Abstract

We present a language-independent clausizer (clause splitter) based on Universal Dependencies (Nivre et al., 2016), and a clause-level tagger for grammatical tense, mood, voice and modality in German. The paper recapitulates verbal inflection in German—always juxtaposed with its close relative English—and transforms the linguistic theory into a rule-based algorithm. We achieve state-of-the-art accuracies of 92.6% for tense, 79.0% for mood, 93.8% for voice and 79.8% for modality in the literary domain. Our implementation is available at https://gitlab.gwdg.de/tillmann.doenicke/tense-tagger.

1 Introduction

A clause is a syntactic unit within a sentence that contains a verb and all of its arguments (subject, object etc.) and adjuncts (adverbials of time, location etc.), i.e., clauses describe events (or states) and therefore are the core elements of discourse. Several important properties of an event are expressed by inflectional features of the verb alone: Tense and aspect express the relation between event time, speech time and reference time (Reichenbach, 1947; Boogaart and Janssen, 2007), mood expresses the reality status of an event (Elliott, 2000), and voice expresses a mapping between the syntactic arguments of a verb and semantic roles (agent, patient etc.). Modal verbs further mark the modality of an event, such as deonticity and epistemicity (Leiss, 2008). Hence, extracting these features from a clause is a crucial task for discourse analysis. Following previous work (Bögel et al., 2014; Ramm et al., 2017), we address this task with a rule-based approach.

We use parse trees in the Universal Dependencies (UD; Nivre et al. (2016)) format to split sentences into clauses, which makes our clause-splitting method applicable to all languages with a UD treebank. Nevertheless, the morphosyntactic systems for tense, aspect, mood, voice and modality vary greatly between languages (cf. Aronson (1995), Zeitoun et al. (1996), Lin (2005), Keenan and Dryer (2007), Singh et al. (2007) and many others) and do not allow a crosslinguistic approach. We focus on German which shows strong parallels to English.

This paper presents an approach towards tagging morphosyntactic/grammatical features which do not always correspond to semantic features. This is best observable for tense; all of the following examples feature present tense but describe events in the present, past or future:

(1) a. John sees Mary.
   b. 44 BC, Caesar is stabbed by a group of senators. (historical present, Wolfson (1978))
   c. Tomorrow, we go to the cinema. (future present)

Tagging and normalising temporal expressions such as 44 BC and tomorrow is a separate research task (cf. Strötgen and Gertz (2010), Pustejovsky and Verhagen (2009) and subsequent SemEval tasks) which is not addressed in this paper. In the long run, both temporal expressions and grammatical tense together are helpful for inferring semantic tense.

The difference between syntax and semantics also affects the other features under consideration. The presence of a modal verb, for example, can cause multiple semantic interpretations: he must work is ambiguous between he is required to work (deontic interpretation) and he is very likely to work [according to what the speaker knows] (epistemic interpretation) (Viebahn and Vetter, 2016; Tarvainen, 1976).
| Tense + Aspect       | Alternate names     | Example (indicative, active) |
|----------------------|---------------------|------------------------------|
| present imperfect    | present             | sieht ‘sees’                 |
| present perfect      | perfect             | gesehen hat ‘has seen’       |
| past imperfect       | preterite, imperfect| sah ‘saw’                    |
| past perfect         | pluperfect           | gesehen hatte ‘had seen’     |
| future imperfect     | future, future I    | sehen wird ‘will see’        |
| future perfect       | future II           | gesehen haben wird ‘will have seen’ |

Table 1: Tense–aspect combinations in German.

Grammatical tense also plays an important role in the analysis of narrative texts which are usually written in the simple past. If the tense changes locally, this marks a potential passage of interest. For example, if the tense changes to the simple present, it could be a passage with gnomic reading (i.e. a passage expressing a general truth) as in (2):

(2) John tried to catch a rabbit. Rabbits are fast, but finally he got it.

This paper is structured as follows: section 2 gives an overview of the inflection of verbs in German; section 3 summarises the previous approaches to tagging tense, mood and voice in German; section 4 contains our algorithms and implementation details; sections 5 and 6 contain the evaluation and discussion of our tool, including comparisons with the previous works; sections 7 and 8 conclude with an outlook on future work and a summary.

2 Inflection and Government in German Clauses

German has three tenses: present, past, future, and two aspects: imperfect (= simple) and perfect, and therefore six tense–aspect combinations (Table 1). The composition of verb forms is very similar to their English counterparts; a main verb is extended by auxiliary verb forms of haben ‘have’, sein ‘be’ and werden ‘will/become/get’. For example, the past perfect form of sehen ‘see’ is (er) hatte gesehen ‘(he) had seen’. Since tense and aspect are inseparable, they are sometimes simply referred to as “tense”.

German further distinguishes four moods: indicative, present subjunctive (subjunctive I), past subjunctive (subjunctive II) and imperative, as well as three voices: active, dynamic passive and static passive. All of these are expressed by combinations of the three auxiliary verbs mentioned above.

2.1 Word Order

The basic German word order is S-O-V. All verbs are positioned at the end of a clause; starting with the syntactically lowest verb and ending with the syntactically highest verb. However, this ordering is only maintained in subordinate clauses; in main clauses, the finite verb (which is always the syntactically highest verb) moves to verb-second position:

\[1\] German makes a clear distinction between the dynamic passive using the auxiliary verb werden ‘get’ (3a) and the static passive using the auxiliary verb sein ‘be’ (3b). In English, on the other side, passives with be are ambiguous between a dynamic and a static reading:

(3) a. i. Er wird gefüttert [und verschlingt seinen Fraß].
    
     ii. He is/gets fed [and is devouring his food].

    b. i. Er ist gefüttert [und schläft jetzt].

     ii. He is/*gets fed [and is now sleeping].

\[2\] In polar questions, the finite verb moves to sentence-initial position; in subordinate clauses, the finite verb may move to the so-called Oberfeld (cf. e.g. Hinrichs (2016)). For this paper, it is enough to say that the finite verb can move to a position preceding the non-finite verbs.
English, as an S-V-O language, employs the exact opposite order of verbs. In other words, the direction of verbal government is right-to-left in German, and left-to-right in English:

(5)  
i. (dass) er sie gesehen haben wird.

ii. (that) he will have seen her.

The strict ordering makes it possible to derive the syntactic hierarchy of the verbs in a clause without applying a syntactic parser.

2.2 Morphological vs. Clausal Features

As we have seen in (4) and (5), a verb form can consist of several verbs. Each verb has its own morphological features. The features of a composite verb form (= the clausal features) result from the morphological features of the individual verbs. We use feature structures, i.e. sets of feature-value pairs, (see Jurafsky and Martin (2009) for an introduction), to represent morphological and clausal features. Clausal features cannot be derived by unification of the involved morphological features though; this is why we denote the compositional process with a function \( R \) which maps a set of morphological features to the features of the clause. For (4) we get:

\[
R \left( \begin{array}{c}
\text{LEMA} & \text{sehen} \\
\text{TYPE} & \text{main} \\
\text{ASPECT} & \text{perfect} \\
\text{VOICE} & \text{passive} \\
\end{array} \right), \begin{array}{c}
\text{LEMA} & \text{haben} \\
\text{TYPE} & \text{auxiliary} \\
\text{TENSE} & \text{past} \\
\text{ASPECT} & \text{perfect} \\
\text{MOOD} & \text{indicative} \\
\text{VOICE} & \text{active} \\
\end{array} \right) = \begin{array}{c}
\text{FORM} & \text{finite} \\
\text{TENSE} & \text{past} \\
\text{ASPECT} & \text{perfect} \\
\text{MOOD} & \text{indicative} \\
\text{VOICE} & \text{active} \\
\end{array}
\]

2.3 Modal Verbs

Modal verbs are not part of a composite verb form but possibly take over inflectional features. (6a) and (6b) are identical in terms of tense, mood and voice but the modal verb \( muss \) ‘must’ in (6b) shows the inflectional features of the auxiliary verb \( hat \) ‘has’ in (6a).

(6)  

a. i. (dass) er sie gesehen hat.

ii. (that) he has seen her.

b. i. (dass) er sie gesehen haben muss.

ii. (that) he must have seen her.

To obtain the basic verb form without (interfering) modal verbs, one has to shift their features to the next verb in the direction of verbal government.\(^3\)

---

\(^3\) In English, the shifting of inflectional features is also observable in negation or emphasis with the auxiliary verb \( do \):

(7)  

a. He has seen her.

b. He doesn’t have seen her.
2.4 Substitute Infinitives

In German, modal verbs and some other verbs can exhibit a substitute infinitive (*infinitivus pro partici-pio*), i.e. use the infinitive instead of the perfect participle. *Müssen* ‘have to’ in (8a) and *hören* ‘hear’ in (8b) (Bausewein, 1991) are substitute infinitives:

(8) a. i. (dass) er sie sehen müssen/*gemusst hat.
   ii. (that) he has had to see her.

   b. i. (dass) er sie singen hören/gehört hat.
   ii. (that) he has heard her sing.

If substitute infinitives are governed by an auxiliary verb, this is always a form of *haben* ‘have’.

3 Previous Approaches and Corpora for German

3.1 Bögel et al. (2014)

As part of the heureCLÉA project\(^4\), Bögel et al. (2014) developed a clause-level tagger for five tense–aspect combinations (future imperfect and future perfect are combined into one tag). Their pipeline is implemented in the UIMA framework\(^5\) and makes use of several external resources, such as the TreeTagger (Schmid, 1995) for part-of-speech tagging, the Stanford Parser for constituent parsing and Morphisto (Zielinski et al., 2009) as a morphological analyzer. Clauses (“sub-sentences” in Bögel et al. (2014)) are defined as constituents with an own S root. The final tense is predicted using a small set of rules, e.g.

\[
R \left( \begin{array}{c}
\text{TYPE} \text{ main} \\
\text{FORM} \text{ participle} \\
\text{TYPE} \text{ auxiliary} \\
\text{TENSE} \text{ present} \\
\end{array} \right) \rightarrow \begin{array}{c}
\text{TENSE} \text{ present} \\
\text{ASPECT} \text{ perfect} \\
\end{array}
\]

and a heuristic for discontinuities, which copies the tense for a clause from its neighbouring clauses if \(R\) does not provide an analysis.

The evaluation corpus consists of twenty narrative texts, and the first 20% of each text (nearly 12k tokens in total) are annotated with tense. In the evaluation, they measured (i) all correctly tagged tokens (all tokens in a clause are assigned the same tense as the main verb), as well as (ii) only the correctly tagged main verbs. The reported accuracies are 94.8% and 93.3%, respectively. Most of the tagging errors are caused by incorrect parser outputs (and thus incorrect clause splitting) or incorrect annotations.

The tense tagger was provided through the annotation tool CATMA\(^6\), version 5. Unfortunately, it was not transferred when moving to CATMA 6 (current version) and the account creation for CATMA 5 has been deactivated, which makes the tense tagger inaccessible. The corpus is still available at [https://github.com/heureclea](https://github.com/heureclea).

3.2 Ramm et al. (2017)

The tmv-annotator by Ramm et al. (2017) is a Python tool for tagging preprocessed German, English or French texts with tense mood and voice. For German, the tagsets include all six tenses, three moods (imperative is missing) and two voices (no distinction between static and dynamic passive). To use the tool (available at [https://github.com/aniramm/tmv-annotator](https://github.com/aniramm/tmv-annotator)), the texts have to be preprocessed with MATE tools\(^7\)—or another tool providing the same output—which is implemented in Java and includes tokenisation, part-of-speech tagging, lemmatisation, morphological analysis and dependency parsing (but no sentence splitting although the text has to be split into sentences before applying the tokeniser). Unlike the Stanford Parser which provides constituent parses, the MATE parser provides dependency parses in the German TIGER/CoNLL format (cf. Buchholz and Marsi (2006), Hajič et al. (2009)). The composite verb form of a clause (“verb cluster” in Ramm et al. (2017)) is extracted by first selecting the main verb and then collecting the dependent auxiliary verbs. The final analysis is predicted with a

\(^4\)http://heureclea.de/
\(^5\)http://uima.apache.org/
\(^6\)http://www.catma.de/
\(^7\)https://code.google.com/archive/p/mate-tools/
rule-set similarly as in Bögel et al. (2014). The output of the tool is a table format providing all main verbs and tense/mood/voice tags as well as the clauses which contain the verbs.

The tool was evaluated on 157 randomly selected clauses from the Europarl corpus (Koehn, 2005) which had been annotated with the respective features. The reported accuracies are 80.8% for tense, 84.0% for mood and 81.5% for voice. Unfortunately, the evaluation corpus is not available anymore.

4 Method/Implementation

We implemented the entire pipeline in spaCy\(^8\), an open-source software library for crosslinguistic natural language processing in Python. The pipeline is shown in Figure 1; its individual components are described below.

4.1 Preprocessing

We used the default tokenizer, lemmatizer, part-of-speech tagger and sentencizer (sentence splitter) from the German spaCy model.\(^9\)

4.2 Universal Dependency Parsing

Universal Dependencies (UD; Nivre et al. (2016))\(^{10}\) are a crosslinguistic annotation format and also a collection of treebanks from a wide range of languages annotated in that format. An advantage of the universal annotation format, with respect to our need for clause splitting, is that clauses can easily be identified through certain dependency relations (e.g. \texttt{nsubj} marks a nominal subject whereas \texttt{csubj} marks a clausal subject). This is not the case with, for example, the TIGER annotation scheme for German (here \texttt{sb} marks both non-clausal and clausal subjects). We therefore decided to parse our texts with UD relations.

Unfortunately, German and English are the only languages for which the default spaCy parser does not use UD relations. Therefore—and because there is currently no German UD model for spaCy available—, we trained a new parser on the current version of the UD treebanks (Zeman et al., 2020). In contrast to e.g. the Stanford parser which was solely trained on newspaper texts, the German UD treebanks also contain texts from different domains, including a small proportion of texts from literary history (LIT treebank). We held out the test sets of GSD and HDT (9.3% of the sentences) for testing and achieved a labelled attachment score (Zeman et al., 2017) of 85%. We provide our spaCy model along with the rest of our code.

4.3 Crosslinguistic Clause Splitting

As mentioned above, certain UD relations can be used to split a sentence into clauses. To be more precise, if one of the following relations is encountered in a sentence, the tokens of the corresponding subtree, ignoring punctuation, form a clause: \texttt{root} (matrix sentence), \texttt{acl} (adjectival clause), \texttt{advcl} (adverbial clause), \texttt{ccomp} (clausal complement), \texttt{csubj} (clausal subject), \texttt{discourse} (interjections etc.), \texttt{parataxis}, \texttt{vocative}, \texttt{list}. The relations \texttt{xcomp} (open clausal complement) and \texttt{conj} (conjunct) sometimes but not always mark clauses. We split at these relations if certain conditions are met: at an \texttt{xcomp} if the subtree consists of at least a verb and one additional word which is not a verbal particle (i.e. if the subtree forms an extended infinitive clause); at a \texttt{conj} if the label of its head is one of the clause labels listed above (i.e. if the subtree is conjuncted on clause-level). These conditions are hyperparameters in our implementation and can be easily changed if one prefers another handling of open clausal complements or conjuncts.

\(^8\)https://spacy.io/
\(^9\)The pre-trained German model is available at https://spacy.io/models/de#de_core_news_lg.
\(^{10}\)https://universaldependencies.org/
Our clausizer is applicable to all texts with UD parse trees, either after being parsed accordingly (e.g. with spaCy) or after being manually annotated (e.g. within the UD treebanks project). Figure 2 shows a sentence from the German and English PUD treebanks. Each sentence contains four clauses. We implemented the clausizer to recursively detect nested clauses, e.g. two clauses are detected in (9): Der Mann lacht ‘The man laughs’ and der die Kuh sah ‘who saw the cow’.

(9) i. Der Mann, der die Kuh sah, lacht.
   ii. The man who saw the cow laughs.

4.4 Morphological Analysis

SpaCy already assigns some morphological features to words, e.g. the form of a verb, i.e. whether it is finite, an infinitive or a participle. In addition, we use DEMorphy (Altinok, 2018)\(^\text{11}\), a morphological analyzer for German. Since DEMorphy outputs all analyses for a word—indeed from its context—we filter out unlikely analyses due to case–number–gender congruence. To be more precise, the words within a noun phrase should be congruent in case, number and gender, and a finite verb should be congruent with its subject in number and person.

4.5 TMV Tagging

The algorithm for our tense–mood–voice (TMV) tagger is sketched in Algorithm 1. In the following, numbers in parentheses refer to the corresponding lines in the pseudocode.

Given a clause C, the non-finite verbs, i.e. infinitives and participles, are stored in a list V (l. 1). In contrast to the procedure of Ramm et al. (2017), this step does not rely on the output of a parser. If the

\(^{11}\text{https://github.com/DuyguA/DEMorphy}
Algorithm 1: Compute features of a clause C

1. \( V \leftarrow \text{[non-finite verbs in } C\text{]} \)
2. if finite verb in \( C \) then
3. \( v_{\text{fin}} \leftarrow \text{right-most finite verb in } C \)
4. \( V \leftarrow [v_1, \ldots, v_{|V|}, v_{\text{fin}}] \)
5. if \( C \) is conjunct then
6. \( V \leftarrow \text{copy_verbs}(V, C, \text{head}(C)) \)
7. if \( |V| = 0 \) then
8. \( \text{return } \)
9. else if main verb in \( V \) then
10. \( v_{\text{main}} \leftarrow \text{right-most main verb in } V \)
11. else
12. \( v_{\text{main}} \leftarrow \text{left-most verb in } V \)
13. \( V \leftarrow [v_{\text{main}}, \ldots, v_{\text{fin}}] \)
14. \( M \leftarrow \{\text{features}(v_i)\} \text{ for } i = 1 \text{ to } |V| \)
15. if \( \text{LEMMAT}(\text{haben}) \subseteq \text{first}(m_{|V|}) \) and \( \text{FORM infinitive} \subseteq \text{first}(m_{|V|-1}) \) then
16. \( m_{|V|-1} \subseteq \{\text{FORM participle, ASPECT perfect}\} \)
17. for \( i = |V| \) to 1 do
18. if \( v_i \) is modal verb then
19. \( m_{i-1} \leftarrow m_i \)
20. while \(|V| > 0\) do
21. Set \( v_1 \) to be the main verb
22. \( F \leftarrow \times_{1 \leq i < |V|} m_{|V|-i} \)
23. if \( v_i \) is not modal verb then
24. \( A \leftarrow \{\} \)
25. for \( i = 1 \) to \( |F| \) do
26. if \( R(f_i) \neq \text{NULL} \) then
27. \( A_{\leftarrow} R(f_i) \)
28. if \( |A| > 0 \) then
29. \( a \leftarrow \text{first}(\text{filter}(A)) \)
30. \( V_{\text{modal}} \leftarrow [\text{modal verbs in } V] \)
31. \( a_{\leftarrow} \text{MODALITY } V_{\text{modal}} \)
32. \( V \leftarrow [v_2, \ldots, v_{|V|}] \)
33. \( \text{return } \)

For a set \( S = \{s_1, \ldots, s_{|S|}\} \), first(\( S \)) is identical to \( s_1 \).

\( \leftarrow \) and \( \leftarrow \) are augmented assignment operators for union and unification, respectively.

A clause contains a finite verb, then it is appended to \( V \) (ll. 2–4). In that way, the verbs are sorted in basic word order, i.e. as if the clause was a subordinate clause.

If \( C \) is a conjunct, the potentially missing verbs are copied from the head clause (ll. 5–6). For example, (10) contains the clauses er sie gesehen hatte ‘he had seen her’ and und gerufen ‘and called’; hatte ‘had’ has to be copied from the first to the second clause to complete the composite verb form gerufen hatte ‘had called’.

(10) i. (dass) er sie gesehen und gerufen hatte.
ii. (that) he had seen and called her.

The next step is to select the clause’s main verb. If there is at least one genuine main verb in \( V \), the right-most (= syntactically highest) one is chosen (ll. 9–10). In (11a), this is gelernt ‘learned’. (11b) and (11c) illustrate that auxiliary verbs and modal verbs can function as main verb as well. If there is no genuine main verb in the clause, the left-most (= syntactically lowest) verb is chosen (ll. 11–12). In (11b), this is gewesen ‘been’; in (11c), this is kann ‘can’. Note that speak is the main verb of the English translation since can cannot be used alone here; German is much freer in using modal verbs as main verbs.

(11) a. i. (dass) er sprechen gelernt hatte.
ii. (that) he had learned to speak.

b. i. (dass) er dort gewesen war.
ii. (that) he had been there.

c. i. (dass) er Englisch kann.
ii. (that) he can [speak] English.

Only the verbs from the main verb to the finite verb are interesting for TMV tagging, because the main verb is the syntactically lowest verb of a composite verb form; all other verbs which precede the main verb are removed from \( V \) (l. 13). \( M \) contains the feature structures for every word, i.e. \( m_i \) (\( 1 \leq i \leq |V| \))
is a set of possible morphological analyses for \( v_i \) (l. 14). If the second verb from the right \( v_{|V|-1} \) is a potential substitute infinitive, the feature structure of a perfect participle is added to \( m_{|V|-1} \) (ll. 15–16). Having all verbs of interest together, the features of modal verbs are shifted to their predecessors as described in section 2.3 (ll. 17–19).

The Cartesian product of \( m_1, \ldots, m_{|V|} \) (now ignoring modal verbs) yields all possible combinations of morphological analyses of the involved verbs and is stored in \( F \) (l. 22). Every combination \( f_i \in F \) is then tried to be mapped to the clausal features \( R(f_i) \). Instead of using hand-crafted rules like previous work, we created a table of all possible verb forms for the look-up (a table with all verb forms can be found in the appendix). If \( f_i \) is in the table, then \( R(f_i) \) is saved in the final set of analyses \( A \) (ll. 23–26).

If no analysis is found, the first verb in \( V \) is removed (l. 32) and the last paragraph is repeated (ll. 20–21). This counteracts tagging and parsing errors and makes it possible to also tag rarely used verb combinations such as sequences of auxiliaries as in (12a) or double perfect constructions (Ammann, 2007) as in (12b).

(12) a. i. (dass) er dort gewesen gewesen ist.
   ii. (that) he has been been there.
   b. i. (dass) er sie gesehen gehabt hat.
   ii. (that) he has had seen her.

As soon as one or more analyses are found, one of them is selected and returned (ll. 27–31). In German, most verbs express the perfect aspect with the auxiliary verb ‘have’ (e.g. hat gesehen ‘has seen’) but some use ‘be’ (e.g. ist gegangen ‘is gone’) and others can use either depending on the context or regional varieties (whereas in English it is almost always have). Since forms of sein can not only mark perfect aspect but also static passive, this causes ambiguous verb forms. To resolve these ambiguities, we filter the analyses with respect to the main verb’s possible perfect auxiliaries (this is also done by Ramm et al. (2017)). We extracted the possible perfect auxiliaries for every German verb in the German Wiktionary12.

Before the final analysis is returned, its modality feature is set to the list of modal verbs in the current \( V \) (ll. 29–30) (syntactically lower modal verbs are not returned).

5 Evaluation

We compared the performances of our tagger and the tagger from Ramm et al. (2017) on the texts in the heureCLÉA corpus as well as on a text annotated by ourselves.

5.1 Annotation

We annotated the German translation of the preface of Don Quijote by Miguel de Cervantes Saavedra13 (3,200 tokens) which contains a lot of complex (multi-clause) sentences and examples for all six tenses, four moods, three voices and the modal verbs ‘can’, ‘may’, ‘must’, ‘shall’ and ‘want’. Two annotators annotated the text with tense. After calculating the inter-annotator agreement (\( \kappa = 96\% \), Fleiss et al. (2003)), we combined the two annotations into a gold annotation and extended it with finiteness, mood, voice and the modal verbs involved in a verb form.

We used the official German Duden grammar (Dudenredaktion, 2009, pp. 476 ff.) as reference guide for our annotation of tense, mood and voice. We also annotated non-finite clauses (with infinitive or participle forms) with tense and voice14—non-finite forms do not feature mood—, whereas Ramm et al. (2017) only consider finite verb forms and in heureCLÉA non-finite clauses are either not annotated or receive the tense of the corresponding matrix clause.

12https://dumps.wikimedia.org/dewiktionary/
13The text is available at https://www.projekt-gutenberg.org/cervante/quietote1/quietote1.html.
14It is debatable whether infinitives and participles feature tense or only aspect. This is, however, only a matter of definition. Since we only tag tense–aspect combinations, we use the present imperfect or present perfect for all non-finite verb forms.
Table 2: Inter-annotator agreements and tense tagging accuracies for the heureCLÉA corpus and/or our test text. Numbers in brackets are copied from Bögel et al. (2014). Accuracies are shown for all tokens or only main verbs.

|                  | heureCLÉA | Don Quijote |
|------------------|-----------|------------|
|                  | Tokens    | Verbs      | Tokens    | Verbs      |
| Fleiss’ κ        | (89.7)    | (84.0)     | 96.3      | 96.0       |
| Bögel et al. (2014) | (93.3)    | (94.8)     | –         | –          |
| Ramm et al. (2017)| 74.9      | 81.9       | 55.8      | 63.7       |
| this work        | 88.8      | 90.8       | 87.2      | 92.6       |

Table 3: Comparison of two taggers for tense, mood, voice and modality on our test text. Accuracies are calculated for main verbs in finite clauses. The first column shows the accuracy distinguishing main verbs in finite clauses from main verbs in non-finite clauses.

5.2 Tense Evaluation

The first evaluation concentrates on tense tagging. Following Bögel et al. (2014), we provide the accuracy for correctly tagged tokens (where each token is assigned the tense of the clause) as well as the accuracy for the correctly tagged main verbs. Table 2 shows the accuracies for testing on the heureCLÉA corpus and our gold annotation of *Don Quijote*.

For heureCLÉA, there is no gold annotation but only the unmerged annotations from two annotators. As in Bögel et al. (2014), we only use those tokens for accuracy calculation which had been annotated with the same tense from both annotators, and we combine future imperfect and future perfect into one tag.

5.3 TMV and Modality Evaluation

For the second evaluation, we used the annotations of finiteness, tense, mood, voice and modality for *Don Quijote*. Since Ramm et al. (2017)’s tagger only tags finite verb forms, we decided to only compare the performances of the taggers on clauses annotated as finite. We further combined indicative and imperative mood as well as static passive and dynamic passive to have the same categories as Ramm et al. (2017). The first column of Table 3 shows the performance of Ramm et al. (2017)’s and our tagger for detecting whether a verb form is finite or non-finite. The other columns show the accuracies for correctly tagged main verbs in finite clauses. The last row shows the accuracies for our tagger when not merging mood and voice to Ramm et al. (2017)’s categories and evaluating on all verbs, including those in non-finite clauses.

5.4 Clause Evaluation

We also tested the sole performance of our clausizer. For the evaluation on *Don Quijote*, we compared the clause boundaries of the annotation $B_{gold}$ with the predicted boundaries $B_{pred}$ (cf. Jurish and Würzner (2013)). We define a clause boundary as a tuple $(e_i, s_{i+1})$ of character positions, namely the end position $e_i$ of a clause and the start position $s_{i+1}$ of the next clause in the text. Precision, recall and $F_1$-score are calculated respectively as

\[ P = \frac{|B_{gold} \cap B_{pred}|}{|B_{pred}|}, \quad R = \frac{|B_{gold} \cap B_{pred}|}{|B_{gold}|}, \quad \text{and} \quad F_1 = \frac{2 \cdot P \cdot R}{P + R}. \]

\(^{15}\)A clause inside another clause produces the same boundaries as three subsequent clauses. It is not possible to distinguish these cases in the calculations, because the annotation format does not distinguish them either.
Table 4: Clause splitting precisions, recalls and $F_1$-scores of our clausizer on our test text (German) and the CoNLL-2001 shared task test set (English). The first two rows show the number of gold and predicted instances.

We additionally applied the clausizer to the test set from the CoNLL-2001 shared task on clause identification (in English) (Tjong Kim Sang and Déjean, 2001). The goal in the shared task was the automatic detection of 1) start tokens, 2) end tokens, and 3) entire spans of clauses. The evaluation of our tool on this dataset is somewhat problematic because the concept of what a clause is differs in several aspects. The main difference is that every token belongs to exactly one clause in our concept, namely the syntactically deepest clause where it appears in, whereas a token also belongs to all of its superordinate clauses in the shared task’s concept. Therefore, our clausizer would definitely not detect the same spans as in the test set. However, we can evaluate the clausizer on the detection of clause starts and ends; here, the actual number of clauses that start or end on those positions is not considered. For the prediction, we used the sentence boundaries and part-of-speech tags as in the test set, the pre-trained English spaCy model for parsing, and our clausizer in the same configuration as for German, with a small modification: As noted earlier, the English spaCy model does not use UD relations, but instead produces the earlier Stanford relations (déMarneffe and Manning, 2008) which are quite similar to the UD relations. We added $csubjpass$, $intj$, $pcomp$, and $relcl$ (which do not appear in the UD inventory) to the list of clause-marking relations.

Table 4 shows the performances of the clausizer on Don Quijote and the English test set. We achieve $F_1$-scores of 81.3% for clause boundaries in Don Quijote, and of 73.5% for clause starts and 76.7% for clause ends in the English test set, respectively. Note that the number of predicted starts is identical to the number of predicted ends, since every token is only part of one clause in our system. The number of gold starts and ends varies, since every token can be start and end of several (nested) clauses in the test set. The scores of the systems designed for and submitted to the shared task range between 50% and 92% for clause starts and 60% and 90% for clause ends, respectively.

6 Discussion

Our tagger achieves adequate accuracies for tense, mood and voice on the preface of Don Quijote, and outperforms the tagger from Ramm et al. (2017) in every evaluation condition, both on our test text as well as the heureCLÉA corpus. We perform about 4% worse on the heureCLÉA corpus than the original tagger of Bögel et al. (2014). A frequent cause for mismatches is the different treatment of non-finite clauses, which frequently receive the tense of the matrix clause in the heureCLÉA corpus but are standardly tagged with present or perfect tense from our tagger. Clauses are not annotated with finiteness in heureCLÉA and it is therefore neither possible to exclude non-finite clauses from the evaluation, nor to estimate their exact impact. In Don Quijote, about 12% of the main verbs are annotated as non-finite, and one can assume that the amount in heureCLÉA is approximately the same.

A manual inspection of the tagger outputs shows that Ramm et al. (2017)’s tagger sometimes leaves entire clauses within complex sentences untagged which is probably an indication of incorrectly split clauses. Our clausizer, on the other hand, is more robust when it comes to these kinds of sentences. Ramm et al. (2017)’s tagger also tags verbs in past subjunctive, e.g. dächte ‘would think’, as present tense (which is usually the semantic tense) although its grammatical tense is the past tense. Again, our

---

16The pre-trained English model is available at https://spacy.io/models/en#en_core_web_lg.
complete look-up table is not as prone to errors as a set of rules.

Our comparatively low accuracy for mood mainly results from open clausal complements (xcomp in UD) that are not treated as clauses in our annotation but are recognised as such by the clausizer. Such clauses are non-finite and hence not tagged with mood. Mostly, these are cases where the annotators had overlooked an embedded infinitive clause, such as the underlined clause in (13), and then annotated it as part of the finite clause.

(13) i. (Gedichte,) die man den Büchern an den Eingang zu setzen pflegt
    ii. (poems) that one uses to place at the beginning of the books

The tagging of modal verbs also leaves room for improvement. The main cause for this are conjuncted clauses in which the modal verb is not correctly copied from a main clause to its conjuncts by our conjunct handling algorithm.

Another type of error are incorrect analyses caused by preprocessing components. An example for this are perfect and pluperfect forms (e.g. hatte gesehen ‘had seen’) which are sometimes tagged as their respective imperfect tenses, present and preterite; e.g. because the morphological analyzer does not recognise the participle as such or the clausizer separates the verbs due to an incorrect parser output. Given parsing and clausizing performances of 85% and 81%, it is encouraging that we reach TMV tagging accuracies of over 90%. The influence of the syntactic preprocessing might be partially alleviated by the fact that our tagger itself does not use dependency information. Nevertheless, improvements in the parser would surely improve the performances of the clausizer and subsequently the tagger.

7 Future Work

As mentioned above, we oriented ourselves to usual German school grammars (Dudenredaktion, 2009) when building our tagsets for tense, mood and voice. However, it might be useful to also include non-canonical, but grammaticalised composite verb forms such as the already mentioned double perfect/pluperfect or the recipient passive (e.g. Ziering et al. (2012)) with the auxiliary verb bekommen ‘receive’. To do so, nothing more is required than to extend the table of possible verb forms (the look-up function $R$).

Our approach works for every language with a hierarchically ordered verb structure, such as German and English. To adapt our approach to another language, a morphological analyzer of that language, a table of verb forms and perhaps a list of modal verbs is required. Resources such as Wiktionary provide verb type information and inflection tables for numerous languages and can be used with little effort. Our clausizer, which relies on Universal Dependencies relations, already works language-independently.

Future work could also address the transition from rule-based systems to distributional models. Although mapping morphological features to clausal features is a strictly rule-based process, grouping verbs into verb forms and selecting context-specific analyses for all relevant verbs is not. Since training these models usually requires a certain amount of annotated data, a preliminary step would be the creation of sufficient corpora. For example, clause-level features could be added to the Universal Dependencies treebanks, as they already have the concept of clause-marking dependency relations.

8 Conclusion

In this work, we provide a rule-based method to detect grammatical/morphosyntactic tense, mood, voice and modality on clause level in German. Our algorithm is grounded in linguistic theory and makes use of the hierarchically ordered verb structure in German. We also provide our preprocessing pipeline (implemented in Python/spaCy), including a German parsing model for Universal Dependencies (UD), a language-independent clausizer that splits sentences with UD parses into clauses, and an interface to the morphological analyzer DEMopry. We evaluated our approach on literary texts and achieve new state-of-the-art accuracies in all categories. Since our algorithm is rule-based, it does not require any training data and can be used for other text domains as well.
Acknowledgements

My thanks go to Caroline Sporleder, Luisa Gödeke and Franziska Pannach, who advised me on the development of this work. I further thank the three anonymous reviewers for their valuable comments and their suggestions for how to improve this paper. This work was supported by the VolkswagenStiftung within the initiative “Geistes- und Kulturwissenschaften – digital: Forschungschancen, Methodenentwicklung und Reflexionspotenziale”.

References

Duygu Altinok. 2018. DEMorphy, German language morphological analyzer. arXiv:1803.00902.

Andreas Ammann. 2007. The fate of ‘redundant’ verbal forms – double perfect constructions in the languages of Europe. *STUF – Language Typology and Universals*, 60(3):186–204.

Howard I. Aronson. 1995. Towards a typology of verbal categories. In *New vistas in grammar: Invariance and variation*, pages 111–131. John Benjamins Publishing.

Karin Bausewein. 1991. AcI-Konstruktionen und Valenz. In Eberhard Klein, Françoise Pouradier Duteil, and Karl Heinz Wagner, editors, *Betriebslinguistik und Linguistikbetrieb*, number 260 in Linguistische Arbeiten, pages 245–251. Tübingen: Niemeyer.

Thomas Bögel, Jannik Strötgen, and Michael Gertz. 2014. Computational narratology: Extracting tense clusters from narrative texts. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 950–955, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Ronny Boogaart and Theo Janssen. 2007. Tense and aspect. In *The Oxford handbook of cognitive linguistics*.

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

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. Stanford typed dependencies manual. Technical report, Stanford University.

Dudenredaktion, editor. 2009. *Die Grammatik. Unentbehrlich für richtiges Deutsch*. Number 4 in Duden. Dudenverlag, 8th edition.

Jennifer R. Elliott. 2000. Realis and irrealis: Forms and concepts of the grammaticalisation of reality. *Linguistic Typology*, 4(1):55–90.

Joseph L. Fleiss, Bruce Levin, and Myunghee Cho Paik, 2003. *The measurement of intrarater agreement*, chapter 18. John Wiley & Sons, 3rd edition.

Jan Hajic, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Márquez, Adam Meyers, Joakim Nivre, Sebastian Pado, 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, June. Association for Computational Linguistics.

Erhard Hinrichs. 2016. Substitute infinitives and Oberfeld placement of auxiliaries in German subordinate clauses: A synchronic and diachronic corpus study using the CLARIN research infrastructure. *Lingua*, 178:46–70.

Linguistic Research in the CLARIN Infrastructure.

Daniel Jurafsky and James H. Martin. 2009. Features and unification. In *Speech and Language Processing*, chapter 15. Pearson, 2nd edition.

Bryan Jurish and Kay-Michael Würzner. 2013. Word and sentence tokenization with hidden Markov models. *Journal for Language Technology and Computational Linguistics*, 28(2):61–83.

Edward L. Keenan and Matthew S. Dryer. 2007. *Passive in the world’s languages*, volume 1, pages 325–361. Cambridge University Press, 2nd edition.
Appendix A. German verb forms with tense, mood, voice

| Aux.         | Example       | Tense + Aspect | Mood (if finite) | Voice     |
|--------------|---------------|----------------|-----------------|-----------|
| haben        | (zu) sehen    | present imperfect (infinitive) | active          |
| haben        | gesehen (zu) werden | present imperfect (infinitive) | dynamic passive |
| haben        | gesehen (zu) sein | present imperfect (infinitive) | static passive  |
| haben        | gesehen (zu) haben | present perfect (infinitive)  | active         |
| haben        | gesehen worden (zu) sein | present perfect (infinitive)  | dynamic passive |
| haben        | gesehen gewesen (zu) sein | present perfect (infinitive)  | static passive  |
| haben        | sehend        | present imperfect (participle) | active         |
| haben        | gesehen        | present perfect (participle)  | passive         |
| haben        | sieh           | present imperfect imperative | active         |
| haben        | werde gesehen | present imperfect imperative | dynamic passive |
| haben        | sei gesehen    | present imperfect imperative | static passive  |
| haben        | habe gesehen   | present perfect imperative   | active         |
| haben        | sei gesehen worden | present perfect imperative | dynamic passive |
| haben        | sei gesehen gewesen | present perfect imperative | static passive  |
| haben        | [er] sieht     | present imperfect indicative | active         |
| haben        | [er] sehe       | present imperfect present subjunctive | active |
| haben        | [er] wird gesehen | present imperfect indicative | dynamic passive |
| haben        | [er] werde gesehen | present imperfect present subjunctive | dynamic passive |
| haben        | [er] ist gesehen | present imperfect indicative | static passive  |
| haben        | [er] sei gesehen | present imperfect present subjunctive | static passive |
| haben        | [er] sah        | past imperfect indicative    | active         |
| haben        | [er] sähe       | past imperfect past subjunctive | active |
| haben        | [er] würde gesehen | past imperfect past subjunctive | dynamic passive |
| haben        | [er] war gesehen | past imperfect indicative    | static passive  |
| haben        | [er] wäre gesehen | past imperfect past subjunctive | static passive |
| haben        | [er] hat gesehen | present perfect indicative   | active         |
| haben        | [er] habe gesehen | present perfect present subjunctive | active |
| haben        | [er] ist gesehen worden | present perfect indicative | dynamic passive |
| haben        | [er] sei gesehen worden | present perfect present subjunctive | dynamic passive |
| haben        | [er] ist gesehen gewesen | present perfect indicative | static passive  |
| haben        | [er] sei gesehen gewesen | present perfect present subjunctive | static passive |
| haben        | [er] hätte gesehen | past perfect indicative | active         |
| haben        | [er] war gesehen worden | past perfect past subjunctive | dynamic passive |
| haben        | [er] wäre gesehen worden | past perfect past subjunctive | dynamic passive |
| haben        | [er] war gesehen gewesen | past perfect indicative | static passive  |
| haben        | [er] wäre gesehen gewesen | past perfect past subjunctive | static passive |
| haben        | [er] wird sehen | future imperfect indicative  | active         |
| haben        | [er] werde sehen | future imperfect present subjunctive | active |
| Aux. | Example | Tense + Aspect | Mood (if finite) | Voice |
|------|---------|---------------|-----------------|-------|
| haben | [er] würde sehen | future imperfect | past subjunctive | active |
| haben | [er] wird gesehen werden | future imperfect | indicative | dynamic passive |
| haben | [er] werde gesehen werden | future imperfect | present subjunctive | dynamic passive |
| haben | [er] würde gesehen werden | future imperfect | past subjunctive | dynamic passive |
| haben | [er] wird gesehen sein | future imperfect | indicative | static passive |
| haben | [er] werde gesehen sein | future imperfect | present subjunctive | static passive |
| haben | [er] würde gesehen sein | future imperfect | past subjunctive | static passive |
| haben | [er] wird gesehen haben | future perfect | indicative | active |
| haben | [er] werde gesehen haben | future perfect | present subjunctive | active |
| haben | [er] würde gesehen haben | future perfect | past subjunctive | active |
| haben | [er] wird gesehen worden sein | future perfect | indicative | dynamic passive |
| haben | [er] werde gesehen worden sein | future perfect | present subjunctive | dynamic passive |
| haben | [er] würde gesehen worden sein | future perfect | past subjunctive | dynamic passive |
| haben | [er] wird gesehen gewesen sein | future perfect | indicative | static passive |
| haben | [er] werde gesehen gewesen sein | future perfect | present subjunctive | static passive |
| haben | [er] würde gesehen gewesen sein | future perfect | past subjunctive | static passive |
| sein | (zu) gehen | present imperfect | infinitive | active |
| sein | gegangen (zu) werden | present imperfect | infinitive | dynamic passive |
| sein | gegangen (zu) sein | present imperfect | infinitive | static passive |
| sein | gegangen worden (zu) sein | present imperfect | infinitive | dynamic passive |
| sein | gegangen gewesen (zu) sein | present imperfect | infinitive | static passive |
| sein | gehend | present imperfect | participle | active |
| sein | gegangen | present imperfect | participle | pass |
| sein | werde gegangen | present imperfect | imperative | dynamic passive |
| sein | sei gegangen | present imperfect | imperative | static passive |
| sein | sei gegangen | present perfect | imperative | active |
| sein | sei gegangen worden | present perfect | imperative | dynamic passive |
| sein | sei gegangen gewesen | present perfect | imperative | static passive |
| sein | [er] geht | present imperfect | indicative | active |
| sein | [er] gehe | present imperfect | present subjunctive | active |
| sein | [er] wird gegangen | present imperfect | indicative | dynamic passive |
| sein | [er] werde gegangen | present imperfect | present subjunctive | dynamic passive |
| sein | [er] ist gegangen | present imperfect | indicative | static passive |
| sein | [er] sei gegangen | present imperfect | present subjunctive | static passive |
| sein | [er] ging | past imperfect | indicative | active |
| sein | [er] ginge | past imperfect | present subjunctive | active |
| sein | [er] wurde gegangen | past imperfect | indicative | dynamic passive |
| sein | [er] würde gegangen | past imperfect | present subjunctive | dynamic passive |
| sein | [er] war gegangen | past imperfect | indicative | static passive |
| sein | [er] wäre gegangen | past imperfect | present subjunctive | static passive |
| sein | [er] ist gegangen | present perfect | indicative | active |
| sein | [er] sei gegangen | present perfect | present subjunctive | active |
| sein | [er] ist gegangen worden | present perfect | indicative | dynamic passive |
| sein | [er] sei gegangen worden | present perfect | present subjunctive | dynamic passive |
| Aux. Example | Tense + Aspect | Mood (if finite) | Voice       |
|--------------|----------------|-----------------|-------------|
| sein [er] ist gegangen gewesen | present perfect | indicative | static passive |
| sein [er] sei gegangen gewesen | present perfect | present subjunctive | static passive |
| sein [er] war gegangen | past perfect | indicative | active |
| sein [er] wäre gegangen | past perfect | past subjunctive | active |
| sein [er] war gegangen worden | past perfect | indicative | dynamic passive |
| sein [er] wäre gegangen worden | past perfect | past subjunctive | dynamic passive |
| sein [er] war gegangen gewesen | past perfect | indicative | static passive |
| sein [er] wäre gegangen gewesen | past perfect | past subjunctive | static passive |
| sein [er] wird gehen | future imperfect | indicative | active |
| sein [er] werde gehen | future imperfect | present subjunctive | active |
| sein [er] würde gehen | future imperfect | past subjunctive | active |
| sein [er] wird gegangen werden | future imperfect | indicative | dynamic passive |
| sein [er] werde gegangen werden | future imperfect | present subjunctive | dynamic passive |
| sein [er] würde gegangen werden | future imperfect | past subjunctive | dynamic passive |
| sein [er] wird gegangen sein | future imperfect | indicative | static passive |
| sein [er] werde gegangen sein | future imperfect | present subjunctive | static passive |
| sein [er] würde gegangen sein | future imperfect | past subjunctive | static passive |
| sein [er] wird gegangen sein | future perfect | indicative | active |
| sein [er] werde gegangen sein | future perfect | present subjunctive | active |
| sein [er] würde gegangen sein | future perfect | past subjunctive | active |
| sein [er] wird gegangen worden sein | future perfect | indicative | dynamic passive |
| sein [er] werde gegangen worden sein | future perfect | present subjunctive | dynamic passive |
| sein [er] würde gegangen worden sein | future perfect | past subjunctive | dynamic passive |

Table 5: Composite verb forms in German. The first column shows the auxiliary verb used for the perfect aspect. An example for a verb using haben ‘have’ is sehen ‘see’; an example for a verb using sein ‘be’ is gehen ‘go’.