Dependencies over Times and Tools (DoTT)

Andy Lücking[*], Giuseppe Abrami[*], Leon Hammerla[*],
Marc Rahn[†], Daniel Baumartz[*], Steffen Eger[‡], Alexander Mehler[*]

[*]Goethe University Frankfurt, Text Technology Lab
Robert-Mayer-Str. 10, 60325 Frankfurt
{luecking, abrami, baumartz, mehler}@em.uni-frankfurt.de, leon.hammerla@gmx.de

[†]Senckenberg Society for Nature Research, Leibniz Institution for Biodiversity and Earth System Research
Senckenberganlage 25, 60325 Frankfurt
marcrahn51@gmail.com

[‡]Universität Bielefeld
Inspiration 1, 33619 Bielefeld
steffen.eger@uni-bielefeld.de

Abstract

Purpose: Based on the examples of English and German, we investigate to what extent parsers trained on modern variants of these languages can be transferred to older language levels without loss. **Methods:** We developed a treebank called DoTT (available at https://github.com/texttechnologylab/DoTT) which covers, roughly, the time period from 1800 until today, in conjunction with the further development of the annotation tool DEPENDENCYANNOTATOR. DoTT consists of a collection of diachronic corpora enriched with dependency annotations using 3 parsers, 6 pre-trained language models, 5 newly trained models for German, and two tag sets (TIGER and Universal Dependencies). To assess how the different parsers perform on texts from different time periods, we created a gold standard sample as a benchmark. **Results:** We found that the parsers/models perform quite well on modern texts (document-level LAS ranging from 82.89 to 88.54) and slightly worse on older texts, as expected (average document-level LAS 84.60 vs. 86.14), but not significantly. For German texts, the (German) TIGER scheme achieved slightly higher LAS/UAS scores than UD. **Conclusion:** Overall, this result speaks for the transferability of parsers for German and English to past language levels, at least dating back until around 1800. This very transferability, it is however argued, means that studies of language change in the field of dependency syntax can draw on dependency *distance* but miss out on some grammatical phenomena.

1. Introduction

Given that natural language processing tools such as dependency parsers are trained on sentences from resources of more recent language use, the question arises of how they perform on older texts. This becomes even more pressing if former language levels (i.e., language levels that are known to be subject to linguistic change) are concerned. It is well-known, for instance from Old French, that the automatic annotation of dependency relations in historic text is a challenge (Guibon et al., 2014; Stein, 2014, 2016). Accordingly, a considerable decrease in performance is expected (Lazaridou et al., 2021). Knowing about the reliability of modern parsers for historical data is relevant, among others, when assessing dependency distance (i.e., the number of words intervening between two syntactically related words, Liang et al. 2017) over time (Juzek et al., 2020; Lei and Wen, 2020; Temperley, 2007), or for assessing word order variation over time (Gulordava and Merlo, 2015). Here we complement previous work on diachronic UD corpora of scientific German and English, which focuses on normalization of historical data (Krielke et al., 2022), or on improving automatic annotation of diachronic data (Schneider et al., 2014). To this end, we (i) provide a heterogeneous diachronic corpus of dependency parsed texts, and (ii) analyze the reliability of modern dependency parsers on this corpus – that is, we provide an evaluation of *dependencies over times and tools* (DoTT) without additional pre-processing. Our procedure if summarized in figure 1.

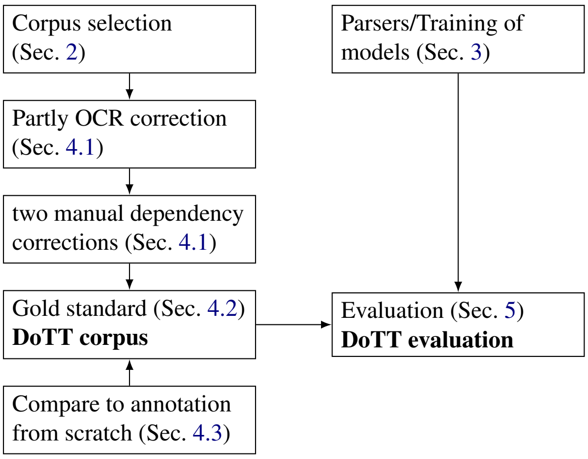
Corpus selection (Sec. 2)
Partly OCR correction (Sec. 4.1)
two manual dependency corrections (Sec. 4.1)
Gold standard (Sec. 4.2) DoTT corpus
Compare to annotation from scratch (Sec. 4.3)
Parsers/Training of models (Sec. 3)
Evaluation (Sec. 5) DoTT evaluation

Figure 1: Workflow diagram and overview of paper

LREC-COLING 2024, pages 4641–4653
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

Hypothesis: Given the pre-processing steps for improving automatic annotation in previous work, we hypothesize to detect a decrease of parser performance for older texts.

2. Corpora

DoTT focuses on German and English. To this end, four kinds of diachronic resources have been selected, two of the German language, and two of the English language. Corpus selection is driven by two considerations: domain diversity and diachronic span. The text samples cover political speech (German and English), literary texts (German) and everyday texts (e.g., newspaper; English). Given that any aspects of language may subject to change (Crystal, 1997, p. 330), there is no fixed "temporal unit" of change. Our reasoning with regard to the diachronic span of corpora therefore is as follows: an important part of language change is L1 acquisition (Lightfoot and Westergaard, 2007). Language acquisition happens, roughly, between two generations. The "length" of a generation can be approximated with 30 years (Tremblay and Vézina, 2000). Let us assume that it takes a further generation for changes to enter written texts, a duration of three generations, i.e., 90 years, seems to be a useful diachronic range to start with. We give a brief summary of our text resources in Table 1. For evaluation (Section 4), we additionally use older texts from German biodiversity literature from the BIOfid project (www.biofid.de).

3. Dependency Annotation Tools

We focus on two dependency tag sets, namely Universal Dependencies (UD; de Marneffe et al., 2021) and Tiger2 Dep. The latter is a dependency scheme derived from the TIGER treebank (Brants et al., 2002) by converting constituency structures to dependency relations (Seeker and Kuhn, 2014). To use these schemes, four of the best-performing (at time of writing; determined according to F-score for German or English) parser architectures with different language models have been used.

3.1. Pre-trained Models

The sentences selected from the Hansard corpus for manual correction have first been processed with spaCy v3 (*efficient model*; https://spacy.io), if not indicated otherwise, for basic tokenization and segmentation. We applied various kinds of dependency models on the pre-processed sentences, namely: spaCy v3 Dependency (*efficient model*), Supar (with the pre-trained models Biaffine (Dozat and Manning, 2017), CRF/matrix-tree (Koo et al., 2007; Ma and Hovy, 2017), TreeCRF-2o (Zhang et al., 2020a), Stanford UD (Chen and Manning, 2014), STEPS (Grünewald et al., 2021) (with the pre-trained models basic_mbert, basic_xlmr) and German models exclusively trained for DoTT (see Section 3.2).

3.2. Training Models for Supar

For the evaluation carried out in this paper, several dependency models for Supar have been trained using the "Supar Python Package"[1]. Table 2 collects the corresponding evaluation scores, which are measured in terms of the following metrics (ignoring punctuation). UAS: unlabeled attachment score (proportion of tokens whose head has been correctly assigned), LAS: labeled attachment score (proportion of tokens whose head has and dependency label has been correctly assigned), UCM: unlabeled complete match (proportion of sentences with correctly assigned heads), LCM: labeled complete match (proportion of sentences with correctly assigned heads and dependency labels), and processing speed (sentences per second). All values are averages obtained from all sentences contained in the sub-documents of each resource. For training, the German treebank from Universal Dependencies "UD German-HDT"[2] was used (Borges Völker et al., 2019; Hennig and Köhn, 2017; Foth et al., 2014; Foth, 2006). The evaluation dataset contains approximately 17,000 sentences, the training dataset about 170,000 sentences. Two versions of the biaffine parser were trained with the addition of a transformer model taken from the *Hugging Face* library: (i) a German BERT-large model (Chan et al., 2020);[3] (ii)a multilingual RoBERTa model, which has been fine-tuned for German.[4]

All bash-scripts for training these models can be obtained from https://github.com/texttechnologylab/DoTT/tree/main/training_scripts.

4. Corpus for Evaluation

In order to assess how the parsers perform on the heterogeneous text collection of DoTT, a subset and additional texts from biodiversity literature has been manually inspected by two linguistically trained human annotators (Section 4.1).

4.1. Manual Correction

The evaluation sample consists of 831 dependency-annotated sentences, where 164 sentences occur twice but with different dependency labels from different tag sets. Manual correction concerns OCR correction for the parliamentary texts, and correction of automatic dependency annotations by two human annotators. The sentences are distinguished according to language, corpus and dependency tag set into four partitions:

[1] https://github.com/yzhangcs/parser; (Zhang et al., 2020a,b)

[2] https://github.com/UniversalDependencies/UD_German-HDT

[3] https://huggingface.co/deepset/gbert-large

[4] https://huggingface.co/xlm-roberta-large-finetuned-conll03-german

	GerParCor (Abrami et al., 2022), (Abrami et al., 2021) includes Bundestag and DEUParl (Walter et al., 2021)
link	`https://github.com/texttechnologylab/GerParCor`
description	Plenary protocols of all German-speaking parliaments (Austrian National Council, Swiss National Council, Liechtenstein National Parliament, German Bundestag, German Bundesrat, and the 16 German national parliaments).
time / lang.	1867–2021 / DE
	coha (Davies, 2010)
link	`https://www.english-corpora.org/coha/`
description	Corpus of Historical American English, 400 million words / 107,000 texts
time / lang.	1810–2009 (acc. to website) / EN
	hansard (Davies, 2015)
link	`https://www.english-corpora.org/hansard/`
description	nearly every speech given in the British Parliament (about 1.6 billion words total acc. to website)
time / lang.	1803–2005 / EN
	dta (Deutsches Textarchiv, 2007–2016)
link	`https://www.deutschestextarchiv.de`, `http://media.dwds.de/dta/download/dta_kernkorpus_2020-07-20.zip`
description	Kernel corpus of the German Text Archive (*Deutsches Textarchiv*), version from July 20, 2020, 1,472 texts, 359M; Belle lettres (552 texts, 92M), functional literature (266 texts, 71M), science (654 texts, 198M)
time / lang.	1600–1899 (1600–1699: 237 texts, 60M; 1700–1799: 526 texts, 122M; 1800–1899: 689 texts, 167M) / DE
	BIOfid
link	`https://www.biofid.de/de/#digital-collection` (CC BY-NC-SA 4.0)
description	German botanical and biodiversity texts from the collections of the library of Goethe University Frankfurt, accessed via the specialized information service BIOfid (Driller et al., 2020).
time / lang.	since 1753 / DE

Table 1: Overview of corpora.

model-id	UCM	LCM	UAS	LAS	Sent/s
biaffine	69.20%	61.13%	96.78%	95.46%	2823.58
+gbert	75.26%	68.43%	97.56%	96.59%	360.99
+roberta	74.90%	67.90%	97.50%	96.49%	328.59
crf	69.02%	61.08%	96.69%	95.36%	2411.48
crf2o	70.22%	62.14%	96.85%	95.54%	1914.47

Table 2: Evaluation scores (assessed ignoring punctuation on the “Supar Python Package”) for the newly trained Supar models, sentence-based averages.

- parliamentary: German, 9 files à 20 sentences (= 180 samples) from 4 time periods of DEUparl (1895, 1918, 1933, 1942). These samples have been cleaned up before processing in such a way that token and sentence boundaries have been corrected if necessary and fragments have been deleted. The reason for cleaning-up is that we are interested in how well dependency parsers perform, not in how well OCR and related preprocessing tools work. This partition is, however, the only one for which such a basic correction has been effected. 164 tidied-up sentences remain. One of these sentences still contains an error: it lacks a trailing period. Furthermore, some of these sentences involve semicolons. These two properties (missing period and semicolons) are treated in different ways by different pre-processing tools: the sentence lacking a trailing period is merged in different ways with its surrounding sentences and sentences may or may not be split at the semicolon, leading to different numbers of actual sample sentences.
  - spacy, tiger2dep, 170 sentences (168 of which have been used for gold standard). For pre-processing, spaCy has been used for tokenization and sentence boundary recognition (and dependency parsing), lemmatization has been carried out with MarMot (Mueller et al., 2013).
  - steps xmlr, UD, 163 sentences (161 of which have been used for gold standard). Tokens and sentence boundaries have been brought about by the LanguageToolSegmenter from DKPro (Eckart de Castilho and Gurevych, 2014).
- German biological texts (spaCy v2, tiger2dep), 10 files from 5 time periods (pre 1890, 1890–1920, 1920–1930, 1930–1950, since 1950; each period twice) à 50 random sentences per period (= 500 input samples in total, 270 of which have been used for gold standard, the remainder had to be excluded due to some sort of error, see below).
- hansard (steps xmlr, UD, pre-processing with spaCy v3), English, 11 files à 10 sentences, from 11 time periods (1803, 1820, 1840, 1860, 1880,

1900, 1920, 1940, 1960, 1980, 2000; = 110 samples, 107 of which have been used for gold standard),

Why is the number of manually corrected sentences used for gold standard smaller than the number of samples? It is well-known that binary relations are insufficient for expressing all of syntactic configurations (de Marneffe et al., 2021, 199 et seq.). The text collection underlying DoTT is no exception. Accordingly, sentences exhibiting phenomenon that go beyond binary dependency edges had to be excluded from annotation. An example in case is verb ellipsis in the TIGER scheme, illustrated in (1), sentence no. 28 from the 1930–1950 partition of the BIOfid sample:

(1) 1942 kam der erste Storch am 15. März, der zweite einige Tage später.
(*In 1942 the first stork arrived on March 15, the second a few days later.*)

Here, it is not simply possible to connect the second sentential conjunct, since it lacks the verb.

Resolving elided predicates in Tiger2Dep requires post-processing in terms of so-called secondary edges (Albert et al., 2003, p. 117 et seqq.), which are neither supported in treebanks using the Tiger2Dep scheme we know of, nor in most annotation tools, including DEPENDENCYANNOTATOR (Abrami et al., 2021). For that reason, sentences containing verb ellipsis are excluded from Tiger2Dep corrections.[5]

There are more trivial reasons for excluding examples, however. Within the BIOfid sample, which is taken from a corpus digitized by means of OCR, we find, for instance, the following two "sentences": (i) "*), *Die Fichte (Picea excelsa Lmk.*", and (ii) "*detritus O.F.M*". Such fragments probably arise due to several pre-processing problems, and since there is no point in imposing dependency relations in such cases, they have been excluded.

An overview of the resulting dataset used for goldstandard annotation (see Section 4.2) is given in Table 3.

Manual correction has been carried out by using the DEPENDENCYANNOTATOR from the TEXTANNOTATOR suite (Abrami et al., 2019, 2020). DEPENDENCYANNOTATOR displays two dependency trees on the workspace: on the bottom half, the automatically annotated dependency structure of a sentence is shown. The upper half hosts a modifiable copy of the parser output, the dependency tree where the human annotator makes revisions, if needed. A sample screenshots is shown in Figure 2.

lang	corpus	time	dep tag	#sent	#bckt
DE	parl	1895–1942	TIGER	168	96
DE	parl	1895–1942	UD	161	93
DE	bio	1753–today	TIGER	270	—
EN	hans	1803–2005	UD	107	—
			sums:	706	189

Table 3: Summary of goldstandard DoTT. 'DE' indicates the German, 'EN' the English language. "parl" refers to GerParCor, "bio" to BIOFid, and "hans" to the Hansard corpus – see Table 1. "TIGER" is the Tiger2Dep tag set, Universal Dependencies are abbreviated as "UD". The number of sentences ('#sent') and "bucket" files (i.e., sentences that lack time allocation) are given ('#bckt').

Table 7 in the appendix provides a detailed overview of the agreement values of the two annotators $a1$ and $a2$ on the corrected files. The results show that the individual agreement of the human annotators with the machine is slightly better than the agreement among the human annotators. That is, annotators corrected slightly different things. Therefore, negotiation of annotators with regard to a gold standard is needed.

4.2. Gold Standard

All sentences which have been independently corrected by two human annotators in the evaluation corpus (Section 4) have been jointly inspected by the two annotators to create the goldstandard. Agreeing on a gold standard – including negotiating which samples have to be excluded – has been carried out by means of a newly implemented function in DEPENDENCYANNOTATOR, namely automatic annotation graph comparison. Having created a new view (*viz.*, the gold standard view) from an existing one, both annotators' dependency trees are compared, and any differences are highlighted – see the purple edge in Figure 3. That is, annotators only need to consider divergent edges. Potential corrections are made in the top-most dependency graph, which hosts the gold standard. The resulting gold-standard is used as the benchmark for evaluating dependency annotation tools described in Section 5 and is available at https://github.com/texttechnologylab/DoTT.

4.3. Annotation from Scratch

In correcting automatically generated annotations, human annotators may be biased by trusting parser annotations (Fort and Sagot, 2010). Notwithstanding, correcting automatically produced dependency annotations increases consistency compared to fully manual annotations (Mikulová et al., 2022). In order to assess the magnitude of this bias within DoTT, we created

[5] Verb ellipsis can be resolved in *enhanced* UD, however (https://universaldependencies.org/u/overview/enhanced-syntax.html#ellipsis; we thank an anonymous reviewer for pointing this out). In standard UD, verb ellipsis can be labeled, though not resolved: missing material is indicated by an ORPHAN edge to the "next" non-elided expression. Accordingly, sentences containing elided verbs remain included in UD files. Since this phenomenon is very rare in DoTT (only a few instances), we do not expect any bias therefrom.

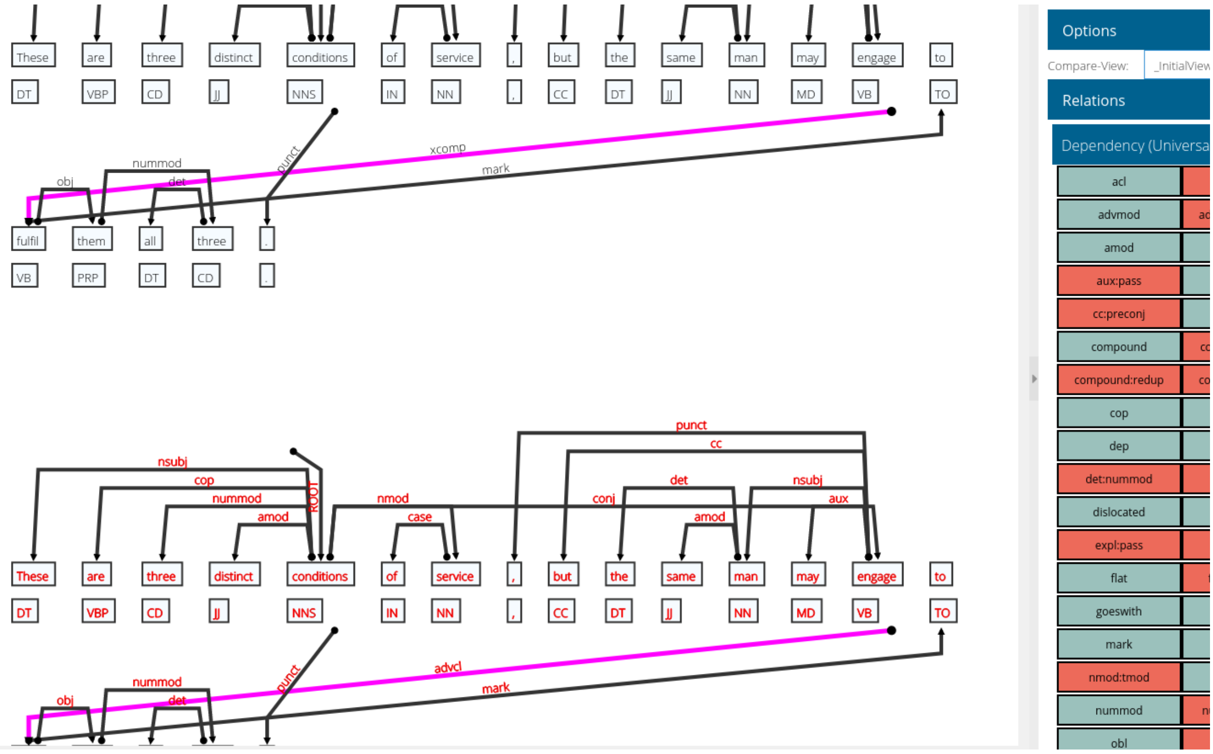

Figure 2: Evaluating Steps with the xmlr model and the Universal Dependency tag set. In the example sentence, the annotator corrected the ADVCL edge connecting *engage* and *fulfill* to XCOMP, a rather common mistake of parsers (see Section 4.4).

a sample of sentences which are manually annotated from scratch. To this end, a selection of 46 sentences from the parliamentary texts has been randomly chosen according to six partitions distinguished by sentence length, since longer sentences are in general more difficult to annotate: “short” (less than 11 words), “semi-short” (between 11 and 16 words), “middle” (between 16 and 20 words), “semi-middle” (between 20 and 25 words), “long” (between 25 and 38 words), and “very long” (more than 38 words). There are two “very long” sentences and four sentences from each of the remaining partitions. Annotations have been carried out by one of the annotators of the evaluation and goldstandard corpus, respectively, in terms of the Tiger2Dep scheme a couple of month after the goldstandard annotation. Hence, the annotator has seen the sample sentences before, but given the elapsed time and the amount of previously annotated DoTT sentences, subliminal influence is likely to be small – as is indicated by the results collected in Table 4. The results are obtained by comparing the annotations from scratch to the gold-standard annotation for the sentences in question. LAS numbers of about 0.74 are slightly worse than those of dependency tools, confirming the bias to believe the machine when correcting sentences automatically pre-annotated with dependency relations. Note, however, that there is some leeway in annotation due to several legal or prescribed interpretations; Examples being multi-token proper names, or the ambiguity of attachment points of prepositional phrases or relative clauses. This does not explain the full bias, but sets it into perspective.

Score	
LAS micro	0.78
LAS macro	0.77
UAS micro	0.84
UAS macro	0.85

Table 4: Comparing gold-standard annotations and annotation from scratch of one human annotator in terms of labeled and unlabeled attachment scores (LAS and UAS, respectively), per sentence (macro) and per document (micro).

4.4. Observations and Conventions

Besides trivial differences between dependency schemes such as chained vs. stacked annotation of conjunctions or the place of modifier attachment, there are a couple of observations.

- Consider a typical German construction such as *Der Staat und die Politik setzen die Rahmenbedingungen dafür, dass die Gewerkschaften und die Arbeitgeberverbände diese Aufgabe auch umsetzen können.*[6] How to deal with the pronominal

[6] *The state and the politicians set the framework condi-*

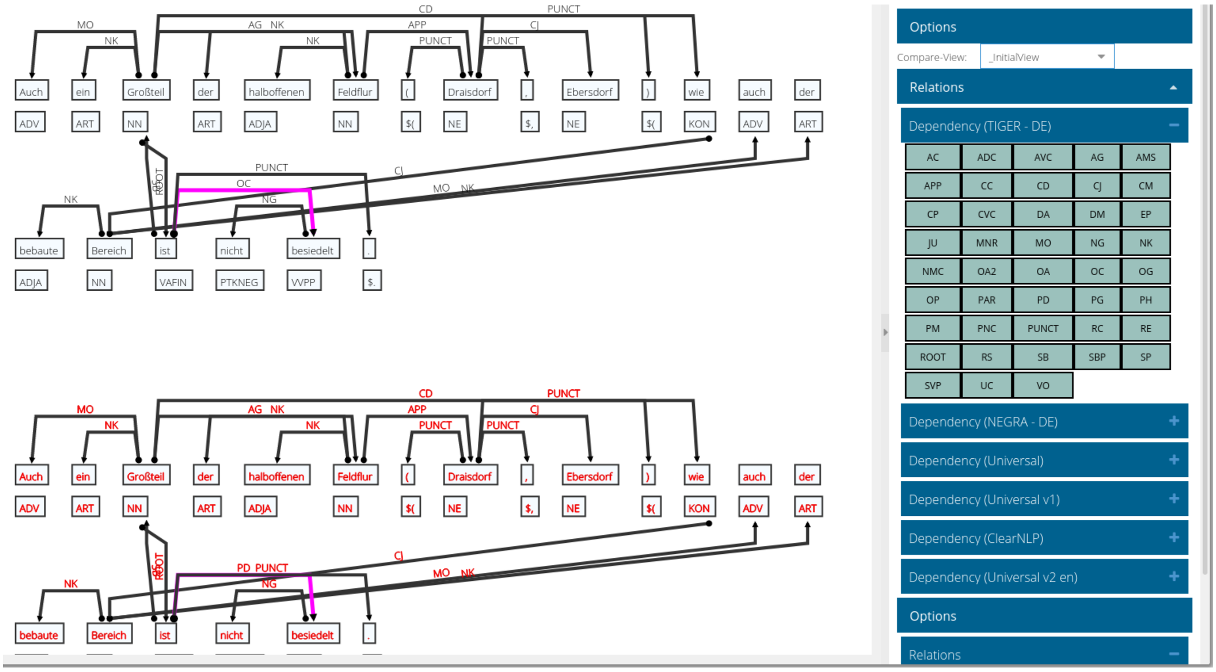
Auch ein Großteil der halboffenen Feldflur (Draisdorf , Ebersdorf) wie auch der
ADV ART NN ART ADJA NN $(NE $, NE $(KON ADV ART
bebaute Bereich ist nicht besiedelt .
ADJA NN VAFIN PTKNEG VVPP $.
MO NK AG NK CD APP PUNCT CJ ROOT OC NG PD
Options
Compare-View: _InitialView
Relations
Dependency (TIGER - DE)
AC ADC AVC AG AMS APP CC CD CJ CM CP CVC DA DM EP JU MNR MO NG NK NMC OA2 OA OC OG OP PAR PD PG PH PM PNC PUNCT RC RE ROOT RS SB SBP SP SVP UC VO
Dependency (NEGRA - DE)
Dependency (Universal)
Dependency (Universal v1)
Dependency (ClearNLP)
Dependency (Universal v2 en)
Options
Relations

Figure 3: Automatic dependency graph comparison of two annotations of the same sentence within the gold standard view in DEPENDENCYANNOTATOR. The example show a typical cause for disagreement, namely the often difficult to substantiate distinction between clausal objects (OC) and predicatives or statal passives (PD).

adverb (*dafür*) and its co-referential resolution (*dass* 'that' clause) in UD? The problem is that the syntactic position from the main verb is already occupied by the pronominal adverb before the clause can be attached. The UD guidelines do not seem to cover these examples, so we issued a corresponding poll on UD (https://github.com/UniversalDependencies/docs/issues/840). We decided to annotate pronominal adverbs and the clausal constituents they stand for in terms of OBL–ADVCL structures such that the proform becomes an oblique argument of the verb and governs a clause modifier. Note that such problems do not arise in Tiger2Dep because this scheme provides a *repeated element* relation (RE).

- How to deal with the difference between *halten* 'to hold' and "halten … für" 'to deem/think sb./sth.'? Following examples in UD_German-HDT@2.9 from Grew (e.g., the two first examples from this search: http://match.grew.fr/?corpus=UD_German-HDT@2.9&custom=61dc7560210b5), we opted for an annotation involving XCOMP or OBJ and CASE.[7] That is, in the German phrase *ich halte es für unmöglich (dass) …* 'I think it is impossible (that) …' the adjective *unmöglich* 'impossible' becomes the XCOMP of the head *halten* (which, in this reading, does not have a verbal translation into English). In case of a nominal argument as in *für Aufwand halten* 'think it is an effort', *Aufwand* 'effort' becomes the OBJ (but not the OBL) of the head *halten*. In both cases the preposition *für* is the CASE dependent of XCOMP respectively OBJ.

The general impression was that parsers perform worse on longer sentences. Since German sentences (English ones too, but to a lesser degree) of older time periods tend to be longer than more recent ones – see Figure 4 for a mean sentence length plot for the English Hansard and the German BIOfid sample – this also means that parsers can make more errors on older texts. However, this impression is statistically tested in the following section and is not confirmed.

5. Evaluation: Dependencies over Tools and Times

The "gold" dependency trees agreed upon in Section 4.2 are used as a benchmark to assess and compare various parsers and dependency schemes. The consistency of gold-standard and dependency tool annotations have been assessed in terms of *labeled* and *unlabeled attachment scores* (LAS and UAS, respectively), sentence-wise and corpus-wise (i.e., per all documents contained by a given corpus) (Kübler et al., 2009, chap. 6). Nine dependency parsing models were trained for the comparison, three (biaffine, crf and crf2o) for three different dependency label schemes used by the gold-standards. For training, the follow-

tions so that the trade unions and the employers' associations can also implement this task. The example is taken from the parliamentary corpus.

[7] We thank Kim Gerdes for pointing out these examples.

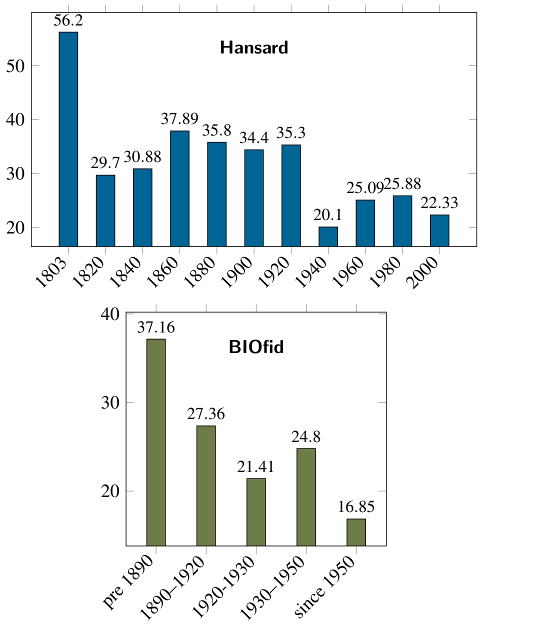

Figure 4: Mean sentence length (tokens, including punctuation) for the Hansard and BIOfid gold standard samples.

ing three treebanks were used: The German TIGER Corpus (Brants et al., 2002), the UD-German-GSD treebank `https://universaldependencies.org/treebanks/de_gsd/index.html`, and the UD-English-GUM treebank (Zeldes, 2017). Since gold annotation ignores punctuation, edges with label PUNCT are also ignored in calculating LAS and UAS. The results are collected in Table 5 ('macro' is the sentence-based average, 'doc' the average drawn over the temporal sub-documents of the data sets). With attachment score values around 80 % to 90 %, the dependency parsers perform reasonably well. The Tider2Dep scores are slightly higher than the corresponding UD scores. This is presumably due to the fact that UD is more expressive (TIGER, for instance, covers many expressions by means of noun kernel and modifier edges). But is there a difference in performance assessed in terms of attachment scores over time? To answer this question, the texts from DoTT have been partitioned into an older and a newer half.[8] The mean values of the document-wise LAS and UAS scores for both partitions are shown in Table 6, indicating that dependency tools indeed achieve better annotation results on the newer text partitions than on the older ones. The LAS macro scores of both partitions have been compared using a Student t-test (provided by R's (R Core Team, 2022) rstatix package (Kassambara, 2021), data preparation has been carried out by means of the tidyverse package collection (Wickham et al., 2019)), exemplified by the biaffine model and Stanford UD. Parsers perform about two percentage points better on newer texts, but this difference does not turn out to be statistically significant at a 5 % error level (t-test; the combined data are significant on a 10 % error level, however). Details are given in Table 8 in the appendix.

model	LAS doc	LAS macro	UAS doc	UAS macro
biofid				
biaffine tiger	85.36	83.64	88.61	87.02
crf tiger	83.52	81.56	86.95	85.35
cfr2o tiger	83.28	80.98	86.73	84.84
parliamentary ud				
biaffine gsd	82.89	83.67	90.51	91.42
crf gsd	83.54	84.19	90.61	91.49
crf2o gsd	83.78	84.24	90.83	91.39
Stanford UD	85.33	86.67	90.93	92.08
parliamentary tiger				
biaffine tiger	88.54	88.38	91.16	91.13
crf tiger	87.57	87.66	90.46	90.78
crf2o tiger	87.13	87.15	89.94	90.14
hansard				
biaffine gum	85.93	86.45	89.59	89.25
crf gum	85.40	85.54	88.94	88.24
crf2o gum	85.04	85.47	88.75	88.98
Stanford UD	86.76	88.06	87.70	90.91

Table 5: Attachment scores for parsers/models on gold standard, ignoring punctuation. LAS and UAS are the labeled respectively unlabeled attachment score, averaged in two ways: sentence-wise (*macro*), and sub-document-wise (*doc*).

Model	UAS old / new	LAS old / new
biaffine_dep_de_tiger	88.64 / 91.03	85.79 / 88.12
crf_dep_de_tiger	87.17 / 89.36	83.91 / 86.09
crf2o_dep_de_tiger	86.64 / 89.35	83.75 / 86.13
biaffine_dep_en_gum	88.64 / 91.26	85.32 / 87.01
crf_dep_en_gum	88.54 / 89.64	85.46 / 85.29
crf2o_dep_en_gum	88.21 / 89.71	84.82 / 85.43
biaffine_dep_de_gsd	89.18 / 89.68	79.47 / 83.13
crf_dep_de_gsd	88.46 / 89.81	79.35 / 83.66
crf2o_dep_de_gsd	88.66 / 90.49	80.68 / 83.93
Stanford UD	89.53 / 90.67	86.03 / 86.86
average	88.78 / 90.36	84.60 / 86.14

Table 6: Attachment scores for older and newer text partitions on document level. The different sub-models are due to the two languages of the corpus sample (German and English).

6. Discussion

There is an expectation that dependency parsers perform worse on older texts – see Section 1. In line with this expectation, we found that dependency parsers indeed perform worse on older texts, but only to a degree

[8] The bucket files from the German parliamentary debates – see Table 3 – have been omitted since they lack a clear time allocation.

that is not statistically significant. Given that the data underlying DoTT is comparatively small and from a restricted temporal period, the observed effect may turn significant on a larger and older dataset, of course.

This finding confirms that dependency parser can reasonably be applied to heterogeneous, diachronic corpora. However, it creates a conundrum in particular for dependency-based language change studies: Are dependency parsers insensitive to syntactic change in the end? Let us discuss some examples.

One reason for the unexpected outcome might be that older texts differ from newer ones stylistically, but not syntactically. Consider the rather random example in (2). While its wording suggests that the sentence has been produced in earlier times, there is – perhaps except for the salutation supplement – nothing suspicious dependency-wise.

(2) These, my lord, are a few of the leading particulars of the meetings which I attended. (Hansard corpus, 1840)

It is therefore possible, that the sentences within DoTT do not exhibit phenomena of proper syntactic change to a noteworthy extent. Indeed, the covered diachronic span considered in Section 2 may still be too short. DoTT covers about 200 years. Syntactic change, however, often takes several hundred years to develop (Mair and Leech, 2020). For instance, Middle English (up to around 1550) allowed for postverbal placement of adverbials, as in (3a), which is not licensed in Modern English anymore (Kroch, 1989), which requires the word order shown in (3b).

(3) a. Quene Ester looked never with swich an eye. (Chaucer, *Merchant's Tale*, line 1744, from the end of 14th century, cited in Kroch 1989, p. 226)

b. Quene Ester never looked with swich an eye.

The corresponding syntactic change initiated more than 500 years ago, a time span which is not covered by any of the DoTT resources. Accordingly, we expect dependency parsers to struggle with (3a) and detect a changed syntactic pattern. This is, however, not the case: Both adverbial placements (i.e., *looked never* and *never looked*) are mapped onto equivalent dependency trees by dependency parsers (tested with spaCy and biaffine). Parsers struggle with the now obsolete form "swich", a precursor of today's adjective *such*, however, which is recognized as the head noun within a propositional object. So, even syntactic change which happened between language levels not covered in our treebank need not pose a problem for dependency parsers, adding to the transferability of modern parsers to older texts.[9]

As a less clear example, consider sentence final NPs in Old High German: NP complements appeared in the postfield, which is not possible in current German any more, as in (4), taken from Meibauer et al. (2015, 318):

(4)	*dhazs*	*ir*	*chihoric*	*uuari*	*gote*
	dass	er	gehorsam	war	Gott
	that	he	obedient	was	God

"dass er gehorsam war gegenüber Gott" / *that he was obedient to God*

The free gloss in the last line shows the clause in current German, where the NP complement has to be embedded in a right dislocated PP construction. The free gloss is correctly parsed by modern dependency parsers (tested with spaCy).[10] Modern parsers, however, struggle with the Old High German form, but simply for spelling reasons (e.g., *chihoric uuari* is recognized as a named entity). If we use the old syntactic pattern, but in modern spelling (i.e., *dass er gehorsam war Gott* 'that he obedient was God'), the clause is parsed, but both the adjective *gehorsam* and the noun *Gott* are annotated as predicatives of the main verb. Thus, the functional relation of the noun – which is the subject of syntactic change in the first place – is lost. Accordingly, diachronic dependency parsing might detect a quantitative difference in arc labelling in such constructions; but since the difference rests on an unrecognized parsing error for Old High German texts, it is questionable whether this counts as evidence for syntactic change – regardless the antecedent problem that modern parsers simply cannot handle Old High German spelling.

Furthermore, some instances of historical variation documented in previous research[11] pertain to an *extension* of allowable syntactic pattern over time. For example, using the adverb *hopefully* in sentence initial position, as shown in (5), is a rather recent phenomenon.

(5) Hopefully, pointless controversies ... are dying down. (Shapiro, 1998, p. 294)

A parser which is trained on modern data that contain sentence initial as well as mid-sentence *hopefully* has no problem to annotate older texts, which are (qua syntactic change hypothesis) devoid of sentence initial uses.

Hence, while there is the expectation that considering yet older texts presumably also increases a temporal influence on dependency annotation performance of current tools, the above-given examples provide reason to assume the opposite. From this point of view, the quantitative analysis of syntactic change *in terms of dependency distance*, as studied, for instance, by Juzek et al.

[9] Kroch (1989) explains the syntactic change by the loss of verb-to-infl movement in generative grammar.

[10] The adjective *gehorsam* 'obedient' predicates of the verb, the modifier phrase *gegenüber Gott* 'to God' is attached to the finite verb *war* 'was'.

[11] See, e.g., `https://www.english-corpora.org/variation.asp`

(2020), is feasible (but still needs to be evaluated with respect to observable (here non-significant) differences between historical and contemporary data using much larger corpora). Given that dependency parsers fail to detect some attested instances of language change, as exemplified above, the DoTT evaluation raises the question of what kind of language change is covered by such methods at all. Taking older texts into considerations raises a couple of linguistic and in particular technical issues of other kinds, however, ranging from form variants to the recognition of non-standardized typescripts (Tauber, 2019; Krielke et al., 2022).

Turning to manual annotation, Tables 4 and 5 (appendix) reveal that annotation from scratch and annotation by correction bring about different results. Since the agreement (assessed in terms of LAS/UAS micro/macro) of annotation from scratch with the gold standard is, with values ranging from 0.78 to 0.85 (Table 4), lower than that of the dependency tools (0.8 to 0.92, see Table 5), there is a preference of following a given pre-annotation. Since it is known that the correction of automatic pre-annotation leads to higher consistency (Mikulová et al., 2022), the converse argument then would be that part of the lower agreement of annotation from scratch to the gold standard is due to greater variability in annotating the same phenomena. Another factor is that dependency schemes are collections of examples and heuristics, so that there is some leeway in application. Annotators who correct pre-annotations likely follow a "principle of charity": they do not correct instances that are not clearly wrong (see also the examples given as part of the discussion in Section 4.4). Hence, a gold standard derived from pre-annotations is quite consistent but is a "perspectival gold standard" seen from the point of view of the parsing tools used to bring about the pre-annotation. A way to assess the leeway of dependency schemes could be to compare dependency tags to parses of grammar-based parsers (e.g., Müller 2007; at least for those instances which are covered by lexicon and grammar rules of the grammar).[12]

7. Conclusion

We presented DoTT, a collection of gold-standard treebank of manually corrected dependency annotations for German and English texts from different time spans. DoTT has been used to evaluate a couple of dependency parsers. With an average LAS (sentence) of 85.29, parsers show to work quite reliable, comparable in sheer number to human annotation from scratch. Dependency parsers, however, tend to perform slightly worse on older texts, although not to a statistically significant degree. We hypothesize that this is due to the fact that older texts tend to contain longer sentences, which are more difficult to parse than shorter ones. In any case, we take our evaluation as evidence, that dependency parsing works well on "usual in the trade" diachronic corpora, at least for corpora containing texts from the time period covered by DoTT. Accordingly, dependency-based analyses on resources containing older texts are feasible, although raising questions on their role for *syntactic* change studies (in contrast to dependency length quantification).

We also briefly presented extensions of the DependencyAnnotator, which facilitate the gold standard annotation by means of visual dependency graph comparison and allows for a fine-grained marking of annotation samples as being "good" or exhibiting some sort of issue. That said, DoTT (https://github.com/texttechnologylab/DoTT) provides a benchmark for assessing dependency parsers using the UD or the Tiger2Dep dependency relations.

Acknowledgements

We want to thank three anonymous LREC-COLING reviewers for their valuable feedback, which led to an improvement of presentation and discussion, among others. This work is partially funded by the *Deutsche Forschungsgemeinschaft*, grant ME 2746/5-2 and grant MO 412/54-2.

8. Ethics Statement

The authors have no competing interests to declare that are relevant to the content of this article.

9. Bibliographical References

References

Giuseppe Abrami, Mevlüt Bagci, Leon Hammerla, and Alexander Mehler. 2022. German Parliamentary Corpus (GerParCor). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1900–1906, Marseille, France. European Language Resources Association.

Giuseppe Abrami, Alexander Henlein, Andy Lücking, Attila Kett, Pascal Adeberg, and Alexander Mehler. 2021. Unleashing annotations with TextAnnotator: Multimedia, multi-perspective document views for ubiquitous annotation. In *Proceedings of the Seventeenth Joint ACL - ISO Workshop on Interoperable Semantic Annotation (ISA-17)*, ISA-17.

[12] Some readers may wonder about the term "grammar-based parsers". The reason is that in linguistics a grammar is a formal generative or declarative system that aims at grammaticality and grammatical analyses. Parsing is the process of assigning a syntactic structure to an input string according to such a grammar. Dependency arc labels, in contrast, usually come as "syntactic analysis schemes" (de Marneffe and Nivre, 2019). Parsing, therefore, becomes an annotation process where a dependency graph is assigned to an input sentence (Kübler et al., 2009). Hence, such dependency parsers are not parsers in the linguistic, grammar-theoretical sense, but linguistic grammars seems to be required to analyze some sorts of language change, as above exemplified.

Giuseppe Abrami, Alexander Mehler, Andy Lücking, Elias Rieb, and Philipp Helfrich. 2019. TextAnnotator: A flexible framework for semantic annotations. In *Proceedings of the Fifteenth Joint ACL – ISO Workshop on Interoperable Semantic Annotation*, ISA-15.

Giuseppe Abrami, Manuel Stoeckel, and Alexander Mehler. 2020. TextAnnotator: A UIMA based tool for the simultaneous and collaborative annotation of texts. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 891–900, Marseille, France. European Language Resources Association.

Stefanie Albert, Jan Anderssen, Regine Bader, Stephanie Becker, Tobias Bracht, Sabine Brants, Thorsten Brants, Vera Demberg, Stefanie Dipper, Peter Eisenberg, Silvia Hansen, Hagen Hirschmann, Juliane Janitzek, Carolin Kirstein, Robert Langner, Lukas Michelbacher, Oliver Plaehn, Cordula Preis, Marcus PuSSel, Marco Rower, Bettina Schrader, Anne Schwartz, George Smith, and Hans Uszkoreit. 2003. *TIGER Annotationsschema.* Universität des Saarlandes, FR 8.7 Computerlinguistik, Universität Stuttgart, Institut für Maschinelle Sprachverarbeitung, Universität Potsdam, Institut für Germanistik.

Emanuel Borges Völker, Maximilian Wendt, Felix Hennig, and Arne Köhn. 2019. HDT-UD: A very large Universal Dependencies treebank for German. In *Proceedings of the Third Workshop on Universal Dependencies (UDW, SyntaxFest 2019)*, pages 46–57, Paris, France. Association for Computational Linguistics.

Sabine Brants, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith. 2002. The TIGER treebank. In *Proceedings of the First Workshop on Treebanks and Linguistic Theories*, TLT2002, pages 24–41, Sozopol, Bulgaria.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. arXiv eprint 2010.10906.

Danqi Chen and Christopher D. Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of EMNLP 2014*, pages 740–750.

David Crystal. 1997. *The Cambridge Encyclopedia of Language*, 2 edition. Cambridge University Press, Cambridge, UK.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational Linguistics*, 47(2):255–308.

Marie-Catherine de Marneffe and Joakim Nivre. 2019. Dependency grammar. *Annual Review of Linguistics*, 5(1):197–218.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5th International Conference on Learning Representations*, ICLR 2017. OpenReview.net.

Christine Driller, Markus Koch, Giuseppe Abrami, Wahed Hemati, Andy Lücking, Alexander Mehler, Adrian Pachzelt, and Gerwin Kasperek. 2020. Fast and easy access to Central European biodiversity data with BIOfid. In *Biodiversity Information Science and Standards*, volume 4 of *BISS*.

Richard Eckart de Castilho and Iryna Gurevych. 2014. A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Karen Fort and Benoît Sagot. 2010. Influence of Pre-annotation on POS-tagged Corpus Development. In *The Fourth ACL Linguistic Annotation Workshop*, pages 56–63, Uppsala, Sweden.

Kilian Foth. 2006. *Eine umfassende Constraint-Dependenz-Grammatik des Deutschen.* Universität Hamburg, Fachbereich Informatik. `http://edoc.sub.uni-hamburg.de/informatik/volltexte/2014/204/`.

Kilian Foth, Arne Köhn, Niels Beuck, and Wolfgang Menzel. 2014. Because size does matter: The Hamburg Dependency Treebank. In *Proceedings of the Language Resources and Evaluation Conference*, LREC'14, pages 2326–2333. European Language Resources Association (ELRA).

Stefan Grünewald, Annemarie Friedrich, and Jonas Kuhn. 2021. Applying Occam's razor to transformer-based dependency parsing: What works, what doesn't, and what is really necessary. arXiv eprint 2010.12699.

Gaël Guibon, Isabelle Tellier, Mathieu Constant, Sophie Prévost, and Kim Gerdes. 2014. Parsing Poorly Standardized Language Dependency on Old French. In *Thirteenth International Workshop on Treebanks and Linguistic Theories*, TLT13, pages 51–61, Tübingen, Germany.

Kristina Gulordava and Paola Merlo. 2015. Diachronic trends in word order freedom and dependency length in dependency-annotated corpora of Latin and Ancient Greek. In *Proceedings of the Third International Conference on Dependency Linguistics*, Depling 2015, pages 121–130.

Felix Hennig and Arne Köhn. 2017. Dependency tree transformation with tree transducers. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 58–66, Gothenburg, Sweden. Association for Computational Linguistics.

Tom S Juzek, Marie-Pauline Krielke, and Elke Teich. 2020. Exploring diachronic syntactic shifts with dependency length: the case of scientific English. In *Proceedings of the Fourth Workshop on Universal Dependencies (UDW 2020)*, pages 109–119, Barcelona, Spain (Online). Association for Computational Linguistics.

Alboukadel Kassambara. 2021. *rstatix: Pipe-Friendly Framework for Basic Statistical Tests*. R package version 0.7.0.

Terry Koo, Amir Globerson, Xavier Carreras Pérez, and Michael Collins. 2007. Structured prediction models via the matrix-tree theorem. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL, pages 141–150.

Marie-Pauline Krielke, Luigi Talamo, Mahmoud Fawzi, and Jörg Knappen. 2022. Tracing syntactic change in the scientific genre: Two Universal Dependency-parsed diachronic corpora of scientific English and German. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4808–4816, Marseille, France. European Language Resources Association.

Klaus Krippendorff. 2004. *Content Analysis*, 2 edition. Sage, Thousand Oaks, CA.

Anthony S. Kroch. 1989. Reflexes of grammar in patterns of language change. *Language Variation and Change*, 1(3):199–244.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. *Dependency Parsing*. Number 2 in Synthesis Lectures on Human Language Technologies. Morgan & Claypool.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomáš Kočiský, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. Mind the gap: Assessing temporal generalization in neural language models. In *Advances in Neural Information Processing Systems*.

Lei Lei and Ju Wen. 2020. Is dependency distance experiencing a process of minimization? A diachronic study based on the State of the Union addresses. *Lingua*, 239:102762.

Junying Liang, Yuanyuan Fang, Qianxi Lv, and Haitao Liu. 2017. Dependency distance differences across interpreting types: Implications for cognitive demand. *Frontiers in Psychology*, 8.

David Lightfoot and Marit Westergaard. 2007. Language acquisition and language change: Interrelationships. *Language and Linguistics Compass*, 1(5):396–415.

Xuezhe Ma and Eduard Hovy. 2017. Neural probabilistic model for non-projective MST parsing. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 59–69.

Christian Mair and Geoffrey N. Leech. 2020. Current changes in English syntax. In *The Handbook of English Linguistics*, chapter 14, pages 249–276. John Wiley & Sons, Ltd.

Jörg Meibauer, Ulrike Demske, Jochen GeilfuSS-Wolfgang, Jürgen Pafel, Karl Heinz Ramers, Monika Rothweiler, and Markus Steinbach. 2015. *Einführung in die germanistische Linguistik*, third, revised and expanded edition edition. Metzler, Stuttgart and Weimar.

Marie Mikulová, Milan Straka, Jan Štěpánek, Barbora Štěpánková, and Jan Hajic. 2022. Quality and efficiency of manual annotation: Pre-annotation bias. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2909–2918, Marseille, France. European Language Resources Association.

Thomas Mueller, Helmut Schmid, and Hinrich Schütze. 2013. Efficient higher-order CRFs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332, Seattle, Washington, USA. Association for Computational Linguistics.

Stefan Müller. 2007. The Grammix CD Rom. A software collection for developing typed feature structure grammars. In Tracy Holloway King and Emily M. Bender, editors, *Grammar Engineering across Frameworks 2007*, Studies in Computational Linguistics ONLINE, pages 259–266. CSLI Publications, Stanford.

R Core Team. 2022. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Gerold Schneider, Hans Martin Lehmann, and Peter Schneider. 2014. Parsing early and late modern English corpora. *Digital Scholarship in the Humanities*, 30(3):423–439.

Wolfgang Seeker and Jonas Kuhn. 2014. An out-of-domain test suite for dependency parsing of German. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, LREC'14, pages 4066–4073, Reykjavik, Iceland. European Language Resources Association (ELRA).

Fred R. Shapiro. 1998. A study in computer-assisted lexicology: Evidence on the emergence of *Hopefully* as a sentence adverb from the JSTOR journal archive and other electronic resources. *American Speech*, 73(3):279–296.

Achim Stein. 2014. Parsing heterogeneous corpora with a rich dependency grammar. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, LREC'14, pages 2879–2886, Reykjavik, Iceland.

Achim Stein. 2016. Old French dependency parsing: Results of two parsers analysed from a linguistic point of view. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, LREC'16, pages 707–713.

James K. Tauber. 2019. Character encoding of classical languages. In Monica Berti, editor, *Digital Classical Philology: Ancient Greek and Latin in the Digital Revolution*, number 10 in Age of Access? Grundfragen der Informationsgesellschaft, pages 137–157. De Gruyter Saur, Berlin and Boston.

David Temperley. 2007. Minimization of dependency length in written English. *Cognition*, 105(2):300–333.

Marc Tremblay and Hélène Vézina. 2000. New estimates of intergenerational time intervals for the calculation of age and origins of mutations. *American Journal of Human Genetics*, 66(2):651–658.

Tobias Walter, Celina Kirschner, Steffen Eger, Goran Glavavs, Anne Lauscher, and Simone Paolo Ponzetto. 2021. Diachronic analysis of German parliamentary proceedings: Ideological shifts through the lens of political biases. In *ACM/IEEE Joint Conference on Digital Libraries*, JCDL'21, pages 51–60.

Hadley Wickham, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, Alex Hayes, Lionel Henry, Jim Hester, Max Kuhn, Thomas Lin Pedersen, Evan Miller, Stephan Milton Bache, Kirill Müller, Jeroen Ooms, David Robinson, Dana Paige Seidel, Vitalie Spinu, Kohske Takahashi, Davis Vaughan, Claus Wilke, Kara Woo, and Hiroaki Yutani. 2019. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.

Amir Zeldes. 2017. The GUM corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

Yu Zhang, Zhenghua Li, and Min Zhang. 2020a. Efficient second-order TreeCRF for neural dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3295–3305. Association for Computational Linguistics.

Yu Zhang, Houquan Zhou, and Zhenghua Li. 2020b. Fast and accurate neural CRF constituency parsing. In *Proceedings of IJCAI*, pages 4046–4053.

10. Language Resource References

Abrami, Giuseppe and Bagci, Mevlüt and Hammerla, Leon and Mehler, Alexander. 2021. *German Parliamentary Corpus (GerParCor)*. Text Technology Lab, Goethe University Frankfurt.

Davies, Mark. 2010. *The Corpus of Historical American English (COHA)*. `https://www.english-corpora.org/coha/`.

Davies, Mark. 2015. *Hansard Corpus*. `https://www.english-corpora.org/coha/`.

Deutsches Textarchiv. 2007–2016. Berlin-Brandenburgischen Akademie der Wissenschaften (BBAW), `https://www.deutschestextarchiv.de`.

A. Appendix

Detailed agreement values per files. The table lists all possible combinations: agreement of individual annotators and the dependency parser (*a*1/*a*2 machine), inter-annotator agreement (iaa), and agreement between machine and both annotators (both machine). Agreement is assessed in terms of Krippendorff's alpha (Krippendorff, 2004) for labeled starting edges in two variants: including or excluding the target node of a dependency edge (the former is indicated by "end" in the table).

parliamentary tiger

file	a1 machine	a1 machine end	a2 machine	a2 machine end	iaa	iaa end	both machine	both machine end
1895	0.943	0.943	0.833	0.833	0.820	0.814	0.877	0.876
1918	0.927	0.927	0.878	0.878	0.876	0.872	0.908	0.907
1933	0.984	0.984	0.933	0.933	0.926	0.926	0.948	0.948
1942	0.955	0.955	0.917	0.917	0.910	0.901	0.928	0.928
bucket1	0.958	0.958	0.933	0.933	0.928	0.926	0.945	0.944
bucket5	0.967	0.967	0.947	0.947	0.932	0.932	0.953	0.952
bucket10	0.925	0.925	0.900	0.900	0.883	0.878	0.921	0.921
bucket15	0.955	0.955	0.893	0.893	0.887	0.874	0.913	0.913
bucket19	0.964	0.964	0.911	0.911	0.907	0.903	0.934	0.934

parliamentary ud

file	a1 machine	a1 machine end	a2 machine	a2 machine end	iaa	iaa end	both machine	both machine end
1895	0.971	0.971	0.923	0.923	0.919	0.919	0.944	0.944
1918	0.980	0.980	0.933	0.933	0.923	0.923	0.949	0.948
1933	0.956	0.956	0.947	0.947	0.933	0.933	0.955	0.955
1942	0.966	0.966	0.928	0.928	0.917	0.917	0.942	0.942
bucket1	0.971	0.971	0.962	0.962	0.952	0.950	0.965	0.965
bucket5	0.969	0.969	0.947	0.947	0.943	0.941	0.960	0.960
bucket10	0.949	0.949	0.957	0.957	0.938	0.938	0.957	0.957
bucket15	0.977	0.977	0.945	0.945	0.945	0.943	0.962	0.962
bucket19	0.985	0.985	0.951	0.951	0.951	0.951	0.966	0.966

hansard

file	a1 machine	a1 machine end	a2 machine	a2 machine end	iaa	iaa end	both machine	both machine end
1803	0.988	0.988	0.950	0.950	0.941	0.941	0.960	0.960
1820	0.970	0.970	0.990	0.990	0.966	0.966	0.975	0.975
1840	0.996	0.996	0.978	0.978	0.974	0.974	0.983	0.983
1860	0.985	0.985	0.972	0.972	0.964	0.964	0.976	0.976
1880	0.994	0.994	0.992	0.992	0.992	0.992	0.994	0.994
1900	0.983	0.983	0.965	0.965	0.962	0.959	0.971	0.971
1920	0.986	0.986	0.949	0.949	0.943	0.943	0.962	0.962
1940	0.995	0.995	0.975	0.975	0.975	0.975	0.983	0.983
1960	1.000	1.000	0.989	0.989	0.989	0.989	0.993	0.993
1980	0.980	0.980	0.988	0.988	0.980	0.980	0.984	0.984
2000	0.978	0.978	0.942	0.942	0.942	0.942	0.958	0.958

biofid

file	a1 machine	a1 machine end	a2 machine	a2 machine end	iaa	iaa end	both machine	both machine end
pre 1890 A	0.895	0.895	0.890	0.890	0.870	0.859	0.898	0.897
pre 1890 B	0.949	0.949	0.949	0.949	0.949	0.949	0.957	0.957
1890–1920 A	0.885	0.885	0.830	0.830	0.813	0.806	0.861	0.860
1890–1920 B	0.920	0.920	0.883	0.883	0.871	0.862	0.899	0.899
1920–1930 A	0.926	0.926	0.930	0.930	0.912	0.901	0.928	0.927
1920–1930 B	0.939	0.939	0.932	0.932	0.919	0.911	0.934	0.933
1930–1950 A	0.939	0.939	0.932	0.932	0.919	0.911	0.934	0.933
1930–1950 B	0.923	0.923	0.886	0.886	0.874	0.866	0.908	0.908
since 1950 A	0.953	0.953	0.940	0.940	0.938	0.936	0.949	0.949
since 1950 B	0.952	0.952	0.939	0.939	0.925	0.923	0.945	0.945

Table 7: Detailed agreement values for two annotators', a1 and a2, manual corrections of dependency annotations. Numbers are Krippendorff's alpha scores, and are assessed by ignoring the target node of a dependency edge or counting it in ("end").

data set	part year	mean new	sd new	mean old	sd old	stat	df	*p*
BIOfid	1930	88.800	3.180	84.500	3.660	1.930	8	0.090
DEUparl UD	1933	85.900	2.230	82.500	4.310	1.390	6	0.214
Hansard	1920	88.400	5.860	86.900	4.350	0.687	20	0.500
DEUparl Tiger	1933	88.700	2.750	88.200	1.990	0.210	2	0.853
Combined data sets		88.000	4.310	85.700	4.260	1.680	42	0.100

Table 8: Details of t-tests comparing parser performance on older and newer partitions of the data sets (Section 4). The partitioning year is given in column "part year".