The revision of linguistic annotation in the Universal Dependencies framework: a look at the annotators' behavior

Magali S. Duran and Lucelene Lopes and Thiago A. S. Pardo
Núcleo Interinstitucional de Linguística Computacional, Universidade de São Paulo – Brazil
magali.duran@gmail.com lucelene@gmail.com taspardo@icmc.usp.br

Abstract

This paper presents strategies to revise an automatically annotated corpus according to the Universal Dependencies framework and discusses the learned lessons, mainly regarding the annotators' behavior. The revision strategies are not relying on examples from any specific language and, because they are language-independent, can be adopted in any language and corpus annotation initiative.

1 Introduction

The construction of annotated datasets is a challenging task, especially for low-resource languages. In order to take advantage of the experience of high-resource languages, projects in other languages have adopted successful annotation models, "skipping" the steps of instantiating a theory (i.e., the linguistic model to be used) and creating tag sets, which are steps discussed by Hovy and Lavid, 2010 and Pustejovsky et al., 2017. Reutilizing annotation models is important, but is also key to have information on how to design an annotation task. It has become clear to the scientific community that sharing the know-how to building annotated corpora can encourage other research groups to undertake their own annotation projects. For this reason, over the last two decades, discussion on the corpus annotation process has been gaining prominence in the Natural Language Processing (NLP) scene.

Seminal works laid the foundations of "annotation science" (Ide, 2007; Hovy and Lavid, 2010; Ide and Pustejovsky, 2017). The availability of new technologies has brought new possibilities, such as crowdsourcing the annotation (Snow et al., 2008; Hovy et al., 2013) and using LLMs as annotators (Pavlovic and Poesio, 2024; Weissweiler et al., 2023; Torrent et al., 2024). In addition, annotation has expanded its purposes, as shown by the case of perspectivism (Leonardelli et al., 2023; Akhtar et al., 2021), which takes into account annotation disagreements. However, perspectivism hardly applies to the traditional prescriptive paradigm, which is the case of the annotation discussed here (see Röttger et al., 2022 for a comparison between prescriptive and descriptive annotation paradigms).

Depending on the annotation model, different annotation formats and standards are adopted. For the Universal Dependencies (UD) framework (de Marneffe et al., 2021) – the focus of this paper – the CoNLL-U format is the standard. This format is an evolution of CoNLL-X (Buchholz and Marsi, 2006) and was developed to annotate datasets used in the shared tasks of 2017 and 2018 (Hajič and Zeman, 2017; and Zeman et al., 2018).

To get an idea of the scope of the UD, its current version (May, 2025) has 319 treebanks and 179 languages, representing a valuable resource for training multilingual models and developing cross-language studies. Thanks to this resource, several multilingual parsers have been trained, such as UDPipe 2 (Straka, 2018), UDify (Kondratyuk and Straka, 2019) and Stanza (Qi et al., 2020), which makes it possible to start a new annotation project by automatically pre-annotating the corpus and posteriorly manually revising it, which is another well established annotation method.

The revision of a pre-annotated corpus is significantly different from annotating from scratch. Correcting an entire corpus in order to improve the performance in some NLP task is a big challenge. It is not evident which sentences contain errors or how many errors there are. In particular, when the tool used for pre-annotation already has good accuracy, the annotators need to be very good judges in order to analyze the sentences, identify errors and propose corrections. In the particular case of CoNLL-U, annotators have to deal with dozens of labels and a multilayered annotation.

Drawing on five years of experience with annotation, this paper presents adopted (language

agnostic) annotation strategies and discusses the lessons learned – mainly those regarding annotator behavior – for a corpus of news texts in Portuguese, following the UD framework. We believe that the fundamental lessons can provide insights for similar projects in other languages, and, for this reason, we have purposely not presented any examples in Portuguese, and, where we considered important to provide an example, we have given it in English to increase its usefulness.

Basically, we decided to adopt a "divide-and-conquer" strategy, which consisted of revising linguistic layers (in some of the 10 CoNLL-U columns) separately and sequentially, as the information of one layer benefits from the corrections made in the others. This strategy allowed us to learn during the process and inspired us to develop resources to improve consistency, a fundamental requirement for building a gold standard corpus.

This paper is organized as follows. In Section 2, we comment on our project and on the reasons that led us to choose the UD annotation. Section 3 presents our approach to annotation revision and the strategies developed to iteratively combine the best of human annotation skills with the best of computational power, doing our best to ensure consistency and to save time. Section 4 comments on related work, and Section 5 draws some conclusions and presents insights for future work.

2 The Porttinari Project

The aim of the Porttinari (Pardo et al., 2021) project is to annotate corpora from different genres according to UD, with a view to train robust and multigenre parsers in Portuguese that benefit downstream applications.

The idea of choosing language-dependent theories, instantiating them, and creating our own annotation model was soon discarded, as this would limit the future use of our parsers in multilingual tasks. The reasons that led us to choose the UD "universal" annotation model were:

- it is a model that has come a long way in refining tag sets applicable to different languages;
- 179 languages have already been annotated with UD tag sets (UD v2.16, May, 2025);
- the maintainers are speakers of different languages, constituting a multilingual initiative;
- the community is active and open to discussion, taking into account problems from different language families;
- the set of annotated corpora has already proven results both in multilingual applications and in typological studies;
- although the tag sets of Universal Part-of-Speech tags (UPOS, hereafter) and dependency relations (DEPRELs, hereafter) are fixed and do not allow changes, the CoNLL-U model reserves a column for annotating language-specific Part-of-Speech tags and allows DEPRELs to have subtypes, which gives some flexibility for language-specific phenomena to be covered (the CoNLL-U format is described in Table 1 and exemplified in Table 2);

In what follows, we describe and comment on the main steps of the annotation effort carried out on our initial corpus, called Porttinari-base, composed of news texts, containing 168,080 tokens and 8,418 sentences.

2.1 Tokenization and sentence segmentation

It is important to note that the minimum scope of UD annotation is the token (which almost always coincides with the concept of a word) and the maximum scope is the sentence. Therefore, the segmentation into sentences and tokenization processes need to be carried out carefully so that the CoNLL-U files are well formed. Errors on these levels may result in structural changes to the CoNLL-U files and affect the entire annotation.

2.2 Selection of parser and annotation tool

We opted for UDPipe 2 (Straka, 2018) to pre-annotate our data because it was already widely adopted in the international research community, reaching state-of-the-art results. We also previously evaluated annotation tools and chose Arborator-Grew (Guibon et al., 2020) because it has a very user-friendly graphic interface and allows several annotators to work at the same time, both in blind and visible modes. Moreover, in Arborator-Grew we can choose which layers to exhibit. Fig. 1 shows the graphic interface used for human revisions, with all layers exhibited.

2.3 Drawing up guidelines in Portuguese

When we started our annotation project following the UD model, there were already annotated UD

1	2	3	4	5	6	7	8	9	10
ID	**FORM**	**LEMMA**	**UPOS**	**XPOS**	**FEATS**	**HEAD**	**DEPREL**	**DEPS**	**MISC**
Token identifier (numeric)	Token form (word or symbol)	Lemma of the token form	PoS tag in the UD tag set	Optional extended (language-specific) PoS tag	List of morphological features associated to the token	ID of the token's head for the dependency tree	Dependency relation tag of the token towards the token's head	HEAD-DEPREL pairs for the enhanced dependency graph	Any additional annotation

Table 1: CoNLL-U 10-columns format to each token of a sentence (official UD abbreviation and content description).

ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	DEPS	MISC
1-2	I'd	_	_	_	_	_	_	_	_
1	I	I	PRON	_	Case=Nom|Number=Sing|Person=1|PronType=Prs	3	nsubj	_	_
2	would	would	AUX	_	VerbForm=Fin	3	aux	_	_
3	love	love	VERB	_	VerbForm=Inf	0	root	_	_
4	to	to	PART	_	_	5	mark	_	_
5	set	set	VERB	_	VerbForm=Inf	3	xcomp	_	_
6	them	they	PRON	_	Case=Acc|Number=Plur|Person=3|PronType=Prs	5	obj	_	_
7	free	free	ADJ	_	Degree=Pos	5	xcomp	_	SpaceAfter=No
8	.	.	PUNCT	_	_	3	punct	_	_

Table 2: Example of CoNLL-U annotation for the sentence "I'd love to set them free.".

corpora in Portuguese, but they had only used the generic UD guidelines. As Röttger et al. (2022) argue, annotation for training models needs to be prescriptive and accompanied by very clear guidelines, so that annotators can consult them during the annotation process, improving the annotation consistency. For this reason, our first step was to produce two manuals explaining and exemplifying, in Portuguese, the use of the two UD tag sets: UPOS and DEPREL, bridging the gap between general UD guidelines and observable phenomena in Portuguese (Duran, 2021; and Duran, 2022). The first versions of both manuals were enriched throughout the process, adding examples of not-so-frequent constructions found in the corpus (currently the UPOS manual has 55 pages, and the DEPREL manual has 166 pages with 308 annotated examples).

3 The annotation strategy: divide-and-conquer

Differentiating among 17 UPOS and 37 DEPREL labels is a complex task, even for experienced linguists. For this reason, we divided the revision task into four steps, based on CoNLL-U columns:

- Step 1 - column 4: UPOS;
- Step 2 - column 3: LEMMA;
- Step 3 - column 6: FEATS;
- Step 4 - columns 7 and 8: HEAD/DEPREL.

This revision strategy was adopted with the belief that it would create a cascade effect, yielding the following outcomes:

- gradual accumulation of expertise in the tasks;
- the mitigation of error propagation across annotation layers, as errors corrected in initial columns reduce the likelihood of inconsistencies in later ones;
- the ability to select and train annotators for the tasks, starting with those deemed simpler;
- the opportunity to retrain the parser at the conclusion of each step and to apply it to the portion of the corpus yet to be revised.

Although we did not anticipate a cyclical nature, any decision that affected the entire corpus was followed by a punctual revision of the already annotated sentences, in order to maintain consistency.

The remaining columns of CoNLL-U were not revised: columns 1, 2, and 10 (ID, FORM, and MISC) were only changed when we corrected segmentation and tokenization problems; column 5 (XPOS) was left blank because we had no need to use another PoS tag set; column 9 (DEPS) was left blank because multilingual parsers were not (and are not at the time of writing this paper) prepared to simultaneously annotate enhanced dependencies. In the following, we comment on lessons learned during each of the four revision steps.

3.1 STEP 1 - Revising UPOS

We started with UPOS because it constitutes the smallest and simplest set of UD labels with great equivalence to the set of labels of the Brazilian grammatical nomenclature. Furthermore, this

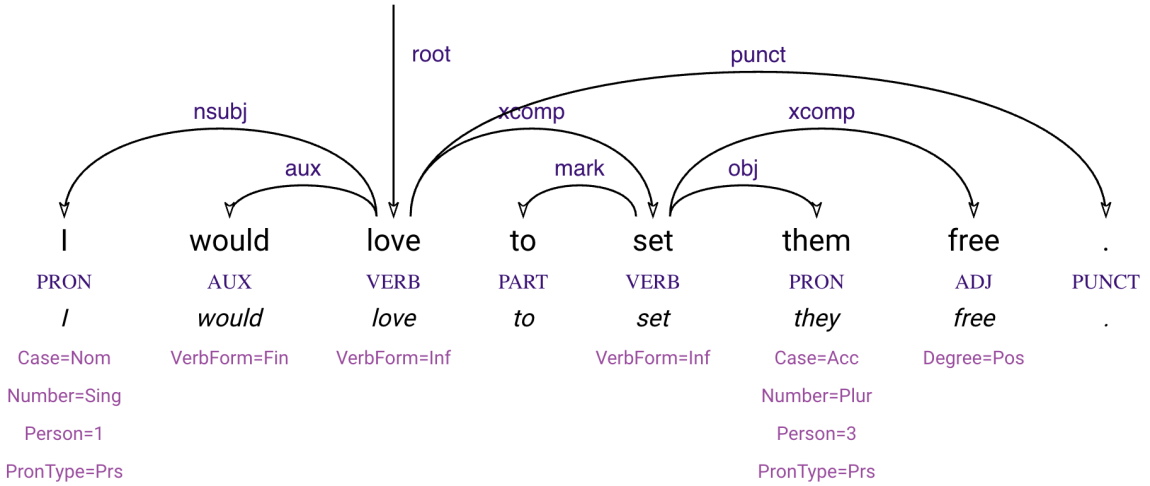


Figure 1: Example of the tree representation of a sentence – codified in CoNLL-U – using Arborator-Grew.

nomenclature is a background that annotators already had and which could facilitate their training. Additionally, from the UPOS, we can restrict the FEATS and DEPREL accepted, making the next steps easier.

The task of UPOS revision proved to be more laborious than we first imagined. As the parser we used had a good performance[1], finding errors required an "eagle eye" and the ability to stay focused. Not all annotators had this ability and this step helped us to identify annotators with best performance in revision tasks, whom we invited to the next steps.

The task involves two sub-tasks: identifying the error and suggesting the correct UPOS label. In each package, all disagreement cases were analyzed by an experienced linguist who made the adjudication and used what she learned during this experience to give feedback to the annotators. The assessment of the annotators' work, therefore, was based on the adjucator's analysis of the disagreements. This does not guarantee that all errors in the corpus have been corrected. In fact, the maintenance of the corpus always brings some corrections to errors identified after the first annotation has been completed.

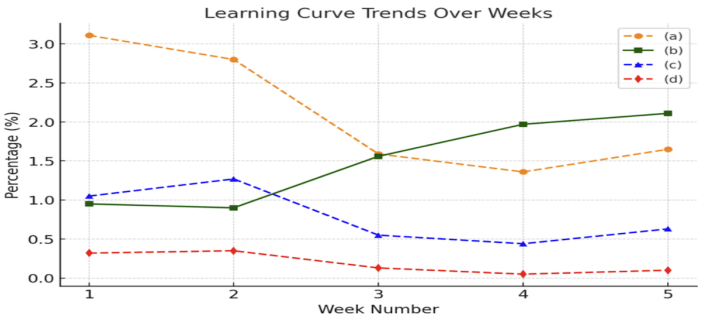
In some cases, annotators overlooked errors and made no changes (a). When corrections were made, three scenarios emerged: the error was correctly identified and appropriately corrected (b); the error was detected, but an incorrect correction was applied (c); or, more rarely, a non-existent error was mistakenly introduced (d). Fig. 2 shows the results of UPOS correction for the first 2,177 sentences from a total of 8,418 sentences in the corpus and the learning curve during this initial phase. It is very interesting to note that:

- the proportion of tokens that needed correction but were missed by annotators decreases as the annotation process runs (probably due to acquired annotation experience);
- the proportion of tokens that should be and were corrected increased (same reason above);
- in the last week, there are still 2.38% of tokens that showed annotation problems (cases (a), (c) and (d)), but this value is almost half of what occurred in the first week (4.48%).

In the first four weeks, the sentences were shorter (around 14 tokens per sentence) than in the last week (29 tokens per sentence). Following this revision, these sentences were used to retrain the parser, and the remaining sentences were re-annotated and manually revised until all UPOS were corrected.

We selected ten annotators for this step (undergraduate linguistics students) because we wanted to speed up the task without overburdening the annotators. That expectation, however, did not materialize. There were many disagreements, both in the errors detected and in the proposed corrections, which required a lot of adjudication. As the errors detected were distributed among the sentences, in the first weeks almost 50% of the sentences needed adjudication. However, these disagreements in errors detected and corrected do not stand out when we used Kappa (Carletta, 1996), as the unchanged PoS

[1]UPOS: 92%, LEMMA: 90%, FEATS: 76%, UAS (correct HEAD): 88%; LAS (correct HEAD and DEPREL): 87%.



week number	total number of sentences	total number of tokens	(a) tokens needing correction that were missed		(b) tokens that should and were corrected		(c) tokens that should be corrected, but were changed to an incorrect tag		(d) tokens already correct, but changed (error insertion)	
1	481	6,857	213	3.11%	65	0.95%	72	1.05%	22	0.32%
2	492	6,919	194	2.80%	62	0.90%	88	1.27%	24	0.35%
3	482	6,787	108	1.59%	106	1.56%	37	0.55%	9	0.13%
4	480	6,553	89	1.36%	129	1.95%	29	0.44%	3	0.05%
5	242	7,104	117	1.65%	150	2.11%	45	0.63%	7	0.10%
total	2,177	34,220	721	2.11%	512	1.50%	271	0.79%	64	0.19%

Figure 2: Manual revision outcomes for the first five weeks of UPOS revision.

tags (more than 90%) counted as agreements (and they really should be counted, because, although it may not seem obvious, all the tokens were actually revised, even those left unchanged). During the analysis of disagreements, we learned that the majority was not always right, which means that a majority voting strategy would not be a good solution to substitute adjudication.

Dealing with remote annotators was underestimated (in 2021 we were in isolation due to Covid-19). We even implemented a log in the annotation tool to study the behavior of annotators who missed many errors. This was important to identify undesirable behaviors, such as annotators who checked sentences a few seconds after opening them for annotation, without enough time to at least read them. Then we realized an important feature of the revision task: as there is no blank space to fill in, it is difficult to distinguish an annotator who has agreed with the automatic annotation from an annotator who has barely read the sentence.

3.1.1 Splitting the workload into packages

We made packages of 20 sentences, starting with the smallest sentences in the corpus, and when we learned something recurrent, we systematized the automatic revision of what had already been annotated, ensuring homogeneity. Every 200 sentences, we automated the correction of recurring errors in the next packages. Every 2,000 sentences, we retrained the parser, so that the number of errors in the packages to be revised gradually decreased.

In the final count, 168.080 UPOS (one per token) were human revised, of which 6,437 (3.83%) were manually corrected. In addition to correcting the errors, the most important thing is that we confirmed the accuracy of the unedited UPOS, which led us to obtain a corpus with 100% of the revised UPOS, as far as we could tell, correct.

3.1.2 New lexical resources

Within this step, we developed lists of non-ambiguous single tokens and non-ambiguous co-occurring tokens (regardless of whether they constitute multiword expressions or not) and used them to automatically annotate the respective UPOS (Lopes et al., 2021).

These lists mainly contain function words (conjunctions, adpositions, determiners, etc.) and crystallized constructions.

3.2 STEP 2 - Revising LEMMA

Our initial plan was to make a fully automatic revision of the lemmas, using a lexicon. We thought that, by providing the token form and its UPOS as input, we would obtain a unique possible lemma, so that only out-of-vocabulary tokens would require human revision. This is true in most cases, but we found exceptions: in Portuguese, there are identical forms of nouns and verbs, with the same UPOS (NOUN or VERB), with different lemmas. For example, “fui”, “foi”, “fomos”, “foram” are

verbal forms of both verbs “ir” (to go) and “ser” (to be), both in the present tense, requiring humans in the loop to “disambiguate” the lemma in context.

We employed a single annotator (with lexicographical expertise) for the whole task: revision of the lemmas of 1,825 tokens (out of the 168,080 tokens), being 1,708 of them disambiguated and 117 annotated (out-of-vocabulary words).

When searching for a lexicon to correct the lemmas, we found one that contained all possible PoS tags for each form, with all possible lemmas and morphological features such as: gender (used for nouns, adjectives and pronouns), tense, mode, person (used for verbs), and number (used for various categories). We saw the opportunity to map the tag set used by the resource to the UD tag set, which allowed us to automatically check the lemma and feature annotations. This mapping proved to be more complex than expected, and we ended up having to make several improvements in the process, but the resulting lexicon (Lopes et al., 2022) has helped us automate several tasks ever since.

This step turned out to be the shortest (excluding the time spent on building the lexicon), since 98.91% of the lemmas were automatically revised using the lexicon and only 1.09% required manual revision.

3.3 STEP 3 - Revising FEATS

Unlike the UPOS and LEMMA columns, which have a label and a lemma for each token respectively, the FEATS column does not have a one-to-one relationship with the tokens. In fact, 42.8% of the 168,080 tokens in the corpus did not require any feature, and 57.2% required one or more features, depending on their UPOS. The corpus has a total of 281,970 features unequally distributed among the 96,134 tokens that require them. Given a token, plus its LEMMA and UPOS, we expected to automatically solve the FEATS revision, using the lexicon we customized in the previous step. However, even with this triple data input, there were tokens that admit more than one possible set of features in Portuguese. In this step, human intervention was required to resolve 8,050 cases (7,933 ambiguities and 117 out-of-vocabulary words). These ambiguous tokens pertain to the VERB (7,543 cases), PRON (3,822) and NOUN (132) classes, while the out-of-vocabulary words pertain to NOUN (93), ADJ (22), VERB (1), and ADP (1).

Therefore, the FEATS revision was predominantly automatic, with only 4.79% of the tokens requiring human revision, as described in more detail in Lopes et al. (2024).

3.4 STEP 4 - Revising HEAD-DEPREL

The task of revising dependency relations involves several operations: identifying HEAD errors, detecting DEPREL errors, and suggesting both a corrected HEAD and an appropriate DEPREL label to replace the incorrect annotation. Furthermore, when the error affects the annotation of the sentence root, a series of additional modifications is required, making this step the most complex in the entire process. Just like in the UPOS step, in some instances annotators overlooked errors and made no changes. However, when corrections were made, several scenarios occurred:

- the error was correctly identified and appropriately corrected;
- the error was correctly identified, and the DEPREL was correctly changed, but a necessary change of HEAD had not been made;
- the error was correctly identified, but an incorrect correction was applied to HEAD or DEPREL or both;
- the error was incorrectly identified and the correction introduced a HEAD or DEPREL error or both.

In this phase, our team consisted of four annotators and one adjudicator. The best annotators from the UPOS step were hired for the DEPREL step. However, not all of them repeated their good performance, perhaps because DEPRELs are harder and require more in-depth logical thinking, which is not always the case with the UPOS revision.

At the beginning of this step, 400 sentences received double-blind annotation from two annotators (200 of each pair) and, after calculating the inter-annotator agreement, all the sentences were analyzed by a more experienced linguist, in order to check the complexity of the task as a whole.

The inter-annotator agreement (Table 3) combines relations that were revised and considered correct and relations that were changed in the same way by both annotators (which we refer by pairs of annotators A1-A2 and A3-A4), but does not reflect all possible scenarios. When analyzing the results of the first 400 sentences, we noticed that in most cases one annotator saw an error and another

annotator saw another, both of which were relevant. In several cases, both annotators missed an error. In addition, we noticed some cases of intra-annotator disagreement (when annotators deviated from the guidelines and disagreed with their own earlier decisions for similar cases).

Annotators	DEPREL (%)	HEAD (%)	HEAD+DEPREL (%)
A1-A2	96.92	97.21	95.96
A3-A4	97.67	97.79	96.62
average	97.50	97.29	96.29

Table 3: Human annotators agreement for HEAD-DEPREL revision.

To overcome these problems, instead of using double-blind annotation and inter-annotator agreement to guide the adjudication, we adopted in this step the double non-blind revision: the annotators checked each other's work (each package received a first and a second revision sequentially) and they were allowed to communicate to discuss disagreements. This proved to be an appropriate decision, as we combined the revision capacities, generating synergy. Moreover, we noticed greater motivation on the part of the annotators when the task was no longer totally solitary. The cases in which the annotators were unable to reach a consensus were revised by an experienced linguist. These cases sometimes required study before a decision was adopted and became part of our annotation manual. Problems for which we could not find a clear solution were discussed via issues on UD's github.

At this step, we verified two facts that probably occur in other languages: a) there is not always a direct correlation between sentence length and annotation complexity (many long sentences are a combination of very simple clause patterns); b) nominal predicates presented more difficult constructions to annotate than verbal ones.

During DEPREL revision, we noticed correlations between UPOS and DEPREL, as well as correlations between some features and DEPREL, which could be used to identify recurring errors. These findings inspired the construction of an error checker (Lopes et al., 2023), which played a crucial role in improving the consistency of the annotation.

The HEAD and DEPREL of the 168,080 tokens (100% of the corpus) were fully revised by humans. Of this total, 15,358 (9.14%) had a HEAD change and 13,816 (8.22%) had a DEPREL change. Of these, a total of 6,542 (3.89%) tokens had their HEAD and DEPREL changed simultaneously.

The DEPREL revision provides a very suitable scenario for doing what Pandey et al. (2020) proposed: studying annotation as a psychological process. Building on that, we observed these interesting things on our psychological process analysis:

- when annotators realize that the parser makes few mistakes, they begin to "trust" the parser and start to question the annotation less, missing the errors;
- annotators believe that, if the parser gets difficult things right, it will not get easy things wrong; therefore, things that are considered "easy" are taken out of the focus of the revision and "silly" mistakes are no longer corrected (for example, in Portuguese, as in English, the copula verb is also a passive auxiliary (to be), but this is so often well distinguished by the parser that a label mistake goes unnoticed);
- annotators also believe that the "lightning does not strike the same tree twice" and, when they find an error in a sentence, they sometimes are blind to other errors in the same sentence;
- annotators often do not recognize patterns in less frequent constructions, separated by a long time interval (3 days or more); this leads them to annotate similar constructions in different ways, what seems to be a case of slip, that is, an error type caused by reasons different from absence of knowledge, probably due to memory decay (with specific regard to memory decay in human annotation, see Pandey et al. 2020);
- annotators miss most frequently errors regarding functional words, as they naturally tend to engage in a "skimming and scanning" reading process, focusing more on content words.

3.5 Overview of the process

We gained valuable insights throughout the process. Primarily, we learned that each annotation layer requires different linguistic knowledge and different annotator profiles. The cascade approach required human annotators at all steps, including STEPS 2 and 3, where the automation of most cases relieved the workload. Although both STEPS 1 and 4 heavily employed human resources, STEP 1 required annotators focused on pattern recognition with some

step	CoNLL-U column	human revision		tool to revise	performed tasks	required knowledge	automatic revision		tokens changed		tokens unchanged	
1	UPOS	168,080	100.0%	Arborator-Grew	revision	morphosyntax	–	0%	6,440	3.83%	161,640	96.17%
2	LEMMA	1,825	1.09%	spreadsheet	disamb./annot.	morphology	166,255	98.91%	3,649	2.17%	164,431	97.83%
3	FEATS	8,050	4.79%	spreadsheet	disamb./annot.	lexicography	160,030	95.21%	29,274	17.42%	138,806	82.58%
4	HEAD	168,080	100.0%	Arborator-Grew	revision	syntax	–	0%	15,358	9.14%	152,722	90.86%
	DEPREL	168,080	100.0%	Arborator-Grew	revision	syntax	–	0%	13,816	8.22%	154,264	91.78%

Table 4: Summary of revision steps.

knowledge of morphosyntax, while STEP 4 required annotators with in-depth logical reasoning and solid knowledge of syntax. As the learning curve is long, we should avoid hiring a workforce with high turnover and, ideally, multitasking annotators should be trained. People with knowledge of Computational Linguistics are essential both for designing the tasks and for spotting opportunities to optimize them. Likewise, computer support is essential at all stages of the process. Table 4 summarizes the results of each step.

4 Related work

The lack of a parser was a barrier for low-resource languages to start annotation for the morphosyntactic and syntactic layers. However, with datasets and multilingual models, the barrier is no longer the lack of a parser, but the lack of resources and systematic procedures to efficiently revise the pre-annotated corpus. In recent years, various proposals have been put forward to save effort in human revision. The following are some of them.

Hovy et al. (2014) adopt crowd-sourced lay annotators to annotate PoS tags, putting the target word in bold, one context token on the left and one on the right, and presenting multiple choice questions, abridging the process of annotating from scratch. They used majority voting to decide disagreements. The model trained on the resulting data achieved slightly less than an expert in the task (82.6% and 86.8%, respectively). Using a lexicon, they performed a new task, only restricting the labels available for a given token, achieving 83.7%.

Weissweiler et al. (2023) examined the morphological capabilities of ChatGPT in 4 languages (English, German, Turkish and Tamil) and found that in none of them did LLM achieve human-level performance in the proposed tasks, nor did it match the state-of-the-art models.

Freitas and de Souza (2024) used two different models to annotate the corpus (UDPipe 2 and Stanza) and performed a human revision of all cases of disagreement between the two automatic annotations, adopting the heuristic that the agreement of the systems would be indicative of the correct annotation.

Machado and Ruiz (2024) evaluated 3 LLMs in PoS tag assignment using UD tag set in texts written in Brazilian Portuguese and showed that the best performance was achieved by ChatGPT-3, with 90% of accuracy.

None of them, however, covers the complete revision of the corpus.

5 Final remarks

Porttinari-base was launched in 2023 (Duran et al., 2023) and has been used to train a state-of-the-art parser (Lopes and Pardo, 2024), reaching over 96% of accuracy. We have been using this parser to pre-annotate corpora of new genres within the larger multi-genre project Porttinari.

The divide-and-conquer strategy was very successful: the expected cascade effect was achieved, leading to an increasing reduction in errors. We hypothesize that, just as one annotation layer benefits greatly from improvements in another layer, small improvements in the performance of a tagger or parser can significantly impact the performance of downstream applications.

For the interested reader, all the resources and tools that we mentioned are freely available on the POeTiSA project website: `https://sites.google.com/icmc.usp.br/poetisa`

Acknowledgments

This work was carried out at the Center for Artificial Intelligence of the University of São Paulo (C4AI - http://c4ai.inova.usp.br/), with support by the São Paulo Research Foundation (FAPESP grant #2019/07665-4) and by IBM Corporation. The project was also supported by the Ministry of Science, Technology and Innovation, with resources of Law n. 8.248, of October 23, 1991, within the scope of PPI-SOFTEX, coordinated by Softex and published as Residence in TIC 13, DOU 01245.010222/2022-44.

References

Sohail Akhtar, Valerio Basile, and Viviana Patti. 2021. Whose opinions matter? perspective-aware models to identify opinions of hate speech victims in abusive language detection. *CoRR*, abs/2106.15896.

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 '06, pages 149–164, USA. Association for Computational Linguistics.

Jean Carletta. 1996. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Magali Duran, Lucelene Lopes, Maria das Graças Nunes, and Thiago Pardo. 2023. The dawn of the Porttinari multigenre treebank: Introducing its journalistic portion. In *Anais do XIV Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, pages 115–124, Porto Alegre, RS, Brasil. SBC.

Magali Sanches Duran. 2021. Manual de anotação de PoS tags: Orientações para anotação de etiquetas morfossintáticas em língua portuguesa, seguindo as diretrizes da abordagem Universal Dependencies (UD). Technical Report 434, ICMC-USP.

Magali Sanches Duran. 2022. Manual de anotação de relações de dependência: Orientações para anotação de relações de dependência em língua portuguesa, seguindo as diretrizes da abordagem Universal Dependencies (UD). Technical Report 440, ICMC-USP.

Cláudia Freitas and Elvis de Souza. 2024. A study on methods for revising dependency treebanks: in search of gold. *Language Resources and Evaluation*, 58(1):111–131.

Gaël Guibon, Marine Courtin, Kim Gerdes, and Bruno Guillaume. 2020. When collaborative treebank curation meets graph grammars. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5291–5300, Marseille, France. European Language Resources Association.

Jan Hajič and Dan Zeman, editors. 2017. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics, Vancouver, Canada.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

Dirk Hovy, Barbara Plank, and Anders Søgaard. 2014. Experiments with crowdsourced re-annotation of a POS tagging data set. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 377–382, Baltimore, Maryland. Association for Computational Linguistics.

Eduard Hovy and Julia Lavid. 2010. Towards a science of corpus annotation: a new methodological challenge for corpus linguistics. *International Journal of Translation*, 22:13–36.

Nancy Ide. 2007. Annotation science: From theory to practice and use (invited talk). In *Data Structures for Linguistic Resources and Applications. Proceedings of the Biennial GLDV Conference 2007, 11.–13. April, Universität Tübingen*, pages 1–5, Tübingen. Narr.

Nancy Ide and James Pustejovsky, editors. 2017. *Handbook of Linguistic Annotation*. Springer, Dordrecht.

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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2779–2795, Hong Kong, China. Association for Computational Linguistics.

Elisa Leonardelli, Gavin Abercrombie, Dina Almanea, Valerio Basile, Tommaso Fornaciari, Barbara Plank, Verena Rieser, Alexandra Uma, and Massimo Poesio. 2023. SemEval-2023 task 11: Learning with disagreements (LeWiDi). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2304–2318, Toronto, Canada. Association for Computational Linguistics.

Lucelene Lopes, Magali Duran, Paulo Fernandes, and Thiago Pardo. 2022. PortiLexicon-UD: a portuguese lexical resource according to Universal Dependencies model. In *Proceedings of the Language Resources and Evaluation Conference*, pages 6635–6643, Marseille, France. European Language Resources Association.

Lucelene Lopes, Magali Duran, and Thiago Pardo. 2024. Desambiguação de lema e atributos morfológicos na anotação do córpus Porttinari-base. In *Anais do XV Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, pages 336–345, Porto Alegre, RS, Brasil. SBC.

Lucelene Lopes, Magali S. Duran, and Thiago A. S. Pardo. 2021. Universal dependencies-based pos tagging refinement through linguistic resources. In *Intelligent Systems*, pages 601–615, Cham. Springer International Publishing.

Lucelene Lopes, Magali S. Duran, and Thiago A. S. Pardo. 2023. Verifica UD - a verifier for Universal Dependencies annotation in Portuguese'. In *Proc. of the UDFest-BR 2023*, UDFest-BR, pages 1–8.

Lucelene Lopes and Thiago Pardo. 2024. Towards Portparser - a highly accurate parsing system for Brazilian Portuguese following the Universal Dependencies framework. In *Proceedings of the 16th International Conference on Computational Processing of Portuguese - Vol. 1*, pages 401–410, Santiago de Compostela, Galicia/Spain. Association for Computational Lingustics.

Mateus Machado and Evandro Ruiz. 2024. Evaluating large language models for the tasks of PoS tagging within the Universal Dependency framework. In *Proceedings of the 16th International Conference on Computational Processing of Portuguese - Vol. 1*, pages 454–460, Santiago de Compostela, Galicia/Spain. Association for Computational Lingustics.

Rahul Pandey, Carlos Castillo, and Hemant Purohit. 2020. Modeling human annotation errors to design bias-aware systems for social stream processing. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '19, pages 374–377, New York, NY, USA. Association for Computing Machinery.

Thiago Pardo, Magali Duran, Lucelene Lopes, Ariani Felippo, Norton Roman, and Maria Nunes. 2021. Porttinari - a large multi-genre treebank for Brazilian Portuguese. In *Anais do XIII Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, pages 1–10, Porto Alegre, RS, Brasil. SBC.

Maja Pavlovic and Massimo Poesio. 2024. The effectiveness of llms as annotators: A comparative overview and empirical analysis of direct representation. In *3rd Workshop on Perspectivist Approaches to NLP, NLPerspectives 2024 at LREC-COLING 2024 - Workshop Proceedings*, 3rd Workshop on Perspectivist Approaches to NLP, NLPerspectives 2024 at LREC-COLING 2024 - Workshop Proceedings, pages 100–110. European Language Resources Association (ELRA). Publisher Copyright: © 2024 ELRA Language Resource Association.; 3rd Workshop on Perspectivist Approaches to NLP, NLPerspectives 2024 ; Conference date: 21-05-2024.

James Pustejovsky, Harry Bunt, and Annie Zaenen. 2017. *Designing Annotation Schemes: From Theory to Model*, pages 21–72. Springer Netherlands, Dordrecht.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. *CoRR*, abs/2003.07082.

Paul Röttger, Bertie Vidgen, Dirk Hovy, and Janet Pierrehumbert. 2022. Two contrasting data annotation paradigms for subjective NLP tasks. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 175–190, Seattle, United States. Association for Computational Linguistics.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Honolulu, Hawaii. Association for Computational Linguistics.

Milan Straka. 2018. UDPipe 2.0 prototype at CoNLL 2018 UD shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207.

Tiago Timponi Torrent, Thomas Hoffmann, Arthur Lorenzi Almeida, and Mark Turner. 2024. *Copilots for Linguists: AI, Constructions, and Frames*. Elements in Construction Grammar. Cambridge University Press.

Leonie Weissweiler, Valentin Hofmann, Anjali Kantharuban, Anna Cai, Ritam Dutt, Amey Hengle, Anubha Kabra, Atharva Kulkarni, Abhishek Vijayakumar, Haofei Yu, Hinrich Schuetze, Kemal Oflazer, and David Mortensen. 2023. Counting the bugs in ChatGPT's wugs: A multilingual investigation into the morphological capabilities of a large language model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6508–6524, Singapore. 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. Association for Computational Linguistics.