# Universal Dependencies for Danish

Anders Johannsen Héctor Martínez Alonso Barbara Plank
Center for Language Technology, University of Copenhagen, Denmark
ajohannsen@hum.ku.dk,alonso@hum.ku.dk,bplank@cst.dk

**Abstract**

The Universal Dependencies (UD) project aims at developing treebank annotations consistent across many languages. In this paper, we present the conversion of the Copenhagen Dependency Treebank (CDT) into Universal Dependencies (UD). We describe the original CDT annotation and detail the mapping into the new UD formalism, which we accomplish by taking a *test-driven* approach. We present parsing experiments with both formalisms. Additionally, we quantitatively compare the resulting Danish UD treebank to the other languages available in the UD project (v1.2), discussing constructions that are specific to Danish. Our results show that the newly created Danish UD treebank is closely related to treebanks of typologically similar languages. However, parsing with the new treebank becomes more difficult, relative to the old formalism.

## 1 Introduction

The Universal Dependencies (UD) project[1] [15] is an on-going research effort that aims to facilitate multilingual and cross-lingual language technology. The UD project develops a dependency formalism that maximizes parallelism between languages, while allowing for language-specific extensions. In UD, content words are first-class citizens, and syntactic analyses that directly connect content words are preferred, whenever possible. The treebank annotation scheme grew out of three research related projects, namely the Stanford dependencies [7], the Google universal Part-of-Speech tag set [16] and the Interset interlingua for morphology [20]. The latest version of UD, v1.2, released November 2015, contains treebanks for 33 languages [14].

We introduce UD in Section 2.1, followed by a description of the somewhat atypical original annotation of the Copenhagen Dependency Treebank (CDT). Our conversion steps are described in Section 3. In Section 4 we assess the learnability of automatic parsers from the converted treebank and provide a quantitative evaluation of the resulting treebank when compared to the other UD languages.

[1] http://universaldependencies.github.io/docs/

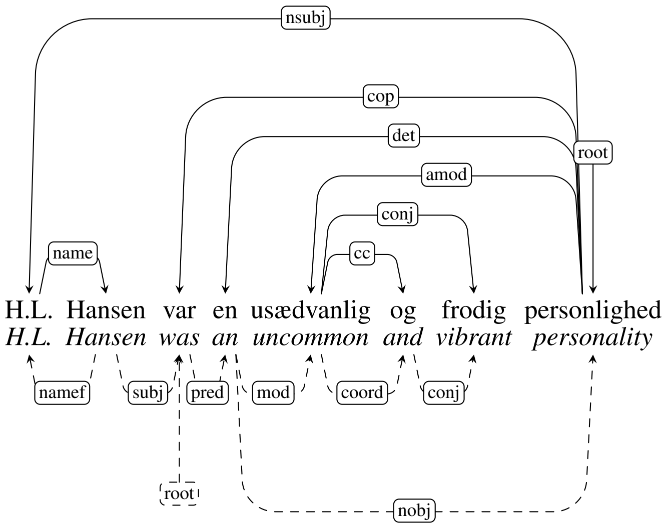

nsubj
cop
det
root
amod
conj
cc
name
H.L. Hansen var en usædvanlig og frodig personlighed
H.L. Hansen was an uncommon and vibrant personality
namef
subj
pred
mod
coord
conj
root
nobj

Figure 1: Dependency tree example. Above: UD, below: CDT (dashed).

# 2 Differences between UD and CDT

In this section we provide a brief overview on the principles of the UD formalism. For further details on syntactic and morphological annotation, we refer to [6, 13]. We then describe the design choices of CDT that are characteristically different from UD.

## 2.1 Universal Dependencies

The UD formalism has, roughly speaking, three driving principles:

1. **Content over function**: Content words are the heads of function words, e.g. lexical verbs are the head of periphrastic verb constructions, nouns are the heads of prepositional phrases, and attributes are the head of copula constructions.
2. **Head-first**: In spans where it is not immediately clear which element is the head (the content-over-function rule does not apply straightforwardly), UD takes a head-first approach: the first element in the span becomes the head, and the rest of the span elements attach to it. This applies mostly to coordinations, multiword expressions, and proper names.
3. **Single root attachment**: Each dependency tree has exactly one token directly dominated by the artificial root node. Other candidates for direct root attachment are instead attached to this root-dominated token.

An illustrative example is shown in Figure 2.1. Here, the copula is headed by the attribute *personlighed* (content over function). The proper name span *H.L. Hansen* has the first element as head (head-first) and the tree has a single node dominated by the root. Apart from these three principles, UD imposes no further projectivity constraints, other than punctuation attachment must preserve projectivity. Further examples of annotation differences are given in the appendix (Figure 5).

UD uses a common set of 17 POS tags [13] and 40 syntactic relations [6].

## 2.2 Copenhagen Dependency Treebank

The Copenhagen Dependency Treebank (CDT) [11] consists of 5,512 sentences (about 100k tokens). The Danish source texts were collected and part-of-speech annotated by the PAROLE-DK project [10]. Although the CDT annotation scheme is very rich, it departs from all three UD principles listed in the previous section.

Perhaps the most salient difference is that CDT has determiners as heads. This is illustrated in Figure 2.1 (dashed), where the determiner *en* is the head of the noun *personlighed*. This analysis is known as Determiner Phrase (DP) analysis. While common in generative grammar frameworks [1, 9, 8, 17], it is very rarely implemented in dependency-based treebanks. To the best of our knowledge, the only other dependency treebank that uses DP analysis besides CDT is the Turin University Treebank [3].

In contrast to UD, dependencies in CDT follow a chain structure, resulting in trees with more levels. For instance, periphrastic verb constructions ("jeg *ville have kunnet* købe", "I *would have been able to* buy") in CDT are headed by the first auxiliary, and each following verb depends on the previous one. In our example in Figure 2.1, the copula is verb-headed. Coordinations are headed by the first conjunct, but the second conjunct is a dependent of the conjunction (cf. *frodig* depends on the conjunction *og*). This coordination structure deviates from the second UD principle, which specifies that all elements in a span attach to the first element. Finally, the CDT treebank contains several multi-rooted trees.

Moreover, CDT has no special part-of-speech tags for determiners. Many Danish determiners come with a homographic pronoun (e.g. '*min* jakke er **min**', '*my* jacket is **mine**'), and CDT provides the same tag, interpreted as a pronoun, for all forms. Thereby, the determiner-pronoun distinction is not recoverable at the part-of-speech level. Figure 2 shows three examples of noun phrase annotation with different specifiers from CDT. Like in English, Danish possessive constructions such 'Anna's parents' show complementary distributions with possessive determiners and receive the same dependency analysis, namely as heads of their following noun (Example b). The third example is a noun phrase complementing a pronoun, but it has the same structure as the second. Nevertheless, *ham* in Example c) is a case-marked pronoun that has no homographic determiner and will not be interpreted as determiner during the conversion process, unlike the determiner *de* in Example b) or *den* in Example c).

Similarly, CDT tags for verbs do not distinguish between lexical verbs and functional verbs, such as modals and auxiliaries. We apply tree-structure heuristics to disambiguate between verb-auxiliary and pronoun-determiner (cf. Section 3).

Our starting point for CDT is the data distributed in the CoNLL 2006 multilingual-parsing shared task [4]. We keep the same test set (322 sentences), but add a fixed

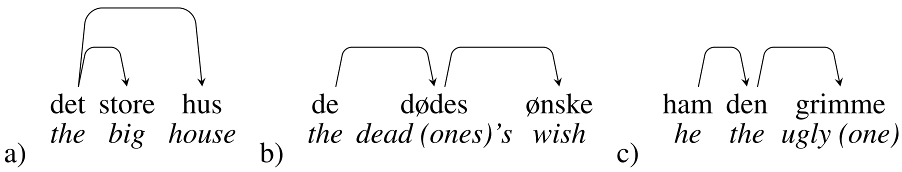


Figure 2: CDT treatment of different noun specifiers.

development set of the same size by randomly sampling from the training set. All three data sets are disjoint. Prior to conversion, we added some missing lemmas and manually enforced single-rootness. The treebank consists of 100,777 tokens in 5,512 sentences with an average sentence length of 18.3 tokens.

# 3 Conversion

Inspired by best practices in software development, we take a *test-driven* approach to the tree-bank conversion. The most direct application of this methodology would be to establish a set of reference annotations and compare them to the result of running the conversion procedure. If they are not equal, the test "fails". However, this binary setup is not sufficient for larger conversions which are the result of chained application of many smaller conversion procedures or steps: the individual steps should be tested as well.

After each step we therefore automatically calculate a series of quality measures with respect to the reference set, such as labeled and unlabeled attachment score (LAS/UAS), tree-consistency (weak connectedness, single-rootness), and other indicators like number of non-projective edges, and average number of transformed edges.

**Test bench** The reference set contains 28 sentences, randomly sampled from the CDT treebank and annotated manually from scratch using the UD guidelines. The reference set has a *goal* subset of 17 sentences, where we expect the conversion process to obtain a LAS of 100% (test passed). These sentences exhibit all the basic syntactic phenomena addressed by the conversion steps listed in Table 1. The remaining 11 sentences in the reference set contain more rare phenomena like fragments of compounds, coordinated applications of several prepositions to the same noun ('*on and under* a tree'), or clausal complement labels like *csubj*. We use the whole reference set to measure the overall quality of the conversion and to provide directions for future improvements. The purpose of the goal subset is to test that the current conversion works predictably. If in the future we decide that a syntactic phenomenon like coordinated prepositions should fall within the scope of the conversion, we simply move sentences with this phenomenon to the goal subset. The resulting LAS for the goal subset is 100%, whereas for the overall reference set, the converted treebank scores 86.44% LAS and 89.54% UAS.

The conversion tool is implemented in Python as a sequence of rewrite operations on a graph structure. The conversion framework, which is treebankindependent, is available for download.[2] Table 1 shows the conversion steps and their score on the reference set. The following section describes the rewrite operations.

**Rewrite operations** The treebank conversion is the result of 18 sequentially applied rewrite operations. These fall into four broad categories. We first apply global operations involving conjunctions, then do local operations involving nouns, followed by operations that are verb-centric. During the conversion, we use part-of-speech information from the CDT. As one of the last steps, “Map POS and feats”, features and dependency relations are mapped into UD labels. Finally, certain multi-word units (MWU), which in CDT appear as one token (e.g. ‘i dag’, lit. ‘in day", ‘today’), are split into their component tokens.

1. **Conjunction-centric (C)** The first group of operations involves flattening coordinating conjunction chains and applying the head-first principle.
2. **Noun-centric (N)** This group of operations rewrites DP analyses, making nouns heads. In particular, it implements rewriting of proper names, switching the headedness of determiners and possessives, as well as making adpositions case-markers of the content noun. We disambiguate determiners and pronouns according to their form and dependency relations. Specifically, an ambiguous pronoun becomes a determiner if it introduces a noun.
3. **Verb-centric (V)** The rewrite operations for verbs mainly involves flattening verb chains and making them content-headed, identifying content heads for copula constructions and making adpositions clause markers for their introduced verbs. We disambiguate verbs in auxiliaries and content verbs according to their form and dependency relations, namely by determining whether a verb belongs to the closed class of functional verbs and it introduces a lexical verb.
4. **Label-centric (L)** This group contains mappings and heuristics for relabeling of POS, morphological features and dependency relations. Most POS and features are obtained from Interset [20], while other traits (determiner, auxiliary, copula) are calculated from edge properties. The Danish UD dependency relation inventory comprises the standard UD inventory, plus three language-specific labels, namely *nmod:loc*, *nmod:tmod* and *nmod:poss*.

We observe how UAS and LAS increase monotonically from the first step to the last. We cannot say the same about projectivity, because the average non-projectivity increases after e.g. reattaching conjunctions and copulas. The last step involves re-tokenization and can only be applied when the rest of the tree structure has been reassigned. Therefore no scores are provided for it in Table 1. The operations that have a larger impact in terms of how many edges are reattached are “switch article head” and “switch preposition head”, which reattach determiners and prepositions respectively. Moving determiners and prepositions from func-

[2] https://github.com/andersjo/ud-test-driven-conversion

|  |  |  | Changes per sent. |  | Scores |  |
|---|---|---|---|---|---|---|
| Conversion step |  | Non-projectivity | Labeled | Unlabeled | UAS | LAS |
| – | Identity transform | 0.35 | 0.00 | 0.00 | 30.45 | 5.30 |
| – | Preprocess | 0.35 | 0.00 | 0.00 | 30.45 | 5.30 |
| C | Discourse conjunctions | 0.35 | 0.18 | 0.18 | 32.41 | 6.61 |
| C | Switch sconj headness | 0.35 | 0.00 | 0.00 | 32.41 | 6.61 |
| C | Modify conjunctions | 0.41 | 0.71 | 0.35 | 33.56 | 8.90 |
| C | Switch clause relating element head | 0.41 | 0.12 | 0.12 | 33.98 | 8.90 |
| N | Proper names head first | 0.41 | 0.59 | 0.59 | 37.72 | 12.19 |
| N | Switch possessive head | 0.41 | 0.59 | 0.59 | 39.27 | 13.42 |
| N | Switch article head | 0.41 | 2.88 | 2.88 | 58.15 | 28.57 |
| V | Switch particle head | 0.41 | 0.00 | 0.00 | 58.15 | 28.57 |
| N | Switch preposition head | 0.41 | 3.06 | 3.06 | 82.82 | 51.29 |
| V | Infinite verb chains | 0.41 | 0.00 | 0.00 | 82.82 | 51.29 |
| V | Verb chains to content head | 0.12 | 1.06 | 1.06 | 89.42 | 53.50 |
| V | Copula to content head | 0.29 | 1.12 | 1.12 | 98.17 | 56.25 |
| – | Punct and subordination | 0.29 | 0.29 | 0.29 | 100.00 | 56.25 |
| L | Map POS and feats | 0.29 | 0.00 | 0.00 | 100.00 | 56.25 |
| L | Map deprels | 0.29 | 6.06 | 0.00 | 100.00 | 100.00 |
| – | Prepositions as leaves | 0.29 | 0.00 | 0.00 | 100.00 | 100.00 |
| – | Split multi-word units | – | – | – | – | – |

Table 1: Conversion statistics on the *goal* reference annotations. Lab and unlab Δ: mean number of labeled or unlabeled changes.

tional heads to leaves in the tree has a large impact on the overall structure of the trees, because their dependents must also be reattached.

## 4 Evaluation

**Parsing** We train two state-of-the-art graph-based dependency parsers, MST (2nd order, non-proj) and Mate [12, 2] on the original (CDT) and converted UD-Danish data. The results in Table 2 show a 4-5% accuracy drop when parsing UD-Danish with standard features. This is not surprising, as CDT and UD-Danish are now quite different treebanks. In fact, the attachment of 65% of the edges changed during the conversion. In contrast to our results, Pyysalo et al. [18] observed only a minor drop in performance (0.5%) on Finnish. However, their original annotation is based on Stanford dependencies and is thus closer to UD than CDT.

The MST parser's performance drop on labeled accuracy (compared unlabeled accuracy) is remarkable. Both parsers are second-order, but Mate has more context features. To get an intuition about the difficulty of predicting dependency labels for CDT and UD, respectively, we train a simple linear-chain CRF model which for each token outputs the label of its head relation. Only the word itself and the universal part-of-speech tag is used as input. The model obtains slightly higher accuracies for predicting UD labels (88.21% vs 87.97% on the test set). So, in the absence of structural information, there seems to be little difference in the predictability of labels in UD and CDT.

**Similarity with other UD treebank** We estimate the similarity with other UD treebanksby comparing several distributions, i.e. distributions over labels, over

|  | Mate |  |  | MST |  |  |
|---|---|---|---|---|---|---|
|  | LAS | UAS | LA | LAS | UAS | LA |
| CDT DEV | 85.20 | 89.38 | 90.83 | 84.59 | 89.46 | 90.61 |
| CDT TEST | 84.38 | 88.70 | 90.17 | 84.11 | 89.44 | 90.69 |
| UD-DANISH DEV | 81.87 | 84.51 | 92.10 | 65.87 | 81.57 | 75.71 |
| UD-DANISH TEST | 81.56 | 84.64 | 92.00 | 63.87 | 80.91 | 74.54 |

Table 2: Parsing accuracy including punctuation.

POS, and over labeled head-dependent head triples. We compare Danish to three sets of languages; **Scandinavian** (no,sv), **Germanic** languages (de,en,nl,no,sv) and **all** languages.[3] Due to space restrictions, we here mainly focus on the comparison with Norwegian and Swedish (cf. Figure 3a). Danish has fewer `det` relations than the other two Scandivinavian languages, but even fewer than the average language in UD. We attribute this difference between the Scandinavian languages and the rest to their nominal definite inflection pattern [5].

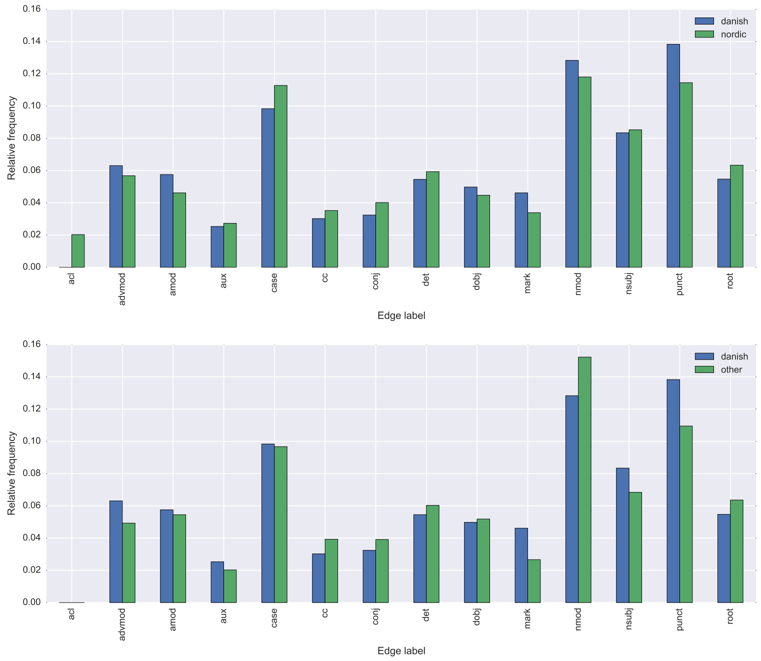


Figure 3: Dependency labels distribution comparison for Danish vs. Nordic Swedish and Norwegian (above), and for Danish vs. all languages (below).

More surprisingly, we observe that Danish stands out in the amount of *punct* relations. Examining Figure 4, we observe that punctuations have far longer average dependency length for punctuations than the UD treebanks as a whole. This difference might be a result of the relatively high number of punctuation sym-

[3] We compare with the over UD treebanks from version 1.2, released November 15th, 2015.

bols, as well as the reattachment operations that attach punctuation far from the dependent to avoid crossing edges. We observe a similar pattern for average head distance in coordinations, which might also be a result of the heuristics applied in the coordination.

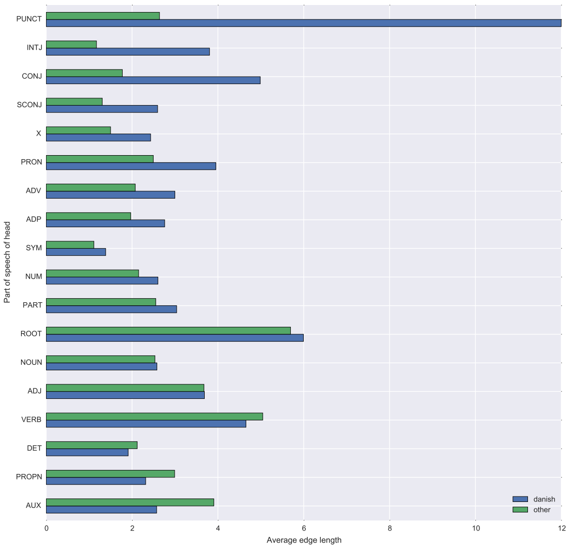


Figure 4: Average distance to head by part of speech of the dependent, compared between Danish and the average of all other UD treebanks.

## 5 Conclusions and future work

We presented a test-driven conversion of the Copenhagen Dependency Treebank (CDT) into Universal Dependencies (UD).

Conversion to UD is an ongoing process, as the standard converges across languages. We expect to revise several aspects of the treebank for a future release: 1) a homogenous analysis of proper-name headedness in the presence of other nominal complements (‘the newspaper *The New York Times*’); 2) a semi-manual validation of the *aux/auxpass* labels for periphrastic movement verbs, because Danish movement verbs like *ankomme* (‘arrive’) use the verb *være* (‘be’) as auxiliary, and it should not be treated as *auxpass*; 3) a revision of the re-attachment of coordinating conjunctions and punctuations to control for distance to head node, and; 4) a revision of the labels for clausal complements like ccomp or csubj. This step is arguably the most difficult to automate, and might require an annotation task. Silviera & Manning [19] discuss in more details the issues of labeling phrasal and clausal relations on one layer in dependency analyses.

## References

[1] Steven Paul Abney. *The English noun phrase in its sentential aspect*. PhD thesis, Massachusetts Institute of Technology, 1987.

[2] Bernd Bohnet. Top accuracy and fast dependency parsing is not a contradiction. In *COLING*, 2010.

[3] Cristina Bosco, Vincenzo Lombardo, Daniela Vassallo, and Leonardo Lesmo. Building a treebank for Italian: a data-driven annotation schema. In *LREC*, 2000.

[4] Sabine Buchholz and Erwin Marsi. Conll-x shared task on multilingual dependency parsing. In *CoNLL*, pages 149–164, 2006.

[5] Östen Dahl. Definite articles in scandinavian: Competing grammaticalization processes in standard and non-standard varieties. *Dialectology Meets Typology: Dialect grammar from a cross-linguistic perspective*, pages 147–180, 2004.

[6] Marie-Catherine De Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D Manning. Universal stanford dependencies: A cross-linguistic typology. In *Proceedings of LREC*, pages 4585–4592, 2014.

[7] Marie-Catherine de Marneffe and Chris Manning. Stanford typed dependencies manual. In *Technical report*, 2008.

[8] Jorge Hankamer and Line Mikkelsen. A morphological analysis of definite nouns in danish. *Journal of Germanic Linguistics*, 14(02):137–175, 2002.

[9] Richard A Hudson. *Word grammar*. Blackwell Oxford, 1984.

[10] Britt Keson. Det danske morfosyntaktisk taggede PAROLE-korpus. Technical report, DSL, 2004.

[11] M.T. Kromann and S.K. Lynge. Danish Dependency Treebank v. 1.0. Department of Computational Linguistics, Copenhagen Business School., 2004.

[12] Ryan McDonald, Koby Crammer, and Fernando Pereira. Online large-margin training of dependency parsers. In *ACL*, 2005.

[13] Joakim Nivre. Towards a universal grammar for natural language processing. In *Computational Linguistics and Intelligent Text Processing*, 2015.

[14] Joakim Nivre, Željko Agić, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Cristina Bosco, Sam Bowman, Giuseppe G. A. Celano, Miriam Connor, Marie-Catherine de Marneffe, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Daniel Galbraith, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Yoav Goldberg, Berta Gonzales, Bruno Guillaume, Jan Hajič, Dag Haug, Radu Ion, Elena Irimia, Anders Johannsen, Hiroshi Kanayama, Jenna Kanerva, Simon Krek,

Veronika Laippala, Alessandro Lenci, Nikola Ljubešić, Teresa Lynn, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Shunsuke Mori, Hanna Nurmi, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Prokopis Prokopidis, Sampo Pyysalo, Loganathan Ramasamy, Rudolf Rosa, Shadi Saleh, Sebastian Schuster, Wolfgang Seeker, Mojgan Seraji, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Kiril Simov, Aaron Smith, Jan Štěpánek, Alane Suhr, Zsolt Szántó, Takaaki Tanaka, Reut Tsarfaty, Sumire Uematsu, Larraitz Uria, Viktor Varga, Veronika Vincze, Zdeněk Žabokrtský, Daniel Zeman, and Hanzhi Zhu. Universal dependencies 1.2, 2015. LINDAT/CLARIN digital library at Institute of Formal and Applied Linguistics, Charles University in Prague.

[15] Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. Universal dependencies v1: A multilingual treebank collection. *LREC*, 2016, under review.

[16] Slav Petrov, Dipanjan Das, and Ryan McDonald. A universal part-of-speech tagset. CoRR abs/1104.2086, 2011.

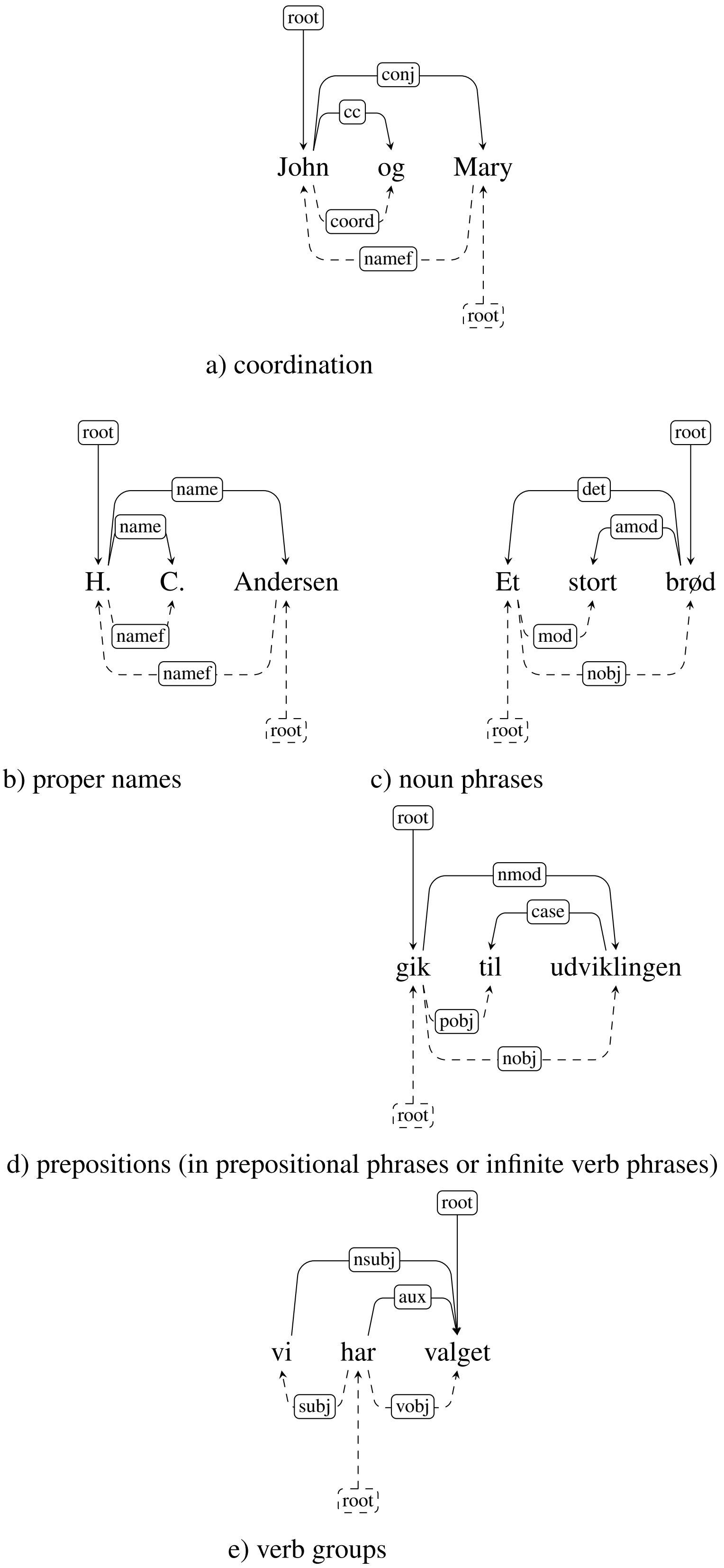
[17] Jeffrey Punske. Functional structure inside nominal phrases. *The Routledge Handbook of Syntax*, page 65, 2014.

[18] Sampo Pyysalo, Jenna Kanerva, Anna Missilä, Veronika Laippala, and Filip Ginter. Universal Dependencies for Finnish. In *Nordic Conference of Computational Linguistics NODALIDA 2015*, page 163, 2015.

[19] Natalia Silveira and Christopher Manning. Does universal dependencies need a parsing representation? an investigation of english. *Depling 2015*, page 310, 2015.

[20] Daniel Zeman. Reusable tagset conversion using tagset drivers. In *LREC*, 2008.

# Appendix



a) coordination

b) proper names

c) noun phrases

d) prepositions (in prepositional phrases or infinite verb phrases)

e) verb groups

Figure 5: Example of annotation differences; CDT scheme (dashed); UD (solid).