Semantic Parsing and Sense Tagging the Princeton WordNet Gloss Corpus

Alexandre Rademaker
IBM Research and FGV/EMAp
alexrad@br.ibm.com

Abhishek Basu
IBM

Rajkiran Veluri
IBM

Abstract

In 2008, the Princeton team released the last version of the "Princeton Annotated Gloss Corpus". In this corpus. The word forms from the definitions and examples (glosses) of Princeton WordNet are manually linked to the context-appropriate sense in WordNet. However, the annotation was not complete, and the dataset was never officially released as part of WordNet 3.0, remaining as one of the standoff files available for download. Eleven years later, in 2019, one of the authors of this paper restarted the project aiming to complete the sense annotation of the approximately 200 thousand word forms not yet annotated. Here, we provide additional motivations to complete this dataset and report the progress in the work and evaluations. Intending to provide an extra level of consistency in the sense annotation and a deep semantic representation of the definitions and examples promoting WordNet from a lexical resource to a lightweight ontology, we now employ the English Resource Grammar (ERG), a broad-coverage HPSG grammar of English to parse the sentences and project the sense annotations from the surface words to the ERG predicates. We also report some initial steps on upgrading the corpus to WordNet 3.1 to facilitate mapping the data to other lexical resources.

1 Introduction

In the "Princeton Annotated Gloss Corpus" (GlossTag), the content word forms in the definitions and examples in WordNet[1] (Fellbaum, 1998) are manually linked to the context-appropriate sense in WordNet itself. Thus, the glosses are a sense-disambiguated corpus, and WordNet is the dictionary against which the corpus was annotated. The corpus is available for download on the Princeton WordNet website as a standoff package supplementing the WordNet 3.0 release. Although it has already been recognized as a precious resource, not all words have been annotated yet. According to the statistics in the website,[2] by 2008, the corpus contains 206,711 words (including collocations and multi-word expressions) yet to be disambiguated. In (Rademaker et al., 2019), one of the authors reported initial efforts to complete the annotation of the corpus (release 2019). This paper reports the progress on this endeavor (release 2022) with improvements in the methodology and evaluation.

The glosses were introduced in WordNet around 1989 (Fellbaum, 1998); before them, senses were distinguished only by synonyms and semantic relations. From this same reference,"As the number of words in WordNet increased, it became increasingly difficult for us, purely on the basis of synonyms, to keep all the different word senses distinct."

On the other hand, introducing glosses has its problems. First, it is not always easy to write good definitions, and second, glosses introduce information redundancy.

> "In the course of incorporating this kind of explanatory information, we have all acquired greater respect for traditional lexicographers."

[1] We are using the trademark 'WordNet' for the Princeton English Wordnet.

[2] http://wordnetcode.princeton.edu/glosstag.shtml

> "The (somewhat idealistic) hope was that the definition of any word could be inferred from its position in this network of semantic relations and that definitional glosses would be redundant [. . .] If more distinguishing features could be indicated by pointers representing additional semantic relations, the glosses would become even more redundant. An imaginable test of the system would then be to write a computer program that would synthesize glosses from the information provided by the pointers." (Fellbaum, 1998)

Nevertheless, since its introduction, researchers have found many applications for WordNet glosses. Harabagiu and Moldovan (Fellbaum, 1998; Harabagiu et al., 1999; Moldovan and Novischi, 2004; Mihalcea and Moldovan, 2001) propose disambiguating the content words in the glosses to increase the semantic connections among the words and to establish relations among them between different syntactic categories to support common-sense reasoning. In one example, they explain how the disambiguation of the words 'eat' and 'food' in the definitions of the adjective 'hungry,' the verb 'eat,' and the noun 'refrigerator' establish a semantic path between the concepts expressed by the words. Thus, one can infer from 'being hungry' the action of 'going to the refrigerator.' Increasing the semantic connections among WordNet synsets also improves the results of many word sense disambiguation (WSD) algorithms that use the network structure of WordNet to identify the most plausible sense for the words in a context (Agirre and Soroa, 2009; Banerjee and Pedersen, 2002; Basile et al., 2007). By disambiguating the words in the glosses, we add pointers between synsets A and B whenever we annotated a word with a sense from A in B's definition. With that approach, we can increase the connectivity between the WordNet synsets by approximately an order of magnitude.

The disambiguation of words in the glosses can also improve WordNet and provide completeness and consistency. For instance, the initial versions of WordNet do not contain relations that indicate how words like 'racquet', 'ball', and 'net', and the concepts behind them, are part of another concept that can be expressed by 'court game' (Fellbaum, 1998). In WordNet 3.0 the 'domain relations' between synsets were introduced to alleviate this so-called 'tennis problem' of WordNet (Miller, 1993), but the disambiguated gloss of the synset $\{tennis, lawn\_tennis\}$ (1) would already enrich the connections among the concepts. Another desired property is that all words used in the definitions are defined in this same resource. Hopefully, this completeness could also help us ensure quality in our long-term endeavor during the expansion of WordNet to highly technical domains. Once more concepts are added or redefined, the glosses would be refined and disambiguated, forcing us to use the newly added senses in a productive cycle of editing, testing, and correcting.

(1) a game played with **rackets** by two or four players who hit a **ball** back and forth over a **net** that divides the court (00482298-n) [3]

Beyond the disambiguation of words in the glosses, (Clark et al., 2008a,b) used manually constructed logical forms from a subset of the WordNet glosses for text understanding and question answering.

In our approach, we aim at high-quality human annotation, leveraging the lessons learned and directives developed for the project in Princeton but adapting them to our tools. Data is available using the same open license used by Princeton for the initial version of the data (called GlossTag 2008). In (Rademaker et al., 2019), we reported the initial steps of data preparation, our annotation interface's implementation, and a preliminary experiment on the inter-annotator agreement. The result was called GlossTag 2019 (the 2019 release of the corpus).

[3] Glosses, definitions or examples will be followed by the corresponding synset identifiers from WordNet 3.0.

Here we present the GlossTag 2022. In Section 2, we discuss some inconsistencies in the data identified during the annotation; mainly, we ensure that GlossTag 2022 sentences are all effectively derived from the WordNet 3.0 glosses. In Section 3, we explain why we employed the English Resource Grammar (ERG) to parse the sentences into syntactic and semantic structures, hopefully more consistent than previous pre-processing annotations in GlossTag 2008 and manually constructed logical forms (Clark et al., 2008a,b; Niles and Pease, 2003; Pease and Cheung, 2018). We also explain the projection of the sense annotations from the surface words to the ERG predicates. This helped us improve consistency and facilitate sense annotation, mainly of verbs, where senses are normally related to specific syntactic valence alternations. In Section 4 we evaluated the UKB word-sense disambiguation algorithm (Agirre and Soroa, 2009) in the annotation of the GlossTag itself. The idea was to test the feasibility of having an algorithm give hints to the sense annotator and produce intermediary 'silver' versions of the data in future releases. In Section 5 we make initial considerations about the challenge to migrate the GlossTag 2022 annotations to WordNet 3.1. We presented some final considerations and future work in Section 6.

2 Data validation and preparation

As reported in (Rademaker et al., 2019), from the GlossTag 2008 XML files, we built the GlossTag 2019 JSON-Lines files with one JSON object per line for each gloss. The transformation was done mainly to facilitate the data ingestion in the backend of our annotation tool with elementary validations. In GlossTag 2019, we focused on the annotation job and assessing its complexity only. Nevertheless, the work over the last three years reveals inconsistencies, and data need to be prepared to be combined with their semantic representation obtained from the English Resource Grammar.

In (Miller and Fellbaum, 2007), the authors briefly mentioned the sense tagging of the WordNet glosses. The description is not detailed enough to conclude if they are describing the GlossTag 2008. The best documentation about the dataset is provided as comments on a DTD file that specifies the `wordnet` document type of the XML files in the 'merged' folder of the dataset.[4] Every 'synset' element in the DTD contains three 'gloss' child elements. One with the attribute 'orig', is a string that allegedly matches the gloss of this same synset in the WordNet 3.0 DB files. The second, with the attribute 'text', is also a string, with extra spaces indicating the tokenization and quotes encoded in UTF-8. The third, with the attribute 'wsd', is the one that holds the actual annotations of words and collocations in child elements.

The glosses in WordNet 3.0 can be preceded by a domain classification fragment and/or an auxiliary fragment, both usually in parenthesis and optionally followed by more auxiliary fragments and zero or more examples. For sense tagging purposes, the original annotators ignored the classification fragments, as the information is normally repeated in the usage, region, and category pointers. The auxiliary fragments are always secondary to the primary sense of the synset; they can precede or follows the definition but can also be embedded within the definition. Auxiliary fragments are tagged with 'ignore' or 'arg.' Those assigned with the tag 'ignore' are ignored for sense tagging and contain mainly grammatical or usage information, some qualifying text such as a year born, time, date range, or a chemical or other symbols.

In (2) we show the gloss of the synset $\{wash\}$ (verb) with a fragment assigned with tag 'arg', the argument or typical argument (in green), for the preceding verb (in blue). They are set off in this way so that the syntax of the definition fits that of the lemma (the defining verb is intransitive if the lemma is intransitive).

(2) to cleanse (itself or another animal) by

[4] https://wordnetcode.princeton.edu/glosstag.shtml

licking; "The cat washes several times a day" (00036178-v)

Inside the definitions and examples, we have the 'wf' (word form), 'cf' ('wf' that are part of one or more collocations), and 'mwf' elements. The 'wf' and 'cf' are marked up with parts of speech and potential lemma forms at WordNet 3.0. The collocations are marked in a way that can even indicate discontiguous forms. The 'wf' can also be annotated with some semantic classes: punctuation, year, chemical name, number, time, currency, abbreviations, or mathematical symbol. The 'mwf' are multi-word forms composed by 'wf' and 'cf' children that can also be annotated with semantic classes: date, date/numeric range, numeric form, currency, measurement, mathematical formula, and other groups of symbols. The 'wf' and 'cf' that have been disambiguated are further annotated with WordNet sense keys and the flag indicating if the annotation was done automatically or manually. Furthermore, 'wf' and 'cf' elements may contain a separator attribute with the character separating the corresponding form from the next in print. Valid values for this attribute are hyphen, empty string, and space for hyphenated words not in WordNet, contractions that get split (in 'cf' forms), and cases where no space follows the form. The default value is a space, not explicitly assigned. We should be able to reconstruct the original text of the glosses in the WordNet DB files using the separators, but this was not true for approximately 2100 glosses; we found and fixed some mismatches also caused by extra semi-colons added in the end of the examples.

We know almost nothing about how the original text of the glosses was processed to produce all these mark ups in the GlossTag 2008. What are the tokenization criteria? How were the semantic classes identified? How the definitions and examples segments were identified in a given gloss text. Moreover, some essential details are provided in the DTD. The part-of-speech (POS) tags were automatically assigned only to word forms in the definitions, not in the examples, apparently because examples were supposed to be partially annotated; according to another comment, that says 'only synset terms in examples should be sense tagged,' but we have a more ambitious goal.

Once the tokenization issues were solved, and data was confirmed to correspond to the original WordNet 3.0 DB files, we split the glosses into sentences (definitions and examples). Approximately 758 examples in WordNet 3.0 are quotes such as (3), that is, quotes followed by the author's name or source. We removed the quote marks and moved the author's name (or title of the publication) to the metadata associated with the example.

(3) "their views of life were reductive and depreciatory" - R.H.Rovere (00050446-a)

Finally, we calculated the text span of each word form (also known in the literature as token ranges) in the sentences. As we will see in the rest of the paper, once we parse the sentences with ERG to produce the semantic representations, we need to match the predicates obtained from the ERG analysis with the word forms in the GlossTag 2022 using the text spans. That is the reason for such careful considerations about tokenization. The missing POS in the examples were obtained from ERG analysis.

3 Parsing with English Resource Grammar

The English Resource Grammar (ERG) (Flickinger, 2000, 2011) is a broad-coverage, general-purpose computational grammar that, combined with specialized tools, can map running English text to highly normalized logical-form representations of meaning. ERG is a linguistically precise HPSG-based grammar of English and semantically grounded in Minimal Recursion Semantics (MRS) (Copestake et al., 2005), which is a form of flat semantic representation capable of supporting underspecification. The ERG is developed as part of the international Deep Linguistic Processing with HPSG Initiative

(DELPH-IN) [5] and can be executed by some parsing and realization systems, including the LKB grammar engineering environment (Copestake, 2002), as well as the more efficient ACE parser,[6] for applications.

We grouped the GlossTag 2022 sentences in profiles, test suites, collections of test items for judging the performance of an implemented grammar within DELPH-IN. While the original purpose of test suites is to aid in grammar development, they are more generally useful for batch processing. [incr tsdb()](Oepen, 2001) is the canonical software for managing the profiles but PyDelphin Library (Goodman, 2019) is an alternative. A profile is just a relational database. However, the data are stored in flat text files on disk instead of using a standard SQL database, and the profile is the folder. The file relations describe the database schema of this profile; its syntax is described in (Oepen, 2001). Individual relations (or tables) are stored in separate files with the same name as the relation. The SQL-like query language TSQL can be used to query profiles.

After creating the profiles with 2000 sentences each, we processed them with the Ace parser in a cluster, running each profile in parallel. It took about 30 minutes. For each sentence, we asked for the top-best analysis of ERG. GlossTag 2022 contains 165,976 sentences; from these, only 5,282 were not parsed by ERG. Using some heuristics (the most productive one is adding an extra 'X' in sentences ending with the preposition 'of', e.g. "get the votes *of X*"), we were able to parse roughly 600 more sentences (only 2% are not parsed).

Since ERG is a computational grammar and sentences are typically ambiguous, we can have hundreds or thousands of readings for each sentence. We stored only the top-best analysis according to the pre-trained parsing ranking model distributed with ERG.[7] This is not to say that all analyses were the expected ones, but informal evaluation gives us some great expectations. In a future experiment, we aim to employ FFTB (Packard, 2015) for gradually treebanking all sentences. FFTB allows the selection of an arbitrary tree from the 'full forest' without enumerating/unpacking all analyses in the parsing stage. The treebanking of all sentences would ensure the data's quality and the actual evaluation of the parsing selection model. We aim to turn GlossTag 2022 into a dynamically annotated treebank (Flickinger et al., 2012; Oepen et al., 2002).

For each item (sentence) in a profile, once it was processed, we have the derivation tree and the semantic representation MRS.[8] Figure 1 presents one MRS. Predicate–argument structure is expressed in a bag of n-ary elementary predications (EP) linked together by typed variables.[9] The predicate symbols can be divided into surface predicates and abstract predicates. Surface predicates follow a naming convention where the symbol is composed of three components, called 'lemma', 'pos' (mostly align with a crude inventory of word classes (n)oun, (q)uantifier, (v)erb and (a)djective, etc), and 'sense' (coarse-grained senses, ERG only marks those sense distinctions that are morphosyntactically marked). Surface predicates, by convention, are marked by a leading underscore and are exclusively introduced by lexical entries from the grammar, whose orthography is a (possibly inflected) form of the lemma field in the predicate. The predicate `_palmately/rb_u_unknown` is a generic predicate instantiated by ERG for dealing with the unknown word.[10] The numbers following the predicate name indicate the text span to which the EP corresponds.[11]

[5] https://github.com/delph-in/docs/wiki/

[6] https://github.com/delph-in/docs/wiki/AceTop

[7] We are ignoring details about all other parameters that control the ACE parser.

[8] Among many additional information that we do not have space to describe.

[9] Eventualities (e), instances (of type x), labels or handles (of type h), and underspecified (u and i).

[10] Not explicited defined in its lexicon.

[11] The most complete and up-to-date presentation of ERG semantics can be found in https://github.com/delph-in/docs/wiki/ErgSemantics.

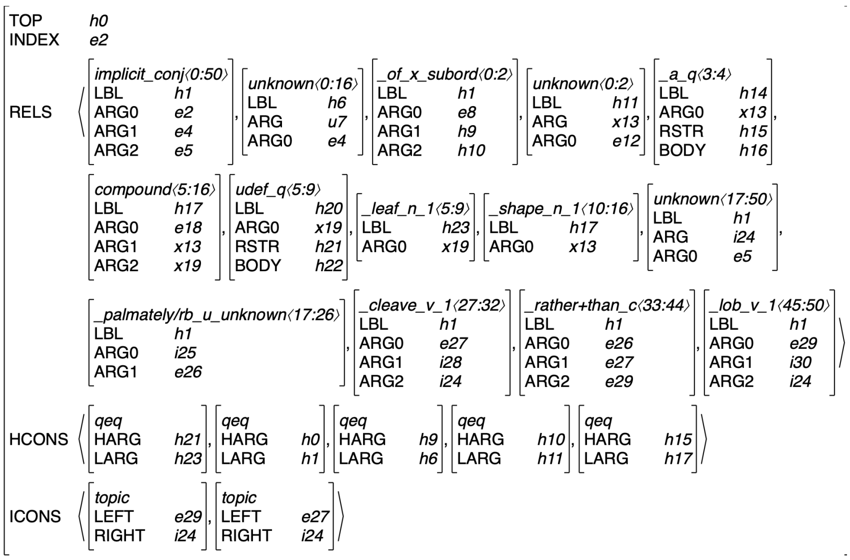

Figure 1: MRS of the definition "of a leaf shape; palmately cleft rather than lobed" (02173264-a)

4 Speeding up the annotations

Manual word sense disambiguation (WSD) is an arduous task, but many techniques for automatic WSD are being investigated. Automatic WSD methods include graph-based (or knowledge-based), supervised and unsupervised machine learning methods (Bevilacqua et al., 2021). Since GlossTag 2022 is still not wholly annotated, having an automatic method to complete the annotation or filter the most plausible senses for the human annotator is appealing. The automatic annotation would allow us to provide intermediary releases of the GlossTag, but we need to estimate the quality of such 'silver' version.

Note that the GlossTag 2008 was already used by many WSD approaches (Bevilacqua et al., 2021). It has been used as a dataset for training supervised WSD algorithms.[12] and also to increase the connectivity among synsets by (Agirre and Soroa, 2009). In this section, we used UKB (Agirre and Soroa, 2009), a graph-based approach for WSD. It applies random walks, e.g., Personalized PageRank, on the Knowledge Base (KB) graph to rank the vertices according to the given context. UKB has been shown to perform almost as well as supervised methods or even outperform them on specific domains (Agirre et al., 2018, 2009). Since UKB uses GlossTag, this creates a possible circularity, problematic for WSD evaluation but not for our goal. We took the GlossTag 2022 sentences, removed all the annotated senses, and passed the sentences to UKB. Given the results of UKB, for each word, we compare the annotations we already have in the data with the sense provided by UKB, evaluating the performance of UKB.

Figure 2 presents the GlossTag information of the same definition processed by ERG and presented in Figure 1 in a tabular format. To produce the UKB input (Figure 3), we have to consolidate the information obtained from ERG with the GlossTag annotations, which is not easy. MWE must be combined into a single token, and all tokens must have lemma and POS so that UKB can disambiguate them. But in Figure 2, tokens 9-10 are not marked as 'cf' (MWE). Token 11 was tagged as an adjective but manually disambiguated and analyzed by ERG as a verb. On the other hand, the MWE 'leaf shape' (Tokens 3-5) matches with the ERG analysis that identified the expression with the 'compound' predicate.[13]

[12] Replacing the well-known but controversial SemCor (semantic concordance), a subset of the Brown Corpus (Miller, 1993) and other small corpora used in the previous SemEval tasks.

[13] We are skipping details related to obtaining the lemmas `rather` and `leaf_shape` from the predicates

```
# text = of a leaf shape; palmately cleft rather than lobed
# id = 02173264-a
# type = def
1	wf	ignore	0:2	IN	of	of	_
2	wf	ignore	3:4	DT	a	a	_
3	glob|a	auto	_	_	_	leaf_shape%1	leaf_shape%1:25:00::
4	cf|a	un	5:9	NN	leaf	leaf%1|leaf%2	_
5	cf|a	un	10:15	NN	shape	shape%1|shape%2	_
6	wf	ignore	15:16	:	;	_	_
7	wf	auto	17:26	RB	palmately	palmately%4	palmately%4:02:00::
8	wf	man	27:32	VBN	cleft	cleft%1|cleave%2|cleft%3	cleft%5:00:00:compound:00
9	wf	un	33:39	RB	rather	rather%4	_
10	wf	ignore	40:44	IN	than	than	_
11	wf	man	45:50	JJ	lobed	lob%2|lobed%3	lob%2:35:00::
```

Figure 2: GlossTag 2022 tabular presentation of a sentence. The lines starting with hash contains sentence metadata. Each word is presented in a line, column 1 is the identifier, column 2 is the word type, column 3 the annotation flag, column 4 the text span, column 5 the part-of-speech tag (when available), column 6 the form in the sentence, column 7 the possible WordNet 3.0 lemmas and column 8 the sense, when annotated.

Figure 3 presents the UKB inputs for the sentence from Figures 1 and 2. Two consecutive lines represent each context. The first line contains the context identifier, whereas the second one contains the words to be disambiguated. Each element in a context has four mandatory fields; lemma and POS are the most important ones. UKB then disambiguates all the words from the input in a single run. UKB can deal with partially disambiguated contexts and use the provided concept identifiers (synset identifiers) to give extra information in the disambiguation of the remaining tokens. Given that, we generated two contexts for each sentence. In the first context in Figure 3, we included one extra token, the synset identifier.

For evaluation, we only considered the words in the GlossTag which are associated with at least one sense.[14] We have looked for a match by checking if UKB generated sense is a subset of the senses provided in the annotations. The total number of words disambiguated and considered for evaluation using UKB was 819,533. Among them, the ones with senses that were also disambiguated by UKB sum up to 442,782. Table 1 shows the results for the contexts with the additional synset identifier (a) and the results for the contexts without the additional synset identifier (b).

	Total	# (a)	# (b)	% (a)	% (b)
All	442782	413546	374648	93.39	84.61
Noun	329692	308245	287033	93.49	87.06
Adj	64298	60591	52008	94.23	80.89
Verb	41520	37832	29529	91.11	71.12
Adv	7272	6878	6078	94.58	83.58

Table 1: UKB evaluation results by part-of-speech. Columns # shown the counts of matches and columns % the percentage.

The majority of the UKB errors involved words that are highly polysemic. For example, the verb ‘make’ has 52 senses in WordNet 3.0. Synset 00891038-v has the definition “assure the success of” and example “A good review by this critic will **make** your play!”. UKB does not annotate the correct sense of ‘make’ in the example, even in the context where the synset identifier itself is added as an extra fake word. Finally, we have definitions such as “of or relating to taxonomy” (03018498-a) with only two content words, not enough information to UKB. After some error additional analysis, we have found some necessary improvements for further evaluation. UKB did not find many lemmas in WordNet 3.0 because they were not lower-cased properly. Many cases of MWE were not annotated in the GlossTag nor detected correctly by ERG also need to be fixed. The mapping of ERG com-

_rather+than_c, _leaf_n_1 and _shape_n_1.

[14] In our annotation guideline, the annotators can annotate more than one sense for each word.

```
ctx-02173264-a/a
leaf_shape#n#w4#1#1 palmately#r#w7#1#1 cleave#v#w8#1#1 rather#r#w9#1#1 lob#v#w11#1#1 02173264-a#a#fake1#2#1

ctx-02173264-a/b
leaf_shape#n#w4#1#1 palmately#r#w7#1#1 cleave#v#w8#1#1 rather#r#w9#1#1 lob#v#w11#1#1
```

Figure 3: UKB Input Context Example

pounds [15] and GlossTag globs needs improvements. Nevertheless, we can safely conclude that adding the synset identifier as an additional word in the context helps UKB. It seems to justify the use of UKB to automatically annotate missing senses and thus generate a ‘silver’ release of GlossTag 2022.

5 The ongoing update to WordNet 3.1

In the latest version of WordNet, Princeton team applied minor fixes in the texts of the glosses and removed many newly considered offensive words. Besides adding (676 senses) and removing senses (382 senses), some WordNet 3.0 senses have moved between synsets, or the corresponding synsets were changed in WordNet 3.1. Given these changes, projecting the annotations in GlossTag 2022 to the senses of WordNet 3.1 needs some careful consideration. This section presents our initial considerations and plans to make the migration.

An extra motivation for moving GlossTag 2022 to WordNet 3.1 is that other lexical resources like VerbNet (Schuler, 2005) are already mapped to WordNet 3.1. Using those mappings, one can enrich the information of verbs in WordNet, restricted to verb frames (‘Somebody –s something’) with additional information like valences, semantic restrictions, etc. This extra information could facilitate sense annotation. For example, the verb ‘make’ has 52 senses in WordNet 3.0, grouped into six classes in VerbNet. If the annotator is first presented with the information from VerbNet, it can first choose the VerbNet class by selecting the proper syntactic restrictions and later select the WordNet senses in that class.

The projection of the annotations of GlossTag to WordNet 3.1 needs to deal with the following cases. First, we need to identify which definitions and examples changed. The new sentences need to be processed by ERG and prepared for manual annotation from scratch. The removed sentences can be just removed. Next, we must consider each word in the sentences preserved in WordNet 3.1. We need to consider the annotated words only and what happens with the used sense key. If sense keys were not reused with a different meaning, we would have no problem. Unfortunately, we found cases where a given sense key got a different meaning in WordNet 3.1.

For example, in WordNet 3.1 we have the word ‘Pluto’ with the sense key `pluto%1:17:00::` which has the gloss “a large asteroid that was once thought to be the farthest known planet from the sun; it has an elliptical orbit” and the example “Pluto was discovered by Clyde Tombaugh in 1930”. In WordNet 3.0, the same sense key `pluto%1:17:00::` is part of a synset with the definition “a small planet and the farthest known planet from the sun; it has the most elliptical orbit of all the planets”. Note how the definition changed from planet to asteroid. Since the concept has changed, the relations have also changed; it is now an instance of ‘asteroid’ instead of ‘outer planet’ and ‘superior planet’. In this case, it would have been more appropriate to introduce a new sense key to signify a deviation of the new definition from the old one. Another sense of ‘Pluto’ in WordNet 3.0 is part of the synset “(Greek mythology) the god of the underworld in ancient mythology; brother of Zeus and husband of Persephone”. In WordNet 3.0, this sense of Pluto

[15] Compounding comprises a variety of (semantic) head–modifier structures that can often be paraphrased using overt prepositions.

is part of the synset "(Roman mythology) god of the underworld; counterpart of Greek Hades". In WordNet 3.0, Pluto was defined as a synonym of Hades, but WordNet 3.1 revised that definition making it part of Roman mythology and a counterpart of Hades. There are eight occurrences of 'pluto' in the WordNet 3.0 sentences. For instance, the definition "United States astronomer who discovered the planet Pluto (1906-1997))" was not updated to follow the new definitions in WordNet 3.1. This shows how hard it is to keep the glosses consistent with the WordNet structure.

Another challenge arises when a new sense is introduced in WordNet 3.1, and some words in the sentences could be better annotated with the new sense. For example, if we look at the senses of the word 'technology', we note that there is a new sense introduced in WordNet 3.1, with the synset 03707142-n and the sense key `technology%1:06:00::` with the definition "machinery and equipment developed from engineering or other applied sciences". In the GlossTag 2022 we found 53 instances of the word 'technology' annotated, and the new sense from WordNet 3.1 may be more appropriate for some of them. Upon manual inspection, we found that this is indeed the case in one of the examples of synset 08343534-n "has procured nuclear technology and delivery capabilities". In this gloss definition, 'technology' may be better mapped to the new sense at WordNet 3.1 rather than any of the other existing senses in WordNet 3.0. All annotated instances of 'technology' need to be checked manually.

We are refining the idea of sense stability. For example, for the sense `a._noam_chomsky%1:18:00::` in WordNet 3.0, we have the synset 10896452-n, which contains two co-occurring senses. In WordNet 3.1, we have the related synset 10916204-n which contains the same and no new senses. Thus, we call this sense stable. An example where the senses diverge is for the sense `constrain%2:35:00::`. In WordNet 3.0, this sense is part of the synset 01301051-v with three other senses. In WordNet 3.1, this sense was moved to synset 01304044-v. Given that, all senses of 01301051-v (WordNet 3.0) became unstable. Considering all the challenges ahead, GlossTag 2022 is still based on WordNet 3.0.

6 Conclusion

In this paper, we presented the GlossTag 2022 release. The project is hosted in the https://github.com/own-pt/glosstag repository, and it will be updated in the following days. As put by (Miller et al., 1993), the semantic annotation of corpora helps improve both the coverage and the precision of the semantic resource being used in the annotation. This work is thus part of our effort in expanding and improving WordNet-like resources in an application-driven and domain-specific way.

Besides continuing the manual annotation, we plan to improve the annotation interface[16] and experiment with alternative WSD methods (McCord, 2004). Concerning the annotation tool, we intend to improve its performance and make it a wordnet editor, allowing the sense annotation to influence wordnet improvements. We also aim for a workflow with feedback between annotation and ERG analysis, one supporting the other. Additionally, we also intend to develop querying and visualization tools. Finally, we need to finish the migration to WordNet 3.1 before forking it from the Princeton official release (or further mapping to (McCrae et al., 2020)) for changes driven by the annotation.

Acknowledgments The authors would like to thank the support from Francis Bond and Dan Flickinger for all their support and valuable comments and suggestions.

References

Eneko Agirre, Oier Lopez De Lacalle, Aitor Soroa, and Informatika Fakultatea. 2009. Knowledge-based wsd and specific domains: Performing better than generic supervised wsd. In *IJCAI*, pages 1501–1506.

[16] https://github.com/own-pt/sensetion.el

Eneko Agirre, Oier López de Lacalle, and Aitor Soroa. 2018. The risk of sub-optimal use of open source nlp software: Ukb is inadvertently state-of-the-art in knowledge-based wsd.

Eneko Agirre and Aitor Soroa. 2009. Personalizing pagerank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 33–41. Association for Computational Linguistics.

Satanjeev Banerjee and Ted Pedersen. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Computational Linguistics and Intelligent Text Processing*, pages 136–145, Berlin, Heidelberg. Springer Berlin Heidelberg.

Pierpaolo Basile, Marco de Gemmis, Anna Lisa Gentile, Pasquale Lops, and Giovanni Semeraro. 2007. Uniba: Jigsaw algorithm for word sense disambiguation. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 398–401, Prague, Czech Republic. Association for Computational Linguistics.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, Roberto Navigli, et al. 2021. Recent trends in word sense disambiguation: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. International Joint Conference on Artificial Intelligence, Inc.

Peter Clark, Christiane Fellbaum, and Jerry Hobbs. 2008a. Using and extending wordnet to support question-answering. In *Proceedings of the 4th Global Wordnet Conference*, pages 111–119, Hungary.

Peter Clark, Christiane Fellbaum, Jerry R Hobbs, Phil Harrison, William R Murray, and John Thompson. 2008b. Augmenting wordnet for deep understanding of text. In *Semantics in Text Processing. STEP 2008 Conference Proceedings*, pages 45–57.

Ann Copestake. 2002. *Implementing typed feature structure grammars*, volume 110. CSLI publications Stanford.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A Sag. 2005. Minimal recursion semantics: An introduction. *Research on language and computation*, 3(2):281–332.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press.

Dan Flickinger. 2000. On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6(1):15–28.

Dan Flickinger. 2011. Accuracy v. robustness in grammar engineering. In Emily M. Bender and Jennifer E. Arnold, editors, *Language from a Cognitive Perspective: Grammar, Usage and Processing*, pages 31–50. CSLI Publications, Stanford, CA.

Dan Flickinger, Yi Zhang, and Valia Kordoni. 2012. Deepbank. a dynamically annotated treebank of the wall street journal. In *Proceedings of the 11th International Workshop on Treebanks and Linguistic Theories*, pages 85–96.

Michael Wayne Goodman. 2019. A python library for deep linguistic resources. In *2019 Pacific Neighborhood Consortium Annual Conference and Joint Meetings (PNC)*, pages 1–7. IEEE.

Sanda M. Harabagiu, George A. Miller, and Dan I. Moldovan. 1999. Wordnet 2: a morphologically and semantically enhanced resource. In *Proceedings of SIGLEX99: Standardizing Lexical Resources*, pages 1–8.

Michael C McCord. 2004. Word sense disambiguation in a slot grammar framework. Technical Report RC23397, IBM.

John Philip McCrae, Alexandre Rademaker, Ewa Rudnicka, and Francis Bond. 2020. English WordNet 2020: Improving and extending a WordNet for English using an open-source methodology. In *Proceedings of the LREC 2020 Workshop on Multimodal Wordnets (MMW2020)*, pages 14–19, Marseille, France. The European Language Resources Association (ELRA).

Rada Mihalcea and Dan I. Moldovan. 2001. extended wordnet: progress report. In *Proceedings of NAACL Workshop on WordNet and Other Lexical Resources*, pages 95–100.

George A Miller. 1993. The association of ideas. *The General Psychologist*, 29:69–74.

George A Miller and Christiane Fellbaum. 2007. WordNet then and now. *Language Resources and Evaluation*, 41(2):209–214.

George A Miller, Claudia Leacock, Randee Tengi, and Ross T Bunker. 1993. A semantic concordance. In *Proceedings of the workshop on Human Language Technology*, pages 303–308. Association for Computational Linguistics.

Dan Moldovan and Adrian Novischi. 2004. Word sense disambiguation of wordnet glosses. *Computer Speech & Language*, 18(3):301–317.

Ian Niles and Adam Pease. 2003. Linking lexicons and ontologies: Mapping wordnet to the suggested upper merged ontology. In *Ike*, pages 412–416.

Stephan Oepen. 2001. [incr tsdb()] — competence and performance laboratory. User manual. Technical report, Computational Linguistics, Saarland University, Saarbrücken, Germany. In preparation.

Stephan Oepen, Kristina Toutanova, Stuart M Shieber, Christopher D Manning, Dan Flickinger, and Thorsten Brants. 2002. The lingo redwoods treebank: Motivation and preliminary applications. In *COLING 2002: The 17th International Conference on Computational Linguistics: Project Notes*.

Woodley Packard. 2015. *Full forest treebanking*. Ph.D. thesis, University of Washington.

Adam Pease and Andrew Cheung. 2018. Toward a semantic concordancer. In *Proceedings of the 9th Global Wordnet Conference*, pages 97–104, Nanyang Technological University (NTU), Singapore. Global Wordnet Association.

Alexandre Rademaker, Bruno Cuconato, Alessandra Cid, Alexandre Tessarollo, and Henrique Andrade. 2019. Completing the Princeton annotated gloss corpus project. In *Proceedings of the 10th Global Wordnet Conference*, pages 378–386, Wroclaw, Poland. Global Wordnet Association.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. Ph.D. thesis, University of Pennsylvania.