Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 4833–4841
Marseille, 20-25 June 2022

RRGparbank: A Parallel Role and Reference Grammar Treebank

Tatiana Bladier, Kilian Evang, Valeria Generalova, Zahra Ghane, Laura Kallmeyer, Robin Möllemann, Natalia Moors, Rainer Osswald, Simon Petitjean
Heinrich Heine University Düsseldorf
Universitätsstr. 1, 40225 Düsseldorf, Germany
first.last@hhu.de

Abstract

This paper describes the first release of RRGparbank, a multilingual parallel treebank for Role and Reference Grammar (RRG) that contains annotations of George Orwell's novel *1984* and its translations. The release comprises the entire novel for English and a constructionally diverse, parallel "seed" sample for German, French, Russian, and Farsi. The paper gives an overview of the annotation decisions taken and describes the adopted treebanking methodology. As a possible application, a multilingual parser is trained on the treebank data. RRGparbank is one of the first resources for which RRG has been applied to large amounts of real-world data. It enables comparative and typological corpus studies in RRG and creates new possibilities of data-driven NLP applications based on RRG.

Keywords: Syntax, Treebank, Parallel Corpus, Role and Reference Grammar, English, German, French, Russian, Farsi

1. Introduction

Role and Reference Grammar (RRG) (Van Valin and LaPolla, 1997; Van Valin, 2005; Van Valin, 2010) has been proposed as a theory of grammar with an emphasis on typological adequacy. More recently, RRG has also been studied from the perspective of formal and computational linguistics: A formalization of RRG has been proposed in (Kallmeyer et al., 2013; Osswald and Kallmeyer, 2018), based on which a symbolic parser for precision grammars has been developed (Arps et al., 2019). Moreover, there have been recent initiatives for creating treebanks for RRG (Bladier et al., 2018; Chiarcos and Fäth, 2019).

In this paper, we present the first release of RRGparbank, a multilingual parallel treebank for RRG, based on George Orwell's novel *1984* and translations thereof.[1] RRGparbank is the first effort to apply RRG to a parallel large-scale corpus, making RRG usable as a framework for data-driven NLP and corpus-linguistic research. We expect the parallel nature of the treebank to make it especially useful for comparative and typological studies, for which RRG has been designed. Applying RRG to large amounts of real data has raised (and already helped answer) a number of questions about the details of RRG analyses which were previously undefined. We have used several innovative techniques to make treebanking efficient, including rule-based conversion from Universal Dependencies to RRG (Evang et al., 2021) and statistical parsing as starting points for annotation, as well as incremental improvements of these starting points with human annotators in the loop. We make the resulting treebank available with various download and search options.

In this paper, we give a brief introduction to RRG (Section 2), describe our treebanking methodology and highlight important aspects of the annotation guidelines we developed (Section 3), describe the released resource and tools (Section 4), and demonstrate statistical parsing as one possible use of an RRG treebank (Section 5).

2. Role and Reference Grammar

2.1. Background

Role and Reference Grammar (RRG) is a functional theory of grammar whose development has been strongly driven by the investigation of typologically varied languages. RRG aims at integrating syntactic, semantic and pragmatic levels of description which are related to each other by the "linking system", an elaborate system of linguistic rules and constraints (Van Valin and LaPolla, 1997; Van Valin, 2005; Van Valin, 2010). Since the focus of RRGparbank is primarily on syntactic annotation, RRG's syntax-semantics-pragmatics interface will not be discussed here in more detail.

A key syntactic concept of RRG is the "layered structure of the clause" comprising the layers *nucleus*, *core* and *clause*. The nucleus encodes the main predicate, the core consists of the nucleus and the syntactic realizations of the predicate's arguments, and the clause includes the core plus extracted arguments. Each layer can be accompanied by *peripheral* structures for attaching adjuncts. For instance, in a verbal constituent, aspectual modifiers attach to the nucleus, locative and temporal modifiers attach to the core, while modal adverbials attach to the clause. The layered structure is not restricted to verbal phrases but applies also to constituents headed by other elements such as nouns, prepositions, etc.

Closed-class morphosyntactic elements for encoding tense, modality, aspect, or definiteness, among others, are referred to as *operators* in RRG. They attach to the

[1] RRGparbank is available at: https://rrgparbank.phil.hhu.de

layers over which they take scope, and it is a crucial assumption of RRG that the surface ordering of the operators is aligned with the height of their attachment site. Since the surface order of the operators relative to arguments and adjuncts would often require crossing branches in the syntactic representations, RRG considers the constituent structure and the operator structure as different syntactic *projections* of the clause.

Concerning the structure of complex sentences, RRG draws not only a distinction between embedded, dependent structures (*subordinations*) and non-embedded, independent ones (*coordinations*), but assumes in addition non-embedded dependent structures, so-called *cosubordinations*. Cosubordinate structures have the general form $[[\ ]_X\ [\ ]_X]_X$. In such constructions, operators that apply to category X are usually realized only once but have scope over both constituents. An example for a CORE cosubordination is en-2304[2], *Presumably [[she could be trusted]$_{CORE}$ [to find a safe place]$_{CORE}$]$_{CORE}$*, where we have a modal operator (*could*) that is part of the first CORE but scopes also over the second. The three different nexus types can occur at all layers. For instance, English resultative constructions such as *tore open* in *He tore open a corner of the packet* (en-2889) are analyzed as nuclear cosubordinations since they function as complex predicates. By comparison, raising constructions like *He seemed to know the place* (en-1788) are generally analyzed as core coordinations.

2.2. Formalization

The syntactic annotations in RRGparbank build on the formalized version of RRG proposed by Kallmeyer and Osswald (2017) and Osswald and Kallmeyer (2018) (see also Kallmeyer et al. (2013)). An important difference between the structures used in this formalization and the syntactic representations found in RRG textbooks is that operators are integrated into the constituent projection. They are attached where they take scope, e.g., tense attaches at the CLAUSE level and negation attaches for instance at the CORE level, see Figure 1. The attachment of operators and also of modifiers (periphery structures in RRG) can lead to crossing branches.

The proposed formalization treats RRG as a *Tree Wrapping Grammar (TWG)*, which is based on a tree-rewriting formalism in the spirit of Tree Adjoining Grammar (Joshi and Schabes, 1997). A TWG consists of a finite set of elementary trees that can be combined by the following three basic operations: *(simple) substitution* (replacing a leaf by a new tree); *sister adjunction* (adding a new tree as a subtree to an internal node); and *wrapping substitution* (splitting the new tree at a dominance-edge, filling a substitution node with the lower part and adding the upper part to the root of the target tree). For more details on this formalization, see Kallmeyer et al. (2013; Osswald and Kallmeyer (2018). While not directly relevant to the annotation task *per se*, viewing RRG as a TWG allows us to extract grammars from the annotated corpora, which in turn can be employed for parsing purposes (see Section 5), and which was also used for selecting the seed data (see Section 4.5).

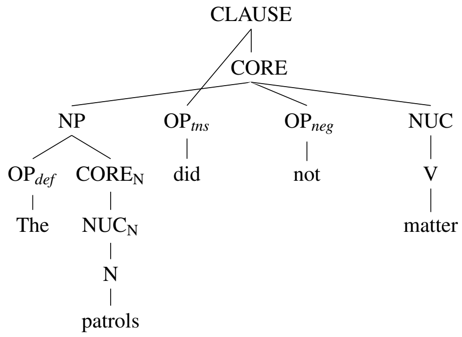


Figure 1: Examples of definiteness, tense and negation operators (en-28)

3. Annotation

3.1. Annotation pipeline

We provide annotators with initial, automatically created trees for all sentences, which they then correct using a web-based annotation interface (see Figure. 2).[3] For creating the initial trees, we first parsed the sentences with an off-the-shelf Universal Dependencies parser and converted them to RRG using a rule-based algorithm (Evang et al., 2021). Later, as annotators produced enough corrected annotations to train a statistical parser (see Section 5), we started to use the results of this parser instead for selected languages because it provided more accurate syntactic structures [4].

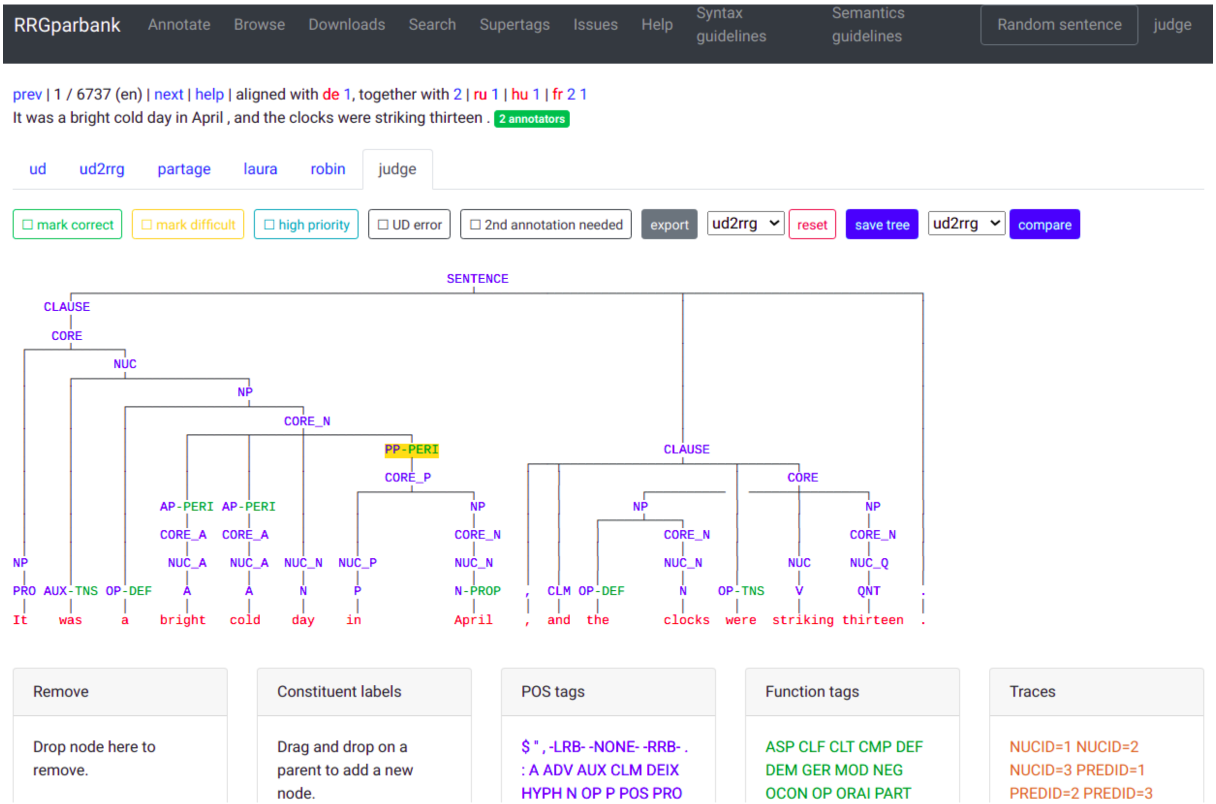
In total, 11 annotators were involved in creating the data in the release described in this paper. They were presented with sentences pre-annotated using the automatically generated trees, corrected them using the drag-and-drop web interface, and finally marked their version as correct. Annotators do not see each other's annotations. A tree marked correct by at least one annotator has *silver* status (the release includes the latest silver tree in such cases).

Some of the annotators (who are RRG experts) are also allowed to sign in using a special *judge* account, where they can see all annotations, and a diff view highlighting the parts of trees where annotations differ (i.e.,

[2] Throughout the paper, *L-n* is used as id for sentence number *n* in language *L* in RRGparbank.

[3] The first prototype of the interface was implemented by Andreas van Cranenburgh using components of his disco-dop framework (van Cranenburgh et al., 2016).

[4] Evang et al. (2021) showed that a statistical parser starts to outperform the rule-based conversion algorithm from UD dependencies to RRG structures at about 2000 training sentences for English. Although a further fine-tuning of the rule-based approach is possible, it would not be practical due to a large number of additional required rules.



prev | 1 / 6737 (en) | next | help | aligned with de 1, together with 2 | ru 1 | hu 1 | fr 2 1

It was a bright cold day in April , and the clocks were striking thirteen . 2 annotators

Drop node here to remove.

Drag and drop on a parent to add a new node.

$ " , -LRB- -NONE- -RRB- . : A ADV AUX CLM DEIX HYPH N OP P POS PRO

ASP CLF CLT CMP DEF DEM GER MOD NEG OCON OP ORAI PART

NUCID=1 NUCID=2 NUCID=3 PREDID=1 PREDID=2 PREDID=3

Figure 2: The drag-and-drop annotation interface of RRGparbank (view as judge)

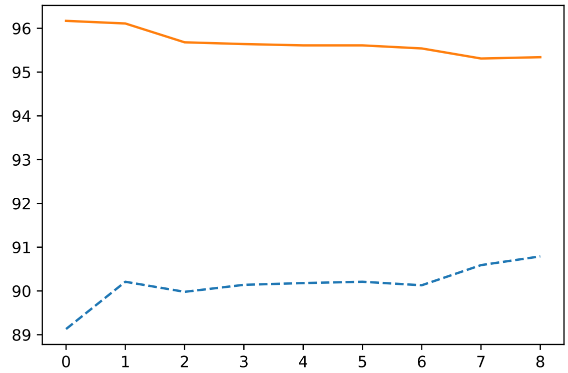


Figure 3: Cumulative quarterly inter-annotator f-score from January 2020 to March 2022, overall (solid curve) and for sentences with disagreements (dashed curve).

inter-annotator disagreements). The judge then has to decide which annotation decisions are the correct ones and create a final authoritative tree (based on the latest silver tree) using the normal tree editing operations. A tree marked correct by the judge has *gold status*. When it is not clear how to resolve a disagreement, the sentence is discussed between annotators at regular adjudication meetings before being marked as gold.

In the beginning, each sentence was annotated by at least two annotators before being judged. As annotators gained more experience and the guidelines were extended to cover more cases explicitly, we gradually moved to a more speedy annotation workflow where the expert annotators were allowed to use the judge account to directly mark a single annotation (that is not their own) as gold in easier cases, i.e., where they feel the existing annotation is clearly correct. They could also correct small, trivial annotation mistakes (for instance, deleting a second NP node below a first one with just a unary branch between the two). However, if, beyond that, they disagreed with something in the annotation, they were again instructed to create an alternative annotation using their regular annotator account, and leave the judging to another annotator. This workflow speeded up the annotation process considerably without sacrificing too many checks and balances compared to the complete workflow with at least two annotators and one judge. Between 30% (for Russian) and 60% (for English) of all released gold sentences have at least two annotations (see Section 4 for details). For these sentence pairs, we have an overall inter-annotator agreement of 91.04% measured as EVALB f-score (Collins, 1997). In Figure 3, we show cumulative agreement over time, binned by quarter. The solid curve represents overall agreement, counting sentences where the second annotator accepted the first annotation as agreeing perfectly. The dashed curve considers only sentences where a second annotation was provided. Overall agreement starts at 96.2% and goes down to 95.3% over time, as the "easy cases" tended to be annotated early. For sentences with two annotations, agreement starts at 89.1% and goes up to almost 91% as annotation guidelines got more fleshed out and annotators gained experience. In order to find out whether the possibility for the second annotator to start from the first annotation unduly biases them, we also compared agreement in the month before and after this possibility was introduced, finding no dramatic difference (87.64% to 89.88%).

3.2. Selected phenomena

RRGparbank, along with RRGbank (Bladier et al., 2018), a previous RRG treebank of text from the Penn Treebank, also annotated by our group, is the first endeavour to annotate large amounts of corpus data with RRG structures. The only other electronic syntactic RRG resource is that of Chiarcos and Fäth (2019)[5], a corpus consisting of 351 examples from the textbook of Van Valin and LaPolla (1997). In contrast to them, we were faced with a variety of constructions that RRG had not considered so far, which means that, besides annotating, we also had to take numerous decisions concerning syntactic analyses in RRG.
For a detailed description of the annotation decisions, see the guidelines available on the treebank website. In the following, we discuss a few interesting questions that came up during the annotation process.

Copula constructions. Most copula constructions feature a verb (usually *'to be'*) annotated as AUX (auxiliary). It is placed under NUC and is thus one of the predicating parts. It can also bear some operator features, e.g., tense, aspect, or modality. The other part of the predicate (mostly AP, PP, NP, or participle) is also dependent on the NUC. There is no auxiliary in the present tense in Russian, so the only predicating part and the only descendant of the NUC is the non-verbal constituent.

Discontinuous structures. Discontinuities (i.e., crossing branches) can arise in the treebank trees due to elements belonging to a higher layer but being positioned between elements belonging to a lower layer. These can be not only operators or periphery elements (as mentioned above) but also arguments. In these cases as well, the annotation contains crossing branches. Examples are discontinuous NUC constituents as in Figure 4 and discontinuous CORE constituents as in 1 below (the relevant part of the tree is given in Figure 5). We found this type of discontinuous CORE mainly in German.

(1) *Merkwürdigerweise schien ihn das Schlagen*
Curiously seems him$_{acc}$ the chiming
der vollen Stunde mit neuem Mut
of.the full hour$_{nom}$ with new courage
erfüllt zu haben .
filled to have .

'Curiously, the chiming of the full hour seems to have filled him with new courage.'

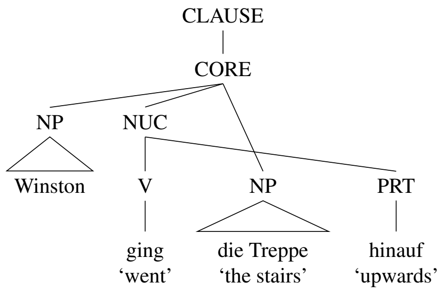


Figure 4: Discontinuous NUC for German particle verb (de-5)

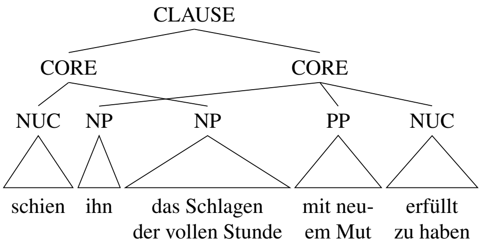


Figure 5: Discontinuous COREs in German (de-481)

Non-local dependencies. Two types of non-local dependencies are annotated in RRGparbank: one is long-distance dependencies arising from a fronted *wh*-phrase, or relative pronoun (in the pre-core slot (PrCS) in RRG) that does not belong to the CORE it precedes but to another CORE. In these cases, the feature NUCID identifies the predicate on which the PrCS depends. The PrCS, in turn, is provided with a PREDID feature pointing at the predicate. The coindexing of these two features expresses the predicate-argument relation in a long-distance dependency construction, see Figure 6.

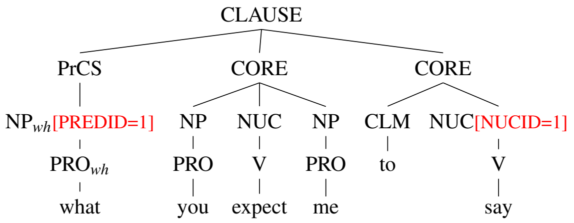


Figure 6: Subordinate interrogative clause with long-distance dependency (en-1761)

The second type of non-local dependency covered by the annotations are extraposed relative clauses (attached as a periphery element at the higher clause) that are linked to their antecedent NPs via coindexation in a feature REF. An example is given in Figure 7. Such constructions are particularly frequent in the German RRGparbank data (due to German's free word order); 4.8% of the German treebank sentences contain an extraposed relative clause.

[5] https://github.com/acoli-repo/RRG

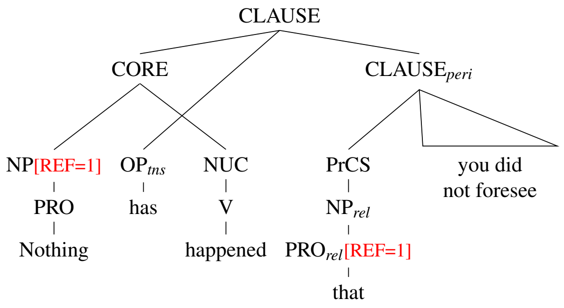


Figure 7: Extraposed relative clause (en-5922)

Multi-word expressions (MWEs). Fixed MWEs (e.g., *Big Brother*, *of course*) that cannot be modified and are syntactically inflexible are annotated in a flat way, i.e., with all POS tags below the same NUC (resp. NUC_X) node. This includes also inherently reflexive verbs (for instance German *sich erinnern* 'remember', or French *se trouver* 'be located', *se souvenir* 'remember'), where the reflexive pronoun and the verb are daughters of NUC, as well as fixed V N combinations such as English *give way*, *get hold*, and French *avoir lieu* 'take place'.

In contrast to this, light verb constructions (LVC), which are more flexible and productive, are annotated like full verbs, i.e., the non-verbal part (usually an NP or a PP) is placed under CORE, and the light verb is a V under NUC. An example is French *donner un bain* ('give a bath', fr-6400) and its English translation in en-5957.

Negation and modality. Expressions of negation are usually analyzed as operators (indicated by OP$_{neg}$ or OP-NEG). They can attach to any layer (CLAUSE, CORE or NUC) depending on their scope. Syntactic tests (for instance, addition of peripheral elements) show that English and German negation scopes over the CORE, and over the NUC in Russian. This difference is reflected in the annotation: negation elements are attached to NUC structures in Russian and to COREs in English and German (cf. en-106, de-105 vs. ru-104).

In French, the negation usually consists of two parts, i.e., we have negative concord. The particles *ne* and *pas* are annotated as operators exclusively as their unique function is to introduce the negation. In contrast, negative adverbs (like *jamais* 'never') and pronouns (like *rien* 'nothing') are heads of their respective phrases. In this case, the functional tag NEG is attached to the respective category labels.

The same applies to annotating modality: there are modality operators (e. g., the Russian particle *by* used for building the irrealis mood) as well as words that receive their own part-of-speech together with the functional MOD tag. For instance, Russian modal predicative adverbs, see ru-452 in 2, are annotated as ADV-MOD and can take their own dependencies.

(2)
Proshloe	*umer-lo*	*budushhee*
past	die-3SG.PST	future

nel'zja	*voobrazi-t'*	.
impossible.ADV.MOD	imagine-INF	.

'The past was dead, the future was unimaginable.'
(lit.: 'impossible to imagine')

Reported speech. The literary text contains many cases of direct and indirect reported speech. Direct speech includes a clause with a verb of saying and a quoted block with the contents of the utterance, e.g., *"And now let's see which of us can touch our toes!" she said enthusiastically* (en-611). In these cases, the quoted text is annotated as a separate SENTENCE subordinate to the main SENTENCE, while the reporting part appears under the usual spine, see Fig.8. Note, however, that not all cases of direct speech come with quotes; in French, we frequently have cases without quotes, for instance *Ils sont si bruyants! dit-elle.* (""They are so noisy!", she says.', fr-434).

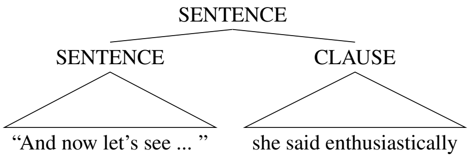


Figure 8: Direct speech (en-611)

Indirect reported speech often contains complementizers, anaphoric pronouns and relative tense marking. In these cases, the contents of the speech is treated as an argument of the saying predicate and appears as a subordinate CLAUSE, see Fig. 9 illustrating en-593: *The Party said that Oceania had never been in alliance with Eurasia.*

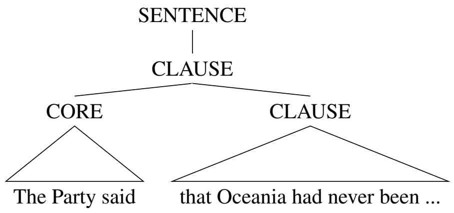


Figure 9: Indirect speech (en-593)

4. Resource

4.1. Source texts and sentence alignments

The annotated texts in RRGparbank are taken from George Orwell's novel *1984*. The English and Russian tokenized texts and sentence alignments are taken from the MULTEXT-East dataset (Erjavec, 2017). The corresponding French and German data was built manually using the published translations Orwell (1972) and Orwell (2003), respectively.

	EN	EN-SEED	DE-SEED	FR-SEED	RU-SEED	FA-SEED
Number of sentences	6 737	1 450	1 454	1 555	1 416	1 476
Number of tokens	122 843	23 750	23 444	24 670	17 697	22 456
Average sentence length	18.2	16.4	16.1	15.9	12.5	15.2
Not yet annotated	0	0	0	0	0	1 010
Silver	348	0	889	1 309	1 019	589
Gold with 1 annotation	2 691	575	286	112	183	0
Gold with $\geq$ 2 annotations	3 698	875	279	134	214	0

Table 1: Statistics beginning of May 2022 (preliminary—further annotations will be added for first release in June 2022). The release includes the entire novel for English and the seed sentences for German, French, Russian, and Farsi. Sentences that are not annotated yet (this concerns the Farsi seed data) will be released with so-called bronze trees, which means with automatically obtained parse trees.

4.2. Coverage

We make all sentences from the English text available. Currently, they are all at least silver, by the time of the conference they will be gold. This means that gold RRG trees for the entire English 1984 corpus will be provided in the planned release. Furthermore, we make a part of the German, French, Russian, and Farsi data publicly available as a "seed corpus".[6] We aim at representing a broad variety of linguistic phenomena across the languages in the seed data. We also aim at a high degree of parallelism in the seed, making cross-linguistic comparisons possible. We describe the selection of seed sentences in Section 4.5. Table 1 gives some statistics of the first release.

4.3. Download and search options

All released data can be downloaded in NEGra treebank export format (Brants, 1997), which is suitable to represent trees with crossing branches. We provide a suggested split into training, development, and test data for experiments: all sentences whose numbers end with [1-8] are used for training, sentences with numbers ending in 9 go into development and in 0 into the test set. The sentence alignments between the English text and each translation are availale for download as text files in a simple column-based format.

We make it easy for linguists to find certain constructions of interest in RRGparbank by providing the possibility to search the trees via RRGparbank's Web interface using the TGrep2 tree search tool (Rohde, 2005). Users can query the trees whose structure matches a specified pattern. For example, the search query 'NUC < (V $.. PRT)' returns all trees in which the node NUC directly dominates a verb V with a separate particle PRT, such that V precedes PRT, for example *stood up* in English or *fuhr fort* ('continued') in German. The query '/=SAID$/ . /=WINSTON$/' returns all trees in which the word WINSTON comes directly after SAID.

4.4. Annotation guidelines

We document our annotation decisions in the form of an annotation manual, available on the RRGparbank website. These guidelines are work in progress since the annotation process still leads to discussions of previously unseen phenomena or, sometimes, to revisions of earlier decisions.

4.5. Selection of seed data

To select seed corpora with a high degree of parallelism and a broad coverage of constructions, we extracted a Tree Wrapping Grammar for each language (see Section 5 for more details), which included assigning syntactic supertags (unanchored elementary trees) to tokens. We then selected a set of sentences together with their translations in all four languages in a way that maximizes the number of distinct supertags per language. Doing this optimally is an NP-complete problem, so we opted for a greedy approximation. We selected seed training sentences for the language with the highest annotation coverage first (German) and then proceeded to add sentences for English, Russian, and French. For each language, we iterated until all supertags that occur in the silver and gold training split at least twice were included. At each iteration, we added the sentence that maximizes the ratio u/l, where u is the number of unseen supertags in the sentence (i.e., supertags that are not yet in the seed) and l is the length of the sentence. Before moving on to the next language, we added all sentences that occur in the same sentence-level translation unit according to our sentence alignments (regardless of whether they are training, development, or test sentences) in order to ensure parallelism. As a result, for English, Russian, and French, the seed was already initialized with parallel sentences, and the iterative algorithm only had to "fill the remaining gaps" in the supertag coverage.

For future languages added to RRGparbank, we will just add all sentences aligned to the English seed data. This is the case for Farsi, for instance, for which alignments were added only recently.

[6] For copyright reasons, we cannot provide all annotated sentences in the other languages.

5. Applications

One of the motivating factors behind RRGparbank is to create a sufficiently large linguistic resource to be used in different NLP contexts. This is made possible by the formalization of RRG and the extraction of formal grammars for the languages in RRGparbank. These grammars consist of elementary tree templates (i.e., *supertags*). They can be used to formulate compositional analyses of sentence syntax and semantics, and to design both precision grammars and statistical parsers. We use such a (syntactical) statistical parser for generating trees as starting points for annotation (cf. Section 3). Beyond this, syntactic parsers can be useful for downstream NLP tasks such as semantic parsing. In this section, we describe our statistical syntactic parsing architecture and carry out parsing experiments that demonstrate the usefulness of RRGparbank as a resource for training syntactic parsers.

As mentioned in Section 2.2, the formalization of RRG that underlies RRGparbank is based on Tree Wrapping Grammars (TWG). TWGs can be extracted from treebanks using an automatic extraction process described in Bladier et al. (2020a). TWGs typically consist of several thousand unlexicalized elementary trees, about half of which appear only once in the corpus. As an example, Figure 10 shows the clause 'what you expect me to say' from Figure 6 annotated with elementary tree templates. TWGs can be used for statistical parsing, for example with the parser ParTAGe [7] (Waszczuk, 2017; Bladier et al., 2019; Bladier et al., 2020b). The pipeline of this parser consists of supertagging (i.e. assigning the *n*-best elementary tree templates to each word in a sentence) and a subsequent A* parsing step.

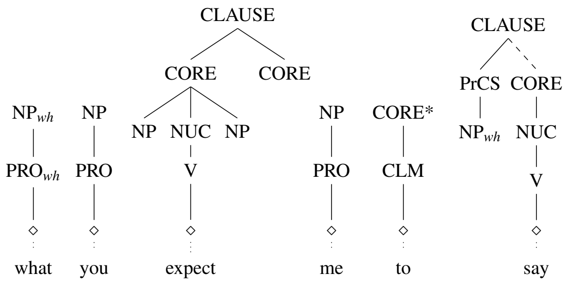


Figure 10: Extracted TWG supertags for the clause 'what you expect me to say' from Fig. 6. The sister-adjoining tree *to* is marked with an asterisk on the root node. The wrapping elementary tree *say* has a dominance link, notated as a dashed edge.

For our parsing experiments, we extract TWGs from the English, German, French, and Russian gold and silver subcorpora of a pre-release snapshot[8] of RRGparbank (including all training data, not just the one from the seed subcorpora). We train the statistical TWG parser ParTAGe using training and development sets and parse the corresponding test set to evaluate the language models. Table 2 gives an overview of the number of sentences and elementary trees for different languages. Many of the extracted supertags are common for all grammars. We found 426 supertags which appear in all four extracted TWGs.

We fine-tune the multilingual BERT model[9] and single-language BERT models for the supertagging component of the parser (similar to Schmidt (2021)) and compare the parsing accuracies. The experimental results are given in Table 3. We use the following single-language Transformer models: *bert-base-cased*[10] for English, *bert-based-german-cased*[11] for German, *camembert-base*[12] for French, *rubert-base-cased-sentence*[13] for Russian, and *bert-base-parsbert-peymaner-uncased* [14] for Farsi. We train all models for 20 epochs and use the same hyper-parameters across all models (see Table 4).

The results show that the TWG grammars extracted from RRGparbank have sufficient quality to be used for statistical multilingual parsing and that the parser trained on these grammars generalizes well. We also observe that the parser based on the multilingual model shows better performance compared to single-language models for all languages except English[15]. The single-language models on the other hand show a higher number of exactly matching parses. We assume that the better performance of the multilingual model can be explained by the cross-lingual transfer property of the multilingual BERT model (Wang et al., 2019; Ahmad et al., 2021) and some overfitting of the monolingual models. It would be interesting however to explore which role is played by the supertags common in all languages for the better performance of the multilingual parsing model. In our future work we will include further languages in the parsing experiments as the annotation of RRGparbank continues. We also plan to explore how to use extracted grammars and trained parsing models for cross-lingual parsing.

6. Conclusion

In this paper, we presented the first release of RRGparbank, a parallel treebank based on George Orwell's novel *1984* and its translations. The sentences in the corpus are annotated with RRG structures. For English, we include the entire novel, while for other languages (so far German, French and Russian), we provide a parallel seed corpus.

[7] https://rrgparser.phil.hhu.de/

[8] The data were downloaded on 2021-12-23.

[9] https://huggingface.co/bert-base-multilingual-cased

[10] https://huggingface.co/bert-base-cased

[11] https://huggingface.co/bert-base-german-cased

[12] https://huggingface.co/camembert-base

[13] https://huggingface.co/DeepPavlov/rubert-base-cased-sentence

[14] https://huggingface.co/HooshvareLab/bert-base-parsbert-peymaner-uncased

[15] All fine-tuned multilingual and single language models can be downloaded from the TWG parsing repository https://github.com/TaniaBladier/Transformer-based-TWG-parsing.

lang.	train	dev.	test	TWG size
en	4635 (4432)	574 (535)	566 (532)	3861 (2378)
de	4452 (1440)	566 (189)	561 (189)	4590 (2956)
fr	2324 (238)	273 (27)	289 (30)	2272 (1388)
ru	3877 (712)	480 (91)	486 (92)	3425 (2295)
fa	1169 (0)	146 (0)	128 (0)	1532 (992)
total	16457	2039	2030	–

Table 2: Number of sentences in data split for parsing experiments. The number in brackets indicate the gold sentences among the train, development and test data. The column TWG size shows the number of elementary trees in the extracted grammars, the numbers in brackets show how many supertags appear only once in each training set. Please note that the annotation of RRGparbank is not yet finished.

	multilingual model	single-language models	exact match (mult. model)	exact match (sing. model)	# sents	∅ len.
en	86.27	**86.56**	122	155	566	15.43
de	**85.19**	84.15	95	80	561	13.86
fr	**85.68**	85.21	66	71	289	11.66
ru	**86.16**	84.74	115	108	486	9.68
fa	**80.80**	74.37	37	17	127	8.66

Table 3: Parsing results (labeled F1 score) with the ParTAGe parser based on a fine-tuned multilingual BERT model and single-language BERT models. The results are shown for the test data without considering punctuation and function tags.

Hyper-parameters	**Value**
Max_seq_length	128
Train batch sizes	8
Learning rate	4e-05
Optimizer	AdamW
Lower case	False
Attention probability dropout rate	0.1
Hidden layer activation function	gelu
Hidden size	768
Warmup proportion	0.06
Warmup steps	1337
Number of hidden layers	12
Number of attend heads	12
Number of training epochs	20

Table 4: Hyper-parameters of the Transformer models.

RRGparbank is a valuable resource for several reasons. First, while building the treebank, we encountered numerous constructions that had not been taken into consideration in the RRG literature and for which we propose an analysis, documented in the guidelines. In this sense, RRGparbank contributes to the domain of syntactic analyses in RRG. Second, by building a treebank, in particular a parallel treebank, data-driven syntactic processing such as the parsing application presented in Section 5 become possible. Third, RRGparbank, together with options for download and for search, and also in combination with supertag extractions, enables corpus-based investigations of RRG structures, also across languages.

In the future, we plan to add further languages. Furthermore, besides syntactic annotations, we also started annotating semantic roles.

7. Acknowledgments

We thank all those that were involved in the creation and annotation of RRGparbank, aside from the authors of this paper (David Arps, Davy Baardink, Kata Balogh, Elif Benli, Andreas van Cranenburgh, Sarah Fabian, Naima Grebe). We also thank Behrang QasemiZadeh, Jule Pohlmann and Roland Eibers for preparing the German data. Furthermore, we are grateful to Robert D. Van Valin, Jr. for numerous invaluable in-depth discussions of annotation decisions. We also thank Jakub Waszczuk and Svetlana Schmidt who implemented parts of the ParTAGe parsing pipeline. This work was carried out as a part of the research project TreeGraSP[16] funded by a Consolidator Grant of the European Research Council (ERC).

8. Bibliographical References

Ahmad, W. U., Li, H., Chang, K.-W., and Mehdad, Y. (2021). Syntax-augmented multilingual BERT for cross-lingual transfer. *arXiv preprint arXiv:2106.02134*.

Arps, D., Bladier, T., and Kallmeyer, L. (2019). Chart-based RRG parsing for automatically extracted and hand-crafted RRG grammars. In *Role and Reference Grammar RRG Conference 2019*. University at Buffalo.

[16] https://treegrasp.phil.hhu.de

Bladier, T., van Cranenburgh, A., Evang, K., Kallmeyer, L., Möllemann, R., and Osswald, R. (2018). RRGbank: a Role and Reference Grammar Corpus of Syntactic Structures Extracted from the Penn Treebank. In *Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories (TLT 2018), December 13–14, 2018, Oslo University, Norway*, pages 5–16. Linköping University Electronic Press.

Bladier, T., Waszczuk, J., Kallmeyer, L., and Janke, J. (2019). From partial neural graph-based LTAG parsing towards full parsing. *Computational Linguistics in the Netherlands Journal*, 9:3–26, Dec.

Bladier, T., Kallmeyer, L., Osswald, R., and Waszczuk, J. (2020a). Automatic extraction of tree-wrapping grammars for multiple languages. In *Proceedings of the 19th Workshop on Treebanks and Linguistic Theories*, pages 55–61.

Bladier, T., Waszczuk, J., and Kallmeyer, L. (2020b). Statistical parsing of tree wrapping grammars. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6759–6766.

Brants, T. (1997). The NEGRA export format. CLAUS Report 98, Universität des Saarlandes, Computerlinguistik, Saarbrücken, Germany.

Chiarcos, C. and Fäth, C. (2019). Graph-Based Annotation Engineering: Towards a Gold Corpus for Role and Reference Grammar. In Maria Eskevich, et al., editors, *2nd Conference on Language, Data and Knowledge (LDK 2019)*, volume 70 of *OpenAccess Series in Informatics (OASIcs)*, pages 9:1–9:11, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Collins, M. (1997). Three generative, lexicalised models for statistical parsing. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 16–23, Madrid, Spain, July. Association for Computational Linguistics.

Erjavec, T. (2017). MULTEXT-East. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, pages 441–462, Dordrecht. Springer Netherlands.

Evang, K., Bladier, T., Kallmeyer, L., and Petitjean, S. (2021). Bootstrapping Role and Reference Grammar treebanks via Universal Dependencies. In *Proceedings of Universal Dependencies Workshop 2021 (UDW 2021)*.

Joshi, A. K. and Schabes, Y. (1997). Tree-adjoining grammars. In *Handbook of formal languages*, pages 69–123. Springer.

Kallmeyer, L. and Osswald, R. (2017). Combining predicate-argument structure and operator projection: Clause structure in role and reference grammar. In *Proceedings of the 13th International Workshop on Tree Adjoining Grammars and Related Formalisms*, pages 61–70.

Kallmeyer, L., Osswald, R., and Van Valin, Jr., R. D. (2013). Tree Wrapping for Role and Reference Grammar. In G. Morrill et al., editors, *Formal Grammar 2012/2013*, volume 8036 of *LNCS*, pages 175–190. Springer.

Orwell, G. (1972). *1984*. Gallimard. French translation by Amélie Audiberti (first published 1950).

Orwell, G. (2003). *1984*. Ullstein, 37th edition. German translation by Kurt Wagenseil (first published 1950 by Alfons Bürger Verlag).

Osswald, R. and Kallmeyer, L. (2018). Towards a formalization of Role and Reference Grammar. In Rolf Kailuweit, et al., editors, *Applying and Expanding Role and Reference Grammar*, pages 355–378. Albert-Ludwigs-Universität, Universitätsbibliothek. [NIHIN studies], Freiburg.

Rohde, D. L. (2005). TGrep2 user manual version 1.15. Massachusetts Institute of Technology.

Schmidt, S. (2021). *Transformers-based Supertagging Model for Statistical Parsing of Tree Wrapping Grammars*. Bachelor's thesis, Heinrich-Heine-Universität Düsseldorf.

van Cranenburgh, A., Scha, R., and Bod, R. (2016). Data-oriented parsing with discontinuous constituents and function tags. *Journal of Language Modelling*, 4(1):57–111.

Van Valin, Jr., R. D. and LaPolla, R. J. (1997). *Syntax: Structure, meaning, and function*. Cambridge University Press.

Van Valin, Jr., R. D. (2005). *Exploring the syntax-semantics interface*. Cambridge University Press.

Van Valin, Jr., R. D. (2010). Role and Reference Grammar as a framework for linguistic analysis. In Bernd Heine et al., editors, *The Oxford Handbook of Linguistic Analysis*, pages 703–738. Oxford University Press, Oxford.

Wang, Z., Mayhew, S., Roth, D., et al. (2019). Cross-lingual ability of multilingual BERT: An empirical study. *arXiv preprint arXiv:1912.07840*.

Waszczuk, J. (2017). *Leveraging MWEs in practical TAG parsing: towards the best of the two worlds*. Ph.D. thesis, Université François Rabelais Tours.