunction.	Figure 13
it	det	Child is determiner, parent is noun.	Figure 13
it	modal	Child is modal (dovere, volere, potere) or aspectual (andare, venire, stare) verb, parent is content verb.	Figure 13
it	pred	Parent is verb (often it is copula), child is predicative complement (nominal predicate).	Figure 13
it	sogg	Child is subject of parent.	Figure 13
ja	ADJ	Child is adjunct of parent.	Figure 25
ja	COMP	Complement, e.g. verb attached to another verb form, noun attached to postposition etc.	Figure 25
ja	SBJ	Child is subject of parent.	Figure 25
nl	det	Child is determiner, parent is noun.	Figure 21
nl	mod	Child is adverbial modifier (bijwoordelijke bepaling) of parent.	Figure 21
nl	obj1	Child is direct object; this includes nouns attached to prepositions!	Figure 21
nl	predm	Child determines state (adverbial modifier), parent is predicate.	Figure 22
nl	su	Child is subject of parent.	Figure 21
nl	vc	Verbal complement. Example: parent is modal, child is infinitive.	Figure 21
pt	>N	Child is left dependent of nominal core.	Figure 24
pt	ADVL	Child is adverbial adjunct (adjunto adverbial) of parent.	Figure 24

pt	MV	Child is main verb, parent may be e.g. modal verb.	Figure 24
pt	N<	Child is right dependent of nominal core.	Figure 24
pt	P<	Child is right dependent of preposition.	Figure 24
pt	PRT-AUX<	Child is verbal particle (partícula de ligação verbal), e.g. between modal and content verb, parent would be modal.	Figure 24
pt	PUNC	Child is punctuation symbol.	Figure 24
pt	SC	Child is nominal predicate (predicativo do sujeito), parent is copula.	Figure 24
pt	SUBJ	Child is subject of parent.	Figure 24
ro	rel.conj.	Parent is coordinating conjunction, child is conjunct.	Figure 7
ru	1-компл	Child is argument other than subject. Also: genitive noun modifier of another noun.	Figure 17
ru	агент	Child is agent-object of passive parent.	Figure 17
ru	опред	Parent is noun, child is its attribute.	Figure 17
ru	пасс-анал	Child is passive participle, parent is finite auxiliary verb.	Figure 17
ru	предик	Parent is predicate, child is subject.	Figure 17
ta	AComp	Child is (obligatory) adverbial complement of parent.	Figure 8
tr	OBJECT	Child is object of parent.	Figure 16
tr	QUESTION .PARTICLE	Child is question particle, parent is verb.	Figure 16
tr	SUBJECT	Child is subject of parent.	Figure 16
tr	VOCATIVE	Child is vocative noun phrase serving as doer (actor) of parent verb.	Figure 16