Converting the TüBa-D/Z treebank of German to Universal Dependencies

Çağrı Çöltekin[1] **Ben Campbell**[2] **Erhard Hinrichs**[1] **Heike Telljohann**[1]
Department of Linguistics, University of Tübingen
[1] {cagri.coeltekin, erhard.hinrichs, heike.telljohann}@uni-tuebingen.de
[2] ben.campbell@student.uni-tuebingen.de

Abstract

This paper describes the conversion of TüBa-D/Z, one of the major German constituency treebanks, to Universal Dependencies. Besides the automatic conversion process, we describe manual annotation of a small part of the treebank based on the UD annotation scheme for the purposes of evaluating the automatic conversion. The automatic conversion shows fairly high agreement with the manual annotations.

1 Introduction

During the past decade, dependency annotations have become the primary means of syntactic annotation in treebanks. Compared to more traditional constituency annotations, the increasing popularity of dependency annotations has multiple reasons, including the easy interpretation of dependency annotations by non-experts, more successful applications of the dependency parses to NLP tools, faster parsing methods, and the community formed around successive dependency parsing shared tasks. In recent years, we have seen a surge of interest towards unified tagsets and annotation guidelines for various types of annotations found in (dependency) treebanks (Zeman, 2008; de Marneffe and Manning, 2008; Petrov et al., 2012; Zeman et al., 2012). The Universal Dependencies (UD) project (Nivre et al., 2016) is a large-scale community effort to build unified tagsets and annotation guidelines across many languages.

Despite the growing popularity of UD, and the growing number of new treebanks using the UD annotation scheme, many of the large treebanks with high-quality annotations are still constituency treebanks. Since Collins (1999), a well-known solution for obtaining high-quality dependency annotations is automatically converting the constituency annotations to dependency annotations, which includes some of the present UD treebanks which were converted from constituency treebanks or dependency treebanks with different annotation schemes. In this paper, we describe our efforts of automatically converting one of the major German treebanks, TüBa-D/Z (Hinrichs et al., 2004; Telljohann et al., 2004), to UD annotation scheme version 2.

German is one of the few languages with multiple large hand-annotated treebanks. Apart from the TüBa-D/Z, the TIGER treebank (Brants et al., 2002) is another large constituency treebank of German, as well as the NEGRA treebank (Skut et al., 1997). Another large German treebank is the Hamburg dependency treebank (HDT; Foth (2006), Foth et al. (2014)), which is natively annotated as a dependency treebank. The Universal Dependencies distribution also includes a German dependency treebank (UD German), which is based on the Google Universal Dependencies treebanks (McDonald et al., 2013), and converted to Universal Dependencies annotation scheme by the UD contributors.[1] The dependency and POS tag annotations in the UD German treebank were based on manual annotations, while other annotation layers, e.g., morphological features and lemmas, are automatically annotated. Besides being the smallest of the treebanks listed above, and despite continuous improvements over the previous UD versions, UD German does not yet seem to have the same level of annotation quality as the other German treebanks listed above. An overview of the treebanks with the indication of their sizes is presented in Table 1. In this study we focus only on conversion of TüBa-D/Z, but note that the present effort may be a precursor to obtaining a very large dependency treebank of German annotated uniformly using the UD scheme.

[1] `http://universaldependencies.org/`.

Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017), pages 27–37,
Gothenburg, Sweden, 22 May 2017.

The remainder of this paper is organized as follows: in the next section we provide a brief description of our source treebank, and review the earlier constituency-to-dependency conversion efforts of German treebanks. Section 3 describes the automatic conversion process. Section 4 describes the manual annotation of the evaluation set, and compares the automatic conversion with human annotations. We conclude in Section 5 after a brief discussion.

2 Background

TüBa-D/Z is a large German constituency treebank. We used version 10.00 of the treebank (released in August 2015) which comprises 95 595 sentences and 1 787 801 tokens of 3644 articles from the daily newspaper 'die tageszeitung' (taz). The treebank annotations include lemmas, POS tags, morphological features, syntactic constituency, and grammatical functions. For example, the grammatical function, or edge label, `HD` indicates the head of a phrase, while `OA` indicates the accusative object of the head predicate (Telljohann et al., 2015). The grammatical function labels are important for recovering dependencies in German. Since the language exhibits a relatively free word order, one cannot reliably predict the grammatical functions from the word order. The grammatical functions are also annotated in other German constituency treebanks TIGER and NEGRA. Also helpful for recovering dependencies, in TüBa-D/Z (unlike TIGER and NEGRA) phrases are annotated in a more detailed manner, e.g., noun phrases are not annotated as flat structures but with a constituency structure indicating their syntactic makeup (see Figure 1 for an example). Besides the morphosyntactic annotations that we are interested here, TüBa-D/Z also includes a rich set of linguistic annotations such as anaphora and coreference relations (Naumann, 2007), partial annotation of word senses (Henrich and Hinrichs, 2014) and named entity categories. Figure 1 presents an example tree from TüBa-D/Z.

Since its early releases, the TüBa-D/Z distribution also feature an automatically converted dependency version. The dependency conversion is based on Versley (2005), and uses the same dependency tagsets as the HDT annotations (Foth, 2006). However, similar to other automatic conversion efforts, the conversion is based on a set of heuristic 'head-finding rules', and there are some systematic differences from the HDT, such as default location of attachment of syntactically and semantically ambiguous prepositional phrases and adverbials. These differences are discussed by Versley (2005) in detail. The conversion tool by Versley (2005) was also used for converting TüBa-D/Z to the dependency treebank used in the Parsing German (PaGe) shared task (Kübler, 2008), where both TüBa-D/Z and TIGER treebanks were used in their original form, and as converted dependency treebanks.

treebank	type	sentences	tokens
TüBa-D/Z	const	95 595	1 787 801
TIGER	const	50 472	888 238
NEGRA	const	20 602	355 096
HDT	dep	206 794	3 823 762
UD German	dep	14 917	277 089

Table 1: An overview of large-scale (mostly) hand-annotated German treebanks. The number of sentences and tokens are from the latest versions of the treebanks as of this writing, namely, TüBa-D/Z version 10.0, TIGER version 2.2, NEGRA version 2, HDT version 1.0.1 (counting only the hand-annotated parts A and B), and UD German version 2.0.

There have been other constituency-to-dependency conversion efforts for German treebanks. Bohnet (2003) and Daum et al. (2004) present methods for converting NEGRA to a dependency treebank. Hajič et al. (2009) convert TIGER to a dependency treebank for use in the CoNLL-2009 multi-lingual dependency parsing shared task. The same conversion method is also used in Zeman et al. (2012), again in a multilingual setting, but also with an effort to unify the annotation scheme. In a more recent study, Seeker and Kuhn (2012) convert TIGER to a dependency treebank. They focus on representation of empty nodes in resulting dependency annotations.

In all of the earlier studies listed above, with the exception of Zeman et al. (2012), the target dependency treebanks share the tagsets for POS and morphological annotations, and to a large extent the dependency heads already annotated in the source treebank. However, in the present study the morphosyntactic annotations have to diverge, sometimes in non-trivial manner, from the source annotations. We will describe these differences in detail below.

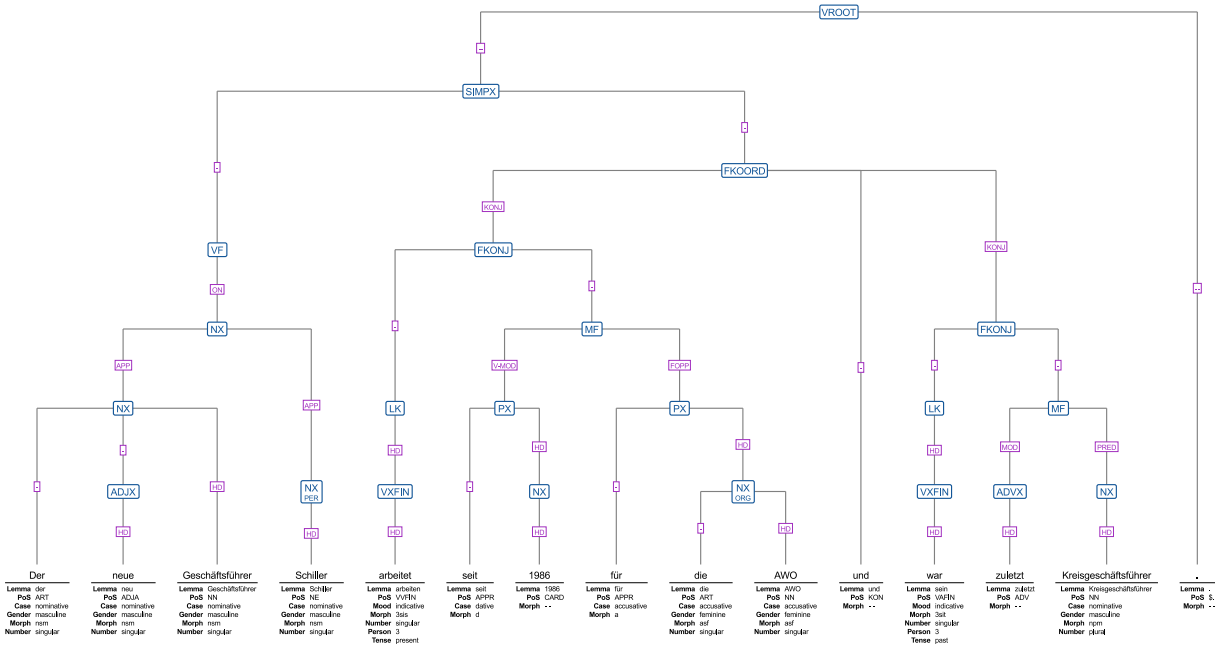

Figure 1: An example sentence from TüBa-D/Z.

3 Conversion process

3.1 POS tag conversion

TüBa-D/Z uses a version of STTS tag set (Schiller et al., 1995) for tagging parts of speech, with slight differences (Telljohann et al., 2015, p.21). We map the POS tags automatically as shown in Table 2. Since TüBa-D/Z POS tagset is more fine-grained than the UD POS tagset, most POS tags can trivially be mapped. However, some of the mappings deserve additional discussion.

In STTS, the tag `PWAV` covers adverbial interrogatives or relative pronouns, 'wh-words' such as *warum* 'why' or *wobei* 'wherein'. We mapped all words with STTS tag `PWAV` to UD tag `ADV`. However, some of these words may function as subordinating conjunctions (`SCONJ`), as in (1) below where *wobei* is tagged as `SCONJ` in UD German treebank. This STTS tag also has one of the highest rate of uncertainty with respect to the number of corresponding UD POS tags in the UD German treebank (see Table 5 in Appendix A).

(1) *Ab 1972 spielte er noch 121 mal für den*
From 1972 played he yet 121 times for the
Würzburger FV ***wobei*** *er 10 Tore erzielte.*
Würzburger FV where he 10 goals scored.

'From 1972 he played 121 times for the Würzburger FV *where* he scored 10 goals.'

Following the current UD practice, we split the preposition + determiner contractions, such as *zur (zu + der)* 'to the', into two syntactic tokens. These closed-class words are marked with STTS tag `APPRART` in the source treebank. These words are split only if the POS tag is `APPRART`.

Another interesting case concerns the STTS POS tag `TRUNC`. This tag is used for split words in coordination constructions, such as *Journalisten mit Fernseh- und Photokameras* 'Journalists with TV and photo cameras'. The well-known complexity and productivity of compounding in German results in quite frequent use of these constructions. In TüBa-D/Z version 10, the number of words with `TRUNC` tag is 1740. Most of the time, the words marked as `TRUNC` are nouns or adjectives. However, their forms often include remnants of the compounding process, and does not match with the exact form of the word's usage outside a compound. Since most of these structures contain nouns, we currently mark these words as `NOUN`, and add a special feature `Trunc=Yes` in the `MISC` field.

An alternative approach would be to mark the truncated part with the POS tag of the complete compound, and introduce a syntactic word similar to the tokenization of contracted forms discussed above. In the alternative annotation, the syntactic tokens of the example phrase '*Journalisten mit Fernseh- und Photokameras*' would be '*Journalisten mit Fernseh**kameras** und Photokameras*'. This can easily be represented in the CoNLL-U file for-

TüBa-D/Z	UD	TüBa-D/Z	UD
ADJA	ADJ	PRF	PRON
ADJD	ADJ	PROP	ADV
ADV	ADV	PTKA	ADV
APPO	ADP	PTKANT	INTJ
APPR	ADP	PTKNEG	PART
APPRART	ADP, DET	PTKVZ	ADP
APZR	ADP	PTKZU	PART
ART	DET	PWAT	DET
CARD	NUM	PWAV	ADV
FM	X	PWS	PRON
ITJ	INTJ	TRUNC	NOUN
KOKOM	ADP	VAFIN	AUX
KON	CCONJ	VAIMP	AUX
KOUI	SCONJ	VAINF	AUX
KOUS	SCONJ	VAPP	AUX
NE	PROPN	VMFIN	AUX
NN	NOUN	VMINF	AUX
PDAT	DET	VMPP	AUX
PDS	PRON	VVFIN	VERB
PIAT	DET	VVIMP	VERB
PIDAT	ADJ	VVINF	VERB
PIS	PRON	VVIZU	VERB
PPER	PRON	VVPP	VERB
PPOSAT	PRON	XY	X
PPOSS	PRON	$,	PUNCT
PRELAT	DET	$.	PUNCT
PRELS	PRON	$(	PUNCT

Table 2: POS conversion table.

mat by utilizing the range records used for multiword tokens, but specifying a span of only a single surface token. Listing 1 presents an example CoNLL-U fragment demonstrating the alternative coding for TRUNC. This is relatively straightforward to include in TüBa-D/Z conversion since the treebank encodes most of the relevant information in the lemma of the truncated word. Although getting the alternative annotation correct is not as straightforward for automated annotators, e.g., parsers, there are successful tools for German compound splitting (Ma et al., 2016, for example) that can be used for this purpose.

3.2 Morphology

TüBa-D/Z annotates each word with a lemma, and nouns, adjectives, determiners and verbs are also annotated for morphological features. Nouns, adjectives and determiners are marked for *number*, *gender* and *case*, and verbs are marked for *tense* and *mood*. These TüBa-D/Z morphological features map to the UD morphological features in a straightforward manner. We also assign values to the UD features PronType, VerbType, NumType, Poss, Reflex, Foreign, Definite, Voice, and Polarity based on the STTS tags, lemmas (either information coded explicitly, as in sollen%aux, or forms of the closed-class words).

```
1 Journalisten    _ NOUN  NN
2 mit             _ ADP   APPR
3-3 Fernseh-      _ _     _
3 Fernsehkameras  _ NOUN  TRUNC
4 und             _ CCONJ KON
5 Photokameras    _ NOUN  NN
```

Listing 1: An example of the alternative prposal for STTS TRUNC tag (only the relevant columns are included).

Lemmas are also mapped to the UD version of the treebank with minor modifications. In the current conversion, we strip all the additional information specified on the TüBa-D/Z lemmas, such as grammatical function marking like sollen%aux, the only exception is the separable verb information as in ein#setzen. Reflexive pronouns are always marked as #refl in TüBa-D/Z. Following UD German, we map #refl to the lemma used for the personal pronoun with the same number and person features (e.g., lemma of *mich* 'myself' gets the lemma *ich* 'I') and mark the reflexiveness of the pronoun with the Reflex morphological feature. Finally, we map the ambiguous lemmas as is.

3.3 Extracting dependencies

As in earlier examples in the literature, the dependency extraction uses a set of heuristic 'head-finding rules'. The conversion software first pre-processes constituency trees, since some of the information is easy to extract from the original constituency annotations. Then, the pre-processed trees are converted to UD-like dependencies with a set of head-finding rules. And finally, a post-processing stage, including operations like attaching punctuation to the right parent according to the UD guidelines, makes sure that the dependency trees are UD version 2 compliant. The UD dependency types (including the subtypes) we use, and their counts in the converted treebank are

dep. type	count	percentage
acl	5643	0.328
acl:relcl	14906	0.867
advcl	16495	0.959
advmod	115480	6.715
advmod:neg	12250	0.712
amod	108922	6.334
appos	32246	1.875
aux	42994	2.500
aux:pass	12213	0.710
case	166418	9.677
cc	48115	2.798
ccomp	9174	0.533
compound:prt	9199	0.535
conj	64525	3.752
cop	23109	1.344
csubj	3396	0.197
csubj:pass	326	0.019
dep	37	0.002
det	224248	13.040
det:neg	3418	0.199
discourse	206	0.012
expl	1899	0.110
fixed	326	0.019
flat	20937	1.217
flat:foreign	3677	0.214
iobj	4911	0.286
mark	32575	1.894
nmod	50142	2.916
nmod:poss	55783	3.244
nsubj	126182	7.337
nsubj:pass	10485	0.610
nummod	14330	0.833
obj	74650	4.341
obl	115096	6.693
parataxis	20251	1.178
punct	262109	15.242
xcomp	13029	0.758

Table 3: Number and percentage of dependencies in the converted TüBa-D/Z treebank.

listed in Table 3. In the resulting dependency treebank 4.90 % of the dependencies are crossing dependencies, and analyses of 20.92 % of the sentences contain at least one crossing dependency. The conversion based on Versley (2005) yields 2.33 % crossing dependencies and 20.17 % sentences with crossing dependencies. In this section, we discuss some of the interesting or difficult-to-convert structures, rather than the implementation details. The source code of the conversion software along with detailed documentation is released on GitHub.[2]

TüBa-D/Z (and other German constituency treebanks noted earlier) marks the heads of the phrases explicitly. As a result, finding heads of the words is trivial for most cases. The main difficulty arises because, unlike the earlier constituency-to-dependency conversion efforts (Versley, 2005, for example), conversion to UD requires changing many of the head choices in the original treebank. This is also apparent in Figure 2, where we present dependency representations of the example tree in Figure 1. For example, the head of the phrase *war zuletzt Kreisgeschäftsführer* is the copula *war* in both the TüBa-D/Z and Versley's conversion. However, in the UD conversion, the head needs to be the subject complement *Kreisgeschäftsführer*. Note that this also interacts with the coordination in this example. The head of the first conjunct, the verbal predicate *arbeitet*, is coordinated with the head of the second conjunct which becomes the noun *Kreisgeschäftsführer* after re-structuring. The other re-assigning requirements include finite auxiliaries and subordination markers which are marked as heads in TüBa-D/Z but should be dependents in UD.

Another difficulty in choosing heads arise because TüBa-D/Z does not mark heads in some grammatical structures, such as in coordinated constituents. In some structures, choice of heads are non-trivial. As noted above, head assignment in coordination also interacts with the change of head direction, for example, between the finite auxiliary verbs and the content verbs (as in 2b).

One more issue that deserves a brief note here is the ambiguities in the conversion process. Although TüBa-D/Z annotations are more detailed than the basic UD annotations in general, for some grammatical relations, a single grammatical function may be mapped to more than one dependency label or structure. For example, TüBa-D/Z functional label `APP` (apposition) may map to `appos`, `flat` or `compound`. For `APP`, we use `flat` if the noun phrase corresponds to a named entity, otherwise we use the UD dependency `appos`. Another source of head-assignment ambiguity concerns multiply-rooted sentences which are described as side-by-side or run-on sentences in the UD specification. In TüBa-D/Z these sentences are annotated under two separate trees, and connected to a 'virtual root' node without any functional relation assigned. An example constituency tree of multiply-rooted sentence is given in Figure 3. We connect these sentences using `parataxis` relation during the UD conversion, always marking the first one as the head. However, the UD specification allows marking the 'more prominent' sentence as

[2]The source code of the converter is available at `https://github.com/bencampbell30/TuebaUdConverter`.

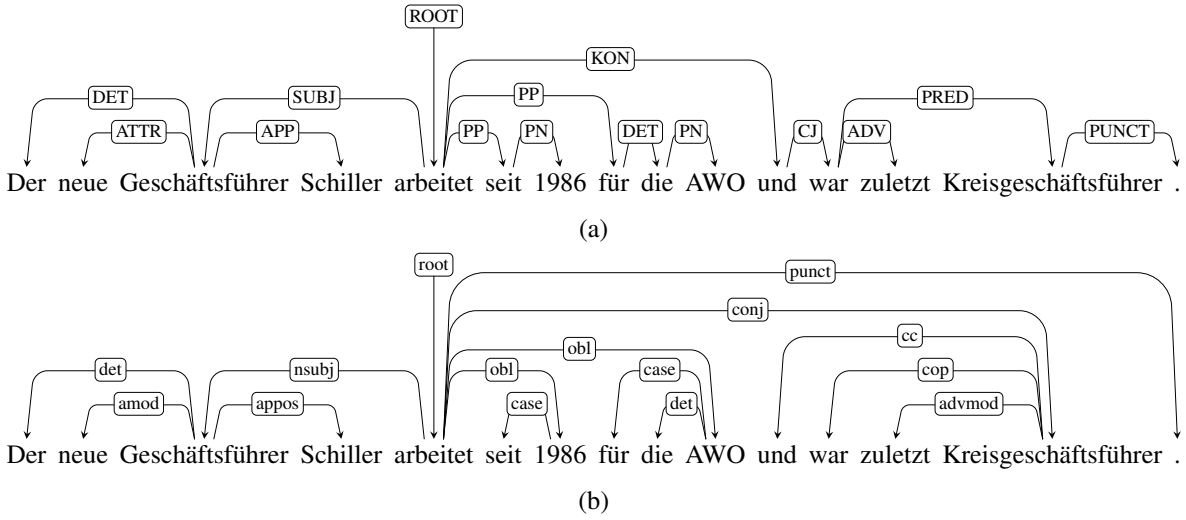

Figure 2: The dependency trees for the sentence in Figure 1, automatically converted using tools from (a) Versley (2005) and (b) this study.

the head of a parataxis relation. Since this information is not available in the TüBa-D/Z annotations, the head of the parataxis relation is expected to be wrong in some cases.

The head-finding rules fail to assign the head and the dependency relation for 37 dependencies in 35 sentences (out of 95 595) in the TüBa-D/Z. In these cases, we attach the token to the highest node that preserves projectivity, and use the dependency label `dep`. A close examination of the sentences show that most of these cases (22 of 35 sentences) involve either errors in the TüBa-D/Z annotation, or in the original sentence (such as an unintelligible sequence of letters within the sentence). The remaining 13 cases are unusual constructions that the heuristic head-finding rules do not address currently.

3.4 Topological fields

As noted in Section 2, TüBa-D/Z includes some annotations along with the morphosyntactic structure. Among these is the topological field information. Traditionally, topological field information is used to account for word order of different clause types (Höhle, 1986). All clauses in TüBa-D/Z contain nodes that are labeled `LK` (left bracket), `RK` (right bracket), `MF` (middle field), and optionally `VF` (initial field) and `NF` (final field). Besides being instrumental in linguistic description, the topological field annotation has been shown to improve the accuracy of both constituency (Kübler et al., 2008) and dependency (de Kok and Hinrichs, 2016) parsers.

Similar to de Kok and Hinrichs (2016), we mark the topological field information at the token level. We include a special feature label `TopoField` in the `MISC` field of the CoNLL-U file with a variable length sequence of topological field labels listed above. Unlike de Kok and Hinrichs (2016) who marked tokens only with a single (most specific) topological field label, this representation allows recovering the topological field of the token within all parent clauses. For example, `TopoField=VF-NF-MF` indicates that the token is within the middle field (MF) of the most-specific clause, which is within the final field (NF) of another sub-clause which, in turn, is within the initial field (VF) of the main clause. The maximum depth of the recursion goes up to 9 clauses, but most (62.31 %) of the tokens are direct descendants of the main clause, and the tokens within a hierarchy of clauses up to depth three cover more than 95 % of the tokens in TüBa-D/Z.[3]

4 Evaluation

To evaluate the accuracy of the automatic conversion, we annotated a selection of 200 sentences from TüBa-D/Z. About half of the sentences (116) were selected manually to cover a wide range of syntactic constructions, while the remaining sentences are randomly sampled. In total, the selection includes 3134 tokens. The overall average tokens per sentence is 15.67. However, the hand-

[3]Following TüBa-D/Z style book (Telljohann et al., 2015, p. 25), we also include nodes corresponding to coordinated phrases, e.g., `FKOORD`, `FKONJ`, as `TopoField` labels.

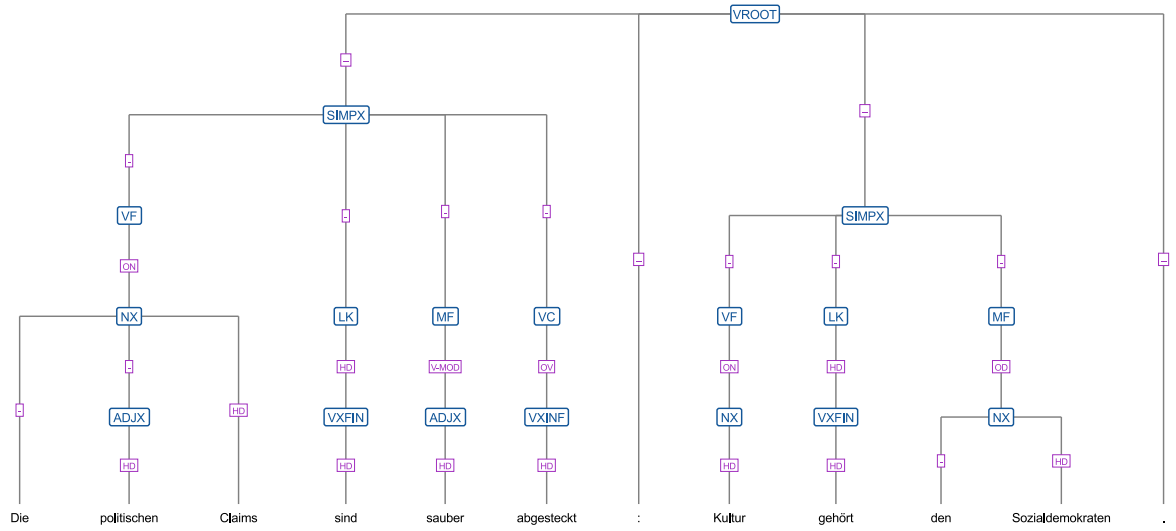

Figure 3: An example multiply-rooted sentence from TüBa-D/Z.

picked sentences are shorter on average (12.05) than the randomly sampled sentences (20.67). We used WebAnno (Eckart de Castilho et al., 2014) for the annotation process.

Only the dependency relations were annotated manually. In manual annotations, we did not use sub-types of the UD relations. The additional information required for all sub-types we use (Table 3) is unambiguously available in the original TüBa-D/Z annotations.

We compare the gold-standard annotations from the human annotators with the automatic conversion on the 200 sentences described above. The labeled and unlabeled attachment agreements on 3134 tokens are 83.55 % and 87.13 % respectively. The agreement values are slightly better for hand-picked linguistic examples, which are also shorter.

A closer look at the disagreements reveal only a few general tendencies. For the attachment disagreements, the annotators seem to be less consistent in attaching punctuation. All cases of punctuation disagreements are annotator mistakes. If we disregard punctuation, the agreement values increase by about three percent. Other head-assignment errors do not follow a clear pattern. The ones we inspected manually correspond to either ambiguous cases like prepositional phrase or adverbial attachment, or annotator errors.

As noted in Section 3, the head of `parataxis` relation is ambiguous. As a result, the direction of `parataxis` relation often disagrees between the automatic conversion and the manual annotations. For example, for the sentence in Figure 3, the human annotator marked the head of the second sentence as the head of the dependency tree, while automatic annotation picks the first one. This is also visible in Table 4, where `parataxis` and `root` labels seem to be confused rather frequently.

The confusions between dependency types are presented in Table 4. The most common mismatch in label assignment occurs with the `appos` dependency. As noted earlier, the automatic conversion assigns the label `appos` in all non-head-marked dependencies between two noun phrases. A large number of dependencies that are labeled as `appos` by the automatic conversion are labeled `flat` or `nmod` by the human annotators. Besides the attachment ambiguity discussed above, `parataxis` is another frequently confused label, which is often marked as `list` by the human annotators. This is one of the cases where TüBa-D/Z annotations do allow distinguishing between two dependency relations (in this case, UD `parataxis` and `list` dependencies). Other notable label confusions in Table 4 include `nmod` and `obl` which is often an annotator error, and `expl` and `nsubj` which is often a difficult annotation decision.

In general, we found the cross-tabulation in Table 4 useful. Besides revealing some of the inherent ambiguities for the conversion process, we discovered some of the converter errors, and some of the errors in the original TüBa-D/Z annotations. The remaining items indicate annotation errors, some of which are indications of difficult annotation decisions (such as punctuation attachment) for manual annotation with the UD scheme.

	acl	advcl	advmod	amod	appos	aux	case	cc	ccomp	compound	conj	cop	csubj	det	expl	fixed	flat	iobj	mark	nmod	nsubj	nummod	obj	obl	parataxis	punct	root	xcomp
acl	24										1	1																2
advcl	2	20	2	1							1																	2
advmod			196	8	2						1									3			2	6	2		3	3
amod				180																								
appos					17						2									1				1	3			
aux	1	1				102					1																2	
case			1				245												1									4
cc								72																	1			
ccomp		1							17				1															3
compound					1					13	1																	
conj	1										84									2				2	1	1		1
cop						1						37													1			
csubj													9														1	
dep														1											2			
det									1					357						1	3		1					
discourse																									2		1	
expl															5						3		1	1				
fixed			2				2									1												
flat			1		17												42								2			
goeswith												1																
iobj																		7					1					
list					1																	1			5			
mark		1					2	2											54					3				
nmod				1	13									1						140	8			9				
nsubj					2									1						1	231		3				3	
nummod				1	2						1										1	19		1	1			
obj											1												118	1	2		1	
obl			1																	13				168			1	1
orphan			1								1														2			
parataxis	2	1							2		3	2												1	15		11	
punct							1																			507		
root	1	1			2	1			2			3									2		1	1	7		177	2
vocative																							1		2			
xcomp	1	1				1																		1				13

Table 4: Label agreement between automatic conversion and manual annotation. The row labels are the labels assigned by the human annotator. The columns correspond to the labels assigned by the automatic conversion. The automatic conversion does not use `vocative`, `list`, and `goeswith` relations, it did not find any `discourse` relation according to its head-finding rules, `dep` relation was not used since the head-finding heuristics did not fail in this set.

5 Summary and outlook

Automatic conversion of high-quality constituency treebanks to dependency treebanks allow us to make use of earlier high-quality treebanks with new tools and techniques that require dependency annotations. In this paper we describe our efforts of automatic conversion of TüBa-D/Z to Universal Dependencies. We also describe a small-scale annotation project, where we manually annotated 200 sentences from TüBa-D/Z using UD dependency relations. The automatic conversion is based on traditional head-finding heuristics, and agrees well with the manual annotations. The unlabeled and labeled attachment scores are 88.55 % and 87.13 % respectively. Considering that some of these errors are manual annotation errors, the agreement indicates that the result is a high-quality treebank.

The detailed analysis of disagreements between the manual annotations and automatic conversion were primarily motivated for pinpointing the mistakes in automatic conversion in order to refine the heuristic conversion rules. However, the analysis and documentation of the disagreements are also important for a number of other reasons. First, it provides the users of the converted treebank an indication of quality of the annotations for their particular purpose. Second, knowing the disagreements may also improve the accuracy and speed of a manual correction after the automatic conversion. Third, the disagreements also reveal common annotator errors, informing the designers of the target annotation scheme and future annotation project about the difficult cases of annotation. Finally, some cases where conversion heuristics fail and/or disagreements occur indicate annotation errors in the source treebank, which is a valuable feedback for improving the source treebank.

In this study, we used a one-to-one 'best-effort' mapping of the POS tags. In future work, we plan to improve the POS conversion by utilizing syntactic structure. Furthermore, our focus in this study was only on TüBa-D/Z, however, the work can be extended to cover other German treebanks. The result would be a very large, high-quality, uniformly-annotated dependency treebank of approximately 400 000 sentences.

References

Bernd Bohnet. 2003. Mapping phrase structures to dependency structures in the case of (partially) free word order languages. In *Proceedings of the first international conference on Meaning-Text Theory*, pages 217–216.

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.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Michael Daum, Kilian A Foth, and Wolfgang Menzel. 2004. Automatic transformation of phrase treebanks to dependency trees. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'02)*, Lisbon, Portugal.

Daniël de Kok and Erhard Hinrichs. 2016. Transition-based dependency parsing with topological fields. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–7, Berlin, Germany, August. Association for Computational Linguistics.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The Stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8, Manchester, UK.

Richard Eckart de Castilho, Chris Biemann, Iryna Gurevych, and Seid Muhie Yimam. 2014. Webanno: a flexible, web-based annotation tool for clarin. In *Proceedings of the CLARIN Annual Conference (CAC) 2014*, page online, Utrecht, Netherlands. CLARIN ERIC. Extended abstract.

Kilian A. Foth, Arne Köhn, Niels Beuck, and Wolfgang Menzel. 2014. Because size does matter: The hamburg dependency treebank. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Kilian A. Foth. 2006. Eine umfassende constraint-dependenz-grammatik des deutschen. Technical Report 54.75, University of Hamburg.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pages 1–18, Boulder, Colorado. Association for Computational Linguistics.

Verena Henrich and Erhard Hinrichs. 2014. Consistency of manual sense annotation and integration into the tüba-d/z treebank. In *Proceedings of The 13th Workshop on Treebanks and Linguistic Theories (TLT13)*, pages 62–74.

Erhard Hinrichs, Sandra Kübler, Karin Naumann, Heike Telljohann, and Julia Trushkina. 2004. Recent developments in linguistic annotations of the TüBa-D/Z treebank. In *Proceedings of the Third Workshop on Treebanks and Linguistic Theories*, pages 51–62.

Tilman Höhle. 1986. Der begriff "mittelfeld", anmerkungen über die theorie der topologischen felder. pages 329–340.

Sandra Kübler. 2008. The page 2008 shared task on parsing german. In *Proceedings of the Workshop on Parsing German*, PaGe '08, pages 55–63, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sandra Kübler, Wolfgang Maier, Ines Rehbein, and Yannick Versley. 2008. How to compare treebanks. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 2322–2329, Marrakech, Morocco.

Jianqiang Ma, Verena Henrich, and Erhard Hinrichs. 2016. Letter sequence labeling for compound splitting. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 76–81.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria, August. Association for Computational Linguistics.

Karin Naumann. 2007. Manual for the annotation of in-document referential relations. Technical report, University of Tübingen.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 23–28.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey.

Anne Schiller, Simone Teufel, and Christine Thielen. 1995. Guidelines für das tagging deutscher textcorpora mit STTS. Technical report, Universities of Stuttgart and Tübingen.

Helmut Schmid. 1994. "probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, page 154.

Wolfgang Seeker and Jonas Kuhn. 2012. Making ellipses explicit in dependency conversion for a german treebank. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Wojciech Skut, Thorsten Brants, Brigitte Krenn, and Hans Uszkoreit. 1997. Annotating unrestricted german text. In *Fachtagung der Sektion Computerlinguistik der Deutschen Gesellschaft fr Sprachwissenschaft*.

Heike Telljohann, Erhard Hinrichs, and Sandra Kübler. 2004. The TüBa-D/Z treebank: Annotating German with a context-free backbone. In *In Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 2229–2232.

Heike Telljohann, Erhard Hinrichs, Heike Zinsmeister, and Kathrin Beck. 2015. Stylebook for the tübingen treebank of written German (TüBa-D/Z). Technical report, University of Tübingen, Seminar für Sprachwissenschaft.

Julia Trushkina and Erhard Hinrichs. 2004. A hybrid model for morpho-syntactic annotation of german with a large tagset. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 238–245, Barcelona, Spain, July. Association for Computational Linguistics.

Yannick Versley. 2005. Parser evaluation across text types. In *Proceedings of the Fourth Workshop on Treebanks and Linguistic Theories*.

Daniel Zeman, David Mareček, Martin Popel, Loganathan Ramasamy, Jan Štěpánek, Zdeněk Žabokrtský, and Jan Hajič. 2012. Hamledt: To parse or not to parse? In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Daniel Zeman. 2008. Reusable tagset conversion using tagset drivers. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

A STTS to UD POS tag mapping in UD German version 2.0 treebank

	ADJ	ADP	ADV	AUX	CCONJ	DET	NOUN	NUM	PART	PRON	PROPN	PUNCT	SCONJ	VERB	X
XY	0	2	0	0	0	0	39	5	0	0	5	5	0	0	21
VVIMP	1	0	1	0	1	0	16	0	0	0	19	0	0	27	1
APPRART	0	10	0	0	0	0	1	0	0	0	2	0	0	0	7
APZR	0	19	13	0	0	0	0	0	8	0	0	0	0	0	0
ITJ	0	0	1	0	0	0	0	0	1	0	6	0	0	0	1
PTKANT	0	0	1	0	0	0	2	0	4	0	0	0	0	0	0
PTKA	1	12	34	0	0	0	0	0	4	0	0	0	0	0	0
PWAV	0	0	190	0	4	0	0	0	0	26	0	0	60	0	0
APPO	1	29	7	0	0	0	2	0	0	0	0	0	0	1	0
ADJD	4898	12	1161	0	2	0	100	19	3	0	283	3	0	132	16
PTKVZ	28	912	623	0	0	0	4	0	17	0	5	0	0	4	0
TRUNC	2	0	0	0	0	0	5	0	0	0	19	0	0	0	0
VAFIN	0	0	0	4241	1	0	7	0	0	0	14	1	0	4645	9
VAPP	0	0	0	115	0	0	0	0	0	0	0	0	0	68	0
PIAT	265	0	20	0	0	19	5	0	1	1315	10	0	0	4	0
KOUS	0	139	55	0	56	0	0	0	0	0	3	0	1441	0	2
PPOSAT	3	0	0	0	0	1841	2	0	0	468	13	0	0	3	0
VAINF	0	0	0	462	0	0	0	0	0	0	0	0	0	151	0
KOKOM	0	1511	42	0	187	0	0	0	2	0	1	0	36	0	0
NN	264	19	31	8	2	0	48880	24	2	0	11891	27	1	34	84
PDAT	44	0	0	0	0	113	0	0	0	991	1	0	0	0	0
ADJA	13597	8	21	0	0	0	276	23	2	0	1431	5	3	107	22
PRELAT	0	0	0	0	0	67	0	0	0	12	0	0	0	0	0
PAV	3	5	1136	1	5	0	0	0	0	133	0	0	3	0	0
VVPP	715	1	23	0	0	0	4	0	0	0	3	0	0	4936	0
VAIMP	0	0	0	6	0	0	0	0	0	0	0	0	0	1	0
PDS	0	0	0	0	0	53	1	0	0	470	1	0	5	0	0
PIS	28	0	19	0	0	0	30	3	0	1103	7	0	0	1	0
KON	0	546	73	0	8079	0	2	0	0	0	89	0	36	0	3
PWS	0	0	4	0	0	0	1	0	0	79	2	0	0	0	0
FM	1	4	0	0	3	0	7	1	0	0	420	0	0	0	13
KOUI	0	216	1	0	3	0	1	0	0	0	1	0	8	0	0
ADV	237	36	10075	0	119	0	44	0	8	1	31	0	33	10	49
PRELS	0	0	0	0	0	112	1	0	0	1779	3	0	6	2	1
VVFIN	110	4	14	9	7	0	62	1	4	0	243	0	9	10702	31
NE	56	14	21	2	1	0	622	19	2	0	15721	0	0	12	55
CARD	3	0	0	0	0	0	23	7204	0	0	363	0	0	0	0
VMFIN	0	0	0	1445	0	0	4	0	0	0	3	0	0	41	0
APPR	30	27264	118	0	3	0	16	0	22	0	400	0	12	1	7
VMINF	0	0	0	82	0	0	0	0	0	0	0	0	0	2	0
ART	0	0	1	1	0	33063	3	68	0	252	256	0	5	0	1
VVINF	10	0	3	1	0	0	10	0	0	0	7	0	0	2439	0
PPER	0	0	0	1	1	0	6	0	4	5655	57	1	0	0	4
PTKZU	0	5	0	1	0	0	0	0	927	0	1	0	0	0	0
PTKNEG	0	0	2	0	0	0	1	0	978	0	2	0	0	0	0
$(	0	4	0	0	0	0	10	0	0	0	2	10160	0	0	28
VVIZU	0	0	0	0	0	0	0	0	0	0	1	0	0	240	0
PRF	0	0	1	0	0	0	0	0	0	1648	1	0	0	0	0
$.	0	0	0	0	0	0	1	0	0	0	4	15389	0	0	1
PWAT	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0
$,	0	0	0	0	0	0	0	0	0	0	0	11126	0	0	0

Table 5: The cross-tabulation of STTS and UD POS tag sets in UD version 2.0 treebank. The rows are sorted by entropy. The UD POS tags converted from the manual annotations of Google Universal dependencies treebanks. The (language-specific) STTS tags are automatically added using TreeTagger (Schmid, 1994). The table shows that, despite the fact that UD POS tags are more coarse in comparison to STTS, mapping from STTS tags to UD is not always straightforward. Correct conversion of the POS tags often require paying attention to syntactic structure, which has been successful in earlier similar studies (Trushkina and Hinrichs, 2004).