TLT 2020

Proceedings of the 19th International Workshop on Treebanks and Linguistic Theories

27–28 October, 2020
University of Düsseldorf
Düsseldorf, Germany

hhu Heinrich Heine
Universität
Düsseldorf

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN 978-1-952148-01-9

Introduction

Welcome to the 19th edition of the International Workshop on Treebanks and Linguistic Theories! It was meant to take place in Düsseldorf, but like so many events in 2020 had to pivot to online-only. We hope it will be a great experience for everyone all the same!

TLT's aim is to bring together developers and users of linguistically annotated natural language corpora. It addresses all aspects of treebank design, development, and use. By "treebank" we mean any pairing of natural language data (spoken, signed, or written) with annotations of linguistic structure at various levels of analysis, including e.g., morpho-phonology, syntax, semantics, and discourse. Annotations can take any form, including trees and general graphs.

The program of TLT 2020 reflects this broad view of work on treebanks. It includes papers on the construction of annotated resources, parsing, typology and universals, under-resourced and historical languages, and new tools for processing and querying. The program shows that an increasing amount of work in the treebank space – as well as in computational linguistics and natural language processing in general – is multilingual, looking at multiple languages from the start instead of just one. Another increasingly important theme is semantic annotation. Both themes are reflected in our invited talks. Miryam de Lhoneux will talk about parsing multiple languages, especially truly low-resource ones, about when cross-lingual learning helps and where more work is needed. And Johan Bos will talk about possibilities and difficulties in large-scale deep semantic annotation, and present a new method that may make it easier. We are delighted they accepted our invitations and we include their abstracts in this volume.

We received a total of 4 short paper submissions, of which 3 were accepted (75%), and a total of 13 long paper submissions (not counting one submission that was withdrawn), of which 11 were accepted (85%) following the reviews by our program committee. We are very grateful for the hard work of the reviewers, as well as for that of the authors, especially seeing as moving the event online resulted in a tight schedule and a video requirement.

This will be the first TLT that is held online. We opted for a setup where a regular two-day schedule of sessions takes place via video chat with talks and Q&A, but talks are pre-recorded to minimize the impact of any technical difficulties, and to make them more accessible, e.g., to participants in different timezones. A social event will also take place via video chat on the eve of the workshop. In parallel, we use text chat for asynchronous communication between participants before, during and after the workshop. We hope that it will work out well and inspire people to come back for many more TLTs, online and offline!

Kilian Evang, Laura Kallmeyer, Rafael Ehren, Simon Petitjean, Esther Seyffarth, and Djamé Seddah

Düsseldorf & Paris

October 2020

Organisers:

Kilian Evang
Laura Kallmeyer
Rafael Ehren
Simon Petitjean
Esther Seyffarth
Djamé Seddah

Program Committee:

Lasha Abzianidze, Patricia Amaral, Emily M. Bender, Johan Bos, Cristina Bosco, Giuseppe Giovanni Antonio Celano, Silvie Cinková, Daniel Dakota, Miryam de Lhoneux, Jennifer Foster, Carlos Gómez-Rodríguez, Daniel Hershcovich, Sandra Kübler, François Lareau, Nicholas Lester, Haitao Liu, Nicolas Mazziotta, Alexander Mehler, Yusuke Miyao, Jiří Mírovský, Sven Naumann, Joakim Nivre, Pierre Nugues, Stephan Oepen, Alain Polguère, Rudolf Rosa, Rik van Noord, Amir Zeldes

Invited Speakers:

Johan Bos, University of Groningen
Miryam de Lhoneux, University of Copenhagen

Invited Talks

Johan Bos: Grammar, Meaning & Annotation

What is the role of computational grammars in semantic annotation? In the Parallel Meaning Bank, grammar plays a pivotal role. This has good sides, and bad sides. It is good, because annotation is ensured to be carried out in a systematic, consistent and efficient way. But it can also be counterproductive, as linguistic input can be full of surprises. In such cases the grammar is a showstopper. Well, you might say, why not bypass the grammar in such cases? Sure, but annotating meanings from scratch is not straightforward when the targets are expressive semantic representations, such as the Discourse Representation Structures from Discourse Representation Theory used in the Parallel Meaning Bank. I present a new notation for these meaning representations: without variables, without explicit recursion, and without reliance on grammar.

Miryam de Lhoneux: Parsing Typologically Diverse Languages

This talk is about parsing typologically diverse languages. I first argue that the Universal Dependencies (UD) dataset is the best multilingual dataset that we currently have and allows us to ask general questions that are relevant for multilingual NLP. I then ask the question of how well our current parsers generalize across languages and the question of how we evaluate that.

I subsequently ask the question of how accurate our parsers currently are for truly low-resource languages. I explain recent developments in cross-lingual learning that are great at leveraging data from related languages and that improve parsing accuracy for low-resource languages. I show that for low-resource languages for which we do not have a high-resource related language, our parsers are currently highly inaccurate. Since such cases represent the majority of world languages, we might want to shift our focus on these. I finally suggest that we may find answers in the use of typological information, discuss work that has tried to do that and highlight what more can be done.

Table of Contents

Clause-Level Tense, Mood, Voice and Modality Tagging for German

Tillmann Dönicke
University of Göttingen
Göttingen Centre for Digital Humanities
Papendiek 16, 37073 Göttingen, Germany
tillmann.doenicke@uni-goettingen.de

Abstract

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

1 Introduction

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

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

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

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

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

The difference between syntax and semantics also affects the other features under consideration. The presence of a modal verb, for example, can cause multiple semantic interpretations: *he must work* is ambiguous between *he is required to work* (deontic interpretation) and *he is very likely to work [according to what the speaker knows]* (epistemic interpretation) (Viebahn and Vetter, 2016; Tarvainen, 1976).

Tense + Aspect	Alternate names	Example (indicative, active)
present imperfect	present	*sieht* ‘sees’
present perfect	perfect	*gesehen hat* ‘has seen’
past imperfect	preterite, imperfect	*sah* ‘saw’
past perfect	pluperfect	*gesehen hatte* ‘had seen’
future imperfect	future, future I	*sehen wird* ‘will see’
future perfect	future II	*gesehen haben wird* ‘will have seen’

Table 1: Tense–aspect combinations in German.

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

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

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

2 Inflection and Government in German Clauses

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

German further distinguishes four moods: indicative, present subjunctive (subjunctive I), past subjunctive (subjunctive II) and imperative, as well as three voices: active, dynamic passive and static passive[1]. All of these are expressed by combinations of the three auxiliary verbs mentioned above.

2.1 Word Order

The basic German word order is S-O-V. All verbs are positioned at the end of a clause; starting with the syntactically lowest verb and ending with the syntactically highest verb. However, this ordering is only maintained in subordinate clauses; in main clauses, the finite verb (which is always the syntactically highest verb) moves to verb-second position[2]:

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

(3) a. i. Er wird gefüttert [und verschlingt seinen Fraß].
ii. He is/gets fed [and is devouring his food].
b. i. Er ist gefüttert [und schläft jetzt].
ii. He is/*gets fed [and is now sleeping].

[2] In polar questions, the finite verb moves to sentence-initial position; in subordinate clauses, the finite verb may move to the so-called *Oberfeld* (cf. e.g. Hinrichs (2016)). For this paper, it is enough to say that the finite verb can move to a position preceding the non-finite verbs.

(4) a. i. (dass) er sie gesehen hatte.

ii. (that) he had seen her.

b. i. $\text{Er}\ \ \underset{\uparrow\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|}{\text{hatte}_1\ \ \text{sie}\ \ \text{gesehen}\ \ t_1}.$

ii. He had seen her.

English, as an S-V-O language, employs the exact opposite order of verbs. In other words, the direction of verbal government is right-to-left in German, and left-to-right in English:

(5) i. (dass) er sie $\underleftarrow{\text{gesehen haben wird}}$.

ii. (that) he $\underrightarrow{\text{will have seen}}$ her.

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

2.2 Morphological vs. Clausal Features

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

$$R\left(\left\{\begin{bmatrix}\textsc{lemma} & \textit{sehen}\\ \textsc{type} & \text{main}\\ \textsc{form} & \text{participle}\\ \textsc{aspect} & \text{perfect}\\ \textsc{voice} & \text{passive}\end{bmatrix},\begin{bmatrix}\textsc{lemma} & \textit{haben}\\ \textsc{type} & \text{auxiliary}\\ \textsc{form} & \text{finite}\\ \textsc{tense} & \text{past}\\ \textsc{mood} & \text{indicative}\\ \textsc{voice} & \text{active}\end{bmatrix}\right\}\right)=\begin{bmatrix}\textsc{form} & \text{finite}\\ \textsc{tense} & \text{past}\\ \textsc{aspect} & \text{perfect}\\ \textsc{mood} & \text{indicative}\\ \textsc{voice} & \text{active}\end{bmatrix}$$

2.3 Modal Verbs

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

(6) a. i. (dass) er sie gesehen$_{[\textsc{form}\ \text{participle}]}$ hat$_{\begin{bmatrix}\textsc{form} & \text{finite}\\ \textsc{tense} & \text{present}\end{bmatrix}}$.

ii. (that) he has$_{\begin{bmatrix}\textsc{form} & \text{finite}\\ \textsc{tense} & \text{present}\end{bmatrix}}$ seen$_{[\textsc{form}\ \text{participle}]}$ her.

b. i. (dass) er sie gesehen$_{[\textsc{form}\ \text{participle}]}$ haben$_{[\textsc{form}\ \text{infinitive}]}$ muss$_{\begin{bmatrix}\textsc{form} & \text{finite}\\ \textsc{tense} & \text{present}\end{bmatrix}}$.

ii. (that) he must$_{\begin{bmatrix}\textsc{form} & \text{finite}\\ \textsc{tense} & \text{present}\end{bmatrix}}$ have$_{[\textsc{form}\ \text{infinitive}]}$ seen$_{[\textsc{form}\ \text{participle}]}$ her.

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

[3] In English, the shifting of inflectional features is also observable in negation or emphasis with the auxiliary verb *do*:

(7) a. He has$_{\begin{bmatrix}\textsc{form} & \text{finite}\\ \textsc{tense} & \text{present}\end{bmatrix}}$ seen$_{[\textsc{form}\ \text{participle}]}$ her.

b. He does$_{\begin{bmatrix}\textsc{form} & \text{finite}\\ \textsc{tense} & \text{present}\end{bmatrix}}$ (not) have$_{[\textsc{form}\ \text{infinitive}]}$ seen$_{[\textsc{form}\ \text{participle}]}$ her.

2.4 Substitute Infinitives

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

(8) a. i. (dass) er sie sehen müssen/*gemusst hat.
ii. (that) he has had to see her.
b. i. (dass) er sie singen hören/gehört hat.
ii. (that) he has heard her sing.

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

3 Previous Approaches and Corpora for German

3.1 Bögel et al. (2014)

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

$$R\left(\left\{\begin{bmatrix}\text{TYPE} & \text{main}\\ \text{FORM} & \text{participle}\end{bmatrix}, \begin{bmatrix}\text{TYPE} & \text{auxiliary}\\ \text{TENSE} & \text{present}\end{bmatrix}\right\}\right) = \begin{bmatrix}\text{TENSE} & \text{present}\\ \text{ASPECT} & \text{perfect}\end{bmatrix},$$

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

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

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

3.2 Ramm et al. (2017)

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

[4] `http://heureclea.de/`

[5] `http://uima.apache.org/`

[6] `http://www.catma.de/`

[7] `https://code.google.com/archive/p/mate-tools/`

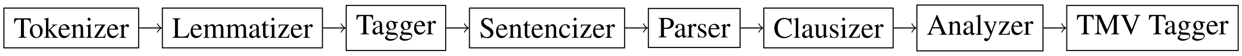


Figure 1: Our processing pipeline: from raw text to clause-level tagging.

rule-set similarly as in Bögel et al. (2014). The output of the tool is a table format providing all main verbs and tense/mood/voice tags as well as the clauses which contain the verbs.

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

4 Method/Implementation

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

4.1 Preprocessing

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

4.2 Universal Dependency Parsing

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

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

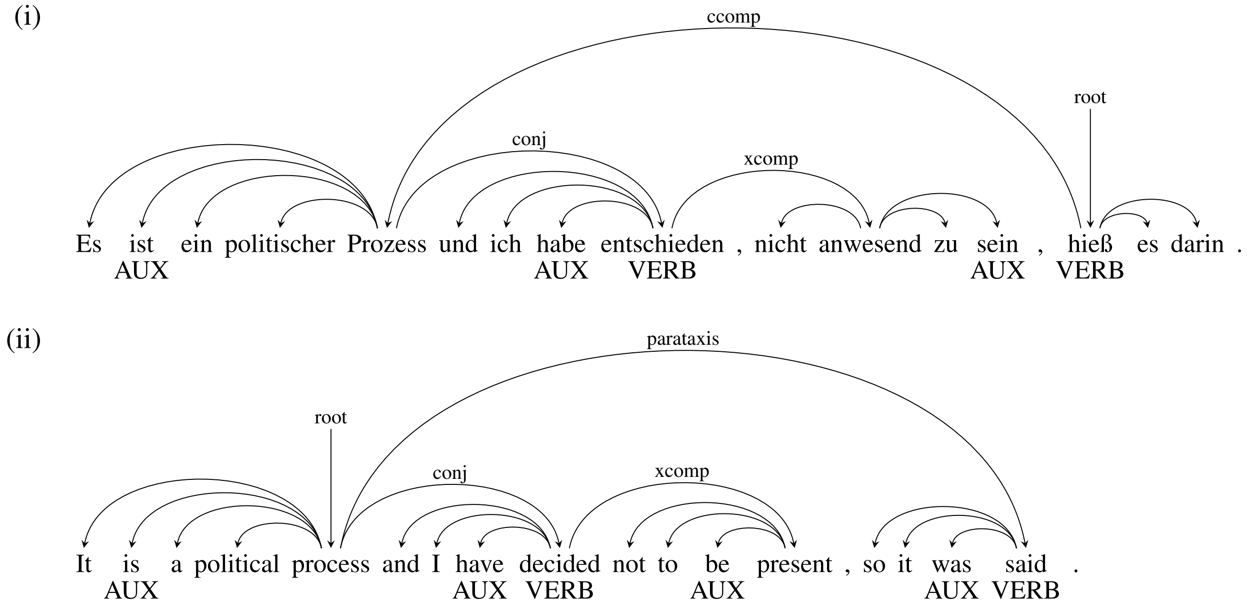
4.3 Crosslinguistic Clause Splitting

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

[8] `https://spacy.io/`

[9] The pre-trained German model is available at `https://spacy.io/models/de#de_core_news_lg`.

[10] `https://universaldependencies.org/`


(i)
ccomp
root
conj
xcomp
Es ist ein politischer Prozess und ich habe entschieden , nicht anwesend zu sein , hieß es darin .
AUX
AUX VERB
AUX VERB
(ii)
parataxis
root
conj
xcomp
It is a political process and I have decided not to be present , so it was said .
AUX
AUX VERB
AUX
AUX VERB

(iii)

Relation	Clause
`ccomp`	Es ist ein politischer Prozess
`conj`	und ich habe entschieden
`xcomp`	nicht anwesend zu sein
`root`	hieß es darin

(iv)

Relation	Clause
`root`	It is a political process
`conj`	and I have decided
`xcomp`	not to be present
`parataxis`	so it was said

Figure 2: Dependency trees for a sentence in the (i) German and (ii) English PUD treebanks (ID: n02030005). Relations are only labelled if marking a clause. Tables (iii) and (iv) show the extracted clauses; verbs are underlined.

Our clausizer is applicable to all texts with UD parse trees, either after being parsed accordingly (e.g. with spaCy) or after being manually annotated (e.g. within the UD treebanks project). Figure 2 shows a sentence from the German and English PUD treebanks. Each sentence contains four clauses. We implemented the clausizer to recursively detect nested clauses, e.g. two clauses are detected in (9): *Der Mann lacht* 'The man laughs' and *der die Kuh sah* 'who saw the cow'.

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

4.4 Morphological Analysis

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

4.5 TMV Tagging

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

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

[11] `https://github.com/DuyguA/DEMorphy`

Algorithm 1: Compute features of a clause C

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

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

$\overset{\cup}{\leftarrow}$ and $\overset{\sqcup}{\leftarrow}$ are augmented assignment operators for union and unification, respectively.

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

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

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

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

(11) a. i. (dass) er sprechen gelernt hatte.
ii. (that) he had learned to speak.
b. i. (dass) er dort gewesen war.
ii. (that) he had been there.
c. i. (dass) er Englisch kann.
ii. (that) he can [speak] English.

Only the verbs from the main verb to the finite verb are interesting for TMV tagging, because the main verb is the syntactically lowest verb of a composite verb form; all other verbs which precede the main verb are removed from V (l. 13). M contains the feature structures for every word, i.e. m_i ($1 \le i \le |V|$)

is a set of possible morphological analyses for v_i (l. 14). If the second verb from the right $v_{|V|-1}$ is a potential substitute infinitive, the feature structure of a perfect participle is added to $m_{|V|-1}$ (ll. 15–16). Having all verbs of interest together, the features of modal verbs are shifted to their predecessors as described in section 2.3 (ll. 17–19).

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

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

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

As soon as one or more analyses are found, one of them is selected and returned (ll. 27–31). In German, most verbs express the perfect aspect with the auxiliary verb *haben* 'have' (e.g. *hat gesehen* 'has seen') but some use *sein* 'be' (e.g. *ist gegangen* 'is gone') and others can use either depending on the context or regional varieties (whereas in English it is almost always *have*). Since forms of *sein* can not only mark perfect aspect but also static passive, this causes ambiguous verb forms. To resolve these ambiguities, we filter the analyses with respect to the main verb's possible perfect auxiliaries (this is also done by Ramm et al. (2017)). We extracted the possible perfect auxiliaries for every German verb in the German Wiktionary[12].

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

5 Evaluation

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

5.1 Annotation

We annotated the German translation of the preface of *Don Quijote* by Miguel de Cervantes Saavedra[13] (3,200 tokens) which contains a lot of complex (multi-clause) sentences and examples for all six tenses, four moods, three voices and the modal verbs *können* 'can', *mögen* 'may', *müssen* 'must', *sollen* 'shall' and *wollen* 'want'. Two annotators annotated the text with tense. After calculating the inter-annotator agreement ($\kappa = 96\%$, Fleiss et al. (2003)), we combined the two annotations into a gold annotation and extended it with finiteness, mood, voice and the modal verbs involved in a verb form.

We used the official German Duden grammar (Dudenredaktion, 2009, pp. 476 ff.) as reference guide for our annotation of tense, mood and voice. We also annotated non-finite clauses (with infinitive or participle forms) with tense and voice[14]—non-finite forms do not feature mood—, whereas Ramm et al. (2017) only consider finite verb forms and in heureCLÉA non-finite clauses are either not annotated or receive the tense of the corresponding matrix clause.

[12] `https://dumps.wikimedia.org/dewiktionary/`

[13] The text is available at `https://www.projekt-gutenberg.org/cervante/quijote1/quijote1.html`.

[14] It is debatable whether infinitives and participles feature tense or only aspect. This is, however, only a matter of definition. Since we only tag tense–aspect combinations, we use the present imperfect or present perfect for all non-finite verb forms.

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

Table 2: Inter-annotator agreements and tense tagging accuracies for the heureCLÉA corpus and/or our test text. Numbers in brackets are copied from Bögel et al. (2014). Accuracies are shown for all tokens or only main verbs.

	Fin.	Tense	Mood	Voice	Mod.
Ramm et al. (2017)	82.7	71.5	75.7	82.5	–
this work	88.1	92.9	82.2	93.5	79.8
		92.6	79.0	93.8	79.8

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

5.2 Tense Evaluation

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

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

5.3 TMV and Modality Evaluation

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

5.4 Clause Evaluation

We also tested the sole performance of our clausizer. For the evaluation on *Don Quijote*, we compared the clause boundaries of the annotation B_{gold} with the predicted boundaries B_{pred} (cf. Jurish and Würzner (2013)). We define a clause boundary as a tuple (e_i, s_{i+1}) of character positions, namely the end position e_i of a clause and the start position s_{i+1} of the next clause in the text.[15] Precision, recall and F$_1$-score are calculated respectively as

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

[15] A clause inside another clause produces the same boundaries as three subsequent clauses. It is not possible to distinguish these cases in the calculations, because the annotation format does not distinguish them either.

	Don Quijote	CoNLL-2001	
	clause boundaries	clause starts	clause ends
Gold instances	443	4497	3364
Pred. instances	388	4598	4598
Precision	87.1	72.7	66.4
Recall	76.3	74.3	90.8
F_1-score	81.3	73.5	76.7

Table 4: Clause splitting precisions, recalls and F_1-scores of our clausizer on our test text (German) and the CoNLL-2001 shared task test set (English). The first two rows show the number of gold and predicted instances.

We additionally applied the clausizer to the test set from the CoNLL-2001 shared task on clause identification (in English) (Tjong Kim Sang and Déjean, 2001). The goal in the shared task was the automatic detection of 1) start tokens, 2) end tokens, and 3) entire spans of clauses. The evaluation of our tool on this dataset is somewhat problematic because the concept of what a clause is differs in several aspects. The main difference is that every token belongs to exactly one clause in our concept, namely the syntactically deepest clause where it appears in, whereas a token also belongs to all of its superordinate clauses in the shared task's concept. Therefore, our clausizer would definitely not detect the same spans as in the test set. However, we can evaluate the clausizer on the detection of clause starts and ends; here, the actual number of clauses that start or end on those positions is not considered. For the prediction, we used the sentence boundaries and part-of-speech tags as in the test set, the pre-trained English spaCy model[16] for parsing, and our clausizer in the same configuration as for German, with a small modification: As noted earlier, the English spaCy model does not use UD relations, but instead produces the earlier Stanford relations (de Marneffe and Manning, 2008) which are quite similar to the UD relations. We added `csubjpass`, `intj`, `pcomp`, and `relcl` (which do not appear in the UD inventory) to the list of clause-marking relations.

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

6 Discussion

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

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

[16] The pre-trained English model is available at `https://spacy.io/models/en#en_core_web_lg`.

complete look-up table is not as prone to errors as a set of rules.

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

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

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

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

7 Future Work

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

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

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

8 Conclusion

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

Acknowledgements

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

References

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Edward L. Keenan and Matthew S. Dryer, 2007. *Passive in the world's languages*, volume 1, pages 325–361. Cambridge University Press, 2nd edition.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Conference Proceedings: The Tenth Machine Translation Summit*, Phuket, Thailand.

Elisabeth Leiss. 2008. The silent and aspect-driven patterns of deonticity and epistemicity. *Modality–aspect interfaces: Implications and typological solutions*, pages 15–41.

Jo-Wang Lin. 2005. Time in a Language Without Tense: The Case of Chinese. *Journal of Semantics*, 23(1):1–53, 09.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia, May. European Language Resources Association (ELRA).

James Pustejovsky and Marc Verhagen. 2009. SemEval-2010 task 13: Evaluating events, time expressions, and temporal relations (TempEval-2). In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions (SEW-2009)*, pages 112–116, Boulder, Colorado, June. Association for Computational Linguistics.

Anita Ramm, Sharid Loáiciga, Annemarie Friedrich, and Alexander Fraser. 2017. Annotating tense, mood and voice for English, French and German. In *Proceedings of ACL 2017, System Demonstrations*, pages 1–6, Vancouver, Canada, July. Association for Computational Linguistics.

Hans Reichenbach. 1947. Elements of symbolic logic.

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to German. In *Proceedings of the ACL SIGDAT-Workshop*, Dublin, Ireland.

Anil Kumar Singh, Samar Husain, Harshit Surana, Jagadeesh Gorla, Dipti Misra Sharma, and Chinnappa Guggilla. 2007. Disambiguating tense, aspect and modality markers for correcting machine translation errors. In *Proceedings of the Conference on Recent Advances in Natural Language Processing (RANLP)*.

Jannik Strötgen and Michael Gertz. 2010. Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 321–324.

Kalevi Tarvainen. 1976. Die Modalverben im Deutschen Modus- und Tempussystem. *Neuphilologische Mitteilungen*, 77(1):9–24.

Erik F. Tjong Kim Sang and Hervé Déjean. 2001. Introduction to the CoNLL-2001 shared task: clause identification. In *Proceedings of the ACL 2001 Workshop on Computational Natural Language Learning (ConLL)*.

Emanuel Viebahn and Barbara Vetter. 2016. How many meanings for 'may'? The case for modal polysemy. *Philosopher's Imprint*, 16(10).

Nessa Wolfson. 1978. A feature of performed narrative: The conversational historical present. *Language in Society*, 7(2):215–237.

Elizabeth Zeitoun, Lillian M. Huang, Marie M. Yeh, Anna H. Chang, and Joy J. Wu. 1996. The temporal, aspectual, and modal systems of some formosan languages: A typological perspective. *Oceanic Linguistics*, 35(1):21–56.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinková, Jan Hajič jr., Jaroslava Hlaváčová, Václava Kettnerová, Zdeňka Urešová, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher D. Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria de Paiva, Kira Droganova, Héctor Martínez Alonso, Çağrı Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonça, Tatiana Lando, Rattima Nitisaroj, and Josie Li. 2017. CoNLL 2017 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada, August. Association for Computational Linguistics.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Željko Agić, Lars Ahrenberg, Chika Kennedy Ajede, Gabrielė Aleksandravičiūtė, Lene Antonsen, Katya Aplonova, Angelina Aquino, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Furkan Atmaca, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Victoria Basmov, Colin Batchelor, John Bauer, Kepa Bengoetxea, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Kristina Brokaitė, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Savas Cetin, Fabricio Chalub, Ethan Chi, Jinho Choi, Yongseok Cho, Jayeol Chun, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Hanne Eckhoff, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Jena Hwang, Takumi Ikeda, Radu Ion, Elena Irimia, Ọlájídé Ishola, Tomáš Jelínek, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Lucia Lam, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phuong Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Cheuk Ying Li, Josie Li, Keying Li, KyungTae Lim, Yuan Li, Nikola Ljubešić, Olga Loginova, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Tomohiko Morioka, Shinsuke Mori, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Luong Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ Olúòkun, Mai Omura, Emeka Onwuegbuzia, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Arzucan Özgür, Balkız Öztürk Başaran, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Guy Perrier, Daria Petrova, Slav Petrov, Jason Phelan, Jussi Piitulainen, Tommi A Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Valentin Roșca, Davide Rovati, Olga Rudina, Jack Rueter, Shoval Sadde, Benoît Sagot, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Muh Shohibussirri, Dmitry Sichinava, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Antonio Stella, Milan Straka, Jana Strnadová, Alane Suhr, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Takaaki Tanaka, Samson Tella, Isabelle Tellier, Guillaume Thomas, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sumire Uematsu, Roman Untilov, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Aya Wakasa, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Paul Widmer, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Amir Zeldes, Hanzhi Zhu, and Anna Zhuravleva. 2020. Universal Dependencies 2.6. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Andrea Zielinski, Christian Simon, and Tilman Wittl. 2009. Morphisto: Service-oriented open source morphology for German. In Cerstin Mahlow and Michael Piotrowski, editors, *State of the Art in Computational Morphology*, pages 64–75, Berlin, Heidelberg. Springer Berlin Heidelberg.

Patrick Ziering, Sina Zarrieß, and Jonas Kuhn. 2012. A corpus-based study of the German recipient passive. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1637–1644, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Appendix A. German verb forms with tense, mood, voice

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

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

Aux.	Example	Tense + Aspect	Mood (if finite)	Voice
sein	[er] ist gegangen gewesen	present perfect	indicative	static passive
sein	[er] sei gegangen gewesen	present perfect	present subjunctive	static passive
sein	[er] war gegangen	past perfect	indicative	active
sein	[er] wäre gegangen	past perfect	past subjunctive	active
sein	[er] war gegangen worden	past perfect	indicative	dynamic passive
sein	[er] wäre gegangen worden	past perfect	past subjunctive	dynamic passive
sein	[er] war gegangen gewesen	past perfect	indicative	static passive
sein	[er] wäre gegangen gewesen	past perfect	past subjunctive	static passive
sein	[er] wird gehen	future imperfect	indicative	active
sein	[er] werde gehen	future imperfect	present subjunctive	active
sein	[er] würde gehen	future imperfect	past subjunctive	active
sein	[er] wird gegangen werden	future imperfect	indicative	dynamic passive
sein	[er] werde gegangen werden	future imperfect	present subjunctive	dynamic passive
sein	[er] würde gegangen werden	future imperfect	past subjunctive	dynamic passive
sein	[er] wird gegangen sein	future imperfect	indicative	static passive
sein	[er] werde gegangen sein	future imperfect	present subjunctive	static passive
sein	[er] würde gegangen sein	future imperfect	past subjunctive	static passive
sein	[er] wird gegangen sein	future perfect	indicative	active
sein	[er] werde gegangen sein	future perfect	present subjunctive	active
sein	[er] würde gegangen sein	future perfect	past subjunctive	active
sein	[er] wird gegangen worden sein	future perfect	indicative	dynamic passive
sein	[er] werde gegangen worden sein	future perfect	present subjunctive	dynamic passive
sein	[er] würde gegangen worden sein	future perfect	past subjunctive	dynamic passive
sein	[er] wird gegangen gewesen sein	future perfect	indicative	static passive
sein	[er] werde gegangen gewesen sein	future perfect	present subjunctive	static passive
sein	[er] würde gegangen gewesen sein	future perfect	past subjunctive	static passive

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

Building a Treebank for Chinese Literature for Translation Studies

Hai Hu† **Yanting Li**‡ **Yina Patterson**† **Zuoyu Tian**†
Yiwen Zhang† **He Zhou**† **Sandra Kübler**† **Chien-Jer Charles Lin**†
†Indiana University Bloomington ‡Northwestern University
{huhai,yinama,zuoytian,yiwezhan,hzh1,skuebler,chiclin}@indiana.edu
yanting.li@northwestern.edu

Abstract

We present a new Chinese Treebank in the literary domain, the Treebank for Chinese Literature (TCL), with an aim to foster translation studies by providing an annotated collection of Chinese texts from both translated and non-translated literature. In the current stage, our constituency treebank consists of 2 069 trees, annotated and cross-checked by six Chinese linguists, following and adapting the Chinese Penn Treebank (CTB) annotation guidelines. We discuss the issues that we encountered while annotating literary texts, and we demonstrate the usefulness of our treebank by comparing it against the news portion of CTB, and by analyzing the syntactic features of non-translated literary texts and translationese in Chinese.

1 Introduction

Despite Chinese being one of the most widely spoken languages in the world, there is still a lack of diverse treebanks in terms of genres. The largest proportion of the most widely used Penn Chinese Treebank (CTB) (Xue et al., 2005) contains texts from the news domain (plus small samples from magazines, telephone transcripts, chat messages, etc.). To the best of our knowledge, the only large-scale, freely available constituency treebank in Chinese in a different domain is the Chinese Treebank in Scientific Domain (Chu et al., 2016), with text from Chinese scientific papers.

Without the availability of high-quality, expert-annotated treebanks in domains other than the above two, it is difficult for corpus linguists to compare syntactic features of multiple domains (Xiao, 2010; Zhang, 2012; Xiao and Hu, 2015), and it is difficult to train parsers beyond the news domain. Research on domain adaptation for parsing is limited by the few available domains covered in (Chinese) treebanks.

Our overarching goal is to develop a reliable parser for Chinese for translation studies of literary texts[1]. To this end, we present our initial effort to build a Chinese treebank for literary texts. Specifically, to enable the comparison of translated and non-translated Chinese, half of our texts are originally written in Chinese and the other half translated from English to Chinese. While our intention is to create a parser for translation studies, our treebank will be a valuable resource for stylistics, translation studies (Hu et al., 2018; Lin and Hu, 2018; Rubino et al., 2016), corpus linguistics research in Chinese (Wu et al., 2010), as well as for domain adaptation for Chinese parsing (Li et al., 2019). To the the best of our knowledge, our treebank is the first sizable Chinese treebank in the literary domain[2], and also the first designed specifically for translation studies.

The paper is structured as follows: Section 2 introduces the source text and the annotation guidelines. Then section 3 presents our annotation procedure and the final annotated treebank. In section 4, we analyze the linguistic characteristics of TCL, with reference to the widely used Penn Chinese Treebank. Additionally, we compare the translation and non-translation sections within TCL.

[1] We plan to use the parser to extend prior work on translationese (Hu and Kübler, 2020; Lin, 2017; Lin and Hu, 2018) to the domain of literature.

[2] We use "literary Chinese" to mean Chinese in the domain of literature, rather than "classical Chinese", which is sometimes also referred to as "literary Chinese".

Corpus	Example sentences
TCL$_{\text{original}}$	Ex.1: 身边的小贩儿嗓门儿比他还高，低着头用小叉子拢着豆芽粗吼着：豆芽儿，绿豆的，败火，贱卖，两毛了！ ‘The peddler beside him had a higher voice than him, and he lowered his head gathering the bean sprouts with a small fork and roared: bean sprouts, mung bean sprouts, relieve heatiness, low prices, only twenty cents!’ Ex.2: 后辈儿孙不负浩荡皇恩，深感五坛、八庙倒可少一点儿，可那老北京的小玩艺儿：溜个马，架个鹰，斗个蛐蛐儿，玩个鸟儿的，却绝对不能少。 ‘The descendants live up to the mighty emperor’s grace, and feel that the altars and the temples can be a little less, but the games of old Beijing: walking the horses, falconry, cricket fighting and playing with birds, definitely cannot be less.’
TCL$_{\text{translated}}$	Ex.1: 价值的确是特殊的，因为它隐而不露，所以它当然会在日后增加，尤其当这些物品被后代们视若珍宝的时候。 ‘The value is indeed special. Because it is hidden, it will increase in the future, especially when these objects are viewed as treasures by the descendants.’ Ex.2: 新闻传媒很快就对此失去了热情，警方遮遮掩掩不知所云，联邦调查局干脆说是地方当局的事而一推了之。 ‘The media soon lost interest in this; the police was trying to hide something and there was nothing concrete in their statements; FBI shirked their responsibility by saying it was an issue for the local authorities.’

Table 1: Example sentences from the original and translated section of TCL.

2 Treebank Development

2.1 Data Source

Starting from our goals of creating a treebank for original and translated Chinese literature, we have selected the literary subset from two widely used corpora of Chinese. Specifically, we use the Lancaster Corpus of Mandarin Chinese (LCMC) (McEnery and Xiao, 2004) as our source for original Chinese, and the Zhejiang-University Corpus of Translated Chinese (ZCTC) (Xiao et al., 2010) for translated Chinese. LCMC has been widely used in linguistic studies of Chinese (Duanmu, 2012; Song and Tao, 2009; Zhang, 2017). Similarly, ZCTC is considered a standard resource for translation studies in Chinese (Xiao, 2010; Xiao and Hu, 2015; Hu and Kübler, 2020).

We select the literature genre (index “K”) from both corpora, which in both cases is composed of 29 texts, each about 100 sentences. The texts are from different literary works in the 90s[3], for example, *To Live* by Yu Hua, *Memoirs of a Geisha* by Arthur Golden. We chose to annotate an equal number of sentences from each of the 29 texts since sampling from a more diverse set of texts will enhance the representativeness of the treebank.

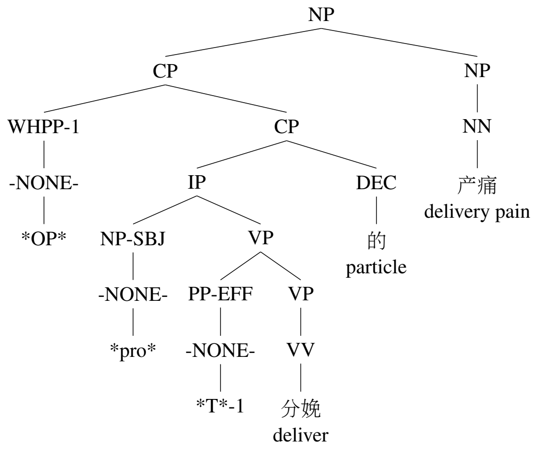
Both corpora have been segmented and part-of-speech (POS) tagged automatically using the Chinese Lexical Analysis System (Zhang et al., 2002). We did not use the segmentation and POS tags provided in the corpora because the segmentation and POS annotations are not compatible with those from the Chinese Penn Treebank, whose guidelines we follow for the syntactic annotation. In Table 1, we show example sentences from the two portions of TCL. These examples show that the language used in the literary texts is informal, and the translations show traces of English syntax.

2.2 Pre-processing

For sentence splitting, we split at the following types of punctuation signs: period (。), exclamation mark (！), question mark (？), semi-colon (；) and ellipsis (……). Then we used the default models and settings of the Stanford CoreNLP (Manning et al., 2014) to segment, POS tag, and parsed all sentences. The automatically analyzed sentences were then manually corrected by our annotators. Corrections include adjusting wrong segmentation, POS tags, and tree structures. Additionally, we add functional tags and empty categories according to our extended guidelines (see section 2.3).

In pre-processing, we encountered the following issues:

[3] The full lists can be found at `https://www.lancaster.ac.uk/fass/projects/corpus/LCMC/lcmc/kat_k.htm` and `https://www.lancaster.ac.uk/fass/projects/corpus/ZCTC/source_K.htm`.



Eng.: the pain caused by delivery (of a baby)

Figure 1: Example of an adjunct relative clause that has the new functional tag `EFF`.

OCR errors The text in LCMC are mostly “provided by the SSReader Digital Library in China”, which has a 1-3% error rate in the OCR process (McEnery and Xiao, 2004). We corrected OCR errors if we were certain of the mis-recognized characters, based on the context and the shape of the characters: For example we corrected, 存人→ 存入 ‘*to deposit*’, 陷阶→ 陷阱 ‘*trap*’, and 村当于→ 相当于 ‘*equivalent to*’.

Normalization of punctuation signs We translated all the half-width punctuation signs to full-width ones, e.g., “.” →“。”, “?” →“？”. We also normalized other punctuation signs, such as ellipsis, which are not consistent across LCMC and ZCTC.

2.3 Annotation Guidelines

We followed the guidelines of the Penn Chinese Treebank (Xue et al., 2005), but adopted modifications from the Chinese Treebank in Scientific Domain (SCTB) (Chu et al., 2016) where applicable for literary texts. We kept the constituent annotations in CTB as consistent with those of the Penn English Treebank (Marcus et al., 1993) as possible. The latest version of the treebank (V9) contains texts from the following genres: newswire, magazine articles, broadcast news/conversations, weblogs and discussion forums. No literary texts are included. CTB is based on the Theory of Government and Binding[4] (Chomsky, 1981), and uses empty categories and traces, which we also adopt in our annotation.

SCTB relies heavily on the annotation guidelines of CTB, but modifies them to better model scientific texts, such as creating specific POS tags for suffixes. Scientific writing is characterized by a high density of highly specialized technical terms created by suffixation. Since suffixation is very productive, as in VV + suffix (for example: 生育 ‘*breed*’ + 期 ‘*period*’ = ‘breeding period’), SCTB treats these technical terms as two individual words and assigns separate POS labels to suffixes such as 期 ‘*period*’. We have incorporated those annotation rules of SCTB that are applicable for literary texts. We describe the most important extensions here[5].

Adjunct Relative Clauses Adjunct relative clauses are relative clauses where the gap in the relative clause is not clearly identifiable. For example, in Figure 1, the head noun “delivery pain” is not an argument (subject or object) in the relative clause `*pro* delivers`, but rather the *effect* or *result* that is caused by delivering a baby (see translation at bottom of Figure 1).

There has been much discussion in theoretical and psycho-linguistics on how such relative clauses are generated (Cha, 1999; Lin, 2018; Patterson, 2020; Ning, 1993). CTB treats all of these as a PP modifier inside the relative clause and provides several function tags to describe the functions of the head noun, for example, TMP (temporal) and MNR (manner). In our annotation of TCL, we found many cases of

[4] See (Xue et al., 2005, p. 4).

[5] We will release a full list of added annotation guidelines along with the treebank.

effect, where the head noun describes an effect resulting from the activity described in the relative clause. Consequently, we add EFF (effect) as a new functional tag in our guidelines.

Suffixes There are two suffixes that are frequent in our literary texts but uncommon in Chinese news texts. The first suffix is the *Erhua* (i.e., rhoticization) suffix 儿 *er* (from here on SFE), and the second is the plural suffix 们 *men* (from here on SFP).

Erhua is a morpho-phonological process that adds r-coloring, in the form of the suffix *er* [ɚ], to syllables in spoken Mandarin, the written form being 儿, as in 事儿 (*thing-er*, 'thingy'), 绝活儿 (*specialty-er*, 'claim to fame'). It is usually semantically vacuous and used in informal contexts to add a diminutive sense to the stem. In Beijing Mandarin, it has been used as a marker of local identity in contrast with a cosmopolitan global identity (Zhang, 2008). In the sampled news of CTB, we found no cases of rhoticization, but in TCL$_{\text{original}}$, there are 48 such cases. This is an indication that our literary texts are more informal and colloquial than CTB news.

The plural suffix, 们 *men*, is usually attached to animate nouns, which we decided to separate from the preceding noun and label as SFP in TCL. This suffix is more frequent in TCL (145 in TCL$_{\text{original}}$ and 219 in TCL$_{\text{translated}}$, compared to 53 in the sampled CTB) and has a wider range of metaphorical usage in that it can be attached after an inanimate noun such as 眼 'eye' in the literary genre, which is rarely found in news texts.

3 The Literary Chinese Treebank

Annotation team Our tree annotation team consists of six linguists (MA/PhD students in linguistics), all native speakers of Chinese. Additionally, two experienced (computational) syntacticians are available for consultation.

Annotation procedure The annotation process consisted of four phases. In the first phase, the annotators familiarized themselves with the CTB guidelines. In the second step, each annotator annotated 10 sentences, followed by a discussion of points of uncertainty and differences in annotation. In the third phase, each annotator was assigned 230 trees to annotate. Every tree was cross-checked by a different annotator. If differences occurred, they were discussed, and the trees were corrected if necessary. Annotation issues were discussed in weekly meetings. During this process, the extended guidelines were produced, covering new cases due to the linguistic differences between news and literature, and also documenting decisions in cases of inconsistencies in the CTB. With the enhanced guidelines, each annotator annotated an additional 100 trees, after which each tree was cross-checked by a different annotator.

Size Currently, the treebank consists of 2 069 trees: 1 029 from translated literature and 1 040 from original Chinese literature, amounting to 42 054 words. These sentences are sampled from 58 works of fiction from both LCMC and ZCTC (29 each).

Inter-annotator agreement (IAA) To compute IAA, our six annotators annotated the same 47 trees, and then had a discussion to decide on the gold standard for these sentences. We compute IAA as the averaged F-measure between an annotator's trees and the agreed upon final trees. This resulted in an agreement of 92.94%, thus indicating high agreement among our annotators.

4 Analysis of TCL

It is not always clear how to evaluate a treebank, and there are many angles to investigate. In this section, our intention is to document a range of differences that give an indication of how useful the addition of this treebank will be to the existing Chinese treebanks. The investigation is mainly driven by our goal of using the treebank for translation and contrastive linguistic studies. We first look at the overall statistics of complexity across the three treebank sections. Then we investigate differences between the news and literary genres, focusing on two phenomena that are less frequent or non-existent in the CTB. Finally, we look into differences between the original and translated portions of the TCL.

In order to perform the between-genre comparison, we sampled 1 040 trees from the CTB news portion to match the number of our annotated data in TCL$_{\text{original}}$. In sampling these CTB trees, we removed the

	$\text{TCL}_{\text{original}}$	$\text{TCL}_{\text{translated}}$	CTB	`Tregex` pattern
# sent	1 040	1 029	1 040	
mean sent. length	19.74	17.92	27.77	
mean word length	1.36	1.41	1.73	
vocab. size	4 439	4 026	6 012	
mean tree depth	10.73	10.94	11.25	
# rules	27 250	24 960	34 042	
# rule types	1 800	1 484	2 167	
entropy of rules	6.84	6.67	7.38	
per 1 000 words				
# IP	175.85	178.15	128.46	`/^IP/`
# CP	47.59	55.37	47.82	`/^CP/`
# subordinate clause	1.17	3.31	0.52	`/^CP/ <1 (/^ADVP/<CS)`
# relative clause	17.73	20.83	23.61	`/^P/ <1 /^WH(NP|PP)/`

Table 2: Statistics of subsets of TCL, in comparison with the sampled news section in CTB. (Sentence and word lengths are computed based on the number of syllables, which is equivalent to the number of monosyllabic morphemes in Chinese. Tree depth refers to the greatest number of syntactic levels embedded in a constituent.)

header and trailing information about the name of the reporter or the dates, and only kept the content of the news.

4.1 Linguistic Characteristics of TCL

Linguistic complexity Here, we compare the linguistic complexity across the different treebank sections. We chose complexity for several reasons. First, it is an important linguistic feature, receiving attention from various branches of linguistics, e.g., typology (Juola, 2008), corpus linguistics (Covington and McFall, 2010; Kettunen, 2014), psycholinguistics (Futrell et al., 2015; Gibson, 1998; Hawkins, 2004; Lin, 2018), and language acquisition (Lu, 2010; O'Grady, 1997). Second, in translation studies, a well-known hypothesis states that translated texts are lexically and syntactically simpler than texts originally written in a language (Baker, 1993; Baker, 1996). Empirical results of this *simplification* hypothesis have been mixed (Laviosa-Braithwaite, 1996; Ilisei and Inkpen, 2011; Volansky et al., 2013; Hu and Kübler, 2020). TCL can provide a high quality data source for evaluating this hypothesis.

Table 2 presents a range of statistics on the two subsets of TCL and the sampled news section of CTB. We first notice that news texts have considerably longer sentences, longer words, slightly deeper trees, a larger vocabulary size, as well as considerably more rules and rule types. By rules we mean all non-terminal context-free rules extracted from the trees, e.g., `NP -> DP ADJP NP`. Rule type refers to the number of unique rules. All these criteria suggest that news texts are syntactically more complex than their literary counterparts.

In the second part of Table 2, which focuses on grammatical rules, we calculated the entropy of the distribution of grammar rules. The numbers show that the news domain has a higher entropy, indicating more uncertainty and complexity of its grammar rule distribution. The numbers in the third part of the table, however, are more diverse: While both parts of TCL has a higher number of IPs[6] (indicating more main clauses) and a higher number of subordinate clauses, CTB has more relative clauses than both $\text{TCL}_{\text{original}}$ and $\text{TCL}_{\text{translated}}$. In terms of CPs (small clauses), the translated text $\text{TCL}_{\text{translated}}$ outnumbers both $\text{TCL}_{\text{original}}$ and CTB.

Focusing on $\text{TCL}_{\text{original}}$ and $\text{TCL}_{\text{translated}}$, we observe that the original literature domain is more complex in terms of mean sentence length, vocabulary size, as well as the number of rules and rule types. This lends some support for the simplification hypothesis at both the lexical and sentence levels. However, for the other measures in Table 2, the differences are either too small or even reversed. We will look at the simplification hypothesis more closely in section 4.4.

[6] These structures were extracted using `Tregex` patterns (Levy and Andrew, 2006).

	$\text{TCL}_{\text{original}}$		$\text{TCL}_{\text{translated}}$		CTB (news, sampled)	
No.	POS tag	Percentage	POS tag	Percentage	POS tag	Percentage
1	VV	17.55%	VV	16.34%	NN	28.30%
2	NN	16.29%	NN	15.23%	PU	12.46%
3	PU	13.06%	PU	12.43%	VV	11.73%
4	AD	10.09%	AD	9.85%	-NONE-	7.02%
5	-NONE-	7.37%	PN	7.84%	NR	6.37%
6	PN	4.89%	-NONE-	7.30%	AD	4.90%
7	M	2.80%	P	3.05%	P	3.68%
8	AS	2.75%	DEG	2.89%	CD	3.17%
9	NR	2.69%	VA	2.56%	JJ	3.00%
10	CD	2.63%	M	2.41%	M	2.87%

Table 3: The 10 most frequent POS tags in $\text{TCL}_{\text{original}}$ and CTB news (sampled).

POS distribution We also had a closer look at the distribution of POS tags in $\text{TCL}_{\text{original}}$ and CTB news, to check for differences on the morpho-sytactic level. Table 3 presents the 10 most frequent POS tags and their proportions per corpus. A comparison shows interesting differences:

One clear difference concerns the proportion of nouns (NN) in the two corpora. In $\text{TCL}_{\text{original}}$, 16.29% of the words are nouns, in CTB, the proportion is almost twice as high, 28.30%. The prominence of NN in news texts is in line with previous empirical results (e.g., Zhang (2012)). A more detailed analysis shows that 经济 '*economy*', 企业 '*enterprise*', 公司 '*company*', 发展 '*development*' and 国 '*country*' are the five most frequent nouns in CTB, compared to 人 '*human*', 事 '*thing*', 话 '*speech*', 家 '*home*' and 父亲 '*father*' in $\text{TCL}_{\text{original}}$. They also show the trend that monosyllabic nouns are generally preferred in spoken and less formal genres, as previously observed by Zhang (2012). The lower proportion of nouns in $\text{TCL}_{\text{original}}$ corresponds to a higher frequency of verbs (VV), which indicates the "verbi-ness" of Chinese literature texts (Zhang, 2012). Directly related is the high frequency of adverbs (AD) since literary texts tend to use more adverbs for detailed and vivid description of actions.

Previous corpus studies (e.g., Zhang (2017)) have shown that personal pronouns, especially in third person, are associated with narrative discourse while first and second persons are linked to interactive discourse. Our analysis provides supporting evidence: We see a much higher frequency of pronouns (PN) in literary texts overall: 4.89% in $\text{TCL}_{\text{original}}$ vs. 0.87% in news texts (ranked 18th in CTB, not shown in Table 3). This is due to the fact that literature uses both narrative and interactive discourse while news mainly uses narrative discourse. While 他 '*he*' is the most frequent pronoun in both texts, the other frequent pronouns have different distributions: In the literary texts, we have first and second person pronouns (我 '*I*', 你 '*you*') along with the reflexive (自己 '*self*'). In contrast, for news, we find the neutral third person pronoun, two demonstratives, and finally the first person pronoun: 其 '*it*', 此 '*this*', 这 '*this*' and 我 '*I*'.

We also observe a wider range of POS tags used in $\text{TCL}_{\text{original}}$. Apart from the two new tags we created for suffixes (SFE and SFP), there are two tags that occur in $\text{TCL}_{\text{original}}$ but not in CTB: IJ (interjection) and ON (onomatopoeia), both typical for colloquial expressions. From the POS distribution of $\text{TCL}_{\text{translated}}$, in contrast, we see that translated Chinese overuses pronouns (PN), prepositions (P) and the marker 的(DEG), confirming the results from previous translation studies in Chinese (Xiao and Hu, 2015; Hu et al., 2018; Hu and Kübler, 2020).

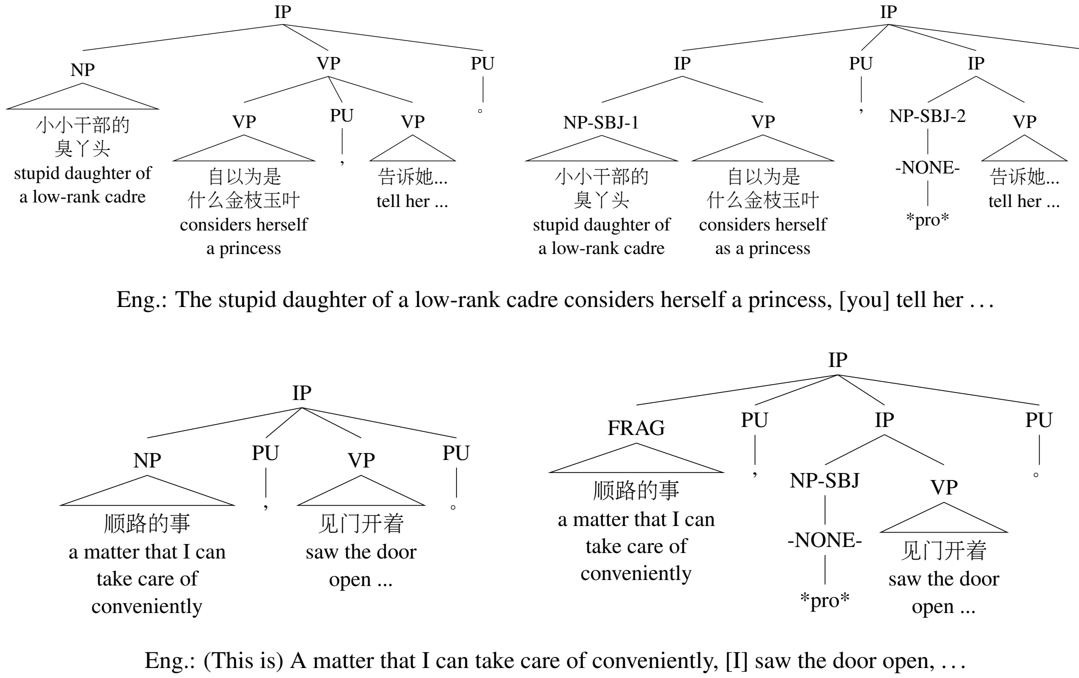
4.2 Comparing the News and Literary Genres

In this section, we provide a comparison of $\text{TCL}_{\text{original}}$ and CTB. We focus on two syntactic phenomena that are either less frequent in CTB or completely absent, (a) the pro-drop phenomena and (b) fragments and incomplete sentences. Both phenomena would cause lower parser performance in a domain adaptation scenario where the parser needs to parse literary texts but has been trained on CTB.

Pro-drop phenomena Chinese is known for its extensive use of pro-drop, especially in informal language. Xiao and Hu (2015) suggest that pro-drop is a significant indicator for specific genres. However, in order to test this hypothesis, they need syntactically annotated texts, or a parser that can produce

Structure	TCL$_{\text{original}}$	CTB (news, sampled)
pro-drop	614	343
pro-drop (per 1000 words)	30.0	11.9
subject pro-drop	602	334
object pro-drop	12	9

Table 4: Statistics of pro-drop phenomena in TCL$_{\text{original}}$ and CTB.



Eng.: The stupid daughter of a low-rank cadre considers herself a princess, [you] tell her . . .

Eng.: (This is) A matter that I can take care of conveniently, [I] saw the door open, . . .

Figure 2: Parsing errors involving pro-drop phenomena. Left: incorrect parser analysis. Right: gold tree in TCL. The dropped pronouns are in square brackets in the English translations.

empty categories. Since neither option was available, their hypothesis could not be tested empirically. However, the annotated TCL$_{\text{original}}$ and CTB do include empty categories, thus allowing us to investigate this hypothesis. We present the statistics of pro-drop in the two treebanks in Table 4. Since the treebanks contain a similar number of sentences, but CTB's sentences are considerably longer, we do not only report the absolute counts but also the counts normalized per 1 000 words. Pro-drop is much more common in literary texts: 614 occurrences in TCL$_{\text{original}}$ vs. 343 in CTB, or 30.0 normalized occurrences vs. 11.9.

Table 4 also shows that subject pro-drop is much more prevalent in both genres. Object pro-drop is rarely used and only occurs around 10 times in either treebank. However, the high percentage of subject pro-drop (602 cases in TCL0 vs. 334 cases in CTB) can provide challenges for the automatic parser and may cause systematic errors in the sentence structure. We show some parsing errors related to pro-drop in Figure 2.

In the first example, the gold tree is composed of two independent clauses: [NP$_1$ + VP$_1$] + [NP$_2$ (pro-drop) + VP$_2$], where the second clause has a dropped subject pronoun. However, since the parser cannot generate empty categories and would have to create an untypical IP with a single VP daughter, it failed to recognize the two clauses and instead grouped VP$_1$ and VP$_2$ into a coordinated VP with NP$_1$ acting as the shared subject. For the second example, we see that a dangling NP (a fragment) was incorrectly parsed as the subject whereas the correct analysis should insert a dropped pronoun in the subject position.

Fragments and incomplete phrases There are 30 fragments (FRAG) and incomplete phrases (INC) in TCL$_{\text{original}}$, which are often dangling PPs or NPs. In CTB, in contrast, the only fragments and incomplete phrases are found in the headers of the news articles, which we excluded from our sample. This means

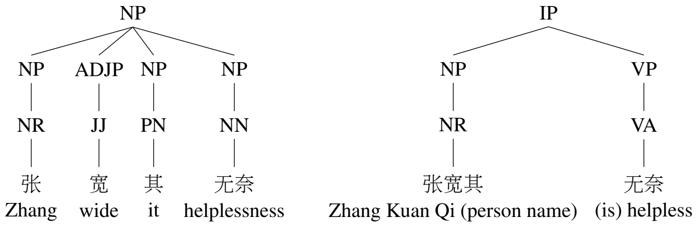


Figure 3: Parser error involving a person name. Left: parser mistake. Right: gold tree in TCL.

that in a formal genre such as news, all sentences are complete. Thus, if we train a parser on news texts, and then use it to parse literary texts, the parser may not be able to parse the incomplete structures in literary texts (see the second example in Figure 2).

This comparison only scratches the surface of the differences between the two genres. Considering the unique features of literary texts, our treebank will not only be a valuable resource for linguists interested in specific syntactic phenomena (such as pro-drop), but also be useful for building more reliable parsers for the literary domain.

4.3 Analysis of Parser Errors

Following the analysis above, we also looked at the actual parser errors. Since the trees in TCL are first automatically parsed using the default Chinese parser in Stanford CoreNLP (Manning et al., 2014) trained on news texts, we can analyze the errors in an out-of-domain parsing setting, after having manually corrected the trees. Here, we show two of the most common types of errors that the parser has made.

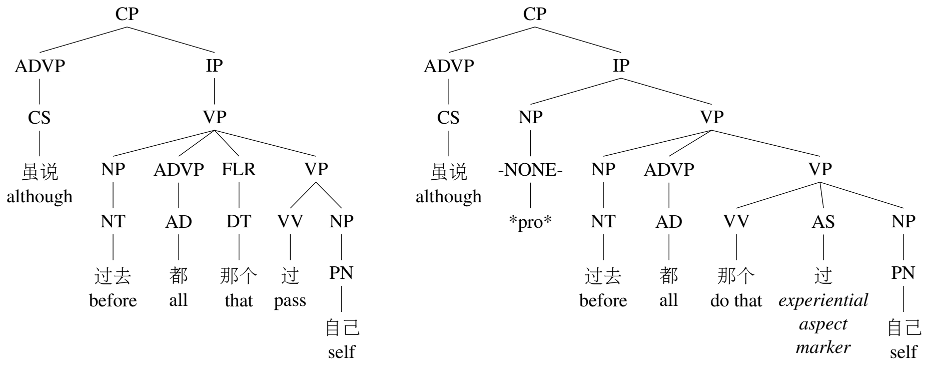
Named entities It is difficult for the parser to detect the named entities in literary texts, especially person names. We manually checked 20 named entities in 58 trees. Out of these 20 named entities, the parser only correctly recognized 8. All errors were due to over-segmentation. For example, the person name 苦根 (literally '*bitter root*') was segmented into two words and tagged `JJ NN`, rather than `NR` as a whole. The person name 张宽其 (literally '*Zhang wide he/it*') was segmented into three words and tagged `NR JJ PN`. I.e., only the surname 'Zhang' was recognized correctly (see Figure 3). There are also cases where a surname was labeled `VV` (e.g., 许, which can be a verb meaning '*allow*'). In general, the names in literary texts are more atypical and thus present a challenge to the parser.

Creative use of words In the literary treebank, there are cases where a word is used atypically, often as a part of speech different from its typical use. For instance, the word 臭 is an adjective meaning '*smelly/stinky*'. However, in one sentence, it is used as a verb meaning 'to trash (sth.)': 臭广告 'to trash the commercials'. The parser analyzed the phrase as an NP 'stinky commercials': (NP (JJ 臭) (NN 广告)). Another example is given in Figure 4. Here the demonstrative 那个 '*that*' is used as a verb to mean 'do so', which is a euphemism in spoken Chinese where the unspoken action it refers to needs to be reconstructed from the context. This type of flexibility and creative use in terms of parts of speech almost exclusively happens in literary texts. Such cases tend to lead to parse trees with very low accuracy since these wrong analyses require major changes to the rest of the tree.

4.4 Comparing Original and Translated Chinese Literary Texts

In this section, we have a closer look at the linguistic complexity of translated and original Chinese in literary texts.

As described above, one prominent hypothesis from translation studies states that translated texts are lexically and syntactically simpler than the texts originally written in the same language (Baker, 1993). This is often referred to as the *simplification hypothesis*, and is often assumed to be a universal feature of all translations. With our human-annotated, high-quality treebank, we can provide empirical evidence for/against the hypothesis in a language vastly different from Indo-European languages, for which the hypothesis has mostly been investigated (Ilisei and Inkpen, 2011; Volansky et al., 2013).



Eng.: ... although (they) all did that to (my)self before ...

Figure 4: Parser error involving creative use of words. Left: wrong parse from the parser. Right: gold tree in TCL.

	count		mean XP length in words			mean XP depth		
XP	orig	trans	orig	trans	p value	orig	trans	p value
CP	1277	1368	4.51	4.77	0.0623	5.60	5.80	0.0124
DNP	460	580	2.68	2.63	0.5738	3.45	3.44	0.8977
PP	676	704	3.49	**4.09**	0.0011	4.28	**4.70**	0.0011
NP	8830	8449	1.60	**1.72**	0.0001	2.64	**2.72**	0.0006
VP	9314	8090	4.14	4.31	0.0333	3.98	**4.25**	0.0
IP	3610	3285	10.83	10.95	0.6553	6.80	**7.27**	0.0
DP	344	350	1.60	1.51	0.1481	2.59	2.50	0.1214
ADVP	2216	2012	1.01	1.01	0.7233	2.01	2.01	0.2044
LCP	297	345	3.39	**4.10**	0.0014	4.11	**4.58**	0.0019
ADJP	384	279	1.03	**1.09**	0.0069	2.02	**2.06**	0.0025

Table 5: Statistics for XP structures in TCL$_{\text{original}}$ and TCL$_{\text{translated}}$. Greater values are in bold if $p < 0.01$, indicating more complexity, i.e., longer or deeper XP.

There are many ways to determine the complexity of sentences. Here we focus on two measures for linguistic complexity: the *length* and the *tree depth* of a linguistic unit. Specifically, we extract the treelets of the major phrases such as NPs, and VPs, and compare their complexity in literary texts of translated Chinese and those written in Chinese originally.

The comparisons of mean XP lengths and mean XP depths are shown in Table 5, along with the *p* values of the t-tests. For all the phrase types that show a significant difference between TCL$_{\text{original}}$ and TCL$_{\text{translated}}$, it is the *translated* texts that are more complex: PP, NP, LCP, and ADJP have longer mean lengths while PP, NP, VP, IP, LCP and ADJP have greater depths. This means that translated literary texts tend to have more complex (i.e., longer and deeper) linguistic units. These results contradict the simplification hypothesis and show that for many important phrases in Chinese, translations exhibit greater complexity.

While it is difficult to determine the exact reasons, for Chinese, these phrases are more complex in translations, there have been attempts. For example, Lin (2011) argues that the relative position of the modifier and the head inside a phrase has critical influence on human sentence processing. That is, for complex NPs with relative clauses, "the later the head noun is encountered, the greater temporary uncertainty exists in (human) parsing, and therefore the more difficult for (human) parsing" (Lin, 2011). Since Chinese is head-final in NPs and VPs (see the left two trees in Figure 5), long pre-head modifiers are generally dispreferred because they put too much processing pressure on the human processor. In contrast, English does not have such problems of "uncertainty" because the head precedes the modifier (see trees on the right in Figure 5), allowing the human processor to be able to comprehend and produce long RC and PP modifiers inside NPs and VPs respectively[7].

[7] We note that the issue of headedness has been extensively investigated by Liu (2010). Unfortunately, Liu (2010) only offers

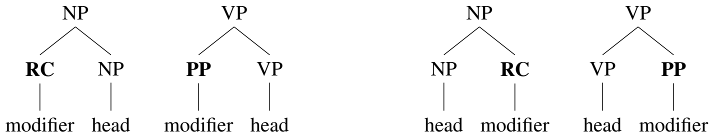


Figure 5: NP and VP structures in Chinese (left) and English (right)

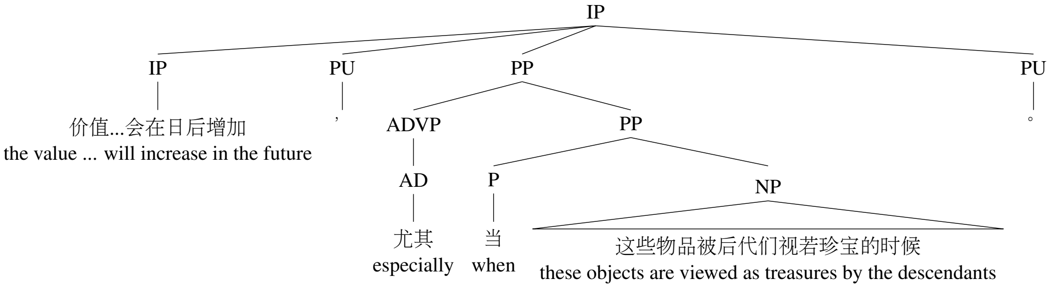

IP
IP
价值...会在日后增加
the value ... will increase in the future
PU
，
PP
ADVP
AD
尤其
especially
PP
P
当
when
NP
这些物品被后代们视若珍宝的时候
these objects are viewed as treasures by the descendants
PU
。

Figure 6: Example of a long dangling PP at the end of a sentence in $\text{TCL}_{\text{translated}}$, a feature of translated Chinese.

If the assumption that English has more complicated RCs and PPs is true (for which Lin (2011) provides preliminary corpus evidence), then the trend of longer PPs in English-to-Chinese translations that we find can be attributed to the *interference* effect; i.e., the syntax of the source language interferes with the production of the same structures in the translations (Toury, 1995).

We can further investigate this hypothesis in TCL. As an example, we find interference of word order from English PP structures in sentences in $\text{TCL}_{\text{translated}}$, as is illustrated by Figure 6. The sentence has a sentence-final PP, which is not the typical position for PPs in original Chinese. As shown in Figure 5, PPs usually precede the verbal head inside the VP in Chinese. The structure presented in Figure 6 is common in English as in *IP, especially when* .... Furthermore, PPs of the structure "当..." (*when* ...) have been identified as a characteristic of Europeanized Chinese (Wang, 1944; He, 2008). Here we see an example, which gives an indication of the reason for this phenomenon: Chinese texts translated from English inherit the linear ordering of constituents.

In sum, our preliminary analysis provides counter-evidence for the simplification hypothesis but some evidence for the interference hypothesis. Putting together the findings in Table 5 and the results from Table 2, which showed that translations have shorter sentences but longer words and slightly deeper trees, we conclude that the simplification hypothesis may be an over-simplification of the complex correlations between translations and originals, and we may need a combination of the simplification and interference hypotheses to explain the syntactic differences between translations and originals.

5 Conclusion and Future Work

In this paper, we have presented the Treebank for Chinese Literature (TCL), a novel Chinese treebank in the literary domain. The treebank contains texts from both translated and original Chinese and is thus suitable for translation and contrastive linguistic studies. We have compared our treebank with the news section of the Penn Chinese Treebank, and we have carried out a comparison of the translated and original portions of the new treebank. We have shown significant differences between the treebanks, from which we conclude that having such a treebank will be invaluable not only for linguistic analyses of literary texts but also for training parsers.

statistics for subject-verb or adjective-noun orders, but not for PPs and RCs. Thus we leave it for future work to follow this line of research and use dependency treebanks to look into the order and complexity of PPs and RCs in Chinese.

Acknowledgements

We thank our anonymous reviewers for helpful suggestions. This work is supported by the Indiana University – Renmin University Strategic Seed Fund and a summer incubator grant from the Institute for Digital Arts and Humanities at Indiana University. He Zhou is funded by China Scholarship Council.

References

Mona Baker. 1993. Corpus linguistics and translation studies: Implications and applications. In Mona Baker, Gill Francis, and Elena Tognini-Bonelli, editors, *Text and Technology: In Honour of John Sinclair*, pages 233–250. Amsterdam: John Benjamins.

Mona Baker. 1996. Corpus-based translation studies: The challenges that lie ahead. In Harold Somers, editor, *Terminology, LSP and Translation. Studies in Language Engineering in Honour of Juan C. Sager*, volume 18, pages 175–186. Amsterdam and Philadelphia: Benjamins.

Jong-Yul Cha. 1999. Semantics of Korean gapless relative clause constructions. *Studies in the Linguistic Sciences*.

Noam Chomsky. 1981. *Lectures on Government and Binding: The Pisa Lectures*. Walter de Gruyter.

Chenhui Chu, Toshiaki Nakazawa, Daisuke Kawahara, and Sadao Kurohashi. 2016. SCTB: A Chinese treebank in scientific domain. In *Proceedings of the 12th Workshop on Asian Language Resources (ALR12)*, pages 59–67, Osaka, Japan.

Michael A Covington and Joe D McFall. 2010. Cutting the Gordian knot: The moving-average type-token ratio (MATTR). *Journal of Quantitative Linguistics*, 17(2):94–100.

San Duanmu. 2012. Word-length preferences in Chinese: A corpus study. *Journal of East Asian Linguistics*, 21(1):89–114.

Richard Futrell, Kyle Mahowald, and Edward Gibson. 2015. Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.

Edward Gibson. 1998. Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68(1):1–76.

John Hawkins. 2004. *Efficiency and Complexity in Grammars*. Oxford University Press.

Yang He. 2008. *A Study of Grammatical Features in Europeanized Chinese*. Commercial Press. In Chinese.

Hai Hu and Sandra Kübler. 2020. Investigating translated Chinese and its variants using machine learning. *Natural Language Engineering (Special Issue on NLP for Similar Languages, Varieties and Dialects)*.

Hai Hu, Wen Li, and Sandra Kübler. 2018. Detecting syntactic features of translated Chinese. In *Proceedings of the Second Workshop on Stylistic Variation*, pages 20–28.

Iustina Ilisei and Diana Inkpen. 2011. Translationese traits in Romanian newspapers: A machine learning approach. *International Journal of Computational Linguistics and Applications*, 2(1-2):319–32.

Patrick Juola. 2008. Assessing linguistic complexity. *Language complexity: Typology, Contact, Change*, 89:107.

Kimmo Kettunen. 2014. Can type-token ratio be used to show morphological complexity of languages? *Journal of Quantitative Linguistics*, 21(3):223–245.

Sara Laviosa-Braithwaite. 1996. *The English Comparable Corpus (ECC): A Resource and a Methodology for the Empirical Study of Translation*. Ph.D. thesis, University of Manchester.

Roger Levy and Galen Andrew. 2006. Tregex and Tsurgeon: Tools for querying and manipulating tree data structures. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, pages 2231–2234, Genoa, Italy.

Zhenghua Li, Xue Peng, Min Zhang, Rui Wang, and Luo Si. 2019. Semi-supervised domain adaptation for dependency parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2386–2395.

Chien-Jer Charles Lin and Hai Hu. 2018. Syntactic complexity as a measure of linguistic authenticity in modern Chinese. In *26th Annual Conference of International Association of Chinese Linguistics and the 20th International Conference on Chinese Language and Culture*, Madison, WI.

Chien-Jer Charles Lin. 2011. Chinese and English relative clauses: Processing constraints and typological consequences. In *Proceedings of the 23rd North American Conference on Chinese Linguistics (NACCL-23)*, Eugene, OR.

Chien-Jer Charles Lin. 2017. Head-modifier relations in Europeanized Chinese: Linguistic authenticity and sentence processing. In *29th North American Conference on Chinese Linguistics (NACCL)*.

Chien-Jer Charles Lin. 2018. Subject prominence and processing filler-gap dependencies in prenominal relative clauses: The comprehension of possessive relative clauses and adjunct relative clauses in Mandarin Chinese. *Language*.

Haitao Liu. 2010. Dependency direction as a means of word-order typology: A method based on dependency treebanks. *Lingua*, 120(6):1567–1578.

Xiaofei Lu. 2010. Automatic analysis of syntactic complexity in second language writing. *International Journal of Corpus Linguistics*, 15(4):474–496.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics*, pages 55–60, Baltimore, MD.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Anthony McEnery and Zhonghua Xiao. 2004. The Lancaster Corpus of Mandarin Chinese: A corpus for monolingual and contrastive language study. In *LREC*, pages 1175–1178.

Chunyan Ning. 1993. *The Overt Syntax of Topicalization and Relativization in Chinese*. Ph.D. thesis, University of California, Irvine, CA.

William O'Grady. 1997. *Syntactic development*. University of Chicago Press.

Yina Patterson. 2020. *A study of nominal-clausal relations in Mandarin Chinese*. Ph.D. thesis, Indiana University Bloomington, IN.

Raphael Rubino, Ekaterina Lapshinova-Koltunski, and Josef Van Genabith. 2016. Information density and quality estimation features as translationese indicators for human translation classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 960–970.

Zuoyan Song and Hongyin Tao. 2009. A unified account of causal clause sequences in Mandarin Chinese and its implications. *Studies in Language*, 33(1):69–102.

Gideon Toury. 1995. *Descriptive Translation Studies and Beyond*. John Benjamins.

Vered Volansky, Noam Ordan, and Shuly Wintner. 2013. On the features of translationese. *Digital Scholarship in the Humanities*, 30(1):98–118.

Li Wang. 1944. *Theory of Chinese Grammar*. Commercial Press. In Chinese.

Fuyun Wu, Elsi Kaiser, and Elaine Andersen. 2010. Subject preference, head animacy and lexical cues: A corpus study of relative clauses in Chinese. In *Processing and Producing Head-Final Structures*, pages 173–193. Springer.

Richard Xiao and Xianyao Hu. 2015. *Corpus-Based Studies of Translational Chinese in English-Chinese Translation*. Springer.

Richard Xiao, Lianzhen He, and Ming Yue. 2010. In pursuit of the third code: Using the ZJU corpus of translational Chinese in translation studies. In *Using Corpora in Contrastive and Translation Studies*, pages 182–214. Cambridge Scholars Newcastle.

Richard Xiao. 2010. How different is translated Chinese from native Chinese?: A corpus-based study of translation universals. *International Journal of Corpus Linguistics*, 15(1):5–35.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer. 2005. The Penn Chinese TreeBank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 11(2):207–238.

Kevin Zhang, Qun Liu, Hao Zhang, and Xue-Qi Cheng. 2002. Automatic recognition of Chinese unknown words based on roles tagging. In *Proceedings of the first SIGHAN workshop on Chinese language processing-Volume 18*, pages 1–7. Association for Computational Linguistics.

Qing Zhang. 2008. Rhotacization and the “Beijing Smooth Operator": The social meaning of a linguistic variable. *Journal of Sociolinguistics*, 12(2):201–222.

Zheng-Sheng Zhang. 2012. A corpus study of variation in written Chinese. *Corpus Linguistics and Linguistic Theory*, 8(1):209–240.

Zheng-Sheng Zhang. 2017. *Dimensions of Variation in Written Chinese*. Routledge.

Meta-dating the PArsed Corpus of Tibetan (PACTib)

Marieke Meelen
University of Cambridge
Trinity Hall, Trinity Lane
Cambridge, UK, CB2 1TJ
mm986@cam.ac.uk

Élie Roux
Buddhist Digital Resource Center
1430 Massachusetts Ave., 5th floor
Cambridge, MA, USA 02138
roux.elie@gmail.com

Abstract

This paper presents PACTib, the PArsed Corpus of Tibetan. This new resource is unique in bringing together a large number of Tibetan texts (>5000) from the 11th century until the present day. The texts in this diachronic corpus are provided with metadata containing information on dates and patron-/authorship and linguistic annotation in the form of tokenisation, sentence segmentation, part-of-speech tags and syntactic phrase structure. With over 166 million tokens across 11 centuries and a variety of genres, PACTib will open up a wide range of research opportunities for historical and comparative linguistics and scholars in Tibetan Studies, which we illustrate with two short case studies.

1 Introduction

In recent years a large number of Tibetan manuscripts and books have been digitised and electronic texts (manually transcribed or corrected after OCR/HTR) have been made available online by the Old Tibetan Documents Online project (OTDO), the Buddhist Digital Resource Center (BDRC) and Esukhia. In addition to these historical Tibetan texts, modern written Tibetan etexts can now be found on the websites of the Timeless Treasuries and Tibetan e-books initiatives, which include a mixture of genres and styles from around 1980 until today. Finally, a collection of songs, folktales and other oral narratives in Present-Day Spoken Tibetan was transcribed and deposited on Zenodo as the 'University of Virginia' (UVA) corpus. Despite the recent growth in digitised text materials, from an NLP point Tibetan is still an under-resourced and under-researched language. Most Tibetan NLP research to date has been carried out in China. However, the resulting publications[1] rarely make data or code available, effectively making it impossible to test, verify or use the results in any way. Instead, for the development of PACTib, we build on recent work on segmenting and POS tagging Tibetan by Garrett et al. (2014), Meelen and Hill (2017) and Faggionato and Meelen (2019) (see Section 3). In Section 2 we discuss the composition of the corpus and a proposal to allow for distinguishing easily between prose and verse. Section 3 focuses on the linguistic annotation. In Section 4 we add a brief note on how the texts in the corpus are linked to the relevant metadata. Finally, Section 5 presents two short case studies to illustrate the use of this unique historical treebank of Tibetan.

2 Composition of the annotated corpus

PACTib consists of a variety of digitised materials that have been made available online. For the historical materials (up to the 21st century), we initially only selected texts that were originally composed in Tibetan. We furthermore included texts containing teachings of the Buddha and commentaries on those (so-called *eKangyur* and *eTengyur* collections respectively) that were generally translated from Indic languages into Tibetan. The first witnesses of these translated texts sometimes date back to the 10th century. The digitised versions available today, however, are based on an 18th century edition, in which they have been substantially revised and edited.

[1] e.g. Liu et al. (2011)

Because of these issues with uncertain dates of origin (including revisions) and the fact that they are not originally composed in Tibetan, both the *eKangyur* and *eTengyur* collections are kept separate from the rest of the PACTib subcorpora in the results of the diachronic case studies (see Section 5). For comparative purposes, however, and because these texts are intensely studied by Buddhist scholars, we do include them in PACTib as it could be of interest to Tibetan Studies scholars studying these canonical texts and to linguists looking at potential issues of translated versus native Tibetan texts.

Subcorpus	**"Genre"**	**Century**	**Tokens**
Old Tib. Annals & Chronicle	Historical	9-11th	22,978
Shenrab Miwo Bio. (*gZer mig*)	Biography (Bon)	11th	260,087
BDRC collection	Mixed (mainly Buddhist)	11th	2,197,474
"	Mixed (mainly Buddhist)	12th	4,639,041
"	Mixed (mainly Buddhist)	13th	1,188,324
"	Mixed (mainly Buddhist)	14th	10,504,224
"	Mixed (mainly Buddhist)	15th	11,135,952
"	Mixed (mainly Buddhist)	16th	9,881,222
"	Mixed (mainly Buddhist)	17th	9,805,019
"	Mixed (mainly Buddhist)	18th	10,817,489
"	Mixed (mainly Buddhist)	19th	1,787,061
Mipham works	Buddhist	19th	6,360,711
BDRC collection	Mixed (mainly Buddhist)	20th	2,465,143
14th Dalai Lama oral teachings	Buddhist	20th	706,274
Oral teachings by other lamas	Buddhist	20th	923,630
Mixed Modern Tibetan ebooks	Mixed (mainly Buddhist)	20th	156,880
Present-Day Tibetan blog posts	Mixed	21st	3,971,574
Present-Day Tibetan newspapers	Mixed	21st	3,185,631
UVA Present-Day Spoken corpus	Folktales, songs etc.	21st	990,722
eKangyur (Buddha Teachings)	Translated (Buddhist)	n/d	27,520,732
eTengyur (Commentaries)	Translated (Buddhist)	n/d	57,865,443
		Total	**166,385,611**

Table 1: Overview of PACTib Subcorpora

Table 1 gives an overview of all subcorpora that are currently included in the PACTib. The second column provisionally labelled "Genre" provides a rough indication of the type of texts contained in the subcorpora. The *Annals* and *Chronicle* are the earlier substantive amounts of Tibetan writing found in the Dunhuang caves in Gansu (Western China). These caves were sealed off in the 11th century and all manuscripts found in the caves are referred as 'Old Tibetan', the language spoken in the Yarlung Valley from where the Tibetan empire started its initial expansion. Most Old Tibetan texts are short inscriptions or more fragmentary parts of manuscripts and blockprints, but the *Annals* and *Chronicle* are longer and show more linguistic variety. Philologists generally consider the *Annals*, that record historical events in the 7-8th centuries, to be older than the more extensive *Chronicle*, although exact dates of origin are still a matter of ongoing debate (cf. Faggionato and Meelen (2019)). Tibetan texts written between the 11th and mid-20th centuries are generally referred to as 'Classical Tibetan', without further chronological subclassification. The two-volume biography of Shenrab Miwo (the founder of the Bon, i.e. a religion preceding Buddhism in Tibet) goes back to the 11th century, but is kept separate from the Old and Classical Tibetan texts since Bon texts generally contain non-Tibetan vocabulary as well (Snellgrove, 1967, 10). No systematic studies on differences in grammatical features have been done yet, although Snellgrove (1967, 8-9) makes some general remarks on the frequent mixing of genitive/agentive, locative/elative and allative/ablative case markers in Bon texts in particular. The selection of electronic texts from the BDRC contain a wide variety of Buddhist writings in a range of topics from philosophy to religious teachings and commentaries, prayers in verse, ritual texts, songs and sometimes even novels dating from the 11th to the 20th centuries. It is important to note that in the texts from the BDRC collection not all centuries are equally well-represented: the amount of data from the 11th, 13th, 19th and 20th centuries in particular is rather low compared to other centuries. For this reason, we have decided to supplement the data for those centuries with texts from other sources as best as we could. For the 11th century data remains scarce in general and Shenrab Miwo's Bon Biography may not be the best point of comparison with the rest of the texts that are overwhelmingly Buddhist.[2] For the 13th century, there are no other sources we could use and therefore this century remains significantly underrepresented. When doing diachronic research, it is important to bear this in mind, in particular when aberrant patterns are found in the results from the 13th century.

The data from the 19th century could be supplemented by the works of the prolific Buddhist philosopher Mipham Jamyang Namgyal Gyamtso (1846–1912), who wrote over 32 volumes on a variety of topics such as poetics, sculpture, medicine, tantra and logic, digitised by Adarsha. Finally, from the 20th century onwards (in particular after the 1980s), Buddhist oral teachings by the 14th Dalai Lama and other Tibetan lamas were recorded, transcribed and published as (electronic) books, a selection of which were added to PACTib as well. The Modern Spoken Tibetan had by that time already started to diverge significantly from the the Classical Literary language, but transcriptions of oral teachings are often edited to make them more similar to the written standard. In addition to oral teachings, at the end of the 20th century a number of Tibetan novels were published on a variety of topics. From the 21st century, we include collections of Tibetan blog posts and online newspaper articles, as well as the transcribed version of the Spoken Tibetan Corpus consisting of folktales, songs and other fieldwork done in the early 2000s in Tibet (Germano et al., 2017). All subcorpora differ significantly in size, ranging from $\sim 22k$ tokens in Old Tibetan to collections of millions of tokens from the BDRC as well as the translated Buddhist canon. For our present purposes, we aimed to annotate everything that was available in digital form and could be dated. In future work, when more studies of the materials become available, more careful selections can be made to create a more balanced annotated corpus suited for specific research questions.

2.1 Verse vs Prose

Because metadata for all of our subcorpora is extremely limited or non-existent, it is impossible to distinguish between verse and prose texts.[3] Automatic detection of verse is often done based on phonetic structure and rhyme (cf. Kesarwani (2018)). Since these features do not necessarily characterise Tibetan verse, we searched for other features. In Tibetan verse, the end of a line is always indicated by a ། *shad* marker. In prose texts, these *shad* markers can function as the equivalent of commas in enumerations, but are also used as semi-colons, colons or at the end of sentences. Since Tibetan verse lines are short (generally nine syllables at most), poetic texts have a much higher number of *shad* markers than prose equivalents of comparable length. This ratio of *shad* could thus be used as a very rough indicator of whether we are dealing with verse or prose.

For each text in our corpus we thus calculated the '*shad*-index' (the ratio of *shads* and overall tokens) and found a variety of 4.2-15.3: the higher the *shad*-index, the more likely it is that the text contains a large amount of verse. We verified the range with a known poetic text with verse lines of nine syllables (i.e. a long verse line in Tibetan, thus indicating a low boundary of the *shad*-index). This poetic text (Karu, 1974) has a *shad*-index of 10.41. It therefore seems reasonable to use a *shad*-index of 10.0 as a cut-off point when using the treebank for syntactic

[2] More Bon texts are available and some of those are already digitised: as soon as they become publicly available, we will incorporate them in PACTib.

[3] Note that in Table 1 we provisionally mark the topics or general text genres when they are commonly known; more specific information on verse vs prose, however, does not exist for most texts in our corpus.

queries in particular that are likely to be influenced by poetic styles. We briefly touch on this in Section 5.2.

A further indication that this cut-off point is on the right track is provided by the results of online news articles and blog posts from the 21st century, which we know are not focused on poetry. They have a *shad*-index of 5.08 and 4.78 respectively. Finally, it is important to note that the Old Tibetan *Annals* and *Chronicle* have a *shad*-index of 12.48 and 12.07 respectively, which would thus place them on the poetic side of the divide according to our calculations. However, the Old Tibetan language can be characterised by a number of features that distinguish it from Classical Tibetan. For instance, the texts are known to be formulaic in nature (Takeuchi, 2011) and in addition there are specific features of the punctuation that drive up the number of *shad*s per token (e.g. །།། ༈།།) resulting in a higher *shad*-index than we would normally expect for known prose texts in Classical and Present-Day Tibetan.[4]

3 Linguistic Annotation

The linguistic annotation of PACTib consists of tokenisation, sentence segmentation, part-of-speech tags and syntactic phrase structure labels building for a constituency treebank on recent work by Meelen and Hill (2017) and Faggionato and Meelen (2019). We optimised their methods after an error analysis and for the purposes of this paper, focused mainly on creating meaningful sentence segmentation.

3.1 Tokenisation and sentence segmentation

The Tibetan script has no markers to indicate word and sentence boundaries. Alongside morphosyntactic information, the linguistic annotation for PACTib therefore necessarily includes tokenisation and sentence segmentation as it can have consequences for any subsequent NLP tasks like part-of-speech (POS) tagging or Named Entity Recognition (NER) as well as for diachronic linguistic studies of the corpus. Tokenisation of PACTib was done using Meelen and Hill (2017)'s method combining memory-based syllable tagging and rule-based recombination of syllables into words. Clitics and case markers were considered separate tokens to reduce the overall number of different morphosyntactic tags. Sentence segmentation in the most recent version of the ACTib (Meelen et al., 2017) was purely done automatically, with utterance boundaries added after the Tibetan punctuation marker ། *shad* or །། *double shad*. The single *shad* in particular, however, is often more like the equivalent of a comma in English, as it is used in enumerations and subordinate clauses as well. When doing syntactic research on the clause or sentence level in particular, these forced sentence fragments are often too short to yield meaningful data. For this parsed version of ACTib, we therefore aimed to optimise the segmentation of sentences in a linguistically informed way through a series of rule-based replacements combining sentence fragments to fully grammatical sentences and splitting up combinations of what we would consider main clauses.

As a rigid head-final language, Tibetan exhibits object-verb (OV) order (DeLancey, 2003a) and verbs therefore always appear at the very end of the clause or sentence. Although Tibetan verbs exhibit no person-number agreement affixes, overt tense/aspect/mood (TAM) markers are attached to the right of verbal stems. In addition, Tibetan verbal forms can be nominalised with a variety of nominalisation suffixes. Nominalised verbs (with their arguments) do not function as the main verb of the sentence and were therefore, unlike their verbal 'conjugated' counterparts not used to identify sentence boundaries, as shown in example (1), where the nominalised verb *bkru* 'wash' (in bold) is not the matrix verb, but modifying the noun *dkaryol* 'cup' instead:

(1) དཀར་ཡོལ་བཀྲུ་ཡག་དེ་ཚོ་ག་པར་ཡོད་རེད་
[$_{NP}$ karyöl ***bkru*** *yag detsho] gapar yod red*
cup wash NOM these where EXIST.COP

[4] See Dotson and Helman-Ważny (2016, 82-85) for a detailed overview of punctuation and the use of *shad* and other markers in Early Tibetan documents. In future work we will refine our methods for the *shad*-calculation to be able to deal with specific orthographic features that lead to aberrant *shad* counts like these.

‘Where are the cups to be washed?’ (i.e. that need washing) (Tournadre and Dorje, 2003, 178)

The verb stem *bkru* in (1) would receive a verbal part-of-speech tag, but as it is followed by the nominaliser *yag*. In addition to conjugated verbs at the end of matrix clauses, Tibetan can exhibit sentence-final particles འོ་ དོ་ ཏོ་ ནོ་ བོ་ མོ་ སོ་ ངོ་ རོ་ *’o do to no bo mo so ngo ro* that indicate the end of the sentence. Finally, for the purpose of correcting parsed structures and a range of syntactic research it is more convenient to split coordinate main clauses into two separate sentences (Meelen and Willis, 2020). Therefore, conjugated verbs that are followed by the conjunction དང་ *dang*, tagged as an associative converb (`cv.ass`) are also followed by a sentence boundary. Sentence boundaries were therefore inserted according to the following set of sequential rules:

1. conjugated verbs + `cv.ass` + *shad*[5]
2. conjugated verbs + (final particle) + *shad*
3. final particles + (*shad*)

This ordered set of rules yields sentence boundaries that form a major improvement on the automatically added utterance boundaries after every *shad*, because *shad* is also used as the equivalent of a comma or semi-colons, resulting in each item of enumerations etc. (of which there are generally many in Buddhist texts) ending up as separate sentences that are not well-suited for morphosyntactic research.

3.2 POS tagging and Parsing

POS tagging was initially done with the memory-based method developed by Meelen and Hill (2017), but extended with a number of further rule-based corrections (e.g. erroneously tagged དང་ *dang* ‘and, (together) with’ > `case.ass` ‘associative case marker’, since in the context directly following nouns, it can never be anything else). Syntactic phrase-structural information was added using the rule-based regular expression parser developed by Faggionato and Meelen (2019) that combines Tibetan POS tags into phrases using an extended form of the NLTK’s regular expression chunkparser. This form of constituency parsing was chosen to facilitate comparative historical syntactic research on phrase structure in the UPenn historical treebank tradition. However, unlike the UPenn historical corpora, we deliberately chose not to add empty categories of any kind, to make PACTib more theory-neutral and because manual correction (which is always necessary as automatic insertion and annotation of empty categories is very prone to error) of such a large corpus is impossible. Another reason to create semi-hierarchical structures only and avoid empty categories for the present corpus is that the resulting bracketed structures can easily be converted to a dependency treebank format in combination with our highly detailed morphosyntactic tag set. Finally, attempts to develop automatically parsed dependency treebanks for Tibetan are already being undertaken by the researchers at SOAS, University of London, in the context of the ‘Lexicography in Motion’ project (Faggionato and Garrett, 2019) so a constituency-based treebank fills this gap in the literature. Example (3) shows the parsed result of a simple transitive clause like (2):

(2) ངས་ཁ་ལག་བཟས་པ་ཡིན།

[$_{NP}$ ngas] [$_{NP}$ kha lag] [$_{VP}$ bzas pa yin]
I.ERG food ate.PAST

‘I have eaten the food’ (Tournadre and Dorje, 2003, 165)

[5] For *shad* here, we mean any variety of Tibetan punctuation marker that conveys a function similar to the single *shad*. Depending on the text type or genre, variants like ༎ *gnyis shad* or “double” *shad*, ༔ *gter tsheg* or ༏ *tsheg shad* are used as the equivalent of commas, semi-colons, colons or full stops, just like regular *shad*.

(3) (S (NP ངས་/p.prop)
(NP ཁ་ལག་/n.mass)
(VP བཟས་པ་ཡིན་/v.past)
(PUNC །/punc))

Hierarchical structures, e.g. noun phrases within postpositional phrases are also automatically captured:

(4) བོད་ལ་གནམ་གྲུ་ཡོད་རེད།
[$_{PP}$ [$_{NP}$ *bod*] *la*] [$_{NP}$ *gnam gru*] [$_{VP}$ *yod red*]
Tibet in aeroplanes EXIST.COP
'There are aeroplanes in Tibet.' (Tournadre and Dorje, 2003, 121)

(5) (S (PP (NP བོད་/n.prop) ལ་/case.all))
(NP གནམ་གྲུ་/n.count
(VP ཡོད་རེད།/v.pres) (PUNC །/punc))

Since the rule-based parser and memory-based taggers were originally developed for Old and Classical Tibetan texts respectively, they are not always optimally suited for the Present-Day Spoken Tibetan language, which has evolved in a number of ways. Present-Day Literary Tibetan (or any form of the present-day written language) still strongly resembles Classical Tibetan (see also Section 5). Present-Day Spoken Tibetan nominalisation markers like ཡག་ *yag* or གར་ *gar* that do not exist in Classical Tibetan receive a special POS tag `nom`, which only exists in transcribed oral texts in Present-Day Tibetan. Since evidential, egophoric and epistemic verbal endings in Present-Day Tibetan have evolved from homophonous verbs and TAM markers in Classical Tibetan we chose to retain the conservative morphosyntactic annotation for those to facilitate research on diachronic changes in this aspect of the grammar.

Finally, it is important to note that Present-Day Tibetan contains a range of modern vocabulary items that are not found in the Old and Classical Tibetan training data. This goes for a number of modern verbs, e.g. ཕབ་ལེན་ *phab len* 'to download'. Most of these verbal forms, however, are based on combined verbs or light-verb constructions that already exist in Classical Tibetan and thus provide no real issue when conservative noun or verb tags are used, e.g. *phab len* 'download' < *phab* 'to bring down' + *len* 'to take', *kha par btang* 'to make a phone call (to)' < *kha par* 'phone' + *btang* 'to send'. Other new vocabulary, mainly from after the industrial and technological revolutions, mostly consists of nouns. Since count nouns (tagged `n.count`) are by far the most frequently-occurring tags, the memory-based tagger (and the neural tagger developed by Faggionato and Meelen (2019)) mainly assign this `n.count` tag to unknown words in the right context, these new vocabulary items pose no significant problem in Present-Day Spoken Tibetan texts.

4 Retrieving and Adding the Metadata

The PACTib is not only unique because of its size and scope, but also because it is the only Tibetan corpus with meaningful metadata linked to every sentence. As discussed in Section 2.1, there is in fact hardly any metadata available for any of the digitised texts that are available. Present-day oral teachings can of course be linked to known lamas and the connections can sometimes be made for well-studied historical texts, such as the works by Mipham in the 19th century and the *gZer mig*. The *Annals* and *Chronicle* have been the main focus of study for scholars of Old Tibetan as well, but they still disagree about the date of origin (ranging from the 9-11th century). Since our current main objective is to create an annotated diachronic corpus suitable for morphosyntactic research, our first aim is to attempt to link *all* the digitised materials in our subcorpora to meaningful dates of origin. Although the e-texts from the BDRC collection did not come with any readily available metadata, it is possible to get an idea about the date of origin because information about the author or a patron of a text (when this is available) is linked to the textIDs of e-texts in the BDRC database, which contains over 21,000 e-texts in total. For many of these authors and patrons, there is furthermore information about either their date of birth, date of death or both. Although this does not give us an exact date of origin for each text, it does provide us with a date range, which can be used to derive an approximate date of origin. We therefore extracted the date range of the life of an author or patron associated to the e-texts for which this was available (a total number of just over 5,000 e-texts, which is about a quarter of the total BDRC collection) from the BDRC's database, using SPARQL queries on:

- the date when the text was composed (rarely known)
- the birth/death date of the main author or patron

In this way, only texts where either the composition date or the birth/death date of the author was available, were added to our corpus. The BDRC has a rather large database, including 18,000 persons (authors, editors, important historical figures) and 40,000 books, historically focused on the Tibetan cultural area. It has recently moved to LOD (Linked Open Data) and is now able to aggregate results from datasets from partner organisations, such as the Sakya Research Centre or the Treasury of Lives, both of which also contain information about Tibetan authors. Finally, we were able to extract additional information regarding the topic of some of the texts. We could thus partially address the issues concerning the lack of metadata by extracting as much information as possible from a range of available resources, combining it in one place and making it accessible (see our annotated corpus and metadata files deposited on Zenodo through the link on our ACTib GitHub repository where all code and queries can be found as well). As the number of partner organisations willing to share their data with the BDRC grows, more and more data will be available on each author, thus allowing more and more e-texts to be added to future versions of this corpus. These dates were made an integral part of the SentenceIDs that were automatically added to all sentences in the treebank. Making dates/date ranges available through the SentenceIDs means the treebank can be queried in any way and results can be easily organised by date, without relying on any further resources. In the next Section we demonstrate this with two short case studies.

5 Tracing Diachronic Stability & Change

To illustrate potential uses of PACTib in this section we present two short case studies of diachronic morphosyntactic research questions that can be investigated with our treebank. Both case studies are based on observations by Tournadre and Dorje (2003) in their section on differences between Classical/Literary Tibetan and Present-Day Spoken Tibetan.

5.1 Oblique Case Markers

Our first case study is a change in the use of case marking particles. Old and Classical Tibetan exhibit a wide range of oblique case markers or postpositions, that vary in form due to their specific phonological contexts (DeLancey, 2003a). Each of these case markers are split off from the preceding words and tagged as `case.all` for 'allative/dative',[6] `case.loc` for 'locative', etc. As Tournadre and Dorje (2003) note, from the outset dative/allative *la*, locative *na* and terminative *du* (and their phonological variants *-r, ru, su, tu*) could function as the locative indicating a specific place (without movement), as shown in example (6):

(6) བོད་དུ་ བོད་ལ་ བོད་ན་
bod ***du;*** *bod* ***la;*** *bod* ***na***
Tibet TER Tibet ALL Tibet LOC
'in Tibet' (Tournadre and Dorje, 2003, 413)

[6] We follow Hill (2007) here calling Tibetan ལ་ *la* the allative marker although it has a range of other functions, e.g. dative, as well, which is why some refer to this as the dative marker (cf. Tournadre and Dorje (2003)).

In Present-Day Spoken Tibetan, in particular in Lhasa Tibetan, the dative/allative case marker *la* has taking over the functions of more and more other oblique case markers, leaving the locative, terminative, etc. as 'relict' forms such as adverbs and complex postpositions (see Section 5.2) only (DeLancey, 2003b, 275), as shown in example (7):

(7) སྒེར་དུ་ ལྷག་པར་དུ་
sger ***du****; lhag par* ***du***
private TER specially TER
'privately, personally; especially' (DeLancey, 2003b, 275)

If we query our treebank looking for postpositional phrases with allative/dative markers as opposed to other oblique cases, we can clearly see a rise of the use of allative *la* at the expense of terminative *du* in particular, as shown in Figure 1. The observations by Tibetan scholars such as DeLancey (2003b) and others that were based on the manual comparison of a small number of Tibetan texts from different time periods were definitely on the right track: in the modern spoken UVA subcorpus in particular we can see this change. The corpus also show a slight rise in dative/allative markers in 21st-century books, but this does not hold for online news articles and blogposts from the same period. This indicates that although the written language has evolved, it is still very far removed from the modern spoken language represented here by the Present-Day UVA subcorpus. Finally, it is actually quite remarkable how stable the distribution of oblique markers is across 11 centuries. From the 11th century onwards, terminative markers form the clear majority, which is not surprising as they have a very wide range of other functions besides the locative of place. Functions of elative, ablative and locative markers are much more restricted, which is clearly reflected in the data.

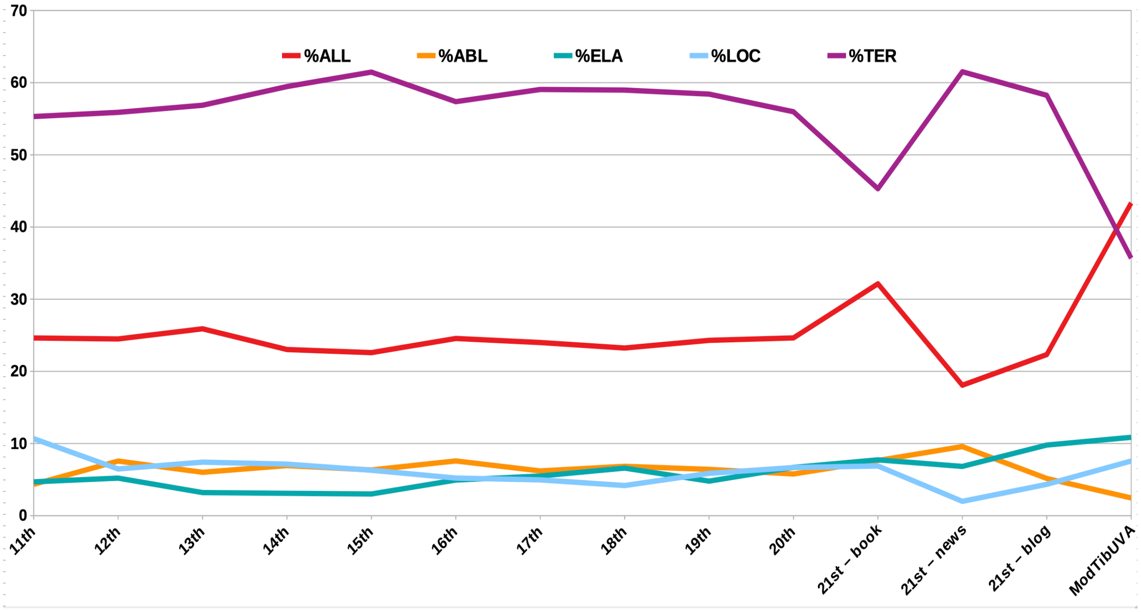


Figure 1: Ratio of oblique case markers from 11th-21st centuries.

5.2 Complex Postpositions

Our second case study concerns the syntax of complex postpositions that are tagged as 'relator nouns' (`n.rel`) in our treebank. These postpositions are originally lexical nouns that through a process of grammaticalisation have changed into functional items in combination with a noun phrase followed by a genitive case marker. The postposition itself can furthermore be followed by an oblique case marker (allative/dative, ablative, elative, locative or terminative). An example with the postposition *nang*, originally a noun meaning 'inside' but now part of the complex postposition preceded by a genitive and followed by a terminative case marker, is shown in (8):

(8) བོད་ཀྱི་ནང་དུ་ཡི་གེ་འབྲི་སྟངས་བཞི་ཡོད་པ་རེད་

bod	*(kyi)*	***nang***	*(du)*	*yige*	*'bri*	*stangs*	*bzhi*	*yod pa red*
Tibet	GEN	inside	TER	letter	writing	styles	four	EXIST.COP

'There are four styles of writing in Tibet.' (Tournadre and Dorje, 2003, 410)

Tournadre and Dorje (2003, 410) observe that in Classical/Literary Tibetan the preceding genitive case marker and the following oblique case markers are optional, whereas in Present-Day Spoken Tibetan these case markers cannot be omitted. Note that for poetic texts with verse lines forced into a predetermined number of syllables (often 5, 7 or 9), deliberate use *or* omission of the genitive marker to make up the right amount of syllables can be expected. Evidence for this particular construction in which the use of the genitive marker is believed to be optional in Old and Classical Tibetan could in theory thus go either way. A complete study of this goes beyond the scope of our present paper, but in future work, we will use the *shad*-index we established in Section 2.1 to test various hypotheses along these lines. If this is a gradual process of change, we would expect an increase in the use of genitive markers at the end of the Classical Tibetan period leading to a ratio of almost 100% genitive case markers in the 21st century, in particular in the spoken UVA subcorpus. Figure 2 shows the results of our complex postpositions with and without preceding genitive case markers. Percentages of the use of preceding genitives with postpositions are split up into different categories determined by the following oblique case markers (allative/dative, ablative, elative, locative and terminative) or '%gen-N', for the final option without final case marker.

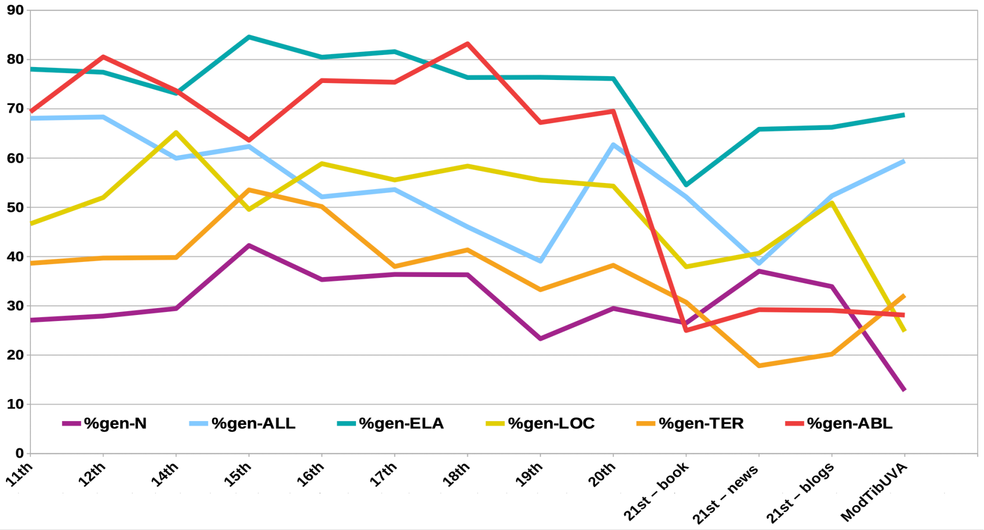


Figure 2: % of preceding genitive case markers in complex postpositions.

A initial interesting observation concerning this variable in our 166*m*-token corpus is again one of remarkable stability as we have seen with the oblique case markers above: the use of genitive case markers in this construction remains relatively stable between the 11th and 20th centuries.[7] Confirming Tournadre & Dorje's observation, the genitive marker was indeed often omitted in these constructions. However, we can observe a clear distinction between those complex postpositions followed by an ablative, allative or elative, where from early on genitive markers were use around 70-80% of the time, whereas numbers for complex postpositions with locatives and terminatives (or without following postpositions for that matter) are much lower.

As we expect changes to occur from the 20th century onwards, we again show the different sources in the 21st century in further detail by book, online news articles, blog posts and the transcription of the Present-Day Tibetan spoken UVA corpus. Interestingly, the use of genitives

[7] Note that the 13th century was omitted here because the lack of data from this period distorts the results of queries for lower-frequency constructions like these complex postpositions.

markers appears to decline at first compared to the previous centuries, but in the blog posts and in particular in the spoken UVA subcorpus, the use of genitive case markers is rising again.[8] The genitive is still not found 100% of the time, however, and numbers for the combinations with locative and ablative case markers are particularly low. The main reason for this is due to scarcity of data, which we will discuss in the final Section.

5.3 Discussion and Limitation of this first version of PACTib

In this final Section we discuss the results of our case studies in light of potential issues and limitations with this first version of PACTib. First, as we already noted, this is not a balanced corpus, but instead a collection of all the digitised Tibetan materials that were available to us. If a research query depends on a well-balanced corpus, it would be good to try and make a selection of selects from the PACTib to achieve this goal. As more and more Tibetan texts are being digitised these days, we expect the PACTib can soon be extended and gaps in time and genre can be filled. From the historical period, it would be good to have more material from the 11th and 13th centuries. From the modern, it would be good to collect more spoken material from a range of Present-Day Tibetan varieties, as the written and spoken language clearly differs and even 21st-century blog posts do not necessarily reflect the language as it is spoken today. As an example from our case study, the low number of locative and ablative case markers in the modern spoken subcorpus distorts the ratios. There are, for instance, only 32 cases of complex postpositions with ablatives overall; 9 of which have the genitive, a ratio of 28.12%. If we look at the numbers for the allative, elative and terminative on the other hand, we get hundreds of examples, therefore showing more robust patterns along the lines of what we expect. Scarcity of data is also an issue for the 21st-century book, which is with only 128,716 tokens, significantly shorter than the contemporary subcorpora containing news articles and blogposts (over 3 million tokens each).

Apart from data scarcity for certain constructions in specific subcorpora, this ablative case marker example illustrates a final limitation of this first version of the PACTib, namely, the possibilities of errors in the annotation. Tibetan ལྟ *lta* , for example, is often tagged as `n.rel` in the training data, because it can indeed have that function with the meaning 'like N' (following the noun N and potential genitive). However, *lta* has a range of other meanings as well and occurs in various phrases and expressions in which its special signification (derived from the verb 'to see') is no longer clearly discernible (Jäschke, 1987, s.v. ལྟ). In the spoken UVA subcorpus, for example, we find a number of examples with ད་ལྟ *da lta* where *lta* is still tagged as `n.rel`, even though in this sequence it actually means 'now...' and a preceding genitive would be impossible. Some results with the ablative case marker *las* in the spoken subcorpus are in fact cases of tagging errors: the sequence ད་ལྟ་ལས་བཟོ་ *da lta las bzo*, for example, was tagged as a complex postpositional phrase with ablative *las*, and counted as a result without a genitive marker. In fact, the segmenter here failed to segment the disyllabic noun ལས་བཟོ་ *las bzo* 'worker' properly and instead identified *las* as an ablative case marker that was part of a complex postpositional phrase. Because our corpus was automatically annotated with tools developed for Classical Tibetan, errors in annotation can always occur and affect the results. With frequent or less complex queries like our case study on oblique case markers in Section 5.1, this is not problematic as despite their ambiguous nature, the Precision and Recall of simple case markers following nouns is very high (Meelen and Hill (2017, 83-85) report an F-score of 0.99 for `case.term` and `case.all` and 0.98 for `case.abl`). With more complex or less frequent constructions more care should be taken. When segmentation has gone wrong in a sequence of syllables that are all highly ambiguous, as is the case of the above example in the context of multifunctional *da* and *lta* followed by the wrongly segmented single syllable *las*, this can affect the results. In this particular case this was exacerbated by the fact that there are relatively few

[8] This is not the case for the %gen-N context without oblique case markers, which is probably due to a change in the use of oblique case markers in general and specific postpositions with changed meaning in Spoken Tibetan, an investigation of which goes beyond the scope of the present paper.

instances of the ablative in Present-Day Spoken Tibetan to begin with. With corrections in a post-processing stage, as suggested by Meelen et al. (forthcoming), some of these issues can be addressed. However, for Present-Day Spoken Tibetan, it would ultimately be best to train a segmenter and tagger on contemporary spoken data, instead of relying on those developed for Classical Tibetan.

6 Conclusion & Future Work

In this short paper we present the first historical Tibetan treebank: the PArsed Corpus of Tibetan. PACTib is a linguistically annotated corpus of $> 166m$ tokens with dates ranging from the 11th to the 21st century. This corpus brings together all digitised historical materials that were available to us and for which at least a rough date of origin could be defined. Dates of origin derived from information about authors/patrons associated with the texts were extracted from the BDRC's database, which is partially fed with information through Linked Open Data protocols and agreements with partner organisations. This information was then systematically added not just to PACTib's metadata file, but also to all SentenceIDs so that results from corpus queries can be easily organised by date. The linguistic annotation consists of word and sentence segmentation, POS tags and constituency-based phrase structure. Our new method of sentence segmentation based on linguistic features means that parsing can be done efficiently and the resulting treebank facilitates any kind of syntactic research of longer and more complex sentences as well. In addition, the metadata for our treebank contains information about the number of tokens as well as the topic of the text (when available). Finally, we proposed the '*shad*-index', the ratio of the Tibetan punctuation marker *shad* and the total number of tokens, that indicates the likelihood of the text containing large amounts of verse. Because there is no information available on the genre of most of these texts, nor is there another way to automatically distinguish prose from poetry, which would be particularly useful for syntactic research. Our first attempt at calculating the *shad*-index of a text could be refined by critically examining more of our source materials, making sure that ornamental sequences of punctuation markers like *shad* such as those in the Old Tibetan texts are not skewing the results, but a first test with some known verse vs prose texts already yields promising results.

We finally presented two short case studies to illustrate how PACTib can be used for morphosyntactic research and to test the limits of the current version. With case studies on oblique case markers and complex postpositions we demonstrate PACTib can be a useful tool to test hypotheses on diachronic morphosyntactic developments. One interesting conclusion from both is that the Tibetan language has remained remarkably stable for over a thousand years in these two aspects of grammar. The main limitations are currently the lack of (balanced) data (especially for the 11th and 13th centuries, as well as the present-day spoken subcorpus) and certain issues with errors in the automatic annotation of ambiguous forms. We addressed some of the latter in forthcoming work (Meelen et al., forthcoming), but acknowledge that in order to really improve the annotation of Present-Day Spoken Tibetan, it would be best to train taggers on data from manually corrected Present-Day Spoken corpora once they become available.

Acknowledgements

The authors gratefully acknowledge Meelen's British Academy Postdoctoral Fellowship pf170063 grant and the ERC Advanced Grant 'Open Philology' (No 741884) for partially funding the research time of the authors that led to this paper and three anonymous reviewers for their helpful comments. In addition, the authors gratefully acknowledge the following individuals who made electronic versions of Tibetan texts available: Robbie Barnett (for the collection of Tibetan online newspaper articles), Gregory Forgues (for the collection of Mipham works), Edward Garrett (for the collection of Present-day Tibetan blog posts), Charles Ramble (for the *gZer mig* and a section of a 20th-century novel) and Ngawang Trinley (for support with the selection of a number of oral teachings and for his work on the *eKangyur* and *eTengyur*).

References

Scott DeLancey. 2003a. Classical Tibetan. In Graham Thurgood and Randy J. LaPolla, editors, *The Sino-Tibetan Languages*, volume 3, pages 255–269. London/New York: Routledge.

Scott DeLancey. 2003b. Lhasa Tibetan. In Graham Thurgood and Randy J. LaPolla, editors, *The Sino-Tibetan Languages*, volume 3, pages 270–288. London/New York: Routledge.

Brandon Dotson and Agnieszka Helman-Ważny. 2016. *Codicology, Paleography, and Orthography of Early Tibetan Documents: Methods and a Case Study.* Arbeitskreis für Tibetische und Buddhistische Studien Universität Wien.

Christian Faggionato and Edward Garrett. 2019. Constraint Grammars for Tibetan Language Processing. In *NEALT Proceedings Series 33:3*, pages 12–16.

Christian Faggionato and Marieke Meelen. 2019. Developing the Old Tibetan treebank. In Nikolova Temnikova Angelova, Mitkov, editor, *Proceedings of Recent Advances in Natural Language Processing*, pages 304–312. Varna: Incoma.

Edward Garrett, Nathan W Hill, and Abel Zadoks. 2014. A rule-based part-of-speech tagger for Classical Tibetan. *Himalayan Linguistics*, 13(2).

David Germano, Edward Garrett, and Stephen Weinberger. 2017. Uva tibetan spoken corpus, June.

Nathan W Hill. 2007. Aspirated and unaspirated voiceless consonants in Old Tibetan. *Languages and Linguistics*, 8(2):471–493.

Heinrich August Jäschke. 1987. *A Tibetan-English Dictionary: with special reference to the prevailing dialects, to which is added an English-Tibetan vocabulary.* London: Routledge & Kegan Paul Ltd.

Grub-dbang Karu. 1974. *The Autobiography of dKar-ru Grub-dbang bsTan-'dzin rin-chen. (dPal snya chen rig 'dzin mchog gi rnam sprul bāi'u ldong btsun grub pa'i dbang phyug bstan 'dzin rin chen rgyal mtshan bde chen snying po can gyi rnam par thar pa rmad 'byung yon tan yid bzhin nor bu' i gter).* Dolanji: Tibetan Bonpo Monastic Centre.

Vaibhav Kesarwani. 2018. *Automatic Poetry Classification Using Natural Language Processing.* Ph.D. thesis, Université d'Ottawa/University of Ottawa.

Huidan Liu, Minghua Nuo, Longlong Ma, Jian Wu, and Yeping He. 2011. Tibetan word segmentation as syllable tagging using conditional random field. In *Proceedings of the 25th Pacific Asia conference on language, information and computation*, pages 168–177.

Marieke Meelen and Nathan Hill. 2017. Segmenting and POS tagging Classical Tibetan using a memory-based tagger. *Himalayan Linguistics*, 16(2).

Marieke Meelen and David Willis. 2020. Towards a Historical Treebank of Middle and Early Modern Welsh, Part I: Workflow and POS Tagging. *Journal of Celtic Linguistics*, 22(1):125–154.

Marieke Meelen, Nathan W. Hill, and Christopher Handy. 2017. The Annotated Corpus of Classical Tibetan (ACTib), Part I - Segmented version, based on the BDRC digitised text collection, tagged with the Memory-Based Tagger from TiMBL, July.

Marieke Meelen, Élie Roux, and Nathan Hill. forthcoming. Optimisation of the largest annotated Tibetan corpus combining rule-based, memory-based deep-learning methods. *Transactions on Asian and Low-Resource Language Information Processing.*

David L Snellgrove. 1967. The nine ways of Bon: excerpts from " gZi-brjid". *London oriental series.*

Tsuguhito Takeuchi. 2011. Formation and transformation of old tibetan. In T. Takeuchi and N. Hayashi, editors, *Historical Development of the Tibetan Languages. Proceedings of the Workshop B of the 17th Himalayan Languages Symposium, Kobe, 6th–9th September*, pages 3–17.

Nicolas Tournadre and Sangda Dorje. 2003. *Manual of standard Tibetan: Language and civilization: Introduction to standard Tibetan (spoken and written) followed by an appendix on classical literary Tibetan.* Boston: Snow Lion Publications.

Fine-Grained Morpho-Syntactic Analysis for the Under-Resourced Language Chaghatay

Kenneth Steimel[†], **Akbar Amat**[‡], **Arienne Dwyer**[‡], **Sandra Kübler**[†]
[†] Indiana University
[‡] University of Kansas
ksteimel@iu.edu, akbar.amat@ku.edu, arienne@ku.edu, skuebler@indiana.edu

Abstract

We investigate part of speech (POS) tagging for Chaghatay, a historical language with a considerable amount of morphology but few available resources such as POS annotated corpora. In a situation where we have little training data but a large POS tagset, it is not obvious which method will be best to obtain an accurate POS tagger. We experiment with a conditional random field and a Recurrent Neural Network, augmenting the models with coarse grained POS tag information, and by utilizing additional data, either additional unannotated data used to train a language model or annotated data from a modern relative, Uyghur. Our results show that the combination of an RNN and pretraining with coarse grained POS tags reaches the highest accuracy of 76.17%.

1 Introduction

Part of Speech (POS) tagging has often been considered a solved problem. For languages with large annotated resources, POS tagging has reached accuracies in the high 90s: For English, the state of the art[1] has reached 97.85% (Akbik et al., 2018), and for French, 97.80% (Denis and Sagot, 2009). However, this is definitely not the case for many other languages with fewer resources, which often also exhibit considerable morphology. In such cases, the POS tags may go beyond pure word classes and may include a range of morphological information[2].

The current paper presents work on creating a POS tagger for Chaghatay (ISO-639 code: chg), using a manually created, annotated corpus[3]. However, in terms of modern POS annotated corpora, this linguistically annotated corpus is small, and the POS tagset is complex, including a considerable amount of morphological information. This is one of the most challenging settings for POS taggers. We investigate which of the approaches to POS tagging that are currently considered state of the art, using conditional random fields (CRF) or recurrent neural networks (RNN), can be successful in such a setting.

Given the complex tagset, we are also interested in determining whether a first analysis using a coarse grained POS tagset can be beneficial. Our assumption is that if we can determine the coarse word class reliably, this information can guide the full POS tagger by restricting the available choices for a given word in context. We finally investigate whether additional data, either additional unannotated Chaghatay data, or annotated data from modern Uyghur, one of the language's modern relatives[4], can be employed to improve accuracy.

Our main goal is creating a POS tagger that can, in the future, be integrated in the annotation process, to alleviate the burden on human annotators. This is especially important for languages such as Chaghatay, where highly specialized knowledge is required for every annotation step, including transcription of the manuscript.

[1] As documented at `https://aclweb.org/aclwiki/POS_Tagging_(State_of_the_art)`.

[2] For convenience, we will use the term POS tag even though the annotations are a combination of POS tags and morphological annotations.

[3] `https://uyghur.ittc.ku.edu/atmo.html`

[4] Another option would be to use data from Uzbek, the other modern relative, but we are not aware of any POS annotated corpus.

Our result show that for POS tagging without any modifications, the CRF reaches a higher accuracy than the RNN. However, adding coarse-grained POS information allows the RNN to surpass the CRF. Adding data from additional sources does not seem to be useful.

The remainder of the paper is structured as follows: Section 2 provides an overview of the language, the corpus, and the tagset. Section 3 explains our research questions in more detail. Section 4 describes the experimental setup. Section 5 discusses our findings, and section 6 concludes.

2 Chaghatay

2.1 Overview of Chaghatay

Chaghatay [trk:chg] was a koiné variety used by Central Asian Turks from the 14th to early 20th century as a prestige literary language from Bukhara to Kashghar. It amalgamated Eastern Turkic, Kwārazm Turkic, and an increasing amount of Persian. Today it is regarded as Classical Uzbek or Classical Uyghur. Since Chaghatay was the prestige form used primarily by élites as a literary and erudite lingua franca, it was fairly uniform, despite its use over a large territory. A late eastern variety of Chaghatay is under examination here.

2.2 The Chaghatay Corpus

The corpus consists of late 19th-early 20th century Chaghatay manuscripts collected in the Kashgar area of the southern Tarim Basin, in Eastern Turkestan (Xinjiang). Comprising the Jarring Collection at Lund University, they were collected by the philologist and diplomat Gunnar Jarring and predecessor Swedish missionaries in the southern Tarim. Metadata, and those manuscripts scanned by the Lund University Library, are available online[5]. Transcriptions (in the original Perso-Arabic script), transliterations (into a lossless Latin script), English translations, and POS annotation for selected manuscripts are also available online[6]. Medicine, healing, and networks were the topical foci.

2.3 The POS Tagset

The tagset is primarily of inflectional morphology, and is described by Dwyer (2018). The tagset is relatively large (about 500 items), given the rich morphology of Turkic languages, and given that the tagset was originally designed for manual part of speech annotation and Interlinear Glossing (ILG) of both Chaghatay and its descendant, Modern Uyghur.

The annotation scheme is primarily sentence-based (for linguists), and for text scholars, line and page breaks were later added. Each sentential unit has the following annotation tiers: a transcription of the original Perso-Arabic; a lossless Latin-script version of the former, and a segmented tier. Each segment was then annotated in two morphological tiers, an all-caps form/function "POS" tag from the tagset, and an interlinear glossing (ILG) tier, in which substantives are glossed in English, and grammatical categories are repeated from the POS tier with all-caps tags. A free translation of the sentence constitutes the next tier, and a final tier contains textual or linguistic comments.

We show an image of an original manuscript opening in Figure 1 and the sentence-based annotation tiers in Figure 2.

Textual scholars are likely to be interested in line and page breaks in the manuscript. Therefore, the annotation scheme also accounts for a line or page break within a sentence using the element <phr/> (phrase), as shown in Figure 4. In the unpunctuated example in Figure 3, we can see that the sentence `akr kmrshnynk / astyma bwlmaqy tn aġyr bwlmaqy āġzy tatlyġ bwlmaq` runs over two lines (here with a slash inserted to represent the line break).

[5] http://www.alvin-portal.org/alvin/resultList.jsf?dswid=2113

[6] https://uyghur.ittc.ku.edu/manuscripts/index.xhtml

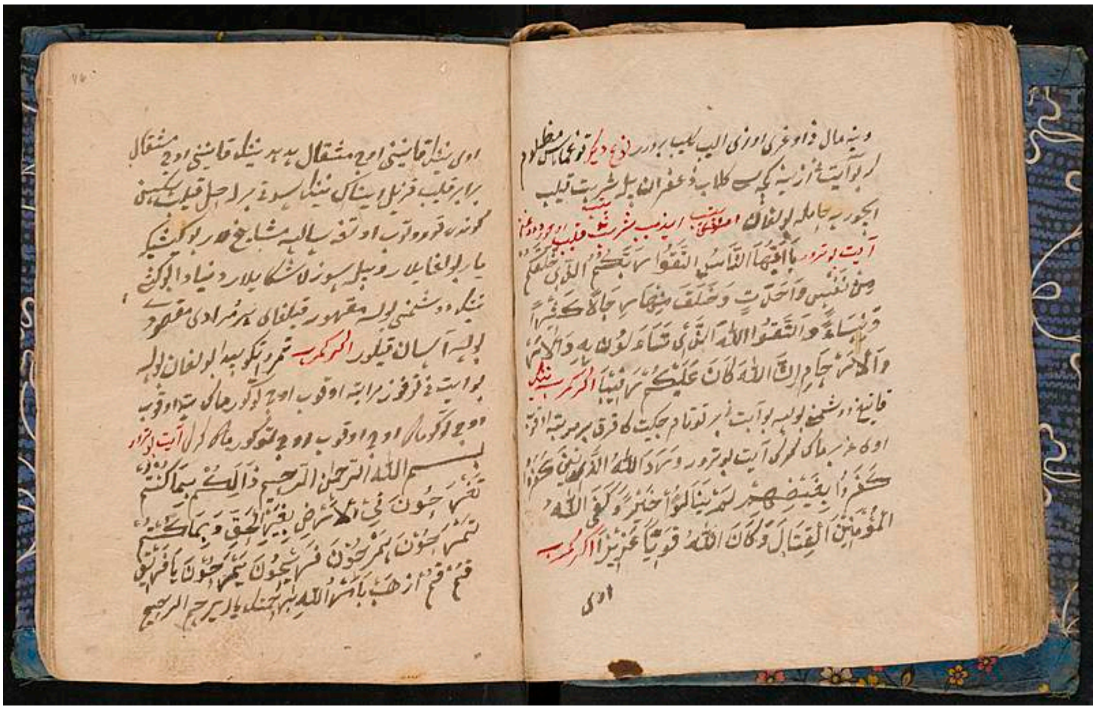
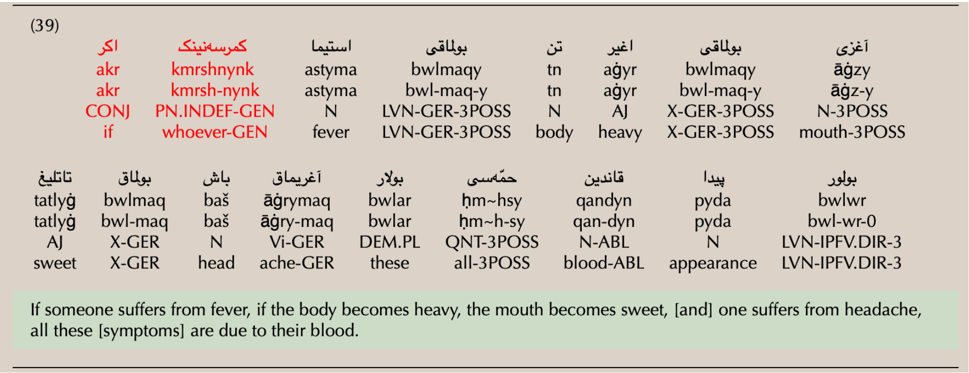


Figure 1: A page from the original manuscript.



(39)

اكر	كمرسەنینک	استیما	بولماقی	تن	اغیر	بولماقی	آغزی
akr	kmrshnynk	astyma	bwlmaqy	tn	aġyr	bwlmaqy	āġzy
akr	kmrsh-nynk	astyma	bwl-maq-y	tn	aġyr	bwl-maq-y	āġz-y
CONJ	PN.INDEF-GEN	N	LVN-GER-3POSS	N	AJ	X-GER-3POSS	N-3POSS
if	whoever-GEN	fever	LVN-GER-3POSS	body	heavy	X-GER-3POSS	mouth-3POSS

تاتلیغ	بولماق	باش	آغریماق	بولار	حمّەسی	قاندین	پیدا	بولور
tatlyġ	bwlmaq	baš	āġrymaq	bwlar	ḥm~hsy	qandyn	pyda	bwlwr
tatlyġ	bwl-maq	baš	āġry-maq	bwlar	ḥm~h-sy	qan-dyn	pyda	bwl-wr-0
AJ	X-GER	N	Vi-GER	DEM.PL	QNT-3POSS	N-ABL	N	LVN-IPFV.DIR-3
sweet	X-GER	head	ache-GER	these	all-3POSS	blood-ABL	appearance	LVN-IPFV.DIR-3

If someone suffers from fever, if the body becomes heavy, the mouth becomes sweet, [and] one suffers from headache, all these [symptoms] are due to their blood.

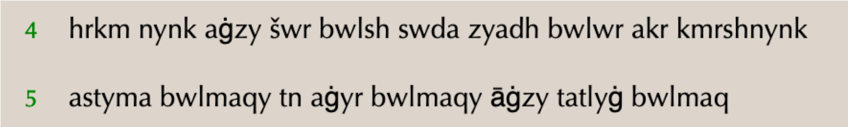
Figure 2: Example of the sentence-based tiers.

3 Research Questions

POS tagging for Chaghatay is one of the most challenging settings for POS tagging in general. The Chaghatay corpus is an ongoing project, thus little annotated data is available. Additionally, since Chaghatay is no longer spoken, there is only a limited amount of textual data available, restricting our ability to train a language model or use semi-supervised strategies. Finally, the POS tagset is large and includes a detailed analysis of morphological features. This leads us to consider the following questions:

3.1 Choice of Classifier

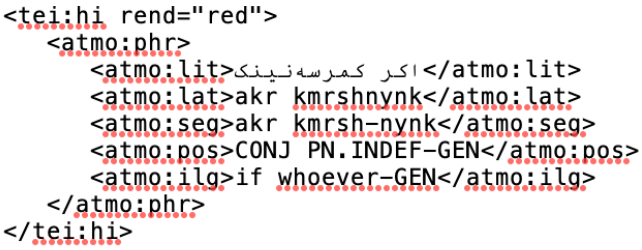
Given the combination of a small training set and a large POS tagset, the choice of classifier is not obvious. We decided to focus on two approaches that have been shown to be successful in POS tagging: Conditional Random Fields (CRF) (Gahbiche-Braham et al., 2012) and Recurrent Neural Networks (RNN) (Shao et al., 2017). RNNs are considered state of the art, but it is well known that they work best when they have access to large amounts of training data (Horsmann and Zesch, 2017).



4 hrkm nynk aġzy šwr bwlsh swda zyadh bwlwr akr kmrshnynk

5 astyma bwlmaqy tn aġyr bwlmaqy āġzy tatlyġ bwlmaq

Figure 3: Example of an unpunctuated example.



```
<tei:hi rend="red">
   <atmo:phr>
      <atmo:lit>اکر کمرسەنینک</atmo:lit>
      <atmo:lat>akr kmrshnynk</atmo:lat>
      <atmo:seg>akr kmrsh-nynk</atmo:seg>
      <atmo:pos>CONJ PN.INDEF-GEN</atmo:pos>
      <atmo:ilg>if whoever-GEN</atmo:ilg>
   </atmo:phr>
</tei:hi>
```

Figure 4: Example of a phrase element.

CRFs may be more amenable to small training data sets, but they may not scale up to a large label set (Horsmann and Zesch, 2017). Additionally, neural models can be pretrained on additional data from other domains and then optimized on our small training set.

3.2 Utilizing Coarse Grained POS Tagging as Preprocessing

The large tagset in this corpus is ideal for corpus-based analysis but provides challenges for statistical taggers. We investigate methods to overcome the challenges of a large tagset by using coarse POS tags as a first step in predicting the fine-grained tags. The most basic approach to POS tagging the data involves simply performing sequence tagging on the data using the fine grained POS tagset. However, given the combination of large tagset and small training set size, it is possible that the fined grained POS tagger could utilize information about the coarse grained category of a word. For example, knowing that a word is a noun will constrain the possible fine grained POS tags. Thus, we investigate for both types of models, CRFs and RNNs, whether utilizing coarse tags will improve the performance of the fine grained POS tagger.

Our approach involves separate coarse taggers, one for the CRF and one for the RNN, that are then leveraged by a more granular tagger. The CRF model uses coarse tags as additional features while the neural model uses transfer learning from a coarse tagger.

For the CRF model, we create a separate model trained on coarse tags. Then the coarse tagger is applied to a text, and the coarse tags predicted are included as features to the fine-grained CRF model. For this two-stage approach to be realistic, the coarse tagger needs to be trained using jackknifing (see Section 4.1.2 for details).

However, where the CRF tagger uses these coarse tags as features in its joint probability model, the neural approach does not use the coarse tag for making a decision about a specific word. Instead, it uses coarse grained tagging to provide a better initialization for the network. A standard method to obtain a better initialization would be to use off-the-shelf embeddings, which have been trained on a large data set of texts. For a low-resource language like Chaghatay, this is is not an option as such embeddings do not exist, and insufficient data is available to create traditional word embeddings like Word2Vec (Mikolov et al., 2013) and FastText (Bojanowski et al., 2017). Instead, we train a coarse-grained part of speech tagger and then transfer that model to fine-grained tagging by optimizing it on the more challenging task. This will provide a better weight initialization, similar to that provided by external embeddings.

3.3 Utilizing Training Data in Different Structure Formats

Since the corpus annotation process has evolved over time (see Section 2.2), we have manually annotated data in three different formats with regard to the marking of units: sentence segmented, phrase segmented, and line segmented. Since our task is POS tagging, we assign one label per word, but we also use information about sentence boundaries. Thus, the most relevant data are the sentence segmented documents. However, we only have very few of those, which raises the question whether we can use the other types of data to augment the training set. Does the additional data help guide the POS tagger, or is the missing information about sentence boundaries detrimental for the POS tagger? Does the difference in segmentation have any effect on POS tagging, or does the need for data override the need for sentence boundary information?

3.4 Pre-Training the RNN

For neural sequence tagging architectures, language model pre-training has been shown to be beneficial (Peters et al., 2018; Ortiz Suárez et al., 2020). In contrast to the CRF model, the transfer learning approach used for the neural network can be adapted from a variety of different initial tasks. We investigate whether this method can be used successfully in a setting where we have access to very little data in the target language. Since we do not have much additional data for Chaghatay, we experiment with two settings: 1) We use data from Chaghatay's modern relative, Uyghur, in the assumption that Uyghur is close enough to Chaghatay to provide a good starting point for the POS tagger. 2) We also experiment with pretraining the RNN using language modeling of Chaghatay as the task. For this pretraining step, we can leverage more training data since we can use all Chaghatay texts, including those that have been annotated for parts of speech yet.

4 Methodology

4.1 Data

As described in Section 2.2, the corpus has been annotated in different phases, with different underlying basic units of annotation, ranging from sentences, to lines in the manuscript and phrases. We use the term "structure" to refer to any of these units. The annotated Chaghatay data used for part of speech tagging contains 5 508 structures including 1 244 sentences, 1 348 lines, and 2 916 phrases. In total there are 30 666 words and 8 767 unique tokens.

Data from all available segmentation formats is combined during model training and evaluation for most models. However, in cases where the performance of models trained on different structure formats are compared, sentence data is used for the test set, and a combination of all structure formats are used for training data.

4.1.1 Data Splits for the Chaghatay Corpus

With the minimal amounts of data available, dedicated training and test datasets would provide a narrow view of the performance of our taggers. To make our results more robust, the available tagged data was randomly divided into 5 parts, and 5-fold cross validation was performed. These parts for cross validation are independent, non-stratified random samples across all three structure formats described in Section 2.3.

4.1.2 Creating Coarse Grained POS Tags

We extract coarse-grained POS tags by breaking a complex morphological tag into a series of smaller tags and looking up each of these smaller tags in a table to identify the appropriate coarse tag of the complex tag. First, a complex tag like VT-ANT.DIR-3=CZR for the word *swrdy0ky* would be split on markers for morpheme boundaries and clitic boundaries ('=' and '-') giving the following smaller tags: 'VT', 'ANT.DIR.3', and 'CZR'. These tags are then each looked up in a table of the correspondences between fine-grained parts of speech and coarse ones. This table was created during the creation of the annotation guidelines. A separate list of inflectional tags (like ANT.DIR.3) is also maintained. We then choose the coarse tag corresponding to the first fine-grained segment included in the table. If none of the tag segments is in the correspondence table, and all are listed in the list of inflectional tags, the coarse tag is INFL. If none of the segments are in the fine to coarse correspondence table, and not all the tag segments are listed as inflectional tags, an unknown coarse tag (XXXX) is assigned. This only affects 78 of the 27 782 words in the corpus.

4.1.3 Data for Language Modeling

As Chaghatay is a historical language, the standard method for collecting data for language models, i.e., scraping text from websites, is impossible. However, language model pretraining can still be useful for part of speech tagging in Chaghatay.

Because the overall annotation process in the Chaghatay corpus is quite time consuming, a considerable number of texts have been transliterated but not linguistically annotated yet. 9 518 structures have been transliterated but, as discussed in Section 4.1, roughly half this number of structures have annotations. These 9 518 structures are used to train the simple language models discussed in Section 4.2.4.

4.1.4 The Modern Uyghur Corpus

For the modern Uyghur data, we use the Uyghur Treebank (Eli et al., 2016), which is part of the Universal Dependencies (UD) project (McDonald et al., 2013). This treebank uses Universal POS tags, conforming to the UD annotation standards. The Universal POS tagset is a very coarse tagset consisting of 17 POS tags. The Uyghur Dependency treebank uses only 16 of those.

The Uyghur treebank is substantially larger than the Chaghatay data we are working with. In total, there are 3 459 sentences and 40 236 words. The data is divided into train, development, and test portions by the treebank creators. Only the training portion is used for pretraining our Chaghatay model with modern Uyghur data.

4.2 Models

We compare two different types of common sequence models: Conditional Random Fields (CRF) and Recurrent Neural Network (RNN) taggers. For both models, no special accommodations were made for out-of-vocabulary (OOV) tokens during training. Instead, feature representations derived from the characters in a word were leveraged.

4.2.1 Conditional Random Field Tagger

A Conditional Random Field model (CRF) (Lafferty et al., 2001) is similar to a Hidden Markov Model (the traditional approach for POS tagging), but with a flexible feature model and a discriminative probability model. CRFs have been shown to be well suited for sequence tagging tasks (Gahbiche-Braham et al., 2012; Sun, 2014). We use the CRF implementation by Okazaki (2007).

For our part of speech tagging task, we model the structure of the sentence as a sequence of words. For each word in an input sentence, we extract the following features: 1) The lowercase word, 2) the identity of the first 10 characters of the word as separate features, 3) the identity of the last 10 characters of the word as separate features, 4) the previous word in the sentence, and 5) the next word in the sentence.

All fine-grained CRF taggers were trained using the averaged perceptron training algorithm. For the coarse grained model, the LBFGS training algorithm was used as training time for the coarse tagset was quite short, and the LBFGS algorithm produced slightly better results.

4.2.2 Neural Tagger

Neural networks have been shown to work well for mono-lingual as well as multi-lingual POS tagging (Huang et al., 2015; Shao et al., 2017). Our neural tagger is a relatively simple Gated-Recurrent-Unit (GRU) network. This network consists of a word embeddings layer, a character embedding layer (the final state of a GRU over the characters in a word), a bidirectional GRU with varying numbers of layers, and a final softmax layer. For all experimental settings, the embeddings are updated during the training process; freezing of layers is not performed. In

Classifier	# hidden layers	Accuracy
CRF	n/a	**74.57**
RNN	1	*73.48*
	2	71.57
	3	68.02

Table 1: Accuracy of CRF and RNN using the fine-grained POS tagset.

the default setting, the character embeddings and word embeddings are randomly initialized. In the various transfer learning settings, the entire network, including the character and word embeddings are intialized using the weights learned from the previous task.

4.2.3 Using Coarse Grained POS Tagging

For creating the coarse CRF tagger, we apply jackknifing: We use 5-fold cross validation on the training data, such that a model is trained on 4 of the folds and predicts coarse tags on the remaining fold. This means we have the full dataset automatically POS tagged for coarse parts of speech.

The coarse RNN model is trained for 50 epochs while the fine-grained model is trained for an additional 75 epochs. To transfer from the coarse part of speech tagging model to the fine-grained model, the top softmax layer of the network is removed and replaced with a new softmax layer containing the relevant number of classes (where each potential tag is a class). The new softmax layer is randomly initialized.

4.2.4 Pretraining the RNN

The neural model allows for us to pretrain using a variety of different tasks. In this case, we pretrain the RNN model on language modeling and part of speech tagging for Uyghur. For these additional experiments, we only use the RNN architecture.

The architecture used for language modeling is very similar to the architecture of the part of speech tagger: A word embeddings layer is concatenated with a GRU-based character embeddings layer, this then passes through some number of GRU layers. The only difference is that the language model calculates a softmax over all possible words for both the forward and backward directions where the part of speech tagger had one softmax over the possible part of speech tags. As with pretraining on coarse part of speech tagging, the top part of the network is removed, and the final linear layers of the part of speech tagger are added on and randomly initialized.

For pretraining on modern Uyghur part of speech tagged data, the same design described in Section 4.2.3 is used.

5 Results

5.1 Choice of Classifier

We first look into the performance of the two classifiers, the CRF and the RNN, when performing fine-grained tagging. For the RNN, we experimented with 1, 2, and 3 hidden layers. These results are shown in Table 1, averaged over 5-fold cross-validation. We reach the best results of 74.57% using the CRF model. The best results for the RNN are 1 point lower, at 73.48%.

For the neural network models, the best results are reached with a single hidden layer in the main GRU. This indicates that larger networks are somewhat over-parameterized given the relatively small size of the training corpus. Reducing the number of hidden units for the single layer RNN shows a decrease in performance.

5.2 Utilizing Coarse Tags

For the second experiment we use coarse part of speech tags, either as a first tagging step for the CRF or as pretraining for the RNN (1 hidden layer). The results are shown in Table 2.

Classifier	Setting	Accuracy	In vocab. acc.	OOV acc.
CRF	fine-grained	74.57	83.15	42.91
	plus coarse-grained	74.68	83.05	43.93
RNN	fine-grained	73.48	81.75	41.78
	plus coarse-grained	**76.17**	**83.37**	**48.65**

Table 2: Comparison between the fine-grained only setting and the setting adding coarse-grained POS tagging, reporting overall accuracy, in-vocabulary accuracy and out-of-vocabulary accuracy.

CRF+coarse			CRF		
gold	tagger	#	gold	tagger	#
AJ	N	363	AJ	N	306
N	AJ	216	N	AJ	229
FOR	N	155	FOR	N	139
DEM	PN.DEM	97	Npr	N	93
Npr	N	85	DEM	PN.DEM	92
N	FOR	71	N	FOR	82
N	Npr	75	N	Npr	71
Npr	FOR	62	FOR	Npr	69
PN.DEM	DEM	65	PN.DEM	DEM	67
AJ	AV	44	AJ	AV	54

RNN+coarse			RNN		
gold	tagger	#	gold	tagger	#
AJ	N	222	AJ	N	290
N	AJ	215	N	AJ	276
Npr	N	128	FOR	N	145
DEM	PN.DEM	107	DEM	PN.DEM	115
N	Npr	92	N	FOR	99
FOR	N	87	Npr	N	96
PN.DEM	DEM	83	N+ACC	N-ACC	88
N+ACC	N-ACC	82	N	Npr	80
N	FOR	77	FOR	Npr	76
FOR	Npr	76	AV	AJ	72

Table 3: The 10 most frequent confusions per setting.

This setup provides a negligible increase in performance for the CRF model, from 74.57% to 74.68%. However, for the neural model, pretraining on coarse tagging is very beneficial. This setup increases accuracy from 73.48% to 76.17%, thus also improving over the CRF model.

When we evaluate in-vocabulary and out-of-vocabulary words separately, we see the same trend, the CRF sees a negligible decrease for known words, and it gains about 1% absolute for out-of-vocabulary words. In comparison, the RNN starts with a low accuracy on known words (81.57% versus 83.30 for the CRF) but gains 1.5% on known words and almost 7% on out-of-vocabulary words.

We also had a look at the confusion matrix for these four settings. The 10 most frequent confusions per setting with their frequencies are shown in Table 3. These confusions show that all models have the tendency to label words as noun (N, Npr, N-ACC): 5-6 of the 10 most frequent confusions involve this label. This is likely due to the prevalence of this tag: Over 23% of all words are tagged as nouns, and thus the model has strong tendencies to confuse other tags for nouns.

All models have difficulty distinguishing proper nouns (Npr) from conventional nouns (N),

Segmentation	Size training data (# structures)	Accuracy
Sentence data	800	67.69
Phrase data	800	52.14
Line data	800	53.34

Table 4: CRF model accuracy with training data from single structure types.

	Size training data	Accuracy	
Type of segmentation	(# structures)	CRF	RNN
Sentences	844	67.24	67.69
Sentences+phrases	844	65.31	64.26
Sentences+lines	844	65.73	61.81
Sentences+phrases	2 192	69.35	70.10
Sentences+lines	3 735	70.50	68.96
Sentences+phrases+lines	5 108	**71.90**	71.22

Table 5: Effectiveness of additional data sources.

likely due to the similar syntactic contexts both can be found in. Demonstrative pronouns (PN.DEM) and demonstratives (DEM) are also frequently mistagged by all four model types. The taggers seem to have difficulty identifying Arabic words. FOR, by definition, denotes an unanalyzed string surrounded by whitespace, usually a code-switch into an Arabic phrase.

When we compare the condition using coarse grained POS information to the base conditions directly performing fine grained POS tagging, we see that most of the error types are the same. It is interesting to see that the CRF+coarse model has higher numbers on the most frequent confusions than the base CRF, which implies that the CRF+coarse has fewer error categories overall while for the base CRF, the errors are distributed over more categories.

5.3 Different Structure Formats

Here, we investigate whether it is more important to have sentence segmented data, or if we need more data even if it is segmented differently. Given this question, we restrict our initial training set and the test set to sentence segmented data. We use 400 sentences from the 1 244 sentences as test data, the rest serves as initial training set.

We first look at the quality of the different segmentation styles. I.e., we carry out an experiment in which we train on the one dataset, using a single segmentation. We use the CRF model for this experiment since it showed a higher performance in the setting without coarse grained POS tags. Note that these results cannot be directly compared to the results in Table 2 since we do not use cross-validation here.

Table 4 shows that the quality of the line and phrase structures is not sufficient to substitute the sentence data: Both types of data result in a decrease of accuracy around 15% absolute even though the training set size is 2.5-6 times larger than for the sentence structures. This shows very clearly that the end of sentences marking is important for the POS tagging task.

Next, we look at the effectiveness of adding data from line and phrase structures to the sentence structures for annotating the sentence level test data. I.e., we start with the sentence segmented training set, and then add the line segmented and phrase segmented data. Note that this means that the size of the training set changes across settings. We also created balanced training sets so that the final training set size is the same as that of the sentence data, 844.

The results of adding training data in different segmentations are shown in Table 5. The results show that the additional data sources are beneficial to both the CRF and RNN models when added to the sentence segmented data. The accuracy increases from 67.24% to 71.90% for the CRF and from 67.69% to 71.22% for the RNN. When we compare the setting of the balanced training sets in the upper part of the table to the setting with all additional data, we see that the

	Fine-grained	Transfer from		
# hidden layers	tagger	coarse POS	Chaghatay lg. model	Uyghur
1	73.48	**76.17**	74.73	68.32
2	71.57	71.74	72.55	63.94
3	68.02	70.02	71.09	56.81

Table 6: Accuracy of different neural taggers

balanced cases lead to lower accuracies than using only sentences. The RNN is more susceptible to the difference in segmentation, reaching 64.26% for sentences plus phrases and 61.81% for sentences plus lines, as opposed to 67.69% for sentences only. Both architectures profit from the additional data, and the CRF reaches 71.90% when all available training data are combined. This shows that while the differences in segmentation do influence the POS taggers, having more training data outweighs these differences. We also see that initially, both architectures show a very similar performance, but the CRF model reaches a higher accuracy on the largest dataset, thus showing that it is better suited to using variable training data successfully.

5.4 Pretraining the RNN

In the final question, we investigate whether we can use pretraining via a Chaghatay language model or with modern Uyghur data to alleviate the data sparsity problem. The flexibility of neural networks allows us to use other tasks for network pretraining.

The results of this experiment are shown in Table 6. For ease of comparison, we repeat the results for the initial setting in the fine-grained setting, where we simply train on the fine-grained training set, and for optimized RNN initially trained on coarse-grained Chaghatay POS tags. The results show that the RNN profits from pretraining using a language modeling task with Chaghatay data. For the model with 1 hidden layer, accuracy increases from 73.48% to 74.73%. Pretraining on the Modern Uyghur data, in contrast, results in a considerable drop in performance by more than 5% absolute. This may illustrate the significant lexical, syntactic, and certainly orthographic differences between the two languages. Another reason for the decrease in accuracy may be the differences in the POS tagsets. This seems unlikely since pretraining on coarse-grained POS tags from the Chaghatay corpus has a beneficial effect, resulting in the highest results in our experiments.

Some spelling differences between Modern Uyghur and Chaghatay that likely lead to errors include the following: In Chaghatay, with the exception of (long) alef, vowels are unspecified or represented with consonants; in Modern Uyghur, each vowel has its own glyph. Further, Chaghatay typically lacks punctuation and (as seen above re: FRAG), scribes may insert a line break in the middle of a word or even morpheme. Finally, due to phonological changes such as vowel raising, modern Uyghur spelling often deviates from that of cognate forms in Chaghatay. This suggests that it may be more useful to use Uzbek data (which has less of these kinds of phonological changes) instead of Modern Standard Uyghur[7].

We have also experimented with different numbers of hidden layers, but the same pattern holds regardless of pretraining: The best results are reached with a single hidden layer.

6 Conclusion and Future Work

We have investigated POS tagging for Chaghatay, in a situation where we have a small training set but a large POS tagset. Our results show that without additional pretraining, the Conditional Random Fields tagger performs better than its neural counterpart. By using pretraining on coarse grained POS tags, the neural models are able to surpass the CRF model's performance. Using additional data from a language model or from modern Uyghur did not improve results.

[7] But we are not aware of any POS annotated corpus for Uzbek.

For the future, we are planning to investigate how well the different POS tagging architectures support manual postcorrection. I.e., does a higher overall accuracy also translate into higher manual annotation rates, or are the types of errors, which we have shown to differ between the architectures, are the determining factor? We will also start annotating the corpus for Universal Dependencies (McDonald et al., 2013).

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 1638–1649, Santa Fe, NM.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Pascal Denis and Benoit Sagot. 2009. Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art POS tagging with less human effort. In *Proceedings of the Pacific Asia Conference on Language, Information and Computation (PACLIC 23)*, Hong Kong, China.

Arienne M Dwyer. 2018. Morphological annotation in the ATMO project. Technical report, University of Kansas. `http://uyghur.ittc.ku.edu/manuals/MorphologicalAnnotation.xhtml`.

Marhaba Eli, Weinila Mushajiang, Tuergen Yibulayin, Kahaerjiang Abiderexiti, and Yan Liu. 2016. Universal dependencies for Uyghur. In *Proceedings of the Third International Workshop on Worldwide Language Service Infrastructure and Second Workshop on Open Infrastructures and Analysis Frameworks for Human Language Technologies (WLSI/OIAF4HLT2016)*, pages 44–50, Osaka, Japan.

Souhir Gahbiche-Braham, Hélène Bonneau-Maynard, Thomas Lavergne, and François Yvon. 2012. Joint segmentation and POS tagging for Arabic using a CRF-based classifier. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC)*, pages 2107–2113, Istanbul, Turkey.

Tobias Horsmann and Torsten Zesch. 2017. Do LSTMs really work so well for PoS tagging? – A replication study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 727–736, Copenhagen, Denmark.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. Technical Report arXiv:1508.01991 arXiv:1508.01991, arXiv.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*, pages 282–289, San Francisco, CA.

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations (ICLR)*, Scottsdale, AZ.

Naoaki Okazaki. 2007. CRFsuite: a fast implementation of Conditional Random Fields (CRFs). `http://www.chokkan.org/software/crfsuite/`.

Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1703–1714, Online.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2227–2237, New Orleans, LA.

Yan Shao, Christian Hardmeier, Jörg Tiedemann, and Joakim Nivre. 2017. Character-based joint segmentation and POS tagging for Chinese using bidirectional RNN-CRF. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (IJCNLP)*, pages 173–183, Taipei, Taiwan.

Xu Sun. 2014. Structure regularization for structured prediction. In *Proceedings of Neural Information Processing Systems (NIPS)*, pages 2402–2410, Montreal, Canada.

Automatic Extraction of Tree-Wrapping Grammars for Multiple Languages

Tatiana Bladier, Laura Kallmeyer, Rainer Osswald, Jakub Waszczuk
Heinrich Heine University Düsseldorf, Germany
{bladier,kallmeyer,osswald,jakub.waszczuk}@phil.hhu.de

Abstract

We present an algorithm for extracting Tree-Wrapping Grammars (TWGs) for multiple languages from constituency treebanks. The TWG formalism, which is inspired by Tree Adjoining Grammar (TAG), has been developed for the formalization of Role and Reference Grammar (RRG). We describe the extraction of TWGs for English, German, French and Russian from the multilingual RRG corpus RRGparbank. A special focus is given to how non-local dependencies are treated by the extraction algorithm. In TWGs, non-local dependencies are considered as arising from local dependencies in elementary trees by the operation of 'wrapping substitution'. The extracted grammars are validated by using them in a subsequent parsing step.

1 Background: Tree-Wrapping Grammars

The Tree Wrapping Grammar (TWG) formalism (Kallmeyer et al., 2013; Kallmeyer, 2016; Osswald and Kallmeyer, 2018) is a tree-rewriting formalism much in the spirit of Tree Adjoining Grammar (TAG) (Joshi and Schabes, 1997) that has been developed for the formalization of Role and Reference Grammar (RRG) (Van Valin, 2005; Van Valin, 2010), a theory of grammar with a strong emphasis on typological concerns. A TWG consists of a finite set of elementary trees which can be combined by the following three operations: a) *(simple) substitution* (replacing a leaf by a new tree), b) *sister adjunction* (adding a new tree as a subtree to an internal node), and c) *wrapping substitution* (splitting the new tree at a d(ominance)-edge, filling a substitution node with the lower part and adding the upper part to the root of the target tree). As in (lexicalized) TAG, the elementary trees of a TWG are assumed to encode the argument projection of their lexical anchors. Figure 1 shows an application of wrapping substitution for generating the German sentence in (1) (the dashed line indicates a d-edge).[1]

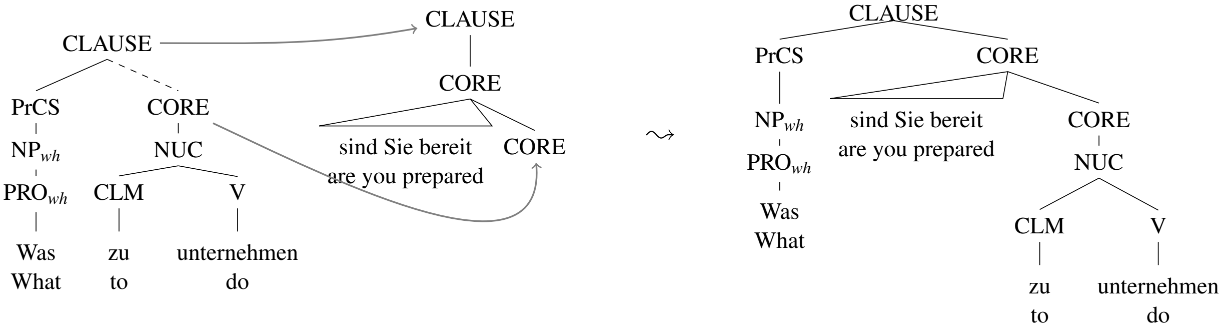


Figure 1: Wrapping substitution for the construction in (1).

(1) *Was sind Sie bereit zu unternehmen ?*
What are you prepared to do ?

[1] Abbreviations: NUC = Nucleus, PrCS = Precore slot. All examples are taken from George Orwell's novel '1984' or its published translations.

The example illustrates how non-local dependencies, here a wh-extraction across a control construction, can be generated by wrapping substitution from local dependencies in elementary trees.

TWG are more powerful than TAG (Kallmeyer, 2016). The reason is that a) TWG allows for more than one wrapping substitution stretching across specific nodes in the derived tree and b) the two target nodes of a wrapping substitution (the substitution node and the root node) need not come from the same elementary tree, which makes wrapping non-local compared to adjunction in TAG. The latter property is in particular important for modeling extraposed relative clauses (see example (3) for a deeper embedded antecedent NP, which requires a non-local wrapping substitution).

In this paper, we adopt a slightly generalized version of wrapping substitution which allows the upper part of the split tree, provided that the upper node of the d-edge is the root, to attach at an inner node of the target tree. For instance, in Figure 1 an additional SENTENCE node above the CLAUSE node in the tree of *bereit* (‘prepared’) would be possible. A further example for this generalized wrapping will be discussed in Figure 2 below.

By using TWG as a formalization of RRG and applying it to multilingual RRG treebanks, we aim at extracting corpus-based RRG grammars for different languages, thereby obtaining in particular a cross-linguistically valid “core” RRG grammar and, furthermore, providing a cross-lingual proof of concept for TWG in general with respect to its ability to model non-local dependencies. The work presented in this paper is a first step towards these goals.

2 Non-local dependencies in RRGparbank

RRGparbank is part of an ongoing project to create annotated treebanks for RRG (Bladier et al., 2018; Bladier et al., 2019).[2] RRGparbank provides parallel RRG treebanks for multiple languages. At present, RRGparbank contains George Orwell’s novel ‘1984’ and its translations in several languages.[3]

RRGparbank provides annotations of non-local dependencies (NLDs) including those given by long-distance wh-extraction (2a), relativization (2b), topicalization (2c), and extraposed relative clauses (2d).

(2) a. *What do you think you remember?*
b. *[. . .] two great problems, which the Party is concerned to solve.*
c. *By such methods it was found possible to bring about an enormous diminution of vocabulary.*
d. *Nothing has happened that you did not foresee.*

In the present context, ‘non-local’ means that the dependency is not represented within a single elementary tree. We refer to non-local wh-extraction, relativization and topicalization as long-distance dependencies (LDDs).

In RRGparbank, LDDs are annotated in the following way: The fronted phrase node carries a feature PRED-ID whose (numerical) value coincides with the value of the feature NUC-ID of the NUCLEUS the fronted phrase semantically belongs to. For instance, in the annotation of sentence (1), the NP$_{wh}$ node in the tree shown on the right of Figure 1 is marked by [PRED-ID 1] while the NUC node above *unternehmen* is marked by [NUC-ID 1]. See Figure 3 for another example of the annotation convention. In the case of extraposed relative clauses, the relative pronoun and the NP modified by the relative clause both carry the feature REF with identical values (cf. Figure 4).

3 Deriving non-local dependencies by wrapping substitution

Similar to TAG, (simple) substitution in TWG represents the mode of tree composition for expanding argument nodes by the syntactic representations of specific argument realizations, while sister adjunction is mainly used for adding peripheral structures (i.e., modifiers) to syntactic representations. Wrapping substitution, on the other hand, is used for linguistic phenomena in which an argument is displaced from its canonical position and which cannot be handled by simple substitution or sister adjunction

[2] https://rrgparbank.phil.hhu.de/

[3] The data are partly taken from the MULTEXT-East resource (Erjavec, 2012).

(Kallmeyer et al., 2013; Osswald and Kallmeyer, 2018). This holds in particular for the cases of non-local dependencies (NLDs) listed in Section 2. The TWG derivation of LDDs by means of wrapping substitution follows basically the pattern illustrated by the example in Figure 1.

Extraposed relative clauses (ERCs), as in (2d), represent a different type of NLD, namely the extraction of a modifier (the relative clause), typically to a position to the right of the CORE, which leads to a non-local coreference link between the relative clause and its antecedent NP. Example (2d) can be analyzed using wrapping as shown in Figure 2. The extraposed relative clause is associated with a tree that contributes a periphery CLAUSE below a CLAUSE node while requiring that an NP node (which serves to locate the antecedent NP) is substituted into an NP node somewhere below the CLAUSE, modeled by a d-edge between the upper CLAUSE node and a single NP node without daughters. This NP is a substitution node that gets filled with the actual antecedent NP tree. Put differently, the antecedent NP merges with this single NP node, which establishes the link to its modifying relative clause.

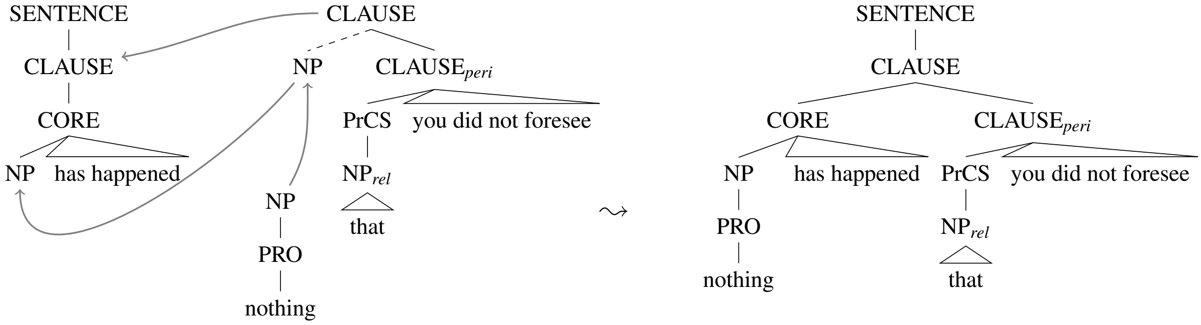


Figure 2: Wrapping substitution for the extraposed relative clause from (2d)

In RRGparbank, we encountered cases where the antecedent NP is further embedded and also cases with more than one relative clause modifying the same antecedent. (3) is an example where we have both: The antecedent NP *Menschen* (‘people’) is embedded in the direct object NP, and we have two extraposed relative clauses, both modifying the same antecedent.

(3) *Unzählige Male hatte sie [...] [die Hinrichtung von Menschen]$_{NP}$ gefordert , [deren Namen sie nie*
Numerous times had she [...] the execution of people demanded , whose names she never
zu vor gehört hatte] [und an deren angebliche Verbrechen sie nicht im entferntesten glaubte] .
before heard had and in whose alleged crimes she not in the least believed .
‘On numerous occasions, she had [...] demanded the execution of people whose names she had never heard before and in whose alleged crimes she did not even remotely believe.’

Another interesting phenomenon is illustrated by the Russian example in (4), which shows both wh-extraction (*čto*) and topicalization (*ja*).

(4) *Ja vot čto xoču skazat’.*
I here what want to.say
‘What I’m trying to say is this.’

The current annotation in RRGparbank presumes a scrambling analysis of this topicalization, which gives rise to an RRG tree with crossing branches not generated by sister adjunction. This case is not yet covered by the extraction algorithm presented in Section 4.

4 TWG Extraction

To extract TWGs from treebanks, we adapt the top-down algorithm from (Xia, 1999) for TAG. While substituting and sister-adjoining trees can be extracted following the procedure described in (Xia, 1999), we developed a new algorithm to extract d-edge trees which we describe in more detail below.[4] Since TWGs do not allow for trees to have crossing branches, but the RRG trees often contain them, such edges need to

[4] Additional information on the extraction algorithm can be found in (Bladier et al., 2020).

be removed following a rule-based algorithm for re-attaching certain subtrees in the original tree in a preprocessing step. The process of decrossing tree branches concerns only local re-attaching of peripheral constituents and operator projections and can be reverted applying a rule-based back-transformation algorithm after the parsing step. We extract lexically unanchored elementary tree templates (i.e. *supertags*) for the TWGs. The lexical anchoring happens in the subsequent parsing step.

1. **Decross tree branches.** First, for local discontinuous constituents (for instance NUCs consisting of a verb and a particle in German), we split the constituent into two components (e.g., NUC1 and NUC2), both attached to the mother of the original discontinuous node.
   Second, if a tree τ still has crossing branches, the tree is traversed top-down from left to right and among its subtrees those trees are identified whose root labels contain one of the following strings: OP-, -PERI, -TNS, CDP, or VOC. For each such subtree γ in question with r being its root, we choose the highest node v below the next left[5] sibling of r such that the rightmost leaf dominated by v immediately precedes the leftmost leaf dominated by r. If r and v are not yet siblings, γ is reattached to the parent of v. If the subtree in question has no left siblings, it is reattached to the right in a corresponding way. After this step, it should be checked if the tree τ still contains crossing branches. If yes, the process of decrossing branches is continued by applying the steps above to the next subtree in question.

2. **Extract NLDs.** Then we traverse each tree τ in a top-down left-to-right fashion and check for each subtree of τ whether it contains the following special markings for NLDs in its root label: PREDID=, NUCID= or REF=. The indexes identify the parts of the NLD which belong together. In case of an LDD, the parts of the minimal subtree which contain both parts of the LDD are extracted within a single tree with a *d-edge* (see the multicomponent NUC and CORE in Figure 3). The substitution site and the mother node are added to the remaining subtree in order to mark the nodes on which the wrapping substitution takes place (see Figure 3). A similar process is applied to extract ERCs.

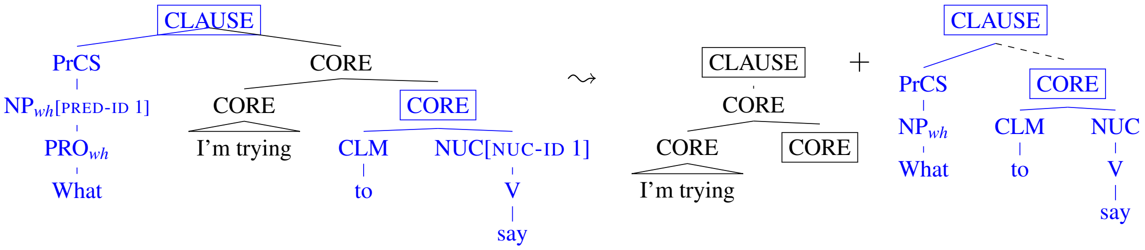


Figure 3: Extraction of tree with a d-edge for an LDD

   The antecedent and the following relative clause (marked with feature REF) are extracted to form a single d-edge tree. The antecedent of the extraposed relative clause is then removed from this d-edge tree and replaced by a substitution slot, as represented in Figure 4.
   After this step, an empty agenda is created and the extracted tree chunks and the pruned tree τ with the remaining nodes are placed into the agenda.

3. **Extract initial and sister-adjoining trees.** If no agenda with tree chunks was created in the previous step, an empty agenda is created in this step and the entire tree τ is placed into it. Each tree chunk in the agenda is traversed and the percolation tables are used to decide for each subtree $\tau_1 \dots \tau_n$ in the tree chunk whether it is a head, a complement or a modifier with respect to its parent. Initial trees for identified complements and sister-adjoining trees for identified modifiers are extracted recursively in the top-down fashion until each elementary tree has exactly one anchor site.

5 Evaluation of extracted TWGs

We extracted four TWGs for English, German, French, and Russian from the subcorpora of RRGparbank. We used silver and gold annotated data for our experiments, which means that each sentence was

[5] A node v_1 is left to another node v_2 if the leftmost leaf dominated by v_1 is left of the leftmost leaf dominated by v_2.

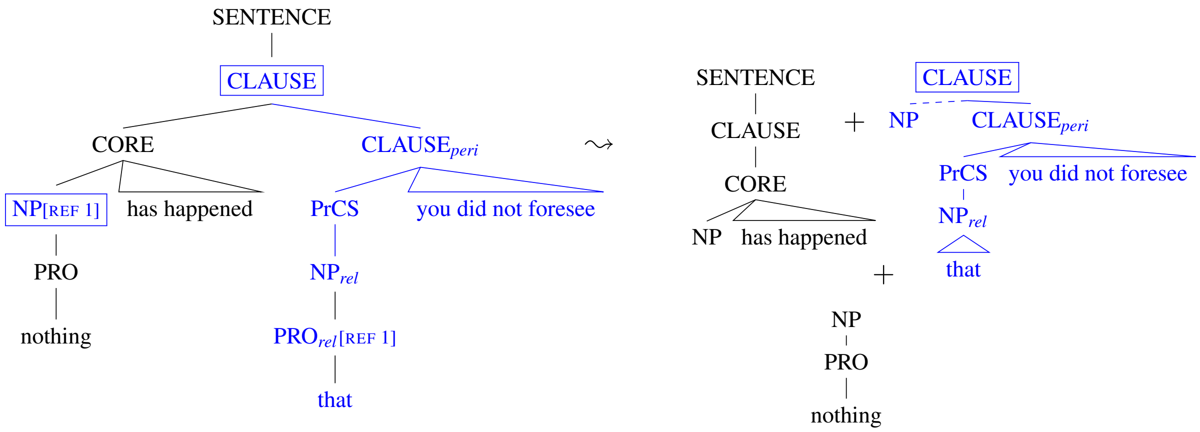


Figure 4: Extraction of a tree with a d-edge for an ERC

annotated and verified manually by at least one linguist. Table 1 provides statistics on the used annotated subcorpora from RRGparbank[6] and the occurrences of non-local dependencies (LDDs and ERCs) in subcorpora. NLDs are generally a relatively rare linguistic phenomenon (Candito and Seddah, 2012; Bouma, 2018). Compared to the other three languages, German shows a fairly large number of ERCs due to its dominant verb-final word order which does not allow putting heavy NPs at the end of the sentence.

Parameters	**English TWG**	**German TWG**	**French TWG**	**Russian TWG**
# word tokens	76893	41324	10550	35975
# word types	7193	7372	2571	9996
Avg. sentence length	14.12	13.5	12.4	10.03
# sentences	5445	3062	851	3586
# LDDs	58	13	36	27
# ERCs	8	110	4	0

Table 1: Statistics on annotated subcorpora in RRGparbank.

The extracted TWGs show a relatively large amount of supertags, more than a half of which occur only once in the corpus. Table 2 shows some statistics on the extracted grammars. The number of supertags with d-edges (which are used for wrapping substitution) is relatively low since the cases of NLDs are not frequent in the data.

Parameters	**English TWG**	**German TWG**	**French TWG**	**Russian TWG**
# supertags	3340	2591	947	2272
# supertags occuring once	1994	1689	584	1503
# initial trees	1727	1490	483	1350
# sister-adjoining trees	1571	1031	431	898
# d-edge trees	42	70	33	22
# nominal supertags	366	299	99	290
# verbal supertags	1382	1164	395	957

Table 2: Statistics on extracted TWG grammars.

We measured the similarity of the extracted TWGs for each language pair. In Table 3 we show the proportions of supertags in one grammar contained in the other grammar[7] (for example, the cell with the row name ‘English TWG’ and the column name ‘German TWG’ shows how many supertags from the German TWG are contained in the English grammar). The numbers show that the extracted grammars

[6] The annotation process of the subcorpora in RRGparbank is still in progress and the coverage of annotated sentences differs across the languages. Currently, around 81% of English data, 47% of German, 12% of French, 54% of Russian, and 15% of Farsi sentences are annotated.

[7] Please note that the annotation for different languages in RRGparbank is still in progress, and the proportion of common supertags can change in future.

tend to have a large number of supertags in common. For example, the smallest grammar French TWG (947 supertags) has around 55% supertags in common with the largest grammar for English (3340 supertags). There are 263 supertags common to all four grammars. In future work, we plan to explore the extent to which common supertags in grammars of different languages can be beneficial for multilingual parsing.

Common supertags	English TWG	German TWG	French TWG	Russian TWG
English TWG	–	24.97 (834)	15.45 (516)	21.8 (728)
German TWG	32.19 (834)	–	15.51 (402)	24.9 (645)
French TWG	54.49 (516)	42.45 (402)	–	37.80 (358)
Russian TWG	32.04 (728)	28.4 (645)	15.76 (358)	–

Table 3: Ratio of common supertags across language pairs in percents and in numbers (in brackets).

We used the TWG parser ParTAGe (Waszczuk, 2017; Bladier et al., 2020) in a symbolic way in order to validate our grammars and to check that the elementary trees in the extracted TWGs can be combined to produce the original trees.[8] While the majority of sentences could be processed by the parser (see Table 4), some complex sentences which contain an ERC resulting from the free-order placement of predicate arguments as in (4) above could not be parsed. We address these cases in our future work.

	English TWG	German TWG	French TWG	Russian TWG
% exactly matching parses	81	79.07	78.86	80.68
# not parsed sentences	13	8	5	10

Table 4: Validation of extracted TWGs on symbolic parsing with TWG parser ParTAGe.

6 Summary and future work

We presented work in progress on the extraction of TWGs for several languages from the multilingual treebank corpus RRGparbank. TWG is a tree-rewriting system developed for the formalization of Role and Reference Grammar (RRG). TWG is related to TAG and allows, among others, the adequate representation of non-local dependencies (NLDs) in sentences using the operation of wrapping substitution. We showed how wrapping substitution can be used to model various cases of NLDs, including long-distance relativization, long-distance wh-movement, long-distance topicalization, and extraposed relative clauses. We noticed cross-linguistic differences concerning the frequency of NLDs and the corresponding applications of wrapping substitution. At the same time, we observed a considerable overlap of supertags in the TWG grammars extracted for different languages. We validated the extracted grammars using a revised version of the TWG parser ParTAGe.

In future work, we plan to extract larger grammars from the RRG corpora (as the annotation of these projects progresses) and to use them in probabilistic parsing experiments. We also intend to include other languages from RRGparbank into parsing experiments, for example Hungarian and Farsi, depending on the availability of annotated data. Moreover, we will explore how wrapping substitution can be applied to model further linguistic phenomena, such as the variable placement of predicate arguments in languages with a relatively free word order. Finally, we plan to perform multilingual TWG parsing experiments, hopefully benefiting from the considerable number of common supertags across the extracted grammars.

Acknowledgements

We would like to thank three anonymous reviewers for their valuable comments. The work presented in this paper has been partially funded by the European Research Council, within the ERC grant TreeGraSP.

References

Tatiana Bladier, Andreas van Cranenburgh, Kilian Evang, Laura Kallmeyer, Robin Möllemann, and Rainer Osswald. 2018. RRGbank: a Role and Reference Grammar Corpus of Syntactic Structures Extracted from the

[8] For statistical TWG-parsing for English see (Bladier et al., 2020).

Penn Treebank. In *Proceedings of International Workshop on Treebanks and Linguistic Theory TLT17*, Oslo, December.

Tatiana Bladier, Kilian Evang, Laura Kallmeyer, Robin Möllemann, and Rainer Osswald. 2019. Creating RRG treebanks through semi-automatic conversion of annotated corpora. Abstract presented at the International Conference on Role and Reference Grammar 2019, Buffalo, USA, August 19-21, 2019.

Tatiana Bladier, Jakub Waszczuk, and Laura Kallmeyer. 2020. Statistical Parsing of Tree Wrapping Grammars. In *Proceedings of COLING*, December. To appear.

Gosse Bouma. 2018. Corpus-evidence for true long-distance dependencies in Dutch. *Grammar and Corpora*, pages 337–356.

Marie Candito and Djamé Seddah. 2012. Effectively long-distance dependencies in French: Annotation and parsing evaluation.

Tomaž Erjavec. 2012. MULTEXT-East: Morphosyntactic resources for central and eastern european languages. *Language Resources and Evaluation*, 46(1):131–142.

Aravind K Joshi and Yves Schabes. 1997. Tree-adjoining grammars. In *Handbook of formal languages*, pages 69–123. Springer.

Laura Kallmeyer, Rainer Osswald, and Robert D. Van Valin, Jr. 2013. Tree Wrapping for Role and Reference Grammar. In G. Morrill and M.-J. Nederhof, editors, *Formal Grammar 2012/2013*, volume 8036 of *LNCS*, pages 175–190. Springer.

Laura Kallmeyer. 2016. On the mild context-sensitivity of *k*-Tree Wrapping Grammar. In Annie Foret, Glyn Morrill, Reinhard Muskens, Rainer Osswald, and Sylvain Pogodalla, editors, *Formal Grammar: 20th and 21st International Conferences, FG 2015, Barcelona, Spain, August 2015, Revised Selected Papers. FG 2016, Bozen, Italy, August 2016, Proceedings*, number 9804 in Lecture Notes in Computer Science, pages 77–93, Berlin. Springer.

Rainer Osswald and Laura Kallmeyer. 2018. Towards a formalization of Role and Reference Grammar. In Rolf Kailuweit, Lisann Künkel, and Eva Staudinger, editors, *Applying and Expanding Role and Reference Grammar.*, pages 355–378. Albert-Ludwigs-Universität, Universitätsbibliothek. [NIHIN studies], Freiburg.

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

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

Jakub Waszczuk. 2017. *Leveraging MWEs in practical TAG parsing: towards the best of the two worlds*. Ph.D. thesis.

Fei Xia. 1999. Extracting tree adjoining grammars from bracketed corpora. In *Proceedings of the 5th Natural Language Processing Pacific Rim Symposium (NLPRS-99)*, pages 398–403.

Cross-Lingual Domain Adaptation for Dependency Parsing

Sara Stymne
Department of Linguistics and Philology
Uppsala University
sara.stymne@lingfil.uu.se

Abstract

We show how we can adapt parsing to low-resource domains by combining treebanks across languages for a parser model with treebank embeddings. We demonstrate how we can take advantage of in-domain treebanks from other languages, and show that this is especially useful when only out-of-domain treebanks are available for the target language. The method is also extended to low-resource languages by using out-of-domain treebanks from related languages. Two parameter-free methods for applying treebank embeddings at test time are proposed, which give competitive results to tuned methods when applied to Twitter data and transcribed speech. This gives us a method for selecting treebanks and training a parser targeted at any combination of domain and language.

1 Introduction

Recent advances in dependency parsing have enabled high-quality parsing for a relatively high number of languages. However, satisfactory results are mainly limited to text types for which there are treebanks for a specific language. Even for high-resource languages, treebanks are typically only available for a small number of domains and genres. In this work we show how we can improve parsing for non-canonical text types by using in-domain annotated data from other languages.

We focus on two low-resource text types that stand out in different respects from canonical written texts: Twitter data and (transcribed) spoken data, for which annotated treebanks exist for only a small number of languages. Twitter data often contains non-standard language and specific features such as hash tags and emoticons. Spoken data tends to be more informal than written texts, and contains features such as fillers, restarts, and reparandums. While Twitter can be regarded as a genre, and spoken data as a medium (Lee, 2001), we will follow previous work in NLP and use the term *domain* to cover both these types of text.[1]

The main novelty in this work is that we combine domain adaptation with cross-lingual learning for dependency parsing. We note that treebanks for a specific domain (IND: in-domain) often exist for some languages, and we show that we can take advantage of such data for parsing this domain in other languages. Our main focus is on the case where we want to parse data for a language that has some resources, but none for the domain in question (OOD: out-of-domain). While there is plenty of work both on cross-lingual parsing (Ammar et al., 2016a; Ahmad et al., 2019; Kondratyuk and Straka, 2019) and domain adaptation for parsing (Kim et al., 2016; Sato et al., 2017; Xiuming et al., 2019), there is to the best of our knowledge no attempts to combine these approaches in a uniform framework for dependency parsing.

We adapt the parsing framework of Smith et al. (2018a) which incorporates treebank embeddings to represent treebanks, similarly to how language embeddings has been used to represent the languages (Ammar et al., 2016b; de Lhoneux et al., 2017a). In this framework each parsing model is trained on a concatenation of different treebanks, and the representation of each input token includes an embedding

[1] The term *domain* has often been used as a catch-all term in NLP, to cover many different types of text type differences, often without being clearly defined, see e.g. (Weiss et al., 2016; Chu and Wang, 2018), even though there has been some attempts to investigate different aspects of *domains*, e.g. (van der Wees et al., 2015; Ruder et al., 2016).

representing the treebank from which the token comes from. Depending on the mix of treebanks, the treebank embedding can encode aspects such as differences between languages, domains, and annotation style. Parsing with treebank embeddings has previously been applied monolingually (Stymne et al., 2018; Wagner et al., 2020) and cross-lingually for related languages, but without taking domain into account (Smith et al., 2018a; Lim et al., 2018),[2] In this paper, we show that joint training with treebank embeddings can be applied simultaneously across both across languages and domains, in effect addressing the task of cross-lingual domain adaptation. It is a simple and efficient method, which does not require expensive pre-processing, pre-training, translation, or similar tasks required by many other cross-lingual approaches, while giving competitive results across many settings. In this work we explore how such a resource lean method can be applied to cross-domain parsing on its own. We leave to future work an investigation of how the proposed technique interacts with other techniques for domain adaptation, for instance based on pre-training contextualized embeddings like BERT (Devlin et al., 2019).

At test time, there is a need to determine which treebank embedding to use, which is straightforward for test data from a treebank used during training. However, when the input sentence is from a treebank not used during training there is a need to determine the treebank embedding. One option is to use a *proxy* treebank (Stymne et al., 2018), i.e. to choose the embedding of one of the treebanks used during training, which can be determined based on development data. Wagner et al. (2020) show that it is often advantageous to interpolate the embeddings of the treebanks used for training instead. They show in a monolingual setting how interpolation weights can be learnt based on sentence similarity. However, their equal weight baseline performs just as well in the majority of cases, and avoids the need of learning interpolation weights, which would also be less straight-forward in the cross-lingual setting. We thus adopt equal-weight interpolation. We also propose the use of an ensembling strategy applied to trees obtained by using all possible proxy treebanks embeddings.

We show that using in-domain data from another language is useful when no in-domain data is available for the target language. Using the proposed methods, we can potentially train a parser for any combination of domain and language, as long as that domain has training data in some language, without the need for tuning on target development data.

2 Experimental Setup

Data We mainly use data from the Universal Dependencies (UD) project (Nivre et al., 2020), version 2.4 (Nivre et al., 2019). We put our main focus on languages with a single-domain dependency treebank with either spoken data or Twitter data, including both training and test data and additional treebank data for other domains. While several UD treebanks contain some data from these domains mixed with other domains, it is often not easily identifiable which domain specific sentences come from. We thus use the three UD single domain treebanks of spoken data for French, Norwegian, and Slovenian, which fulfills our requirements. In addition we evaluate our methods on Komi-Zyrian and Naija, which both have spoken test data, but no training data for any domain in UD. For Twitter we use two treebanks from UD for Italian and code-switching Hindi–English. In addition we use the English Tweebank v2, which is annotated in UD style (Liu et al., 2018). We convert sentences in the English Tweebank with multiple roots to have only one root, which is a UD requirement, by only keeping the first root, and joining the other roots to it with the *parataxis* relation. This happens when a single Tweet contains more than one sentence, and it is the solution adopted in the Italian PoSTWITA treebank.

In addition to the in-domain treebanks we use additional treebanks from the same language, when available, or for related languages otherwise. For Komi Zyrian, a Uralic lanugage, we also use a Russian treebank, since Russian is a contact language, which also shares the Cyrillic script, in contrast to other Uralic treebanks with training data. Table 1 lists the data used for each language. Note that in all cases, the additional data is much larger than the in-domain data, which is typically quite small. For Slovenian SST, no development data was available, so we split off 5% of the training data. In all other cases we use the original splits. While UD treebanks have standard annotation guidelines, there are several inconsistencies

[2] With the exception of a footnote in Smith et al. (2018a), where this type of data combination is mentioned for spoken French and Naija. However, no details or experimental results are provided.

Language	IND Treebank	Train	Dev	Test	Additional OOD data
French	Spoken	15.0K	10.2K	10.2K	GSD (364K), *Partut* (24.9K), Sequoia (51.9K)
Norwegian	NynorskLIA	35.2K	10.2K	10.0K	*Nynorsk* (245K), Bokmaal (244K)
Slovenian	SSJ	18.6K	906	10.0K	*SST* (113K), Croatian_SET (153K), Serbian_SET (74.3K)
Komi Zyrian	IKDP	–	–	1.3K	Finnish_TDT (163K), North_Sami_Giella (16.8K), Russian_Taiga (18.1K)
Naija	NSC	–	–	12.9K	English: EWT (205K), GUM (66.2K), LinES (50.1K) ParTUT (43.5K)
English	Tweebank	24.8K	11.8K	19.1K	EWT (205K), GUM (66.2K), LinES (50.1K) *ParTUT* (43.5K)
Hindi–English CS	HIENCS	19.3K	3.3K	3.1K	English: EWT (205K), GUM (66.2K), LinES (50.1K) *ParTUT* (43.5K), *Hindi_HDTB* (281K)
Italian	PoSTWITA	104K	12.8K	13.2K	*ISDT* (294K), ParTUT (52.4K), VIT (241K)

Table 1: Treebanks and number of tokens in train, dev, and test data sets for the target treebanks. Top of table is spoken data, and bottom is for Twitter data. Additional data lists treebanks used for each target treebank, which is in-language unless otherwise noted, and the number of tokens in the training set for each treebank. Treebanks in italics are used in the contrastive data sets.

between the treebanks used, especially for the rather unusual features of spoken data and Twitter. For instance, see Liu et al. (2018) for a discussion of differences between English and Italian Twitter treebanks, or the Naija-NSC documentation for known deviations from UD standards.[3]

To be able to compare the effect of adding in-domain data, we create a contrastive treebank for each IND language of the same size, counted in the number of tokens. We use data from the treebank(s) marked with italics in Table 1.

We think the language sample is interesting and covers many aspects. Even though the majority of languages are Indoeuropean, they mostly have different genera. They range from having hardly any resources like Komi Zyrian, to large resources, like English, and cover some interesting special cases, such as code switching, a Creole language, Naija, and a language with two written varieties, Norwegian.

Parser We use uuparser[4] (de Lhoneux et al., 2017b) which is a transition-based dependency parser using the arc-hybrid transition system with the addition of a swap transition and a static-dynamic oracle, to be able to handled non-projectivity. The parser uses a two-layer BiLSTM as a feature extractor followed by a multi-layer perceptron predicting transitions, in the style of Kiperwasser and Goldberg (2016). Each word, w_i, is represented by the concatenation of a word embedding, $e_w(w_i)$, a character-level embedding, obtained by running a BiLSTM over the characters ch_j ($1 \leq j \leq m$) of w_i, where m is the word length in characters, and a treebank embedding, $e_{tb}(t^*)$:

$$e_i = [e_w(w_i); \text{BiLSTM}(ch_{1:m}); e_{tb}(t^*)] \tag{1}$$

The treebank embedding represents a treebank, t^*, which is chosen among the set of k treebanks used when training the model. During training, t^* is chosen as the treebank to which the current word/sentence belongs. When applying the model, the treebank of the sentence can be used only if the test sentence comes from a treebank that was used during training. In other cases some other method has to be used. In this work we explore the following methods:

- Proxy treebank: when dev data is available, we can try all possible proxy treebanks i.e. all treebanks used during training the model, and choose the treebank, t^*, which performs best on dev data.
- Interpolation: We interpolate the embeddings from all treebanks used during training by averaging them with equal weights: ($t^* = \sum_{t=1}^{k} \frac{1}{k} e_{tb}(t)$)
- Ensemble: We run the model with each possible proxy treebank, obtaining k output trees. Then we apply the reparsing technique by Sagae and Lavie (2006) which applies the Chu-Liu-Edmonds (Edmonds, 1967) algorithm with each arc being weighted by the number of trees for which that arc was predicted.[5]

Note that in all cases we only apply these techniques at test time. The interpolation method only requires a single test run. Proxy treebank requires k dev test runs, followed by a single test run. Ensembling

[3] https://github.com/UniversalDependencies/UD_Naija-NSC/blob/master/README.md

[4] https://github.com/UppsalaNLP/uuparser

[5] Weighting the arcs by development UAS or LAS instead had little impact on the results, but requires development data.

					Spoken						Twitter					
	Same language		Other language		French		Norwegian		Slovenian		Italian		English		Hindi–English		**Mean**	
	IND	OOD	IND	OOD	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS
1	–	–	–	X	18.2	2.8	24.2	8.8	25.9	7.3	27.8	7.2	50.7	33.2	31.4	19.4	29.7	13.1
2	–	–	X	–	21.8	2.2	19.1	3.6	20.7	4.7	23.7	7.5	53.8	40.5	35.0	23.0	29.0	13.6
3	–	X	–	–	74.8	63.4	60.1	52.8	60.2	46.9	71.7	62.8	68.2	55.7	37.0	25.0	62.0	51.1
4	–	X	–	X	75.3	64.3	62.2	54.4	60.1	47.6	72.5	63.4	67.0	54.6	35.9	24.9	62.2	51.5
5	–	X	X	–	75.9	64.5	59.1	52.0	63.2	52.7	73.9	65.5	69.5	58.9	38.0	25.7	63.3	53.2
6	X	–	–	–	76.6	69.4	74.3	69.4	65.8	57.9	82.3	76.9	74.7	68.6	65.0	52.7	73.1	65.8
7	X	–	X	–	76.1	68.8	73.9	67.8	65.3	57.9	81.8	76.3	76.3	70.6	64.1	52.6	72.9	65.5
8	X	X	–	–	84.0	79.2	78.3	73.5	71.8	65.6	84.2	79.4	82.8	78.0	67.6	52.7	78.1	72.1
9	X	X	X	–	83.7	78.0	78.7	73.7	72.7	66.1	84.5	79.5	82.1	77.4	67.2	56.9	78.2	72.0

Table 2: Test set scores for spoken data with different combinations of training data, using the best proxy treebank. For each line, only data sources marked 'X' are used, sources marked '-' are not used. Note that 'Same language' also includes related Slavic languages for Slovenian.

is heavier, requiring *k* test runs, followed by an application of the CLU algorithm. Interpolation and ensembling both have the advantage of being parameter free, while proxy treebank requires dev data. For languages without dev data we also compare our results to the oracle score, where we pick the best proxy treebank based on test performance.

We use the default hyperparameters of uuparser, as specified in Smith et al. (2018a). Note that no POS-tags are used, since POS-tagging in these difficult domains would lead to the same issues as for parsing. In addition, character embeddings compensate for the lack of POS-tags to a large extent across several typologically different languages (Smith et al., 2018b), and in order for universal POS-tags, the most feasible choice cross-lingually, to be useful for parsing, the tagging quality has to be prohibitively high (Gómez-Rodríguez, 2020). The parser is trained end-to-end on treebank data, without any pre-training. All embeddings are initialized randomly at training time. Each model is trained for 30 epochs, and the best epoch is chosen based on average development scores among treebanks used at training time.

Evaluation Metrics We use unlabelled and labelled attachment score, UAS and LAS, as evaluation metrics. Our system was optimized based on development UAS scores, since we believe that it is a good fit to the case of inconsistent labeling in the treebanks for each target domain. Overall, the test results reflect the trends seen in development data relatively well.

3 Results

We first present results using different sources of training data, IND or OOD, from the same or another language, choosing the best proxy treebank based on development UAS scores. We use the full set of treebanks from Table 1.[6] For other language OOD data, we use the contrastive datasets sampled from the same languages as the other language IND data.

Our main interest is the middle part of Table 2, lines 3–5, where we investigate the effect of adding IND data from other languages to in-language OOD data. Adding out-of-language IND data leads to average improvements of 2.1 LAS points and 1.3 UAS points. It always helps for Twitter, and helps in all cases except Norwegian for spoken data. If we instead add an equivalent amount of out-of-language OOD data, we see minor average gains and a performance that is considerably worse than for IND data. Norwegian is an outlier here as well, with good results for OOD data. We leave an investigation of why to future work. These results confirm that our treebank combination strategy is useful.

The two top lines of Table 2 simulates results when no in-language data is available. As expected these scores are considerably lower than when using in-language OOD data, being so poor that these parsers are hardly useful, confirming previous research, e.g. Meechan-Maddon and Nivre (2019) and Vania et al. (2019). In this case there is no clear difference between IND and OOD data. The scores for English and Hindi–English with IND data are closer to in-language OOD scores, which can be explained by the partial language match between these two treebanks.

As a point of comparison, the bottom part of Table 2 shows the results when data matching both language and domain is available. As expected, it leads to large gains. For all languages, the model trained

[6] Using a subset of these treebanks mostly gave lower scores but showed the same trends.

	Proxy Language			Proxy Domain			Interpolation		Ensemble	
	UAS	LAS	Proxy	UAS	LAS	Proxy	UAS	LAS	UAS	LAS
French	75.9	64.5	fr_partut	76.0	61.7	no_nynorsklia	75.2	63.7	75.6	63.8
Norwegian	59.1	52.0	no_bokmaal	60.2	51.0	sl_sst	61.2	53.5	60.3	50.8
Slovenian	60.7	50.0	sl_ssj	59.6	47.1	no_nynorsklia	60.6	48.8	60.8	47.8
Slovenian+Slavic	63.2	49.6	sr_set	61.3	48.1	no_nynorsklia	63.8	52.2	63.9	48.8
Italian	73.9	65.5	it_partut	67.2	55.2	en_tweet	73.9	64.6	74.4	62.7
English	69.5	58.9	en_partut	66.3	53.8	it_postwita	70.2	61.5	69.4	58.6
Hindi–English	37.6	25.7	en_partut	38.0	26.3	en_tweet	35.4	26.0	37.6	25.6
Mean:	62.8	52.3		61.2	49.0		62.9	52.9	63.1	51.1

Table 3: Test scores for models trained on all available in-language OOD data and IND data from the other languages, using different methods for applying it to the target treebank.

on only the relatively small in-language IND data beats all models trained without it, even though the gap is quite small for French and Slovenian. The gains are especially pronounced for the code-switched Hindi–English and for Norwegian. When in-language IND data is available we see no average gains from adding out-of-language IND data, whereas adding in-language OOD data always helps considerably. We also note that the gap between UAS and LAS gets smaller, when the training data fits the test data better, supporting our intuition that out-of-language OOD data helps more with structure than labels.

Next, we focus on our main scenario of interest, where we have in-language OOD data and out-of-language IND data. We use the model from line 5 in Table 2 and also show results for Slovenian without the additional Slavic languages. We investigate how best to apply the model at test time for cases where the treebank, i.e. the combination of language and domain, has not been seen at training time. We compare using a proxy treebank, matching either language or domain, interpolation, and ensembling. Table 3 summarizes the results.

When choosing a single proxy it is on average 3.3 LAS points and 1.6 UAS points better to use the same language than the same domain, but there is some variation between languages. The interpolation method works well on both metrics, giving the best average LAS scores and competitive UAS scores. Ensembling gives the highest UAS scores by a small margin, but does worse on LAS. We also note that including the related Slavic languages improves parsing for Slovenian considerably, with an LAS gain of 3.4 for the interpolation strategy.

Table 3 also shows the best proxy used, either matching domain or language. For language proxies we note some surprises, Norwegian Bokmaal is a better fit than the matching language variety Nynorsk, and the Serbian corpus is better than Slovenian in the Slavic setting. We also note that the ParTUT treebank is often a good proxy. The differences between proxies are typically small, though. The domain proxies seem more straight-forward, with Norwegian and English being preferred more than the other options. The only small surprise is that Italian was a better fit for English than the partially matching Hindi–English treebank. There could, however, be many reasons for this, such as more similar annotation schemes for Italian and English, or the fact that while there is a partial overlap with English, Hindi is less related to English than Italian.

Finally we apply our methods to the two low-resource languages without any in-language training data. Here, we have no development data for choosing a proxy language, so the focus is on our two parameter free methods: interpolation and ensembling. As a point of comparison we give the oracle score of the proxy treebank with the highest UAS score. We compare three models: using only the close OOD languages from Table 1, and adding either all three IND spoken treebanks or the contrastive OOD treebanks. Results are shown in Table 4. Interestingly, adding the small data from the unrelated languages helps somewhat regardless of if this data is OOD or IND. Adding the IND data do present the overall best scores, though, with the highest UAS scores for Komi Zyrian and the highest LAS scores for Naija. For our target model, interpolation and ensembling works quite well, often tying with the oracle scores, and typically not falling too much behind the oracle. However, in the setting with only related languages, these two methods falls behind the oracle, indicating that these methods works better with a more diverse mix of training languages and domains.[7]

Our experiments confirm the usefulness of our proposed method of mixing training treebanks and

[7] We saw the same trend when we applied these methods to the languages in Table 2.

		Related OOD			Related OOD + other OOD			Related OOD + other IND		
		Oracle	Interp	Ens	Oracle	Interp	Ens	Oracle	Interp	Ens
Komi Zyrian	UAS	32.1	26.4	30.8	32.9	32.9	31.9	35.4	34.4	33.9
	LAS	18.3	14.8	18.4	19.0	19.1	18.2	20.0	19.0	18.7
Naija	UAS	43.1	41.4	41.0	44.1	43.4	43.5	43.2	43.1	44.1
	LAS	28.9	28.0	27.4	29.1	28.6	27.8	30.2	30.0	28.3

Table 4: Test set scores for languages without any training data, using different training data combination, with the oracle proxy treebank, interpolation, or ensembling.

applying the model to new data. Treebank embeddings seem to be capable of encoding aspects both of domain and language.[8] Both interpolation and ensembling have the advantage that they do not require any tuning on development data, which choosing a single proxy does. Interpolation has the further advantage that it requires no extra processing, and seems preferable since it gives the best LAS scores, as well as competitive UAS scores.

4 Conclusion

In this paper we have shown how we can improve parsing for specific domains by combining data in that domain but from another language with in-language out-of-domain data. We show that it is possible to do so using a parsing model with treebank embeddings. We also propose the use of two parameter free methods for applying treebank embeddings to new data at test time, which give competitive results compared to optimizing a proxy treebank based on development data. This indicates that treebank embeddings are able to capture aspects both about text type and language. We also think it is worth noting that in contrast to much previous work, e.g. Smith et al. (2018a), we see gains for languages which are not closely related.

In future work we want to apply our methods also to other text types and to explore how the data selection strategies work with other parsing frameworks. We also want to extend the work on weighted interpolation by Wagner et al. (2020) to the cross-lingual case, to be able to combine it with the proposed methods. Another line of work is to investigate how much annotated data is needed in order to see gains of the same size as when adding IND treebanks from other languages.

In this work we did not take advantage of any type of pre-trained word embeddings. It is likely that either cross-lingual static word embeddings (Ruder et al., 2019) or multilingual dynamic word embeddings, like multilingual BERT (Devlin et al., 2019) could improve the results overall. Using either of these resources would also allow us to utilize IND in-language unlabeled data in the pre-training step, which might potentially lead to improvements. We do believe that seeing labelled data, with arc types that are specific to the text types in question, as we do in this work, is also useful. It is an open question, which we leave to future work, how pre-training would interact with our proposed method.

Acknowledgments

Thank you to current and former members of the Uppsala parsing group for many fruitful discussions: Ali Basirat, Daniel Dakota, Miryam de Lhoneux, Artur Kulmizev, Joakim Nivre, and Aaron Smith. I would also like to thank the anonymous reviewers for their insightful comments.

References

Wasi Ahmad, Zhisong Zhang, Xuezhe Ma, Eduard Hovy, Kai-Wei Chang, and Nanyun Peng. 2019. On difficulties of cross-lingual transfer with order differences: A case study on dependency parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 2440–2452, Minneapolis, Minnesota, US.

Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah Smith. 2016a. Many languages, one parser. *Transactions of the Association for Computational Linguistics*, 4:431–444.

[8] We also experimented with separate embeddings for domain and language, which gave lower scores.

Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2016b. Many languages, one parser. *Transactions of the Association for Computational Linguistics*, 4:431–444.

Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319, Santa Fe, New Mexico, USA.

Miryam de Lhoneux, Yan Shao, Ali Basirat, Eliyahu Kiperwasser, Sara Stymne, Yoav Goldberg, and Joakim Nivre. 2017a. From raw text to universal dependencies - look, no tags! In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 207–217, Vancouver, Canada.

Miryam de Lhoneux, Sara Stymne, and Joakim Nivre. 2017b. Arc-hybrid non-projective dependency parsing with a static-dynamic oracle. In *Proceedings of the 15th International Conference on Parsing Technologies*, pages 99–104, Pisa, Italy.

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*, pages 4171–4186, Minneapolis, Minnesota, US.

Jack Edmonds. 1967. Optimum branchings. *Journal of Research of the National Bureau of Standards*, 71B:233–240.

Mark Anderson Carlos Gómez-Rodríguez. 2020. On the frailty of universal POS tags for neural UD parsers. In *Accepted to CoNLL 2020.*

Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2016. Frustratingly easy neural domain adaptation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 387–396, Osaka, Japan.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional LSTM feature representations. *Transactions of the Association for Computational Linguistics*, 4:313–327.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing universal dependencies universally. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 2779–2795, Hong Kong, China.

David Y. W. Lee. 2001. Genres, registers, text types, domains, and styles: Clarifying the concepts and navigating a path through the BNC jungle. *Language Learning & Technology*, 5(3):37–72.

KyungTae Lim, Cheoneum Park, Changki Lee, and Thierry Poibeau. 2018. SEx BiST: A multi-source trainable parser with deep contextualized lexical representations. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 143–152, Brussels, Belgium.

Yijia Liu, Yi Zhu, Wanxiang Che, Bing Qin, Nathan Schneider, and Noah A. Smith. 2018. Parsing tweets into universal dependencies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 965–975, New Orleans, Louisiana.

Ailsa Meechan-Maddon and Joakim Nivre. 2019. How to parse low-resource languages: Cross-lingual parsing, target language annotation, or both? In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 112–120, Paris, France.

Joakim Nivre, Mitchell Abrams, Željko Agić, et al. 2019. Universal dependencies 2.4. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajic, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of The 12th Language Resources and Evaluation Conference (LREC 2020)*, pages 4034–4043, Marseille, France.

Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016. Towards a continuous modeling of natural language domains. In *Proceedings of the Workshop on Uphill Battles in Language Processing: Scaling Early Achievements to Robust Methods*, pages 53–57, Austin, TX.

Sebastian Ruder, Ivan Vulić, and Anders Sogaard. 2019. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:69–631.

Kenji Sagae and Alon Lavie. 2006. Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 129–132, New York City, USA.

Motoki Sato, Hitoshi Manabe, Hiroshi Noji, and Yuji Matsumoto. 2017. Adversarial training for cross-domain universal dependency parsing. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 71–79, Vancouver, Canada.

Aaron Smith, Bernd Bohnet, Miryam de Lhoneux, Joakim Nivre, Yan Shao, and Sara Stymne. 2018a. 82 treebanks, 34 models: Universal dependency parsing with multi-treebank models. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 113–123, Brussels, Belgium.

Aaron Smith, Miryam de Lhoneux, Sara Stymne, and Joakim Nivre. 2018b. An investigation of the interactions between pre-trained word embeddings, character models and POS tags in dependency parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2711–2720, Brussels, Belgium.

Sara Stymne, Miryam de Lhoneux, Aaron Smith, and Joakim Nivre. 2018. Parser training with heterogeneous treebanks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 619–625, Melbourne, Australia.

Marlies van der Wees, Arianna Bisazza, Wouter Weerkamp, and Christof Monz. 2015. What's in a domain? Analyzing genre and topic differences in statistical machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, Short Papers*, pages 560–566, Beijing, China.

Clara Vania, Yova Kementchedjhieva, Anders Sogaard, and Adam Lopez. 2019. A systematic comparison of methods for low-resource dependency parsing on genuinely low-resource languages. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1105–1116, Hong Kong, China.

Joachim Wagner, James Barry, and Jennifer Foster. 2020. Treebank embedding vectors for out-of-domain dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8812–8818, Online.

Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. 2016. A survey of transfer learning. *Journal of Big Data*, 3(1):1–40.

Qiao Xiuming, Zhang Yue, and Zhao Tiejun. 2019. Learning domain invariant word representations for parsing domain adaptation. In *Natural Language Processing and Chinese Computing (NLPCC 2019)*, pages 801–813, Dunhuang, China.

How tight is your language?

A semantic typology based on Mutual Information

Natalia Levshina
MPI for Psycholinguistics, Nijmegen
Neurobiology of Language Department
natalia.levshina@mpi.nl

Abstract

Languages differ in the degree of semantic flexibility of their syntactic roles. For example, English and Indonesian are considered more flexible with regard to the semantics of subjects, whereas German and Japanese are less flexible. In Hawkins' classification, more flexible languages are said to have a loose fit, and less flexible ones are those that have a tight fit. This classification has been based on manual inspection of example sentences. The present paper proposes a new, quantitative approach to deriving the measures of looseness and tightness from corpora. We use corpora of online news from the Leipzig Corpora Collection in thirty typologically and genealogically diverse languages and parse them syntactically with the help of the Universal Dependencies annotation software. Next, we compute Mutual Information scores for each language using the matrices of lexical lemmas and four syntactic dependencies (intransitive subjects, transitive subject, objects and obliques). The new approach allows us not only to reproduce the results of previous investigations, but also to extend the typology to new languages. We also demonstrate that verb-final languages tend to have a tighter relationship between lexemes and syntactic roles, which helps language users to recognize thematic roles early during comprehension.

1 Theoretical background and aims of the paper

This paper proposes a quantitative bottom-up corpus-based approach to cross-linguistic comparison, determining how tightly or loosely different lexemes can be mapped on basic syntactic roles. The idea goes back to Hawkins (1986: 121– 127, 1995; see also Müller-Gotama 1994), who coined the terms 'tight-fit' and 'loose-fit' languages. The former have unique surface forms that map onto more constrained meanings, whereas the latter have more vague forms with less constrained meanings. For instance, Present-Day English has fewer semantic restrictions on the subject and object than Old English, German or Russian. Consider several examples below.

(1) a. Locative: *This tent sleeps four.*
b. Temporal: *2020 witnessed a spread of the highly infectious coronavirus disease.*
c. Instrument: *10 Euros will buy you a meal.*
d. Source: *The roof leaks water.*

While these sentences are perfectly acceptable in English, their German or Russian equivalents would be unacceptable or strange. This means that subjects in English are less semantically restricted than subjects in German and Russian (see also Plank 1984).

Tightness and looseness have several components. Semantic flexibility of arguments is only one of them. Other features of tight languages include formal case marking, avoidance of raisings and long WH-movements and lower reliance on context in interpretation.

Languages can change their degree of tightness. English is a well-known example of shifting from tight to loose (Hawkins 1986). As the case was lost, the zero-marked NPs in Middle English became more dependent on the verb for theta-role assignment. This is why the rigid SVO order emerged, which helps language users to understand correctly who did what to whom. Also, new instrumental and locative subjects as in (1) became possible, which used to be the case only in prepositional phrases.

In the previous work, the judgements about tightness and looseness were made introspectively and qualitatively. This paper presents a method that allows one to quantify these differences objectively with the help of corpus data. We only focus on the fit between syntactic roles and semantics of lexemes in this study. As a proxy for semantics, we extract frequencies of lexemes in different syntactic roles from syntactically parsed corpora (see Section 2). Next, we compute how much these frequencies diverge from the total frequencies of the roles with the help of the Mutual Information metric. The higher a score, the tighter the language (see Section 3). The scores are then compared with the existing classification of languages. We find a close correspondence between the scores (see Section 4). The scores are computed for lemmas alone and for lemmas plus multiword units. We also investigate the correlation between tightness and the proportion of verb-final frames in a corpus (Section 5). Section 6 provides the conclusions and an outlook.

2 Data

In order to extract the distributional information, one needs large corpora. Available cross-linguistic syntactically annotated collections, such as the Universal Dependencies corpora (Zeman et al. 2020), are too small for the purposes of the present study. The solution was to use the Leipzig Corpora Collection (Goldhahn et al. 2012), which contains freely downloadable web-based corpora of reasonable size in more than 200 languages. The language sample used for the present study includes thirty languages, which are listed in Table 1. Each language is represented by one million sentences from online news (categories 'news' and 'newscrawl'). The corpora contain sentences in random order. The choice of languages was determined by the availability of sufficient data and a reasonably good language model in the UDPipe annotation tools.

The sentences were tokenized, lemmatized and morphologically and syntactically annotated with the help of the UD corpus tools (Straka & Straková 2017) in the R package *udpipe* (Wijffels et al. 2019). The language models, which were trained on the UD corpora (Zeman et al. 2020), provide, among other things, universal parts-of-speech tags and dependency relations, which can be compared across different languages. This is crucial for the purposes of the present study.

One should be aware of risks involved in using automatic parsers for cross-linguistic data analysis. Manual evaluation of the annotation was impossible, given the size and diversity of the data. However, ongoing research (Levshina, Submitted) indicates very strong correlations between diverse morphological and word-order parameters based on the same annotated corpora and on the training corpora from the Universal Dependencies collection, as far as the core arguments are concerned. This gives us some confidence in the results.

The following universal dependencies, which represent syntactic arguments, were extracted from the annotated corpora:

- *nsubj* (lexical, or non-clausal subject), e.g. *The student is reading*. Subjects in transitive and intransitive clauses were treated separately. A head verb was considered transitive if it had an overt object.
- *obj* (object), e.g. *I see the student.*
- *obl* (oblique, i.e. any non-core nominal argument or adjunct), e.g. *I'm talking with a student; She's reading in the library.*

The UD approach does not distinguish between oblique arguments and adjuncts. In addition, many languages do not have indirect object (*iobj*) as a separate dependency. This is why indirect objects, which were not very numerous, were counted as a joined category of indirect objects + obliques for the sake of cross-linguistic comparability. The more detailed tags in the dependencies, such as *nsubj:pass* (subject of a passive clause) were treated as simply *nsubj*, *obj* or *obl*. The reason is that such extended tags are language-specific and not used in a unified way across the languages.

Language	Genus	Family	UD model	Lemmas
Arabic	Semitic	Afro-Asiatic	arabic-padt-ud-2.4	16,799
Bulgarian	Slavic	Indo-European	bulgarian-btb-ud-2.4	11,924
Croatian	Slavic	Indo-European	croatian-set-ud-2.4	13,791
Czech	Slavic	Indo-European	czech-pdt-ud-2.4	11,783
Danish	Germanic	Indo-European	danish-ddt-ud-2.4	16,340
Dutch	Germanic	Indo-European	dutch-alpino-ud-2.4	13,334
English	Germanic	Indo-European	english-ewt-ud-2.4	10,480
Estonian	Finnic	Uralic	estonian-edt-ud-2.4	20,231
Finnish	Finnic	Uralic	finnish-tdt-ud-2.4	20,822
French	Romance	Indo-European	french-gsd-ud-2.4	9,386
German	Germanic	Indo-European	german-gsd-ud-2.4	16,729
Greek (modern)	Greek	Indo-European	greek-gdt-ud-2.4	13,789
Hindi	Indic	Indo-European	hindi-hdtb-ud-2.4	10,546
Hungarian	Ugric	Uralic	hungarian-szeged-ud-2.4	13,931
Indonesian	Malayo-Sumbawan	Austronesian	indonesian-gsd-ud-2.4	9,820
Italian	Romance	Indo-European	italian-isdt-ud-2.4	10,643
Japanese	Japanese	Japanese	japanese-gsd-ud-2.4	19,198
Korean	Korean	Korean	korean-gsd-ud-2.4	29,017
Latvian	Baltic	Indo-European	latvian-lvtb-ud-2.4	12,062
Lithuanian	Baltic	Indo-European	lithuanian-hse-ud-2.4	17,652
Persian	Iranian	Indo-European	persian-seraji-ud-2.4	11,440
Portuguese	Romance	Indo-European	portuguese-bosque-ud-2.4	9.663
Romanian	Romance	Indo-European	romanian-rrt-ud-2.4	12,962
Russian	Slavic	Indo-European	russian-syntagrus-ud-2.4	10,092
Slovenian	Slavic	Indo-European	slovenian-ssj-ud-2.4	13,094
Spanish	Romance	Indo-European	spanish-gsd-ud-2.4	10,317
Swedish	Germanic	Indo-European	swedish-talbanken-ud-2.4	16,096
Tamil	Southern Dravidian	Dravidian	tamil-ttb-ud-2.4	14,737
Turkish	Turkic	Altaic	turkish-imst-ud-2.4	12,554
Vietnamese	Viet-Muong	Austro-Asiatic	vietnamese-vtb-ud-2.4	16,552

Table 1: Languages and UD language models used in the present study.

Next, the lexemes (lemmas) performing these syntactic roles were extracted. The analyses presented below are based only on common nouns, following the tradition of word order research in typology, but the scores for a wider range of lexemes were computed, as well, including proper nouns, verbs, adjectives, symbols and numerals. Pronouns were excluded because of the lack of anaphora resolution in the corpora and the fact that the languages have vastly different pronominal systems with different pro-drop rates. The correlations between the Mutual Information scores based on these lexemes and the ones based on common nouns only are very strong and positive: $r = 0.914$, $p < 0.0001$ for lemmas only and r = 0.944, p < 0.0001 for lemmas and MWE.

If there was coordination (e.g. *Students and teachers came to the party*), the subsequent coordinated elements marked with the dependency 'conj' (i.e. *teachers* in the example) were treated as having the same dependency as the first coordinate member (i.e. *students*). The cleaning procedure involved removing punctuation marks in the beginning and at the end of the strings and normalizing the case. The lemmas with the frequency of 10 and less were left out because they were often analyzed erroneously.

An important issue in language comparison is what to count as a word (Haspelmath 2011). For example, in English, the phrase *art history* consists of two words, but its German equivalent *Kunstgeschichte* is only one word. In order to counterbalance the influence of orthographic conventions, we also computed the scores treating multiword units like *art history* as one lexeme. In order to identify multiword expressions (MWE), we used the following dependencies in the UD annotation: *compound*, *fixed* and *flat*. The dependency *compound* is used to identify parts of compounds, e.g. *art history* or *frying* *pan*. The dependency *fixed* helps to identify grammaticalized MWE, e.g. *in spite of.* Finally, the UD annotation has the dependency *flat*, which helps to identify complex proper names, such as *Angela Merkel.*[1]

3 Information-theoretic measures of semantic fit

For every lexeme, its actual and relative frequencies were computed in each of the four main syntactic roles: subject of an intransitive clause, subject of a transitive clause, object and oblique. Some examples are displayed in Table 2.

Lexeme	**Intransitive subject**	**Transitive subject**	**Object**	**Oblique**
hunter/NOUN	64	40	22	30
evening/NOUN	100	38	150	1145
street/NOUN	155	34	466	1331
t-shirt/NOUN	7	3	118	36

Table 2: A fragment of the lexeme – dependency matrix for English.

On the basis of these matrices, the Mutual Information (MI) scores were computed for each language. This metric represents the degree by which the relative frequencies of the syntactic roles performed by individual lexemes differ from the relative frequencies of these roles in the corpus. The formula for computing the measures based on a matrix of probabilities is given below.

$$I\,(Lex;\ Dep) = \sum\nolimits_{i,j} p\left(lex_i, dep_j\right) log\ \frac{p\left(lex_i, dep_j\right)}{p\left(lex_i\right) p\left(dep_j\right)}$$

where *Lex* stands for lexemes (lemmas) and *Dep* represents the four selected syntactic dependencies.

The greater this divergence, the more biased the lexemes on average towards a particular role, and therefore the tighter the fit between the lexemes and the syntactic dependencies. For instance, human nouns tend to be biased towards the role of intransitive and transitive subjects (e.g. *hunter*), inanimate objects frequently occur in the object role (e.g. *t-shirt*), whereas temporal and locative nouns (e.g. *evening, street*) are frequent in the oblique role.

4 Estimation of tight and loose fit

Figure 1 displays the MI scores in the thirty languages, based on lemmas only and on lemmas plus MWE. The correlation between these scores is high: $r = 0.929$ ($p < 0.001$). For English, Hindi, Indonesian, Japanese, Korean and Vietnamese, the scores based on lemmas plus MWE are higher than the scores based on lemmas only.

The English corpus has the lowest divergence. This means that on average the lexemes in that corpus are 'promiscuous' with regard to the roles. The other Germanic languages, from Swedish and German to Dutch and Danish have higher scores. The Romance languages are loose; they have relatively low scores, with Spanish being the loosest and Portuguese the tightest. Modern Greek and Bulgarian (the most analytic Slavic language) are loose, as well. The other Slavic languages have moderate scores, with Slovene being the tightest. The two Baltic languages (Latvian and Lithuanian) are on the loose-to-

[1] More information on multiword expressions in the UD can be found here: https://universaldependencies.org/u/overview/specific-syntax.html#multiword-expressions.

moderate side of the distribution. The three Uralic languages (Finnish, Estonian and Hungarian) have high scores, especially Finnish, which is among the tightest languages, together with Hindi and Korean. Hungarian is the loosest language of the Uralic languages.

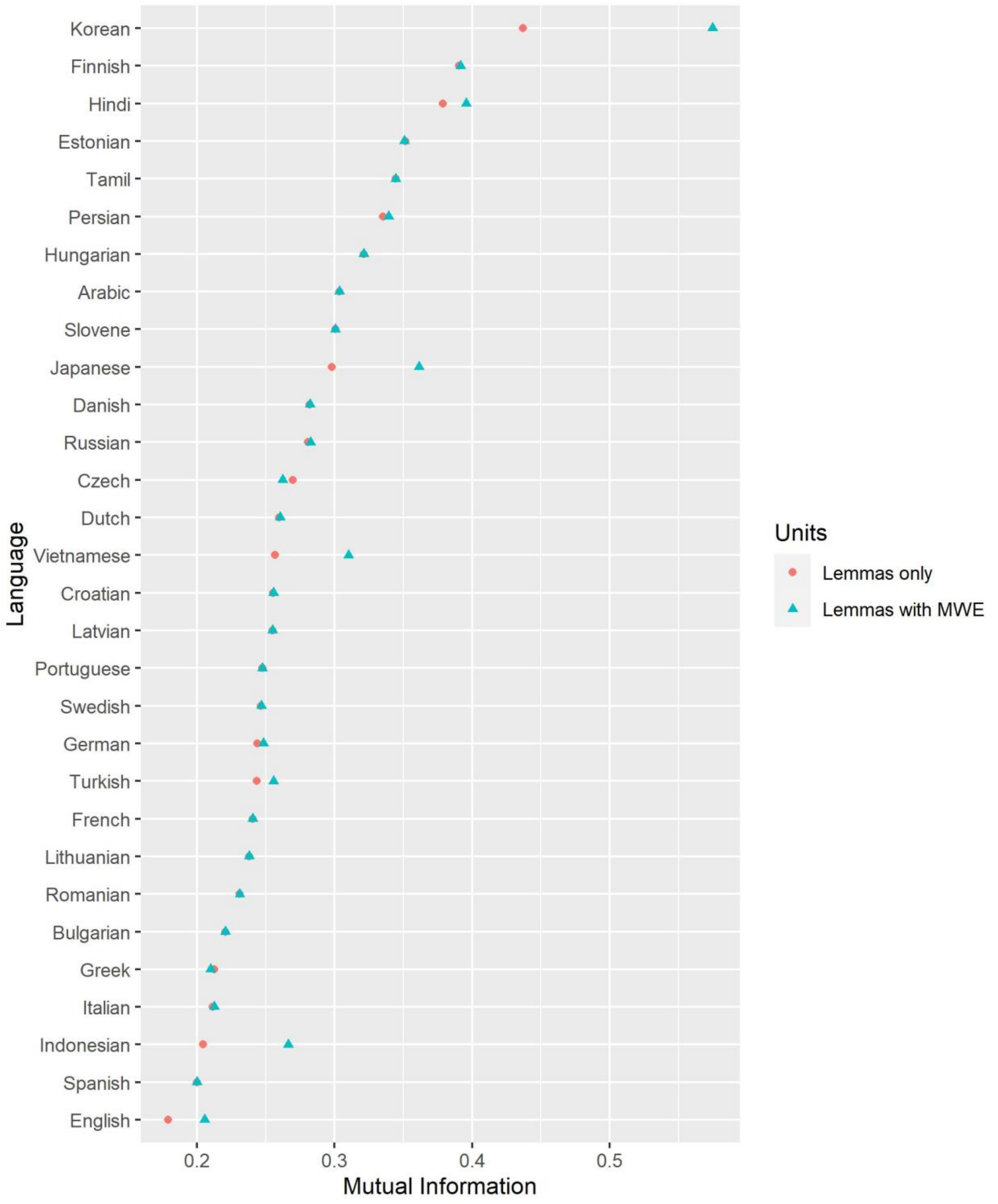


Figure 1: Mutual Information of lexemes and syntactic dependencies in 30 languages.

Overall, the previous observations about loose and tight languages are met. Among the languages represented in our sample, English has been evaluated in the literature as the most flexible, followed by Indonesian, and further by German, Japanese, Korean, Russian and Turkish (Hawkins (1986: 121– 127, 1995) and Müller-Gotama (1994). However, Indonesian is slightly tighter than German or Turkish, contrary to the previous reports (see Section 4), if we take into account MWE. This is also what we see in the data at the levels of lemmas. We also see that there is large variability within the languages that were considered tight, with German and Turkish having moderate scores and Korean having a very large score.

5 Correlation between tightness scores and word order

An important question is, how can we explain the cross-linguistic differences in tightness and looseness? There are substantial differences even among genetically related languages, so this factor does not seem to play an important role. A possible explanation may be related to processing constraints. If a language has the SVO order, the verb is accessed early. As a result, the addressee can use the semantic information in the verb to identify the roles of the other constituents in the clause (in particular, the thematic roles, such as Agent, Patient or Instrument). There is some experimental support of this claim. In particular, when asked to describe events in pantomime, people tend to avoid SOV in favour of SVO if the transitive event is reversible, that is, if each participant can be subject or object, e.g. “The mother hugs the boy” or “The boy hugs the mother” (Hall et al. 2014).

If a verb occurs in the end of the sentence, as in SOV languages, the thematic roles of nouns are more difficult to assign early. In order to mitigate the risk of incorrect interpretation of the frame and to avoid the costs of reanalysis, verb-final languages rely on semantic tightness of the arguments, as well as on case marking and other features of tight-fit languages (see Section 1). This is why, according to Hawkins (1995), the languages with verb-final structures (e.g. Japanese or German) exhibit greater predicate frame differentiation than languages like English.

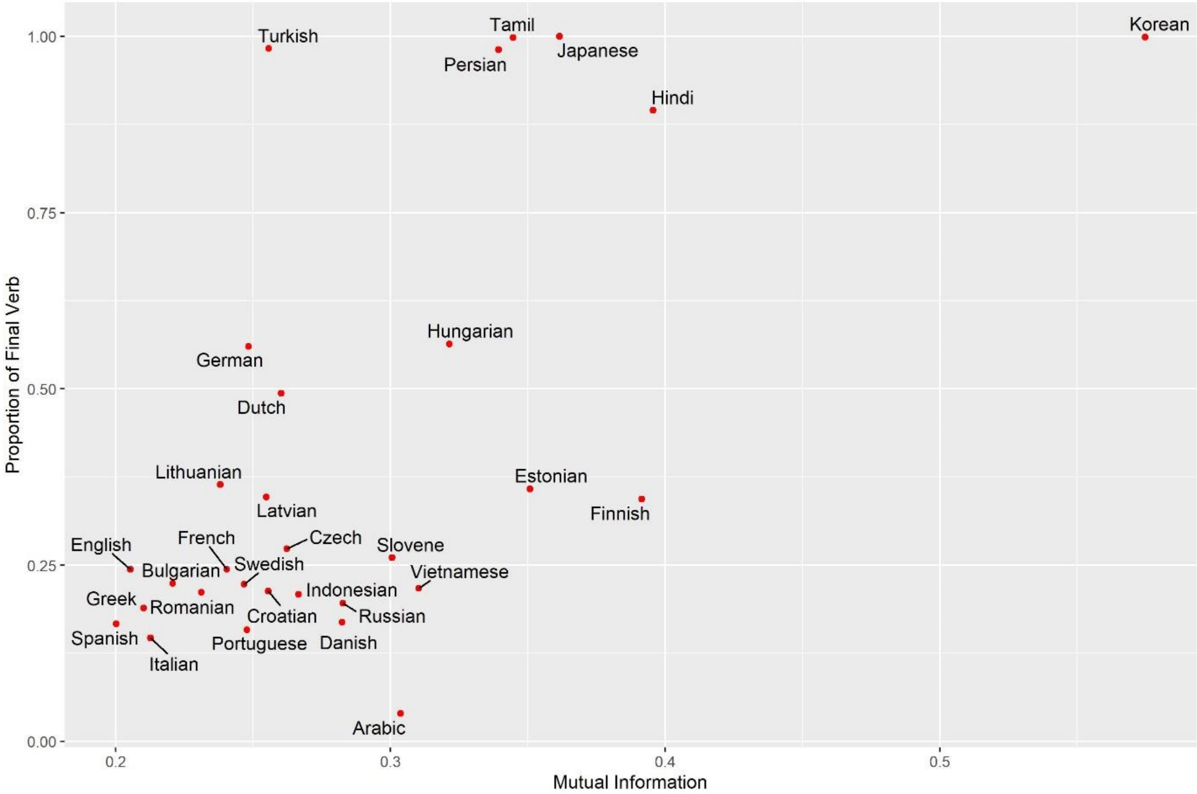


Figure 2: Mutual Information and proportion of verb-final sentences

This explanation, however, has not been systematically tested. In order to fill in this gap, we computed what we call a 'verb-finalness' score for each language. The procedure was as follows. We looked for all verbs with following dependencies: subject, object, oblique (with the exception of adverbs) and/or indirect object, where available. Each verb with at least one dependency from the list was counted as one frame. If a verb was used after all these dependent elements, then the frame was considered verb-final. The verb-finalness score was computed for each language by dividing the number of verb-final frames by the total number of frames. Arguments of nominal predicates were not taken into account.

Figure 2 displays the MI scores based on words and multiword expressions against the verb-finalness scores. The plot suggests that the correlation is positive. That is, the tighter a language, the more frequently the verb is final and therefore the more difficult it is to infer thematic roles from the start.

A Bayesian mixed-effect model with genera (see Table 1) as random intercepts, verb-finalness as the response variable and MI as the fixed effect shows that the effect of verb-finalness is positive, with the estimate *b* = 1.63 and the 95% credible interval between 0.06 and 3.19. This confirms our expectations. The Bayesian R^2 is 0.85, with the 95% credible interval between 0.66 and 0.93, which suggests a strong relationship between semantic tightness and verb-finalness.

If we take the divergence scores based on lemmas only, the effect of verb-finalness is slightly weaker (the estimate *b* = 1.58, with the 95% credible interval between -0.10 and 3.57). The credible interval is this time wider and includes zero, so we can be less confident in this result. The Bayesian R^2 is 0.85 again, with the 95% credible interval between 0.63 and 0.93.

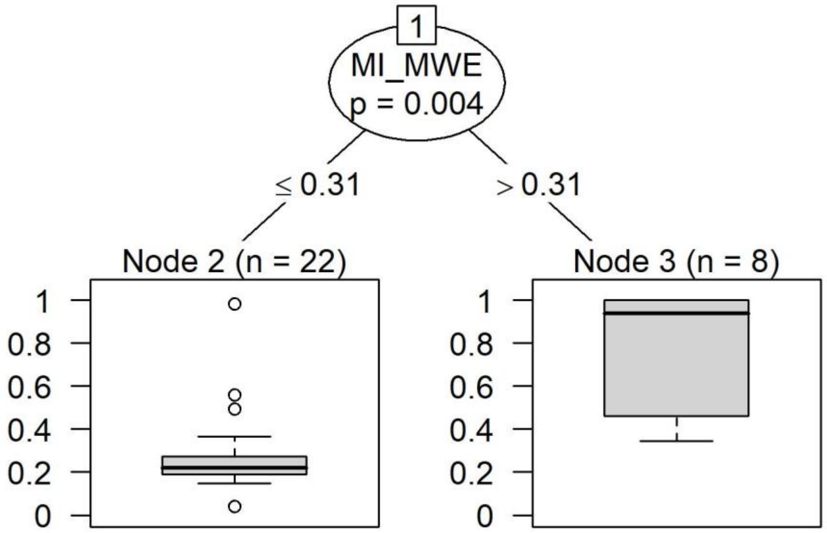


Figure 3: A conditional inference tree predicting verb-finalness

Since the data indicate a heteroscedastic relationship, with more variation in the MI scores as the verb-finalness scores increase, we also used a non-parametric method of conditional inference trees in order to make sure that our conclusions are valid. The method tests the null hypothesis of conditional independence of the response variable given a predictor. Conditional inference trees involve recursive binary partitioning of the data (Hothorn et al. 2006). The algorithm tries to identify the predictor that has the strongest association with the response variable and makes a binary split in that variable. After that, the procedure is repeated for each subset of the data until no further split can be made. In order to make a split, a set of criteria should be met, such as the level of significance at 0.05. Using this method, we can predict the verb-finalness scores (the response variable) from the two types of MI scores – based on lemmas and lemmas plus MWE. The genus was also tested.

Figure 3 displays the conditional inference tree model predicting verb-finalness. It shows that the MWE-based MI scores allow us to predict verb-finalness, and the other variables are not important. If MI is less than or equal to 0.31, then the word order is not likely to be verb-final, as shown by the box plot in Node 2. If MI is higher than 0.31, then we are likely to have a verb-final language. The genealogical factors do not play a significant role because they do not participate in any splits. Adding the family as a predictor does not change the results, either.

Therefore, there is a strong association between verb-finalness and MI. Also, taking into account composite nouns and other MWE leads to a stronger association between word order and semantic tightness.

6 Conclusions

In this paper we have demonstrated how one can use information about attraction between lexemes and syntactic roles (dependencies) measured with the help of Mutual Information for the purposes of language comparison. We have reproduced most of previous observations about languages with tight and loose fit between lexemes and arguments, and computed scores for many new languages. One should also be aware that the ranking changes somewhat depending on whether one takes single lemmas or also takes into account multiword expressions, which usually make the MI scores higher, and the language tighter. This is not surprising because composite nouns can be more semantically specific (e.g. *computer mouse* vs. *field mouse*) than simple lemmas (e.g. *mouse*) and therefore their syntactic behaviour can be more restricted.

The regression analysis also indicates that semantic tightness is associated with the final position of the verb. This relationship is more credible if the divergence scores take into account multiword expressions.

In the future, the results of this study should be tested on new data representing other registers and text types. One can expect substantial intra-linguistic variation. In addition, it would be interesting to investigate correlational and causal relations between tightness and other cues for understanding who did what to whom. One of the most important cues is case marking. As we can see in Figure 1, languages with low tightness scores tend to have fewer nominal cases than languages with high scores, although there are a few exceptions, such as Lithuanian, which has rich case morphology but loose fit between lexemes and dependencies. One should also consider verb agreement, which can help in identification of roles (cf. De Vogelaer 2007). Finally, it would be interesting to test word order entropy (Futrell et al. 2015; Levshina 2019), since rigid word order with low entropy can also be used as a cue for mapping the roles and participants. Other potential factors of interest are extralinguistic. For example, one can imagine tighter semantic relationships in languages with few speakers and closely knit communities, where the semantic restrictions can be easier to maintain and transfer, similar to other high-complexy features, and with few L2 speakers, who can have difficulties acquiring the semantic restrictions.

Acknowledgements

The research in this paper was funded by the Netherlands Organisation for Scientific Research (NWO) under Gravitation grant *Language in Interaction*, grant number 024.001.006. The author is grateful for insightful comments made by anonymous reviewers. All remaining mistakes are her own.

References

Gunther De Vogelaer. 2007. Extending Hawkins' comparative typology: Case, word order, and verb agreement in the Germanic languages. *Nordlyd,* 34 (special issue on Scandinavian Dialect Syntax), 167-182.

Richard Futrell, Kyle Mahowald, and Edward Gibson. 2015. Quantifying word order freedom in dependency corpora. *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, 91–100. Uppsala.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages. *Proceedings of the 8th International Language Resources and Evaluation (LREC'12)*, 2012.

Matthew L. Hall, Rachel Mayberry, and Victor S. Ferreira. 2013. Cognitive constraints on constituent order: evidence from elicited pantomime. *Cognition*, 129(1), 1-17. DOI https://doi.org/10.1016/j.cognition.2013.05.004

Martin Haspelmath. 2011. The indeterminacy of word segmentation and the nature of morphology and syntax. *Folia Linguistica*, 45(1): 31–80. DOI https://doi.org/10.1515/flin.2011.002.

John A. Hawkins. 1986. *A Comparative Typology of English and German: Unifying the Contrasts*. Croom-Helm, London.

John A. Hawkins. 1995. Argument-predicate structure in grammar and performance: A comparison of English and German. In Irmengard Rauch, and Gerald F. Carr (eds.), *Insights in Germanic Linguistics.* Vol. 1: Methodology in Transition, 127–144. Mouton de Gruyter, Berlin.

Torsten Hothorn, Kurt Hornik and Achim Zeileis. 2006. Unbiased Recursive Partitioning: A Conditional Inference Framework. *Journal of Computational and Graphical Statistics*, 15(3): 651--674.

Natalia Levshina. 2019. Token-based typology and word order entropy: A study based on Universal Dependencies. *Linguistic Typology,* 23(3): 533–572.

Natalia Levshina. Submitted. Efficient trade-offs as explanations in functional linguistics: some problems and an alternative proposal.

Franz Müller-Gotama. 1994. *Grammatical Relations: A Cross-Linguistic Perspective on Their Syntax and Semantics*. Mouton de Gruyter, Berlin.

Frans Plank. 1984. Verbs and objects in semantic agreement: Minor differences between English and German might that might suggest a major one. *Journal of Semantics*, 3(4): 305–360.

Milan Straka, and Jana Straková. 2017. Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, Vancouver, Canada, August 2017.

Jan Wijffels, Milan Straka, and Jana Straková. 2018. udpipe: Tokenization, parts of speech tagging, lemmatization and dependency parsing with the UDPipe NLP Toolkit. R package version 0.7. https://CRAN.R-project.org/package=udpipe.

Daniel Zeman, Joakim Nivre, Mitchell Abrams et al. 2020. Universal Dependencies 2.6, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University, http://hdl.handle.net/11234/1-3226. See also http://universaldependencies.org

Subjects tend to be coded only once: Corpus-based and grammar-based evidence for an efficiency-driven trade-off

Aleksandrs Berdicevskis
Språkbanken
University of Gothenburg
aleksandrs.berdicevskis@gu.se

Karsten Schmidtke-Bode
Department of English and American Studies
Friedrich Schiller University Jena
karsten.schmidtkebode@uni-jena.de

Ilja Seržant
Project "Grammatical Universals"
Leipzig University
ilja.serzants@uni-leipzig.de

Abstract

Using data from the *World Atlas of Language Structures* and the *Universal Dependencies* treebanks, we provide converging evidence from linguistic typology and comparative corpus linguistics for an efficiency-based trade-off in the encoding of referentially accessible subjects. Specifically, when familiar subjects are marked as bound elements attaching to the verb, the chances of having obligatory independent subject pronouns decrease significantly across the world's languages. At the same time, there is a trend against not encoding the subject at all, leading us to postulate an overall tendency to encode familiar subjects once and only once in a neutral topic-comment utterance. This tendency is mirrored in more fine-grained corpus data from Slavic: East Slavic languages, in contrast to the other members of the genus, have past forms without verbal subject encoding, and it is precisely with these (former participle) forms that the use of independent subject pronouns is significantly higher than with other, non-participial verb forms. By contrast, the occurrence of independent subject pronouns does not differ across various verb forms in other Slavic languages, as none of them has been affected by a loss of verbal subject encoding.

1 Introduction and background

Trade-offs are a particular type of cross-linguistic tendency, which can usually be described as "in languages where feature X is strongly expressed, feature Y tends to be absent or weakly expressed, and vice versa". Examples of trade-offs include, for instance, the following: rich morphological marking of core arguments is negatively correlated with rigidity of word order (Sinnemäki, 2014; Futrell et al., 2015; Levshina, 2019), and, more generally, rigidity of word order is negatively correlated with rich word structure (Koplenig et al., 2017); paradigm size is negatively correlated with irregularity (Cotterell et al., 2019); word length is negatively correlated with phonotactic complexity (Pimentel et al., 2020).

Trade-offs can be explained as adaptations to communicative efficiency (Gibson et al., 2019) and/or learnability (Kirby et al., 2015): some overt coding is necessary to convey information robustly, but redundancy is undesirable (Berdicevskis and Eckhoff, 2016; Fedzechkina et al., 2017). Thus, the identification and description of trade-offs is important for the ongoing discussion about the extent to which language structure is shaped by adaptive pressures (Schmidtke-Bode et al., 2019). However, even the current wealth of databases, corpora and computational tools does not make the identification of cross-linguistic generalizations a trivial task, in large part due to the danger of spurious correlations (Ladd et al., 2015). The most reliable way of identifying a cross-linguistic generalization is to demonstrate its presence through different approaches that rely on different kinds of data (Roberts, 2018), such as typological surveys, corpus-based studies, psycholinguistic experiments, diachronic investigations and computational modelling (Bickel et al., 2015; Blasi et al., 2019; Bentz and Berdicevskis, 2016).

In this paper, we take such an approach in order to adduce new empirical evidence for a conspicuous trade-off that has long been discussed in the typological literature, viz. the trade-off in the encoding of the subject. Subjects, especially those of transitive clauses, generally tend to be highly accessible referents (Du Bois, 2003; Du Bois, 1987; Siewierska, 2004). They are thus less likely to be encoded by full nominal phrases (NPs) than by *reduced referential devices* (Kibrik, 2011) such as independent pronouns and indexation (Ariel, 1990). We use *indexation* as an umbrella term for subject markers that are phonologically bound to the verb (i.e. verbal affixes and clitics) and index referential features of the subject (Haspelmath, 2013), typically person and number, but possibly also gender.

These different types of reduced referential devices are illustrated by the following examples:

(1) Norwegian Bokmål (independent subject pronoun)
Han sov.
'He slept.'

(2) Chalcatongo Mixtec (subject clitic):
Ni-éé=rí *staà.*[1]
CMPL-eat=1SG tortilla
'I ate.'

(3) Spanish (verbal affix):
ve-o
see-1SG.PRS
'I see'

Although languages vary as to the type of reduced referential device they conventionally employ, we argue that there is a strong trade-off pressure that constrains the typological distribution of the devices.

In particular, the trade-off is such that languages disprefer (a) doubling of reduced referential devices to encode the subject and (b) not encoding the subject at all. In other words, if — in information-structurally neutral clauses — a subject is encoded by a reduced referential device, there is a certain functional pressure to encode it once and only once in the clause.

This can be seen in the examples above, where the independent subject pronoun in (1) occurs with a verb that is itself unmarked for the subject, and the opposite situation holds in (2) and (3). What should be dispreferred and hence cross-linguistically rare according to our trade-off hypothesis is languages like German, which marks accessible subjects twice:

(4) German:
Wir lauf-en schnell.
we run.PRS-1PL fast
'We run/are running fast.'

Furthermore, the trade-off also predicts that the option of not encoding the subject at all, as in Chinese, is dispreferred, as it potentially engenders ambiguities in the unfolding discourse and thus requires additional processing effort ("hidden complexity" in Bisang (2015)) and risks a less accurate transfer of information. Although languages differ substantially in the degree to which they tolerate the omission of accessible referents in discourse (Bickel, 2003), our trade-off hypothesis leads us to assume that there is a cross-linguistic tendency against not encoding the subject, especially outside of closely tied syntactic units such as coordinate and subordinate clauses (Siewierska, 2004, p.22).Thus, despite the fact that the choice of the particular reduced referential device is subject to cross-linguistic variation, we suppose that languages tend to converge on optimizing the patterns by avoiding both redundancy (double encoding) and potential ambiguity (no encoding).

The complementary nature of independent subject pronouns and subject indexes has been a prominent feature in the discussion of the "null subject" (or "pro-drop") "parameter" in the formal-generative literature, where it was originally assumed that "only languages with rich verb agreement can license null subjects" (Taraldsen's (1980) generalization, as summarized in D'Alessandro (2015, p.219)). While

[1] CMPL = completive; example from Macaulay (1996, p. 141)

this generalization has been differentiated and refined by subsequent research (e.g. Rizzi (1982); Huang (1984); Müller (2006); Nicolis (2005); Roberts (2009)), its underlying premise is usually that the optionality of subject pronouns is viewed as a variable feature of innate linguistic representations ("Universal Grammar"). In the present paper, by contrast, we see it as a usage-based phenomenon that results from the strive for efficient communication.

Against this background, we probe the indexation vs pronoun trade-off by two complementary approaches. First, using the *World Atlas of Language Structures online* (Dryer and Haspelmath, 2013), we conduct a typological analysis on a broad sample of languages (Section 2). The survey shows that there is indeed a correlation between the presence of indexation, on the one hand, and the optionality of independent pronouns, on the other, and vice versa. This global correlation, per se, however, is not enough to establish a causal link between the two grammatical phenomena.

In a second study (Section 3), we thus perform a more specific, finer-grained corpus study on Slavic languages, using *Universal Dependencies* treebanks (Zeman et al., 2020). After all, in actual discourse, the two phenomena are not binary, but gradual: the proportion of sentences without an independent pronominal subject or in which the subject is indexed on the verb can vary greatly, so that the potential values are not limited to 0 and 1. Corpus-based approaches allow us to capture this variation (Levshina, 2019). We show that there is a split between East Slavic languages, on the one hand, and other modern Slavic languages, on the other. East Slavic languages have a number of constructions (most saliently, past tense) where the verb form (historically a participle) does not allow indexation. We show that in East Slavic, independent pronominal subjects are more frequent in these constructions than in those where indexation is possible, and that they are more frequent in East Slavic than in other Slavic languages (both in participial past-tense constructions and overall).

The results of the two studies, combined with previous quantitative work on other languages, provide strong evidence in favour of our hypothesized trade-off. Before we begin with the analyses, it needs to be pointed out that our prediction is limited to reduced referential devices; we do not make any predictions about full referential devices (NPs), since these fulfill a very different function in discourse.

2 Typological evidence

Our typological approach is similar in spirit to Gilligan's (1987) seminal investigation in being based on a sample of the world's languages, but it draws on a much larger and more contemporary database as well as completely different analytical tools. Specifically, we use a sample of of 241 languages for which data from two *WALS* chapters are available simultaneously, namely Dryer (2013) and Siewierska (2013). Dryer surveys the preferred expression of pronominal subjects, while Siewierska's chapter is concerned with the presence of verbal person marking. These surveys follow intricate coding schemes that come with the usual challenges of (i) reducing the variation range of human languages to a handful of types (see, e.g., Holmberg (2017) for a critique of some of Dryer's categories) and (ii) coding grammatical features whose presence of absence is variable rather than consistent in many languages (e.g. differential indexation). On top of that, the two coding schemes need to be both harmonized and further reduced for our purposes, as we are not interested in all the different types of subject expression and person marking, but in the general pattern of their interaction. Being aware of the risks that are harboured by such further interference with the data, we have opted to recode the two data sets as described in Table 1.

The decision to treat Dryer's "subject clitics" in the same fashion as his subject affixes on verbs is in line with our earlier definition of *indexation*, and this is also reflected by the fact that Siewierska analyzes them as verbal person markers (as long as one of their potential host words is actually the verb). Note that we left out Dryer's "mixed" category.

The 241 languages in this sample come from all six macro areas distinguished in *WALS*, spanning 100 language families (e.g. "Indo-European") and 179 lower genetic groupings called genera (e.g. "Germanic"). Since we thus have multiple data points for at least some of the language families and genera

[2] We understand Dryers category "subject pronouns in other position" to mean that these pronouns are obligatory. Note that this category in Dryer's coding may occasionally have been taken to instantiate subject clitics by Siewierska. For this reason, we also run an alternative model of the data in which this category is removed from the analysis (see footnote 4 below).

Feature	**Variable name in our model**	**Our coding scheme**	**Conflates the original *WALS* categories of ...**
Expression of pronominal subjects (Dryer, 2013)	Subject pronouns	Obligatory	obligatory pronouns in subject position, subject pronouns in other positions[2]
		Optional	subject affixes on verb, subject clitics, optional pronouns
Verbal person marking (Siewierska, 2013)	Subject indexation	Present	A&P, A only, A or P
		Absent	P only, No verbal person marking

Table 1: Coding of the typological data

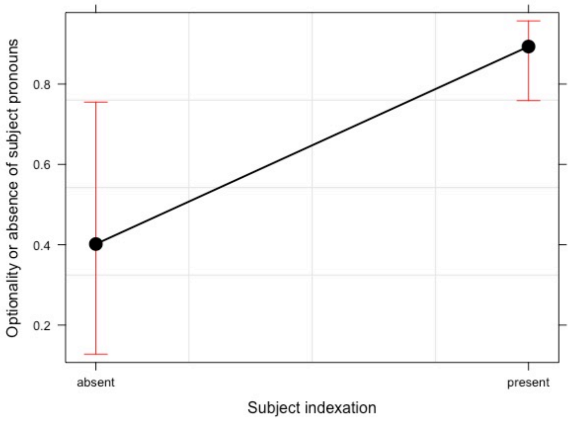


Figure 1: Visual representation of the effect of indexation on the optionality of subject pronouns in our model

in the sample, we use mixed-effects logistic regression modelling to take these dependencies into account. Specifically, we model the probability of having optional (or no) subject pronouns as a function of indexation, but control for repeated measurements. First, by modelling random intercepts for FAMILY and GENUS, we allow the genealogical units to have different baseline preferences for independent subject pronouns. Second, following arguments by Dryer (1989) and Bickel (2013), we assume that the most robust signals for universals come from those effects which take the same directionality across all of the world's macro areas. For this reason, we also include in our model by-AREA random slopes for the hypothesized effect of person marking on the occurrence of subject pronouns. We thus arrive at the following model formula[3]:

$$PronSubj \sim Index + (1|Family) + (1|Genus) + (1 + Index|Area)$$

The results of the modelling process show that the fixed effect of subject indexation on the absence of obligatory subject pronouns is significant ($\beta = 2.52$, $z = 2.3$, $p = 0.021$): on average, the odds for optional subject pronouns ("pro-drop") increase by 12.47 when we go from "absent" to "present" verbal person marking (Figure 1, which shows actual probabilities rather than odds).

[3] All statistical analyses were performed in *R* 3.3.0 (R Core Team, 2016), using the packages `lme4` (Bates et al., 2015), `effects` (Fox and Hong, 2009) and `rms` (Harrell Jr, 2020).

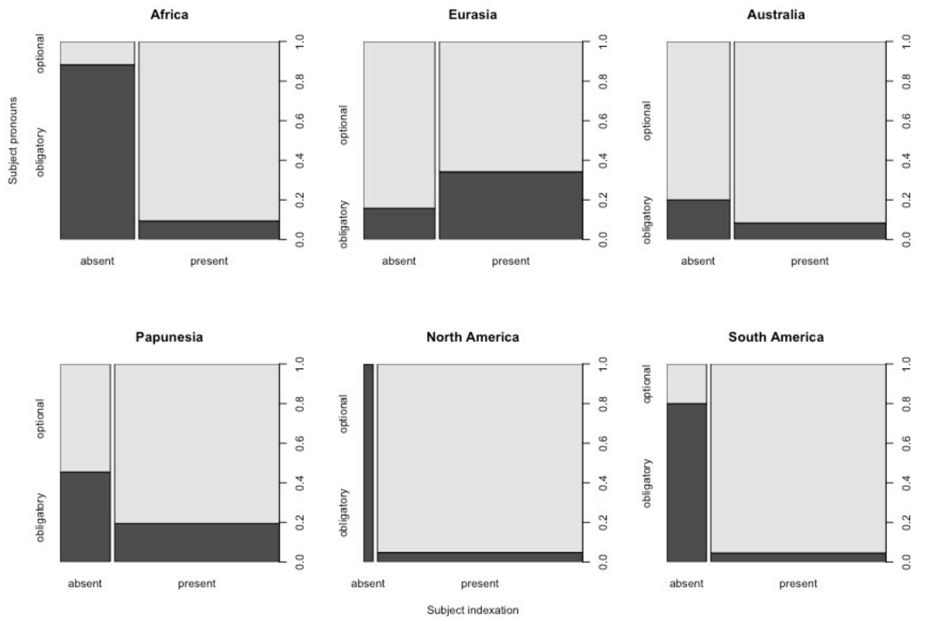


Figure 2: Areal distribution of the effect of indexation on the optionality of subject pronouns in the *raw* data (i.e. uncontrolled for genealogical dependencies)

Our model shows no traces of overdispersion and achieves very good discrimination (C = 0.88, Somers' D_{xy} = 0.75). In this respect, there appears to be robust cross-linguistic evidence for a trade-off between independent subject pronouns and subject indexation, indeed.

However, it must be conceded that our model is considerably "stressed" when it comes to the random effects structure relating to AREA, where it shows large standard deviations for both the intercept (*sd* = 1.56) and the slope adjustments (*sd* = 2.33). If we inspect the raw data again for the different macro areas (Figure 2), we can see that this problem mainly stems from the fact that one large and diverse area – Eurasia – goes against the otherwise consistent cross-linguistic trend, i.e. it reverses our hypothesized effect of indexation on subject pronouns. This is due to languages like French, German or Georgian on the one hand, which have obligatory subject pronouns despite verbal person marking (see Seržant (forthcoming a; forthcoming b) about this phenomenon), and to certain North and particularly South East Asian languages (e.g. Nivkh, Japanese, Korean, Chinese, Vietnamese, Burmese) which allow optional subject pronouns despite the lack of verbal person marking (for the latter, see e.g. Bisang (2014) for a diachronic account).[4]

The typological data also provide evidence for a general trend against having the option of not encoding the subject at all. We can illustrate this clearly again by juxtaposing how often the combination "no indexation and optional subject pronouns" (= no encoding occurs in discourse) is found with all other combinations where some subject encoding is always present. As can be seen in Table 2, the language families with at least one member of the no-encoding type are in the clear minority in all macro areas except again for Eurasia.

In conclusion, there is significant but not entirely consistent typological evidence for both aspects of the trade-off concerned here. In the corpus study to follow, we show, however, that our hypothesized pressures on subject encoding can be reliably detected even within Eurasia, and even in a genus which is

[4] If we run a mixed-effects regression model solely on the Eurasian languages in our sample (with random intercepts for FAMILY and GENUS), there is indeed an effect in the opposite direction from our trade-off hypothesis (i.e. the odds for obligatory subject pronouns decrease with the presence of verbal indexes), but it is not significant (p = 0.21 if the model does not contain by-FAMILY random slopes for the effect, and p = 0.33 if it does). If we run the same model as above but without Dryer's category "subject pronouns in other position", the model actually improves in the sense that the effect of indexation becomes even more robust (β = 4.112, z = 2.45, p = 0.014), as the distribution is even clearer in some of the macro areas and less pronouncedly reversed in Eurasia.

Macro area	**No. of sample languages with no encoding**	**No. of sample families with no encoding**
Africa	2/49	1/12
Eurasia	16/41	8/20
Australia	4/13	2/12
Papunesia	6/41	3/11
North America	0/44	0/27
South America	1/26	1/21

Table 2: Sample languages and families with no encoding of subject

mixed with regard to the possibility of not using independent subject pronouns. Specifically, we turn to the Slavic genus to buttress this claim.

3 Corpus-based evidence from Slavic

Indexation in modern East Slavic languages (Russian, Ukrainian, Belarusian) is different in past and non-past tenses. In the non-past, verb forms fusionally mark the subject's person and number; in the past, they mark number and, in the singular, also gender, but not person. The reason is that the modern past tense was originally the analytic perfect, consisting of a so-called *l*-participle and an optional copula. In East Slavic languages, the copula was gradually lost, and the participle was reanalyzed as a finite form. In all other Slavic languages, the copula was retained (in Polish, it has become a bound morpheme on the main verb), and it unambiguously marks the person of the subject.

If the trade-off described in Section 1 exists, we would expect that, in modern East Slavic languages, a subject would more often be encoded by an independent pronoun when the verb is in the past tense (or any other construction that is based on the *l*-participle, e.g. conditional) than in the tense-mood combinations that index person. Note that our decision to treat non-copular *l*-participle-based constructions as non-indexing is a simplification. While they do not index person, they do index number and gender, which can also aid the hearer's interpretation of subject reference. Nonetheless, it seems reasonable to believe that person marking provides much more information, and thus its absence should yield some effect, even if mitigated by the presence of number and gender.

In other Slavic languages, where person is always marked, either on an obligatory copula or a clitic, there should be no such difference between the *l*-participle and other tense-mood combinations. On the other hand, we may expect that in East Slavic, subjects of the verbs in *l*-participle-based constructions will be more often encoded by independent pronouns than in West and South Slavic.

Thus, we are effectively testing whether a typological generalization holds *within* individual languages. This enables us to perform a more direct test of the following causal relationship in East Slavic: the loss of indexation leads to the more frequent use of overt pronominal subjects. This is in line with what has been hypothesized about Slavic by Kibrik (2011). An opposing view had also been expressed in previous work (Ivanov, 1982; Zaliznjak, 2004), namely that subject pronouns expanded beyond their original limited usage, making person-marked copulas redundant and causing their gradual elimination. Note that under the latter view, however, there is no reason to expect differences either between *l*-participle and other forms in East Slavic or between East Slavic and other subgroups (see more in Section 4).

To test our prediction, we take all Slavic treebanks that are available in *Universal Dependencies* (UD) 2.6 (Zeman et al., 2020): Russian, Ukrainian, Belarusian (East); Czech, Polish, Slovak (West)[5]; Bulgarian, Slovenian, Serbian and Croatian (South). When several treebanks are available for one language, we concatenate them all. The resulting treebank sizes are very different across languages (from 13K tokens for Belarusian to 2.3M for Czech), but that is not a problem for our approach.

We take all finite verbs whose lemmas occur at least once in an *l*-participle-based construction (e.g.

[5] We include Upper Sorbian as well, but the treebank is too small and yields no datapoints that pass our filters.

	rus(E)	bel(E)	ukr(E)	pol(W)	cze(W)	slk(W)	crt(S)	srb(S)	blg(S)	slv(S)
l-participles	0.59	0.52	0.61	0.96	0.97	0.97	0.95	0.86	0.81	0.96
other forms	0.67	0.82	0.73	0.93	0.97	0.94	0.95	0.96	0.88	0.96

Table 3: Proportion of clauses without a free pronominal subject across **E**ast, **W**est and **S**outh Slavic.

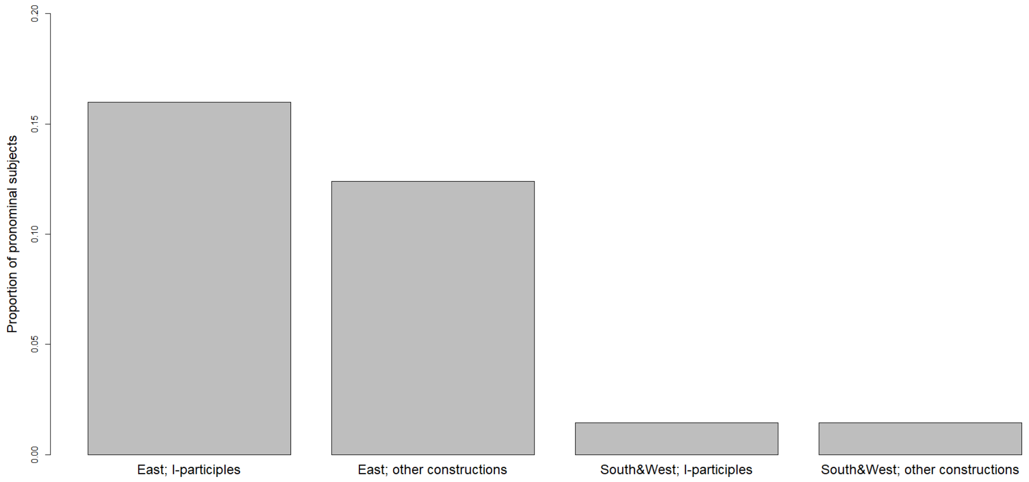


Figure 3: Proportion of clauses with an independent pronominal subject in the *raw* data (i.e. uncontrolled for language-specific and verb-specific effects)

past tense in East Slavic) and at least once in any other tense-mood combination. This filter is meant to exclude lemmas that occur only in a particular form. In addition, we require that the lemma occurs at least once with an independent nominal subject (related to the verb via NSUBJ). Here, we do not impose any additional limitations on the part-of-speech of the subject (it can be pronoun, noun or adjective), since the sole purpose of this filter is to exclude impersonal verbs (which never take a subject). For every verb token that passes these filters, we note whether it has an independent subject encoded by a personal pronoun or not. Since our trade-off hypothesis is limited to reduced referential devices, we do not include subjects encoded by NPs or anything else apart from personal pronouns (such clauses are ignored).

It could be argued that the analysis has to be narrowed down to first- and second-person pronouns, since in these cases the choice the speaker makes is clearly between a pronoun and its absence, while in the third person the choice is between an NP, a pronoun and absence of both. Furthermore, first- and second-person pronouns denote the referent much less ambiguously than the third-person pronouns. However, there is no way to automatically determine person in those East Slavic clauses where the verb is in an *l*-participle-based construction and the subject is not encoded by an independent pronoun (i.e. the person is not marked anywhere). For this reason, we do not choose the “first- and second-person only” design.

Our search for subjects does not extend beyond the clause boundaries. If the clause is coordinate or subordinate, we treat it irrespectively of what happens in other conjuncts or the main clause. In other words, in *She sings and walks* the verb *sing* would be treated as having a pronominal subject while the verb *walk* would not. The same is true for **She sings when walks* (which is a possible construction in Slavic).

We do not perform any analysis of the individual languages, but for illustrative purposes we provide information on the proportion of clauses without free pronominal subjects in different constructions across all languages in Table 3. Note that unlike all other languages, Bulgarian and Serbian do not follow the prediction, behaving rather like East Slavic languages.

The proportion of clauses with an independent pronominal subject across language groups and con-

Predictor	**Estimate**	**SE**	***z* value**	***p* value**
(Intercept)	-0.62	0.27	-2.3	0.020*
constr=person-marking	-0.38	0.03	-14.5	<0.001*
group=West&South	-2.58	0.32	-8.1	<0.001*
constr=person-marking x group=West&South	0.44	0.04	10.2	<0.001*

Table 4: Summary of the logistic-regression model: presence of a pronominal subject as predicted by construction and language group with by-VERB and by-LANGUAGE random effects. Asterisks denote significance at the 0.05 level.

struction types is visualized in Figure 3 (see also Table 3). The observed differences are in line with our expectations. To test whether they are significant, we fit a mixed-effects logistic regression model with the binary dependent variable being whether the subject is encoded by an independent pronoun. The predictors are CONSTRUCTION, or tense-mood combination (*l*-participle vs person-marked), GROUP (East vs West&South), and their interaction. We add by-VERB and by-LANGUAGE random intercepts in order to control for language-specific idiosyncrasies and individual lexical preferences of verbs. In *lme4* (Bates et al., 2015) notation, the model looks as follows:

$$PronSubj \sim construction * group + (1|verb) + (1|language)$$

The coefficient for CONSTRUCTION shows how frequently we find pronominal encoding of the subject in East Slavic in person-marking constructions (as opposed to *l*-participles), and we expect it to be negative. The coefficient for GROUP is meant to capture how frequently we find pronominal encoding of the subject in *l*-participle-based constructions in West and South Slavic (as opposed to East Slavic), and again, we expect it to be negative. We do not make specific predictions about the interaction, but we do not expect it to revert the individual effects.

We performed the calculations in *R* 4.0.2 (R Core Team, 2020), using the `lmerTest` package (Kuznetsova et al., 2017) to calculate *p*-values and `Hmisc` (Harrell Jr and others, 2020) to estimate discrimination. The total number of observations is 138,879; the number of unique verb lemmas is 9,363. The summary of the model is presented in Table 4.

It can be seen that all of our predictions are borne out. Within East Slavic languages, independent pronominal subjects are significantly more frequent in *l*-participle-based constructions. Within *l*-participle-based constructions, independent pronominal subjects are significantly more frequent in East Slavic languages than in other groups (cf. Seo (2001)). The interaction coefficient is positive, indicating that the joint effect is smaller than could be expected from individual coefficients. However, the interaction coefficient is noticeably smaller than the sum of the individual coefficients, suggesting that the joint effect is still significantly different from zero. The model shows very good discrimination (C = 0.88, Somers' D_{xy} = 0.76).

It is reasonable to expect that the means of encoding subject will strongly vary across clause types (simple sentence, subordinate, superordinate, coordinate), as we alluded to above. However, adding CLAUSE TYPE as a predictor leads to severe convergence problems, rendering the models unusable. Instead, we opt for a simpler way of controlling for a potential effect of CLAUSE TYPE. We fit a separate model with the same specification, but use only clauses from simple sentences (i.e. excluding all clauses from complex sentences: subordinate, superordinate and coordinate). The model yields similar results (see Supplementary materials).

Just like the typological data, the corpus data also provide evidence against not encoding subjects at all. We illustrate this by visualizing the number of clauses with no encoding, double encoding or one of two possible single encodings across all Slavic languages in our sample (Figure 4). Note also that "No encoding" column means "no person marking", while there still is gender and number marking.

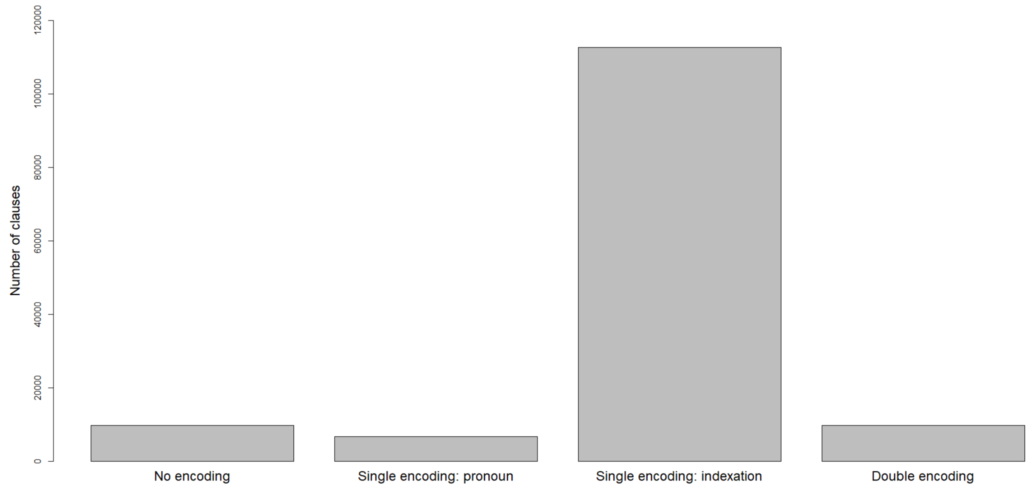


Figure 4: Distribution of subject-encoding strategies across all Slavic languages

4 Discussion

We have shown that grammar-based typological data and corpus-based data from UD provide converging evidence for our hypothesized trade-off between independent pronominal subjects and indexation. This, in turn, supports our claim that languages prefer to encode accessible subject referents once and only once within a clause. The typological study shows that the presence of subject indexation significantly increases the optionality of subject pronouns. The trend, however is not exceptionless: it is reversed in Eurasia, by an unusual confluence of double-encoding languages (e.g. German, Standard French, Georgian) and no-encoding languages (e.g. Nivkh, Japanese, Chinese). The UD study provides more direct evidence that the absence of indexation encourages speakers to encode accessible subject referents by independent pronouns significantly more often. It thus suggests, in keeping with the original *WALS* data, that there is also a certain pressure against not encoding the subject.

This observation raises an important question about the directionality of change: does the loss/emergence of indexation make pronouns more or less obligatory or is it the other way around (Kibrik, 2004)? Our evidence suggests that the causal path "loss of indexation → higher proportion of pronouns" is present in Slavic. Nonetheless, we cannot discard earlier claims to the opposite effect, viz. "higher proportion of pronouns → loss of indexation" (Ivanov, 1982; Zaliznjak, 2004), since we provide no evidence either for or against it. It may be that both causal paths exist (both within Slavic and universally).

Simonenko et al. (2019) addressed the question of directionality, performing a diachronic study on Medieval French. They show that the syncretization of verbal endings (which is presumably phonologically-driven and eventually leads to the near-disappearance of indexation) occurs at almost the same rates as the spread of pronominal subject encoding, which suggests that the two processes are likely to be related. They compare two models: in the first one, indexation and absence of independent subject pronouns are manifestations of the same grammatical property (it is unclear how this model can be interpreted outside the generative framework); in the second one, not encoding the subject is assumed to create a parsing difficulty and is thus dispreferred by language learners. The second model is shown to have a higher fitness, which is interpreted as evidence in favour of the causal link "loss of indexation → higher proportion of pronouns". That said, it should be noted that Simonenko et al.'s study is deeply rooted in generative assumptions, in particular, the constant rate hypothesis (Kroch, 1989), and is difficult to evaluate outside of this framework. Furthermore, the authors did not explicitly test the reverse causal path.

Our data support the earlier observation that pronominal subjects are more frequent in East Slavic than in other subgroups (Seo, 2001), not only with *l*-participle-based constructions, but in all clauses. This is in keeping with the common view that East Slavic languages are gradually undergoing a development towards double encoding with both obligatory independent pronouns and verbal affixes, while most other modern Slavic languages remain to be single-encoding languages. It seems reasonable to assume that this is explained by analogical extension: pronominal subjects are spreading from *l*-participle-based constructions to other clauses (Kibrik, 2004).

A promising research avenue would be a quantitative diachronic investigation of the loss of the copula and the spread of the subject pronoun in Slavic. Our pilot study, however, suggests that the current treebanks of older stages of Slavic languages might not be large enough to yield robust results.

Note also that the trade-off is not absolute within Slavic (see Figure 4), as there are clauses where the subject is not encoded at all (*l*-participles without a pronoun in East Slavic languages) and clauses with double encoding (other forms with a pronoun). The non-negligible proportion of counterexamples both within and across languages suggests that the pressure for the optimization of subject encoding is relatively weak, and probably moderated by other factors. Double encoding of the subject, for instance, is normally reserved for the rare, pragmatically marked clause types (e.g. marked-focus or topic-shift subjects, cf. English *Me, I like booze* or *Him, he's crazy*[6]). Historically, such double encoding can spread to topic-comment clauses via the overuse of what originally used to be a pragmatically marked information structure, as described in detail in Givón (1976) (see also Ariel (2000)). This is, for example, what happened in the development from Old High German with its optional free pronouns into Modern German with obligatory free pronouns (Axel and Weiß, 2011). Double-encoding systems often turn into single-coding systems by abandoning the indexation, as in English or French (Siewierska, 2004, p.295). Absence of encoding, in turn, sometimes arises as a by-product of the emergence of new verbal forms based on nominalized structures which are not amenable to person marking themselves, such as Slavic *l*-participles. And as noted in Sections 1 and 2, typologically dispreferred structures may still spread locally to form areal phenomena, such as not encoding familiar subjects in the languages of South and Southeast mainland Asia (notably the Sinitic subfamily (Sino-Tibetan), the Mon-Khmer subfamily (Austroasiatic), Tai (Tai-Kadai), Hmong-Mien and Chamic (Austronesian), see Bisang (2014) for a historical account).

As was laid out at the beginning, we see the motivation behind the single-encoding pressure in the strive for efficient communication that equilibrates production effort and the robustness of information transfer (cf. also Jaeger and Buz (2017)). No encoding of a central discourse referent potentially jeopardizes the accurate transmission of messages. But double encoding is obviously redundant in pragmatically unmarked topic-comment clauses and is therefore costlier than necessary. In this respect, our findings confirm Haspelmath's form-frequency correspondence universal, according to which languages *generally* tend to have shorter forms for more frequent meanings (see Haspelmath's (2008a; 2008b) generalization across various earlier proposals for coding efficiency in the lexicon, e.g. Zipf (1935), and several domains of grammar (Greenberg (1966); Comrie (1989); Hawkins (2004))).

From this perspective, the frequent double encoding of pragmatically marked subjects (focal subjects, topic-shift subjects, etc.) is also explained: since pragmatically marked subjects are considerably less frequent in discourse than the pragmatically unmarked continuous-topic subjects (Givón, 1992), it is the pragmatically marked subjects that tend to select double encoding, which is costlier than the single encoding of topical subjects.

On a methodological level, our paper illustrates how coarse-grained but broad typological data and more fine-grained but narrower corpus data can fruitfully complement each other. From a technical point of view, it has become obvious to us, however, that the current UD annotation is still far from being fully harmonized. For instance, plural pronouns like 'we' are annotated as 'I'-PL in Slovenian and Upper Sorbian (probably due to the presence of dual forms), while in other languages, 'I' and 'we' are treated as separate lemmas. While such discrepancies are understandable and in certain ways beneficial, they may become a hindrance for cross-linguistic comparison, especially if not thoroughly documented.

The Supplementary materials, including scripts for extracting the data and running the statistical analysis, are openly available[7].

[6] Examples from Rodman (1997, p. 53).

[7] https://github.com/AleksandrsBerdicevskis/subject-encoding

Acknowledgements

KSB and IS were supported by European Research Council (ERC) Advanced Grant 670985, which is gratefully acknowledged.

References

Mira Ariel. 1990. *Accessing noun-phrase antecedents*. Routledge.

Mira Ariel. 2000. The development of person agreement markers: From pronouns to higher accessibility markers. In Michael Barlow and Suzanne Kemmer, editors, *Usage-based models of language*, pages 197–260. CSLI Publications.

Katrin Axel and Helmut Weiß. 2011. Pro-drop in the history of German from Old High German to the modern dialects. In Melani Wratil and Peter Gallmann, editors, *Null pronouns*, pages 21–52. John Benjamins.

Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2015. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software, Articles*, 67(1):1–48.

Christian Bentz and Aleksandrs Berdicevskis. 2016. Learning pressures reduce morphological complexity: Linking corpus, computational and experimental evidence. In *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)*, pages 222–232, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Aleksandrs Berdicevskis and Hanne Eckhoff. 2016. Redundant features are less likely to survive: Empirical evidence from the Slavic languages. In S.G. Roberts, C. Cuskley, L. McCrohon, L. Barceló-Coblijn, O. Fehér, and T. Verhoef, editors, *The Evolution of language: Proceedings of the 11th International Conference (EVOLANGX11)*. Online at `http://evolang.org/neworleans/papers/85.html`.

Balthasar Bickel, Alena Witzlack-Makarevich, Kamal K. Choudhary, Matthias Schlesewsky, and Ina Bornkessel-Schlesewsky. 2015. The neurophysiology of language processing shapes the evolution of grammar: Evidence from case marking. *PLOS ONE*, 10(8):1–22, 08.

Balthasar Bickel. 2003. Referential density in discourse and syntactic typology. *Language*, 79(4):708–736.

Balthasar Bickel. 2013. Distributional biases in language families. In Balthasar Bickel, Lenore A Grenoble, David A Peterson, and Alan Timberlake, editors, *Language typology and historical contingency*, pages 415–444. John Benjamins.

Walter Bisang. 2014. On the strength of morphological paradigms. In Martine Robbeets and Walter Bisang, editors, *Paradigm change: In the Transeurasian languages and beyond*, pages 23–60. John Benjamins.

Walter Bisang. 2015. Hidden complexity–the neglected side of complexity and its implications. *Linguistics Vanguard*, 1(1):177–187.

Damian Blasi, Steven Moran, Scott Moisik, Paul Widmer, Dan Dediu, and Balthasar Bickel. 2019. Human sound systems are shaped by post-neolithic changes in bite configuration. *Science*, 363(6432).

Bernard Comrie. 1989. *Language universals and linguistic typology: Syntax and morphology*. University of Chicago Press.

Ryan Cotterell, Christo Kirov, Mans Hulden, and Jason Eisner. 2019. On the complexity and typology of inflectional morphological systems. *Transactions of the Association for Computational Linguistics*, 7:327–342.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Matthew S Dryer. 1989. Large linguistic areas and language sampling. *Studies in Language*, 13(2):257–292.

Matthew S Dryer. 2013. Expression of pronominal subjects. In Matthew S Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

John Du Bois. 1987. The discourse basis of ergativity. *Language*, 63(4):805–855.

John Du Bois. 2003. Argument structure: Grammar in use. In Lorraine Kumpf and William Ashby, editors, *Preferred argument structure: Grammar as architecture for function*, pages 11–60. John Benjamins.

Roberta dAlessandro. 2015. Null subjects. In Antonio Fábregas, Jaume Mateu, and Michael Putnam, editors, *Contemporary linguistic parameters*, pages 201–226. Bloomsbury London.

Maryia Fedzechkina, Elissa L. Newport, and T Florian Jaeger. 2017. Balancing effort and information transmission during language acquisition: Evidence from word order and case marking. *Cognitive Science*, 41(2):416–446.

John Fox and Jangman Hong. 2009. Effect displays in R for multinomial and proportional-odds logit models: Extensions to the effects package. *Journal of Statistical Software*, 32(1):1–24.

Richard Futrell, Kyle Mahowald, and Edward Gibson. 2015. Quantifying word order freedom in dependency corpora. In *Proceedings of the third international conference on dependency linguistics (Depling 2015)*, pages 91–100.

Edward Gibson, Richard Futrell, Steven P Piantadosi, Isabelle Dautriche, Kyle Mahowald, Leon Bergen, and Roger Levy. 2019. How efficiency shapes human language. *Trends in cognitive sciences*, 23(5):389–407.

Gary Gilligan. 1987. *A cross-linguistic approach to the pro-drop parameter*. Ph.D. thesis, University of California.

Talmy Givón. 1992. The grammar of referential coherence as mental processing instructions. *Linguistics*, 30(1):5–55.

Talmy Givón. 1976. Topic, pronoun, and grammatical agreement. In Charles N. Li, editor, *Subject and topic*, pages 149–188. Academic Press.

Joseph H Greenberg. 1966. *Language universals: With special reference to feature hierarchies*. Mouton.

Frank E Harrell Jr et al., 2020. *hmisc: Harrell Miscellaneous*. R package version 4.4-1.

Frank E Harrell Jr, 2020. *rms: Regression modeling strategies*. R package version 6.0-1.

Martin Haspelmath. 2008a. Creating economical morphosyntactic patterns in language change. In Jeff Good, editor, *Linguistic universals and language change*, pages 185–214. Oxford University Press.

Martin Haspelmath. 2008b. A frequentist explanation of some universals of reflexive marking. *Linguistic Discovery*, 6(1):40–63.

Martin Haspelmath. 2013. Argument indexing: a conceptual framework for the syntax of bound person forms. In Dik Bakker and Martin Haspelmath, editors, *Languages across boundaries: Studies in memory of Anna Siewierska*, pages 209–238. De Gruyter Mouton.

John A Hawkins. 2004. *Efficiency and complexity in grammars*. Oxford University Press.

Anders Holmberg. 2017. Linguistic typology. In Ian Roberts, editor, *The Oxford handbook of Universal Grammar*, pages 355–376. Oxford University Press.

C-T James Huang. 1984. On the distribution and reference of empty pronouns. *Linguistic Inquiry*, pages 531–574.

Valerij V Ivanov. 1982. Istorija vremennyx form glagola. In Ruben Avanesov and Valerij Ivanov, editors, *Istoricheskaja grammatika russkogo jazyka. Morfologija. Glagol*, pages 25–131. Nauka.

T Florian Jaeger and Esteban Buz. 2017. Signal reduction and linguistic encoding. In Eva Fernández and Helen Smith Cairns, editors, *The Handbook of psycholinguistics*, pages 38–81. Wiley-Blackwell.

Andrej Kibrik. 2004. Zero anaphora vs. zero person marking in Slavic: A chicken/egg dilemma? In António Branco, Tony McEnery, and Ruslan Mitkov, editors, *Proceedings of the 5th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC)*, pages 87–90. Edicoes Colibri.

Andrej Kibrik. 2011. *Reference in discourse*. Oxford University Press.

Simon Kirby, Monica Tamariz, Hannah Cornish, and Kenny Smith. 2015. Compression and communication in the cultural evolution of linguistic structure. *Cognition*, 141:87–102.

Alexander Koplenig, Peter Meyer, Sascha Wolfer, and Carolin Müller-Spitzer. 2017. The statistical trade-off between word order and word structure – large-scale evidence for the principle of least effort. *PLOS ONE*, 12(3):1–25, 03.

Anthony S. Kroch. 1989. Reflexes of grammar in patterns of language change. *Language Variation and Change*, 1(3):199–244.

Alexandra Kuznetsova, Per B Brockhoff, Rune HB Christensen, et al. 2017. lmertest package: Tests in linear mixed effects models. *Journal of Statistical Software*, 82(13):1–26.

D Robert Ladd, Seán G Roberts, and Dan Dediu. 2015. Correlational studies in typological and historical linguistics. *Annu. Rev. Linguist.*, 1(1):221–241.

Natalia Levshina. 2019. Token-based typology and word order entropy: A study based on Universal Dependencies. *Linguistic Typology*, 23(3):533–572.

Monica Ann Macaulay. 1996. *A grammar of Chalcatongo Mixtec*. Univ of California Press.

Gereon Müller. 2006. Pro-drop and impoverishment. In Patrick Brandt and Eric Fuss, editors, *Form, structure, and grammar. A festschrift presented to Günther Grewendorf on occasion of his 60th birthday*, pages 93–115. Narr.

Marco Nicolis. 2005. *On pro drop*. Ph.D. thesis, University of Siena.

Tiago Pimentel, Brian Roark, and Ryan Cotterell. 2020. Phonotactic complexity and its trade-offs. *Transactions of the Association for Computational Linguistics*, 8:1–18.

R Core Team, 2016. *R: A language and environment for statistical computing, Version 3.3.0*. R Foundation for Statistical Computing, Vienna, Austria.

R Core Team, 2020. *R: A language and environment for statistical computing, Version 4.0.2*. R Foundation for Statistical Computing, Vienna, Austria.

Luigi Rizzi. 1982. *Issues in Italian syntax*. Foris.

Ian Roberts. 2009. A deletion analysis of null subjects. In Theresa Biberauer, Anders Holmber, and Ian Roberts, editors, *Parametric variation: Null subjects in minimalist theory*, pages 58–87. Cambridge University Press.

Seán G. Roberts. 2018. Robust, causal, and incremental approaches to investigating linguistic adaptation. *Frontiers in Psychology*, 9:166.

Robert Rodman. 1997. On left dislocation. In Frans Zwarts Elena Anagnostopolou, Henk van Riemsdijk, editor, *Materials on left dislocation*, pages 31–54. John Benjamins.

Karsten Schmidtke-Bode, Natalia Levshina, Susanne Maria Michaelis, and Ilja A Seržant, editors. 2019. *Explanation in typology: Diachronic sources, functional motivations and the nature of the evidence*. Language Science Press.

Seunghyun Seo. 2001. *The frequency of null subject in Russian, Polish, Czech, Bulgarian and Serbo-Croatian: An analysis according to morphosyntactic environments*. Ph.D. thesis, Indiana University.

Ilja A Seržant. forthcoming-a. Cyclic changes in verbal indexes are not drift processes. *Folia Linguistica Historica*.

Ilja A Seržant. forthcoming-b. The dynamics of slavic morphosyntax is primarily determined by the geographic location and contact configuration. *Scando-Slavica*.

Anna Siewierska. 2004. *Person*. Cambridge University Press.

Anna Siewierska. 2013. Verbal person marking. In Matthew S Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Alexandra Simonenko, Benoît Crabbé, and Sophie Prevost. 2019. Agreement syncretisation and the loss of null subjects : quantificational models for medieval french. *Language Variation and Change*, 31(3):275–301.

Kaius Sinnemäki. 2014. Complexity trade-offs: A case study. In Frederick Newmeyer and Laurel Preston, editors, *Measuring grammatical complexity*, pages 179–201. Oxford University Press.

Knut Tarald Taraldsen. 1980. *On the nominative island condition, vacuous application and the that-trace filter*. Indiana Univ. Linguistics Club.

Andrej Zaliznjak. 2004. *Drevnenovgorodskij dialekt. 2-e izd., dop. i pererab*. Jazyki slavjanskoj kul'tury.

Daniel Zeman, Joakim Nivre, et al. 2020. Universal dependencies 2.6. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

George Kingsley Zipf. 1935. *The psychobiology of language*. Houghton-Mifflin.

Estimating POS Annotation Consistency of Different Treebanks in a Language

Akshay Aggarwal
Univerzita Karlova
Matematicko-fyzikální fakulta
Malostranské náměstí 25
Prague, Czechia
akshayantimatter@gmail.com

Daniel Zeman
Univerzita Karlova
Matematicko-fyzikální fakulta
Malostranské náměstí 25
Prague, Czechia
zeman@ufal.mff.cuni.cz

Abstract

We introduce a new symmetric measure (called θ_{pos}) that utilises the non-symmetric KL_{cpos^3} measure (Rosa and Žabokrtský, 2015) to allow us to compare the annotation consistency between different treebanks of a given language, annotated under the same guidelines. We can set a threshold for this new measure so that a pair of treebanks can be considered harmonious in their annotation if θ_{pos} does not surpass the threshold. For the calculation of the threshold, we estimate the effects of (i) the size variation, and (ii) the genre variation in the considered pair of treebanks. The estimations are based on data from treebanks of distinct language families, making the threshold less dependent on the properties of individual languages. We demonstrate the utility of the proposed measure by listing the treebanks in Universal Dependencies version 2.5 (UDv2.5) (Zeman et al., 2019) data that are annotated consistently with other treebanks of the same language. However, the measure could be used to assess inter-treebank annotation consistency under other (non-UD) annotation guidelines as well.

1 Introduction

There exist a multitude of treebanks for different languages (Zeman et al., 2014). As noted by Kakkonen (2006), there exist a variety of formats and annotation schemes even for the treebanks for the same language. As an example, two well known POS tagging schemes for English language include the POS tagging scheme of the Penn Treebank[1] (Marcus et al., 1994) and the Universal POS tagset (Petrov et al., 2012).

The Universal Dependencies (UD) Project (Nivre et al., 2016b; Nivre et al., 2020) was introduced in 2014 as a means of unifying all the novel features of different annotation formats as a universal annotation scheme consistent across different languages. It has since become a standard reference to compare scores relating to parser performance (Che et al., 2018; Martínez Alonso et al., 2017), study of language-specific features (Alzetta et al., 2018), and for dependency parsing shared tasks on UD (Zeman et al., 2018).

UDv2.5 (Zeman et al., 2019) contains 157 treebanks in 90 languages, with multiple treebanks for some languages. Regardless of the differences in genre or the teams involved in building the treebanks, all treebanks of one language should be consistent with respect to the annotation guidelines, both intra and inter treebanks. However, this is often not the case, primarily because of the different sources of origin of the annotated data. The problem of determining the degree to which the different treebanks differ from each other has been studied in some detail over multiple years, but is not yet entirely solved.

The rest of the article is organised as follows. The literature relevant to the problem is discussed in Section 2, followed by a short introduction to the KL_{cpos^3} measure and a definition of the proposed measure in Section 3. Section 4 lists the constraints for choosing the dataset for the experiments as listed in Sections 5 and 6. The results of the experiments are summarised in Section 7. A discussion of the measure concludes the article in Section 8. The treebanks

[1] https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

in UDv2.5 are marked for consistency or inconsistency of their POS annotation based on the proposed measure in Appendix A; Appendix B demonstrates the calculation of the measure for a concrete pair of treebanks.

2 Related Work

One of the most commonly used approaches to find inconsistencies in annotation is to train a high quality tagger or parser on the given training data, and evaluating the cases where the prediction from the trained model differs from the annotation of the test data. This approach can also be extended by bootstrapping different trained models, with the majority consensus being compared against the available annotation. Martínez Alonso and Zeman (2016) assessed the similarity of the Spanish treebanks in UDv1.3 (Nivre et al., 2016a) using dependency parsing. A high-efficiency parser was trained on one of the treebanks, and then tested on another. If a drop in parsing accuracy was more than what was intuitive, the treebanks were marked as not similar enough. The same technique was employed to evaluate the different Russian treebanks in UDv2.2 (Nivre et al., 2018) against each other by Droganova et al. (2018). It is worth stating here that the performance of the used tagger or parser may be a bottleneck, with the additional variables of the size and genre composition of the evaluated treebanks, among others. Furthermore, the acceptable variability in score in such cases depends on the architecture of the trained model, and is not comparable across different languages, or even when a different architecture is employed on the same data.

Dickinson and Meurers (2003a; 2003b) focus on finding an n-gram of tokens in the corpus that occurs in the same context (referred to as a *variation nucleus*) such that its different occurrences are annotated differently. Originally coined for continuous annotation,[2] the method was eventually adapted to look for inconsistencies in discontinuous annotation as well (Dickinson and Meurers, 2005).

Chun et al. (2018) compare the POS annotation consistency for several Korean treebanks by using the relative frequency of the individual POS tags, while also briefly mentioning the cause of the variation in their distribution. While such analysis is slightly helpful in terms of drawing a comparison, it does not consider the interaction of different POS tags with each other. To illustrate such interactions, an n-gram-based approach might be utilised.

3 KL_{cpos^3} and Measure Definition

In a delexicalised cross-language parser transfer scenario, Rosa and Žabokrtský (2015) show that the KL-Divergence score of POS trigrams, referred to as KL_{cpos^3}, can be effectively used for selection of the source language.

$$KL_{cpos^3}(tgt, src) = \sum_{\forall cpos^3 \in tgt} f_{tgt}(cpos^3) \log \frac{f_{tgt}(cpos^3)}{f_{src}(cpos^3)} \tag{1}$$

where $cpos^3$ is a coarse-grained[3] POS tag trigram, and

$$\begin{aligned} f(cpos^3) &= f(cpos_{i-1}, cpos_i, cpos_{i+1}) \\ &= \frac{count(cpos_{i-1}, cpos_i, cpos_{i+1})}{\sum_{\forall cpos_{a,b,c}} count(cpos_a, cpos_b, cpos_c)} \end{aligned}$$

with $count_{src}(cpos^3) = 1$ for each unseen trigram and a special value for $cpos_{i-1}$ or $cpos_{i+1}$ when $cpos_i$ lies on the sentence beginning or end.

[2] The annotation of the current token is based on the annotation of a contiguous token in word order. Discontinuous annotation implies the annotation of the current token is dependent on another token that might not be contiguous in the word order, as in the case of dependency parsing.

[3] For example, the coarse-grained POS associated with different nouns would be `NOUN` while the fine-grained POS would include `NN`, `NNP`, `NNS`, etc. We use UPOS tags for UD data, which are already coarse-grained in nature.

Considering that a treebank of the same language (despite the differences in the genres[4] covered) should be a better fit for POS transfer than a treebank from another language, we employ a symmetric variant of KL_{cpos^3}, called θ_{pos}, to assess the annotation consistency among the different treebanks of a language. θ_{pos} is a non-negative divergence measure. However, the measure scores cannot be compared directly across different languages. For a language-independent usage, there should be an empirical upper bound that needs to be placed on the θ_{pos} scores. As long as the θ_{pos} scores are lower than this empirical bound, the considered pair of treebanks can be considered harmonious in terms of their POS annotation. We denote this empirical upper bound by Θ_{pos}. The measures θ_{pos} and Θ_{pos} are linked together in the following definition:

Definition 1. Given two treebanks A and B, we say the treebanks are consistent in their POS annotation if the symmetric measure of their mutual divergence (given by θ_{pos}) is less than or equal to a threshold (given by Θ_{pos}). Formally, it can be represented as:

$$\begin{aligned}\theta_{pos}(A,B) &= KL_{cpos^3}(A,B) + KL_{cpos^3}(B,A) \qquad (2)\\ &\leq \Theta_{pos}(A,B) \qquad (3)\end{aligned}$$

where $KL_{cpos^3}(P,Q)$ indicates the KL_{cpos^3} score of Q as an estimator for P.

Even though Θ_{pos} is an empirical bound on the θ_{pos} measure, the former is essentially a property of the latter. The empirical upper bound value would need to be estimated anew for a different set of annotation guidelines. In the remaining article, we estimate the empirical upper bound in a language-independent manner by looking at the influence of size of data, and the POS distribution in individual genres on θ_{pos} in different UDv2.5 treebanks (Zeman et al., 2019).

4 Assumptions while Working with UD Data

The UD website[5] provides a star ranking of individual treebanks within each language. The ranking is calculated heuristically[6], depending on multiple factors including the size of the treebank and the number of genres present in the data. The score also incorporates the output from the official UD validator[7] and from the search for known error types[8] in UDAPI (Popel et al., 2017). The treebank's compliance with the UD guidelines thus plays an important role in the score. While it is possible for a treebank to have a high score without being internally consistent, we assume that a treebank that adheres better to the guidelines also contains fewer inconsistencies. Therefore, we trust treebanks rated 3.5 stars or more (out of 5 stars).

Sometimes a whole treebank may not be sufficiently internally consistent because different genres have different distributions of POS n-grams. We may then require that the data belonging to one particular genre is annotated consistently.

5 Dataset Size and θ_{pos}

The value of θ_{pos} may depend on data size, as some POS trigrams may not be present in small datasets. We use k-fold cross validation to check the effect of presence or absence of POS trigrams in the data, based on the data size.

Experimental Setup

$KL_{cpos^3}(tgt, src)$ is defined on distributions of trigrams found in tgt and src. The calculated scores (and consequently θ_{pos} scores) are therefore affected by the presence or absence of the

[4] The usage of 'genre' in this context should also account for domain distinctions. In case such a distinction is available explicitly, data from each domain should be considered a separate 'genre'. To some extent this is actually the case with the 'genre' labels that are available in UD data and used in our experiments.

[5] universaldependencies.org

[6] For more details on the associated heuristics, refer to https://github.com/UniversalDependencies/tools/blob/master/evaluate_treebank.pl

[7] https://github.com/UniversalDependencies/tools/blob/master/validate.py

[8] https://udapi.readthedocs.io/en/latest/_modules/udapi/block/ud/markbugs.html

POS trigrams. In order to discount variability of θ_{pos} because of genre distribution, we use data from a single genre *(news)*. We take two UDv2.5 treebanks that have a large number of *news* sentences, high star ranking, and that belong to different language families: Czech-PDT (Indo-European, rated 4.5 stars) and Estonian-EDT (Uralic, rated 4 stars). For easier manipulation, we downsample the *news* data from either treebank as shown in Table 1.

Treebank	**Genre**	**Sentences**	**Downsampled to**
Czech-PDT	News	53,075	50,000
Estonian-EDT	News	13,557	12,000

Table 1: Sentence Counts in the *news* genre in Czech-PDT and Estonian-EDT.

To check the effect of data size on θ_{pos}, we run k-fold cross-validation on the downsampled data with different k-values. For each value of k, the downsampled data gets split to k folds, we select randomly one fold as test set and compute θ_{pos} of each of the remaining $k-1$ folds and the test set. This way we obtain $k-1$ values of θ_{pos}; their average is the θ_{pos} value we report for the given k in Table 2.

In addition to finding the values of θ_{pos}, we are also interested in finding its relationship with the count of unique trigrams common to the pair of distributions. We define coverage for a fold as the count of unique trigrams common to both training and test sets in the fold, expressed as a ratio of the count of all unique trigrams in the larger training set.

Experimental Scores and Inference

k **value**	θ_{pos} **Score**	**Coverage (in %)**
5	0.021 ± 0.001	83.872 ± 0.552
10	0.037 ± 0.001	75.447 ± 0.619
20	0.069 ± 0.002	66.131 ± 0.691
50	0.161 ± 0.005	52.768 ± 0.806
100	0.304 ± 0.011	42.373 ± 0.868
250	0.663 ± 0.028	29.345 ± 0.926
500	1.092 ± 0.053	20.784 ± 0.952

(a) *news* Data from UDv2.5 Czech-PDT, downsampled to 50,000 sentences

k **value**	θ_{pos} **Score**	**Coverage (in %)**
4	0.064 ± 0.002	76.139 ± 0.814
6	0.087 ± 0.003	69.742 ± 0.835
8	0.109 ± 0.004	65.177 ± 0.855
12	0.155 ± 0.005	58.72 ± 0.934
16	0.2 ± 0.007	54.142 ± 0.948
24	0.286 ± 0.011	47.727 ± 0.964
48	0.52 ± 0.022	37.094 ± 1.01
120	1.039 ± 0.052	24.474 ± 1.055

(b) *news* Data from UDv2.5 Estonian-EDT, downsampled to 12,000 sentences

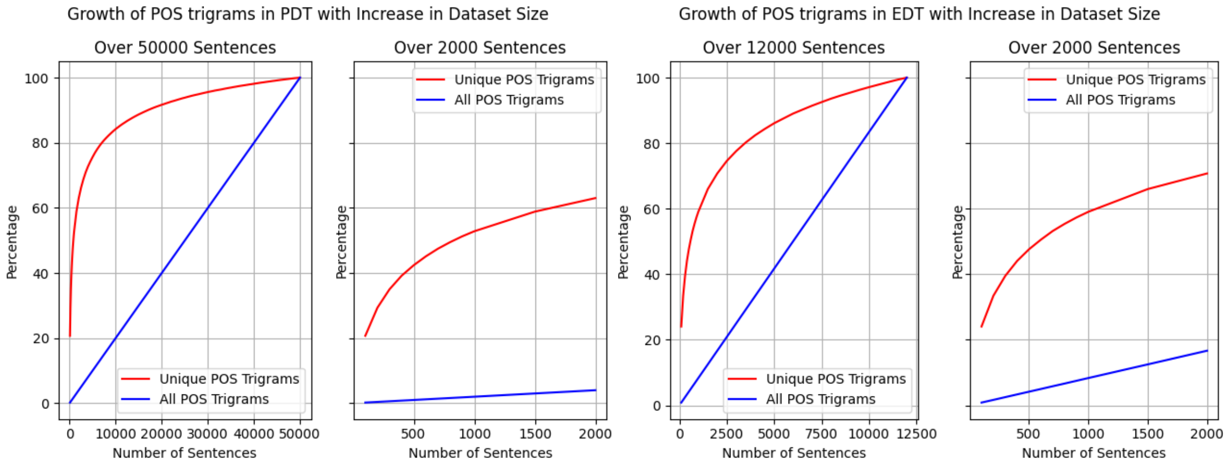
Table 2: θ_{pos} and coverage of POS trigram scores (± standard deviation) averaged over 100 different k-fold iterations. Each iteration results in a different downsample.

While there exists a strong negative correlation (Pearson correlation coefficient, r = -0.9075 and -0.9252 in Tables 2a, 2b respectively) between coverage of POS trigrams and the θ_{pos} scores, the coverage is, however, dependent on the size of the datasets being compared. Figures 1a and 1b show the variability in (i) number of distinct POS trigrams, and (ii) total number of POS trigrams, as the data size changes.

As evident from the figures, the growth pattern of counts is similar in both languages. The POS trigrams in a small part of the dataset obviously cannot be considered representative of those present in the entire dataset. Based on the observed coverage curve, we set 400 sentences[9] as the minimum size of a dataset whose consistency with another dataset is assessed.

However, difference in average sentence length is a factor that needs to be taken in account as well. If the two treebanks differ considerably in their average sentence length, then the size expressed in number of sentences does not reflect the number of tokens (and, consequently, the number of POS trigrams). For example, consider the Arabic treebanks in Table 3. If we take an equal number of sentences from Arabic-PUD and either of the other two treebanks, the total number of words will differ by a factor of almost 2.

[9] At about 400 sentences the percentage in Figure 1 crosses 40%.



(a) In *news* genre of Czech-PDT

(b) In *news* genre of Estonian-EDT

Figure 1: Growth of POS trigrams with increase in dataset size

Counts	Arabic-NYUAD	Arabic-PADT	Arabic-PUD
Syntactic words	738,889	282,384	20,751
Sentences	19,738	7,664	1,000
Average length	37.434	36.845	20.751

Table 3: Average sentence lengths in Arabic treebanks. A syntactic word (node in the dependency tree) typically corresponds to a surface token but some tokens are split to multiple syntactic words.

Accommodating the dataset-size comparison, we can formally set the conditions such that the datasets can be compared amongst each other. Given two datasets A, B; the pair can be checked for annotation consistency if the following heuristic constraints are satisfied:

1. Individual dataset has at least 400 sentences, i.e. $\big(size(A) \geq 400 \ \& \ size(B) \geq 400\big)$; and

2. Dataset with smaller average sentence length has at least as many syntactic words as 400 sentences in the other dataset, i.e.

$$\boxed{\big(AvgSentLen(B) \leq AvgSentLen(A)\big) \implies \big(TotalSyntacticWords(B) \geq 400 \cdot AvgSentLen(A)\big)}$$

From Table 2, when the test split is composed of 500 sentences ($k = 100$ for Czech; $k = 24$ for Estonian), the θ_{pos} measure is ≈ 0.3. Considering that the larger values of k in either dataset do not satisfy heuristic constraint 1, we estimate the empirical upper bound of θ_{pos} based on $k = 100$ (Czech) and $k = 24$ (Estonian), respectively.

When estimating Θ_{pos}, we do not want to be too restrictive because the observed $\theta_{pos} \approx 0.3$ is based on internal consistency of a good treebank, which will be very hard to match for consistency between two different treebanks. We, therefore, round off the maximum observed θ_{pos} score from ≈ 0.3 to 0.5. Formally, if the datasets A, B contain data from the same genre, and the size of the datasets is comparable (as per heuristic constraints defined before), the upper limit on the θ_{pos} score can be specified in Equation 4.

$$\boxed{\theta_{pos}(A,B) \leq \Theta_{pos}(A,B) = 0.5} \tag{4}$$

6 Genre Distribution and θ_{pos}

In the previous experiments we assumed that the two compared datasets consist of the same language *and genre*. It is likely that the distribution of POS trigrams will differ when the two datasets consist of different genres. We now proceed to investigate cross-genre variability inside a treebank that we believe is reasonably internally consistent. We are looking for Θ_{pos} thresholds that could be used to assess annotation similarity of two treebanks that differ in genre.

6.1 Inter-Genre Similarity

The Polish-LFG treebank in UDv2.5 (rated 4 stars) contains data from different genres,[10] the counts of which are shown in Table 4a. Table 4b shows the genres in UDv2.5 Finnish-TDT treebank (rated 3.5 stars). In this case, the data labeled *europarl* and *uni_articles* (university articles) is kept separate and not used in the estimation of variability of $\boldsymbol{\theta}_{pos}$ across genres.

Genre (X)	*size*(*X*)	*AvgSentLen*(*X*)
fiction	7,252	7.124
news	6,744	8.401
nonfiction	1,273	7.719
social	526	6.977
spoken	1,253	6.047
academic	51	8.118
blog	136	7.772
legal	11	9.273

(a) In UDv2.5 Polish-LFG

Source (X)	*Size*(*X*)	*AvgSentLen*(*X*)
fiction	2,739	11.981
wiki	2,269	14.049
grammar	2,002	8.48
blog	1,781	12.533
legal	1,141	20.968
news	3,064	13.026
europarl	1,082	18.441
uni_articles	1,058	13.261

(b) In UDv2.5 Finnish-TDT

Table 4: Sources of genre data in UDv2.5 treebanks. Genres used in estimation of $\boldsymbol{\theta}_{pos}$ scores are marked in bold.

As can be seen from Table 5, the different genres in Finnish-TDT are internally consistent in their annotation, as per the constraint in Equation 4. For each genre source, the dataset is downsampled to 900 sentences, and the results are presented on the individual folds resulting from 2-fold cross-validation on the downsampled data. The similar analysis for genres in Polish-LFG is omitted here because the *social* genre does not have enough data.

Genres	θ_{pos} (± sd)	Θ_{pos}
fiction	0.316 ± 0.015	0.5
wiki	0.3 ± 0.017	0.5
grammar	0.427 ± 0.021	0.5
blog	0.332 ± 0.017	0.5
legal	0.216 ± 0.035	0.5
news	0.286 ± 0.015	0.5
europarl	0.233 ± 0.017	0.5
uni_articles	0.3 ± 0.014	0.5

Table 5: $\boldsymbol{\theta}_{pos}$ (± standard deviation) averaged over 100 runs for each genre in UDv2.5 Finnish-TDT. Each run results in a different downsample.

Experimental Setup

We compare different genres in the Polish-LFG and Finnish-TDT treebanks by presenting the $\boldsymbol{\theta}_{pos}$ scores for each pair of genres (as per Table 4). Each genre is downsampled to the number of instances as listed in Table 6 such that the heuristic constraints for dataset comparison are satisfied.

Experimental Scores and Inference

Tables 7 and 8 list the $\boldsymbol{\theta}_{pos}$ scores for data from Polish-LFG and Finnish-TDT, respectively. It is worth noting that for most genre pairs, the Θ_{pos} constraint as employed in Equation 4 is not enough, as $\boldsymbol{\theta}_{pos}$ frequently surpasses the imposed limit of 0.5.

[10] https://github.com/UniversalDependencies/UD_Polish-LFG#data-split-and-genres

Genre (X)	Downsampled to	$\frac{TotalSyntacticWords(X)}{AvgSentLen(A)}$
fiction	500	424
news	500	500
nonfiction	500	459
social	500	415
spoken	600	432

(a) UDv2.5 Polish-LFG

Genre (X)	Downsampled to	$\frac{TotalSyntacticWords(X)}{AvgSentLen(A)}$
fiction	1,000	571
wiki	1,000	670
grammar	1,000	404
blog	1,000	598
legal	1,000	1,000
news	1,000	621

(b) UDv2.5 Finnish-TDT

Table 6: Counts of sentences for different genres in data downsampled from UDv2.5 treebanks. A in $Avg(A)$ in the third column refers to the genre with the highest number of average words per sentence in each language, marked in bold.

Genres	news	nonfiction	social	spoken
fiction	0.754 ± 0.047	0.556 ± 0.028	0.726 ± 0.032	1.059 ± 0.047
news	-	0.55 ± 0.032	0.906 ± 0.044	1.53 ± 0.071
nonfiction	-	-	0.624 ± 0.027	1.285 ± 0.046
social	-	-	-	1.178 ± 0.033

Table 7: θ_{pos} scores (± standard deviation) averaged over 100 runs for inter-genre analysis in downsampled UDv2.5 Polish-LFG data. Each run results in a different downsample.

Genres	blog	grammar	wiki	legal	news
fiction	0.356 ± 0.014	0.47 ± 0.019	1.552 ± 0.041	1.559 ± 0.04	1.323 ± 0.044
blog	-	0.504 ± 0.018	1.307 ± 0.042	1.328 ± 0.026	1.113 ± 0.043
grammar	-	-	1.166 ± 0.041	1.554 ± 0.036	0.888 ± 0.035
wiki	-	-	-	1.229 ± 0.032	0.473 ± 0.021
legal	-	-	-	-	1.078 ± 0.026

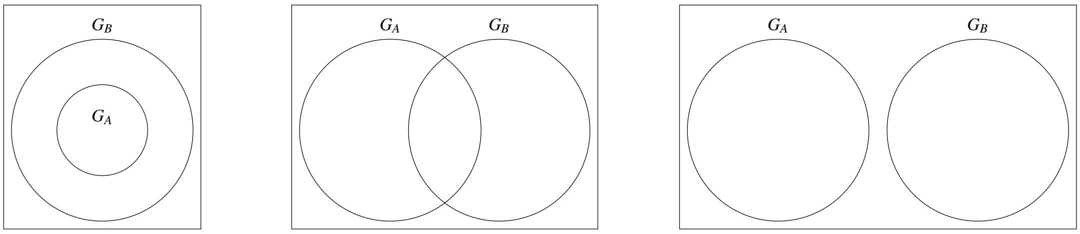
Table 8: θ_{pos} scores (± standard deviation) averaged over 100 runs for inter-genre analysis in downsampled UDv2.5 Finnish-TDT data. Each run results in a different downsample.

As expected, we need a higher threshold when comparing datasets whose genre does not match. While a threshold of 1.6 would accommodate data in Polish-LFG and Finnish-TDT, we again allow some room to reduce false alarms about inconsistent pairs of treebanks, and frame the empirical upper bound on θ_{pos} between genre x in dataset A (written as A_x) and genre y in dataset B (B_y) as in Equation 5, given below:

$$\boxed{\theta_{pos}(A_x, B_y) \leq \Theta_{pos}(A_x, B_y) = 2.0} \tag{5}$$

6.2 Combination of Genres

We denote the set of genres in treebank X as G_X. Given two treebanks with at least one different genre, the different genres in the two treebanks can interact in either of the three cases as shown in Figure 2. To see how the θ_{pos} scores are affected in either of the cases, we experiment with the data from UDv2.5 Polish-LFG.



(a) Case 1: $G_A \subseteq G_B$ (b) Case 2: $G_A \not\subseteq G_B$; $G_A \cap G_B \neq \phi$ (c) Case 3: $G_A \not\subseteq G_B$; $G_A \cap G_B = \phi$

Figure 2: Interaction of genres in treebanks A and B, such that $|G_A| \leq |G_B|$

Experimental Setup

We start by downsampling the data from the *fiction* and *news* genres to 2000 sentences each. Using 2-fold cross-validation, the downsampled data is then split into 2 halves, termed as *base* and *test* set for the genre. In addition, we downsample the data from the *spoken* genre to 1000 sentences and use it as a *test* set (without corresponding *base* set).

We try to understand θ_{pos} variability in the scenarios depicted in Figure 2. The different genres combining together to form a dataset can be identified by the name of the concatenated dataset. The trailing *base* in the dataset name marks that it is composed of data from the *base* set of the genre(s). The datasets using *test* set of genre(s) can similarly be identified by trailing *test* in the dataset name.

Experimental Scores and Inference

We present the calculated scores for different cases in Table 9.

	news_base	*fiction_base*	*news_fiction_base*
news_test	0.257 ± 0.010	0.64 ± 0.034	0.3 ± 0.015
fiction_test	0.646 ± 0.034	0.278 ± 0.013	0.351 ± 0.021
spoken_test	1.503 ± 0.049	0.99 ± 0.036	1.144 ± 0.035
spoken_news_test	0.489 ± 0.022	0.499 ± 0.020	0.338 ± 0.014
spoken_fiction_test	0.854 ± 0.036	0.41 ± 0.018	0.498 ± 0.023
news_fiction_test	0.304 ± 0.016	0.348 ± 0.019	0.17 ± 0.007
all_genres	0.463 ± 0.022	0.351 ± 0.014	0.247 ± 0.011

	news_test	*fiction_test*	*news_fiction_test*
spoken	1.493 ± 0.048	0.987 ± 0.034	1.138 ± 0.03

Table 9: θ_{pos} (± standard deviation) scores averaged over 100 runs, reported for different genre combinations. Each run results in a different downsample. The scores marked in blue indicate that the genre sets overlap, while those in red indicate the genre sets are disjunct. The scores without color-code indicate that one genre set is a subset of the other.

It is noteworthy that the decomposition of a treebank into its constituent genres forms the first basis for the study of variance of θ_{pos} scores with a combination of the different genres. Upon a closer inspection, it was discovered that when there are multiple genres present in the treebank, the θ_{pos} measure score is dominated by the POS trigrams that are typical of the language, and the genre-specific POS trigrams become increasingly obscure.

Once the individual genres have been identified and checked for the inter-genre θ_{pos} scores, the overall measure score is less than the average of the measure scores calculated for individual pair of genres in the treebank(s). Formally, assuming treebanks A and B can be split into their constituent genres such that $G_A = \{A_1, A_2, ..., A_i\}$ and $G_B = \{B_1, B_2, ..., B_j\}$, the overall limit on the $\theta_{pos}(A,B)$ score can be specified as in Equation 6.

$$\boxed{\theta_{pos}(A,B) \le \Theta_{pos}(A,B) \le Average(\theta_{pos}(A_x,B_y))} \qquad \forall[A_x \in G_A; B_y \in G_B] \tag{6}$$

6.3 Adulterant Genres

In our analysis so far, we have restricted ourselves to instances where the data in the different genres could be reliably compared. We define a genre in the dataset as *adulterant* if the number of sentences in the genre does not satisfy either or both the constraints pertaining to dataset comparison. In this subsection, we take a look at how the presence of adulterant genres affects the θ_{pos} scores.

Experimental Setup

To study the effect of adulterant genres, we first downsample data from the *fiction*, *news* and *spoken* genres in Polish-LFG to 500, 500 and 600 sentences respectively. For adulterant genres, we work with the data from the *academic*, *blog* and *legal* genres. The data from all the adulterant genres is concatenated to form a dataset labeled *others*. Non-adulterant genres are then combined with adulterant genres to result in a dataset identified as *X*-*Y*, where *X* contains data from *news*, or *fiction*, a combination of the two genres. *Y* may be an individual adulterant genre, or a combination of all adulterant genres *(others)*. All the datasets created from the downsampled data are compared with the downsampled data from *spoken*.

Experimental Scores and Inference

The calculated θ_{pos} scores for each pair, averaged over 100 runs, are reported in Table 10.

	spoken
fiction	1.059 ± 0.047
fiction-academic	1.072 ± 0.046
fiction-blog	1.09 ± 0.044
fiction-legal	1.065 ± 0.047
fiction-others	2.413 ± 0.384

	spoken
news	1.53 ± 0.071
news-academic	1.552 ± 0.069
news-blog	1.54 ± 0.065
news-legal	1.547 ± 0.071
news-others	2.63 ± 0.334

	spoken
fiction_news	1.196 ± 0.048
fiction_news-academic	1.215 ± 0.048
fiction_news-blog	1.223 ± 0.046
fiction_news-legal	1.206 ± 0.048
all-genres	2.309 ± 0.358

Table 10: θ_{pos} Scores (± standard deviation) averaged over 100 different runs with adulterant genres present in Polish-LFG. Each run results in a different downsample.

From the table, we observe that a low number of adulterant genres in the data does not affect the θ_{pos} scores heavily. However, the presence of multiple adulterant genres pushes the θ_{pos} scores by almost 1.5 as compared to when there are no adulterants present. Taking into account also the standard deviation score, and the high annotation quality of the treebank, we can add a headroom of +2.0 if adulterant genres are present.

Formally, assuming treebanks A and B can be split into their constituent genres such that $G_A = \{A_1, A_2, ..., A_{n1}\}$ and $G_B = \{B_1, B_2, ..., B_{n2}\}$. Of all the constituent genres in $G_A \cup G_B$, the set of adulterant genres can be represented as $G_{adulterant}$. The overall limit on the $\theta_{pos}(A,B)$ score, as specified in Equation 6, can be updated as in Equation 7

$$\boxed{\theta_{pos}(A,B) \leq \Theta_{pos}(A,B) \leq \begin{cases} Average(\theta_{pos}(A_x,B_y)) + 2.0 & \text{if } G_{adulterant} \neq \phi \\ Average(\theta_{pos}(A_x,B_y)) & \text{if } G_{adulterant} = \phi \end{cases}} \tag{7}$$

$$\forall [A_x, B_y \in (G_A \cup G_B) - G_{adulterant}]$$

7 Framing the Overall θ_{pos} Limit

In a case when the data from individual genres in the data is not annotated consistently, the θ_{pos} score might be within the bounds of averaged scores for individual genres, therefore marking the pair as consistent. To avoid this, we calculate the idealistic Θ'_{pos} as the average of Θ_{pos} values for the genres.

$$\Theta'_{pos}(A,B) = Average(\Theta_{pos}(A_x,B_y)) \qquad \forall [A_x, B_y \in (G_A \cup G_B)] \tag{8}$$

where $\Theta_{pos}(A_x,B_x) = 0.5$ and $\Theta_{pos}(A_x,B_y) = 2.0$ as per Equations 4 and 5, respectively.

For overall calculation of Θ_{pos} scores for treebanks with multiple genres, the overall computation can be given by:

$$\boxed{\theta_{pos}(A,B) \leq \Theta_{pos}(A,B) = \begin{cases} Minimum(\Theta'_{pos}(A_x,B_y),\ Average(\theta_{pos}(A_x,B_y),\ 2.0) & \text{if } G_{adulterant} = \phi \\ Minimum(\Theta'_{pos}(A_x,B_y),\ Average(\theta_{pos}(A_x,B_y),\ 2.0) + 2.0 & \text{if } G_{adulterant} \neq \phi \end{cases}} \tag{9}$$

$$\forall [A_x, B_y \in (G_A \cup G_B) - G_{adulterant}]$$

where $\theta_{pos}(A_x,B_y)$ refers to the θ_{pos} score calculated between genre x present in treebank A and genre y present in treebank B.

Regardless of the genre composition of the treebanks under consideration, the treebanks with $\theta_{pos} \leq 0.5$ are termed as consistent in their POS annotation. Similarly, the treebanks with $\theta_{pos} \geq 4.0$ are termed as inconsistent in their POS annotation. In case of multiple genres present in either treebank, Equation 9 can be employed if just the percentage composition of different genres in the treebanks is known, regardless of whether it is possible to split the treebank into the constituent genres. However, for a fine-tuned estimation, it is imperative to be able to split the treebank into its constituent genres.

For treebanks with adulterant genres, the higher Θ_{pos} limit on the θ_{pos} scores can be problematic. If possible, the adulterated genres should be isolated and the annotation consistency of the treebank should be checked without presence of any adulterant genre(s).

8 Discussion and Conclusion

8.1 Using θ_{pos} to Localise Inconsistency

While the θ_{pos} measure is primarily meant to identify whether two given treebanks are consistent in their POS annotation, the measure can also be employed to localise points of inconsistency, if required.

Consider the example of two Finnish treebanks in UDv2.5, FTB and TDT. While the data in the former is composed of a single genre, *grammar-examples*, the data in the latter consists of multiple genres, including *grammar-examples*. We can observe that

$$\theta_{pos}(\text{Finnish-TDT}_{grammar-examples}, \text{Finnish-FTB}_{grammar-examples}) = 0.707 > 0.5$$

which is a clear violation of the condition as specified in Equation 4. We believe that the inconsistency in the annotation can be localised to the *grammar-examples* part of Finnish-TDT. Consequently, concentrating simply on the instances from this genre should be enough to bring the overall θ_{pos} score between the two treebanks under the Θ_{pos} limit.

8.2 Split into Constituent Genres as a Requirement

The estimation of Θ_{pos} is primarily based on the requirement that the genre composition of treebanks is known. While the limit is best estimated when the genres can be isolated and the adulterant genres identified, it is possible to get a crude estimate of the limit. For example, one can estimate all the common genres with θ_{pos} scores of 0.5, and the different genres have a θ_{pos} score of 2.0. An average of these estimates should give a crude estimate on the Θ_{pos} limit without accounting for an adulterant genre. Data with multi-genre classification can also be handled in a similar manner.

8.3 Conclusion

We proposed a numeric measure based on the KL_{cpos^3} measure (Rosa and Žabokrtský, 2015) to attest the POS annotation consistency across treebanks that allegedly follow the same guidelines, for the same language. Through the use of the measure, we sought to answer how the different treebanks of a language, with variable size and genre distributions but following the same annotation guidelines, can be compared against each other. We also defined a reliable threshold on the proposed measure that would inform the annotators if the treebanks being compared are not consistent with each other. In addition, the measure can also be used intra-treebank to localize the genre(s) that cause the inconsistency with another treebank. We also evaluated different treebanks in UDv2.5 (Zeman et al., 2019) and identified the consistent and inconsistent treebank pairs based on the proposed measure. To the best of our knowledge, this is the first such measure that compares treebanks directly, without an added variable of tagger performance. At present, the measure does not allow checking for consistency in treebanks with syntactic annotation. Perhaps similar ideas might lead to a syntactic version of the measure in the future.

References

Chiara Alzetta, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2018. Universal Dependencies and Quantitative Typological Trends. A Case Study on Word Order. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. 2018. Towards Better UD Parsing: Deep Contextualized Word Embeddings, Ensemble, and Treebank Concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 55–64, Brussels, Belgium, October. Association for Computational Linguistics.

Jayeol Chun, Na-Rae Han, Jena D. Hwang, and Jinho D. Choi. 2018. Building Universal Dependency Treebanks in Korean. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 2194–2202, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Markus Dickinson and W. Detmar Meurers. 2003a. Detecting Errors in Part-of-speech Annotation. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 1*, EACL '03, pages 107–114, Stroudsburg, PA, USA. Association for Computational Linguistics.

Markus Dickinson and W. Detmar Meurers. 2003b. Detecting Inconsistencies in Treebanks. *IEEE Transactions on Learning Technologies - TLT*, 01.

Markus Dickinson and W. Detmar Meurers. 2005. Detecting Errors in Discontinuous Structural Annotation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 322–329, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kira Droganova, Olga Lyashevskaya, and Daniel Zeman. 2018. Data Conversion and Consistency of Monolingual Corpora: Russian UD Treebanks. In *Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories (TLT 2018)*, number 155, pages 53–66, Linköping, Sweden. Linköping University Electronic Press.

Tuomo Kakkonen. 2006. Dependency treebanks: methods, annotation schemes and tools. In *Proceedings of the 15th Nordic Conference of Computational Linguistics (NODALIDA 2005)*, pages 94–104, Joensuu, Finland, May. University of Joensuu, Finland.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating Predicate Argument Structure. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, page 114–119, USA. Association for Computational Linguistics.

Héctor Martínez Alonso and Daniel Zeman. 2016. Universal Dependencies for the AnCora treebanks. *Procesamiento del Lenguaje Natural*, (57):91–98.

Héctor Martínez Alonso, Željko Agić, Barbara Plank, and Anders Søgaard. 2017. Parsing Universal Dependencies without training. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 230–240, Valencia, Spain, April. Association for Computational Linguistics.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Yevgeni Berzak, Riyaz Ahmad Bhat, Cristina Bosco, Gosse Bouma, Sam Bowman, Gülşen Cebiroğlu Eryiğit, Giuseppe G. A. Celano, Çağrı Çöltekin, Miriam Connor, Marie-Catherine de Marneffe, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Daniel Galbraith, Sebastian Garza, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gokirmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Normunds Grūzītis, Bruno Guillaume, Jan Hajič, Dag Haug, Barbora Hladká, Radu Ion, Elena Irimia, Anders Johannsen, Hüner Kaşıkara, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Jessica Kenney, Simon Krek, Veronika Laippala, Lucia Lam, Alessandro Lenci, Nikola Ljubešić, Olga Lyashevskaya, Teresa Lynn, Aibek Makazhanov, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Keiko Sophie Mori, Shunsuke Mori, Kadri Muischnek, Nina Mustafina, Kaili Müürisep, Vitaly Nikolaev, Hanna Nurmi, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Slav

Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Lauma Pretkalniņa, Prokopis Prokopidis, Tiina Puolakainen, Sampo Pyysalo, Loganathan Ramasamy, Laura Rituma, Rudolf Rosa, Shadi Saleh, Baiba Saulīte, Sebastian Schuster, Wolfgang Seeker, Mojgan Seraji, Lena Shakurova, Mo Shen, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Kiril Simov, Aaron Smith, Carolyn Spadine, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Takaaki Tanaka, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Larraitz Uria, Gertjan van Noord, Viktor Varga, Veronika Vincze, Jing Xian Wang, Jonathan North Washington, Zdeněk Žabokrtský, Daniel Zeman, and Hanzhi Zhu. 2016a. Universal Dependencies 1.3. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016b. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Joakim Nivre, Mitchell Abrams, Željko Agić, Lars Ahrenberg, Lene Antonsen, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, John Bauer, Sandra Bellato, Kepa Bengoetxea, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Rogier Blokland, Victoria Bobicev, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Gülşen Cebiroğlu Eryiğit, Giuseppe G. A. Celano, Savas Cetin, Fabricio Chalub, Jinho Choi, Yongseok Cho, Jayeol Chun, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Arantza Diaz de Ilarraza, Carly Dickerson, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Tomaž Erjavec, Aline Etienne, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Nizar Habash, Jan Hajič, Jan Hajič jr., Linh Hà Mỹ, Na-Rae Han, Kim Harris, Dag Haug, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Jena Hwang, Radu Ion, Elena Irimia, Tomáš Jelínek, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşıkara, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Tolga Kayadelen, Václava Kettnerová, Jesse Kirchner, Natalia Kotsyba, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Cheuk Ying Li, Josie Li, Keying Li, KyungTae Lim, Nikola Ljubešić, Olga Loginova, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Gustavo Mendonça, Niko Miekka, Anna Missilä, Cătălin Mititelu, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Shinsuke Mori, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, Adédayọ̀ Olúòkun, Mai Omura, Petya Osenova, Robert Östling, Lilja Øvrelid, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Siyao Peng, Cenel-Augusto Perez, Guy Perrier, Slav Petrov, Jussi Piitulainen, Emily Pitler, Barbara Plank, Thierry Poibeau, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Siva Reddy, Georg Rehm, Michael Rießler, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Davide Rovati, Valentin Roșca, Olga Rudina, Shoval Sadde, Shadi Saleh, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Baiba Saulīte, Yanin Sawanakunanon, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Muh Shohibussirri, Dmitry Sichinava, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Isabela Soares-Bastos, Antonio Stella, Milan Straka, Jana Strnadová, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Dima Taji, Yuta Takahashi, Takaaki Tanaka, Isabelle Tellier, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Veronika Vincze, Lars Wallin, Jonathan North Washington, Seyi Williams, Mats Wirén, Tsegay Woldemariam, Tak-sum Wong, Chunxiao Yan, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Amir Zeldes, Daniel Zeman, Manying Zhang, and Hanzhi Zhu. 2018. Universal Dependencies 2.2. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth International Conference on Language Resources and Evaluation (LREC'20)*. European Language Resources Association (ELRA).

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A Universal Part-of-Speech Tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2089–2096, Istanbul, Turkey, May. European Languages Resources Association (ELRA).

Martin Popel, Zdeněk Žabokrtský, and Martin Vojtek. 2017. Udapi: Universal API for Universal Dependencies. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 96–101, Gothenburg, Sweden, May. Association for Computational Linguistics.

Rudolf Rosa and Zdeněk Žabokrtský. 2015. KLcpos3 - a Language Similarity Measure for Delexicalized Parser Transfer. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 243–249, Beijing, China, July. Association for Computational Linguistics.

Daniel Zeman, Ondřej Dušek, David Mareček, Martin Popel, Loganathan Ramasamy, Jan Štěpánek, Zdeněk Žabokrtský, and Jan Hajič. 2014. HamleDT: Harmonized Multi-Language Dependency Treebank. *Language Resources and Evaluation*, 48(4):601–637.

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21, Brussels, Belgium, October. Association for Computational Linguistics.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Noëmi Aepli, Željko Agić, Lars Ahrenberg, Gabrielė Aleksandravičiūtė, Lene Antonsen, Katya Aplonova, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Victoria Basmov, Colin Batchelor, John Bauer, Sandra Bellato, Kepa Bengoetxea, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Kristina Brokaitė, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Savas Cetin, Fabricio Chalub, Jinho Choi, Yongseok Cho, Jayeol Chun, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Hanne Eckhoff, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Nizar Habash, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Kim Harris, Dag Haug, Johannes Heinecke, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Jena Hwang, Takumi Ikeda, Radu Ion, Elena Irimia, Ọlájídé Ishola, Tomáš Jelínek, Anders Johannsen, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Kamil Kopacewicz, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Lucia Lam, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Cheuk Ying Li, Josie Li, Keying Li, KyungTae Lim, Maria Liovina, Yuan Li, Nikola Ljubešić, Olga Loginova, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Tomohiko Morioka, Shinsuke Mori, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima

Nitisaroj, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ̀ Olúòkun, Mai Omura, Petya Osenova, Robert Östling, Lilja Øvrelid, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Guy Perrier, Daria Petrova, Slav Petrov, Jason Phelan, Jussi Piitulainen, Tommi A Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Davide Rovati, Valentin Roșca, Olga Rudina, Jack Rueter, Shoval Sadde, Benoît Sagot, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Muh Shohibussirri, Dmitry Sichinava, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Antonio Stella, Milan Straka, Jana Strnadová, Alane Suhr, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Takaaki Tanaka, Isabelle Tellier, Guillaume Thomas, Liisi Torga, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Amir Zeldes, Manying Zhang, and Hanzhi Zhu. 2019. Universal Dependencies 2.5. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

A Appendix A: θ_{pos} Scores for UDv2.5 Treebanks, Annotated to Mark Consistent and Inconsistent Treebanks

This appendix lists the θ_{pos} scores in the UDv2.5 data (Zeman et al., 2019) with the annotations used as per Table 11. In the listing of scores, small treebanks where the total number of sentences is 1,000 or less are not included.

Color	Significance
Red	Inconsistent in POS Annotation
Green	Consistent in POS Annotation
Gray	Could Not Be Estimated

(a) Color Codes Used for Scores

Superscript	Significance
Asterisk ($*$)	Cannot split into constituent genres
Dagger ($\dagger$)	Adulterant Genre(s) Present

(b) Superscripts against Treebank Names

Table 11: Annotations Used in Table 13

Treebank1	Treebank2	θ_{pos}
Ancient_Greek-Perseus	Ancient_Greek-PROIEL	4.641
Arabic-NYUAD	Arabic-PADT	2.497
Dutch-Alpino	Dutch-LassySmall	0.664
Chinese-GSD	Chinese-HK	1.958
Estonian-EDT	Estonian-EWT	0.413
Finnish-FTB	Finnish-TDT	1.195
*Galician-CTG	Galician-TreeGal	0.714
Japanese-BCCWJ	*Japanese-GSD	0.951
*Korean-GSD	Korean-Kaist	2.56
Polish-LFG	*Polish-PDB	0.623
Portuguese-Bosque	*Portuguese-GSD	0.678
Romanian-Nonstandard	Romanian-RRT	1.233
†Slovenian-SSJ	Slovenian-SST	2.405
Spanish-AnCora	Spanish-GSD	0.352
Swedish-LinES	Swedish-Talbanken	0.443
Turkish-GB	Turkish-IMST	1.477

German	***GSD**	***HDT**
***HDT**	0.49	-
LIT	1.383	1.1

Latin	**ITTB**	†**Perseus**
†**Perseus**	1.106	-
PROIEL	3.763	3.901

Norwegian	**Bokmaal**	**Nynorsk**
Nynorsk	0.095	-
NynorskLIA	2.291	2.375

Czech	**CAC**	**CLTT**	**FicTree**
CLTT	1.453	-	-
FicTree	1.138	2.657	-
PDT	0.373	1.935	1.006

Russian	***GSD**	†**Taiga**
†**Taiga**	1.027	-
SynTagRus	0.567	0.631

English	**EWT**	**GUM**	**LinES**	**ParTUT**
GUM	0.26	-	-	-
LinES	0.407	0.455	-	-
ParTUT	0.62	0.432	0.581	-
ESL	0.592	0.799	0.564	0.823

French	†**FQB**	***GSD**	†**ParTUT**	**Sequoia**	**Spoken**
***GSD**	1.582	-	-	-	-
†**ParTUT**	1.942	0.683	-	-	-
Sequoia	1.693	0.248	0.524	-	-
Spoken	3.644	3.089	2.599	2.732	-
FTB	2.226	0.379	0.7	0.272	3.507

Italian	**ISDT**	**ParTUT**	***VIT**	**PoSTWITA**
ParTUT	0.133	-	-	-
***VIT**	0.121	0.194	-	-
PoSTWITA	1.67	1.478	1.764	-
TWITTIRO	1.501	1.376	1.594	0.347

Table 13: θ_{pos} Scores in UDv2.5 Marked for Consistency or Inconsistency in POS Annotation

Table 14 marks the Θ_{pos} limit for treebanks that were marked as inconsistent in the table above. We omit the Θ_{pos} limit for Ancient_Greek treebanks, since the reported θ_{pos} score for the treebanks in the language exceed the hard limit of 4.0.

Treebank Pair	θ_{pos}	Θ_{pos}	**Comments**
Arabic-NYUAD & Arabic-PADT	2.497	0.5	Same Genre Violation of Equation 4
Czech-CAC & Czech-CLTT	1.453	1.388	No Adulterant Genre Violation of Equations 4, 7
Czech-CLTT & Czech-FicTree	2.657	2.0	One Genre Each Violation of Equation 5
Czech-CLTT & Czech-PDT	1.935	1.688	No Adulterant Genre Violation of Equation 7
Finnish-FTB & Finnish-TDT	1.195	1.187	No Adulterant Genre Violation of Equations 4, 7
French-FTB & French-Spoken	3.507	2.0	One Genre Each Violation of Equation 5
French-Sequoia & French-Spoken	2.732	2.0	No Adulterant Genre Violation of Equations 5, 7
Latin-ITTB & Latin-PROIEL	3.763	1.25	No Adulterant Genre Violation of Equations 4, 5, 7
Latin-Perseus & Latin-PROIEL	3.901	3.625	Adulterant Genre Violation of Equations 4, 5, 7
Norwegian-Bokmaal & Norwegian-NynorskLIA	2.291	2.0	No Adulterant Genre Violation of Equations 5, 7
Norwegian-Nynorsk & Norwegian-NynorskLIA	2.375	2.0	No Adulterant Genre Violation of Equations 5, 7

Table 14: Comparison of θ_{pos} Score and Θ_{pos} Limit for Pairs of Treebanks Marked as Inconsistent in Table 13

There are a few important points that need to be specified here:

1. The affiliation of individual sentences in any given treebank is optional and not standardized. If the `README.md` file associated with a treebank in question does not specify how to split the treebank into the constituent genres, the information can be queried through the data providers of the treebank in question. Turkish-IMST could not be assessed for the annotation consistency with the other Turkish treebank as the information on their genre split could not be fetched through either source.

2. While the methods that we discussed can be applied for estimations across different guidelines, care must be taken while estimating the empirical upper bound for a new guideline. If the estimated value of Θ_{pos} is too large, we run the risk of saying the treebanks are harmonious even when they might not be. Also, if the value is too small, we could be overlooking at the effect of domain change and dataset size, to mistakenly announce the pair of treebanks as being non-harmonious to each other.

B Appendix B: Working Example to Mark Pair of Treebanks as Consistent or Inconsistent in POS Annotation

We demonstrate the calculation of Θ_{pos} in the case of the Latin-ITTB and Latin-PROIEL treebanks. Neither of them contains any adulterant genre. The sentence and word count statistics for the two treebanks can be seen in Table 15. The calculated θ_{pos} scores across genres in the two treebanks are shown in Table 16.

Treebank (A)	**Genre (x)**	$size(A_x)$	$TotalSyntacticWords(A_x)$	$AvgSentLen(A_x)$
Latin-ITTB	*nonfiction*	21,011	353,035	16.802
Latin-PROIEL	*nonfiction*	6,626	90,600	13.673
Latin-PROIEL	*bible*	11,785	109,563	9.297

Table 15: Statistics of constituent genres in Latin-ITTB and Latin-PROIEL

$TreebankA_{GenreA}$	$TreebankB_{GenreB}$	$\theta_{pos}(TreebankA_{GenreA}, TreebankB_{GenreB})$
Latin-ITTB$_{nonfiction}$	Latin-PROIEL$_{bible}$	3.702
Latin-ITTB$_{nonfiction}$	Latin-PROIEL$_{nonfiction}$	3.558
Latin-ITTB$_{nonfiction}$	Latin-PROIEL$_{nonfiction,bible}$	3.763

Table 16: Calculated θ_{pos} for different genres in Latin-ITTB and Latin-PROIEL

From Table 16, we notice

1. $\theta_{pos}(\text{Latin-ITTB}_{nonfiction}, \text{Latin-PROIEL}_{nonfiction}) = 3.558 > 0.5$, which is a violation of Equation 4

2. $\theta_{pos}(\text{Latin-ITTB}_{nonfiction}, \text{Latin-PROIEL}_{bible}) = 3.702 > 2.0$, which is a violation of Equation 5

Given the θ_{pos} score calculations, we can estimate the Θ_{pos} threshold in accordance with Equation 6 as follows:

$$\begin{aligned}
\theta_{pos}(\text{Latin-ITTB}_{nonfiction}, \text{Latin-PROIEL}_{bible}) &= 3.702 \\
\theta_{pos}(\text{Latin-ITTB}_{nonfiction}, \text{Latin-PROIEL}_{nonfiction}) &= 3.558 \\
Average(\theta_{pos}) &= \frac{3.558 + 3.702}{2} \\
&= 3.63 \\
\Theta'_{pos}(\text{Latin-ITTB}_{nonfiction}, \text{Latin-PROIEL}_{nonfiction,bible}) &= \frac{0.5 + 2.0}{2} \\
&= 1.25
\end{aligned}$$

$$\boxed{\Theta_{pos}(\text{Latin-ITTB}_{nonfiction}, \text{Latin-PROIEL}_{nonfiction,bible}) = Minimum(Average(\theta_{pos}),\ \Theta'_{pos},\ 2.0) = 1.25}$$

We observe that the calculated θ_{pos} score exceeds the estimated Θ_{pos} threshold, thereby judging the pair of treebanks as **inconsistent** in their POS annotation.

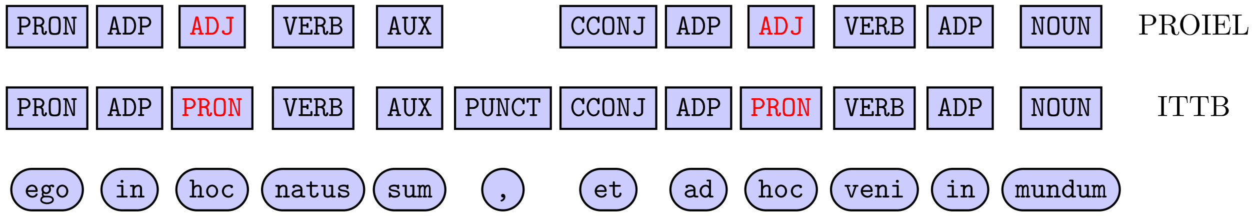
To further validate that the two treebanks are not consistent in their POS annotation, consider the following sentence present in either treebank.[11] The difference in annotation is shown beneath the example.

[11] Latin-IITB contains the sentence as such, without any modifications, while the sentence in Latin-PROIEL is without punctuation marks.

(1) *ego in hoc natus sum , et ad hoc veni in mundum , ut testimonium perhibeam*
I in this born am , and to this I came in world , that testimony I bestow

veritati .
to truth .

‘*I was born, and for this I came into the world, to testify to the truth.*’



The Latin-PROIEL treebank's lack of punctuation marks is also well reflected in its trigram distribution. Table 17 shows the 10 most frequent POS trigrams in different Latin treebanks listed in order of their frequency in the corresponding treebank.

Latin-PROIEL		**Latin-Perseus**		**Latin-ITTB**	
POS Trigram	**Freq (%)**	**POS Trigram**	**Freq (%)**	**POS Trigram**	**Freq (%)**
NOUN VERB #	1.06	VERB PUNCT #	4.65	NOUN PUNCT #	2.086
VERB ADP NOUN	1.005	NOUN VERB PUNCT	3.604	VERB PUNCT #	1.976
NOUN CCONJ NOUN	0.843	NOUN PUNCT #	2.114	NOUN VERB PUNCT	1.702
NOUN NOUN VERB	0.787	NOUN NOUN VERB	1.541	VERB ADP NOUN	1.374
ADP NOUN VERB	0.77	VERB NOUN PUNCT	1.469	ADP NOUN PUNCT	1.104
# CCONJ VERB	0.735	ADJ NOUN VERB	1.174	NOUN NOUN PUNCT	0.993
NOUN ADP NOUN	0.726	ADJ VERB PUNCT	1.095	NOUN ADP NOUN	0.844
ADP NOUN NOUN	0.692	VERB VERB PUNCT	1.081	NOUN ADJ PUNCT	0.836
ADJ NOUN VERB	0.615	VERB NOUN NOUN	0.988	ADP NOUN NOUN	0.811
ADP NOUN ADJ	0.606	NOUN VERB NOUN	0.982	ADJ NOUN PUNCT	0.772

Table 17: Most Frequent POS Trigrams in Different Latin Treebanks with Frequency Percentage
Note: # denotes the POS of sentence boundary token

From the table, the reason of Latin-PROIEL treebank being inconsistent in annotation with the other two is clear. While the POS tag associated with punctuation (PUNCT) contributes to at least 6 of the top 10 trigrams in Latin-Perseus and Latin-ITTB, the POS tag (and therefore the trigrams) is missing in Latin-PROIEL.

Intelligenti Pauca
Probing a Novel Alternative to Universal Dependencies for Under-Resourced Languages on Latin

Daniel Couto-Vale

Kurfürstenstr. 148
10785 Berlin, Germany
danielvale@icloud.com

Konstantin Schulz
Humboldt University Berlin
Unter den Linden 6
10099 Berlin, Germany
konstantin.schulz@hu-berlin.de

Abstract

In this paper, we aim at improving the study of Latin in three ways: 1) by providing better visualizations of syntagma and structure for both research and the classroom, 2) by supporting a high-level search interface for corpus exploration, and 3) by improving the accuracy of taggers and parsers. To achieve this, we introduce a new linguistic description called Intelligenti Pauca, an alternative to Universal Dependencies for under-resourced languages. We show the key differences between the two linguistic descriptions, how the structure of Intelligenti Pauca favours our goals, and the effect it has on parsing accuracy for the Index Tomisticus Treebank.

1 Motivation

For Latin and Ancient Greek, researchers want to search for words and grammatical structures and view word features such as class and inflections (Monachini et al., 2018). Meanwhile, Latin and Greek teachers frequently make use of tools for visualizing and exploring grammatical structures in the classroom (Ellis, 2009; Mambrini, 2016; Augustinus et al., 2017; Guibon et al., 2020).

Annotated text corpora were built for implementing components for such tools including taggers, parsers, and searches (Abeillé, 2012, xiv), resulting in three dependency treebanks (Vincze et al., 2010, 1855): the Index Thomisticus Treebank (ITTB) (Passarotti, 2019), the Pragmatic Resources of Old Indo-European Languages (PROIEL) (Haug and Johndal, 2008), and the Ancient Greek and Latin Dependency Treebank (AGLDT) (Bamman and Crane, 2011). However, current tools for Latin present three issues: structures are 1. not highlighted, 2. unrelated to meaning (Khalili and Auer, 2013), and 3. often wrong (Monachini et al., 2018, 4), which is a problem for teaching (Müller and Oeste-Reiß, 2019, 59).

At the first frontier, attempts were made to represent features and grammatical structures graphically: e.g. adding information to a concordance line (Fischl and Scharl, 2014, 194) and showing a dependency tree for a sequence of words as in Figure 1, which reads «*However, women love chocolate desserts.*».

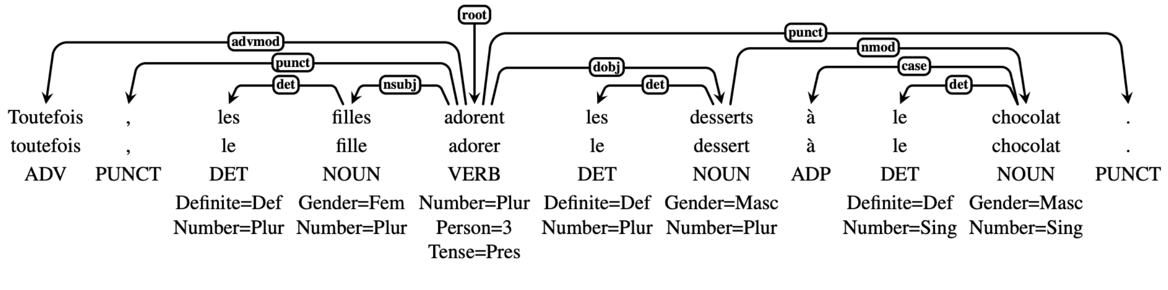

root
advmod
punct
det
nsubj
dobj
det
punct
nmod
case
det
Toutefois , les filles adorent les desserts à le chocolat .
toutefois , le fille adorer le dessert à le chocolat .
ADV PUNCT DET NOUN VERB DET NOUN ADP DET NOUN PUNCT
Definite=Def Number=Plur
Gender=Fem Number=Plur
Number=Plur Person=3 Tense=Pres
Definite=Def Number=Plur
Gender=Masc Number=Plur
Definite=Def Number=Sing
Gender=Masc Number=Sing

Figure 1: Visualization for Universal Dependencies (Nivre et al., 2016, 1660).

These visualizations are unsuitable for schools because they do not highlight grammatical structure in any way: e.g. frames or boundary markers. Highlighting is necessary to make structure observable (Arneson and Offerdahl, 2018, ar7,1), thus making it accessible to visual learners (Pitta-Pantazi et al., 2013, 201) and easier to process for all learner types (Kollöffel, 2012, 704). Besides, the grammatical structures here do not correspond enough to semantic structures for such a highlighting to be useful. In Section 2, we shall show how highlighting can be achieved and shallow semantic information displayed.

At the second frontier, Latin researchers need high-level searches for grammatical structures whereby they can answer their questions: a linguist may want to verify if 'medium-receptivity' ('middle' voice) as in *movetur* (*it moves*) and *movetur ab alio* (*it moves due to something else*) was the original meaning of *tur*-eding verbs; a theologian may be collecting evidence that a particular author assumed that three people emanate from God; a historian may be interested in how the actions of legates affected soldiers' morale; and a sociologist may want to know the relation between the origin of people's names and Roman identity. However, current search tools operate at a far too low level for them. In Section 4, we illustrate how to support high-level structural search for evidetiating such hypotheses.

Finally, we face an issue when improving parsing accuracy for historical languages: corpora will never increase. Here we must either improve generalization methods, annotations, or annotate more extant texts. In this paper, we focus on annotation improvement for better parsing accuracy.

Aiming at advances at these frontiers, we propose **Intelligenti Pauca** (IP), an alternative linguistic description to **Universal Dependencies** (UD, Nivre et al. (2016)), among others such as **Stanford Dependencies** (De Marneffe and Manning, 2008), based on a theory of ideational semantics by Halliday and Matthiessen (1999)[1]. Like UD, IP relies on dependency structures, not phrase structures, but it adds back a feature from the latter: the rank (Section 2.1), thus enabling visualization of grammatical structures (Goal 1) and corpus exploration (Goal 2). We converted UD annotations into IP. Since we needed to let dependency rules be learned from fewer examples (Goal 3), we aimed at reducing the number of rules that need to be learned for a particular labelled attachment (Goal 3.1) and reducing the number of features a rule is grounded on (Goal 3.2).

2 Intelligenti Pauca

Nivre's visualization does not highlight the grammatical structure and the amount of information it displays at once is far too great for the classroom. To improve it, we should aim at minimal intervention. The first step is to hide the structure and let only one layer of features visible without abbreviations as in Figure 2 (Goal 1). Structure and other features should be displayed only when needed[2,3]. The syntagma is highlighted by a frame, making it easier to understand for learners (Todi et al., 2018, 556).

cum	*ipse*	*deus*	*sit*	*nostrae*	*auctor*	*naturae*	.
conjunction	noun	noun	verb	noun	noun	noun	punctuation

Figure 2: Syntagma (ITTB, 198, 1)
«*since God himself is the creator of our nature.*»

However, this does not solve the whole issue. Grammatical structure must be highlighted and it must be meaningful. To achieve this, we can highlight the structure by framing it and reduce the number of dependencies shown at once, leaving only those that are related to each other semantically. In this way, we emphasize one aspect of meaning at a time, guiding viewers to comprehension (Goal 1). For instance, Figure 3 shows a structure with a lexical verb and two arguments. Here *Marker*, *Identified*, and *Identifier* are dependency labels and *Process* is the type of semantic element represented by the lexical verb.

Marker	Identified		Process	Identifier			Marker
cum	*ipse*	*deus*	*sit*	*nostrae*	*auctor*	*naturae*	.
conjunction	noun	noun	verb	noun	noun	noun	punctuation

Figure 3: Syntagma + Structure (ITTB, 198, 1)
«*since God himself is the creator of our nature.*»

[1] The available features and functions in the IP description are systematized in a SYS description, which can be imported as data into a database by a SYS description interpreter, also made available (JAR Scripts).

[2] Examples are referenced as (corpus, sentence id, word id).

[3] Some of the features such as 'seams' to be presented in Chapter 2.2 should be avoided in the classroom because they are meant to support the parsing mechanism and not to support teaching.

In this figure, we show only a selection of the dependencies and we provide labels from Halliday's theory of experiential semantics (Halliday and Matthiessen, 1999), which are more meaningful than the ones currently in use: namely, *Mark*, *Nsubj*, *Cop*, *Punct*. The resulting tabular visualization is easier to understand. Next, we explain how this visualization can be achieved with a dependency structure.

2.1 The rank

One way to reduce dependencies shown at once is to add **ranks** to dependency structures (Halliday, 1966). Ranks function as tags for grammatical units, indicating the type of phenomena units represent. There are three types of phenomena: figures are represented by **clauses**, sequences by **clause complexes**, and elements by **groups** and **phrases** (Halliday and Matthiessen, 1999, 48-49).

To add ranks to dependency structures, there must be an alignment between grammatical and semantic heads. In IP, auxiliary verbs such as *is* in *is coming* (*est* in *locutus est*) depend on the lexical verb 'in a verbal group' (Halliday and Matthiessen, 2014, 398) and other words depend on that verb 'in a clause' (Halliday and Matthiessen, 2014, 220). Participle verbs as in *the one moving* and *the one moved* (*illud movens* and *illud motum*) constitute a clause embedded 'in a nominal group' (Halliday and Matthiessen, 2014, 127). The same applies to other verbs linked to relative pronouns. Finally, lexical verbs depend on one another 'in a clause complex' (Halliday and Matthiessen, 2014, 428). This enables different visualizations: clause complexes as in Figure 4, clauses as in Figure 5, and groups as in Figure 6.

Extended					Extending					
veritatem	*meditabitur*	*guttur*	*meum*	,	*et*	*labia*	*mea*	*detestab...*	*impium*	.

Figure 4: Clause complex (ITTB, 2, 1)
«*My throat will judge the truth, and my lips will hate the wicked.*»

Phen.	Process	Senser		Marker
veritatem	*meditabitur*	*guttur*	*meum*	,

Marker	Senser		Process	Phen.	Marker
et	*labia*	*mea*	*detestab...*	*impium*	.

Figure 5: Clauses (ITTB: 2, 1; 2, 12)
«*My throat will judge the truth,*» «*and my lips will hate the wicked.*»

Thing	Possessor
guttur	*meum*

Thing	Possessor
labia	*mea*

Figure 6: Composed groups (ITTB: 2, 3; 2, 7)
«*my throat*» «*my lips*»

Ranked dependency structures differ from phrase structures because they can be discontinuous, thus there is no need to reconstruct word 'movements' (Mahajan, 2003, 218). However, both ranked dependents and phrases are semantic constituents, whereas non-ranked dependents are not. Given that discontinuities are frequent in Latin and Ancient Greek (Mambrini and Passarotti, 2012, 136), ranked dependency structures do not face the same challenges for these languages as phrase structure. Statistical dependency parsing with UDPipe (Straka et al., 2016) can produce ranked dependency structures as well as combinatory categorial parsing with OpenCCG (Bozsahin et al., 2005) and other parsing strategies.

2.2 Nouns, adjectives, numbers

Dependencies can be learned better if words share features in similar structures (Kübler and Hinrichs, 2001), especially **word classes** (Alfared and Béchet, 2012). Currently, dependencies in ITTB do not reflect meaning and word classes do not favour rule learning. IP solves these two issues by anchoring word classes onto types of represented elements (Goal 3.1). If the element being represented is a **simple thing** such as *guttur* (*the throat*) or *ego* (*me*), the word is a **noun**. Figures 7 and 8 illustrate the difference.

Head	Nmod
intelligere	*dei*
verb	propn
–	genitive

Head	Nmod
intelligere	*eius*
verb	pron
–	genitive

Amod	Head
suum	*intelligere*
adj	verb
nominative	–

Figure 7: UD – Modifiers (ITTB: 2049, 2; 2077, 12; 2050, 11)
«*God's intelligence*» «*his intelligence*» «*his intelligence*»

Thing	Possessor
intelligere	*dei*
noun	noun
–	genitive

Thing	Possessor
intelligere	*eius*
noun	noun
–	genitive

Possessor	Thing
suum	*intelligere*
noun	noun
genitive	–

Figure 8: IP – Possessors (ITTB: 2049, 2; 2077, 12; 2050, 11)
«*God's intelligence*» «*his intelligence*» «*his intelligence*»

In IP all **pronouns**, **proper nouns**, and **common nouns** are nouns because they represent things. Noun class is an extra feature. Pronouns such as *meum*, *tuum*, and *suum* (*my*, *your*, *his/her*) are pronouns, thus nouns, which differs from tradition (Oniga and Schifano, 2014, 95), and they are 'genitive' like other nouns with the same function (Rubenbauer and Heine, 2012, 54). They have a secondary case agreeing with the case of the modified noun, like adjectives do (Priscianus, 2010, 207). Agreement features are annotated as **seams** for both adjectives and such 'genitive' nouns inflected like adjectives. This lets the *Possessor* rule be heavily grounded on word class, subclass, and case (Goal 3.2).

In turn, **adjectives** represent additional **qualities** for simple things. Some function as classifiers in the nominal group (Halliday and Matthiessen, 2014, p.383), representing a more specific class of things than the noun represents on its own. This is the case of *pigmentaria* in *arte pigmentaria* (*the art of solution mixing*). Oftentimes, such a compound is synonymous to a noun such as *pigmentariae* (*the art of solution mixing*). UD treats those nouns as adjectives (see Figure 9), IP does not (see Figure 10). In turn, this separation between nouns and adjectives lets rules such as the *Classifier* rule be heavily grounded on word classes and subclasses (Goal 3.2).

Head	Amod
arte	*pigmentaria*
noun	adj

Head
pigmentariae
adj

Figure 9: UD – Modifiers & Heads (ITTB: 10, 4; 10, 26)
«*the art of solution mixing*» «*the art of solution mixing*»

Thing	Classifier
arte	*pigmentaria*
noun	adjective

Thing
pigmentariae
noun

Figure 10: IP – Classifiers & Things (ITTB: 10, 4; 10, 26)
«*the art of solution mixing*» «*the art of solution mixing*»

Thirdly, **numbers** such as *unus* (*one*), *primus* (*first*), *simplex* (*simple*), and so on represent a **quantity**. In UD, non-cardinal numbers are treated as adjectives. This poses an issue for the parsing of compound numbers such as *vigenti et unus* (*twenty one*), *vicesimus primus* (*twenty first*), *vigentuplex simplex* (*with twenty one parts*) and the like because rules cannot be learned across compounds in different number classes (see Figures 11).

Nummod	Amod	Amod	Head
unum	*simplex*	*suum*	*esse*
num	adj	adj	noun

Figure 11: UD – Modifiers (ITTB, 1482, 6)
«*his one simple being*»

Quantifer	Multiplier	Possessor	Thing
unum	*simplex*	*suum*	*esse*
number	number	noun	noun

Figure 12: IP – Quantifiers & Multipliers (ITTB, 1482, 6)
«*his one simple being*»

To solve this, in IP all numbers count as numbers as shown in 12. Numbers have different functions — e.g. Quantifier, Ordinator, Multiplier — depending on their class. Besides number class, numbers also carry features for modulo and house: e.g. *unus* (*one*) is a ‘cardinal’ ‘decimal’ ‘one’ and *vicesimus* (*twentieth*) is an ‘ordinal’ ‘decimal’ ‘ten’. These features enable compounding rules for different decimal houses within a numeric group and it enables different functions for different number classes (Goal 3.2). Figure 13 illustrates a compound quantity group in Latin.

Thousand				Hundred	Ten	Quantity
Hundred	Ten	Unit	House			
ducenta	*viginti*	*duo*	*milia*	*ducenti*	*viginti*	*unus*
number	number	number	number	number	number	number
hundreds	tens	units	thousands	hundreds	tens	units

Figure 13: IP – House, Hundred, Ten, Unit
«*two hundred twenty two thousand two hundred twenty one*»

In short, IP offers meaningful functions such as Classifier, Quantifier, Multiplier, and Possessor (also Ordinator, Deictic, Epithet) where UD offers only Modifier (Nmod, Amod, Nummod). In IP, word classes coincide with element types, which limits the number of rules (Goal 3.1), and they determine potential functions together with a small set of other features such as subclasses and cases (Goal 3.2).

2.3 Verbs

Transitivity A clause represents a figure composed of a process, participants, and circumstances (Halliday and Matthiessen, 1999, 128-172). The **lexical verb** represents the **process** in which things, qualities, and quantities take part. Let us consider the lexical verbs *habet* and *sit* in Figures 14 and 15.

Carrier	Marker	Process...	Attributor	...Process	Attribute	Marker
hoc	*autem*	*habet*	*aristoteles*	*pro*	*impossibili*	,
noun	adverb	verb	noun	adposition	adjective	punctuation

Figure 14: Transitive attributive clause (ITTB, 457, 1)
«*however, that was considered impossible by Aristotle,*»

Marker	Attribute	Process	Carrier		Marker
ut	*vehemens*	*sit*	*gaudium*	*eius*	.
conjunction	adjective	verb	noun	noun	punctuation

Figure 15: Intransitive attributive clause (ITTB, 154, 23)
«*that his joy is enourmous.*»

In these examples, *impossibili* (*impossible*) and *vehemens* (*enormous*) are attributes carried by, respectively, *hoc* (*that*) and *gaudium eius* (*his joy*). In turn, *Aristoteles* (*Aristotle*) is the person who attributes a quality to something. In IP, participant roles as in *Attribute*, *Carrier*, and *Attributor* are labelled instead of *Xcomp*, *Cop*, *Obj*, and *Nsubj*, thus enabling a visualization that guides readers towards a reasonable interpretation of transitivity (Goal 1) and high-level exploration of a corpus (Goal 2).

Verbal group Every time two or more verbs represent a single process, the lexical verb represents a process with participants and the others are **auxiliary verbs** (Halliday and Matthiessen, 2014, 396). Figure 16 contains such a verbal group with two verbs.

In Figure 16, the verb *est* (*must*) does not agree with the quantity of Actors nor with their role in speech. In addition, the Actor is represented by a genitive noun, not a nominative one typical of Actors (Menge et al., 2012, 383). This structure resembles that of more typical clauses with *ordinare* (*put order*), which shows that it is grounded more heavily on word classes such as nouns and lexical verbs than on inflectional features. On the one hand, the similarity in experiential semantics is an obvious improvement for visualization (Goal 1) and exploration (Goal 2). On the other, fewer rules (Goal 3.1) over fewer more general features (Goal 3.2) have a positive impact in parsing accuracy.

Marker	Actor	Process		Marker
		Auxiliary	Process	
quod	*sapientis*	*est*	*ordinare*	.
conjunction	noun	verb	verb	punctuation

Figure 16: Verbal group (ITTB, 5, 13)
«because the wise must put order»

Tense/mode Verbal groups can represent past, present, and future processes, the three **primary tenses** relative to 'now' (Halliday and Matthiessen, 1999, 214), in one of a few different **clause-linkage modes** (Whorf, 1956, 186). In free clauses, processes are placed in time in the injunctive mode as in the first column of Table 1. In bound clauses, clause-linkage modes realize types of logical relation together with conjunctions. Table 1 systematizes three modes of construing tense in Latin: here *ut* and *dum* are representatives of conjunctions used with conjunctive modes. Secondary tense (Halliday and Matthiessen, 1999, 399) such as 'past in past' in *moverat* (*had moved*) are left out for simplicity. Latin past verbs that oppose each other textually and interpersonally (Aerts, 2018) are placed in the same cell.

		conjunctive	
	injunctive	*ut*	*dum*
past	*movit, movebat, movet*	*moveret*	*movet*
present	*movet*	*moveat*	
future	*movebit*		

Table 1: Modes of construing primary tense in ITTB

Since these patterns are not covered by UD, current tools and components cannot determine primary tense. The root is also missed out because there are no features in UD for clause-linkage modes. In IP, this issue is solved by replacing traditional features by semantic and grammatical features, the latter being divided into group, word, and morpheme features. Table 2 shows morphemic features.

Verb	Aspect	Branch	Leaves	Verb	Aspect	Branch	Leaves
move ba t	*ō*	*bā*	*t*	*move ba t ur*	*ō*	*bā*	*tur*
move t o	*ō*	*ō*	*t*	*move t o r*	*ō*	*ō*	*tur*
move re	*ō*	*re*	–	*move ri*	*ō*	*rī*	–
mov it	*ī*	–	*it*	*mot um*	*ū*	–	*um*

Table 2: Stem aspect, branch, and leaves

There are three morpheme classes (Rubenbauer and Heine, 2012, 66-71): **Stem**, **Branch**, and **Leaf**. The available leaves depend on the selected branch, and the available branches depend on the selected stem aspect (Oniga and Schifano, 2014, 111). At the group rank, mode is partially determined by other words around the verbal group. For instance, if «*move t*» follows *dum*, it is *dum*-conjunctive, otherwise injunctive, a task modern taggers can do. Once a particular mode of construing primary tense is established, a primary tense can usually be determined solely based on the selection of verbs. This allows visualization of the tense (Goal 1) and searches for processes in particular primary tenses (Goal 2). Moreover, a parser can use the verbal modes in a verbal group together with conjunctions surrounding them to assess the chances that a particular lexical verb is the root of a dependency tree (Goal 3.2).

Finiteness In Latin, participants interacting in the dialogue such as *ego* (*I*) and *tu* (*you*) are usually left **implicit** if they are the subject (Oniga and Schifano, 2014, 209-213) (Rubenbauer and Heine, 2012, 115-116) and things that take part in two consecutive processes are left **elided** in the second clause (Kühner, 1879, 1042). **Finite** bound clauses are those that follow this pattern of implicitness and elision whereas **non-finite** bound clauses are those for which one participant is necessarily elliptic (Halliday and Matthiessen, 2014, 477). In Figure 17, we see three examples of non-finite bound clauses.

Marker	Phen.	Process
ad	*deum*	*cognoscendum*
adp.	noun	verb

(a) Non-finite verb seamed to *deum*

Marker	Phen.	Process
ad	*divina*	*cognoscenda*
adp.	noun	verb

(b) Non-finite verb seamed to *divina*

Marker	Process
ad	*ostendendum*
adp.	verb

(c) Unseamed non-finite verb

Figure 17: Non-finite bound clauses (ITTB: 121, 10; 238, 8; 563, 10)
«to know God» «to know the divine» «to show»

In UD, unseamed verbs such as *ostendendum* are 'gerunds' and seamed verbs such as *cognoscendum* are 'gerundives' and there is no feature that both have in common despite the fact that both gerunds and gerundives are *nd*-branch verbs. For every two rules that emerge from the examples in UD, IP lets one emerge by ascribing an *nd*-branch feature to these verbs (Goal 3.1). Departing from tradition (Rubenbauer and Heine, 2012, 202), it also makes the dependency between participants and processes be the same as in finite clauses, letting a single rule emerge from both finite and non-finite clauses.

Agreement The need for examples is further contained in IP by replacing original features (case, number, gender, person, tense, mode...) by word features for seam (agreement feature), and **foliage**, a set of leaves mapped to seams. Word-rank features result in matrices such as the one shown in Table 3.

	a-foliage	*am*-foliage	*ae-ī*-foliage	*ae-ō*-foliage	*ā*-foliage	
a-am-seam	*dic end a*	*dic end am*	*dic end ae*	*dic end ae*	*dic end a*	
um-um-seam	*dic end um*	*dic end um*	*dic end i*	*dic end o*	*dic end o*	unseamed
us-um-seam	*dic end us*	*dic end um*	*dic end i*	*dic end o*	*dic end o*	
ae-ās-seam	*dic end ae*	*dic end as*	*dic end arum*	*dic end is*	*dic end is*	
a-a-seam	*dic end a*	*dic end a*	*dic end orum*	*dic end is*	*dic end is*	
ī-ōs-seam	*dic end i*	*dic end os*	*dic end orum*	*dic end is*	*dic end is*	

Table 3: Gerunds and gerundives as *nd*-branch verbs

Gerunds and gerundives share the same stem aspect and an *nd*-branch (Rubenbauer and Heine, 2012, 71). In addition, all verbs following the adpositional marker *ad* in non-finite bound clauses have a leaf from the *am*-foliage, if they are seamed, or the *um*-leaf, otherwise.

Realization of conjunction	**Realization of seams**	
ad-conjunctive & seamed > *am*-foliage	*am*-foliage & *a-am*-seam > *am*-leaf	*am*-foliage & *ae-ās*-seam > *ās*-leaf
ad-conjunctive & unseamed > *um*-leaf	*am*-foliage & *um-um*-seam > *um*-leaf	*am*-foliage & *a-a*-seam > *a*-leaf
	am-foliage & *us-um*-seam > *um*-leaf	*am*-foliage & *ī-ōs*-seam > *ōs*-leaf

There is a total of 12 leaves for *nd*-branch verbs, five of which can occur in non-finite bound clauses with the adpositional marker *ad*. Twelve different verbs with two common feature (namely, aspect and branch) is a more general classification than 30 gerundives and 5 gerunds (Goal 3).

Potential seams can be determined based on morphemic features and contextual cues. The foliage can be determined based on the seam, if any, and contextual cues. Here, even if a word-rank tagging mistake is made at seam and foliage, the parser can still rely on the presence of an adposition such as *ad* and on lower-rank morphemic features such as *nd*-branch to determine that this is a non-finite clause. As a result, since the parser will count on fewer (Goal 3.2) more general (Goal 3.1) features, generalization will take place across examples with gerunds and gerundives for seldom adpositional markers.

Embedding Only some adnominal clauses in UD count as embedded clauses, namely those which contribute to reference. Embedded clauses are not logically related to other clauses directly, but rather modify a noun (Halliday and Matthiessen, 2014, 127, 382). In Latin, embedded clauses are either finite and have a 'relative' word[4] (Rubenbauer and Heine, 2012, 285, 287) or they are non-finite and have

[4] 'Relativsatz nach einer Einschränkung bzw. näheren Bestimmung bedürfenden Bezugswort'

a ‘participle’ verb[5] (Rubenbauer and Heine, 2012, 209-211). While in non-finite bound clauses verb foliage construes a type of logical relation together with adpositions, ‘participle’ verbs agree with the modified noun in case, thus they realize a case seam like adjectives do (see Figure 18).

Thing	Qualifier	
	Process	Goal
aliquod	*movens*	*se*
noun	verb	noun

(a) Operative embedded clause

Thing	Qualifier		
	Process		Actor
esse	*motum*	*ex*	*se*
noun	verb	adposition	noun

(b) Goal-receptive embedded clause

Figure 18: Embedded clauses (ITTB: 527, 10; 557, 16)
«*something moving itself*» «*a being moved by itself*»

Currently, the embeddedness of such clauses cannot be represented properly in UD. Nouns such as *aliquod* (*something*) are annotated as adjectival modifiers of verbs such as *movens* (*moving*), which are clausal subjects or objects of other verbs. In turn, nouns such as *esse* (*a being*) are annotated as auxiliaries of verbs such as *motum* (*moved*), which is a clausal subject or object of another verb. This categorial shifting generates instability between word classes and word functions. In IP the instability is reduced by having verbs in embedded clauses annotated as adposition-like modifiers of nouns (Goal 3.1). In this case, embedded clauses function as qualifiers within nominal groups as illustrated above.

Metaphor Finally, we come to the point where grammar ‘folds on itself’ (Halliday and Matthiessen, 1999, 227-293) (Halliday and Matthiessen, 2014, 659-707). We stop referring to *the thing moving* (*hoc movens*) or claiming that *this thing moves* (*hoc movetur*) and we start referring to *the mover* (*motor*) and *his motion* (*motus suum*). Examples of this can be found in Figure 19.

‘Actor’	‘Goal’
Thing	Possessor
motor	*universi*
noun	noun

(a) Actor as thing

‘Process’	‘Medium’	*congruent*
Thing	Possessor	*metaphorical*
motus	*sui*	
noun	noun	

(b) Process as thing

Figure 19: Grammatical metaphor (ITTB: 19, 5; 381, 12)
«*the mover of everything*» «*his motion*»

Parsing results for *the mover of everything* and *his motion* in IP will represent a thing possessed by another (the ‘metaphorical’ structure). Such a parsing result cannot be understood as a direct representation of our experience. In the first example, *the mover* is ‘possessed’ by *everything else* only metaphorically. It actually moves everything else. In the second, *the motion* is a ‘thing’ and is ‘possessed’ by *something* only metaphorically. It is actually a process affecting that thing, the affected medium.

A full analysis must include the ‘congruent’ structure, which we could achieve by carrying out a second parse on nominal groups. This second-level parser should rely not only on grammatical features, but also on the semantic features of the represented elements, such as a further classification of things (‘classified thing’, ‘actor as thing’, ‘process as thing’, etc.) and their functions in the first-level structure (Halliday and Matthiessen, 1999, 278-296). This would guide the second-level parser towards an interpretation of the transitivity packed within such nouns. This second level of interpretation will not be integrated in the initial version of the IP description (1.0), but rather in a subsequent release cycle.

2.4 Cohesive ties

Some word links are not dependencies and are better understood as cohesive ties between constituents of different grammatical units. The clause complex in Figure 20 illustrates two types of cohesive ties.

[5] ‘Attributives Partizip’ in Hofmann et al.’s description.

Actor	Goal	Circum.	Process	Marker	Goal	Circum.	Process
qui	*res*	*directe*	*ordinant*	*et*	*eas*	*bene*	*gubernant*
noun	noun	adverb	verb	conjunction	noun	adverb	verb

Figure 20: Elision & anaphora (ITTB, 4, 17)
«who straighten things up and drive them well»

In the first clause, *qui* (*who*) and *res* (*things*) play the roles of, respectively, Actor and Goal of the action. In the second clause, the actor is elided to avoid repetition. This means that *qui* (*who*) in the first clause plays the role of ElidedActor of *gubernant* (*drive*) in the second. Moreover, *res* (*things*) in the first clause is the Same thing As *eas* (*them*) in the second. Both of these are cohesive ties in IP.

In OWL (Antoniou and van Harmelen, 2004), one can specify inference rules over cohesive ties such as the ones in Table 4 and let reasoners such as FaCT++ (Tsarkov and Horrocks, 2006) or HermiT (Glimm et al., 2014) follow the logical chain for «Actor», «Goal», and «Carrier».

Actor → «Actor»	Goal → «Goal»	Carrier → «Carrier»
ElidedActor → «Actor»	ElidedGoal → «Goal»	ElidedCarrier → «Carrier»
SameAs ∘ «Actor» → «Actor»	SameAs ∘ «Goal» → «Goal»	SameAs ∘ «Carrier» → «Carrier»

Table 4: Inference rules in Protégé SuperPropertyOf syntax

While *qui* (*who*) is the Actor of *ordinant* (*put order*) and the ElidedActor of *gubernant* (*drive*), it is the «Actor» of both. Thus if such inferred functions are stored in a DB, a researcher can search for all actions carried out by a given person, not only for those where the person is mentioned by name in the clause. In turn, this elevates the level at which one can query a corpus structurally (Goal 2).

3 Operations

3.1 Converting treebanks

We specified an SQL schema called **Dependency Base** (DB), which enables multiple analyses to be stored in parallel for the same text (DB Scheme). Since all three treebanks are available as CONLL-U files at LINDAT/CLARIAH-CZ (Universal Dependencies 2.6), we implemented a command line script for importing the text and its UD analysis from a CONLL-U file into a DB (JAR Scripts) and another for exporting an analysis as a CONLL-U file. In this setup, CONLL-U files work as an exchange format.

We specified a language called DUX for implementing conversion scripts for dependencies[6] and we implemented a DUX interpreter as a command line script, which converts a text analysis from a source linguistic description (e.g. UD description) into a target linguistic description (e.g. IP description). The DUX interpreter adds the resulting analysis into the DB as a stand-off annotation (Celano, 2019, 150). Finally, we implemented the conversion script from UD to IP in DUX, which can convert 93% of the ITTB in its current version.

To align grammatical and semantic heads, we needed to swap the direction of some dependencies and changed other structures entirely. Word features are determined by both form and context.

3.2 Creating a better parser

Since a different set of features and functions (dependency labels) exists for IP and UD and words depend on each other differently, we need to compare how easy it is for a parser to learn how to analyze text according to each description. For that purpose, we exported 398 lines of ITTB-train and 198 of ITTB-dev as CONLL-U files for UD and IP descriptions (Parallel Annotation). We compared the two file pairs for 'anchors', a tuple composed of tail class, head class, and function, which allows us to estimate how much evidence there is for each attachment/labelling rule and how many rules there are. For the same corpus segment, UD has roughly twice as many anchors as IP (108:59) and its anchor frequency

[6] `https://github.com/DanielCoutoVale/Dependencies/tree/master/ittb-ip` . It does not produce cohesive ties when converting a dependency treebank.

distribution has a longer tail (see Chart 1). Parsing shows a much better unlabelled attachment score (UAS) and a marginally better labelled one (LAS) (see Table 5).

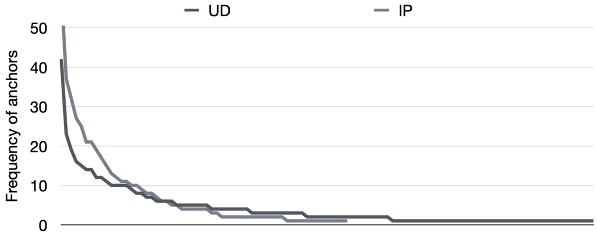


Chart 1: Anchor frequency distribution

	'Golden Tokens'		'Golden POS'	
	UAS	LAS	UAS	LAS
UD	28.40%	17.28%	36.42%	24.07%
IP	39.51%	19.75%	47.53%	26.54%

Table 5: Parsing scores

3.3 Creating a searchable resource

Treebanks are very expensive resources. One can progressively increase treebanks by adding verified parsing results to it, thus saving some time if one has a good parser. However, if this investment is not possible, one can automatically create an IP analysis layer with the parser above for the remaining texts, store the annotations in a DB, and search the DB for the desired syntagmata and structures. With that purpose in mind, we implemented a command line script for querying a DB (JAR Scripts). For generic search and visualization in IP resources, we plan to convert the provided CONLL-U files using Pepper (Zipser and Romary, 2010) and make them available in a public instance of ANNIS (Krause, 2019).

4 Exploring the resource

Once some IP annotations are stored in a DB, researchers can carry out high-level queries for a variety of research questions in different areas of humanities as illustrated in Table 6. Each square bracket stands for a word in the searched structure, the labels within it are word features, and the labels followed by parentheses are links between words.

Linguistics	Theology
Did *or*-foliage 'passives' surpass *or*-foliage 'middles' in Latin? If so, when? (Kulikov and Lavidas, 2013)	How does Thomas Aquinas construe God as a single intelligence coming as three people? (Hillar, 2012)
[or-foliage goal-receptive verb] [or-foliage medium-receptive verb]	[number] [noun] Quantifier(1,2)[7] [number] [noun] Multiplier(1,2)
History	**Sociology**
Which actions carried out by the legates increased and decreased soldiers' morale? (Ureche, 2014)	How did people construe a Roman identity and Latin/Greek origins in Ancient Rome? (Elder, 2019)
[proper-noun] [adjective #legatus] Classifier(2,1)[8] [verb] [noun #Piso] «Actor»(2,1)[9]	[proper-noun] [verb] [noun #Corpus] «Carrier»(2,1)

Table 6: Research questions and corresponding corpus queries[10]

For UD-annotated corpora, there is no simple equivalent way to achieve this. For instance, there is no feature for the class of *or*-foliage verbs, no feature for non-cardinal numbers, no set of dependency labels and features associated with the roles of Actor and Carrier. For these questions, the regex-enabled search field found in web browsers might be a more suitable tool than a structural search in a UD treebank.

5 Conclusion

IP is a linguistic description based on Halliday's account of ideational semantics. In this paper, we showed that IP is more suitable than UD for three purposes: 1. visualizing syntagma and structure, 2.

[7] Views all numbers in context representing a quantity attributed to something.

[8] Collects all proper-nouns representing people classified as 'legatus'.

[9] Views all verbs in context representing actions carried out by Piso.

[10] *ElidedActor* and *SameAs* are IP ties, not dependencies. They are not included in the UD-IP conversion presented above.

enabling more detailed search in Latin corpora, and 3. annotating texts for creating taggers and parsers with UDPipe, while reducing the coarseness of functions in the representation. We also showed in which key ways IP differs from UD and explained how these differences improve the accuracy and utility of taggers and parsers in the study of Latin. The conversion script is available as DUX files (DUX Script) and a DUX interpreter is provided as a command line script (JAR Scripts).

Acknowledgements

This work is part of a project funded by the German Research Foundation (project number 316618374) and lead by Malte Dreyer, Stefan Kipf and Anke Lüdeling.

References

Anne Abeillé. 2012. *Treebanks: Building and Using Parsed Corpora*. Springer Science & Business Media, December.

Simon Aerts. 2018. Tense, aspect and Aktionsart in Classical Latin: Towards a new approach. *Symbolae Osloenses*, 92(1):107–149, January.

Ramadan Alfared and Denis Béchet. 2012. POS taggers and dependency parsing. *International Journal of Computational Linguistics and Applications*, 3(3).

Grigoris Antoniou and Frank van Harmelen. 2004. Web Ontology Language: OWL. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, International Handbooks on Information Systems, pages 67–92. Springer, Berlin, Heidelberg.

Jessie B Arneson and Erika G Offerdahl. 2018. Visual literacy in Bloom: Using Bloom's taxonomy to support visual learning skills. *CBE—Life Sciences Education*, 17(1):ar7,1–8.

Liesbeth Augustinus, Vincent Vandeghinste, Ineke Schuurman, and Frank Van Eynde. 2017. GrETEL: A tool for example-based treebank mining. In Jan Odijk and Arjan van Hessen, editors, *CLARIN in the Low Countries*, pages 269–280. Ubiquity Press.

David Bamman and Gregory Crane. 2011. The Ancient Greek and Latin Dependency Treebanks. In *Language Technology for Cultural Heritage*, pages 79–98. Springer.

Cem Bozsahin, Geert-Jan M Kruijff, and Michael White. 2005. Specifying grammars for OpenCCG: A rough guide. *Included in the OpenCCG distribution*.

Giuseppe GA Celano. 2019. An automatic morphological annotation and lemmatization for the IDP Papyri. In *Proceedings of the Third International Conference on Digital Access to Textual Cultural Heritage*, pages 149–153, Brussels, Belgium, May.

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

Olivia Laura Elder. 2019. *Language and the Politics of Roman Identity*. Thesis, University of Cambridge, March.

Rod Ellis. 2009. Task-based language teaching: Sorting out the misunderstandings. *International journal of applied linguistics*, 19(3):221–246.

Daniel Fischl and Arno Scharl. 2014. Metadata enriched visualization of keywords in context. In *Proceedings of the 2014 ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, pages 193–196.

Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. 2014. HermiT: An OWL 2 Reasoner. *Journal of Automated Reasoning*, 53(3):245–269, October.

Gaël Guibon, Marine Courtin, Kim Gerdes, and Bruno Guillaume. 2020. When Collaborative Treebank Curation Meets Graph Grammars. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5291–5300.

Michael A.K. Halliday and Christian M.I.M. Matthiessen. 1999. *Construing Experience through Meaning: A Language-Based Approach to Cognition*. Continuum, London/New York.

Michael A.K. Halliday and Christian M.I.M. Matthiessen. 2014. *Halliday's Introduction to Functional Grammar*. Routledge, London/New York, fourth edition.

Michael A.K. Halliday. 1966. The concept of rank: A reply (1966). In *On Grammar*, pages 118–126. Continuum, London.

Dag TT Haug and Marius Johndal. 2008. Creating a parallel treebank of the old Indo-European Bible translations. In *Proceedings of the Second Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2008)*, pages 27–34.

Marian Hillar. 2012. Thomas of Aquinas and the accepted concept of the trinity. In *From Logos to Trinity: The Evolution of Religious Beliefs from Pythagorians to Tertulian*, pages 249–272. Cambridge University Press.

Ali Khalili and Sören Auer. 2013. WYSIWYM Authoring of Structured Content Based on Schema.org. In Xuemin Lin, Yannis Manolopoulos, Divesh Srivastava, and Guangyan Huang, editors, *Web Information Systems Engineering – WISE 2013*, Lecture Notes in Computer Science, pages 425–438, Berlin, Heidelberg. Springer.

Bas Kollöffel. 2012. Exploring the relation between visualizer–verbalizer cognitive styles and performance with visual or verbal learning material. *Computers & Education*, 58(2):697–706.

Thomas Krause. 2019. ANNIS: A graph-based query system for deeply annotated text corpora. *Humboldt-Universität zu Berlin*, January.

Sandra Kübler and Erhard W. Hinrichs. 2001. From chunks to function-argument structure: A similarity-based approach. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics - ACL '01*, pages 346–353, Toulouse, France. Association for Computational Linguistics.

Raphael Kühner. 1879. *Ausführliche Grammatik Der Latenischen Sprache*. Hansche Buchhandlung, Hannover.

Leonid Kulikov and Nikolaos Lavidas. 2013. Reconstructing passive and voice in Proto-Indo-European. *Journal of Historical Linguistics*, 3(1):98–121.

Anoop Mahajan. 2003. Word order and (remnant) VP movement. In Simin Karimi, editor, *Word Order and Scrambling*, pages 217–237. Wiley Online Library.

Francesco Mambrini and Marco Carlo Passarotti. 2012. Will a parser overtake Achilles? First experiments on parsing the ancient Greek dependency treebank. In *Eleventh International Workshop on Treebanks and Linguistic Theories*, pages 133–144. Edições Colibri.

Francesco Mambrini. 2016. The Ancient Greek Dependency Treebank: Linguistic Annotation in a Teaching Environment. In Gabriel Bodard and Matteo Romanello, editors, *Digital Classics Outside the Echo-Chamber: Teaching, Knowledge Exchange & Public Engagement*, pages 83–99. Ubiquity Press.

Hermann Menge, Thorsten Burkard, and Markus Schauer. 2012. *Lehrbuch der lateinischen Syntax und Semantik*. WBG, Wiss. Buchges, Darmstadt, 5., durchges. und verb. aufl edition.

Monica Monachini, Anika Nicolosi, and Alberto Stefanini. 2018. Digital Classics: A survey on the needs of Ancient Greek scholars in Italy. In *Proceedings of the CLARIN 2017 Conference*, pages 1–5. Linköping University Electronic Press.

Frederike Müller and Sarah Oeste-Reiß. 2019. Entwicklung eines Bewertungsinstruments zur Qualität von Lernmaterial am Beispiel des Erklärvideos. In Jan Marco Leimeister and Klaus David, editors, *Chancen und Herausforderungen des digitalen Lernens: Methoden und Werkzeuge für innovative Lehr-Lern-Konzepte*, Kompetenzmanagement in Organisationen, pages 51–73. Springer, Berlin, Heidelberg.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, and Natalia Silveira. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666.

Renato Oniga and Norma Schifano. 2014. *Latin: A Linguistic Introduction*. Oxford University Press, Oxford.

Marco Passarotti. 2019. The Project of the Index Thomisticus Treebank. *Digital Classical Philology, De Gruyter, Berlin, Boston*, pages 299–320.

Demetra Pitta-Pantazi, Paraskevi Sophocleous, and Constantinos Christou. 2013. Spatial visualizers, object visualizers and verbalizers: Their mathematical creative abilities. *ZDM*, 45(2):199–213.

Priscianus. 2010. *Grammaire. [...] 1: Syntaxe Livre XVII.* Number 41 in Histoire des doctrines de l'Antiquité classique. Libraire Philosophique J. Vrin, Paris.

Hans Rubenbauer and Rolf Heine. 2012. *Lateinische Grammatik.* Buchner, Bamberg, [unveränderter nachdruck der] 12., korr. auflage 1995 edition.

Milan Straka, Jan Hajic, and Jana Straková. 2016. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *LREC*, pages 4290–4297.

Kashyap Todi, Jussi Jokinen, Kris Luyten, and Antti Oulasvirta. 2018. Familiarisation: Restructuring Layouts with Visual Learning Models. In *23rd International Conference on Intelligent User Interfaces*, IUI '18, pages 547–558, New York, NY, USA, March. Association for Computing Machinery.

Dmitry Tsarkov and Ian Horrocks. 2006. FaCT++ description logic reasoner: System description. In Ulrich Furbach and Natarajan Shankar, editors, *Automated Reasoning*, pages 292–297, Berlin, Heidelberg. Springer Berlin Heidelberg.

Petre Ureche. 2014. The soldiers' morale in the Roman army. *Journal of Ancient History and Archaeology*, 1(3), October.

Veronika Vincze, Dóra Szauter, Attila Almási, György Móra, Zoltán Alexin, and János Csirik. 2010. Hungarian dependency treebank. In *Seventh International Conference on Language Resources and Evaluation (LREC)*, pages 1855–1862.

Benjamin Lee Whorf. 1956. The relation of habitual thought and behavior to language (1939). In *Language, Thought and Reality: Selected Writings of Benjamin Lee Whorf*, pages 173–204. The MIT Press, Cambridge, MA.

Florian Zipser and Laurent Romary. 2010. A model oriented approach to the mapping of annotation formats using standards. In *Workshop on Language Resource and Language Technology Standards, LREC 2010*, May.

Akkadian Treebank for early Neo-Assyrian Royal Inscriptions

Mikko Luukko
University of Helsinki
mikko.luukko@helsinki.fi

Aleksi Sahala
University of Helsinki
aleksi.sahala@helsinki.fi

Sam Hardwick
University of Helsinki
sam.hardwick@iki.fi

Krister Lindén
University of Helsinki
krister.linden@helsinki.fi

Abstract

This paper presents the first proper syntactic treebank for Akkadian, an ancient Semitic language which can only be reconstructed from its textual data. We introduce our corpus of early Neo-Assyrian royal inscriptions, present some typical syntactic constructions of this genre and discuss the morphological and syntactic choices we have made. For developing a gold standard for morphological annotations, we tested the manually annotated material against BabyFST, a morphological analyzer of Akkadian. We also tested the reproducibility of the syntactic annotations using the TurkuNLP neural parser.

1 Introduction

This first version of our Akkadian treebank consists of 22 277 words and 1845 sentences. This represents an intact subset of a total of 2211 sentences from the early Neo-Assyrian royal inscriptions[1] of the tenth and ninth centuries BCE. Because of the progressive complexity of linguistic constructions in later texts of our source material,[2] our approach is chronological and we begin with the inscriptions of Aššur-dān II (r. 934–912), published in Grayson (1991).[3] The main sub-corpus of the volume and our first version are thus the inscriptions of Ashurnasirpal II.[4] The language of the corpus is Standard Babylonian,[5] with occasional Assyrianisms,[6] whereas "Akkadian" is the umbrella term for both Assyrian and Babylonian. In the modern world, Akkadian is not as well-known as Latin, Greek, Hebrew or Egyptian languages. In the ancient world, however, Akkadian was an important cultural language with a long history of more than two-thousand-and-five-hundred years as a spoken and written language. The name of the language comes from the capital of the legendary third-millennium King Sargon of Agade or Akkade.

These royal inscriptions were extracted from Oracc (Open Richly Annotated Cuneiform Corpus),[7] where all Neo-Assyrian royal inscriptions[8] are lemmatized word-for-word. More precisely, we have made use of the bound transcription (= normalized text) of the lemmatized texts which had been previously transliterated from clay tablets. The transliteration of Akkadian is based on both syllabically and pictographically, though abstracted, written cuneiform signs. Therefore, for this Akkadian treebank, we have the advantage that we do not have to consult the original tablets or take into account the subtleties of the cuneiform script. Perforce, because of the cuneiform script (writing system), the analysis of Akkadian syntax contains more speculative interpretation than with a modern language. The factor

[1] Sometimes referred to as ARI (Assyrian Royal Inscriptions).

[2] This is a simplification, but a number of royal inscriptions from the eighth and seventh centuries BCE are syntactically much more complicated; consider, e.g., Sargon II's famous Eighth Campaign.

[3] Grayson (1991), also known as RIMA 2 (references to the volume so below; Q-numbers, also below, refer to Oracc text IDs), contains the inscriptions of Tiglath-pileser I and his successors until Tiglath-pileser II, too, but these kings are usually considered Middle Assyrian.

[4] Grayson, 1991: 189–397.

[5] Standard Babylonian is a literary variant of Babylonian dialect (for its use in Neo-Assyrian royal inscriptions, see Frahm, 2019: 144–145); it was never a spoken language.

[6] Assyrianisms in this corpus were already discussed by Deller, 1957a and b.

[7] <http://oracc.museum.upenn.edu/>. By making editions of thousands of cuneiform texts available online for everyone with an Internet connection, Oracc has laid the foundation for Digital Assyriology.

[8] <http://oracc.museum.upenn.edu/riao/>.

contributing to this is the combination of word signs – usually called logograms or Sumerograms – and syllabic signs (syllabograms). One example from each spelling category for the three main parts-of-speech will suffice here:

- Nouns:
  - syllabically written *ma-da-tu* stands for *maddattu* “tribute”;
  - logographically written LUGAL stands for *šarru* “king”;
  - the combination of AN-*e* stands for *šamê* “heaven”.
- Verbs:
  - The normalization of *at-tu-muš* is *attumuš* and it means “I set out”;
  - GUR, *utēr* “It turned into (something)”;
  - KUR-*ud* stands for *akšud* and means “I conquered”.
- Adjectives:
  - *dan-nu-te*, *dannūte* “strong” (masculine plural from *dannu*);
  - DUGUD, *kabta* “heavy” (in the accusative, from *kabtu*);
  - GAL-*te*, *rabīte* “great” (singular feminine in the genitive, from *rabû*).

The distribution of different types of spellings and their combinations in this corpus are provided in Table 1.

Full corpus	**Syllabic**	**Logographic**	**Logo-Syllabic**
Nouns	3739	4179	1837
Verbs	2581	34	234
Adjectives	671	243	194
Other	7030	1570	633

Table 1: Different types of spellings and their combinations in the corpus of early Neo-Assyrian royal inscriptions.

Compared with the cognate Semitic languages, for example, we are in a lucky position and rarely confront a problem of vocalic ambiguity, which in other Semitic languages results from uncertain vocalization that is not marked in the original documents.[9] Moreover, unlike in other Semitic languages, Akkadian dictionaries are based on words and not on roots.

1.1 Basic Characteristics of Akkadian

Akkadian is an extinct Semitic language that has not been spoken anywhere since the first century of the Common Era. It is cognate to ancient and modern languages such as Arabic, Aramaic, Hebrew, Amharic and Maltese.[10] Akkadian, written in cuneiform script, displays several distinctive features. For example, texts do not include punctuation,[11] and the language does not express definiteness by using definite or indefinite articles, but “definiteness” must always be read from the context. As is typical of Semitic languages, Akkadian has a rich (and complex) morphology.

Thanks to the durability of cuneiform tablets written on clay, Akkadian is with its hundreds of thousands of texts a well-known language,[12] but already for decades, it has been a desideratum to enhance

[9] Partially the problem relates to the wide use of different writing systems among Semitic languages; cf., e.g., Zitouni, 2014: 35. In this context, we are not concerned with the correct interpretation of the syllabic C(onsonant)V(ocal)C(onsonant) signs whose reading values do give Assyriologists some trouble.

[10] One can find treebanks of these languages at Universal Dependencies (<https://universaldependencies.org/>).

[11] Akkadian texts rarely make use of a word-divider or any other device that belongs to the area of punctuation. However, especially literary texts may occasionally leave gaps between words but, as a rule, the original texts do not delimit word boundaries by “whitespace” characters.

[12] On the size of the Akkadian text corpus, see Streck, 2010.

our understanding of its syntax. For the most part, Akkadian word order follows the S(ubject)O(bject)V(erb) structure, which probably resulted from the direct influence of the non-Semitic Sumerian language already in the third millennium BCE at the latest.[13] However, while the main tendencies of Akkadian word order are easy to sketch out, in many instances the order is relatively free, although this may signify different semantic nuances in texts. Thus, depending on the types of sentences, the "standard" word order is not always strictly followed, but there are few studies on the significance of this phenomenon. For this reason, an Akkadian treebank will enable us to study Akkadian syntax from a new and much deeper perspective.[14]

2 Current Data Set

This corpus consists of 162 royal inscriptions of four early Neo-Assyrian kings: Aššur-dan II (r. 934–912 BCE), Adad-nerari II (r. 911–891 BCE), Tukulti-Ninurta II (r. 890–884 BCE) and Ashurnasirpal II (r. 883–859 BCE). Neo-Assyrian Royal Inscriptions are rather idiosyncratic commemorative texts, which serve to self-aggrandize Assyrian kings, and distinguish themselves from the other genres of Akkadian literature. These texts often begin with a long introduction, having the king's name (usually with genealogy) or divine invocation, lengthy royal or divine titles and epithets that stress the king's bravery.[15] These epithets given to Neo-Assyrian kings or gods are usually nouns or adjectives or the combinations of the two. For example, King Adad-nerari II says that he is

(1) *hitmuṭ raggi u ṣēni*
burning wicked (person) and evil (one)
"inflamed against the evil and wicked"
RIMA 2 A.0.99.2: 17 (Q006021) and 4:4′–5′ (Q006023).

This is annotated as three nouns and a conjunction, though the latter two nouns are formally adjectives and the first one is a stative or an infinitive. The sentences are rarely complex in an introduction but mainly lengthy nominal clauses, though they may occasionally show changes in word order. After the introduction, there is usually a section on military campaigns and then a separate section on building or renovation projects. Royal inscriptions mostly close with a section on blessings for pious future rulers who will take care of their predecessor's commemorative text. If a future ruler does not respect his predecessor's wishes, curses will befall him. The long narrative texts, with list-like conquests and itemized records of received tribute, are in sharp contrast to the brief labels and epigraphs that were originally attached to objects (especially many among the inscriptions of Ashurnasirpal II). The latter type of documents numerically form a large minority of the inscriptions in this corpus.

As mentioned previously, we annotate the Neo-Assyrian Royal Inscriptions published in Grayson (1991). Since punctuation is not used in Akkadian, we arrive at sentences by syntactically annotating the unsegmented corpus, and identifying words that are head words but are not themselves dependents of other words. The corpus also contains unidentified and partly identified words, and for this reason some sentences are sentence fragments, or contains unannotated material. We excluded them from the current version of our treebank, which thereby comprises 1845 sentences with 22 277 words.

There are a total of 3398 distinct phonologically transcribed word forms in the corpus. A majority of these, 3223, have only a single analysis in terms of lemma and morphology across the corpus, with the remaining 175 receiving different analyses in different contexts. The ambiguous forms represent 4767 tokens out of a total of 22 277 tokens in the corpus, i.e. 21%, meaning that the remaining 79% of the corpus consist of tokens that have only one analysis in this corpus.

3 Morpho-syntactic Analysis

As a first step, we have manually annotated each token in the corpus with a lemma and a part-of-speech (POS) as well as a morphological analysis, i.e. during the manual POS tagging, we separated the morphemes and annotated them with morphological features and syntactic relations. By far the largest group

[13] Edzard, 2003: 174 and Huehnergard and Woods, 2008: 128.

[14] A preliminary treebank for Akkadian with Babylonian Royal Inscriptions of the seventh and sixth centuries BCE called PISANDUB prepared by Kamil Kopacewicz can be found at http://universaldependencies.org/ containing 101 sentences with 1852 tokens. At the time of writing, it only contained POS tags and syntactic relations and no language documentation.

[15] On the structure of Neo-Assyrian royal inscriptions, see, e.g., Frahm, 2019: 146, 149.

of bound morphemes attached to nouns, verbs or prepositions is formed by suffixes which syntactically have different functions depending on their head. In annotating nouns and other parts-of-speech, we closely follow the terminology explained and listed in Reiner (1966: 57, 137). In the morphological analysis and POS tagging, our goal is to provide as much information as is evidenced by the morphemes in context. We annotate the following subcategories of verbs:

- finiteness (finite, infinitive, stative),
- stem (G, D, Š, N etc.),
- mood (indicative, imperative, precative, prohibitive),
- tense (present, preterite, perfect), person (1, 2, 3),
- number (singular, plural) and
- gender (masculine, feminine).

Following Streck (2011: 363), we consider subordinative[16] and ventive as subcategories of their own, which we tag as boolean values. For nouns, adjectives and non-finite verbal forms the subcategories are:

- case (nominative, accusative, genitive),
- number (as above),
- gender (as above) and
- base, which can have four different values:
    - free (status rectus),
    - bound (status constructus),
    - suffixal (followed by pronominal suffixes) and
    - terminal (status absolutus).

In general, our approach to POS tagging and to the syntactic dependency relations of each word follows as closely as possible the standards created, developed and maintained by the Universal Dependencies (henceforth UD) project; these principles are elaborated on the UD website.[17] For visualizing the syntactic analysis, we use a CONLL-U viewer, a tool available on the UD website.

In this corpus, from the seventeen Universal POS tags listed on the UD website, we have used all except auxiliary (AUX; Akkadian does not have genuine auxiliaries), interjection (INTJ) and symbol (SYM). Perhaps more surprisingly, we cannot use the label punctuation (PUNCT), because cuneiform inscriptions are continuous texts without punctuation. E.g. the end of a sentence is explicitly indicated only in exceptional circumstances.[18]

As to proper nouns and ethnic names, which are often called *nisbe* in Akkadian, their morphological annotation has been simplified and does not contain as many labels as regular nouns. This is due to the fact that they were written without inflections. Thus, proper nouns are simply annotated as PROPN + gender (if a personal or a divine name) and ethnic names are labelled as NOUN + gender. The latter are not always true proper names, because they can also refer to any single person of a tribe, although often this principle is reserved for the ruler of a tribe or a town.

According to the UD principles, participles are to be annotated as verbs or adjectives, but the so-called active participles in Akkadian cannot follow this principle, since the active participles in Akkadian act as the performers of action (cf. Arabic), so they are annotated as nouns. By observing UD, we also annotate all day dates as adjectives.

The construct state of Akkadian concerns the relation between two content words (cf. Arabic *Idafa*[19] and Hebrew *smikhut*).[20] This syntactic relation between the construct state noun (possessed/governing noun) and the following noun in the genitive (possessor/governed noun) is expressed with the label nmod:poss.

The frequent determinative pronoun *ša* "of" is annotated as ADP (= preposition) in the same way as is done with "of" in English. Another frequent word *u* "and, but" also has an adverbial meaning "further(more), moreover" at the beginning of a sentence. For example,

[16] We prefer this term instead of subjunctive following von Soden (1995: 135) and Streck (2011).

[17] <http://universaldependencies.org/>.

[18] One of the few exceptions is a section ruling, i.e. a horizontal divider in an inscription, that clearly indicates the end of a sentence, section or paragraph.

[19] Cf. Zitouni, 2014: 19.

[20] On the construct state in Akkadian, see, e.g., Huehnergard, 2005: 56.

(2) *u rapšāte mātāt Nairi ana pāṭ gimriša apēl*
and broad lands Nairi to border totality-its ruled
"Moreover, I gained entire dominion over the extensive lands of Nairi" Ashurnasirpal II (passim).

Akkadian does not have definite or indefinite articles. For those familiar with the Oracc lemmatization,[21] where many determiners appear under the label XP, standing for indefinite pronouns, the UD annotation is notably different. PRON is another label for which it is appropriate to point out the difference between the UD principle and the Oracc lemmatization. In the latter, e.g., Akkadian indefinite pronouns *mamma* "somebody, anybody, (negated) nobody" and *mimmu* "something, anything, everything, (negated) nothing" bear the XP label.

3.1 BabyFST

BabyFST is a finite-state based morphological model for Babylonian, a southern dialect of the Akkadian language (Sahala et al., 2020). The model is capable of providing morphological analysis for different stages of the Babylonian dialect, including Standard Babylonian and some of its typical Assyrianisms. The model is implemented in the LEXC and XFST formalisms, which can be compiled into finite-state transducers by using compilers such as Foma (Hulden, 2009) and HFST (Lindén et al., 2009). BabyFST tags Akkadian word tokens with their morphological features (number, gender, case, construct state, mood, tense, person, verbal affixation and verbal stems including *-t-* and *-tan-* infixation) as well as lemma and part-of-speech.

We verified and normalized the manually produced morphological annotations by using BabyFST to ensure that the human-produced annotations were consistent and formally in line with BabyFST's output, which for the tokens is true for 94.6% of the lemmas, and 85.5% of the morphological analysis. The seemingly low score on morphological annotation is due to underspecification in the manual annotation, local variation in the gender of a few frequent nouns and local spelling variants and Assyrianisms, i.e. Babylonian words with Assyrian influences. We can compare Akkadian writing standards to current writing conventions in social media discussion forums.

Used as a morphological gold standard, the treebank contains fully-specified morphological analyses for 3012 nouns, 2053 verbs and 555 adjectives. The analyses are underspecified for 5317 nouns, 136 verbs and 338 adjectives, often because a word is written using a logogram and the inflected form is not explicitly indicated. Underspecification may also occur in the construct state, where the case endings are not marked. In such instances, one or more subcategories are marked as undefined.

The morphological annotation in the treebank will allow using the current annotation as a gold standard for morphological analysis, e.g. using BabyFST with disambiguation or using neural networks to predict both lemma and annotation in context.

4 Syntax

4.1 Language-specific Remarks

Traditionally, the study of Akkadian grammar has been dominated by morphological and lexical studies, and syntactic studies have been more peripheral. Standard Akkadian grammars, such as von Soden 1995, have been the mainstay of Akkadian syntax and monographs on the topic are still rare. However, relatively recently there has been a clear increase in the large-scale syntactic studies of Akkadian (e.g. Deutscher, 2000 and Cohen, 2012), although mainly syntactic studies are published in articles.

UD lists thirty-seven different syntactic relations. We have used twenty-five of them in our annotation; the following relations have so far not been applied partly due to the nature of the text genre: aux (auxiliary); clf (classifier);[22] compound; cop (copula); csubj (clausal subject); dislocated; expl (expletive); fixed; flat; orphan; punct (punctuation); reparandum.

[21] <http://oracc.museum.upenn.edu/doc/help/languages/akkadian/index.html>.

[22] The original texts include determinatives, but they were omitted, when the bound transcription of a text was prepared.

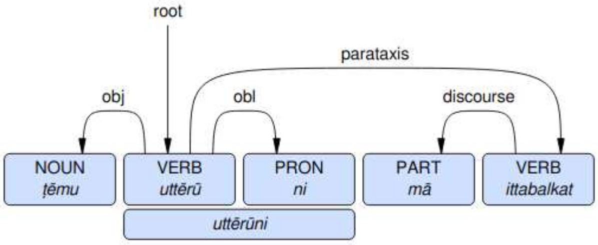
We have used the following UD relations: acl (adjectival clause); advcl (adverbial clause modifier); advmod (adverbial modifier); amod (adjectival modifier); appos (appositional modifier); case; cc (coordinating conjunction); ccomp (clausal complement); conj (conjunct); dep (unspecified dependency); det (determiner); discourse; goeswith; iobj; list; mark (marker); nmod (nominal modifier); nsubj (nominal subject); nummod (numeric modifier); obj (object); obl (oblique nominal); parataxis; root; vocative;[23] xcomp (open clausal complement).

The used relations, discourse, goeswith, list and vocative are rare in this corpus: For the relation goeswith, which mends erroneously split words, we have only one attested case in which a single concept made out of the negation *lā* and the following noun *salīma* has been split over two separate lines in the original: *lā salīma* not peace "truceless" RIMA 2 A.0.101.17 V 101–102 (Q004471).

In a way, early Neo-Assyrian royal inscriptions contain many different types of "lists" (e.g., of conquered cities or received tribute from foreign rulers or of various dishes offered at a special inaugural banquet for a new palace or of exotic plants, trees and animals, etc.). Nevertheless, for the most part we have chosen to tag the items enumerated in such lists with the conj relation.

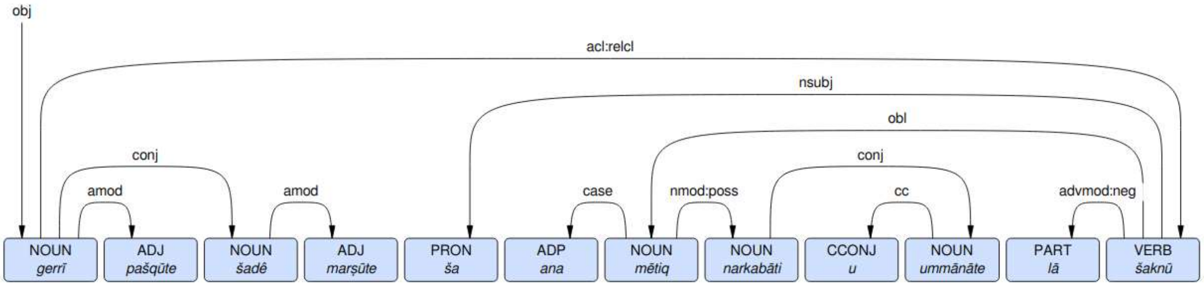
The only exceptional case which we tag as discourse is the use of *mā*[24] to indicate a direct speech quotation, a rare phenomenon in this genre, in a way it equals a colon:

(3) *ṭēmu* *uttērūni* *mā …* *ittabalkat*
report returned-me saying … crossed over
"A report was brought back to me: 'It (= a city) … has rebelled'"
RIMA 2 A.0.101.1 I 75 (Q004455).



The following relation subtypes have been used: acl:relcl for relative clauses, advmod:emph,[25] advmod:neg for the negation particles *lā* and *ul*, det:poss for possessive determiners and nmod:poss for the construct state. They are all frequent in Akkadian.

(4) *gerrī pašqūte šadê marṣūte ša ana mētiq narkabāti u ummānāte lā šaknū*
ways narrow mountains difficult which for route chariots and troops not put
"Difficult paths (and) rugged mountains which were unsuitable for chariotry and troops"
RIMA 2 A.0.101.17 I 65–66 (Q004471).



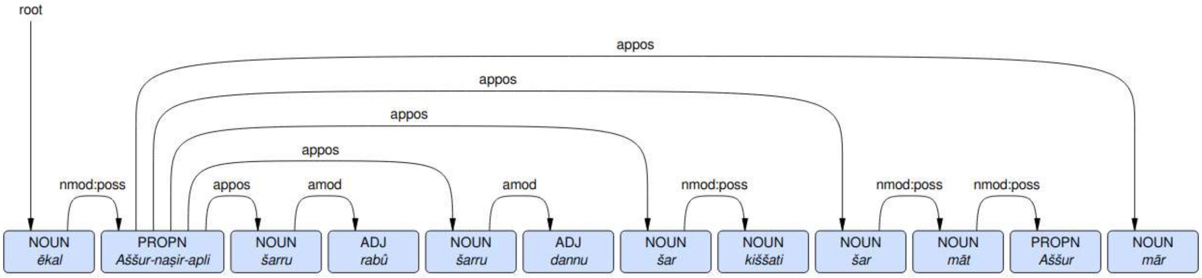
For example, a typical, brief label in this corpus does not include a verbal clause, but it enumerates the ruler (owner) and his immediate ancestors, and begins like this with several nmod:poss cases:

[23] For the only example tagged as vocative, see Adad-nerari II, RIMA 2 A.0.99.2: 77–78 (Q006021-5), in which the king addresses himself in public in front of his magnates in the third person.

[24] For example, Deutscher (2000: 66–91) calls the related Babylonian umma "a quotative marker".

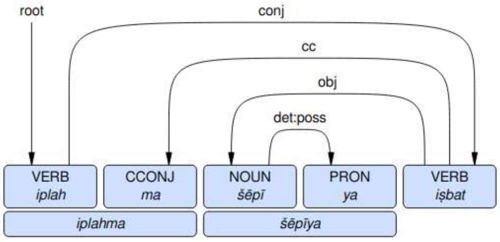
[25] This subtype concerns the particle *lū* (or *lu*) in its asseverative function (Kouwenberg, 2017: 640–43), although no strict attempt has been made here to keep it distinct from the precative *lū*. Hence most of the cases in which *lū* is separate from the verb has got the advmod:emph relation.

(5) *ēkal Ashurnasirpal šarru rabû šarru dannu šar kiššati šar māt Aššur mār …*
palace Ashurnasirpal king great king strong king totality king land Aššur son ...
“(Property of the) palace of Ashurnasirpal, great king, strong king, king of the universe, king of Assyria, son of … (followed by a short genealogy of the king’s father and grand-father)”
RIMA 2 A.0.101.102 (Q004556 and passim).



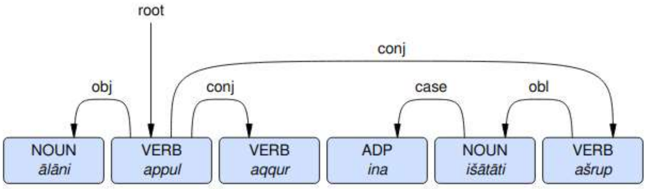
In the contemporary Neo-Assyrian letters the enclitic *-ma* particle has become obsolete in coordinating verbal clauses (Hämeen-Anttila, 2000: 66, 122; Luukko, 2004: 108), and in this corpus its use is also clearly on the decline, though several verbs still take *-ma*. For example,

(6) *iplah-ma … šēpīya iṣbat*
be(come) afraid-*ma* feet-my seized
“He took fright and submitted to me.”
Ashurnasirpal II, RIMA 2 A.0.101.1 III 73 (Q004455).



However, verbal clauses are mainly coordinated asyndetically:

(7) *ālāni appul aqqur ina išātāti ašrup*
cities demolished destroyed in fires burnt
“I razed, destroyed, (and) burnt the cities.” Passim



Along similar lines, in nominal clauses the phrases with the conjunctive *u* “and”, such as *biltu u maddattu* “tribute and tax”, and its equivalent *biltu maddattu* “tribute (and) tax” without a coordination conjunctive appear in free variation with one another.[26]

In this corpus, written in Standard Babylonian, the verbal subordinative is either the Babylonian *-u* or the Assyrian -(*ū…*)*ni*.[27] We label the relation of the Assyrian subordinative marker *-ni* with its main

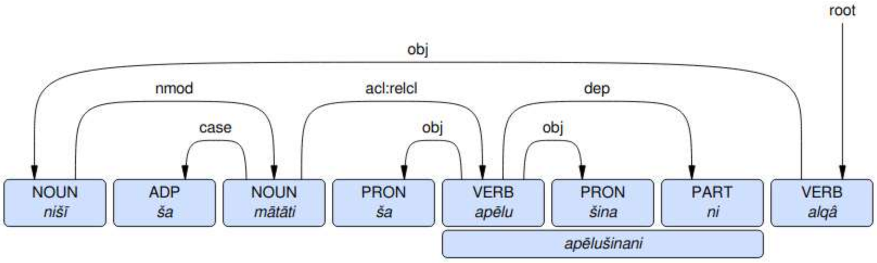
[26] Both these variants occur even on the same line in RIMA 2 A.0.99.2: 115 (Q006021). In the print edition of the two longest texts of the corpus, RIMA 2 A.0.101.1 and 17 (Q004455 and Q004471), there are altogether, i.e., both in nominal and verbal clauses, 389 and 209 restored “(and)” cases respectively!

[27] Some Assyrian examples in this corpus were already given in Deller, 1957a: 153–54 and id. 1957b: 272. For the use of the term subordinative instead of the subjunctive, see (also above) now Bjøru and Pat-El (2020: 71, n. 1) and already von Soden (1973).

word as dep. When *-ma*, which is attached to a verb, appears in clause-final position, we tag its relation similarly to the verb with dep.

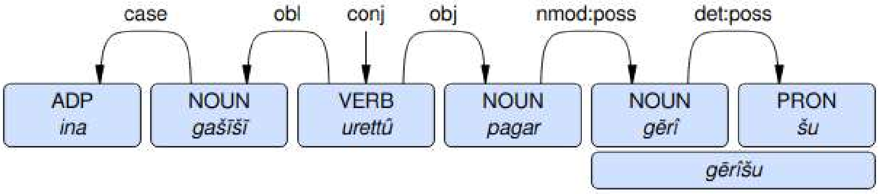
Subordinate clauses precede the main clause but relative clauses, introduced by *ša*, immediately follow the main word (in a main clause) which they qualify:

(8) *nišī … ša mātāti ša apēlušinani … alqâ*
people of lands which ruled (over)-them took
"I took people … from the lands over which I had gained dominion."
Ashurnasirpal II, e.g., in RIMA 2 A.0.101.2: 53–55 (Q004456) and 23: 15–17 (Q004477).



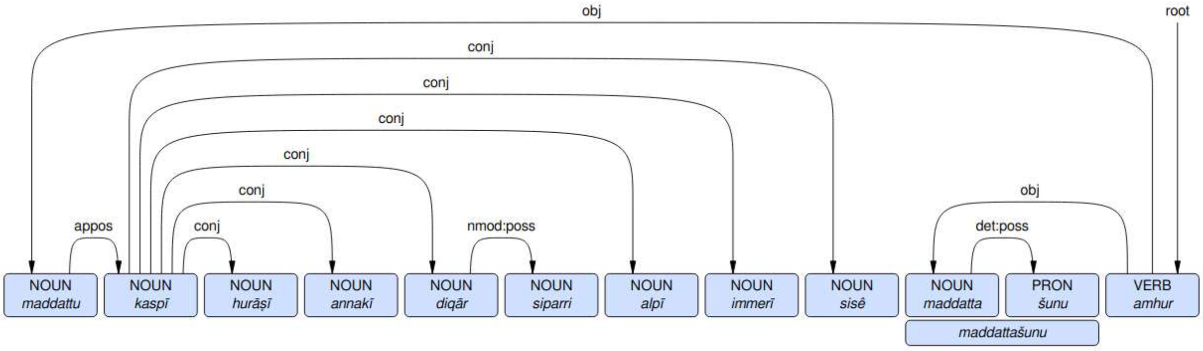
Occasionally, unlike in the standard word order, an object may follow the main predicate:

(9) *ina gašīšī urettû pagar gērî šu*
on stakes installed body enemy-his
"He hung the corpses of his enemies on stakes."
Ashurnasirpal II, RIMA 2 A.0.101.1 I 29 (Q004455)



If there are several tribute bearers and many items, then the main object (tribute) may be repeated before the predicate (unlike here, it is usually left untranslated in editions):

(10) *maddattu ... kaspī hurāṣī annakī diqār siparri alpī immerī sisê maddatta*
tribute silver gold tin large bowl bronze oxen sheep horses tribute
šunu amḫur
their received
"I received the tribute … silver, gold, tin, bronze casseroles, oxen, sheep, (and) horses, their tribute" Ashurnasirpal II, RIMA 2 A.0.101.1 II 21–23 (Q004455).



4.2 Parser Experiment

The TurkuNLP neural parser (Kanerva et al. 2018) is a processing pipeline for segmentation, morphological analysis, dependency parsing and lemmatization. Each of these tasks is implemented by separate neural models, and when combined, the parser is able to produce fully annotated CoNLL-U files from raw text. It was overall a top-ranked parser in the CoNLL-18 shared task for multilingual parsing from raw text to universal dependencies.

The parser is provided with two human-annotated CoNLL-U files: a training set, which is used for adjusting the neural weights, and a development set which is used for observing the performance of the parser during training. In addition, to evaluate its final performance, a test set, unused during training, is annotated by the trained parser.

Manual syntactic annotation of the corpus had resulted in unsegmented running text with dependency markings. We used this annotation to automatically split the texts into sentences. In a perfect situation, this should have required nothing more than allocating each dependency tree into its own sentence resulting in a segmentation of the entire text (i.e. all the tokens). However, parts of the text that were possible to transcribe only in part, or not at all, resulted in incomplete tree structures.

We first attempted to segment the 162-text corpus in its entirety, allocating unidentified or partly labelled tokens to nearby sentences, and use this data to train the parser. The parser both received training data and outputted parsing results that contained tokens with blank fields for lemmas, morphology and syntax. The result of this experiment in terms of numeric scores was, however, disappointing, the system not being designed with this sentence fragment scenario in mind.

We then produced a set of sentences which did not have structural problems resulting from unidentified or partly labelled tokens. These numbered 1845 out of a total of 2211 possible trees. Here, “possible trees” means tokens that could be syntactic roots, i.e. they have dependents but do not depend on other tokens, and are in effect an upper bound. These sentences were randomly shuffled and split into the previously mentioned training (80% of sentences), development (10%) and test sets (10%). We deemed shuffling to be preferable to assigning sentences in running order, as the corpus is rather heterogeneous, a few long texts dominating the word count.

On the test set, we tested both the case where we provided segmentation cues, which in most other treebanks are present in the form of punctuation or formatting, and the case where all the shuffled sentences occurred as a consecutive string of words. In the latter case, the parser infers sentence and token boundaries. Errors in these tasks contribute to lower scores in the parsing task. We calculated the scores with the CoNLL 2018 shared task evaluation script (SIGNLL 2018).

For the segmented case, we obtained a LAS (labeled attachment) score F1 of 93.29, an MLAS (morphology-aware attachment) score of 87.53 and a BLEX (bi-lexical dependency) score of 91.71. These are the main metrics used in the CoNLL 2018 shared task. LAS is a reflection of how well the dependency relations (arc and label) matched between the parser’s output and the gold standard; MLAS includes the requirement that the morphological analysis is also matching; BLEX the requirement that the lemmatization matches. These results are surprisingly good relative to the automatic parsing of most languages, and probably reflects the rather repetitive nature of this corpus, and of course the segmentation provided by us.

When no segmenting cues were provided, we obtained a LAS of 69.95, an MLAS of 58.97 and a BLEX of 62.44. This is on par with that obtained in the “small treebanks” subtask in CoNLL 2018.

5 Discussion and Conclusion

The fragmentary state of cuneiform texts is a frequent problem and it concerns this sub-corpus of Assyrian royal inscriptions too. In Assyriology, indiscernible words in the transcription are indicated with an *x* in the transcription. Sometimes our standard text editions exacerbate the problem by providing too few (or too many) *x*s, making restorations and the syntactical annotation of a text difficult or even impossible in many cases when the *x*s distort the syntactic flow of the text if the number of *x*s given in transliterations or bound transcriptions does not correspond to the situation on the original text carrier. The issue is aggravated in the current text genre which consists of many relatively large artefacts; the shorter and the more standardized the texts are, the easier it is to restore and assign the length of the gaps relatively reliably.[28]

As to restorations in general, we have adhered to the suggestions given in Grayson (1991) to the extent that restorations now appear without brackets, which is the usual way to indicate broken passages in Akkadian texts. Methodologically, this will probably not do much harm when studying Akkadian

[28] On the challenges of preparing a treebank of a language originally written in the cuneiform script according to the UD model, see Inglese (2015).

syntax, but in text research that may delve deeper into the details of a passage, some of the restorations could be questioned.

We have briefly described the Akkadian language, with some of its characteristics, and defined our corpus of early Neo-Assyrian royal inscriptions for building a treebank, the first proper treebank for Akkadian (comprising Assyrian and Babylonian). The manual annotation process is thus far the work of a single expert annotator (Mikko Luukko), who first used the Brat rapid annotation tool but later switched to WebAnno. To achieve a consistent morphological gold standard, the morphological annotation was checked against BabyFST, a morphological analyzer. The syntactic annotation consistency has been tested with the TurkuNLP parser.

Our first treebank will be released under the Universal Dependencies scheme with 1845 out of a total of 2211 possible sentences. When testing a parser on the pre-segmented sentences, we obtained a LAS score F1 of 93.29, an MLAS score of 87.53. When no segmenting cues were provided, we obtained LAS 69.95 and MLAS 58.97, which is on par with that obtained in the "small treebanks" subtask in CoNLL 2018. In the near future, our main challenge is to generalize the annotation to new material from other text genres.

Acknowledgements

The research for this article was carried out as part of the Centre of Excellence in Ancient Near Eastern Empires (ANEE) in cooperation with FIN-CLARIN and the Language Bank of Finland, taking place in Helsinki, and funded by the Academy of Finland. We would like to thank Niek Veldhuis (Berkeley), the initiator of the project, and David Bamman (Berkeley), who helped in setting up the first attempt for annotation on the Brat rapid annotation tool. For making this project possible, we are also indebted to Karen Radner, Jamie Novotny and Nathan Morello (all three LMU, Munich) and to Grant Frame and Steve Tinney (both UPenn, Philadelphia).

References

Øyvind Bjøru and Na'ama Pat-El. 2020. The Historical Syntax of the Subordinative Morphemes in Assyrian Akkadian. *Zeitschrift für Assyriologie und Vorderasiatische Archäologie*, 110(1):71–83.

Eran Cohen. 2012. *Conditional Structures in Mesopotamian Old Babylonian* (Languages of the Ancient Near East, 4). Eisenbrauns, Winona Lake, IN.

Karlheinz Deller. 1957a. Zur sprachlichen Einordnung der Inschriften Aššurnaṣirpals II. (883–859). *Orientalia Nova Series* 26(2):144–156.

Karlheinz Deller. 1957b. Assyrisches Sprachgut bei Tukulti-Ninurta II (888–884). *Orientalia Nova Series* 26(3):268–272.

Guy Deutscher. 2000. *Syntactic Change in Akkadian: The Evolution of Sentential Complementation*. Oxford University Press, Oxford and New York.

Dietz Otto Edzard. 2003. *Sumerian Grammar* (Handbuch der Orientalistik: Der Nahe und der Mittlere Osten, 71). Brill, Leiden and Boston.

Eckart Frahm. 2019. The Neo-Assyrian Royal Inscriptions as Text: History, Ideology, and Intertextuality. In *Writing Neo-Assyrian History: Sources, Problems, and Approaches* (State Archives of Assyria Studies, 29), edited by Giovanni B. Lanfranchi, Raija Mattila, and Robert Rollinger, 139–159. The Neo-Assyrian Text Corpus Project, Helsinki.

A. Kirk Grayson. 1991. *Assyrian Rulers of the Early First Millennium B.C. I (1114 – 859 B.C.)* (Royal Inscriptions of Mesopotamia, Assyrian Periods, 2). University of Toronto Press, Toronto.

Jaakko Hämeen-Anttila. 2000. *A Sketch of Neo-Assyrian Grammar* (State Archives of Assyria Studies, 13). The Neo-Assyrian Text Corpus Project, Helsinki.

John Huehnergard. ^{2}2005. *A Grammar of Akkadian* (Harvard Semitic Museum Studies, 45). Eisenbrauns, Winona Lake, IN.

John Huehnergard and Chris Woods. 2008. Akkadian and Eblaite, In *The Ancient Languages of Mesopotamia, Egypt and Aksum*, edited by Roger D. Woodard. 83–153. Cambridge University Press, Cambridge.

Mans Hulden. 2009. Foma: A Finite-State Compiler and Library. *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics* (*EACL*): Demonstrations Session, 29–32. Association for Computational Linguistics, Athens.

Guglielmo Inglese. 2015. Towards a Hittite Treebank. Basic Challenges and Methodological Remarks. In *Proceedings of the Workshop on Corpus-Based Research in the Humanities (CRH) 10 December 2015, Warsaw, Poland*, edited by Francesco Mambrini, Marco Passarotti, and Caroline Sporleder, 59–68.

Jenna Kanerva, Filip Ginter, Niko Miekka, Akseli Leino, and Tapio Salakoski. 2018. Turku Neural Parser Pipeline: An End-to-End System for the CoNLL 2018 Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, edited by Daniel Zeman and Jan Hajič, 133–142. Association for Computational Linguistics, Brussels.

N.J.C. Kouwenberg. 2017. *A Grammar of Old Assyrian* (Handbuch der Orientalistik, 1/118). Brill, Leiden and Boston.

Krister Lindén, Miikka Silfverberg, and Tommi Pirinen. 2009. HFST tool for morphology: An efficient open-source package for construction of morphological analyzers. In *State of the Art in Computational Morphology*, edited by Cerstin Mahlow and Michael Piotrowski, 28–47. Communications in Computer and Information Science, Berlin and Heidelberg.

Mikko Luukko. 2004. *Grammatical Variation in Neo-Assyrian* (State Archives of Assyria Studies, 16). The Neo-Assyrian Text Corpus Project, Helsinki.

Erica Reiner. 1966. *A Linguistic Analysis of Akkadian* (Janua Linguarum, Series Practica, 21). Mouton, The Hague.

Aleksi Sahala, Miikka Silfverberg, Antti Arppe, and Krister Lindén. 2020. BabyFST – Towards a Finite-State Based Computational Model of Ancient Babylonian. In *Proceedings of the 12th Language Resources and Evaluation Conference* (*LREC*), 3886–3894.

SIGNLL = Special Interest Group on Natural Language Learning of the Association for Computational Linguistics. 2018. Evaluation script for the 2018 CoNLL shared task, web description. http://universaldependencies.org/conll18/evaluation.html

Wolfram von Soden. 1973. Der akkadische Subordinativ-Subjunktiv. *Zeitschrift für Assyriologie und Vorderasiatische Archäologie* 63(1):56–58.

Wolfram von Soden. ^{3}1995. *Grundriss der akkadischen Grammatik* (Analecta Orientalia, 33/47). Pontificium Institutum Biblicum, Rome.

Michael P. Streck. 2010. Großes Fach Altorientalistik: Der Umfang des keilschriftlichen Textkorpus. *Mitteilungen der Deutschen Orient-Gesellscha*ft, 142:35–58.

Michael P. Streck. 2011. Babylonian and Assyrian. In *The Semitic Languages: An International Handbook,* edited by Stefan Weninger, 359–396. De Gruyter Mouton, Berlin and Boston.

Imed Zitouni (ed.). 2014. *Natural Language Processing of Semitic Languages*. Springer, Berlin and Heidelberg.

Dependency Relations for Sanskrit Parsing and Treebank

Amba Kulkarni†, Pavankumar Satuluri‡,
Sanjeev Panchal†, Malay Maity†, and Amruta Malvade†
†University of Hyderabad, India
‡Chinmaya Vishwavidyapeeth, India
ambakulkarni@uohyd.ac.in

Abstract

Dependency relations are needed for the development of a dependency treebank and a dependency parser. The guidelines[1] for the development of treebank for Sanskrit proposed a set of dependency relations. Use of these relations for the development of a sentence generator and a dependency parser for Sanskrit demanded a need for an enhancement as well as a revision of these relations. In this paper, we discuss the revised version of these relations and discuss the cases where there is a possibility of multiple tagging either due to the ellipsis of certain arguments or due to the possible derivational morphological analysis. This led us to arrive at specific instructions for handling such cases during the tagging. A treebank with around 4000 sentences has been developed following these guidelines. Finally we evaluate a grammar based dependency parser for Sanskrit on this treebank and report its performance.

1 Introduction

Sanskrit is one of the oldest languages in the world and has literature at least hundred times that of Greek and Latin together. This literature ranges from scientific disciplines such as Mathematics, Āyurveda, texts dealing with Language Sciences, Ontology, Logic, Metallurgy, Physics, Polity, and Law to Philosophical texts, Epics and several texts of lasting artistic merit. India's contribution to the development of Language Sciences dealing with various branches such as phonetics, phonology, morphology, syntax, semantics, discourse analysis and logic are found to be relevant for Language Technology. Among these, Pāṇini's grammar and the theories of verbal cognition deserve special mention from the Natural Language Processing (NLP) perspective. While the Pāṇini's grammar provides an almost complete grammar for generation, the theories of verbal cognition provide a systematic approach to analyse any text objectively. In this approach attention is paid to the information encoded in a linguistic expression. Division of a word into morphemes, role of some morphemes in connecting other morphemes, deciding the meaning of the morphemes are some of the topics that are discussed in these theories. Pāṇinian grammar provides the detailed description of how the semantic relations are realised through various morphological features, word order, and various other means of information encoding. The theories of verbal cognition use these clues of information encoding and other factors such as expectancy, mututal congruency of word meanings, proximity of the arguments etc. to decide the relations between the words.

The semantic relations used by Pāṇini to describe various relations thus provide a basic set for developing a dependency parser and also for the development of a treebank. This set of relations was enhanced over a period of 2-3 millenia by the grammarians and theoreticians working in the field of verbal cognition. A list of all such relations is compiled by Ramakrishnamacaryulu (2009) and presented as dependency relations for Sanskrit for both inter-sentential as well as intra-sentential tagging. These dependency relations were used as a starting point and the consortium for Sanskrit-Hindi Machine Translation (SHMT) system[2] arrived at a set suitable for the development of Sanskrit treebank. This resulted into the first version of the tagging guidelines for Sansktit treebank[3]. While developing a dependency

[1] http://sanskrit.uohyd.ac.in/scl/GOLD_DATA/Tagging_Guidelines/Tagging_eng_ver1.pdf

[2] funded by Technology Development for Indian Languages, MeiTy, Government of India, 2008-2012

[3] https://sanskrit.uohyd.ac.in/scl/GOLD_DATA/Tagging_Guidelines/Tagging_eng_ver1.pdf

parser, and also a sentential generator for Sanskrit, it was noticed that this set of dependency relations has some limitations and needs further enhancement as well as modifications. In this paper we discuss the revised version of this set. This set of relations is also used to develop a Sanskrit treebank. We present the cases of ambiguities in tagging while developing the treebank. This treebank is also used for the evaluation of the Sanskrit parser. We present the performance of this parser and discuss the limitations of both the parser as well as the dependency relations.

The paper is structured as follows. In the next section, we provide the literature survey of the state-of-art dependency relations and treebanks for parsing. This is followed by the discussion on the modifications to the earlier Sanskrit dependency relations and the enhancement thereupon justifying the necessity. In the fourth section we describe the Sanskrit treebank followed by the evaluation of a grammar based parser on this treebank. This is followed by the conclusion.

2 Brief survey

The last two decades have established the suitability of dependency parse over a constituency parse, even in the case of positional languages, for a wide range of NLP tasks such as Machine Translation, question answering, information extraction. This led to the development of dependency treebanks for various languages. Most of the languages followed an easy path of converting the existing constituency treebanks into dependency treebanks. Therefore the dependency relations used by these treebanks are also more syntactic in nature. At the same time several efforts were on developing a dependency parser for English. For example, the Link grammar, which is closely related to a dependency grammar proposed a set of around 106 relations which were not directional (Daniel and Temperley, 1993). Minipar had 59 relations (Lin, 2003). Caroll et al. (1999) and King et al. (2003) had proposed a set of dependency relations which were used by Marneffe et al. (2006) to convert the Phrase Structure treebanks to Dependency treebanks. This effort also led to some modifications to these relations, largely based on practical considerations. The number of relations proposed by them were 47. Most of these relations were syntactic in nature rather than semantic. These relations were incorporated in the Stanford parser. Thus we see that there was a huge variation between the number of relations used by various research groups, and naturally their semantic content also differed.

For most of the morphologically rich languages like Czech, Hindi, and Finnish manually annotated dependency treebanks were developed. The Prague Dependeny Treebank (PDT) is one of the oldest dependency treebanks (Bejček et al., 2013). This treebank is annotated at both the syntactic as well as semantic (tectogrammatic) level (Böhmovà et al., 2003). AnnCorra, guidelines for annotating dependency relations based on Pāṇinian grammar, was developed for Indian languages, and the treebanks for major Indian languages were developed following these guidelines (Bharati et al., 2002).

The major effort towards bringing in a standard among the dependency relations is by (Nivre et al., 2016) who proposed the Universal dependencies.[4] The Universal dependencies aim for a common annotation scheme for all the languages so that cross-linguistic consistency among the treebanks for several languages is achieved. The Universal dependencies were evolved from the Stanford dependencies (Marneffe and Manning, 2008). Though most of the relations from the Universal dependencies are syntactic in nature, the nsubj relation together with the newly proposed nsubj:pass relation makes this pair equivalent to the concept of *abhihita* of the Pāṇinian dependencies (Bharati and Kulkarni, 2011). Around 90 languages in the world including the three Classical languages viz. Greek, Latin and Sanskrit have dependency treebanks following Universal Dependencies.

Among the classical languages, both Ancient Greek and Latin have dependency treebanks following their own grammars. The ancient Greek dependency treebank consists of 21,170 sentences (309,096 words) from ancient Greek texts (Bamman and Crane, 2011). The Latin dependency treebank (V.1.5) consists of 3473 annotate sentences (53,143 words) from eight texts. The Latin tagset (V.1.3) consists of 20 categories mainly and they are further elaborated into various types. In this tagset, they have explained, with examples, how to annotate specific constructions involving relational clauses, gerunds,

[4] https://universaldependencies.org/

direct speech, comparison etc.[5].

All these dependency relations are mostly syntactic in nature. A strong need is also felt for the semantic annotation. Levin and Rappaport (2005) discuss the problems in thematic level annotation. This led to other models for semantic level tagging. Propbank (Palmer et al., 2005) and FrameNet (Fillmore et al., 2003) are the two prominent among them.

Pāṇini's scheme for annotation of relations is syntactico-semantic (Kulkarni and Sharma, 2019). Unlike the semantics dealt with in Propbank or the FrameNet annotations, in Pāṇini's scheme, the level of semantics is precisely the one that can be extracted only from the linguistic expression (Bharati and Kulkarni, 2010).

3 Saṁsādhanī Dependency Relations

Manually annotated data at various levels has become now an essential resource for computational analysis of texts. Such a resource is not only useful for machine learning but also comes handy as a test data for grammar based systems. To extract various kinds of relations between words in a sentence, it is necessary to have a corpus tagged at the level of relations between the words. Pāṇini's grammar provides semantic definitions of various relations between words and also provides rules that tell us how these relations are realised morphologically. The noun-verb relations are called the *kāraka* relations which refer to six different types of participants of an action viz. *kartā* (roughly an agent), *karma* (roughly a goal or a patient), *karaṇam* (instrument), *sampradānam* (recipient), *apādānam* (source) and an *adhikaraṇam* (location). The Indian grammarians further sub-classified and enhanced these relations by introducing a few more relations that deemed to be necessary from analysis point of view. In addition, two other relations viz. *prayojanaṃ* (purpose) and *hetuḥ* (cause) also involve noun-verb relationship. The list of all these relations, with around 100 entries, is collected and classified by Ramakrishnamacaryulu (2009). This list was the starting point in framing tagging guidelines in building treebanks. It was noticed that these relations were very fine-grained, and were neither suitable for a human annotator nor for computer parsing with high accuracy. Taking into consideration both the aspects viz. the manual tagging as well as the automatic parsing, around 31 relations were chosen from this set (Kulkarni and Ramakrishnamacharyulu, 2013). A treebank of around 3,000 sentences was developed following these guidelines.[6] These dependency relations, when, were examined from the sentence generation point of view, it was noticed that this set has several relations that were not semantic in nature, and referred to the morphological requirement or were syntactic in nature. This forced us to look at these relations afresh.

3.1 Enhancements and Modifications

In Sanskrit, there are certain words, in the presence of which a noun gets a specific nominal suffix. This is a morphological requirement, and in Pāṇini's grammar no semantics associated with such morphological requirements is discussed. As an example of such requirement let us consider the following sentence.

(1)	*Skt:*	*grāmaṃ*	*paritaḥ*	*vṛkṣāḥ*	*santi.*
	Gloss:	village{sg,acc}	surrounding	tree{pl,nom}	be{pres,pl,3p}.

Eng: There are trees surrounding the village.

In this sentence, the verb 'be' is not a copula, but indicates an existence. The word *paritaḥ* (surrounding) refers to the location and has an expectancy of a reference point, and the word denoting this reference point gets an accusative case marker. Figure 1 shows both the old and the new versions. In the old version, the label was *upapadasambandhaḥ* (literally 'a relation due to an adjacent word') which was a morphosyntactic label. In the new version this has been replaced by a semantic label '*sandarbha_binduḥ*' (reference point). When the word *paritaḥ* (surrounding) is used, there is a natural expectancy: 'surrounding what?'. The answer to 'what' gives a reference point for surrounding. Hence this relation is termed 'reference point' (*sandarbha_binduḥ*).

[5] `http://static.perseus.tufts.edu/docs/guidelines.pdf`

[6] http://tdil-dc.in/san (available for research purpose from TDIL)

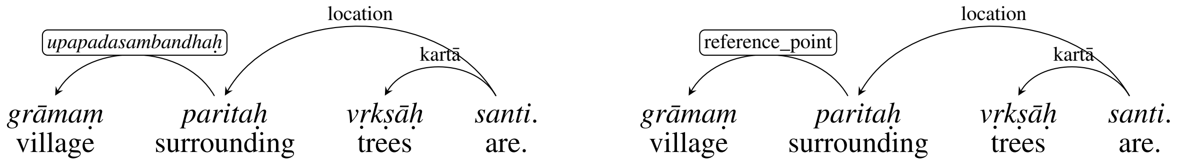


Figure 1: Old and New annotations

Another pair of relations that needed modification was '*anuyogī*' and '*pratiyogī*'. These were the relations used to connect two sentences by a connective. The two words *anuyogī* and *pratiyogī* are from the Indian logic which are used to refer to the two relata of a relation. In the old annotation scheme, some of the relations were not analysed semantically, and hence a general scheme of naming them as relata1 (*anuyogī*) and relata2 (*pratiyogī*) was followed. We illustrate this with an example. Consider the sentence

(2) *Skt: ahaṃ gṛhaṃ gacchāmi iti rāmaḥ avadat.*
Gloss: I home{sg,acc} go{pres,1p,sg} thus Rama{nom} say{past,3p,sg}.
Eng: Rama said that he goes home.

In this sentence the relation of the particle *'iti'* (thus) with *gacchāmi* (goes) and *avadat* (said) was marked as *pratiyogī* and *anuyogī* in the earlier version. The embedded sentence being the sentential argument, we propose *vākyakarma* (literally meaning 'sentential object') relation between the heads of the main and the embedded sentence. And *'iti'* serves as a marker for this relation, and hence it is marked as *vākyakarmadyotakaḥ* (literally meaning 'indicator of sentential argument').

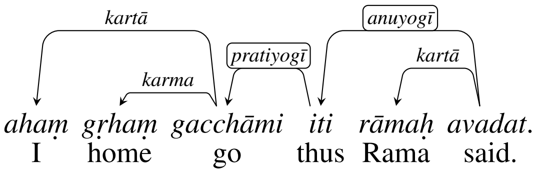


Figure 2: Complementiser: Old version

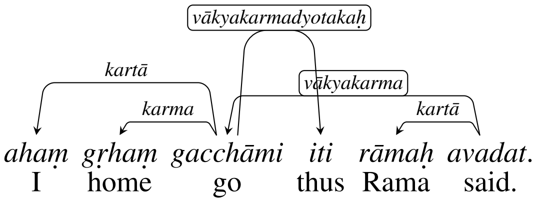


Figure 3: Complementiser: New version

Similarly consider the sentence

(3) *Skt: yadā meghāḥ varṣanti tadā mayūrāḥ nṛtyanti.*
Gloss: When cloud{pl,nom} rain{pres,3p,pl} then peacock{pl,nom} dance{pres,3p,pl}.
Eng: When clouds shower then peacocks dance.

In the earlier version the relations were as shown in Fig. 4. The two relations *anuyogī* (relata1) and *pratiyogī* (relata2) and the relation *sambandhaḥ* (literally 'relation') do not provide any semantics other than that the two words *yadā* (when) and *tadā* (then) are related to each other and they in turn are related to the finite verbs of the respective sentences. But what is the relation between them is not specified. In the revised scheme, these relations are changed as shown in Fig. 5. The modified version clearly marks the relation between co-relatives (when-then), and also marks the semantic relation of each of the co-relative with the verb as a time-location. The revised scheme thus provides a better semantics than the previous one.

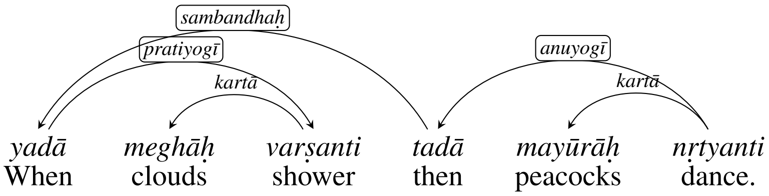


Figure 4: Co-reference: Old version

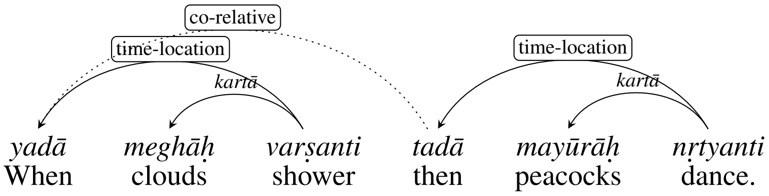


Figure 5: Co-reference: New version

Finally the third major modification was with regards to the co-ordinating conjuncts. In the earlier set of relations the conjunctive particle (*samuccaya-dyotakaḥ*) was marked as the head, connecting the conjuncting co-ordinates by a relation *samuccitam* as shown in Fig 6. This was modified as shown in Fig 7.
Let us look at the following sentence with a conjunct.

(4) *Skt:* *Rāmaḥ* *Sītā* *ca* *vanaṃ* *gacchati.*
Gloss: Rama{nom} Sita{nom} and forest{sg,acc} go{pres,3p,sg}
Eng: Rama and Sita go to forest.

Note here that the verbal form *gacchati* is in singular and not in dual.

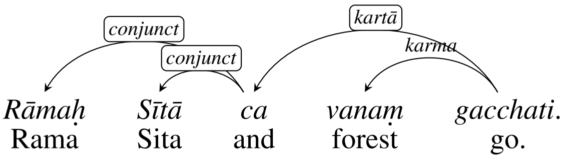


Figure 6: Conjuncts: Old version

In Sanskrit, it is observed that the last conjunct shows concord with the verb (Panchal and Kulkarni, 2019). The conjunctive particle acts as a marker, similar to the case suffix, to mark the relation between the two conjuncts. Hence in the modified analysis, the last conjunct in the phrase is marked as the head, with which the other conjunct is related by a *samuccitam* (conjunct) relation and the conjunctive particle is related to this head by the relation of *samuccaya-dyotakaḥ* (literary 'a marker for conjunction').

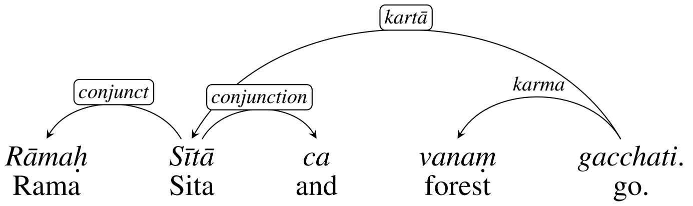


Figure 7: Conjuncts: New version

3.2 Saṁsādhanī Dependency Relations Version 2

The current version has 54 relations (see Appendix A) classified into the following categories.

- Predicate-argument relations
- Non-Predicate argument relations
    - verb-verb relations
    - verb-noun relations
    - noun-noun relations
- Relations due to special words
- Conjuncts and Disjuncts
- Miscellaneous

The predicate-argument relations are known as *kāraka* relations in Pāṇinian terminology. These are six in number with sub-classification of some of them. The six major relations are *kartā* (roughly agent), *karma* (roughly goal or patient), *karaṇam* (instrument), *sampradānam* (recipient), *apādānaṃ* (source) and *adhikaraṇaṃ* (location). If the activity involved is a causative one, then the agent of the basic activity is called *prayojya kartā* and the causative agent is called the *prayojaka kartā*. To account for the arguments of ditransitive verbs, we have introduced two sub categories of *karma* viz. *mukhyakarma* (primary object) and *gauṇakarma* (secondary object). These are something similar to, but not semantically equivalent to, direct and indirect object. As discussed in the previous section, a new tag *vākyakarma* is also introduced to mark a sentential argument to a verb.

Under the non-predicative arguments, the relations are categorised into three sub-categories. The relation of a finite verb with a non-finite verb marking precedence, simultaneity etc. forms the first category. The relation of a verb with a noun marking the cause or the purpose etc. constitutes the second sub-category. The genitive relation between two nouns, the adjectival relation, and the relation due to reduplication are some examples of the relations in the third sub-category. The relations in this category convey only a broad semantics. For example the genitive relation covers various semantic relations such as part-whole relation, kinship relation, and the possessive relation, and many more. Similarly the reduplication may mark a universal quantification, or intensity, etc. The exact semantics depends on the context.

The third category of relations is the set of relations due to certain special words called '*upapada*'s. These words govern the case suffix of the nouns they are in proximity with. Pāṇini has not discussed the semantics of these relations. We found that most of these words are related to the nouns whose case suffix they govern, and they indicate either a reference point or a comparison point. Then there are the relations due to conjuncts and disjuncts and a few miscellaneous relations. The detailed treatment of conjuncts is summarised in (Panchal and Kulkarni, 2019), and we do not discuss these here further. Finally there are relations between sentences. These are typically relations between two full sentences. These relations are marked by cetain indeclinable words such as if then (*yadi-tarhi*), because of (*tataḥ*), hence (*ataḥ*) etc. The relations between them are classified under miscellaneous, since, in the current guidelines we mark them as either relata1 and relata2, or just simply a relation. The terms 'relata1', 'relata2' and 'relation' do not provide any semantics. In Ramakrishnamacaryulu (2009), a semantic classification of inter-sentential relations is provided. The current guidelines need further enhancement to incorporate inter-sentential relations. This is out of scope of this paper and hence is not discussed.

3.3 Saṁsādhanī Parser

During the last decade there is an upsurge in the use of Machine Learning approaches for the development of Dependency parsers. Dependency parsers for several languages including Classical languages such as Latin and Greek are available. Most of these parsers follow the Data Driven approaches. The first parser for Sanskrit was built by Bhattacharyya (1986) using integer programming. Huet (2007) has a shallow parser that uses the minimal information of the transitivity of a verb as a sub-categorisation frame and models it as a graph-matching algorithm. The main purpose of this shallow parser is to filter out non-sensical segmentations. Hellwig et al. (2020) describe a syntactic labeler for manual annotation. This syntactic labeler expects a human being to select the pair of words, and the syntactic labeler suggests a label. This is a first stage towards developing an automatic full syntactic parser.

The first full-fledged parser for Sanskrit is described in Kulkarni (2019). This parser follows the Pāṇinian grammar and the theories of verbal cognition described in the Indian Sanskrit literature. The theories of verbal cognition describe three conditions necessary for verbal cognition. They are *ākāṅkṣā* (expectancy), *yogyatā* (meaning congruity) and *sannidhi* (proximity). Kulkarni (2019) has discussed the computational models of these three factors and describes the design of a parser following the theories of verbal cognition. This parser which is a part of the Saṁsādhanī platform, is implemented as an edge-centric binary join to build a dependency tree, in bottom-up approach, with local and global constraints on the edges and the edge labels. It uses the dependency relations provided in the Appendix A. It differs from the state-of-art parsers in the following aspects.

- It is a grammar based parser and follows the Indian theories of verbal cognition for parsing, while the current trend is to follow data driven approaches.
- It produces all possible parses while a typical parser produces only one parse. There are two reasons for allowing multiple parses. The first reason is, in Sanskrit we come across texts that have multiple readings. These multiple readings may be intended by the author or may be due to different philosophical interpretations. We would like to present all these readings to the reader. The second reason is, and this is purely due to the limitation of the implementation, the mutual congruency (semantic restrictions) between the word meanings is not checked while establishing the relations between words. This leads to over-generation and false positives. It is left to the readers to choose the correct parse from among the possible solutions.
- The solutions are ranked with a cost function which is defined as a sum of product of the cost associated with the relation and the distance between the two relata.
- The parse comes with an intelligent user interface and helps user to select the correct parse if the first parse is not correct.

4 Treebank

The first treebank of dependency analysis for Sanskrit was developed by the Consortium (SHMT-Consortium) executing the project entitled ‘Development of Sanskrit Computational Tools and Sanskrit-Hindi Machine Translation System’ sponsored by TDIL Programme, Ministry of Information Technology, Government of India, 2008-12. This treebank has 3000 sentences, mostly taken from the modern stories. However, this treebank is not available in public domain, and is available with the TDIL only for research. The second treebank was developed following the Universal Dependencies for a tiny corpus of 230 sentences from a *Pañcatantra* story (Dwivedi and Guha, 2017). The third treebank is the treebank of Vedic Sanskrit of 4004 sentences, which consists of both prose as well as verses, developed by Hellwig et al. (2020). This treebank also follows the Universal Dependencies.

We decided to develop a separate treebank from those described above. Firstly, since the dependency relations used by our parser for tagging are different from the Universal Dependency relations, the second and the third treebanks were not useful for us to evaluate our parser. Secondly we wanted to make the treebank thus developed open. The Saṁsādhanī platform contains three manually annotated texts. The first one is the *Saṅkṣepa Rāmāyaṇam* which has 100 verses. All these verses are tagged manually following the guidelines developed for the SHMT Consortium project. Shukla et al. (2013) reported a GOLD data of *Śrimad-Bhagavad-Gītā* (BhG), a philosophical text in verse form, consisting of 700 verses. This text was also tagged at various levels - metrical, segmentation, morphological and dependency (Patel, 2018). For the dependency level tagging, the guidelines of SHMT project were followed. The third manually annotated text consists of the first 10 Cantos of a poem *Śiśupālavadhaṁ*[7] which were tagged following the same guidelines.

While these three tagged texts were available under the Saṁsādhanī platform, we noticed that since these treebanks are created by individuals, and are not cross checked, there are a few inconsistencies. Meanwhile, the development of the parser also prompted us to improve upon the dependency relations. So these treebanks need to be modified as per the new guidelines and need to be cross checked as well for consistency in tagging. During the development of a parser, a need was also felt of controlled texts for testing. This led us to develop a new treebank. The sentences for this new treebank are chosen from four different sources. One set is from the grammar books to ensure that the treebank covers various types of constructions and special cases discussed in the grammar books covering various cases of subcategorization etc. The second set contains 284 sentences from a Sanskrit text book for 9th grade by NCERT (National Council for Education, Research and Training). These sentences are not isolated ones, but they constitute complete meaningful paragraphs or stories. The third set of sentences is from various books on Sanskrit learning. These are independent sentences covering wide vocabulary and syntactic constructions for the beginners. The fourth set of sentences is from the modern stories from a story book[8] which is being cross checked by the annotators. The annotation for *Śrīmad-Bhagavad-Gītā* is also being checked and corrected following the new guidelines. The treebank also contains a few verses from the first chapter of this poem. This treebank is available at http://sanskrit.uohyd.ac.in/scl/GOLD_DATA under the creative commons license.

4.1 Ambiguities during annotation

The annotation of all these four sets was checked by two or more of the authors independently. There were a few cases where there was a difference of opinion among the annotators. We discuss here an example of each type of the difference.

There were certain constructions involving non-finite verbs where two different annotations were possible. Here is an example.

(5)	*Skt:*	*Ṛṣīṇāṃ*	*vacanaṃ*	*pramāṇaṃ*	*asti.*
	Gloss:	Seer{pl,gen}	speech{sg,nom}	authentic{sg,nom}	be{pres,sg,3p}.

Eng: Seer's speech is authentic.

Here the word *Ṛṣīṇāṃ* is in genitive and hence it can be related to the following word *vacanaṃ* by a genitive relation. However, the word *vacanaṃ* itself is a gerund of the verb *vac* (to speak). Hence the relation of *Ṛṣīṇāṃ* with *vacanaṃ* may be considered to be that of a *kartā* (agent), according to Pāṇini's grammar.[9] In such cases we noticed that the annotators do not have consistency in tagging. This difference in tagging is probably not so important from the translation point of view, but it is important for the tasks such as information extraction, question answering etc. As far as the parser is concerned, it marks the relation as genitive if the genrund analysis is not available. If gerund analysis is available then it produces both the genitive as well as agent relation, giving priority to the agent relation. So the performance of the parser depends on the performance of the morphological analyser. Marking the relation as a genitive leads to loss of information. On the other hand, if the relation is marked as *kartā*, then one can always downgrade it to genitive, for translation purpose. A conscious effort on the part of the annotator is needed to mark such relations, and a good coverage morphological analyser producing

[7] `https://sanskrit.uohyd.ac.in/scl/e-readers/shishu/`

[8] "130 *Sanskrit kathā*", Dr. Narayan Shastri Kankar, Neetha Prakashan, New Delhi, 2007.

[9] *Kartṛkarmaṇoḥ kṛti* A2.3.65 - A *kartā* and a *karma* takes genitive case when the verb is in non-finite form denoting the activity.

analysis of derived stems is needed to get a correct parse.

Let us see another example.

(6) *Skt:* *mārgāḥ* *avaruddhāḥ* *bhavanti.*
Gloss: Road{pl,nom} blocked{pl,nom} be{pres,pl,3p}.
Eng: The roads are blocked.

Here the word *avaruddhāḥ* is a past participle of the verb *rudh* with prefix *ava*. Now this sentence can be analysed in two different ways as follows. The verb *bhū* may mean either 'to happen' or 'to become' and also 'to be'. Accordingly, we have two different interpretations.

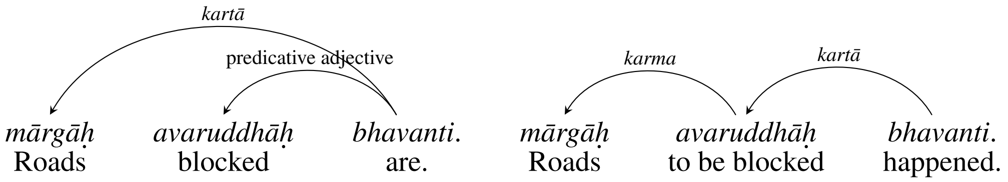


Figure 8: Inflectional Information

Both these analyses are correct. In the first one, the verb acts as a copula. The second one shows the analysis with the verbal meaning 'to happen', and 'being blocked' as its *kartā*. The *mārgāḥ* (roads) is, then, the object of blocking. As in the previous case, the first one is good enough for translation while the second one is better for deeper semantic analysis. In both the above cases, we propose that the manually tagged corpus should produce the analysis that uses the derivational information.

Another observation regarding tagging was with the elliptical sentences. Since Sanskrit is a highly inflectional language, there is no specific position (such as the Subject position in positional languages) that is sacrosanct. This allows Sanskrit to be a pro-drop language as well. Further, even the mandatory arguments such as *kartā* and *karma* may be dropped. For example, in an answer to a question '*rāmaḥ kutra agacchat*' (Where did Rama go?), a simple answer such as '*vanaṃ agacchat*' (went to a forest) is possible where the subject is ellipsed. Here the word *vanaṃ* is ambiguous between a nominative and an accusative analysis with the same stem *vana*. This leads to two parses, one with *vana* as an agent and another with *vana* as a goal. In the absence of any module to deal with meaning congruity between the verb and a noun, the parser fails to select one parse out of the two. The human annotator however marks the correct parse since he knows the meanings of the words. However there are cases where even for a human being the sentence is ambiguous, due to multiple morphological analyses. For example the causative form of the verb *katha* (to tell) is same as its non-causative form. Thus the word *kathayanti* may mean either tell or make somebody tell. So a simple sentence such as

(7) *Skt:* *mitrāṇi* *kathayanti.*
Gloss: friend{pl,nom/acc} tell{pres,pl,3p,[causative]}
Eng: Friends tell / (They) tell friends / Friends make (somebody) tell / (They) make (somebody) tell friends.

is ambiguous between four readings - friends is an agent, friends is a *karma*, friends is the causative agent, and finally friends is the *karma* (object) of the causative verb. This ambiguity is there for a human reader as well, since all the three interpretations are meaningwise compatible. In such cases the annotators are advised to mark all possible readings.

We present the last example where the arguments are shared. Consider an example with one verb in absolutive and the other one in finite form as follows.

(8) *Skt:* *rāmaḥ* *pustakaṃ* *krītvā* *paṭhati.*
Gloss: Rama{nom} book{sg,acc} purchase{abs} read{pres,sg,3p}.
Eng: Rama reads a book after purchasing it.

Here both the *kartā* as well as *karma* viz. Rāma and book are shared between the two verbs purchase and read. Pāṇini has provided a rule for the sharing of the *kartā*, and accordingly, we relate Rāma by the relation of *kartā* with the finite verb read. But, for the sharing of the *karma*, there is no rule in the grammar. Here we fall back to the default word order in prose for deciding which role to mark. If the verb in absolutive were intransitive, then the *karma* would have been always after this absolutive verb and before the final verb, in the default prose word order. Similarly, if the *karma* for both the verbs are different, then the *karma* for the finite verb would be just before it, and that of the one in absolutive would be before it. Taking clues from this, we mark the shared verb as an argument of the verb in absolutive, and then using the rule for sharing of arguments, we share it with the final verb. But if an annotator marks the relation the otherway, we do not want to penalise them. In other words, we provide both possible answers in such cases.

4.2 Evaluation

The sentences in the Saṁsādhanī treebank were run through the Saṁsādhanī parser. Table 1 shows the statistics of the treebank and the performance of the parser on the basis of following parameters: a) exact match, b) totally failed sentences, c) partially correct output, d) Labelled Attachment Score (LAS), and e) Unlabelled Attachment Score (UAS). Totally failed sentences are the ones which the parser fails to parse, either due to Out of Vocabulary words or if any word fails to get connected to any other word in the sentence. Partially correct output are the parses where at least one relation is wrong but not all.

Source	Sentences	Tokens	Exact Match	Failed	Partial Match	LAS	UAS
Grammar	468	1551	343	2 (.4%)	123	89%	97%
9^{th} grade	284	1393	183	15 (.6%)	87	82%	89%
Skt Learner	1070	4987	817	66 (6%)	181	88%	92%
BhG sample(verse)	36	313	7	3 (8%)	26	70%	76%
Average	1858	8244	1350	86 (4.6%)	417	85.5%	91.5%

Table 1: Performance of Parser

Thus we see that the performance of this parser is reasonably good. The percentage of failure is very small. The average LAS is 85.5% and the UAS is 91.5%. We notice that the performance of verse is not good. This is mainly due to some relations such as that of genitive and the adjectival which can move around freely.

The confusion matrix for some of the frequently occuring relations is shown in Table 2. The maximum confusion is with respect to the relation of *kartā* (roughly agent). There are two major reasons for the confusion of any relation with the other one. The first reason is, the relations share the same case marker. For example, both the cause and the instrument always take the instrumental case marker. And in the passive voice, *kartā* also takes the instrumental case marker. Therefore we see the confusion betwen a cause and an instrument and the *kartā*. Similarly the adjective of any of the predicate-argument relation always takes the case of its head noun. Since the relative word order for the adjective and the head noun is not fixed, in the absence of any semantic information about the adjective there is a confusion between which of the two substantives is the head and which one is an adjective. The confusion between a *kartā* and the predicative adjective is also essentially for the same reason. The second reason for the confusion is due to multiple morphological analyses of a word. For example, in the neuter gender, the accusative and nominative word forms are the same. This results in the confusion between a *kartā* and a *karma* (roughly goal).

machine→ manual↓	kartā (agent)	karma (goal)	adjective	pred adj	instrument	cause	..	Total
kartā (agent)	**1322**	14	10	26	6	6	..	1523
karma (goal)	31	**883**	7				..	1069
adjective	29	12	**260**				..	406
pred adj	23			**114**			..	162
instrument	5				**74**	8	..	99
cause					10	**40**	..	77
..	..	..	..	..	..	..	..	..
Total	1460	952	306	140	99	66	..	6226

Table 2: Confusion Matrix

5 Conclusion

In this paper we have discussed the first publicly available Sanskrit treebank developed following the dependency relations based on the Indian grammatical tradition. The presence of derivational analysis leads to deeper semantic analysis. At the same time it also introduces inconsistency in tagging, since most of the time for frequently used derived words such as *vacanam* (speech), the annotator may take these as underived and provide the dependency relations which do not show up the deeper analysis. Such deeper analysis is useful for certain tasks such as question answering and information retrieval, though might be irrelevant for the machine translation purpose.

We have also discussed the improved version of the dependency relations based on the Indian grammatical tradition. Three major improvements related to the treatment of the complementiser, conjunct and co-relative constructions were discussed. The modified version reflects the associated semantics.

Finally we have tested the dependency parser for Sanskrit on the treebank, and noted that the performance of the parser is reasonably good. The confusion matrix conforms with the grammatical sources of ambiguities. The proper modeling of mutual congruency would help in improving the performance of the parser.

References

V. S. Apte. 1885 [1925]. *The Student's Guide to Sanskrit Composition.* The Standard Publishing Company, Girgaon, Bombay, 9 edition.

David Bamman and Gregory Crane. 2011. The ancient greek and latin dependency treebanks. In Caroline Sporleder, Antal van den Bosch, and Kalliopi Zervanou, editors, *Language Technology for Cultural Heritage*, pages 79–98, Berlin, Heidelberg. Springer Berlin Heidelberg.

E Bejček, E Hajičovà, J. Jajič, P. Jinovà, V. Kettnerovà, V. Kolářová, M. Mikulová, J. Mirovskỳ, , A. Nedoluzhko, J. Panevová, L. Polákovà, M. Ševčkovà, J. Štěpànek, and S. Zikánová. 2013. Prague ependency treebank 3.0. Technical report, Institute of Formal and Applied Linguistics, Charles University.

Akshar Bharati and Amba Kulkarni. 2010. Information coding in a language: Some insights from paninian grammar. *Dhīmahi, Journal of Chinmaya International Foundation Shodha Sansthan*, I(1):77–91.

Akshar Bharati and Amba Kulkarni. 2011. 'Subject' in English is abhihita. In Ashok Aklujkar George Cardona and Hideyo Ogawa, editors, *Studies in Sanskrit Grammars (Proceedings of the Vyakarana Section of the 14th World Sanskrit Conference).* D.K. Printworld.

Akshar Bharati and Rajeev Sangal. 1990. A karaka based approach to parsing of indian languages. In *Proceedings of International Conference on Computational Linguistics (Vol. 3)*, Helsinki, Association for Computational Linguistics NY.

Akshar Bharati and Rajeev Sangal. 1993. Parsing free word order languages in the paninian framework. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 105–111. acl.

Akshar Bharati, Vineet Chaitanya, and Rajeev Sangal. 1995a. *Natural Language Processing: A Paninian Perspective*. Prentice-Hall, New Delhi.

Akshar Bharati, Ashok Gupta, and Rajeev Sangal. 1995b. Parsing paninian grammar with nesting constraints. In *Proceedings of 3rd NLP Pacific Rim Symposium*, pages 1–6.

Akshar Bharati, Rajeev Sangal, Vineet Chaitanya, Amba Kulkarni, Dipti Misra Sharma, and K. V. Ramakrishnamacharyulu. 2002. AnnCorra: Building tree-banks in Indian languages. In *COLING-02: The 3rd Workshop on Asian Language Resources and International Standardization.*

Pushpak Bhattacharyya. 1986. A system for sanskrit to hindi translation. Master's thesis, IIT Kanpur.

A. Böhmovà, E. Hajičovà, and B. Hladkà. 2003. The prague dependency treebank. 20.

George Cardona. 2007. *Pāṇini and Pāṇinīyas on śeṣa Relations*. Kunjunni Raja Academy of Indological Research Kochi.

George Cardona. 2009. On the structure of Pāṇini's system. In Gérard Huet, Amba Kulkarni, and Peter Scharf, editors, *Sanskrit Computational Linguistics 1 & 2*. Springer-Verlag LNAI 5402.

John Caroll, Guido Minnen, and Ted Briscoe. 1999. Corpus annotation for parser evaluation. In *Proceedings of EACL.*

Harold G Coward. 1983. *Studies in Indian Thought.* Motilal Banarasidas.

Sleator Daniel and Davy Temperley. 1993. Parsing english with a link grammar. In *Third International Worshop on Parsing Technologies.*

Puneet Dwivedi and Easha Guha. 2017. Universal dependencies of sanskrit. *International Journal of Advance Research, Ideas and Innovations in Technology*, 3:479–482.

C J Fillmore, C R Johnson, and M R L Petruck. 2003. Background to framenet. *Intenational journal of Lexicography*, 16(3):235–250.

Brendan S. Gillon. 2002. Bhartṛhari's rule for unexpressed kārakas: The problem of control in classical sanskrit. *Indian Linguistic Studies, Festschrift in Honor of George Cardona.*

Pawan Goyal, Vipul Arora, and Laxmidhar Behera. 2009. Analysis of Sanskrit text: Parsing and semantic relations. In Gérard Huet, Amba Kulkarni, and Peter Scharf, editors, *Sanskrit Computational Linguistics 1 & 2*, pages 200–218. Springer-Verlag LNAI 5402.

Pawan Goyal, Gérard Huet, Amba Kulkarni, Peter Scharf, and Ralph Bunker. 2012. A distributed platform for sanskrit processing. In *Proceedings of 24th COLING*, Mumbai India.

Oliver Hellwig, Salvatore Scarlata, Elia Ackermann, and Paul Widmer. 2020. The treebank of vedic sanskrit. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 5137–5146. European Language Resources Association.

Oliver Hellwig. 2009. Extracting dependency trees from sanskrit texts. In Amba Kulkarni and Gérard Huet, editors, *Third International Sanskrit Computational Linguistics Symposium*, pages 106–115. Springer-Verlag LNAI 5406.

Gérard Huet. 2007. Shallow syntax analysis in Sanskrit guided by semantic nets constraints. In Majumdar, Mitra, and Parui, editors, *Proceedings of the 2006 International Workshop on Research Issues in Digital Libraries*, New York NY USA. ACM Digital Library.

Gérard Huet. 2009. Formal structure of Sanskrit text: Requirements analysis for a mechanical Sanskrit processor. In Gérard Huet, Amba Kulkarni, and Peter Scharf, editors, *Sanskrit Computational Linguistics 1 & 2*. Springer-Verlag LNAI 5402.

Madhusoodan Pai J. 2020. *Sanskrit Sentence Generator: A Prototype*. Ph.D. thesis, University of Hyderabad, Hyderabad.

J J Katz and J A Fodor. 1963. The structure of a semantic theory. *Language*, 39:170–210.

Tracy H. King, Richard Crouch, Stefan Riezler, Mary Dalrymple, and Ronald Kaplan. 2003. The PARC 700 dependency bank. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC-03)*.

Paul Kiparsky. 2009. On the architecture of panini's grammar. In Gérard Huet, Amba Kulkarni, and Peter Scharf, editors, *Sanskrit Computational Linguistics 1 & 2*, pages 33–94. Springer-Verlag LNAI 5402.

G.-J. M. Kruijff. 2002. Formal and computational aspects of dependency grammar: History and development of dpendency grammar. Technical report.

Amba Kulkarni and K. V. Ramakrishnamacharyulu. 2013. Parsing Sanskrit texts: Some relation specific issues. In Malhar Kulkarni, editor, *Proceedings of the 5th International Sanskrit Computational Linguistics Symposium*. D. K. Printworld(P) Ltd.

Amba Kulkarni and Dipti Sharma. 2019. Pāṇinian syntactico-semantic relation labels. In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 198–208, Paris, France, August. Association for Computational Linguistics.

Amba Kulkarni, Sheetal Pokar, and Devanand Shukl. 2010. Designing a Constraint Based Parser for Sanskrit. In G N Jha, editor, *Fourth International Sanskrit Computational Linguistics Symposium*, pages 70–90. Springer-Verlag, LNAI 6465.

Amba Kulkarni. 2013. A deterministic dependency parser with dynamic programming for Sanskrit. In *Proceedings of the Second International Conference on Dependency Linguistics (DepLing 2013)*, pages 157–166, Prague, Czech Republic, August. Charles University in Prague Matfyzpress Prague Czech Republic.

Amba Kulkarni. 2016. Application of modern technology to south asian languages. In Hans Henrich Hock and Elena Bashir, editors, *The Languages And Linguistics of South Asia: A comprehensive Guide*, pages 744–747. De Gruyter.

Amba Kulkarni. 2019. *Sanskrit parsing based on the theories of śābdabodha*. Indian Institute of Advanced Study, Shimla and D K Publishers (P) Ltd.

Amba Kulkarni. 2020a. Appropriate dependency tagset for sanskrit analysis and generation. *Acta Orientalia*, forthcoming.

Amba Kulkarni. 2020b. Sanskrit parsing following the indian theories of verbal cognition. *TALLIP*, forthcoming.

B. Levin and M Hovav Rappaport. 2005. *Argument realization*. Cambridge University Press.

Dekang Lin. 2003. Dependency-based evaluation of MINIPAR. In Abeillé Anne, editor, *Treebanks: Building and Using Parsed Corpora*, pages 317–329, Dordrecht. Springer Netherlands.

Marie-Catherine de Marneffe and Christopher D Manning. 2008. Stanford dependencies manual.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*.

Ignor Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. The SUNY Press, Albany, New York.

J. Nivre, M.-C de Marneffe, F. Ginter, Y. Goldberg, J. Hajič, C. D. Manning, R. T. McDonald, S. Petrov, S. Pyysalo, N. Silveira, R. Tsarfaty, and D. Zeman. 2016. Universal dependencies c1: A multilingual treebank collection.

M. Palmer, D. Gildea, and N. Xue. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Sanjeev Panchal and Amba Kulkarni. 2018. Yogyatā as an absence of non-congruity. In Gérard Huet and Amba Kulkarni, editors, *Computational Sanskrit and Digital Humanities*. D K Publishers.

Sanjeev Panchal and Amba Kulkarni. 2019. Co-ordination in Sanskrit. *Indian Linguistics*, 80(1-2):59–176.

Sanjeev Panchal. 2020. *Modelling Ākāṅkṣā following Pāṇinian Grammar for Sanskrit sentential parsing*. Ph.D. thesis, University of Hyderabad, Hyderabad.

Gopal Dutt Pande. 2004. *Aṣṭādhyāyī of Pāṇini elaborated by M.M.Panditraj Dr. Gopal Shastri*. Chowkhamba Sur bharati Prakashan Varanasi.

Preeti Patel. 2018. *E-teaching capsule for Śrīmadbhagvadgītā*. Ph.D. thesis, University of Hyderabad, Hyderabad.

K V Ramakrishnamacaryulu. 2009. Annotating Sanskrit texts based on Śābdabodha systems. In Amba Kulkarni and Gérard Huet, editors, *Proceedings Third International Sanskrit Computational Linguistics Symposium*, pages 26–39, Hyderabad India. Springer-Verlag LNAI 5406.

N S Ramanujatatacharya. 2005. *Śābdabodha Mīmāṁsā*. Institute Francis De Pondicherry.

Bhā. Va. Rāmapriya and V. Saumyanārāyaṇa. 2001. Saṅgaṇakayantre nyāyaśāstrīyaśābdabodhaḥ. *Journal of Foundation Research*, VI(1–2):61–68.

Phillip Resnik. 1993. Semantic classes and syntactic ambiguity. In *ARRPA Workshop on Human Language Technology*. Princeton.

Preeti Shukla, Amba Kulkarni, and Devanand Shukl. 2013. Geeta: Gold standard annotated data, analysis and its applicationa. In *Proceedings of ICON 2013, the 10th International Conference on NLP*, Noida, India, December.

J. S. Speijer. 1886; Reprint 2009. *Sanskrit Syntax*. Motilal Banarsidass New Delhi.

Veluri Subbarao. 1969. *The philosophy of a sentence and its parts*. Munshiram Manoharlal, Delhi.

Lucien Tesnière, editor. 1959. *Éléments de Syntaxe Structurale*. Klincksieck Paris.

Gary A Tubb and Emery R Boose. 2007. *Scholastic Sanskrit: A Handbook for students*. The American Institute of Buddhist Studies at Columbia University in the City of New York New York.

A Saṁsādhanī Dependency Relations

- **Predicate argument relations**
    - *kartā* (agent)
        - *prayojaka-kartā* (causative agent)
        - *prayojya-kartā* (causee)
    - *karma* (goal/patient)
        - *mukhya-karma* (direct object)
        - *gauṇa-karma* (indirect object)
        - *vākya-karma* (sentential argument)
    - *karaṇam* (instrument)
    - *sampradānam* (recipient)
    - *apādānam* (source)
    - *adhikaraṇam* (location)
        - *kāla-adhikaraṇam* (location of time)
        - *deśa-adhikaraṇam* (location of space)
        - *viṣaya-adhikaraṇam* (locus indicating the subject)
        - *lyapkarma-adhikaraṇam* (*karma* of an ellipsed absolutive verb form marked as a location)
- **Non Predicate argument relations**
    - **Verb-Verb relations**
        - *pūrva-kālaḥ* (precedence)
        - *vartamāna-samāna-kālaḥ* (simultaneity in present)
        - *bhaviṣyat-samāna-kālaḥ* (simultaneity in future) tense
        - *bhāvalakṣaṇa-pūrva-kālaḥ* (simultaneity in the past without sharing of arguments)
        - *bhāvalakṣaṇa-vartamāna-samāna-kālaḥ* (simultaneity in present without sharing of arguments)
        - *bhāvalakṣaṇa-anantara-kālaḥ* (simultaneity in future without sharing of arguments)
        - *sahāyaka-kriyā* (auxiliary verb)
    - **Verb-noun relations**
        - *sambodhyaḥ* (vocative)
        - *hetuḥ* (cause)
        - *prayojanam* (purpose)
        - *kartṛ-samāna-adhikaraṇam* (predicative adjective)
        - *karma-samānādhikaraṇam*
        - *kriyā-viśeṣaṇam*(manner adverb)
        - *atyanta-saṃyogaḥ*(total contact)
        - *apavarga-sambandhaḥ*
        - *pratiṣedhaḥ* (negation)
    - **Noun-Noun relations**
        - *ṣaṣṭhī-sambandhaḥ* (genitive)
        - *aṅga-vikāraḥ* (body-deformity)
        - *vīpsā* (reduplication)
        - *viśeṣaṇam* (adjective)
        - *sambodhana-sūcakam* (vocative marker)
        - *abhedaḥ* (indifference)
        - *nirdhāraṇam* (determiner)
        - *vākya-karma-dyotakaḥ* (complementiser)
        - *tīvratādarśī* (intensifier)
        - *nāma* (name)
- **Relations due to special words**
    - *sandarbha-binduḥ* (reference point)

- - *tulanābinduḥ* (comparison point)
  - *udgāravācakaḥ* (exclamatory)
  - *saha-arthaḥ* (association)
  - *vinā-arthaḥ* (disassociation)
- **Miscelleneous**
  - *anuyogī* (relata1)
  - *pratiyogī* (relata2)
  - *nitya-sambandhaḥ* (co-reference)
  - *sambandhaḥ* (relation)
- **Conjunct-disjunct**
  - *samuccitaṁ* (conjunct)
  - *samuccaya-dyotakaḥ* (conjunction)
  - *anyataraḥ* (disjunct)
  - *anyatara-dyotakaḥ* (disjunction)

Note: The bold entries are the headings and do not indicate relation labels.
We have not provided the gist/translation of these relation tags. The readers are encouraged to refer to the tagging guidelines available at `http://sanskrit.uohyd.ac.in/scl/GOLD_DATA/Tagging_Guidelines/guidelines.html`.

AlpinoGraph: A Graph-based Search Engine for Flexible and Efficient Treebank Search

Peter Kleiweg
University of Groningen
`p.c.j.kleiweg@rug.nl`

Gertjan van Noord
University of Groningen
`g.j.m.van.noord@rug.nl`

Abstract

AlpinoGraph is a graph-based search engine which provides treebank search using SQL database technology coupled with the Cypher query language for graphs. In the paper, we show that AlpinoGraph is a very powerful and very flexible approach towards treebank search. At the same time, AlpinoGraph is efficient. Currently, AlpinoGraph is applicable for all standard Dutch treebanks. We compare the Cypher queries in AlpinoGraph with the XPath queries used in earlier treebank search applications for the same treebanks. We also present a pre-processing technique which speeds up query processing dramatically in some cases, and is applicable beyond AlpinoGraph.

1 Introduction

Traditionally, treebanks are, of course, collections of trees. Search engines for treebanks therefore often exploit this tree-like nature. Early treebank search tools such as `tgrep`, `tgrep2`, `lpath` (Rohde, 2001; Bird and Lai, 2010) provide a specialized query language over trees. For Dutch, similarly, current tools (van Noord, 2009; Augustinus et al., 2012; van Noord et al., 2013; Odijk et al., 2017; Augustinus et al., 2017; van Noord et al., 2020) are built with the XPath query language which is a standard query language for XML documents. XML documents are, in essence, trees too.

Obviously, not all linguistic annotations fit the concept of trees, and in most treebanks there are ways to encode, for instance, discontinuous constituents, secondary edges, enhanced dependencies etc. Also, feature structures such as those that arise in constraint-based grammatical frameworks (LFG, HPSG, ...) are directed graphs, not trees. It can be argued, therefore, that graphs are a better representation for linguistic annotation. And indeed, several treebank search systems have been based on graphs (Mírovský, 2008; Proisl and Uhrig, 2012; Bonfante et al., 2018).

In this paper, we argue in addition that a graph-like representation is useful because it allows for a straight-forward combination of different types of annotation and annotation layers. In the AlpinoGraph application, four different annotation layers are combined (automatically), including two layers for Universal Dependencies (standard and enhanced) (Nivre et al., 2018), (Bouma and van Noord, 2017), the original Lassy annotation layer (van Eynde, 2005; van Noord et al., 2019), and a simple layer of word pairs inherited from PaQu (Odijk et al., 2017).

The representations used in AlpinoGraph are automatically derived from existing treebanks in the Lassy XML format, a hybrid dependency format with some categorical information as well, originally based on and developed as an alternative of the format used in the Tiger treebank (Brants et al., 2004) and the Dutch Spoken Corpus (CGN)(Schuurman et al., 2003). In addition, information is derived from the CoNLL-U format for Universal Dependencies. In fact, the UD treebanks for Dutch are automatically derived from the treebanks in the Lassy XML format. It should be straightforward to map treebanks in CoNLL-U format (including all UD treebanks) into AlpinoGraph.

In this paper, we do not consider the potential linguistic advantages of graph-based representations, since the linguistic annotations are derived from existing resources, and no further manual annotation efforts have been invested for AlpinoGraph.

AlpinoGraph is built on AgensGraph. AgensGraph provides database technology (PostgreSQL) with the standard search language for graphs, Cypher. This combination provides, on the one hand, a very powerful query language which allows to express very complex linguistic patterns. On the other hand, the database tools ensure a very flexible tool which not only is capable of identifying relevant sentences, but also provides a wealth of functionality for aggregating the information of relevant sentences, structures or words.

In the next sections, we describe how treebanks are represented as graphs, and how we can formulate simple queries over such treebanks. In the fourth section, we compare AlpinoGraph on the full set of more than a hundred queries that are available in the SPOD extension of PaQu (van Noord et al., 2020). This comparison illustrates not only that the tool provides the required expressive power, but also shows that the tool is much faster for our purposes. In section 5, we present a search optimization technique which improves speed for some queries enormously. The technique appears to be applicable for most other treebank search systems.

2 Treebanks as Graphs

Graphs consist of vertices and edges. In AlpinoGraph, vertices can be words as well as constituents. A vertex is written as `()`. A vertex of type word is written as `(:word)`, and a vertex of a higher level constituent, a "node", is written as `(:node)`. We can use the notation `(:nw)` as an alias for a vertex that could be either a word or a node.

If we want to provide further information of a vertex, we use attribute and values within curly brackets. For instance, `(:node{cat:'np'})`, denotes a noun phrase. If we need to refer to a particular vertex, we can place a variable directly after the opening bracket: `(v:node{cat:'np'})`. Here, `v` functions as a variable that we can refer to later.

Edges are represented in much the same way, except that square brackets are used. We use edges to represent universal dependencies. Such dependencies are of type `ud`. For example, the direct object universal dependency is written as `[:ud{rel: 'obj'}]`.

We can use path expressions to combine vertices and edges. Such expressions look like:

```
() -[]-> ()
() <-[]- ()
```

Between brackets, we can specify further requirements. For instance, the following expression describes the direct object relation between a verb and some word `n`:

```
(:word{upos:'verb'}) -[:ud{rel:'obj'}]-> (n:word)
```

If this expression is used in a search, the variable `n` would be instantiated to (heads of) direct objects of verbs.

Each sentence in AlpinoGraph is represented by a graph where the vertices are words and nodes, and a single vertex of type `:sentence`. The attributes of the words are all the attributes available in the standard Lassy annotation guidelines (van Eynde, 2005; van Noord et al., 2019), as well as all the attributes in the UD representation. The attributes of nodes include the attribute `cat` and a few others that we can ignore for now. Multi-word units also have attributes for `word` and `lemma`.

The edges come in four different types for the representation of dependencies: standard universal dependencies `:ud`, enhanced universal dependencies `:eud`, Lassy dependencies `:rel` and simplified Lassy dependencies `:pair` (inherited from the word-pair part of the PaQu search tool). A fifth type of edge is `:next` which links each word to the next word in the sentence.

The Lassy-type dependencies look as follows:

```
(:sentence) -[:rel{rel: 'top'}]-> (:node{cat: 'top'})
(:node) -[:rel]-> (:node)
(:node) -[:rel]-> (:word)
(:node) -[:rel]-> (:nw)
```

The standard UD-type dependencies look as follows:

```
(:sentence) -[:ud{rel: 'root'}]-> (:word)
(:word) -[:ud]-> (:word)
```

And words are connected by means of the `:next` edges:

```
(:word) -[:next]-> (:word)
```

Paths can be longer than an edge connecting two vertices. This path identifies the root of the sentence that has a subject:

```
(:sentence) -[:ud]-> () -[:ud{rel:'nsubj'}]-> ()
```

3 AlpinoGraph by Example

Simple AlpinoGraph queries can be built using the path expressions of the previous section. For example:

```
match (:word{lemma:'drinken'})-[:ud{rel:'obj'}]->(o:word{upos:'NOUN'})
return o
```

For a given corpus, this query will return all direct object nouns of the verb with lemma *to drink*. It is straightforward to combine edges of different types in a query. For instance, suppose you are interested to find (heads of) direct objects which are double-quoted. This can be accomplished by identifying direct objects (in the first clause), and then requiring that both the words to the left and the right are double-quotes:

```
match ()-[:ud{rel:'obj'}]->(o:word),
      (:word{lemma:'"'}) -[:next]-> (o)-[:next]->(:word{lemma:'"'})
return o
```

Queries can also return multiple values. And the values need not be vertices, but could also be edges. Using the '.'-operator you can also return the attributes of vertices or edges. The following example finds nodes with a verb as the head and an indirect object. The result is a table of pairs consisting of the lemma of the verb and the category of the indirect object.

```
match (v:word{pt:'ww'})<-[:rel{rel:'hd'}]-(:node)-[:rel{rel:'obj2'}]->(w:node)
return v.lemma, w.cat
```

It is straightforward to add further conditions on a pattern. The following example provides an illustration, where we want to collect direct objects of the verb "to eat", but ignoring the cases where the direct object is a pronoun:

```
match (:word{lemma:'eten'})-[:ud{main:'obj'}]->(w2:word)
where w2.upos != 'PRON'
return w2
```

In addition to simply returning the matches, we can perform a variety of aggregations on those.

```
match (:word{lemma:'eten'})-[:ud{main:'obj'}]->(w2)
where w2.upos != 'PRON'
return w2.lemma, count(w2.lemma) as frequency
order by frequency desc
```

This results in a table of lemmas with their respective frequencies in decreasing order.

4 Representing secondary edges

In the Lassy treebank, secondary edges are represented using an index attribute associated to nodes of the tree to indicate reentrancies in the graph. In AlpinoGraph, such secondary edges are represented in much the same way as primary edges (although an attribute is added to ensure that the difference can be recovered in the relevant cases). An example will illustrate this.

In the annotation of passives, the subject of the passive auxiliary is also annotated to be the object of the embedded verb. An as example, sentence 1 gets analysed as in the left part of figure 1. In contrast, such "secondary edges" are represented in AlpinoGraph as first class citizens. Since AlpinoGraph is graph-based, there is no problem by having two edges connecting to "het brood". In AlpinoGraph, the resulting graph is displayed on the right of figure 1 (including the UD representation layer for further illustration).

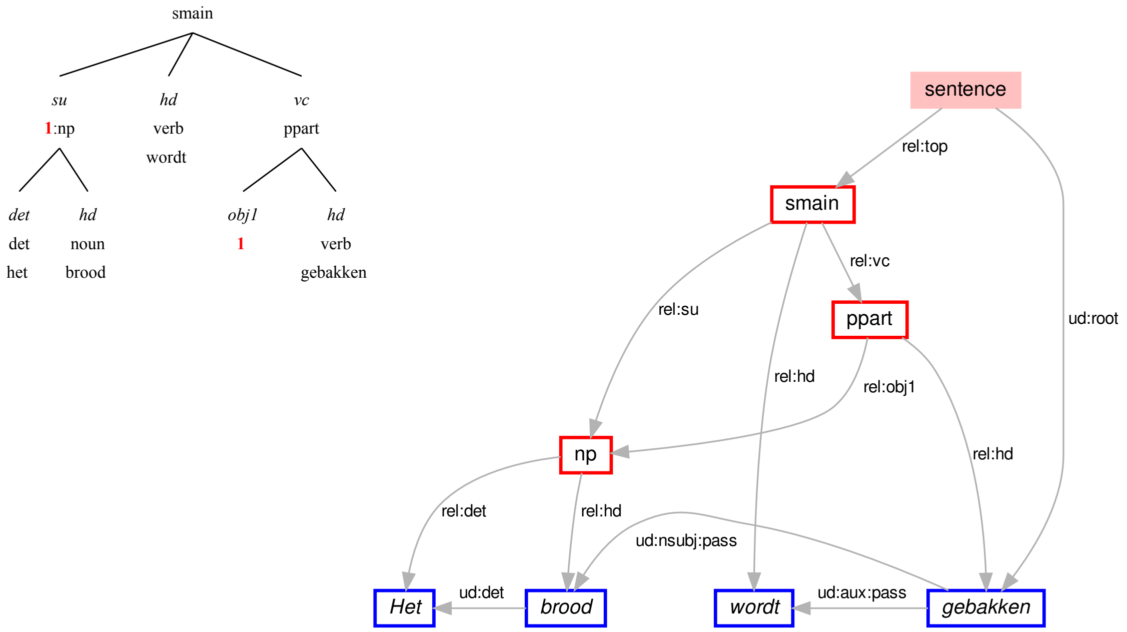


Figure 1: Original Lassy annotation of *Het brood wordt gebakken* (left) and the representation in AlpinoGraph (right), displaying two (:rel and :ud) of the available representation layers.

(1) Het brood wordt gebakken
    The bread is baked

5 Comparison with XPath

5.1 Treebanks and queries

In this section, we compare the Cypher queries of AlpinoGraph with equivalent XPath queries used in the earlier treebank search systems DACT(van Noord et al., 2013), GrETEL(Augustinus et al., 2012), PaQu(Odijk et al., 2017). The comparison between XPath and Cypher is based on the same treebanks, for a large number of queries. We thus need a large number of relevant linguistic queries. This representative set of linguistic queries is taken from the SPOD extension of PaQu. SPOD (Syntactic Profiler of Dutch) (van Noord et al., 2020) provides an interface to a set of over a hundred linguistic queries which can be used to compare texts and corpora. These queries are supposed to be generally useful to obtain a good characterization of the syntactic properties of a text. SPOD has been used to study, for instance, the writing development of Dutch school children. The list of queries has been established in close connection with linguists.

The queries are applied for four different treebanks, described here as follows.

Alpino Treebank. The Alpino Treebank (van der Beek et al., 2002) contains over 7 thousand manually annotated sentences which constitute the newspaper ("cdbl") part of the Eindhoven corpus (uit den Boogaart, 1975). This treebank is one of the UD treebanks. It is available both in CoNLL-U and Lassy XML format.

CGN. The CGN treebank contains the manually syntactically annotated part of CGN ("Corpus of Spoken Dutch") (Schuurman et al., 2003). The treebank consists of 1 million words. The CGN annotation format has been automatically converted to the Lassy XML format.

Eindhoven. The Eindhoven treebank contains over 40 thousand *automatically* annotated sentences. The annotations are provided by the Alpino parser (van Noord, 2006), in the Lassy XML format.

Lassy Small. The Lassy Small treebank (van Noord et al., 2013) is the de facto standard treebank of written Dutch. The size of the manually annotated corpus is 1 million words, and the corpus consists of a variety of text types. Part of this treebank is available as one of the UD treebanks (the limitation is due to copyright reasons).

The list of linguistic queries from SPOD contains 102 items (we ignore the queries about parser performance since most of our treebanks are manually developed). Of those, 18 queries are not available for the timing experiment because the Cypher queries exploit an efficiency improvement which we will only discuss in section 5. Since that improvement is somewhat independent of the actual query engine, including those queries here would be unfair. A further complication is that the automatically annotated treebanks include some information on separable verb prefixes that is not available in the manually annotated treebanks. SPOD includes 6 queries which focus on that information, so naturally those 6 queries are only applied for the Eindhoven treebank. Finally, the CGN treebank pre-dates the other treebanks and does not include certain types of secondary edges which have been added systematically to later treebanks. For that reason, three queries are not applicable to the CGN treebank. Table 1 summarizes the number of queries used per treebank. We list both the number of queries used to compare the results (left) and the number of queries used in the timing experiment (right).

	results	timing
queries in SPOD	102	84
Eindhoven	102	84
Lassy Small	96	78
Alpino Treebank	96	78
CGN	93	75

Table 1: Number of queries used in the two experiments per treebank. On the left, the number of queries used to compare the number of results. On the right, the number of queries used for the timing experiment.

5.2 Differences in query results

The queries available in SPOD have all been re-implemented in AlpinoGraph. As a consequence, we can compare the results of running the original XPath queries on the one hand, and running the newly implemented Cypher queries in AlpinoGraph on the other hand. During the development of the Cypher queries, we carefully compared if the Cypher queries returned the same hits as the corresponding XPath query. In a limited number of cases, it turned out quite hard to obtain precisely the same set of hits. There are two classes of cases where the number of hits differs for some of the queries. Firstly, while we were re-implementing the queries in AlpinoGraph we found a number of subtle problems with the original XPath queries. A few cases are reported below. Secondly, a further important difference is the representation of "secondary edges".

5.2.1 Query improvements

During the process of re-implementing the SPOD queries in AlpinoGraph, we encountered a small number of subtle problems with the original XPath queries.

A simple example concerns the identification of noun phrases. Word groups are labeled by a category attribute, so any node with category "np" is a noun phrase. However, category features are used only for word groups and not for single words. Therefore, single-word noun phrases such as pronouns do not have a category attribute. If noun phrases have to be identified in XPath queries, a disjunction is used to include both word groups with the relevant category attribute as well as single words with appropriate part-of-speech attributes. A further complication arises for coordination. A coordination of two noun phrases is assigned "conj" as category attribute, not "np". In PaQu, a macro is defined to specify what it means to be a noun phrase. That macro essentially states that you are a noun phrase if you are a basic noun phrase, or if you are a coordination of basic noun phrases. And a basic noun phrase is a word group with category "np", or a word with the appropriate part of speech tag (noun, pronoun, proper name). This definition missed the cases where a conjunction was built up of two NP conjunctions, as in:

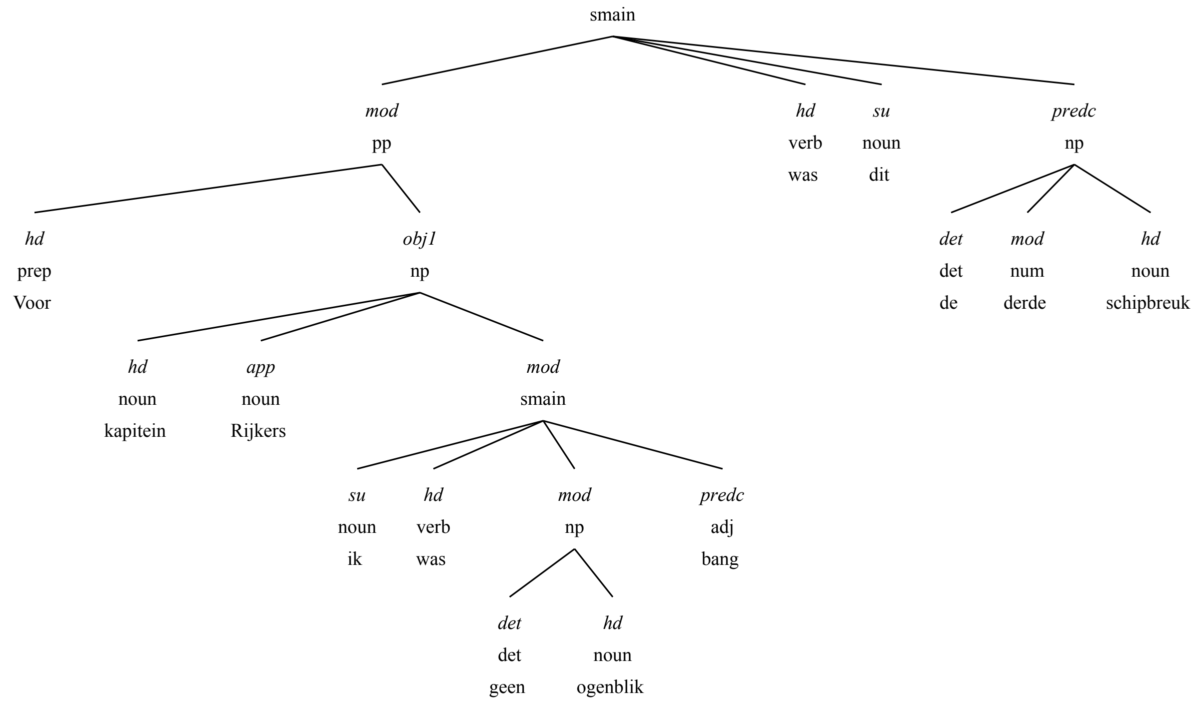


Figure 2: Annotation of *Voor kapitein Rijkers "(ik was geen ogenblik bang)" was dit de derde schipbreuk*

(2) de problemen van misdaad en straf , schuld en vergeving
the problems of crime and punishment , guilt and forgiveness

A more intricate example concerns the definition of the topic position in main sentences, in Germanic studies often called the "vorfeld": the word group that precedes the finite verb in V2 main sentences. In the hybrid dependency annotation format, it is somewhat complicated to define this word group. The word group should be a dependent (either direct or indirectly) of the finite verb, and it should (directly) precede that finite verb.

(3) het huis op de heuvel wordt verkocht
the house on the hill is sold

In this example, "het huis op de heuvel" satisfies those conditions and is a potential vorfeld. In order to rule out dependents of the actual vorfeld constituent, in the example "op de heuvel", the XPath query furthermore required that the vorfeld candidate should not be part of a constituent which is itself a vorfeld candidate. However, after comparing the results of the XPath query and the Cypher variant, it became apparent that this added condition was a bit too strict. That condition also rules out vorfelds of embedded main sentences. An example is listed where, with the analysis illustrated in figure 2. The original XPath query thus missed the fact that "ik" here also should be considered a vorfeld constituent.

(4) Voor kapitein Rijkers " (ik was geen ogenblik bang) " was dit de derde schipbreuk .
for captain Rijkers " (I was no moment afraid) " was this the third shipwreck .

5.3 Differences for secondary edges

Complements of fixed verbal expressions are labeled using the relation "svp". In a few cases, such a fixed part of a fixed verbal expression also functions as the subject in a passive-like construction, as in example 5, analysed as in the left part of figure 3.

(5) Wel werd meteen groot alarm geslagen
Indeed was immediately major alarm raised
"however, a major alarm was immediately raised"

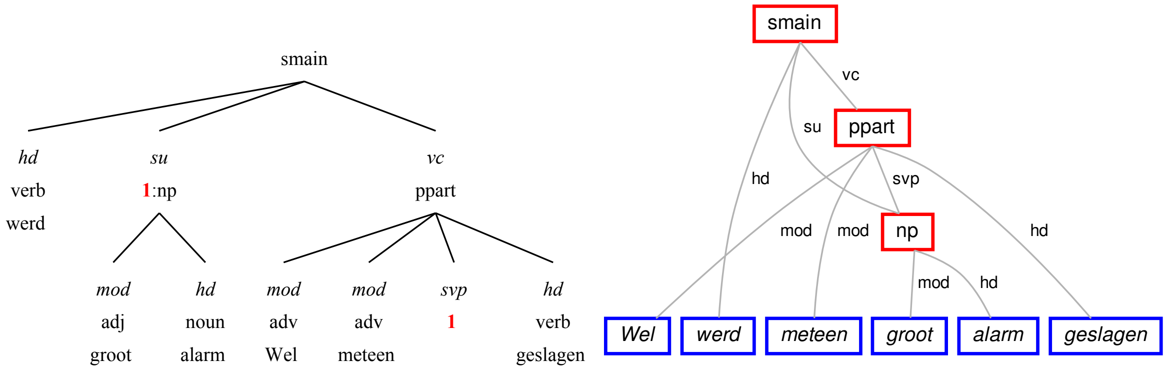


Figure 3: Representation of *Wel werd meteen groot alarm geslagen* in Lassy XML (left) and the :rel representation layer in AlpinoGraph (right)

One of the SPOD queries identifies complements of fixed verbal expressions. It does so using a simple query which identifies all nodes that have a "svp" dependency with a verbal head. In that query, however, no special consideration was made for cases where that node only contains an index. Therefore, the word group "groot alarm" is not found by the XPath query. The right part of figure 3 illustrates the representation used in AlpinoGraph. As a consequence, the AlpinoGraph variant of the query to identify complements of fixed verbal expressions will identify the "groot alarm" word group as a hit.

Almost all differences between XPath and Cypher are caused by this difference in representation of secondary edges, and in most of these cases, the Cypher version of the query is in fact closer to the linguistic intention of the query - as in our running example. As a side note, going over the differences revealed quite a few manual annotation mistakes too.

5.4 Timing experiment

In addition to a comparison of the results of the various queries, it is also interesting to consider the speed of the various queries for both XPath and Cypher.

As explained in the first paragraph of this section, we compare the cputime requirements for about 80 queries applied to four different treebanks. The results are presented in figure 4. Both axes of the graph are in logarithmic scale. Each dot in the graph represents the cputime it took to finish a particular query for a particular treebank. The Y-axis represents the cputime taken by the XPath queries, whereas the X-axis represents the time taken by the Cypher queries.

As can be observed in the graph, in most cases, but not all, the evaluation of the Cypher queries by AlpinoGraph is much faster than the evaluation of the XPath queries. For the few cases for which the Cypher query is slower, the difference is relatively small.

6 Search optimization

Both Cypher and XPath are expressive enough to define complex syntactic patterns. Some of these patterns occur quite frequently. For example, in the Lassy dependency structures, the topological fields known from Germanic syntax, such as *vorfeld*, *mittelfeld* and *nachfeld* are not explicitly encoded. Yet, it is possible to define Cypher expressions and XPath expressions which recover this information. However, such complicated patterns are relatively hard to compute.

The properties of nodes that we regularly want to refer to can be pre-computed. For instance, a special attribute `_vorfeld` has been added in the representation of treebanks in AlpinoGraph. This attribute is assigned the value "True" for the relevant nodes at the time when the corpus is loaded into AlpinoGraph.

Without such an attribute, it would be possible to identify vorfeld constituents using a Cypher query, but that query is quite complicated, since it must recover the surface syntax of the sentence on the basis of a dependency graph. The actual query identifies potential vorfelds which are (potentially indirect) dependents of the finite verb which precede that finite verb. From those potential vorfelds, the query

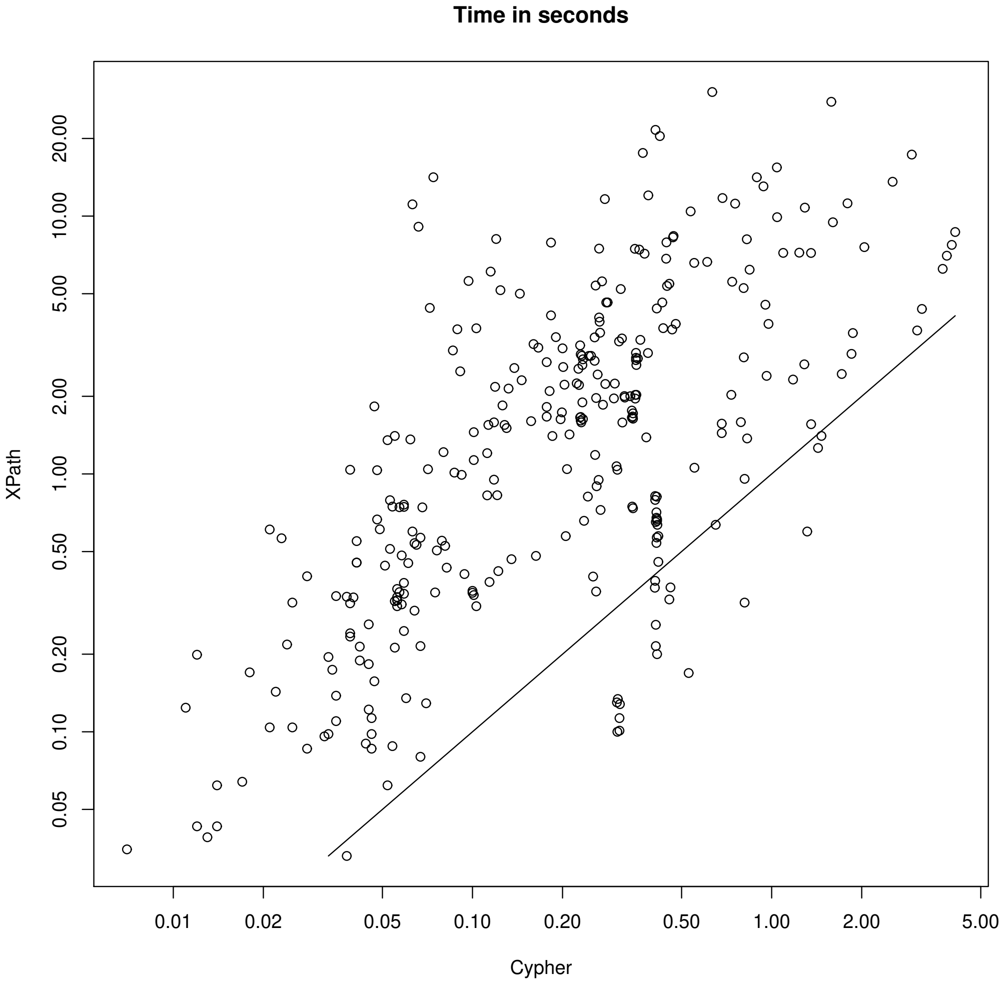


Figure 4: Timing all queries. Note the logarithmic scale. All results above the straight line are cases where the Cypher queries are faster. The few results below the line indicate queries for which the Cypher queries were slower.

then further extracts the maximal one. A further complication is that parts of the vorfeld constituent may actually be extraposed. The full query is given in the appendix.

Running the complicated query over the Lassy Small corpus to identify vorfelds in the corpus takes almost four minutes. After coding the property as an attribute of the relevant nodes, the following, trivial, query finishes within 100 msec:

```
match (n:nw{_vorfeld: true})
return n
```

Properties of nodes that are often used in treebank queries can be encoded by simple attributes. We developed a tool which takes a treebank, a query, an attribute and a value. Each node in the treebank that satisfies the query is augmented with the given attribute and value. This way, treebanks can be enriched with, essentially, redundant information. The benefit will be that queries which rely on that information can be expressed much simpler and will be evaluated much faster.

7 Concluding remarks

In this paper, we introduced AlpinoGraph, a novel graph-based treebank search engine, based on the Cypher query language for graphs. We argued that graphs are an appropriate representation for linguistic annotation, in particular if several annotation layers are combined. We have compared the Cyper queries of AlpinoGraph with the XPath queries that can be used in PaQu, a popular existing treebank search tool for Dutch treebanks. This comparison is based on a large set of relevant syntactic queries, taken from SPOD. Both in XPath and Cypher, it is possible to recover fairly subtle and complicated syntactic patterns. And typically, the Cypher queries are evaluated much faster.

We also described a simple search optimization technique by adding special attributes to nodes which represent properties which are often referred to in queries, but slow to be evaluated on-line. This preprocessing technique is applicable to other treebank search engines too.

AlpinoGraph is open-source and can be used on-line, free of charge. The system is available via `https://urd2.let.rug.nl/kleiweg/alpinograph/`, and the sources are available via `https://github.com/rug-compling/alpinograph`.

Appendix: Query for vorfeld

In order to identify the vorfeld, the following query first identifies the head of main sentences (the finite verb) and then selects embedded dependents for which it is the case that their head precede this finite verb. These potential vorfeld constituents include the actual vorfeld, but also most of the dependents of the vorfeld. Therefore, the query is complicated by removing from the set of potential vorfelds all those nodes that are dominated by a potential vorfeld.

Further complications arise because of the possibility of multi-word-units, and because of the fact that not only real heads (with relation "hd") are treated as heads here, but also dependents of type "crd" and "cmp".

```
select sentid, id
from (
    match (n:node{cat:'smain'}) -[:rel{rel:'hd'}]-> (fin:word)
    match (n) -[:rel*{primary:true}]-> (topic:nw) -[rel:rel*0..1]-> (htopic:nw)
    where (  ( not htopic.lemma is null )
             and htopic.begin < fin.begin
             and ( length(rel) = 0 or rel[0].rel in ['hd','cmp','crd'] )
           ) or
           ( topic.begin < fin.begin and topic.end <= fin.begin )
    return topic.sentid as sentid, topic.id as id, n.id as nid
    except
    match (n:node{cat:'smain'}) -[:rel{rel:'hd'}]-> (fin:word)
    match (n) -[:rel*{primary:true}]-> (topic:nw) -[rel:rel*0..1]-> (htopic:nw)
    where (  ( not htopic.lemma is null )
             and htopic.begin < fin.begin
             and ( length(rel) = 0 or rel[0].rel in ['hd','cmp','crd'] )
           ) or
           ( topic.begin < fin.begin and topic.end <= fin.begin )
    match (topic) <-[:rel*1..]- (nt:node)  <-[:rel*]- (n)
    match (nt) -[relt:rel*0..1]-> (hnt:nw)
    where (  ( not hnt.lemma is null )
             and hnt.begin < fin.begin
             and ( length(relt) = 0 or relt[0].rel in ['hd','cmp','crd'] )
          ) or
          ( nt.begin < fin.begin and nt.end <= fin.begin )
    return topic.sentid as sentid, topic.id as id, n.id as nid
) as foo
```

References

Liesbeth Augustinus, Vincent Vandeghinste, and Frank Van Eynde. 2012. Example-based treebank querying. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC-2012)*, pages 3161–3167, Istanbul, Turkey.

Liesbeth Augustinus, Vincent Vandeghinste, Ineke Schuurman, and Frank Van Eynde. 2017. GrETEL. a tool for example-based treebank mining. In Jan Odijk and Arjan van Hessen, editors, *Clarin in the low countries*. Ubiquity Press, London.

Steven Bird and Catherine Lai. 2010. Querying linguistic trees. *Journal of Logic, Language and Information*, 19:53–73, 06.

Guillaume Bonfante, Bruno Guillaume, and Guy Perrier. 2018. *Application of Graph Rewriting to Natural Language Processing*. Wiley.

Gosse Bouma and Gertjan van Noord. 2017. Increasing return on annotation investment: The automatic construction of a universal dependency treebank for Dutch. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 19–26, Gothenburg, Sweden, May. Association for Computational Linguistics.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. Tiger: Linguistic interpretation of a german corpus. *Journal of Language and Computation*, 2:597–620, 12.

Jiří Mírovský. 2008. Netgraph - making searching in treebanks easy. In *IJCNLP 2008 Proceedings of the Third International Joint Conference on Natural Language Processing*, pages 945–950, Hyderabad, India. International Institute of Information Technology.

Joakim Nivre, Mitchell Abrams, Željko Agić, Lars Ahrenberg, Lene Antonsen, Katya Aplonova, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Victoria Basmov, John Bauer, Sandra Bellato, Kepa Bengoetxea, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Rogier Blokland, Victoria Bobicev, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Savas Cetin, Fabricio Chalub, Jinho Choi, Yongseok Cho, Jayeol Chun, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Arantza Diaz de Ilarraza, Carly Dickerson, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Tomaž Erjavec, Aline Etienne, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Nizar Habash, Jan Hajič, Jan Hajič jr., Linh Hà Mỹ, Na-Rae Han, Kim Harris, Dag Haug, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Jena Hwang, Radu Ion, Elena Irimia, Ọlájídé Ishola, Tomáš Jelínek, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşıkara, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Kamil Kopacewicz, Natalia Kotsyba, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Lucia Lam, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Cheuk Ying Li, Josie Li, Keying Li, KyungTae Lim, Nikola Ljubešić, Olga Loginova, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Gustavo Mendonça, Niko Miekka, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Shinsuke Mori, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, Adedayo Olúòkun, Mai Omura, Petya Osenova, Robert Östling, Lilja Øvrelid, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Siyao Peng, Cenel-Augusto Perez, Guy Perrier, Slav Petrov, Jussi Piitulainen, Emily Pitler, Barbara Plank, Thierry Poibeau, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Siva Reddy, Georg Rehm, Michael Rießler, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Davide Rovati, Valentin Roșca, Olga Rudina, Jack Rueter, Shoval Sadde, Benoît Sagot, Shadi

Saleh, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Baiba Saulīte, Yanin Sawanakunanon, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Muh Shohibussirri, Dmitry Sichinava, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Antonio Stella, Milan Straka, Jana Strnadová, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Dima Taji, Yuta Takahashi, Takaaki Tanaka, Isabelle Tellier, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Lars Wallin, Jing Xian Wang, Jonathan North Washington, Seyi Williams, Mats Wirén, Tsegay Woldemariam, Tak-sum Wong, Chunxiao Yan, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Amir Zeldes, Daniel Zeman, Manying Zhang, and Hanzhi Zhu. 2018. Universal dependencies 2.3. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Jan Odijk, Gertjan van Noord, Peter Kleiweg, and Erik Tjong Kim Sang. 2017. The parse and query (PaQu) application. In Jan Odijk and Arjan van Hessen, editors, *Clarin in the low countries*. Ubiquity Press, London.

Thomas Proisl and Peter Uhrig. 2012. Efficient Dependency Graph Matching with the IMS Open Corpus Workbench. In *Proceedings of LREC*, page 2750–2756, Istanbul. ELRA.

Douglas L. T. Rohde. 2001. Tgrep2 user manual.

I. Schuurman, M. Schouppe, T. Van der Wouden, and H. Hoekstra. 2003. Cgn, an annotated corpus of Spoken Dutch. In A. Abbeilé, S. Hansen-Schirra, and H. Uszkoreit, editors, *Proceedings of 4th International Workshop on Language Resources and Evaluation*, pages 340–347, Budapest.

P. C. uit den Boogaart. 1975. *Woordfrequenties in geschreven en gesproken Nederlands*. Oosthoek, Scheltema & Holkema, Utrecht. Werkgroep Frequentie-onderzoek van het Nederlands.

Leonoor van der Beek, Gosse Bouma, Robert Malouf, and Gertjan van Noord. 2002. The Alpino dependency treebank. In Hendri Hondorp Mariët Theune, Anton Nijholt, editor, *Computational Linguistics in the Netherlands 2001*. Rodopi.

Frank van Eynde. 2005. Part of speech tagging en lemmatizering van het D-COI corpus.

Gertjan van Noord, Gosse Bouma, Frank Van Eynde, Daniël de Kok, Jelmer van der Linde, Ineke Schuurman, Erik Tjong Kim Sang, and Vincent Vandeghinste. 2013. Large scale syntactic annotation of written Dutch: Lassy. In Peter Spyns and Jan Odijk, editors, *Essential Speech and Language Technology for Dutch: Results by the STEVIN programme*, pages 147–164. Springer Berlin Heidelberg, Berlin, Heidelberg.

Gertjan van Noord, Ineke Schuurman, and Gosse Bouma. 2019. Lassy syntactische annotatie.

Gertjan van Noord, Jack Hoeksema, Peter Kleiweg, and Gosse Bouma. 2020. SPOD: Syntactic profiler of Dutch. *Computational Linguistics in the Netherlands Journal*, 10. Accepted.

Gertjan van Noord. 2006. **A**t **L**ast **P**arsing **I**s **N**ow **O**perational. In *TALN 2006 Verbum Ex Machina, Actes De La 13e Conference sur Le Traitement Automatique des Langues naturelles*, pages 20–42, Leuven.

Gertjan van Noord. 2009. Huge parsed corpora in Lassy. In Frank van Eynde, Anette Frank, Koenraad De Smedt, and Gertjan van Noord, editors, *Proceedings of the Seventh International Workshop on Treebanks and Linguistic Theories (TLT 7)*, number 12 in LOT Occasional Series, pages 115–126, Utrecht, The Netherlands. Netherlands Graduate School of Linguistics.

Implementing an End-to-End Treebank-Informed Pipeline for Bulgarian

Alexander Popov
AIaLT
IICT-BAS, Bulgaria

Petya Osenova
AIaLT
IICT-BAS, Bulgaria

Kiril Simov
AIaLT
IICT-BAS, Bulgaria

{alex.popov|petya|kivs}@bultreebank.org

Abstract

The paper reports on the implementation of an NLP pipeline for Bulgarian, developed within the spaCy framework and based on BulTreeBank as main source for training and test data. This new end-to-end pipeline aims to ensure easier technical maintenance and synchronization between modules, superior processing speeds – for use in real applications, and greater flexibility of adaptation. We discuss the challenges encountered in the implementation process and the solutions adopted, including the architecture itself, as well as its quantitative and qualitative evaluation.

1 Introduction

Implementing a pipeline solution for processing a specific language is a task characterised by non-trivial challenges. Different implementation frameworks necessitate decisions about the overall architecture, which in turn constrain the possible solutions to concrete tasks.

A previous end-to-end pipeline for Bulgarian has been implemented in Java and in the XML-based CLaRK system[1]. It includes the following modules: tokenizer, sentence boundary detection, POS tagger, lemmatizer, and a transition-based dependency parser. The tokenization, sentence segmentation and lemmatization modules are implemented as rules in the CLaRK system, while POS tagging and dependency parsing are carried out by statistical models trained on annotated data. The Mate Tools[2] software has been used for training the models, where the training data is an older version of the BTB treebank, which includes fewer sentences and differs in annotation types from the Universal Dependencies (UD) version. Later versions of this pipeline include additional processing modules (e.g. word sense disambiguation), which however are not trained on annotated data and are implemented in other frameworks (Simov et al., 2016).

Currently there is no freely-available end-to-end processing pipeline for Bulgarian that meets the following criteria:

- is implemented within a single framework;
- includes semantic analysis capabilities (word sense disambiguation, named entity recognition);
- achieves competitive accuracy scores;
- affords processing speeds suitable for real applications;
- can handle big volumes of data.

This short paper reports on the implementation process of such a pipeline. The creation of a language processor for Bulgarian is inevitably related to the Bulgarian treebank — BulTreeBank (BTB), a version of which is freely available through the Universal Dependencies initiative. Since the treebank has been annotated with named entities, POS labels and features, as well as converted to syntactic dependencies, it is used as the main training data for the pipeline modules. We have also obtained the current version of the BTB-WordNet, which has made it possible to add a module for semantic analysis as well.

The accuracy scores reported in the paper are generally lower than those achieved with the older pipeline. However, the training and evaluation data has evolved significantly over time, making a

[1] http://bultreebank.org/en/clark/bulgarian-nlp-pipeline-in-clark-system/

[2] https://www.ims.uni-stuttgart.de/en/research/resources/tools/matetools/

fair comparison difficult. The new pipeline modules have been trained on the current version of UD-BulTreeBank that – compared to previous versions – consists of more sentences and more varied syntactic structures. Thus, accuracy has seemingly dropped, but the pipeline can actually handle more challenging syntax. In addition to that, the current system, which is a work in progress, has the advantages that it is much more robust, faster, intuitive, and extensible, making it suitable for practical applications.

2 Implementation of the pipeline

For the implementation of the Bulgarian processing pipeline, the spaCy framework for NLP was chosen – thereby allowing us to implement the pipeline entirely in Python code. spaCy has been developed to be suitable for industrial-strength solutions, which means that it is fast, well-structured, flexible, and easy to use. Comparison with other popular frameworks shows that it works at significantly greater speeds.[3] It also offers a variety of NLP tools that can be easily added to or removed from the pipeline: from tokenization and sentence splitting, to dependency parsing and entity linking. spaCy models trained on data for English and other languages consistently achieve accuracy scores that are close to the state of the art. One potential disadvantage of spaCy is that its neural architectures are fixed in advance and there is not much freedom in configuring different models. However, due to spaCy's modularity and non-destructive tokenization (i.e. the original input can always be recovered from the processed output), it can be combined relatively easily with standard deep learning frameworks.

Adapting spaCy to a new language includes two major steps: 1) adding language-specific lists and rules for tokenization and lemmatization;[4] 2) training statistical models on language-specific data. While model training on available data is largely a seamless process, the first step depends more heavily on the specificities of the particular language.

2.1 Rule-based and dictionary-based language-specific processing

2.1.1 Tokenization

Tokenization is carried out via rules and language-specific exceptions. This amounts to compiling:

- a list of strings, each one of which is to be analyzed a single token;
- attributes associated with abbreviated tokens: lemmas, as well as morphological analyses;
- regular expressions for handling tokens with special symbols, like hyphens, apostrophes, etc.;
- regular expressions for handling punctuation marks that should not split strings into tokens (e.g. if we want to analyze dates in the [DD.MM.YYYY] format into single tokens).

A list of tokens, including additional information about lemmas and morphology, was compiled on the basis of a focused lexicon; regular expression cascades were iteratively devised during development.

2.1.2 Lemmatization

Bulgarian is a relatively rich language in terms of its morphology – the unique fine-grained morphological tags in the Bulgarian treebank number 578.[5] This means that a high ambiguity of word forms can be expected, and simple part-of-speech tags (e.g. the Universal POS tagset (Petrov et al., 2011)) will not always be enough to disambiguate the correct lemma per word form. For instance, the same surface verb forms may have different underlying lemmas depending on how they are used. Consider the example in Table 1. The simple POS tag is the same for both cases and would therefore be insufficient for correct disambiguation. Knowing that one word form belongs to an impersonal verb, and the other does not, however, would be sufficient information – which is indeed provided by the full morphological tag.

Thus, a language-specific morphological dictionary was constructed, on the basis of an existing grammatical resource, and added to the lookup tables that the spaCy language model accesses for lemmatization. The Bulgarian lemmatizer takes a token string (i.e. word form) and attempts to pair it with the predicted morphological tag as a joint dictionary key, so that it can extract the correct lemma from the dictionary. In order to ameliorate errors from the POS tagger, a fallback option is included in the logic of

[3] `https://spacy.io/usage/facts-figures`

[4] While it is true that data-driven approaches can also lead to excellent results (Kondratyuk et al., 2018), we have chosen to follow the spaCy schema for developing language models.

[5] http://bultreebank.org/wp-content/uploads/2017/04/BTB-TR03.pdf

word form	Gloss	POS	Morpho-tag	Lemma
вървя	“luck was on somebody’s side”	VERB	Vniif-o3s	върви-ми
вървя	“he/she/it walked”	VERB	Vpiif-o3s	вървя

Table 1: Comparison of verb forms.

the lemmatizer: in case no matches for the word form/morpho-tag pair are found, the simple POS tag is used to obtain possible (the default logic in spaCy); the word form/POS mappings are bundled in a separate lookup table. Currently the lemmatizer selects the first candidate in the list of lemmas matching the word form/POS pair, which would produce errors in some cases (due to the aforementioned ambiguity); one planned improvement is to order the list of lemma candidates using frequency distributions. In the cases when neither the morphological tag nor the POS tag yield an associated lemma, a simple lookup from word form to lemma is used. If that strategy is also unsuccessful, a lowercased version of the string is returned.

2.2 Statistical models for NLP

2.2.1 POS Tagging and Dependency Parsing

Training models on annotated data with spaCy’s neural architectures is a straightforward process. In our case, we have used the Universal Dependencies files for training the POS tagger, dependency parser (including labeled attachment), and sentence splitter, which depends on the parser module to correctly segment sentences.[6] We present the accuracy results on the development and test sets in the next section.

2.2.2 Named entity recognition

BulTreeBank has been annotated with information about Person, Location and Organization entities, but since NER data is not included in the UD version, we trained the NER module separately, after the POS tagger and parser. We also included data from the BSNLP corpus (Marinova et al., 2020), which has been originally compiled for a special task on NER and includes Event, Product and Other types, in addition to Person, Location and Organization. The two corpora were processed into the spaCy-readable IOB format, concatenated and shuffled, to balance them between the training (20803 sentences) and development (2312 sentences) portions. We provide evaluation metrics for NER in the next section.

2.2.3 Word sense disambiguation

The final module included in the pipeline carries out word sense disambiguation (WSD). The spaCy framework does not provide off-the-shelf WSD functionality, therefore a different solution had to be adapted. The EWISER system (Bevilacqua and Navigli, 2020) was chosen, due to several considerations: superior accuracy compared to other systems, easy integration with spaCy, and multilingual support.

EWISER improves the state-of-the-art in WSD on the popular evaluation framework for English (Raganato et al., 2017), “breaking through the 80% glass ceiling”. It accomplishes that via a novel approach to representing word senses in relation to the other senses to which they are related through the WordNet semantic network. EWISER is a neural network classifier that uses the powerful BERT model (Devlin et al., 2018) for contexualized embeddings to construct the context representation for each input word. It also uses a synset embedding matrix to transform the final hidden layer of the network into a vocabulary-sized vector – thus, instead of randomly initializing the final linear transformation matrix, structured knowledge about senses is added into the calculation of the logits. The logits are additionally “structured” by infusing information about the relations between all synsets in the vocabulary. Apart from achieving the highest reported results on the standard English datasets, EWISER also fares well when tested against data for other languages. Even more impressive is the fact that it achieves state-of-the-art results on German, French, Italian and Spanish data when trained on the SemCor corpus for English and using the multilingual BERT model. The EWISER implementation[7] provides a special class for initial-

[6] https://universaldependencies.org/treebanks/bg_btb/

[7] https://github.com/SapienzaNLP/ewiser

izing the system as a spaCy module, which can then be directly added to a pipeline; it also provides the relevant models for processing multilingual data.

Currently, an enriched version (around 25,000 synsets) of the Bulgarian BTB-WordNet (Osenova and Simov, 2018) is being prepared for official release, which involves consolidating the indices and ensuring their correct mapping to the Princeton WordNet (PWN), version 3.1, as well as the actualization of the BulTreeBank sense-annotated data with the updated IDs. When this work is done, it will be possible to properly train and evaluate an EWISER model on Bulgarian data. As of now, we have only been able to adapt the multilingual model for processing Bulgarian text, and thus provide only a qualitative analysis in the next section. The adaptation itself comprises of a chain of transformations:

- preparing a dictionary that maps lemmas to possible synsets;
- replacing the Bulgarian synset IDs with IDs from PWN;
- mapping the PWN IDs from version 3.1 (used in BTB) to version 3.0 (used in EWISER);
- mapping PWN IDs to BabelNet IDs, in order to produce the final dictionary required by EWISER;
- compiling a list of lemmas that the system can recognize (i.e. they are present in the dictionary).

After that, the spaCy EWISER module can be run on Bulgarian text, with the sense annotations available from the **token._** attribute. In addition to training a model on Bulgarian data, we plan to improve this part of the pipeline by experimenting with custom-made synset embeddings.

3 Evaluation

POS tagging and dependency parsing The POS tagger achieves accuracy of 94.13 % on the development set and 94.49 % on the test set. The metrics for the parser are as follows: 83.03 % for LAS and 88.95 % for UAS on the development data, and 83.95 % LAS / 89.71 % UAS on the test data. In the multilingual setting of the UD parsing shared task, the best model achieved LAS 91.22 % on the Bulgarian treebank (Zeman et al., 2018). However, there is evidence that UD-trained models perform better in multilingual and cross-lingual settings, compared to monolingual ones (Smith et al., 2018). The average reported accuracy across the models in the shared task aligns with the results here.

The detailed results for the UD relations are given in Tables 2, 3 and 4. Only three relations have not been evaluated: **dep**, **vocative** and **appos**. The nominal subject (**nsubj**) and direct object (**obj**) relations get the best results in table 2, as can be expected. On the other hand, clausal subjects (**csubj**) and clausal complements, namely the infinitive-like type (**xcomp**), are hard to detect. The most difficult case is the passive sentential subject (**csubj:pass**). This is also expected, because it is not marked in any special way and depends on the passive word form of the verb.

UD relation	**P**	**R**	**F**	**UD relation**	**P**	**R**	**F**
obj	80.75	76.86	78.75	iobj	61.89	60.65	61.26
csubj	57.57	45.23	50.66	nsubj	81.07	77.61	79.30
ccomp	79.43	57.04	66.40	xcomp	40.0	80.95	53.54
nsubj:pass	60.71	75.55	67.32	csubj:pass	40.0	25.0	30.76

Table 2: Evaluation of the UD core relations.

Table 3 shows that the parser achieves its highest scores in parsing relations that have fixed word order positions in Bulgarian phrases. These are prepositional phrases (**case**), determiners (**det**), adjectival modifiers (**amod**), numeral modifiers (**nummod**), expletives (**expl**). In contrast, relations that are expressible in a variety of ways are scored lower. These are: the oblique relation (**obl**), secondary predication (**acl**) and adverbial clauses (**advcl**), among others.

The highest-scored relation in table 4 is the one which includes the root node, followed by **cc** and **fixed**. Verbless sentences are clearly more challenging, the most difficult relation being that of **parataxis**.

Named entity recognition The combined (cross-category) results for NER are as follows: precision – 92.75 %; recall – 93.31 %; F-measure – 93.03 %. The detailed results from the NER module, the first to be reported on this combination of data, are presented in table 5. The model produces the most

UD relation	P	R	F	UD relation	P	R	F
mark	89.05	89.5	89.27	expl	96.01	92.8	94.37
advcl	69.09	65.8	67.45	discourse	73.68	63.63	68.29
cop	68.18	83.33	75.0	case	96.01	96.57	96.29
aux	94.18	83.21	88.36	det	94.51	94.51	94.51
nmod	79.36	79.52	79.44	amod	93.24	93.54	93.39
advmod	81.53	81.76	81.65	obl	65.29	57.48	61.14
nummod	91.42	91.42	91.42	aux:pass	43.51	91.93	59.06
acl	60.0	53.33	56.47	acl:relcl	73.10	76.31	74.67

Table 3: Evaluation of the non-core UD relations.

UD relation	P	R	F	UD relation	P	R	F
root	89.97	90.13	90.05	cc	89.50	89.15	89.32
conj	66.93	70.65	68.74	fixed	92.68	79.16	85.39
flat	65.07	90.44	75.69	parataxis	9.09	54.54	15.5

Table 4: Evaluation of the other UD relations.

accurate annotations for the event (EVT) class. This is due to the fact that the additional training data (BSNLP) focuses mainly on a single event – Brexit. The identification of locations (LOC) also achieves good results, while those for person (PER) and organization (ORG) are slightly lower – mostly due to the frequent occurrence of regular polysemy and the partial identification of chunks; the model also seems to have a bias toward words with capital letters at the beginning of sentences; it is worth noting that the PER category includes etnonyms in addition to names. The recognition of products (PRO) fares well, while OTH (other), being a catch-all category, has significantly lower recall, and thus – lower F-measure.

Word sense disambiguation The WSD module analysed only content words that can be linked to possible meanings from the Princeton WordNet. It relied on the lemmatizer and POS tagger in order to retrieve the relevant word senses from the dictionary. Here is an example:

Това е първата ми реакция.
(This is my first reaction)

The numeral and the noun are annotated with the following lemma/POS/sense information:

- първата първи ADJ wn:01010862a: preceding all others in time or space or degree
- реакция реакция NOUN wn:00863513n: an automatic instinctive unlearned reaction to a stimulus

For the moment we do not have any automatic evaluation of the WSD module, and only qualitative observations can be made regarding the output of the model. Nouns and verbs in one hundred sentences were checked for the assigned senses. One observable problem are lexemes which have not been mapped to senses – because they are missing in the dictionary, or because of incorrect lemma/POS annotations. Another problematic issue are multi-word expressions — which need to be processed as single tokens by the pipeline in order to be correctly disambiguated. For example, in the MWE “изпускам си нервите” (leave-I REFL nerves) (‘lose one’s temper’), the system identified the lemma ’nerve’, but not the verb ‘leave’. In some cases there is a bias towards one sense of the lexeme, while the rest are ignored. For example, in one and the same sentence the lexeme “скелет” (skeleton) is used in two different senses: “the hard structure (bones and cartilages) that provides a frame for the body of an animal”, and “something reduced to its minimal form”, but the system assigns the first sense in both cases. This might be due to a skewed distribution of samples in the training data (in this case — SemCor).

4 Conclusion

The paper reports on the initial stage in the implementation of an end-to-end pipeline for Bulgarian, using the spaCy framework. The pipeline comprises of rule-based (tokenizer, lemmatizer) and statistical

NER lables	P	R	F	NER lables	P	R	F
PER	91.57	91.47	91.52	EVT	98.73	97.90	98.31
ORG	90.91	93.41	92.14	LOC	95.25	96.87	96.05
PRO	89.36	89.36	89.36	OTH	76.36	56.0	64.61

Table 5: Evaluation of the NER categories.

(POS tagging, NER, UD parsing, and WSD) components, working together in a shared setting. The initial evaluation results provide a promising baseline for future improvements, which would focus on: better tokenization through more precise rules and syntactic parses; better mapping between POS tags and potential lemma candidates; adapting the BTB-Wordnet data for training WSD models; adding more gold data for the training of the NER module and the UD parser.

Acknowledgements

This work was partially supported by *the Bulgarian National Interdisciplinary Research e-Infrastructure for Resources and Technologies in favor of the Bulgarian Language and Cultural Heritage, part of the EU infrastructures CLARIN and DARIAH – CLaDA-BG*, Grant number DO01-272/16.12.2019. Alexander Popov's contribution has been supported by the Bulgarian Ministry of Education and Science under *the National Research Programme "Young scientists and postdoctoral students"* approved by DCM # 577 / 17.08.2018.

References

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Daniel Kondratyuk, Tomáš Gavenčiak, Milan Straka, and Jan Hajič. 2018. Lemmatag: Jointly tagging and lemmatizing for morphologically-rich languages with brnns. *arXiv preprint arXiv:1808.03703*.

Iva Marinova, Laska Laskova, Petya Osenova, Kiril Simov, and Alexander Popov. 2020. Reconstructing ner corpora: a case study on bulgarian. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4647–4652.

Petya Osenova and Kiril Simov. 2018. The data-driven bulgarian wordnet: Btbwn. *Cognitive Studies*, 18.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2011. A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110.

Kiril Simov, Petya Osenova, and Alexander Popov. 2016. Towards semantic-based hybrid machine translation between bulgarian and english. In *Proceedings of the 2nd Workshop on Semantics-Driven Machine Translation (SedMT 2016)*, pages 22–26.

Aaron Smith, Bernd Bohnet, Miryam de Lhoneux, Joakim Nivre, Yan Shao, and Sara Stymne. 2018. 82 treebanks, 34 models: Universal Dependency parsing with multi-treebank models. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 113–123, Brussels, Belgium, October. Association for Computational Linguistics.

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21, Brussels, Belgium, October. Association for Computational Linguistics.