Increasing return on annotation investment: the automatic construction of a Universal Dependency treebank for Dutch

Bouma, Gosse; van Noord, Gerardus

Published in:
Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)

Publication date:
2017

Increasing return on annotation investment: the automatic construction of a Universal Dependency treebank for Dutch

Gosse Bouma
Centre for Language and Cognition
University of Groningen
g.bouma@rug.nl

Gertjan van Noord
Centre for Language and Cognition
University of Groningen
g.j.m.van.noord@rug.nl

Abstract

We present a method for automatically converting the Dutch Lassy Small treebank, a phrasal dependency treebank, to UD. All of the information required to produce accurate UD annotation appears to be available in the underlying annotation. However, we also note that the close connection between POS-tags and dependency labels that is present in UD is missing in the Lassy treebanks. As a consequence, annotation decisions in the Dutch data for such phenomena as nominalization and clausal complements of prepositions seem to differ to some extent from comparable data in English and German.

Because the conversion is automatic, we can now also compare three state-of-the-art dependency parsers trained on UD Lassy Small with Alpino, a hybrid Dutch parser which produces output that is compatible with the original Lassy annotations.

1 Introduction

We present a method for automatically converting Dutch treebanks annotated according to the guidelines of the Lassy project (van Noord et al., 2013) to Universal Dependencies. The Lassy annotation guidelines combine elements from phrase structure treebanks (such as phrasal nodes and use of co-indexed nodes for encoding fronted WH-constituents) with elements from dependency treebanks (such as dependency labels, discontinuous constituents and crossing branches), similar to the Tiger (Brants et al., 2002) and Negra (Skut et al., 1998) corpora for German. The conversion is done by means of an automatic conversion script.[1]

There are two advantages to such a procedure: the original annotation is of high quality as it is the result of careful manual checking and correction of automatically produced parser output. Using this investment as basis for the UD annotation as well means that this investment can also serve as basis for novel annotation projects. Second, an automatic conversion script allows any material that has been annotated according to the guidelines of the Lassy project to be converted to UD, and thus also can be used to convert treebanks outside the UD corpus and to make existing tools compliant with UD.

There are two main challenges for the conversion: the Lassy treebanks contain dependency relations between phrasal nodes, whereas UD uses lexical dependency relations only. Second, the Lassy Treebanks use a more traditional notion of '*head*' whereas UD gives precedence to content words over function words. As a consequence, converting from Lassy to UD requires '*head-switching*' in a number of cases. In section 2 we outline the main principles of the conversion process.

The conversion has been used to produce UD Dutch Lassy Small (v1.3 and 2.0). Lassy Small is a manually verified 1 million word treebank for Dutch, consisting of mixed sources. For reasons of intellectual property rights, only the Wikipedia part (7.641 sentences, 101.841 tokens) is included in the UD corpus.

One of the goals of the UD enterprise is to ensure similar annotations for similar constructions across languages. While the current state of the general and language specific annotation guidelines suggest that this should be possible for the most common syntactic configurations, it is also true that there still appears to be variation in the way less frequent constructions are annotated. This is particularly true if such constructions challenge the UD annotation principles. We

[1] Available at https://github.com/gossebouma/lassy2ud

Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017), pages 19–26,
Gothenburg, Sweden, 22 May 2017.

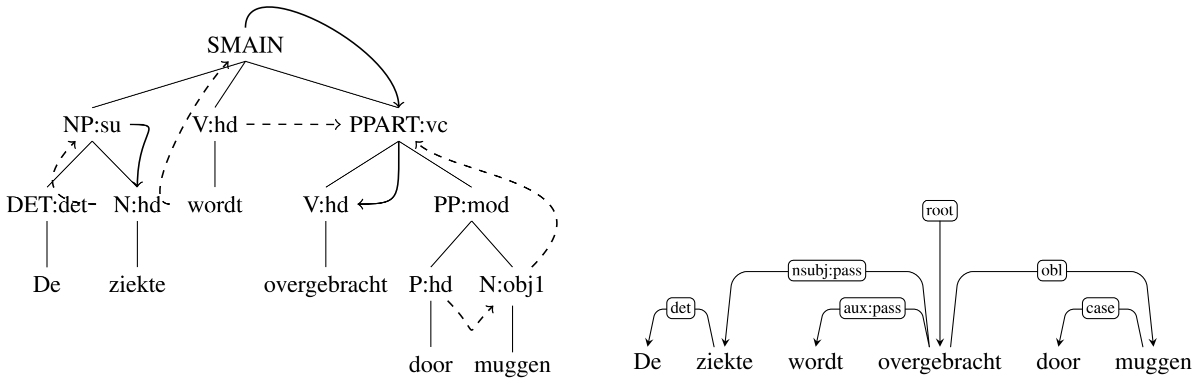

Figure 1: Phrasal annotation and the induced dependency annotation for *de ziekte wordt overgebracht door muggen* (*the disease is transmitted by flies*). External head projection paths are indicated by dashed arrows, and internal heads are indicated by solid arrows.

illustrate this in section 3 by comparing the analysis in Dutch, German, and English, of verbal nominalizations and clausal arguments of prepositions.

In section 4, we compare the performance of three dependency parsers trained on UD Lassy Small with Alpino (van Noord, 2006), a rule-based grammar that produces output compatible with the original Lassy treebank. The comparison crucially relies on the fact that we can use the conversion script to convert Alpino output to UD.

2 Conversion Process

Conversion of a manually verified treebank to UD is possible if the underlying annotation contains the information that is required to do a mapping from the original annotation to POS-tags and bilexical dependencies that is conformant with the annotation guidelines of the UD project. By doing an automatic conversion, we follow a strategy that has been used to create many of the other UD treebanks as well (Zeman et al., 2014; Johannsen et al., 2015; Øvrelid and Hohle, 2016; Ahrenberg, 2015; Lynn and Foster, 2016).

Conversion of Lassy to UD POS-tags can be achieved by means of a simple set of case statements that refer to the original POS-tag and a small set of morphological feature values. The only case that is more involved is the distinction between verbs and auxiliaries. This distinction is missing in the POS-tags and morphological features of the Lassy treebanks, but can be reconstructed using the lemma and valency of the verb (i.e., a limited set of verbs that select for only a subject and a non-finite verbal complement or predicative complement are considered auxiliaries).

Conversion of the phrasal syntactic annotation to dependency relations is driven by the observation that in a dependency graph each word (except the root) is linked via a labeled arc to exactly one lexical head.[2] Given a sentence annotated according to Lassy guidelines, we can use the phrasal syntactic annotation to predict for each token in the input (except the root) its lexical content head and dependency label. Figure 1 gives an overview of the most important Lassy dependency labels and their UD counterparts.

The rules for finding the content head are defined using two auxiliary notions: the *'external head projection'* of a word or phrase is the node that contains the content head for the node. The *'internal head'* of a node or phrase is the node that is the content head of the phrase.

In regular configurations, the external head projection of a non-head word (i.e. a word labeled *su, obj1, det, mod, app, etc.*) is its mother node, and the internal head of this mother node is the node with dependency label *hd*. This is shown for example in Figure 1, where the determiner *De* has the NP as its external head projection. The *hd* node within this NP is the content head of this phrase, and thus the content head of the determiner. The external head projection of the *hd* word itself, *ziekte*, is not the parent but the grandparent node, SMAIN. Thus, the content head of *ziekte* is

[2] Currently, no secondary edges are used in the UD Lassy Small.

Lassy	UD	Interpretation
su	subj | csubj | nsubj:pass | csubj:pass	various kinds of subjects
obj1	obj | obl | nmod	objects of verbs and prepositions
obj2	iobj	indirect objects
mod	obl | advmod | advcl | nmod | amod	various kinds of modifiers
det	det | nummod	determiners and numbers
app	appos	appositions
cmp	mark	complementizers
crd	cc	conjunctions
sup	expl	expletives
pobj1	expl	expletives
hd		heads of phrases: check label of mother node
...	...	...

Table 1: Overview of re-labeling rules

the internal head of the SMAIN. In this case, as we explain below, this is not the *hd* daughter, but the content head of the daughter labeled with dependency label *vc* (*verbal complement*).

Head-switching cases are exceptions to the general rule. The noun *muggen*, for instance, has a prepositional head as sister. As UD specifies that prepositions are dependent on the noun in these cases, we have to specify that the external head projection of the *obj1* child inside a PP is the parent of the PP. For the same reason, the external head projection of the preposition is not the grandparent, but the sister *obj1* node. The same applies to auxiliaries. As they are dependents of the main verb, their external head projection is the sister node labeled *vc* (or *predc* in copula constructions).

Finally, as the main verb *overgebracht* functions as content head of both the PPART and SMAIN, its external head projection should be the parent of SMAIN. As SMAIN is the root of the phrasal tree, we conclude that *overgebracht* must be the root node.

The analysis of WH-questions and relative clauses in the Lassy treebank uses a co-indexing scheme between the fronted element and an empty node that is comparable to a 'trace' in transformational approaches. The content head for such co-indexed fronted elements can be found by starting the external head projection identification from the co-indexed empty 'trace' node.

The identification of the correct dependency label for a bi-lexical dependency uses a mapping from the original Lassy dependency labels to UD. The most important cases are listed in Table 1.

We have used the conversion script to create UD Lassy Small (v1.3 and 2.0). This corpus consists of the Wikipedia section of the manually verified part of the Lassy corpus.

One aspect of the corpus that is not according to UD is the annotation of interpunction. As all punctuation marks are attached to the root node in the original treebank, locating the right attachment site according to UD rules is challenging. So far, we have not been able to come up with an error-free solution.

By way of evaluation of the result, we manually verified the annotation for 50 arbitrarily selected sentences from the corpus (v 2.0).[3] We checked whether the annotation was in accordance with the UD guidelines. In cases where we were not sure about the correct annotation (typically attachment decisions), we compared the annotation with the original Lassy treebank annotation. Ignoring punctuation issues, we observed 4 errors: a passive subject labeled as regular subject, an *amod* that has to be *advmod*, a number marked as *det* (should be *nummod*), and an error resulting from head-switching: the auxiliary *bekend staan* (*be known as*) consists of a vebal head and a particle. The head is marked as *cop*, but as a consequence of head-switching, the particle has been reattached to the predicative head. This is clearly wrong, although the right annotation is not obvious: either this verb should not be considered an auxiliary, or else we must allow for particles to be dependents of auxiliaries. In addition to these errors we also found 6 dubious decisions (unclear distinctions between *amod* and *advmod* (4×), labeling a

[3]All sentences from the training section containing the adverb *ook* (*also*).

predicative phrase as *xcomp*, and a case of an incomplete word (part of a coordination) marked as X (in accordance with the original annotation but not the best option according to UD),

We also tried to compare Lassy Small with the UD Dutch corpus that has been included in UD since v1.2. The latter corpus is a conversion of the Alpino treebank (van der Beek et al., 2002). It was used in the CONLL X shared task on dependency parsing (Buchholz and Marsi, 2006) and converted at that point to CONLL format. The UD version is based on a conversion to HamleDT to UD (Zeman et al., 2014). The various conversion steps have lead to loss of information,[4] and apparent mistakes,[5] and the quality of this corpus in general seems to be lower than the UD Lassy Small corpus. A more systematic comparison will be possible once we have been able to reconstruct the original sources of the material included in the Alpino treebank fragment used for CONLL. At that point, it will also be possible to create an improved version of the data using the current conversion script.

3 Cross-lingual comparison

The inventory of dependency labels in UD is a mixed functional-structural system, which distinguishes oblique arguments, for instance, on the basis of their part-of-speech, i.e. a PP dependent is labeled *obl*, a dependent clause *advcl*, and an adverbial *advmod*. Also, attachment to predicates is differentiated from attachment to nominals.

The original Lassy Small treebank has both phrasal categories and dependency labels, and seems to make a more clear-cut distinction between structural and dependency information. For instance, a single *mod*-relation is used for adjuncts in the verbal domain (PPs, adverbs and adverbial phrases, as well as clausal adjuncts) as well as in the nominal domain. The relevant structural distinctions are not lost, as phrasal nodes can be differentiated by the category and lexical items by their POS.

The mixed functional-structural approach of UD leads to surprising outcomes in cases where

[4]for instance, all heads and dependents of compound relations have been assigned the POS tag X, ignoring the original assignment of POS tags

[5]e.g. in 13.050 sentences, there are 353 cases where a verb or (proper) noun functions as dependent of the *case* relation, and 953 cases where an auxiliary has an *nsubj* dependent

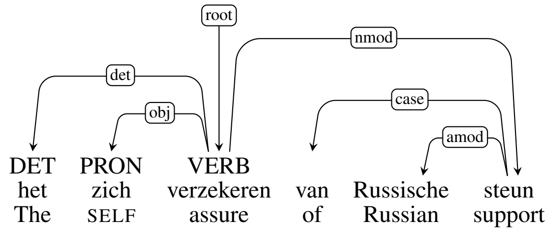

Figure 2: Nominalization in UD-Dutch Lassy Small

structural relations do not align with the predicate/nominal distinction. We discuss two such situations below.

3.1 Nominalizations

Nominalizations are constructions in which a verb functions as a noun. Nominalizations can be formed by means of derivational morphology, but there are also many cases in which there is no (overt) morphological suffix to mark the nominal status of the verb (Chomsky, 1968). Interestingly, in nominalizations we see both dependents typically associated with the verbal domain as well as dependents associated with the nominal domain, as in example (2). Here, a verb clearly heads a nominal phrase, as it is introduced by a determiner. Yet, at the same time, it selects an inherent reflexive pronoun, something that is not possible for nouns. The dependency annotation for this example in Figure 2 also shows that the PP phrase is labeled *nmod*, giving preference to the nominal interpretation of *verzekeren*. Note that the the parallelism between (1) and (2) suggests that it could perhaps also have been labeled *obl*. In fact, the NomBank corpus (Meyers et al., 2004) adopts the rule that the same semantic role labels should be used as much as possible for verbs and nominalised versions of these verbs.

(1) Hij verzekert zich van Russische steun
He assures himself of Russian support

(2) het zich verzekeren van Russische steun
the self assuring of Russian support
was het doel
was the goal
'The assuring oneself of Russian support was the goal'

The presence of such mixed nominal/verbal configurations differs strongly between treebanks. In Table 2 we give counts for the number of verbs that have a *det* dependent, and for verbs that have an incoming dependency label *nsubj* or *obj* in the Dutch Lassy Small and German and English UD treebanks (v2.0).[6] In all cases, we are dealing with a verb that has clearly nominal properties: it has a determiner as dependent, or functions as subject or object of a predicate. The Dutch treebank has the highest number of nominalizations. It should also be noted that the (14) cases in English where a verb has a *det* dependent include bona fide cases like *the following* and *(please use) the attached*, but also several apparent annotation errors. The low number of nominalizations for English is unexpected, as nominalizations involving gerunds appear to be a common phenomenon in English.

Query	NL (101K)	DE (277K)	EN (229K)
`VERB >det _`	112.9	22.0	6.1
`VERB <nsubj _`	62.4	1.4	5.2
`VERB <obj _`	21.8	2.2	8.3

Table 2: Frequency per 100.000 tokens for verbal heads with nominal properties in three UD treebanks.

3.2 Clausal arguments of prepositions

Another situation where the distinction between the predicative and nominal domain gives surprising results are PPs containing a verbal rather than a nominal content head. Some of these are nominalizations, and were already discussed above. However, there are also genuine clausal cases as in Figure 3.

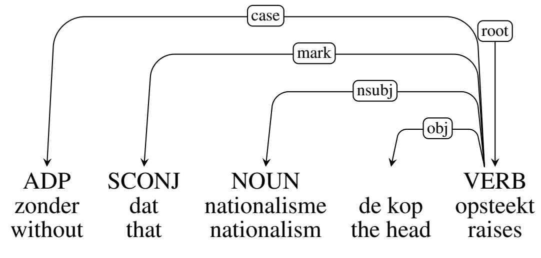

Figure 3: Prepositional phrases containing a clausal argument: *'without nationalism raising its head'*

Again, we compared counts for such phenomena in the Dutch, German, and English UD treebanks. Table 3 shows that verbs with a preposition as dependent or verbs heading a phrase with label *obl* (i.e., verbs heading an oblique dependent of a predicate) do hardly occur in English, but do occur with some frequency in Dutch and German. However, closer inspection of the German data suggests that these are dominated by annotation errors of the form *nach Dortmund gefahren (driven to Dortmund)*, where a regular *obl* dependent of a verb has been annotated erroneously with a *case* relation between the verb and the preposition. Prepositions are seen as case markers in UD, and for that reason should only have dependents themselves in exceptional cases. Most of these cases are fixed phrases of the form *due to* or *because of*. This is true for the Dutch data and to a large extent also for the English data. The German data, however, has a high number of prepositions with dependents that are not labeled *fixed*. This might be another signal of the same annotation error, in that prepositions in the German data apparently head regular PPs in many cases.

query	NL	DE	EN
`VERB >case ADP`	102.0	105.8	0.4
`VERB <obl _`	64.4	0.0	4.8
`ADP > _`	357.4	253.4	97.8
`ADP >fixed _`	318.8	6.9	30.6

Table 3: Frequency per 100.000 tokens for verbs with a *case* dependent and prepositions governing a dependent.

We believe that the relatively high number of 'non-canonical' configurations in the Dutch Lassy Small treebank may well be due to the fact that POS-tagging and syntactic annotation were performed as two independent annotation tasks in the original Lassy treebank. As a consequence, in both annotation tasks annotators made the decision that seemed most appropriate for that task (i.e. choosing the correct POS-tag and syntactic annotation, respectively). The English UD treebanks, on the other hand, is a manually verified and corrected version of an automatic conversion of the Web treebank, where POS-tags have been added automatically. The construction of the German treebank was done automatically and is minimally documented.[7] Therefore, we cannot be sure whether the differences observed above reflect genuine typological differences or whether they are a consequence of the decisions made in

[6] All counts in this section have been collected using the `dep_search` facility of `bionlp-www.utu.fi`.

[7] `https://github.com/UniversalDependencies/UD_German`

the underlying annotation and/or of the conversion method.

4 Parsing Experiments

The inclusion of a large number of languages and corpora in the UD corpus has led to a growing number of parsing toolkits that are language independent and that can be trained and evaluated on any of the UD treebanks. In this section, we compare state-of-the-art dependency parsers for UD trained on Lassy Small with Alpino, a parser based on a hand-written grammar for Dutch.

Andor et al. (2016) introduce SyntaxNet, an open-source implementation of a novel method for dependency parsing based on globally normalized neural networks. They also provide a pretrained parser for English, Parsey McParseface. On the Penn Treebank, the released model for English (Parsey McParseface) recovers dependencies at the word level with over 94% accuracy, beating previous state-of-the-art results.

SyntaxNet has been used to train a parser for a large number of corpora in UD (v1.3). 'Parsey's Cousins'[8] is a collection of syntactic models trained on UD treebanks, for 40 different languages. Per language, more than 70 models have been trained, leading to models that are up to 4% more accurate than models trained without hyperparameter tuning.

The easy-first hierarchical LSTM model of Kiperwasser and Goldberg (2016) introduces a novel method for applying the LSTM framework to tree structures that is particularly apt for dependency parsing. Another notable feature is that it does not use word embeddings. It achieves state-of-the-art results on dependency parsing for English and Chinese, and can be used to train parsers for any language for which a UD treebank is available.[9]

'ParseySaurus' (Alberti et al., 2017) is a collection of models for UD version 2.0 corpora. It uses a variant of SyntaxNet that also includes character level embeddings. The model has a labeled attachment accuracy score that is on average 3.5% better than the SyntexNet models of Parsey's cousins.

Alpino (van Noord, 2006) is a wide-coverage parser for Dutch consisting of a carefully developed hand-written unification-based grammar and a maximum entropy disambiguation model. Its output is compatible with the original Lassy treebank. Although Alpino is not a dependency parser, it can be evaluated on UD Dutch data by converting the parser output into UD compatible annotation using the same conversion script that was also used to convert the original Lassy Small treebank to UD. For the experiment, the disambiguation model for Alpino was trained only on the training section of Lassy Small UD treebank (6641 sentences).

	LAS	UAS
Alpino	84.31	89.22
Parsey's Cousins	78.08	81.63
Easy-first	77.16	81.10
ParseySaurus	80.53	84.02

Table 4: Parse results for UD Dutch Lassy Small (v1.3), using standard training (6641 sentences) and test set (350 sentences), using CONLL 2007 evaluation script, not counting punctuation.

We compare the accuracy of the Dutch SyntaxNet models as well as a model trained with the easy-first LSTM model, with results obtained using Alpino. Table 4 gives labeled and unlabeled attachment accuracy scores on the test set of the Lassy Small corpus. The scores for Parsey's Cousins are the scores reported in the Google blog post. The scores for easy-LSTM were obtained by running the code using the default options. The scores for ParseySaurus are taken from (Alberti et al., 2017).

Among the dependency parsers trained on the treebank data, the ParseySaurus model achieves a 2.5-3.0% LAS improvement over the two other models. Alpino performs even better, with a 3.8% LAS improvement over the best dependency parser model. We can only speculate about the reasons for this difference. The training corpus is relatively small, and it might be that the purely data-driven approaches would benefit relatively strongly from being trained on more data.[10]

On the other hand, results can also be improved by simply correcting errors in the original data. As we pointed out in section 3, one difference between the original Lassy dependency annotation and UD is that UD dependency labels are organized more strongly in accordance with the POS tag of

[8] `research.googleblog.com/2016/08/meet-parseys-cousins-syntax-for-40.html`

[9] https://github.com/elikip/htparser

[10] However, note that if we use the standard Alpino disambiguation component, trained on a larger, news domain corpus, its accuracy slightly decreases (88.21 UAS).

	LAS	UAS
Alpino	84.31	89.22
with corrected treebank	85.95	89.41

Table 5: Parse results for UD Dutch Lassy Small (v1.3 with corrections), using standard training (6641 sentences) and test set (350 sentences), using CONLL 2007 evaluation script, not counting punctuation.

the head and the dependent than the Lassy dependency labels. The relative independence of Lassy POS annotation and syntactic analysis (as well as the fact that these were done by different partners in the Lassy project), has led to a situation where errors in POS annotation have gone largely unnoticed when evaluating parser output. For evaluation on UD treebanks, annotating and predicting the correct POS tag is crucial, as it influences the choice of the dependency label. Thus, correcting POS tags in the original treebank leads to more consistent data in the original treebank as well as in the converted UD treebank. If we evaluate the Alpino parser on a version of the UD treebank based on the corrected underlying Lassy Small treebank, we obtain the accuracy scores given in table 5. These corrections have been included in UD 2.0 release.

5 Conclusions

Automatic conversion of existing treebanks to UD has the advantage that existing annotation efforts can be re-used, that treebanks that for some reason cannot be included in the UD corpus can be converted easily, and that tools developed for the original annotation can be used to produce UD compliant output as well. We have developed a method for converting the Dutch Lassy treebank to UD. It has been used to produce UD Dutch Lassy Small, included in UD v1.3 and v2.0.

Although all information required to do the conversion appears to be present in the underlying annotation (with the exception of punctuation attachment perhaps), we did notice that there are also subtle differences between the Lassy treebank annotation and UD annotation guidelines. This is particularly clear in cases where structural and functional information does not align well, as in nominalizations.

Using our automatic annotation script, we were able to compare parsing accuracies for three dependency parsers and Alpino. Although the results for dependency parsing are encouraging, the Alpino parser, based on a hand-written grammar, still outperforms these approaches.

In future work, we would like to expand the UD Lassy Small corpus by including more of the material of the original Lassy Small corpus (where this is allowed according to IPR). For parser evaluation, it would be interesting to see what the effect is of larger training sets on automatically trained dependency parsers in particular.

References

Lars Ahrenberg. 2015. Converting an English-Swedish parallel treebank to universal dependencies. In *Third International Conference on Dependency Linguistics (DepLing 2015), Uppsala, August 24-26*, pages 10–19. Association for Computational Linguistics.

Chris Alberti, Daniel Andor, Ivan Bogatyy, Michael Collins, Dan Gillick, Lingpeng Kong, Terry Koo, Ji Ma, Mark Omernick, Slav Petrov, Chayut Thanapirom, Zora Tung, and David Weiss. 2017. Syntaxnet models for the CoNLL 2017 shared task.

Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, and Michael Collins. 2016. Globally normalized transition-based neural networks. In *Proceedings of the ACL.*

Sabine Brants, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith. 2002. The TIGER treebank. In *Proceedings of the workshop on Treebanks and Linguistic Theories*, volume 168.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, pages 149–164. Association for Computational Linguistics.

Noam Chomsky. 1968. *Remarks on nominalization.* Linguistics Club, Indiana University.

Anders Johannsen, Héctor Martínez Alonso, and Barbara Plank. 2015. Universal dependencies for Danish. In *International Workshop on Treebanks and Linguistic Theories (TLT14)*, page 157.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Easy-first dependency parsing with hierarchical tree LSTMs. *Transactions of the ACL*, 4:445–461.

Teresa Lynn and Jennifer Foster. 2016. Universal dependencies for Irish. In *Celtic Language Technology Workshop*, pages 79–92.

Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004. Annotating noun argument structure for NomBank. In *LREC*, volume 4, pages 803–806.

Lilja Øvrelid and Petter Hohle. 2016. Universal dependencies for Norwegian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation. Portorož, Slovenia*.

Wojciech Skut, Thorsten Brants, Brigitte Krenn, and Hans Uszkoreit. 1998. A linguistically interpreted corpus of German newspaper text. *arXiv preprint cmp-lg/9807008*.

Leonoor van der Beek, Gosse Bouma, Rob Malouf, and Gertjan van Noord. 2002. The Alpino dependency treebank. In *Computational Linguistics in the Netherlands (CLIN) 2001*, Twente University.

Gertjan van Noord, Gosse Bouma, Frank van Eynde, Daniel de Kok, Jelmer van der Linde, Ineke Schuurman, Erik Tjong Kim Sang, and Vincent Vandeghinste. 2013. Large scale syntactic annotation of written Dutch: Lassy. In Peter Spyns and Jan Odijk, editors, *Essential Speech and Language Technology for Dutch: the STEVIN Programme*, pages 147–164. Springer.

Gertjan van Noord. 2006. At last parsing is now operational. In Piet Mertens, Cedrick Fairon, Anne Dister, and Patrick Watrin, editors, *TALN06. Verbum Ex Machina. Actes de la 13e conference sur le traitement automatique des langues naturelles*, pages 20–42.

Daniel Zeman, Ondřej Dušek, David Mareček, Martin Popel, Loganathan Ramasamy, Jan Štěpánek, Zdeněk Žabokrtskỳ, and Jan Hajič. 2014. HamleDT: Harmonized multi-language dependency treebank. *Language Resources and Evaluation*, 48(4):601–637.