Two's a crowd: Recognizing multiply filled prefields in L2 German

Josef Ruppenhofer[1], Ines Rehbein[2], Matthias Schwendemann[3], Katrin Wisniewski[4], Torsten Zesch[1],
[1] FernUniversität in Hagen, Germany
[2] University of Münster, Germany
[3] Friedrich-Alexander-Universität, Erlangen-Nürnberg, Germany
[4] Leipzig University, Germany
Correspondence: josef.ruppenhofer@fernuni-hagen.de

Abstract

Cases of incorrect word order where learners place multiple constituents into the Vorfeld ('pre-field') of German clauses have been of keen interest to research on German as a second language. Detecting these is difficult because available parsers are not trained on ungrammatical L2 and often fail to correctly analyze the non-canonical structures in learner language. To address this problem, we hypothesize that we may need more and more varied training data for the parser to learn L2 structures that hardly exist in L1 German. We create synthetic data, both in a rule-based way, with tight control of the correctness of the resulting parses, and via LLM-based approaches, which can increase lexical diversity but also run the risk of introducing noise into the resulting trees. In the paper, we present an evaluation of the different approaches, using a verb placement task. While our results are mixed, we present an error analysis and discuss the main reasons why analyzing L2 syntax remains a challenge.

1 Introduction

In this paper, we examine to what extent we can use dependency parsing to identify syntactic structures that are marked in L1 (cf. ex. (1)) and usually ungrammatical in L2 German (cf. ex.(2)), namely cases where the prefield (Vorfeld) is occupied by two constituents (or, subtrees if viewed in terms of dependency grammar).

(1) [Alle Träume] [gleichzeitig] **lassen** sich nur selten verwirklichen . (Müller, 2013)
'It's rare that you can realize all (your) dreams at the same time.'

(2) Ja klar, wir mussen Zussamen zu treffen, aber [jetz] [ich] **habe** viel zu tun. (Merlin corpus)
'Yes sure, we have to meet up but at the moment I have a lot of things to do.'

Our interest in these structures arises in the context of studies on the acquisition of German as an L2. Second language acquisition (SLA) is known to depend on multiple factors, such as learners' age, L1 or educational background, which cause significant variation in learner language. While learners' *interlanguage* (IL) is highly variable, it is known to be systematic at the same time (Selinker, 1972). For that reason, finding stable developmental features that are mostly independent of such factors and of characteristics of the target language has been a key goal in researching L2 development. The concept of *developmental stages* (Meisel et al., 1981) is crucial in this regard. It refers to core grammar phenomena that are hypothesized to be acquired by learners in an ordered, step-like manner: some learners may be faster or slower than others, but stages cannot be skipped or re-ordered (so-called *implicationality* (Rickford, 2002)), not even through instruction (Pienemann, 1989; Arntzen et al., 2019).

In research on German, the phenomenon that has been most studied as a potential index of developmental stages is the acquisition of word order and, in particular, verb placement (Diehl et al. (2000); Gunnewiek (2000); Pienemann (2005); Jansen (2008); Jordens (1990); Baten and Håkansson (2015); *inter alia*) . For learners of German, knowing where to place verbal forms is a crucial challenge. Table 1 presents the word order patterns whose order of acquisition is studied by Processability Theory (PT) (Pienemann, 1998), an influential framework in research on the L2 acquisition of German.

Research into the validity and variation-independent stability of developmental stages is important as a theoretical concern within SLA since the notion of stages is not uncontroversial. Competing approaches exist which conceptualize variation as a driver of L2 development (e.g. Complex Dynamic Systems Theory (Larsen-Freeman, 2020)) and which tend to dispense with the idea of developmental stages, casting doubt on their universality once IL variation is fully reckoned with. Research on developmental

		(VVF)	VF	LSK	MF	RSK	NF
a	SVO		Ich	suche	eine neue Wohnung		
b	SVO	Aber	ich	suche	eine neue Wohnung		
c	ADV		Darum ich	suche	eine neue Wohnung		
d	SEP		Ich	muss	eine neue Wohnung	suchen	
e	SEP	Und	ich	habe	eine neue Wohnung	gesucht	in Bonn
f	INV		Darum	muss	ich eine neue Wohnung	suchen	
g	INV	Und	darum	muss	ich eine neue Wohnung	suchen	
h	V-END			weil	ich eine neue Wohnung	suche	
i	V-END			weil	ich eine neue Wohnung	gesucht habe	

Table 1: German word order types and topological fields

VF =Vorfeld (prefield); LSK=Linke Satzklammar (left sentence bracket);

MF = Mittelfeld (midfield); RSK = Rechte Satzklammer (right sentence bracket); NF = Nachfeld (postfield); VVF = Vorvorfeld (pre-prefield)

Example: Ich (*I*) suche (*search*) eine (*a*) neue (*new*) Wohnung (*flat*). Engl. translation: "I'm looking for a new flat."

stages also interacts with research on proficiency and language testing. In the case of German, the stages play a considerable role in real-world contexts: a number of learner assessments, such as Profilanalyse (Grießhaber, 2012; Heilmann, 2012; Grießhaber, 2019) or LiSe-DaZ (Schulz and Tracy, 2011; Schulz and Grimm, 2012) assume stages. Decisions based on these test instruments can have direct consequences on test-takers and other stakeholders. Complementarily, the stages are also used in grammar curriculum development (Winkler, 2014; Schroeder and Gamper, 2016; Baten, Kristof and Keßler, Jörg, 2019).

Given its theoretical and practical relevance, large-scale automated investigations of German verb placement as a potential indicator of acquisitional stage are needed. So far, many studies were conducted (semi-)manually on individual datasets. But with the recent conclusion of the DAKODA project (Schwendemann et al., 2025b), a large harmonized collection of corpora has become available. However, the automatic processing capabilities for learner language are still somewhat lagging. Among the word orders shown in Table 1, ADV with multiple elements in the prefield is the one that is hardest to recognize in a parse-based approach because, unlike the other word order constellations, it represents a structure that exists barely, if at all, in the L1 German treebanks (cf. Appendix F). In fact, in the acquisition literature ADV is often described as ungrammatical for L1 German, even though there are exceptions: Müller (2003, 2005) and Schalowski (2015) discuss exceptions in written L1 German while Schalowski (2012) and Wiese and Müller (2018) focus on spoken German. Unsurprisingly, parsers trained on L1 news data usually combine the multiple subtrees in VF into a single sub-tree (constituent), thereby obscuring the evidence for ADV. Not only is ADV rare to non-existent in L1 training data, it is also relatively rare in available learner corpora because it is used by early learners who are not well represented in many corpora and who are believed to unlearn or abandon ADV once they master the INV order.[1] However, for validating claims about acquisition sequences it is crucial to be able to recognize instances of ADV with good precision and recall.

To improve parsing for word order analysis, we investigate to what extent we can use synthetic data to improve performance on data from learners of German as a second language. Note that we focus on a parsing-based approach to word order analysis rather than, for instance, a sequence tagging approach because we want to be able to display the basis for our labels to SLA researchers and/or teachers in a linguistically explicit way. Last but not least, a parser trained on such modified data, if its performance on L1 remains undegraded, may also be useful for studying complex L1 word order phenomena, especially L1 ADV.[2]

2 Related work

2.1 Word order and stage-like acquisition

Ruppenhofer et al. (2024) provide extensive discussion of how to operationalize Processability Theory in a way that allows for automatic large-scale cross-corpus investigation of verb placement in learner

[1] PT theorizes that learners use ADV because they have understood that the sentence-initial position is a prominence position but have not yet acquired the single-occupancy constraint on the prefield. Once they do, they move the subject past the finite verb and use proper INV.

[2] We make our code available at URL-HERE. As our annotated data includes documents from corpora with restrictive licensing, we make it available to academic users with authentication provided by EduGain at `https://dakoda.org/restricted_data/index.de.html`.

data. We adopt this operationalization here and briefly review some main points. The authors note that the set of verb placement types discussed by PT represents a partial theory. Certain German clause types are out of scope altogether, among them Yes/No questions and imperatives (as typically verb-initial sentences) as well as all non-finite clauses, even though all are subject to clear verb placement restrictions. For instance, all non-finite clauses in German exhibit V-END (Wöllstein, 2010). Further, Ruppenhofer et al. (2024) note that while PT-based studies do not usually explicate it, the German verb placement types it studies rely on the notion of topological fields (Table 1). According to topological field theory, sentences consist of a sequence of slots (fields) which can contain certain elements (though not others) and which need not always be filled. For instance, in main clauses the finite verb (whether lexical, a tense auxiliary or a modal) is said to occupy the left sentence bracket (Linke Satzklammer (LSK)), whereas any non-finite verb sits in the right sentence bracket (Rechte Satzklammer (RSK)). In subordinate clauses, the finite verb is in the right bracket, potentially preceded by non-finite verb forms. To differentiate among the non-VEND placement constellations, one needs to consider the grammatical relations that the elements occupying the slots bear towards the verb. In SVO, for instance, the subject is in the Vorfeld ('pre-field', VF) and the object in the Mittelfeld ('midfield', MF). In the case of INV, the subject is in MF and some other element occupies the VF slot. In ADV, learners place more than one constituent (or sub-tree root) into VF, which is not allowed in L1 German, whereas multiple occupancy is acceptable for MF and NF. SEP requires that both brackets be occupied and that MF is non-empty. Finally, the types are not all mutually exclusive. In Table 1, examples (f) and (g) show that SEP and INV can co-occur, whereas (d) and (e) show that SEP can co-occur with SVO.

2.2 Word order errors in learner language

While PT and the specifications by Ruppenhofer et al. (2024) focus on the order of acquisition of different verb placement types by learners, other work looks at whether observed verb placement is grammatical for L1 German.

Michael and Horbach (2025) introduce linguistically informed verb placement error detection datasets for learners of Germanic languages. They train multilingual BERT (mBERT) models on synthetic data and show that linguistically informed error induction on Universal Dependencies (UD) treebanks results in more effective models for verb placement error detection than models trained on naively introduced errors. We also construct synthetic data in a linguistically informed fashion on the basis of UD treebank data. However, we restrict ourselves to German and we perform order corruption in a slightly different way (see section §5). Finally, while Michael and Horbach (2025) label verb instances in a binary fashion as correctly or incorrectly placed, we use the more fine-grained descriptive multi-class labels provided by PT, which - with the possible exception of ADV - capture only grammatical uses. For instance, while *hat* (EN *has*) in (3) represents a verb placement error, it receives no verb placement type label from the PT inventory as it represents none of the types as defined.[3]

(3) Er sagt, dass er **hat** keinen Hunger.
He says that he has no hunger.
'He says that he's not hungry.

2.3 Dependency parsing of learner data

Other work very closely related to ours is Masciolini et al. (2024) who generated synthetic word order errors for Swedish L2-data and used them to fine-tune a number of dependency parsers.[4] These authors evaluated parser performance on (1) original L1 data, (2) L1 data modified with synthetic error data, and (3) a small set of 69 authentic L2 sentences containing word errors of the relevant kinds (for Swedish). The authors report that improvement on the L2 data was small, but significant. They found no substantial decrease in performance on normative data.

In our work, we use synthetic data to improve verb placement analysis for German L2. We focus on parsing-based approaches to obtain linguistically informative analyses for feedback, but also compare results to a shallow sequence tagging approach based on a BERT (Devlin et al., 2019) classifier.

3 Classifiers and data

To improve parsing performance for non-canonical learner language and, in particular, for dealing with multiple occupants in the prefield, we train parsing models on a variation of L1, L2 and synthetic data.

[3] While it looks like SVO, SVO word order in German does not allow for subordinated clauses with *dass* (that).

[4] Word order constellation in Swedish is different from German, see, e.g., Masciolini et al. (2024) for details.

3.1 Parsing models

Our dependency parser is based on spaCy[5] (Honnibal et al., 2020), trained on perturbed Universal Dependency (UD) data (cf. §5.1 and §5.2). For the purpose of training, we treat topological features as morphological `feats`, i.e. they are placed in column 6 of the CoNLLU format.

3.2 Verb placement classifiers

We experiment with (1) a system using hand-crafted rules based on UD parse features to predict verb placement, (2) a feature-based ensemble classifier using UD parse tree-derived features and (3) a BERT-based classifier that only uses the surface strings of sentences/clauses. All systems can predict multiple labels for a finite verb instance.

3.3 Datasets

We use a collection of L1 and L2 datasets of varying sizes for parser training and evaluation of (i) parsing performance and (ii) verb placement identification (see Appendix A, Table 9). We also use parts of the data as seeds for deriving synthetic training data for both tasks.

3.3.1 L1 data

Parsing data We use the UD version of the TüBa-D/Z treebank (Telljohann et al., 2006) to train a `base model` for L1 German. We also use parts of the treebank for deriving synthetic data. We choose TüBA-D/Z for two reasons. First, it is large and its textual range is broad as it incorporates data from the *taz* newspaper, which is known to be more colloquial and varied than other German newspapers. Second, the data comes with topological annotations that can aid verb placement analysis. We also use a test set of TüBa-D/Z trees (TubaTest) to evaluate to what extent L2-adapted parsers lose performance on L1 data.

Verb placement data Our first dataset is the topological test battery, TopoBatt, released by Ruppenhofer et al. (2024), which was designed to illustrate PTs verb placement types in varied sentence types. TopoBatt contains sentences drawn from the examples in Wöllstein (2010)'s book on topological fields, sentences from the L1 HDT treebank, and HDT sentences that were manually modified to exhibit ADV. We use this dataset to evaluate the performance of a rule-based verb placement classifier on data that represents L1 German - with the exception of the artificial ADV sentences.

3.3.2 L2 data

Parsing data We constructed a small dataset of German learner data where we manually checked and corrected automatic UD parses.[6] The data in L2Trees represent 59 randomly sampled documents with 637 sentences from 14 different corpora in the DAKODA repository. The finite verb instances are also annotated for verb placement type.

Verb placement data Our L2 data is taken from several learner corpora of the DAKODA repository (cf. Table 10 in Appendix A). They represent samples of learner data where, for finite verb instances, the verb placement type was manually annotated. The L2 data comes in two subsets (`CEFR`, `Diversification`). The `CEFR` sample is a superset of the data released as part of Ruppenhofer et al. (2025),[7] which was constructed to represent most of the levels defined in the Common European Framework of Reference for Languages (CEFR).[8,9] The lower three levels (A1, A2, B1) are sampled from the Merlin corpus (Boyd et al., 2014), levels B2 and C1 from Disko (Wisniewski et al., 2022). Level C2 was omitted since Merlin has only 4 documents at this level. Note that the CEFR ratings apply to the texts, not the learners. Ruppenhofer et al. (2025) report a Fleiss' κ agreement of .83 for verb placement annotation on CEFR.

The Diversification sample draws on six corpora that are not part of the CEFR sample, with the goal of capturing a greater variety of task settings and also including some spoken L2 data (BMAT L2 (Sauer and Lüdeling, 2016), MULT_S (Schroeder et al., 2015)). Unlike the CEFR data, the texts in the Diversification sample have no CEFR ratings. Schwendemann et al. (2025a) reported κ between two coders above .93 for verb placement on this set.

Our primary evaluation here concerns the word order types: after learning parsing models on synthetic data, can we recognize all word order types on

[5] We use spaCy version 3.8.11. with `bert-base-german-cased` to initialize the parser.

[6] The parses represent something like a silver standard. This is so because some learner productions are hard to understand and because the data was checked by only one person, who did, however, have prior experience with UD-based annotation.

[7] While the original annotation covers some documents incompletely, we fully annotated all documents sampled for each corpus.

[8] See the CEFR website for details.

[9] The German rendering is *Gemeinsamer Europäischer Referenzrahmen für Sprachen* (GER).

Category	Precision	Recall	F1-Score	Support
ADV	80.3	85.5	82.8	62
INV	97.6	92.4	94.9	262
SEP	91.8	97.0	94.4	267
SVO	97.1	90.7	93.8	335
VEND	100.0	86.0	92.5	214

Table 2: Manual rules on TopoBatt (L1/synthetic)

Category	Precision	Recall	F1-Score	Support
ADV	26.9	82.4	40.6	17
INV	90.7	81.0	85.6	205
SEP	93.7	97.5	95.5	242
SVO	97.7	90.2	93.8	461
VEND	97.0	91.1	94.0	214

Table 3: Manual rules on L2Trees

the L2 data in a parse-based approach? Note that, beyond ADV, L2 data contains additional orthographic and morpho-syntactic errors and unusual structures that might impinge on parse quality.

4 Rule based verb placement on L1

The first question we need to answer is: *How well does rule based verb placement work on well-parsed L1 data?* Towards that end, we apply our rule-based verb classifier (1) to the L1 sentences in TopoBatt and (2) the L2 sentences in L2Trees.

Table 2 shows results on the TopoBatt data, indicating that on well-parsed L1 data the assignment of verb placement types is a solvable problem. ADV still proves harder than the other types but is at quite high a level already. Table 3 shows that results on learner data are also quite high except for ADV where performance is significantly lower. While the true number of ADV instances in this L2 data is very small, it is notable that precision is very low as the rules predict quite a few false positives. While the data in L2Trees may well be somewhat different from the other L2 datasets we use, it seems reasonable to expect a low ceiling on performance.[10]

5 Generating synthetic data

Synthetic data is commonly used in the field of grammatical error correction where authentic L2 data is sparse. Many kinds of corruptions are possible. We focus on ADV, as our main motivation is the lack of available training data for this structure.

5.1 Rule-based generation of L2 data

Our approach is conceptually similar to Masciolini et al. (2024) who mostly follow an earlier approach that swaps subtrees around roots (Şahin and Steedman, 2018). However, they also randomly swap adjacent tokens to some extent. In our own approach, we forgo such swaps as they seem to hardly ever occur in the L2 data we have. (On this point, also see Michael and Horbach (2025)). We only move whole constituents/subtrees.

We create ADV instances as follows. Given INV instances, we move the post-verbal subject from the midfield to the left into VF. Given SVO instances, we move an item from the post-verbal midfield in front of the subject, which also results in ADV. Note that this modification of SVO would not be allowed in the approach of Michael and Horbach (2025), where the relative order of NPs is always preserved. In addition to creating ADV instances, we also experiment with modifying the post-verbal midfield and, if present, the post-field to the right of the right bracket. This is of interest as learners also produce non-canonical orders in those fields which may be hard for L1-trained parsers to deal with. As pure MF manipulation, we swap random consecutive constituents. We manipulate NF in two ways. When it is empty and MF is filled, we move a random item from MF into NF. When NF contains a clause, we move it to MF.[11] Complementing the movement of items, we also modify parses by removing non-final punctuation, in particular commas. L2 data often omits commas and we would like parsers to be less sensitive to their absence.

We applied the above modifications in two settings. In the **basic** setting (RuleB), applicable modifications were allowed to fire once. In the **exhaustive** setting (RuleE), we generated all possible variants. For instance, in cases of SV followed by multiple arguments and/or modifiers, we produced separate ADV instances by moving each of the post-verbal elements to the front.

5.2 LLM-based generation of L2 data

Given the success of generative AI models for data generation (Liu et al., 2024; Nadăş et al., 2025), we explore two LLM-based approaches to generate synthetic L2 data for parser training.

One-Shot We use seed sentences from our L1 treebank and prompt the model to modify the data so that it matches the target ADV word order. The seed data is filtered for length and POS similarity in order to match the linguistic properties of our target

[10] We also tested a classical ML approach, with results similar to those of our rule-based approach (see Appendix D).

[11] Rarely, there can be multiple clauses in NF. In that situation, we randomly pick one.

	Set	synth.	# trees
[1]	BASE parser	✗	104,787
[2]	RULEB	✓	1100
[3]	RULEE	✓	10314
[4]	LLM_ONE-SHOT	✓	4419
[5]	LLM_MODIFY	✓	2255

Table 4: Parser training data subsets and their properties.

data.[12] We test a one-shot setting where we present the LLM with an example tree with canonical word order and a modified tree in ADV word order and prompt it to modify a new tree accordingly. The modified example tree is randomly selected from a set of five modifications, all following ADV order. The prompt is included in Appendix B, Fig.1.

Modify In our second approach, we present the LLM with a non-canonical tree with ADV word order that was generated by the rule-based approach (§5.1). We then prompt the model in a zero-shot setting to modify *the semantics of the tree* while preserving its syntactic structure. We hypothesize that this might help to increase diversity in the synthetic data. The approach also addresses one of the limitations of the basic rule-based setting which can only provide us with a small number of new trees for ADV. For the prompt template, please refer to Fig.3 in Appendix B.

As both the One-Shot and the Modify approach create some invalid output, we drop all trees that exhibit one of the following problems: (i) incorrect number of columns, (ii) invalid UD or STTS POS label, (iii) invalid dependency relation, (iv) dependency cycle, or (v) multiple roots. For the MODIFY approach, these checks eliminated about 2/3 of the generated trees, for ONE-SHOT around 1/3.

Syn-All We also test whether the rule-based and LLM-based methods can complement each other by combining the controlled generation of the rule-based approach with the LLM-generated trees.

Models We experiment with two open-weight models for tree generation. **Qwen3-30b-A3B-Instruct-2507** is a mixture-of-experts model by Alibaba with 30 billion parameters, tuned for instruction-following, conversation, and general tasks. **Gpt-oss-120B** is a general purpose model by OpenAI and, with 120 billion parameters, significantly larger. We use Gpt-oss-120B for ONE-SHOT and Qwen3-30b-A3B-Instruct-2507 for the MODIFY approach.

[12] We use automatically predicted PoS in the target data to compute similarity.

Eval data	Model	UAS	LAS	Topo
trained on TüBa-D/Z, evaluated on TüBa-D/Z				
TüBba-D/Z	de Kok 2016	92.4	90.0	n/a
TüBba-D/Z	TüBa-base	90.5	89.3	95.9
L2Trees	TüBa-base	88.7	85.0	n/a
base TüBa fine-tuned on synthetic, evaluated on TüBa-D/Z				
TüBba-D/Z	RULEB	88.8	87.3	94.9
TüBba-D/Z	RULEE	**89.7**	**88.3**	95.4
TüBba-D/Z	ONE-SHOT	88.9	87.1	95.3
TüBba-D/Z	MODIFY	89.3	87.7	**95.7**
TüBba-D/Z	SYN-ALL	89.1	87.5	93.8
base TüBa fine-tuned on synthetic, evaluated on L2 data				
L2Trees	RULEB	87.7	83.5	n/a
L2Trees	RULEE	**88.5**	**84.6**	n/a
L2Trees	ONE-SHOT	87.3	82.7	n/a
L2Trees	MODIFY	87.8	83.9	n/a
L2Trees	SYN-ALL	87.9	84.2	n/a

Table 5: Parsing results on TüBa-D/Z (L1) and L2Trees.

The above two models were selected and matched to the tasks after we tested a set of four open-weight LLMs on a small development set of 25 trees and asked the models to generate new trees in each of our two settings, that is the ONE-SHOT approach and the MODIFY approach.[13] The other open-weight models we considered were from the Mistral and Llama model families. **Mistral-Small-24B-Instruct-2501** is a fast and efficient general-purpose model from MistralAI with 24 billion parameters. However, we found that on our task the model outputs many additional comments on the task and on what it intended to do (mostly in English) but failed to generate valid trees in both settings. **Llama-3.3-70B**, developed by Meta, is slightly older but relatively large with 70 billion parameters. Despite its size, the model showed poor performance on our task. It produced many extraneous comments (despite being told to output only the trees) and often deviated from the CoNLL-U format.

6 Experiments & Evaluation

We now train parsing models on different combinations of L1 and synthetic data to investigate whether we can improve results especially for identifying ADV word order with multiple prefield elements.

6.1 Experimental settings

We first train a base parser on the TüBa-D/Z L1 data (BASE). We then continue training that model further with four sets of synthetic data (cf. [2]-[5]

[13] For better performance it may be desirable to explore many more models. However, for this study we wanted to use open source models and we were also constrained to use models that were or could be hosted on the hardware available to us at our institutions.

Model	Cat	Prec	Rec	F1	Support
BASE	ADV	41.5	37.0	39.1	46
RULEB	ADV	43.1	54.4	48.1	46
RULEE	ADV	48.9	50.0	49.5	46
ONE-SHOT	ADV	45.1	50.0	47.4	46
MODIFY	ADV	41.8	50.0	45.5	46
SYN-ALL	ADV	42.9	65.2	51.7	46
BASE	INV	93.4	91.5	92.4	294
RULEB	INV	93.0	90.5	91.7	294
RULEE	INV	94.1	91.5	92.8	294
ONE-SHOT	INV	93.4	91.5	92.4	294
MODIFY	INV	94.0	91.2	92.6	294
SYN-ALL	INV	94.0	89.8	91.8	294
BASE	SEP	91.0	94.7	92.8	299
RULEB	SEP	89.8	94.3	92.0	299
RULEE	SEP	89.9	95.0	92.4	299
ONE-SHOT	SEP	91.8	94.0	92.9	299
MODIFY	SEP	91.9	94.3	93.1	299
SYN-ALL	SEP	91.0	94.7	92.8	299
BASE	SVO	96.7	66.0	78.5	444
RULEB	SVO	96.7	66.4	78.8	444
RULEE	SVO	96.4	66.0	78.3	444
ONE-SHOT	SVO	96.1	66.9	78.9	444
MODIFY	SVO	95.8	66.4	78.5	444
SYN-ALL	SVO	95.8	66.0	78.1	444
BASE	VEND	97.7	91.4	94.5	233
RULEB	VEND	97.2	88.8	92.8	233
RULEE	VEND	95.5	91.0	93.2	233
ONE-SHOT	VEND	97.2	89.7	93.3	233
MODIFY	VEND	98.1	89.7	93.7	233
SYN-ALL	VEND	96.8	90.6	93.6	233

Table 6: Results for models trained on synthetic data and applied to the Diversification sample.

in Table 4). Alongside each of the four synthetic datasets, we use 500 sentences produced by RULEB (see §5.1) as our development set. To evaluate verb placement, we use the manual rules.

Parsing accuracy was evaluated using standard measures, Labelled and Unlabelled Attachment Scores (LAS , UAS). Punctuation tokens are excluded from the parsing scores. We assess the quality of topological field prediction in terms of F1.[14] We report the performance of verb placement classification per label.

6.2 Parsing results

Our BASE parser yields results similar to the ones reported by de Kok and Hinrichs (2016) on the in-domain data (see Table 5).[15] After continued fine-tuning on different sets of our synthetic data, we observe best results for both evaluation datasets for parsing models fine-tuned on the RULEE dataset, with accuracies only slightly lower than when training on the original UD version of TüBa-D/Z.

[14]Topological fields are scored only on the TüBa data because they were not manually labeled on L2trees.

[15]Note that de Kok and Hinrichs (2016) used a different parser architecture, made use of an earlier version of the treebank and had different, unpublished data splits.

6.3 Verb placement results

Tables 6 and 7 show results for rule-based verb placement analysis on the `Diversification` sample. Note that in Table 6 each row represents a version of the dataset parsed by a specific parser model (Table 4, [1]-[5]).

We can see some benefit for each of the models trained with synthetic data, with the largest increase for ADV shown by the rule-exhaustive (RULEE) data. Notably, both models trained on the rule-based synthetic data outperform the models trained on the llm-generated synthetic data (ONE-SHOT, MODIFY). For the other verb placement categories, the picture is slightly different. Here the results for the models trained on the LLM-generated data are in the same range or slightly higher than for the models trained on the rule-based data.

6.4 Shallow verb placement prediction

We now ask whether a shallow tagging approach can outperform our parsing-based approach to predict verb placement in German L2. For this, we train a BERT sequence tagger in a multi-label setting to predict labels for each finite verb in the data. We use the gold POS information to mark finite verbs in the input. The model then learns to predict one or multiple verb placement labels for each finite verb in the input sequence (see Appendix C).

We train the model on CEFR, augmented with 1,000 instances generated by the rule-based approach. We selected trees generated with rules that target the ADV class and add labels for the other classes in a self-training setup (see App. C).

We can see that this approach also yields results for ADV above 50% F1 and additionally increases results for the other verb placement classes (Table 7), in particular for SVO and VEND. However, the results have to be taken with a grain of salt as we observe considerable variation in results, depending on the number of sentences that we add (see results for different settings in the Appendix, Table 15). To test whether results would generalize to new data, we would need another evaluation set for L2 data. As pointed out by an anonymous reviewer, the instability of the BERT classifier with increasing synthetic training data may not only reflect generalizability issues, but also the effect of noise from synthetic examples, and it may be necessary to search for optimal training examples.

Model	Cat	Prec	Rec	F1	Support
BERT	ADV	38.0	76.1	50.7	46
BERT	INV	94.5	94.9	94.7	292
BERT	SEP	93.1	95.3	94.2	299
BERT	SVO	95.1	97.3	96.2	443
BERT	VEND	99.1	97.4	98.2	232

Table 7: BERT results (F1) on the Diversification sample for a model trained on CEFR+1,000 synthetic instances.

Model	ADV	INV	SEP	SVO	VEND
BASE	48.7	93.6	92.4	82.2	95.5
RULEB	37.4	90.9	91.7	81.3	96.2
RULEE	37.8	92.6	93.1	82.8	95.3
ONE-SHOT	38.3	92.5	92.1	81.2	95.5
MODIFY	41.0	92.8	92.3	81.4	95.1
SYN-ALL	41.2	91.7	92.6	82.0	95.4
Support	35	431	347	533	401

Table 8: Results for models trained on the Diversification sample + synthetic data and applied to CEFR.

6.5 Robustness of results

To test how robust our results are, we also apply and evaluate our rule-based classifier on the CEFR dataset (Table 8). Surprisingly, for predicting the ADV class, using any version of the data parsed by a model trained on synthetic data is worse than using the BASE parser. We also observe that the parsers trained on the llm-generated data show a smaller performance drop than the rule-based models. We show further mixed results for the rule-based classifier applied to L2Trees in Appendix E.

7 Error analysis

To find out more about the reasons for the mixed results, we perform error analysis on the rule-based predictions applied to versions of the CEFR data parsed by the spaCy BASE model and the model finetuned on RULEE. In terms of error types, we observe the same kinds of errors for both versions. For ease of exposition, we discuss the major error types grouped by whether they affect recall or precision.

Recall errors Several error types reduce recall. One issue is that some tokens in the parser output lack a TopoField value or have the value `null` rather than any of the standard topological field labels. Given that the rules only inspect the elements in the regular fields but do not look at tokens without a value or with the null value, this leads to misses if a token at the left periphery without a TopoField value is one that should have been assigned to VF.[16]

Another source of errors are incorrect parses of learner sentences with multiple L1-atypical structures. For instance, in (4), the subordinate *als*-clause – which lacks the expected verb-last word order – is treated as the main clause to which the actual main clause is conjoined. As a result, ADV in the main clause is not discoverable.

(4) [Als du hat im Brief geschrieben, dass ich bald Geburtstag habe], [vielleicht] **wünsche** ich mir eine neue Tier haben in diesem Jahr.
'Since you mentioned in your letter that my birthday is coming up soon, maybe I'll ask for a new pet this year.'

In a few cases the gold verb placement annotation seems wrong, in others it seems to be at least debatable because the context is ambiguous. In (5), the *mit*-PP was considered to depend on the right clause (*distanzieren*) for the verb placement annotation, whereas the parser associates it with the first clause (*spielen*). We prefer the first interpretation but cannot rule out the second.

(5) Technologie spielt auch eine große Rolle , [mit den neuen Smartphones , Internet überall] , [die interpersonelle Beziehungen] **distanzieren** sich noch mehr.
'Technology also plays a major role; with new smartphones and ubiquitous internet access, interpersonal relationships are becoming even more distant.'

Precision errors (6) represents a repair or syntactic amalgam as it contains two instances of the copula *sind* around the subject. Both copulas are attached to the root *Teil* with the relation `cop` and there is no use of the UD relation `reparandum` to subordinate the first instance of *sind* to the second one.[17] As a result there are two distinct subtrees with VF labels: *natürlich* and *alle diese Vorteile*.

(6) Aber natürlich sind alle diese Vorteile sind nur ein kleines Teil des Lebens.
'But of course, all these benefits are just a small part of life.'

Another significant subset of errors results from the fact that clauses and related pronominal adverbs may co-occur together as unrelated items in TüBa-

[16] We noticed only after the experiments were concluded that null or missing TopoField values already occur in the Tüba D/Z gold standard. Potentially, one could try to filter out trees where this issue occurs, though this would of course also reduce the amount of training data.

[17] This is unsurprising since Tüba D/Z, as an L1 written treebank, lacks even a single instance of this relation.

D/Z (UD). In (7), the *wenn*-clause and *dann* are treated as two separate items in VF.

(7) [Wenn man die Grafik betrachtet] , [dann] **kann** man erkennen , dass …
'If you look at the graph, you can see that...'

The rules recognize these structures as cases of multiply filled pre-fields but the gold annotation did not treat them as ADV because the annotation of ADV on the learner data focused on L1-inappropriate cases, which examples like (7) are not.

Summary There are a few ways in which to address the above errors. First, we could track ambiguous cases and score them more leniently. Second, we could fix errors in the gold standard. We can also either let L1-appropriate instances like (7) bear the gold label ADV or modify the rules so they get ignored. We might also modify the rules without too much to handle tokens with missing or null topological values on the left periphery. However, for cases like (4) and (6), trying to detect and compensate for incorrect parses with something like *mal-rules* (Schneider and McCoy, 1998) seems quite difficult. It might be preferable to collect / create new naturalistic or synthetic data that exemplify more of the structures that are atypical in written L1 and whose presence negatively impacts parse quality in L2 (and probably also in spoken L1) data.

8 Conclusion

We studied the use of synthetic data in training parsing models that can support linguistically grounded approaches to recognizing verb placement in German learner data. While performance seems adequate for four of the placement types, the ADV constellation with its multiply filled prefield cannot be reliably recognized. Both the lack of instances in L1 training data and the rarity of the phenomenon in available L2 corpora conspire to make recognizing instances of ADV a very hard task. While we could increase ADV recall significantly on some datasets, precision was mostly quite poor. To gain better performance for ADV on our parsing-based approach, we need more robust assignment of topological fields and greater robustness against a broader range of L2 errors that impinge on the analysis of pre-field elements via knock-on effects. Synthetic data that only models ADV in otherwise L1-like sentences does not get us close enough. Finally, the parsing-based approach to verb placement classification as well as the BERT-based approach may both benefit from trying to identify high-quality trainig examples.

Limitations

The annotated datasets we have available for verb placement analysis in learner language are still relatively small, especially when it comes to our focal ADV category, the case of multiply filled pre-fields. However, there is realistically not much that can be done in the near term to increase the number of naturalistic L2 samples.

It would be quite sensible to carry out similar studies for other Germanic languages that have multiple word orders and where learner (and spoken) language may contain finite clauses with multiple pre-verbal constituents (e.g. Swedish V3).

Equally, it would make sense to investigate the use of multiply filled pre-fields in spoken and written L1 German as a point of comparison. We suspect that the distribution of ADV types might be quite different, at least relative to the written L1 cases discussed by (Müller, 2003). Comparisons with naturalistic spoken L1 face the difficulty that little such material is publically available and, more importantly, that it is usually structured into contributions or turns rather than sentences, which are needed for a parse-based analysis.

References

Andrea Abel, Aivars Glaznieks, Lionel Nicolas, and Egon Stemle. 2016. An extended version of the KoKo German L1 Learner corpus. In *Proceedings of the Third Italian Conference on Computational Linguistics CLiC-It 2016*, pages 13–18, Torino, Italy.

Andrea Abel, Aivars Glaznieks, Lionel Nicolas, and Egon W. Stemle. 2014. Koko: an l1 learner corpus for german. In *International Conference on Language Resources and Evaluation*.

Ragnar Arntzen, Jörg-U Keßler, Arnstein Hjelde, and Gisela Håkansson, editors. 2019. *Teachability and learnability across languages*. John Benjamins Publishing Company.

Kristof Baten and Gisela Håkansson. 2015. The development of subordinate clauses in German and Swedish as L2s : a theoretical and methodological comparison. *Studies in Second Language Acquisition*, 37(3):517–547.

Baten, Kristof and Keßler, Jörg. 2019. Research timeline : the role of instruction : teachability and processability. In Arntzen, Ragnar and Håkansson, Gisela and Hjelde, Arnstein and Keßler, jörg, editor, *Teachability and learnability across languages*, volume 6 of

Processability Approaches to Language Acquisition Research & Teaching, pages 9–26. John Benjamins.

Adriane Boyd, Jirka Hana, Lionel Nicolas, Detmar Meurers, Katrin Wisniewski, Andrea Abel, Karin Schöne, Barbora Štindlová, and Chiara Vettori. 2014. The MERLIN corpus: Learner language and the CEFR. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1281–1288, Reykjavik, Iceland. European Language Resources Association (ELRA).

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Erika Diehl. 1998. Vom Deutschunterricht zum Deutscherwerb? Das Genfer DiGS-Projekt [From German Class to German Language Acquisition? The DiGS Project in Geneva]. *Deutsch als Fremdsprache*, 35(3):162–167.

Erika Diehl, Helen Christen, and Sandra Leuenberger. 2000. *Grammatikunterricht: Alles für der Katz? Untersuchungen zum Zweitsprachenerwerb Deutsch [Teaching Grammar: All For Naught? Investigations into the Second Language Acquisition of German]*, 1. edition. Niemeyer, Tübingen.

Aivars Glaznieks, Jennifer-Carmen Frey, Andrea Abel, Lionel Nicolas, and Chiara Vettori. 2023. The Kolipsi Corpus Family: Resources for Learner Corpus Research in Italian and German. *Italian Journal of Computational Linguistics*, 9(2):24.

Aivars Glaznieks, Jennifer-Carmen Frey, Maria Stopfner, Lorenzo Zanasi, and Lionel Nicolas. 2022. Leonide. *International Journal of Learner Corpus Research*, 8(1):97–120.

Wilhelm Grießhaber. 2012. Die Profilanalyse [the profile analysis]. In Bernt Ahrenholz, editor, *Einblicke in die Zweitspracherwerbsforschung und ihre methodischen Verfahren.*, DaZ-Forschung, pages 173–194. De Gruyter.

Wilhelm Grießhaber. 2019. Profilanalysen [profile analyses]. In Stefan Jeuk and Julia Settinieri, editors, *Sprachdiagnostik Deutsch als Zweitsprache*. De Gruyter Mouton.

Lisanne Klein Gunnewiek. 2000. *Sequenzen und Konsequenzen: zur Entwicklung niederländischer Lerner im Deutschen als Fremdsprache [Sequences and consequences: On the Development of Dutch Learners of German as a Foreign Language]*. Rodopi.

Beatrix Heilmann. 2012. *Diagnostik & Förderung - leicht gemacht. Deutsch als Zweitsprache [Assessment & Support — Made Easy. German as a Second Language]*. Deutsch als Zweitsprache in der Grundschule. Klett Sprachen.

Hagen Hirschmann and Andreas Nolda. 2020. Dulko - auf dem weg zu einem deutsch-ungarischen lernerkorpus [dulko — toward a german-hungarian learner corpus]. In Ludwig M. Eichinger and Albrecht Plewnia, editors, *Neues vom heutigen Deutsch. Empirisch - methodisch - theoretisch*, number | 2018 | in Jahrbuch / Institut für Deutsche Sprache, page 339 – 342.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Louise Jansen. 2008. Acquisition of German Word Order in Tutored Learners: A Cross-Sectional Study in a Wider Theoretical Context. *Language Learning*, 58(1):185–231.

Peter Jordens. 1990. The acquisition of verb placement in Dutch and German. *Linguistics*, 28(6):1407–1448.

Katharina Karges, Thomas Studer, and Nina Selina Hicks. 2022. Lernersprache, Aufgabe und Modalität: Beobachtungen zu Texten aus dem Schweizer Lernerkorpus SWIKO [Learner Language, Task, and Modality: Observations on Texts from the Swiss Learner Corpus SWIKO]. *Zeitschrift für germanistische Linguistik*, 50(1):104–130.

Christine Köhn and Arne Köhn. 2018. An annotated corpus of picture stories retold by language learners. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 121–132. Association for Computational Linguistics.

Diane Larsen-Freeman. 2020. Complex Dynamic Systems Theory. In Bill VanPatten, Gregory D. Keating, and Stefanie Wulff, editors, *Theories in Second Language Acquistion*, pages 248–270. Taylor & Francis.

Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, and Andrew M. Dai. 2024. Best Practices and Lessons Learned on Synthetic Data. In *First Conference on Language Modeling*, COLM.

Anke Lüdeling, Artemis Alexiadou, Shanley Allen, Oliver Bunk, Natalia Gagarina, Sofia Grigoriadou, Rahel Gajaneh Hartz, Kateryna Iefremenko, Esther

Jahns, Kalliopi Katsika, Mareike Keller, Martin Klotz, Thomas Krause, Annika Labrenz, Maria Martynova, Onur Özsoy, Tatiana Pashkova, Maria Pohle, Judith Purkarthofer, and 10 others. 2024. Rueg corpus.

Arianna Masciolini, Emilie Francis, and Maria Irena Szawerna. 2024. Synthetic-error augmented parsing of Swedish as a second language: Experiments with word order. In *Proceedings of the Joint Workshop on Multiword Expressions and Universal Dependencies (MWE-UD) @ LREC-COLING 2024*, pages 43–49, Torino, Italia. ELRA and ICCL.

Jürgen M. Meisel, Harald Clahsen, and Manfred Pienemann. 1981. On determining developmental stages in natural second language acquisition. *Studies in Second Language Acquisition*, 3(2):109–135.

Noah-Manuel Michael and Andrea Horbach. 2025. GermDetect: Verb placement error detection datasets for learners of Germanic languages. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 818–829, Vienna, Austria. Association for Computational Linguistics.

Stefan Müller. 2003. Mehrfache Vorfeldbesetzung [Multiple occupation of the pre-field]. *Deutsche Sprache*, 31(1):29–62. https://hpsg.hu-berlin.de/~stefan/Pub/mehr-vf-ds.html.

Stefan Müller. 2005. Zur Analyse der scheinbar mehrfachen Vorfeldbesetzung [An analysis of the apparent multiple occupation of the pre-field]. *Linguistische Berichte*, 203:297–330. https://hpsg.hu-berlin.de/~stefan/Pub/mehr-vf-lb.html.

Stefan Müller. 2013. Datensammlung zur scheinbar mehrfachen Vorfeldbesetzung [Dataset on the Apparent Multiple Occupation of the Pre-field]. Unpublished manuscript.

Mihai Nadăş, Laura Dioşan, and Andreea Tomescu. 2025. Synthetic data generation using large language models: Advances in text and code. *IEEE Access*, PP:134615–134633.

Inger Petersen. 2014. *Schreibfähigkeit und Mehrsprachigkeit [Writing Skills and Multilingualism]*. De Gruyter Mouton, Berlin, Boston.

Manfred Pienemann. 1989. Is Language Teachable? Psycholinguistic Experiments and Hypotheses. *Applied Linguistics*, 10(1):52–79.

Manfred Pienemann. 1998. *Language processing and second language development. Processability theory*. Benjamins,.

Manfred Pienemann. 2005. An introduction to Processability Theory. In Manfred Pienemann, editor, *Cross-Linguistic Aspects of Processability Theory*, pages 1–60. John Benjamins.

John Rickford. 2002. Implicational Scales. In Peter Trudgill James K. Chambers and Natalie Schilling-Estes, editors, *The Handbook of Language Variation and Change*, pages 142–167. Blackwell.

Josef Ruppenhofer, Annette Portmann, Matthias Schwendemann, Christine Renker, Katrin Wisniewski, and Torsten Zesch. 2025. Where it's at: Annotating verb placement types in learner language. In *Proceedings of the 19th Linguistic Annotation Workshop (LAW-XIX-2025)*, pages 187–200, Vienna, Austria. Association for Computational Linguistics.

Josef Ruppenhofer, Matthias Schwendemann, Annette Portmann, Katrin Wisniewski, and Torsten Zesch. 2024. Every verb in its right place? a roadmap for operationalizing developmental stages in the acquisition of L2 German. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6655–6670, Torino, Italia. ELRA and ICCL.

Gözde Gül Şahin and Mark Steedman. 2018. Data augmentation via dependency tree morphing for low-resource languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5004–5009, Brussels, Belgium. Association for Computational Linguistics.

Simon Sauer and Anke Lüdeling. 2016. Flexible multilayer spoken dialogue corpora. *International Journal of Corpus Linguistics*, 21:419–438.

Sören Schalowski. 2015. Wortstellungsvariation aus informationsstruktureller Perspektive [Word Order Variation from the Perspective of Information Structure]. Working paper 18, Sonderforschungsbereich 632 - Informationsstruktur.

Sören Schalowski. 2012. How German sentences begin. On an information-structural typology of multiple prefields in spoken German. Talk given at the Meertens Institute, Amsterdam.

Christin Schellhardt and Christoph Schroeder. 2015. Nominalphrasen in deutschen und türkischen texten mehrsprachiger schülerinnen [noun phrases in german and turkish texts written by multilingual students]. In Arne Ziegler and Klaus-Michael Köpcke, editors, *Deutsche Grammatik in Kontakt: Deutsch als Zweitsprache in Schule und Unterricht*, pages 241–262. De Gruyter, Berlin, München, Boston.

David Schneider and Kathleen F. McCoy. 1998. Recognizing syntactic errors in the writing of second language learners. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2*, pages 1198–1204, Montreal, Quebec, Canada. Association for Computational Linguistics.

Christoph Schroeder and Jana Gamper. 2016. Sprachliche Bildung für Neuzugewanderte. Ein Plädoyer für einen erwerbssequentiellen Ansatz [Language instruction for new immigrants: A plea for an approach

based on order of acquisition]. *Osnabrücker Beiträge zur Sprachtheorie*, 89:217–230.

Christoph Schroeder, Christin Schellhardt, Mehmet-Ali Akinci, Meral Dollnick, Ginesa Dux, Esin Işıl Gülbeyaz, Anne Jähnert, Ceren Koç-Gültürk, Patrick Kühmstedt, Florian Kuhn, Verena Mezger, Carol Pfaff, and Betül Sena Ürkmez. 2015. *MULTILIT : manual, criteria of transcription and analysis for German, Turkish and English*.

Petra Schulz and Angela Grimm. 2012. Spracherwerb [language acquisition]. In Heinz Drügh, Susanne Komfort-Hein, Andreas Kraß, Cécile Meier, Gabriele Rohowski, Robert Seidel, and Helmut Weiß, editors, *Germanistik. Sprachwissenschaft – Literaturwissenschaft – Schlüsselkompetenzen*, pages 155–172. Metzler, Stuttgart.

Petra Schulz and Rosemarie Tracy. 2011. *Linguistische Sprachstandserhebung - Deutsch als Zweitsprache (LiSe-DaZ) [Linguistic Assessment of Language Proficiency - German as a Second Language]*. Hogrefe, Göttingen.

Matthias Schwendemann, Katrin Wiseniewski, Torsten Zesch, Josef Ruppenhofer, Annette Portmann, Christine Renker, and Lisa Lenort. 2025a. Den Verbstellungserwerb in großen Lernerkorpora untersuchen [Investigating verb placement acquisition in larger learner corpora]. Tagung Große Lernerkorpa - Möglichkeiten und Grenzen, Leipzig.

Matthias Schwendemann, Katrin Wisniewski, Lisa Lenort, Annette Portmann, Christine Renker, Josef Ruppenhofer, and Torsten Zesch. 2025b. Zur Entstehung und Erschließung der bislang größten Lernerkorpus-Datenbank des Deutschen: Ein Bericht aus dem DAKODA-Projekt [On the creation and development of the largest learner corpus database of German to date: A report from the DAKODA project]. *Zeitschrift für Germanistische Linguistik*, 53:580–600.

Larry Selinker. 1972. Interlanguage. *International Review of Applied Linguistics in Language Teaching*, 10(1-4):209–232.

Heike Telljohann, Erhard W. Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2006. Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z).

Tobias Thelen. 2010. *Automatische Analyse orthographischer Leistungen von Schreibanfängern [Automatic Analysis of Spelling Performance in Beginner Writers]*. Ph.D. thesis, Universität Osnabrück.

Nina Vyatkina. 2016. The kansas developmental learner corpus (kandel). *International Journal of Learner Corpus Research*, 2(1):101–119.

Helena Wedig, Carola Strobl, Jim J. J. Ureel, and Tanja Mortelmans. 2025. *The Beldeko Corpus as a Resource to Investigate Cohesion in German Learner Language: A Preliminary Analysis of Corpus Homogeneity*, page 15–30. Springer Berlin Heidelberg, Berlin, Heidelberg.

Heike Wiese and Hans G. Müller. 2018. The hidden life of V3: An overlooked word order variant on verb-second. In Mailin Antomo and Sonja Müller, editors, *Non-canonical verb positioning in main clauses*, pages 202–223. Buske, Hamburg.

Steffi Winkler. 2014. Für eine psycholinguistisch orientierte Fremdsprachendidaktik. Anmerkungen auf Grundlage einer exemplarischen Studie zum Negationserwerb im Deutschen [In support of a pyscholinguistically oriented foreign language didactics. Remarks based on a sample study on the acqusistion of negation in German]. In U. Bredel, I. Ezhova-Heer, and S. Schlickau, editors, *Zur Sprache.kɔm*, page 51. Universitätsverlag.

Katrin Wisniewski, Elisabeth Muntschick, and Annette Portmann. 2022. Schreiben in der Studiersprache Deutsch. Das Lernerkorpus DISKO [Writing in German as the Language of Higher Education. The DISKO Learner Corpus]. In Katrin Wisniewski, Wolfgang Lenhard, Leonore Spiegel, and Jupp Möhring, editors, *Sprache und Studienerfolg bei Bildungsausländerinnen und Bildungsausländern [Language and Academic Success Among International Students]*, pages 283–304. Waxmann, Münster, Deutschland.

Katrin Wisniewski, Karin Schöne, Lionel Nicolas, Chiara Vettori, Adriane Boyd, Detmar Meurers, Andrea Abel, and Jirka Hana. 2013. MERLIN: An online trilingual learner corpus empirically grounding the European Reference Levels in authentic learner data. In *ICT for Language Learning Conference (ICT4LL 2013)*.

Zekun Wu and Yuan Li. 2022. Zur syntaktischen Komplexität des Schriftdeutschen chinesischer Deutschlerner/-innen – Eine korpusbasierte Profilanalyse [On the Syntactic Complexity of Written German Produced by Chinese Learners of German – A Corpus-Based Profile Analysis]. *Deutsch als Fremdsprache*, (4).

Angelika Wöllstein. 2010. *Topologisches Satzmodell [Topological Sentence Model]*. Kurze Einführungen in die germanistische Linguistik. Winter.

Amir Zeldes, Anke Lüdeling, and Hagen Hirschmann. 2008. What's Hard? Quantitative Evidence for Difficult Constructions in German Learner Data. In *Proceedings of QITL 3*, Helsinki.

A Datasets

Table 9 lists the main datasets and shows key information.

Composition of CEFR and Diversification samples with verb placement annotation

Table 10 shows the source corpora for the documents included in the two samples. Tables 11 and 12 show the distribution of verb placement types in the two samples. Note the rarity of ADV.

Name	L1/L2	Size	Source
TubaTest	L1	10478	Telljohann et al. (2006)
TopoBatt	L1/synth.	636	Ruppenhofer et al. (2024)
L2Trees	L2	637	this paper
CEFR	L2	962	Ruppenhofer et al. (2025)
Diversification	L2	1078	Schwendemann et al. (2025a) (p.c.)

Table 9: Overview of the different datasets used in our experiments.

Diversification sample		**CEFR sample**	
Corpus	Reference	Corpus	Reference
Bematac L2	Sauer and Lüdeling (2016)	Merlin	Boyd et al. (2014)
CDPS‡	Wu and Li (2022)	Disko‡	Wisniewski et al. (2022)
Kolipsi 1 (KLP1)‡	Glaznieks et al. (2023)		
Multilit spoken (MULT_S)‡	Schellhardt and Schroeder (2015)		
Multilit written (MULT_W)‡	Schellhardt and Schroeder (2015)		
SWIKO/WTLD‡	Karges et al. (2022)		
# sentences	1078	# sentences	962

Table 10: Corpora represented in the manually annotated verb placement data
(Corpora marked with ‡ are available only for academic research)

CEFR	# docs	ADV	INV	SVO	SEP	VEND	Total
A1	27	4	31	67	28	1	131
A2	17	1	36	62	20	7	126
B1	10	6	41	71	49	22	189
B2	11	18	124	143	110	150	545
C1	12	6	198	190	140	221	755
Total	77	35	430	533	347	401	1746

Table 11: Counts of verb placement types per level in CEFR sample

Corpus	# docs	ADV	INV	SVO	SEP	VEND	Total
bematac	1	2	71	65	32	30	200
CDPS	8	10	67	46	50	42	215
KLP1	12	20	67	113	84	66	350
WTLD	23	11	13	89	12	16	141
Mult spoken	11	3	56	90	95	43	287
Mult written	12	0	20	41	26	36	123
Total	67	46	294	444	299	233	1316

Table 12: Counts of verb placement types across corpora in the diversification sample

Composition of L2Trees

The source corpora for the sampled documents are listed in Table 13. They are accessible through the DAKODA repository.

Corpus	Source
Beldeko	(Wedig et al., 2025)
Comigs	(Köhn and Köhn, 2018)
Digs	(Diehl, 1998)
Dulko essays	(Hirschmann and Nolda, 2020)
Kandel	(Vyatkina, 2016)
Koko	(Abel et al., 2014, 2016)
Leonide	(Glaznieks et al., 2022)
Merlin	(Wisniewski et al., 2013)
Osna L2	(Tobias Thelen, 2010)
Pete L1	(Petersen, 2014)
Pete L2	(Petersen, 2014)
Rueg bilingual	(Lüdeling et al., 2024)
Rueg mono	(Lüdeling et al., 2024)
Whig (Falko-Whig)	(Zeldes et al., 2008)

Table 13: Corpora sampled in L2Trees

Table 14 shows the counts for the different verb placement types.

CEFR	Count
ADV	17
INV	205
SEP	242
SVO	461
VEND	214

Table 14: Verb placement types in L2Trees

B Prompt templates

Below we show the prompt templates used. The German version was actually used in the experiments; the English version is offered as a convenience for readers.

Prompt: `one-shot`

Rolle und Ziel:
Du bist ein Experte für Universal Dependency-Syntax und für das Topologische-Felder-Modell. Deine Aufgabe ist es, UD-Bäume zu modifizieren.

Kontext der Aufgabe:
Du erhältst zwei Beispielbäume mit Universal Dependencies. Der erste Baum hat eine kanonische Vorfeldbesetzung, bei zweite Baum zeigt eine modifizierte Wortfolge, bei der mehr als eine Phrase im Vorfeld steht, d.h. er hat eine mehrfache Vorfeldbesetzung.

Zusätzlich erhältst Du einen neuen UD-Baum. Modifiziere den Baum so, dass er ebenfalls mehr als eine Phrase im Vorfeld hat. Du kannst Tokens verschieben oder neue Tokens einfügen. Achte darauf, dass der modifizierte Baum die korrekten Köpfe und Dependenzen hat, d.h. wenn Du Tokens im Baum verschiebst, musst Du die entsprechenden Token-Ids und Kopf-Ids anpassen.

Anweisungen:

1. Analysiere zuerst die topologische Wortstellung im Beispielbaum **Beispiel**.
2. Analysiere dann die topologischen Wortstellung im modifizierten Beispielbaum **Beispiel (modifiziert)**.
3. Analysiere die topolotische Wortstellung im UD-Baum **Originalbaum**.
4. Modifiziere den Originalbaum so, dass er eine mehrfache Vorfeldbesetzung aufweist.
5. Prüfe, ob die Token-Ids und Kopf-Ids im modifizierten Baum nach der Modifikation korrekt sind und korrigiere sie, wenn nötig.

Wichtig: Der Output soll nur den modifizierten UD-Baum enthalten.

Beispiel [CONLLU TREE]

Beispiel (modifiziert) [CONLLU TREE]

Originalbaum [CONLLU TREE]

Figure 1: Prompt template for the ONE-SHOT setting ([CONLLU TREE] is replaced by the conllu trees for the example tree, the modified example and the input tree).

Prompt: `one-shot`

Role and Objective:
You are an expert in Universal Dependency syntax and the Topological Fields Model. Your task is to modify UD trees.

Task Context:
You are provided with two example trees containing Universal Dependencies. The first tree has a canonical VF structure, while the second tree shows a modified word order in which more than one phrase appears in the first field, i.e., it has a multiple VF structure.

In addition, you will be given a new UD tree. Modify the tree so that it also has more than one phrase in the VF. You can move tokens or insert new tokens. Make sure that the modified tree has the correct heads and dependencies; that is, if you move tokens in the tree, you must adjust the corresponding token IDs and head IDs.

Instructions:

1. First, analyze the topological word order in the example tree **Example**.
2. Then, analyze the topological word order in the modified example tree **Example (modified)**.
3. Analyze the topological word order in the UD tree **Original Tree**.
4. Modify the original tree so that it exhibits multiple head positions.
5. Check whether the token IDs and head IDs in the modified tree are correct after the modification and correct them if necessary.

Important: The output should contain only the modified UD tree.

Example [CONLLU TREE]

Example (modified) [CONLLU TREE]

Original tree [CONLLU TREE]

Figure 2: English translation for the ONE-SHOT prompt in Fig.1 ([CONLLU TREE] is replaced by the conllu trees for the example tree, the modified example and the input tree).

Prompt: `zero-shot`

Rolle und Ziel:
Du bist ein Experte für Universal Dependency-Syntax und für das Topologische-Felder-Modell. Deine Aufgabe ist es, Syntaxbäume zu modifizieren.

Aufgabe:
Du erhältst einen Beispielbaum mit Universal Dependencies. Der Baum hat eine nicht-kanonische Wortfolge. Kopiere den Baum und behalte die syntaktischen Dependenzen und POS-Tags, verändere aber die Semantik, indem Du die Wortformen und Lemmas durch neue Wortformen und Lemmas ersetzt, so dass der Satz eine neue Bedeutung bekommt.

Anweisungen:

1. Lies zuerst den Text des Originalbaums und analysiere seine Struktur.
2. Generiere dann einen neuen Text der gleichen Länge und neuer Bedeutung. Der neue Satz soll exakt die gleiche POS-Sequenz und syntaktische Struktur haben wie der Originalsatz.
3. Ersetze die Wortformen und Lemmas im Original-Syntaxbaum mit den Wortformen und Lemmas des neuen Satzes.
4. Prüfe, ob die Syntax des neuen Satzes korrekt ist und der Baum keine strukturellen Fehler enthält.

Wichtig: Der Output soll nur den neuen, modifizierten Syntax-Baum enthalten.

Originalbaum [CONLLU TREE]

Figure 3: Prompt template for the ZERO-SHOT setting ([CONLLU TREE] is replaced by the input tree).

Prompt: `zero-shot`

Role and Objective:
You are an expert in Universal Dependency syntax and the Topological Fields model. Your task is to modify syntax trees.

Task:
You are given a sample tree with Universal Dependencies. The tree has a non-canonical word order. Copy the tree and retain the syntactic dependencies and POS tags, but change the semantics by replacing the word forms and lemmas with new ones so that the sentence takes on a new meaning.

Instructions:

1. First, read the text of the original tree and analyze its structure.
2. Then generate a new text of the same length and with a new meaning. The new sentence should have exactly the same POS sequence and syntactic structure as the original sentence.
3. Replace the word forms and lemmas in the original syntax tree with the word forms and lemmas of the new sentence.
4. Check whether the syntax of the new sentence is correct and the tree contains no structural errors.

Important: The output should contain only the new, modified syntax tree.

Original tree [CONLLU TREE]

Figure 4: English translation for the ZERO-SHOT prompt in Fig. 3 ([CONLLU TREE] is replaced by the input tree).

C BERT verb placement classifier

We train a BERT sequence tagger in a multi-label setup on the CEFR data. To benefit from the additional, synthetic data, we add different-sized samples randomly extracted from the data created by the rule-based approach (RULEB) to our training data.

Model and parameters We initialise our model with the `deepset/gbert-large` model[18] (Chan et al., 2020) and train it for 10 epochs (with early stopping), using a batch size of 8 and a learning rate of 3.843349850997621e-05. The parameters have been determined through hyperparameter tuning, based on Optuna.[19]

Self-training setup for the BERT tagger We augment our training data with N instances sampled from our synthetic training data (RULEB). To improve results for the multiply occupied prefields, we select trees generated with rules that target the ADV class and add labels for the other classes in a self-training setup where we use the BERT classifier trained solely on `CEFR` to predict the labels for INV, SEP, SVO, VEND. We then add the tagged data to the training set and retrain the BERT model.

Table 15 shows results for the different models trained on varying subsets from our synthetic data. We see substantial variance for the different models, showing best results for the CEFR+1000 setting. However, it is unclear whether the results will generalise to new data.

D Additional results for verb placement

To check if the rule-based approach on L2Trees could be outperformed by a classical ML approach, we used `autogluon`[20] to train a classifier ensemble using linguistic tree-based features extracted from the L1 data in TopoBatt. The performance of this classifier on an in-domain test set is shown in Table 16: it seems to be as good as the manual rules, though performance on ADV may be slightly lower.

Training and testing on L2Trees produces a bit lower but still good results for ADV at 0.667 (cf. Table 17. It's not clear if the lower performance relative to TopoBatt is owed to ADV being more difficult to recognize on L2 data or to the fact that we have only half as many instances as in TopoBatt.

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

[19] `https://github.com/optuna/optuna`.

[20] `https://auto.gluon.ai/stable/index.html`

Model	ADV	INV	SEP	SVO	VEND
CEFR	29.1	95.0	95.0	95.4	98.7
CEFR+500	48.4	96.0	94.1	96.7	98.4
CEFR+1000	50.7	94.7	94.2	96.2	98.2
CEFR+1500	36.4	95.7	92.2	94.8	98.0
CEFR+ALL	28.9	95.6	92.5	95.2	98.2

Table 15: BERT classifier results (F1) for different models. CEFR: trained on CEFR only; CEFR+N: trained on CEFR + different portions of synthetic rule-based data with automatic predictions for the INV/SEP/SVO/VEND classes (N=500, N=1000, N=1500, All: N=3022).

Category	Precision	Recall	F1-Score
ADV	0.857	0.667	0.750
INV	1.000	1.000	1.000
SEP	1.000	0.900	0.947
SVO	0.963	1.000	0.981
VEND	1.000	1.000	1.000

Table 16: Ensemble performance: train and test on TopoBatt

(Note that we cannot compare this ADV score directly to the one for the manual rules reported in §4 because the latter results use all of L2Trees as a test set, while here we only use a part of it for testing.)

What is more important to us is that, as shown by the results in Table 18, when the TopoBatt ensemble model is applied to L2Trees, the non-ADV classes seem to generalize but ADV performance drops to a level lower than seen with the manual rules due to a large drop in recall. This again suggests that ADV can look quite different depending on the dataset.

E Results for rule-based verb placement prediction on L2trees

To make use of all of our available data, we also applied the rule-based verb placement classifier to the different versions of L2Trees (cf. Table 19). Results for ADV here are inbetween those for the Diversification and CEFR datasets. While no individual model trained on synthetic data can beat the model trained on the BASE parser, the model trained on the combination of all synthetic data can. The gain again is due to increased recall, which looks particularly drastic in percentage terms but is

Category	Precision	Recall	F1-Score
ADV	0.600	0.750	0.667
INV	0.919	1.000	0.958
SEP	0.925	1.000	0.961
SVO	0.960	1.000	0.980
VEND	1.000	1.000	1.000

Table 17: Ensemble performance: train and test on L2Trees

Category	Precision	Recall	F1-Score	Support
ADV	0.273	0.176	0.214	17.0
INV	0.867	0.961	0.912	204.0
SEP	0.942	0.942	0.942	241.0
SVO	0.894	0.965	0.928	461.0
VEND	0.942	0.991	0.966	214.0

Table 18: TopoBatt-trained ensemble model on L2Trees

probably unrealistically high because of the small number of ADV instances in this dataset.

Model	Cat	Prec	Rec	F1	Support
base	ADV	34.5	58.8	43.5	17
ruleB	ADV	21.6	64.7	32.4	17
ruleE	ADV	32.4	64.7	43.1	17
qwen	ADV	26.3	58.8	36.4	17
oss	ADV	22.0	64.7	32.8	17
synthall	ADV	32.6	88.2	47.6	17
base	INV	88.8	77.6	82.8	205
ruleB	INV	89.5	75.1	81.7	205
ruleE	INV	90.6	75.6	82.5	205
qwen	INV	90.5	74.6	81.8	205
oss	INV	89.9	78.0	83.5	205
synthall	INV	91.3	76.6	83.3	205
base	SEP	89.6	96.3	92.8	242
ruleB	SEP	88.3	96.3	92.1	242
ruleE	SEP	90.7	96.7	93.6	242
qwen	SEP	89.5	94.6	92.0	242
oss	SEP	91.2	94.6	92.9	242
synthall	SEP	88.4	94.2	91.2	242
base	SVO	95.9	71.6	82.0	461
ruleB	SVO	95.1	72.0	82.0	461
ruleE	SVO	96.3	73.5	83.4	461
qwen	SVO	95.4	72.0	82.1	461
oss	SVO	95.4	72.2	82.2	461
synthall	SVO	96.0	73.1	83.0	461
base	VEND	95.7	83.6	89.3	214
ruleB	VEND	96.8	84.1	90.0	214
ruleE	VEND	96.8	84.1	90.0	214
qwen	VEND	96.8	83.6	89.7	214
oss	VEND	96.2	82.7	88.9	214
synthall	VEND	96.3	84.1	89.8	214

Table 19: Comparison of four different models trained on various datasets and applied to the L2Trees sample.

F Multiply filled pre-fields in TüBa-D/Z

To have a better sense of what the parser training data look like, we inspect the parses in the L1 TüBa-D/Z tree bank for cases of ADV.

We proceed as follows. To identify prefields, we inspect each sentence for the maximal contiguous token sequences where each component token has the TopoField value VF (= Vorfeld 'pre-field'). We then analyze the dependency relations of the tokens in each sequence of prefield tokens. If exactly one token has a head outside the prefield, then the prefield holds only one sub-tree ($\sim$ constituent). If there is more than one token whose head is not itself part of the prefield, then the prefield holds multiple subtrees and represents a case of ADV (on the assumption that all parses are indeed correct).

Of the 104,787 sentences in the tree bank, 83,028 have at least one prefield. In these sentences, a total of 100,615 VF sequences were found. Within these, we counted 341 VF sequences with >1 sub-tree, i.e. candidates for ADV. We inspected all of these cases manually to verify if they were real instances. We found that all/most of these matches were false positives. In quite a few cases, the treebank annotation seems wrong. For instance, coordinations like *und* 'and', *aber* 'but', *denn* 'for' are erroneously placed in VF (rather than the PARORD field), as *und* is in (8).

(8) "Unsere Strategie ist ohne Alternative, und sie beginnt zu greifen", resümiert der Kanzler.
"Our strategy is without alternative, and it starts to take-effect", concludes the chancellor.
'There is no alternative to our strategy, and it is beginning to take effect," the chancellor concludes.'

In the other large subset of cases, our algorithm proves too simplistic because we use flat topological annotations. For instance, if we have a noun phrase in the prefield which includes a finite clause as an apposition to the head noun, then the VF of the finite clause is adjacent to the VF tags on the article and the head noun, even though they involve VFs at hierarchically different levels. In (9), *wir* is in the VF of the appositive clause but the whole NP headed by *Slogans* is in the prefield of the clause headed by *versucht* at a higher level.

(9) Mit Slogans "Wir sind das Volk , das die Ernte einbringen will" , versucht die Bewegung Anhänger zu mobilisieren
With slogans "we are the people, which the harvest bring-in wants" , seeks the government supporters to mobilise
'With slogans like "We are the people who want to bring in the harvest," the movement is trying to mobilize supporters.'

The results of our query suggest that we have no real ADV instances that are identifiable by our approach. If there were any correctly labeled clear-cut cases, we should have found them.

Of course, it is possible that there are such cases in the treebank but that they were mislabeled. In terms of training a parser to recognize ADV, this would be worse because it would lead to inconsistency. To investigate the possibility that some ADV

instances might be mislabeled, we search the TüBa-D/Z treebank for prefields containing an adverb following by a determiner (defined as the UPOS sequence ADV+DET). This is a clear, central pattern of ADV cases that Müller (2013) discusses.

However, among the 1170 matches we obtain the vast majority of cases involve adverbs that are correctly attached because they are focus markers (e.g. *allein/einzig* 'only', *gerade* 'even') or quantifiers (e.g. *fast* 'almost', *etwa* 'about, circa'). In other cases, the attachment to the NP seems unusual but is plausible against the wider discourse. For instance, in (10), *[e]rneut* is highly unusual as a modifier of the subject.

(10) Erneut der SD-Bericht zitiert einen empörten Arbeiter
Again the SD-report cites an upset worker
'Once again, the SD report quotes an outraged worker / It is once again the SD-report which cites an outraged worker. '

However, reading the whole article the sentence is taken from, it is clear that semantically the repetition does not pertain to 'cite an outraged worker'. Instead, what is conveyed is that once more it is the SD report that is the source of some juicy information, given its preceding mention in (11).

(11) In Sorge wegen gewisser "Auflockerungen in der Bevölkerung" verbreitet sein Bericht vom 8. Juli 1943 sogar lautstarke Flüsterwitze.
'Concerned about a certain "relaxation of discipline among the population," his report of July 8, 1943, even includes some risqué jokes'.

In some few other cases, it seems to us that discourse markers were erroneously treated as adverbs inside an NP in VF. In (12), the initial token *[a]lso* seems to be used less in the sense 'therefore' and more as a discourse marker 'well then/ok'. As a discourse marker, the token should occupy a separate field preceding VF, that is, correct annotation would not result in ADV either.

(12) Also den Tunnel möchte ich gegenwärtig in dieser Breite nur deshalb haben, damit die Bundesbahn, wenn sie baut, nicht auch noch den dort fließenden Verkehr einengt.
So the tunnel would I currently in this breadth only therefore have, so that the federal-railroad, when they build, not also still the there flowing traffic restricts.
'So the only reason I want the tunnel to be this wide right now is so that when the Federal Railroad builds it, they don't also restrict the traffic flowing through there.'

Overall then, it seems that there are very few, if any, clear cases of ADV that were mislabeled.