Abstract

We introduce the first German treebank for Twitter microtext, annotated within the framework of Universal Dependencies. The new treebank includes over 12,000 tokens from over 500 tweets, independently annotated by two human coders. In the paper, we describe the data selection and annotation process and present baseline parsing results for the new testsuite.

1 Introduction

Recent years have seen an increasing interest in developing robust NLP applications for data from different language varieties and domains. The Universal Dependencies (UD) project (Nivre et al., 2016) has inspired the creation of many new datasets for dependency parsing in a multilingual setting. Treebanks have been created for low-resourced languages such as Bambara, Erzya, or Kurmanji as well as for many new domains, genres and language varieties for which no annotated data was yet available. A case in point are web genres, spoken discourse, literary prose, historical data or data from social media.\footnote{The different treebanks and their description are available from: \url{https://universaldependencies.org/}.}

We contribute to the creation of new resources for different language varieties and introduce tweeDe, a new German UD Twitter treebank. TweeDe has a size of over 12,000 tokens, annotated with PoS, morphological features and syntactic dependencies. TweeDe is different from existing German UD treebanks as its content focuses on private communication. Private tweets share many properties of spoken language. They are often highly informal and not carefully edited, often lack punctuation and can include ungrammatical structures. In addition, the data often includes spelling errors and a creative use of language that results in a high number of unknown words. These properties make user-generated microtext a challenging test case for parser evaluation.

In the paper, we describe the creation of tweeDe, including data selection, preprocessing and the annotation process. We report inter-annotator agreement for the syntactic annotations (§2) and discuss some of the decisions that we have made during annotation (§3). We compare tweeDe to other treebanks in §4. In §5 we present baseline parsing results for the new treebank. Finally, we put our work into context (§6) and outline avenues for future work (§7).

2 tweeDe – A German Twitter treebank

This section describes the creation of the first German Twitter treebank, annotated with Universal Dependencies. The treebank includes 519 tweets with over 12,000 tokens of microtext.

2.1 Data extraction

The annotation of user-generated microtext is a challenging task, due to the brevity of the messages and the missing context information, which often results in highly ambiguous texts. As a result, inter-annotator agreement (IAA) is often below the one obtained on standard newspaper text. To avoid such problems, we opted to extract short communication threads, which range in length from 2 up to 34 tweets. This approach allowed the annotators to see the context of each tweet and was thus crucial for resolving ambiguities in the data.
The conversations were collected in two steps. We first used an existing python tool\(^2\) that supports the downloading of conversations by querying the Twitter API for a set of query terms and then scraping the html page on twitter.com that represents each matching conversation. However, Twitter does not embed complete json files into the html-pages and the existing crawler had some problems in fully retrieving tweet text containing certain special characters. We therefore used the output of the initial crawler only to establish the ids and the sequencing of the tweets in a conversation and then re-downloaded the full json files to be sure we had complete tweets.

The query terms we used were all German stop words, i.e. highly-frequent closed-class function words such as prepositions, articles, modal verbs, and adverbs such as *auch* ‘too’ or *dann* ‘then’. The idea behind this was to avoid any kind of topic bias. Of the threads retrieved, we only retained those representing private communication between two or more participants. Threads consisting mainly of automatically generated tweets, advertisements, and so on were discarded after manual inspection. The treebank preserves the temporal order of the tweets in the same thread. For meta-information, we keep the tweet id, date and time as well as the author’s user name. As is common practise for UD treebanks, we also store the raw, untokenised text for each tweet.

Besides issues arising from brevity, further problems for annotating user-generated social media content are the creative use of language, including acronyms (example 1) and emoticons (example 2), non-canonical spellings (example 3), missing arguments (example 2) and the often missing or inconsistent use of punctuation (examples 1-4). The latter causes segmentation problems like those faced in annotating spoken language where, since no punctuation is given, the annotator has to decide on where to insert sentence boundaries.

\(^2\)https://github.com/song9446/twitter-corpus-crawler-python

\(1\) hdl have you dear
\(2\) Mache deshalb gerne mal mit < 3 participate thus gladly MODAL PTCL VERB PTCL EMOTICON
“Love you” “Hence (I) like to participate once in a while < 3”

\(3\) Is nich wahr ich habe nur einen report bekommen das sie es erhalten haben und überprüfen..<is not true I have only a report got that they it received have and check..<
“It’s not true. I only got a report that they have received it and will check it.”

\(4\) Mahlzeit Arbeit Gassigang Wohnung geputzt Essen gemacht Jaaaa es ist #Freitag und jetzt meal work walking the dog flat cleaned food made Yeeees it is Friday and now
#hochdiehaendewochenende #up-the-hands-weekend

\(5\) @surfguard @Mathias59351078 @ArioMirzaie Über einige amüsiere ich mich köstlich, bei manchen denke ich “hm” und bei wieder anderen bin ich entsetzt. Mit keinem einzigen hab ich irgendwas zu tun. Wenn du mich wegen meiner Hautfarbe den Schuldigen zuordnest, bist du ein Rassist.

“@surfguard @Mathias59351078 @ArioMirzaie Some make me laugh, some make me think "hm" and still others make me feel appalled. I don’t have anything to do with any of them. If you blame me for the color of my skin, you’re a racist.”

- Hashtags and URLs at the beginning or end of the tweet that are not syntactically integrated in the sentence are separated and form their own unit (tree).
- Emoticons are treated as non-verbal comments to the text and are integrated in the tree (figure 1).
• Interjections (Aaahh), inflectives (*grins*), fillers (ähm) and acronyms typical for social media content (lol, OMG) are also not separated but considered to be part of the tree (figure 1).

2.3 Tokenisation
User-generated text often reflects (or mimics) morpho-phonological processes from spoken language that are in conflict with the rules of Standard German orthography. One example are words merged into one token that, according to German grammar, should be separated but in spoken varieties of German are contracted into one token. We split merged tokens to avoid having tokens with more than one PoS tag and grammatical function. To mark that the word has been written as one atomic token, we use the UD feature SpaceAfter=No in combination with CorrectSpaceAfter=Yes in the last column of the CoNLL-UD file. Figure 2 (left) shows an example where the canonical token sequence “Kennst Du ?” is instead fused into the single token “Kennste ?”.

We also observe the opposite case where tokens that should have been written as one word are split into two or more separate tokens in the tweet. Most of these are German noun compounds. We chose to annotate split compounds using the UD relation goeswith. We follow UD conventions to always annotate the first component as the head and attach all remaining components to the first component. One problem with this approach is that in some cases the head of the compound will end up with the wrong PoS tag. Figure 2 (right) gives an example where the whole compound should have been annotated as a noun (Japanurlaub, Japan vacation) but instead now obtains a proper noun PoS tag. A possible solution to this problem is to deviate from UD practise and annotate the second component (i.e. the real head) as the head. As those cases were rare in our data, we refrained from doing so, for the sake of consistency with other UD treebanks.

2.4 Annotation
We annotated two types of PoS tags, based on the UD (Petrov et al., 2012) and Stuttgart-Tübingen (STTS) (Schiller et al., 1995) tag sets. The PoS tags and morphological features represent the annotations of one annotator, correcting the output of the UD processing pipeline for German (UDPipe) (Straka and Straková, 2017). For all dependency annotations, two annotators provided syntactic attachments and dependency labels, which were subsequently adjudicated. The adjudicated syntactic dependency relations were used for consistency checks between the dependency labels and the PoS and morphological tags. Additional consistency checks based on DECCA (Dickinson and Meurers, 2003) verified the compatibility of the different annotation layers. All incompatibilities were manually inspected and resolved. The final testsuite includes 12,073 tokens from 519 tweets, split up into train, development and test data (table 1). Around 10% of the tweets include a non-projective tree structure.
Table 1: Corpus statistics for the tweeDe testsuite (OOV: number of out-of-vocabulary words with regard to the training set; lower: OOV for lower-cased word forms).

<table>
<thead>
<tr>
<th></th>
<th># tweets</th>
<th># tok</th>
<th># vocab</th>
<th>OOV lower</th>
<th># non-projective</th>
</tr>
</thead>
<tbody>
<tr>
<td>train</td>
<td>250</td>
<td>5,747</td>
<td>2,035</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>dev</td>
<td>69</td>
<td>1,917</td>
<td>861</td>
<td>520</td>
<td>479</td>
</tr>
<tr>
<td>test</td>
<td>200</td>
<td>4,409</td>
<td>1,661</td>
<td>1,157</td>
<td>1,034</td>
</tr>
<tr>
<td>total</td>
<td>519</td>
<td>12,073</td>
<td>3,639</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Inter-Annotator Agreement  We computed IAA on a subset of the data with 1,630 tokens. For labelled attachments, the agreement between the two annotators was 0.83 $\kappa$, for unlabelled attachments the score increased to 0.89 $\kappa$.

3 Annotation decisions

Below we discuss decisions we made during the annotation process that deviate from other existing German UD treebanks, i.e. the UD-GSD and the UD-TüBa-D/Z. UD-GSD has been converted from an earlier version of Stanford-style dependencies (McDonald et al., 2013) and contains mostly web reviews while the UD-TüBa-D/Z (Çöltekin et al., 2017) is a conversion of the TüBa-D/Z (Telljohann et al., 2004) and includes articles from a German daily newspaper.

Placeholder sentences  In the UD-GSD treebank, finite subordinate placeholder sentences with dass or ob (that, whether) are mostly analysed as ccomp while infinite correlates are annotated as acl and attached to the placeholder, usually a pronominal adverb. In contrast, the TüBa-D/Z attaches both finite and infinite placeholder clauses as adverbial clause to the verb of the matrix clause.

We decided to annotate finite and infinite placeholder sentences as acl and attach both to their respective placeholder (figure 3).

Fixed multi-word constructions  German has a rich system of adverbs and particles that can form multi-word constructions and so obtain a meaning that is different from the one of their individual components. We annotate those using the dependency label fixed (figure 4 left). Adpositions also frequently form multiword units and have been treated the same (figure 4 right), as have specific combinations of pronouns and prepositions (e.g. Was für ein Unsinn! (What for a nonsense), English translation: “What utter nonsense!”).
Figure 5: Comparative clause with je-desto in tweeDe.

**Correlative construction with two clauses** The correlative construction je X, desto/umso Y (the X, the Y) (figure 5) consists of a subordinate clause marked by je, followed by a matrix clause that is introduced by desto/umso. Each clause needs to contain a comparative form, either of an adjective or of an adverb. Semantically, the construction describes a relationship between an independent and a dependent variable (example 8).

As indicated by word order, the clause expressing the causal variable is the subordinate clause (the finite verb comes last) while the clause describing the dependent variable is syntactically encoded as the matrix clause (the finite verb comes in second position). While je typically only marks the subordinate clause, there also exist variants of the construction where the desto/umso is omitted and a second je is used instead to mark the comparative that describes the dependent variable (example 9).4

(8) Je älter ich werde, umso glücklicher bin ich.  
“arlier I get, the happier I am.”

(9) Je größer die Gruppe, je kleiner der Preis.  
“The larger the group, the smaller the price.”

Based on these observations, we decided to attach the subordinate clause as an adverbial clause to the matrix clause and analyse both particles as adverbial modifiers. We do not assign the mark relation as the particles are not modifiers of the head of the subordinate clause but are modifiers of the comparative forms in the subordinate and in the matrix clause.

This analysis is different from the one in the German UD-GSD and TüBa-D/Z UD treebanks (figure 6) where the head of the subordinate clause is analysed as the root of the sentence and the matrix clause is attached as a conjunct of the subordinate clause. Our analysis is consistent with the one for conditional clauses that are similar in meaning (e.g.: If I scroll down further, I can see more), where the subordinate if-clause is also an adverbial clausal modifier of the matrix clause.

**Figure 6: Comparative clause with je-desto in the TüBa-D/Z-UD.**

4 **Comparison to other German UD treebanks**

We now compare tweeDe to three other German treebanks, i) UD-GSD, ii) TüBa-D/Z and iii) UD-HDT. The UD-HDT (Hennig and Köhn, 2017) is a conversion of the Hamburg Dependency Treebank (Foth et

3While this is the canonical order, it is also possible to switch the order of the matrix and subordinate clauses. Constructions without verbal predicates are also possible: Je mehr, desto lustiger. (The more, the merrier).

4While these are less frequent than the canonical form with je-desto/umso, it is easy to find instances in a large corpus such as the DeWac (Baroni et al., 2009), as well as instances that include only the je without a second particle where the matrix clause then needs to be in V1 word order.
al., 2014) which includes mostly news articles and is also the largest existing German treebank.

Figure 7 shows the distribution of PoS tags in the four treebanks. While the other three treebanks are quite homogeneous (except UD-GSD including more proper names), the most striking difference between tweeDe and the other treebanks is the higher number of adverbs and pronouns. This is typical for informal multiparty communication and is accompanied by a lower percentage of nouns, determiners, adjectives and adpositions as well as a slightly higher amount of verbs. This shows that tweeDe has a more verbal style, as opposed to the nominal style of the other treebanks.

5 Parsing experiments

We present parsing baselines for the new German UD treebank, using the state-of-the-art parser of Dozat et al. (2017). The parser is a neural dependency parser that learns complex, non-linear representations directly from the input text, based on bidirectional LSTMs (Hochreiter and Schmidhuber, 1997). It only considers local context and predicts attachments and labels in a greedy fashion. The huge success of the parser is based on its use of biaffine attention.

In our first experiment, we train the parser on the 250 tweets in the tweeDe training set. We use pretrained skipgram embeddings with 100 dimensions (window size: 5, min word count: 10), trained on a large collection of German tweets, collected in a time period from 2013 to 2017. The embeddings are publically available from https://www.cl.uni-heidelberg.de/research/downloads. All models have been trained with default parameters.

Table 2 (left) shows results for gold PoS and for automatically predicted PoS tags. Using UD PoS tags for parsing outperforms the STTS tags by a large margin, probably due to sparsity caused by the more fine-grained STTS. Feeding both, UD and STTS tags, to the parser can further increase results, but only slightly (less than 1%). Most surprisingly, we obtain higher results when using automatically predicted STTS tags (as compared to using gold STTS tags). This observation, however, is more pronounced for the test set and might not be representative, being an artefact of the small data size.

Results for training on the small tweeDe dataset only are in the range of 74% LAS (gold PoS) and 68% LAS (auto PoS). When adding the training data from the German-GSD UD treebank, results increase to 81% LAS (gold PoS) and 76% LAS (auto PoS). The large gap of 5% between the gold and auto PoS setting highlights the importance of high-quality PoS tags for parsing tweets.

<table>
<thead>
<tr>
<th>PoS tagset</th>
<th>dev</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td>gold UD</td>
<td>82.15</td>
<td>74.26</td>
</tr>
<tr>
<td>STTS</td>
<td>73.48</td>
<td>63.05</td>
</tr>
<tr>
<td>BOTH</td>
<td>82.51</td>
<td>74.94</td>
</tr>
<tr>
<td>auto UD</td>
<td>78.88</td>
<td>69.90</td>
</tr>
<tr>
<td>STTS</td>
<td>72.91</td>
<td>63.21</td>
</tr>
<tr>
<td>BOTH</td>
<td>79.09</td>
<td>70.73</td>
</tr>
</tbody>
</table>

Table 2: Parsing results for the Dozat parser on tweeDe, without (left) and with additional training data from the German-GSD UD treebank (right).
Table 3: Statistics for manually annotated treebanks (*Foster et al. only report # sentences, not # tweets. We expect the no. of tweets to be slightly lower than 500). The data of Blodgett et al. includes AAE and main-stream (MS) English tweets. The last two columns report results for the Dozat & Manning parser (Dozat et al., 2017) (w/o domain adaptation) or the Malt parser from the literature.

6 Related work

Twitter treebanks exist not only for English (Kong et al., 2014; Liu et al., 2018; Blodgett et al., 2018) but also for Italian (Sanguinetti et al., 2018) and Arabic (Albogamy et al., 2017). Foster et al. (2011) were among the first to provide syntactic analyses for Twitter microtext. They created a testset with over 500 sentences extracted from tweets. The data was automatically parsed with a constituency parser and the trees were manually corrected by one annotator. Inter-annotator agreement (IAA) for labelled bracketing, measured on a subset of the data annotated by a second annotator, was quite high with nearly 96%. Parsing accuracy without any domain adaptation, however, was low: the Malt parser (Nivre et al., 2006), trained on the WSJ, achieved an LAS of 63.3% on the Twitter testset.

The Tweebank v1 (Kong et al., 2014) is another English Twitter treebank, with a size of over 900 tweets annotated with unlabelled dependencies. Liu et al. (2018) extend the work of Kong et al. (2014) by enlarging the treebank to more than 3,500 tweets, refining the guidelines and adding labels to the former unlabelled trees. They report an IAA of 84.3% for labelled attachments in the Tweebank v2. A third English Twitter treebank was created by Blodgett et al. (2018). Their corpus includes 250 African-American English (AAE) tweets and 250 tweets of mainstream American English microtext. The data has been annotated by two coders but no inter-annotator agreement is reported.

The Italian Twitter treebank of Sanguinetti et al. (2018) is the largest existing Twitter treebank and includes more than 6,700 trees. The authors report an IAA of 0.92 $\kappa$ for syntactic annotation. The results for a dependency parser (Dozat et al., 2017) trained on a combination of the Italian UD treebank and the new dataset are also quite high, with a labelled attachment score of 81.5%. The high agreement and parsing scores suggest that the dataset is somewhat easier and more well-behaved than the Tweebank (see table 3 for baseline results for the different Twitter treebanks).

For Arabic, a treebank with Twitter microtext has been created fully automatically, based on predictions of a rule-based and a data-driven parser (Albogamy et al., 2017). Efforts have been made to map the annotations to the UD scheme, but, to the best of our knowledge, the data is not yet available.

With over 12,000 tokens, our new German Twitter treebank is comparable in size to TWEEBANK V1 (Kong et al., 2014) even though the number of tweets in our dataset is smaller. This is due to the fact that our data were collected after Twitter raised the maximum length for tweets from 140 to 280 characters.

7 Conclusions

We presented tweeDe, the first German Twitter treebank, as a new training and testsuite for UD parsing. tweeDe includes more than 12,000 tokens of informal private communication, annotated for PoS, morphology and UD syntactic dependencies. The data will be made available to the research community.\(^5\)

We also presented parsing baselines for the new dataset, showing that combining a small amount of in-domain Twitter data in combination with a larger amount of out-of-domain data can yield parsing accuracies in the range of 83% (UAS) and 76% (LAS) on the new testsuite.

\(^5\)https://www.cl.uni-heidelberg.de/research/downloads.
References


