Creation of a Balanced State-of-the-Art Multilayer Corpus for NLU

Normunds Gruzitis, Lauma Pretkalnina, Baiba Saulite,
Laura Rituma, Gunta Nespore-Berzkalne, Arturs Znotins and Peteris Paikens
University of Latvia, Institute of Mathematics and Computer Science, Raina blvd. 29, Riga, Latvia
Latvian Information Agency LETA, Marijas street 2, Riga, Latvia
{normunds.gruzitis,lauma.pretkalnina,baiba.valkovska}@lumii.lv, {peteris.paikens}@leta.lv

Abstract

This paper presents a work in progress to create a multilayered syntactically and semantically annotated text corpus for Latvian. The broad application area we address is natural language understanding (NLU), while more specific applications are abstractive text summarization and knowledge base population, which are required by the project industrial partner, Latvian information agency LETA, for the automation of various media monitoring processes. Both the multilayered corpus and the downstream applications are anchored in cross-lingual state-of-the-art representations: Universal Dependencies (UD), FrameNet, PropBank and Abstract Meaning Representation (AMR). In this paper, we particularly focus on the consecutive annotation of the treebank and framebank layers. We also draw links to the ultimate AMR layer and the auxiliary named entity and coreference annotation layers. Since we are aiming at a medium-sized still general-purpose corpus for a less-resourced language, an important aspect we consider is the variety and balance of the corpus in terms of genres, authors and lexical units.

Keywords: UD, FrameNet, PropBank, AMR, Latvian

1. Introduction

Natural language understanding (NLU) systems for abstractive text summarization, knowledge base population, and many other tasks rely, explicitly or implicitly, on full-stack syntactic and semantic parsing, including semantic role labeling, named entity recognition and linking, and coreference resolution. State-of-the-art parsers, in turn, rely on supervised machine learning which requires substantial language resources – syntactically and semantically annotated text corpora and extensive linked lexicons.

In the industry-oriented research project "Full Stack of Language Resources for Natural Language Understanding and Generation in Latvian", we are creating a balanced text corpus with multilayer annotations, adopting widely acknowledged and cross-lingually applicable representations: Universal Dependencies (UD) (Nivre et al., 2016), FrameNet (Fillmore et al., 2003), PropBank (Palmer et al., 2005) and Abstract Meaning Representation (AMR) (Banarescu et al., 2013).

Figure 1 outlines the inter-layer relationships. First, the UD representation is automatically derived from a more elaborated manually annotated hybrid dependency-constituency representation (see Section 3.). This also ensures that paragraphs, sentences and tokens are correctly and uniformly split, and represented in the standard CoNLL-U data format.[1] Thus, all the annotation layers can be afterwards merged based on the document, paragraph, sentence and token identifiers. Second, the FrameNet annotations are manually added, guided by the underlying UD annotations (see Section 5.1.). Third, the PropBank layer is automatically derived from the FrameNet and UD layers (see Section 5.2.). Fourth, the semi-automatic annotation of named entities, as well as named entity linking, is done in parallel to and independently from the annotation of semantic frames (see Section 4.). Coreference annotations are added afterwards, on top of the named entity annotations, consulting the underlying UD tree if necessary. These two auxiliary layers are required by the ultimate AMR layer. Fifth, draft AMR graphs are to be derived from the UD, PropBank, named entity and coreference annotation layers, with the potential to integrate the FrameNet frames and frame elements into the AMR graphs. As our preliminary experiments show, the semantically richer FrameNet annotations are also helpful in acquiring more accurate AMR graphs (see Section 5.3.). Despite some bootstrapping, all sentences at all layers are eventually checked and post-edited by experienced linguists.

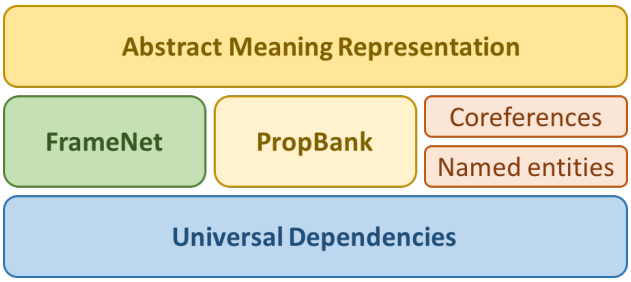

Figure 1: Annotation layers of the NLU corpus.

In this paper, we primarily focus on the consecutive and closely related creation of the treebank and framebank layers, as well as the auxiliary named entity layer. Also note that the above mentioned project addresses only verb frames. A spin-off project has been just launched to work on nominalizations.

The inspiration to create an integrated multilayer corpus comes from the OntoNotes corpus (Hovy et al., 2006) and the Groningen Meaning Bank (GMB) (Bos et al., 2017). The general difference from the OntoNotes approach is that we use the UD model at the treebank layer, and we annotate FrameNet frames in addition to the PropBank frames.

[1] http://universaldependencies.org/format.html

In fact, FrameNet is the primary frame-semantic representation in our approach. Another difference is that we aim at whole-sentence semantic annotation at the ultimate AMR layer. This in some sense is similar to the goal of the GMB project, but the meaning representation used in GMB, Discourse Representation Theory, is a deeper and more complex formalism that can be translated into first-order logic. For pragmatic reasons, we are following the more shallow and lossy AMR formalism. Our experience developing state-of-the-art systems for text-to-AMR parsing (Barzdins and Gosko, 2016) and AMR-to-text generation (Gruzitis et al., 2017), by combining machine learning and grammar engineering approaches, has convinced us that both FrameNet and AMR have a great potential to establish as powerful and complementary semantic interlinguas which can be furthermore strengthened and complemented by other multilingual representations.

Although this work focuses on Latvian, we believe that our experience and findings will be useful for the systematic creation of similar multilayered corpora for other less-resourced languages.

2. Balanced Data Set

Since we are aiming at a medium-sized corpus – around 10,000 sentences – it is crucially important to ensure that it is balanced in terms of text genres and writing styles, as well as lexical units.

A fundamental design decision is that the text unit in our multilayered corpus is an isolated paragraph. The corpus therefore consists of manually selected paragraphs from many different texts of various types.

Regarding genres, representative paragraphs are selected in different proportions from a balanced 10-million-word text corpus: around 60% come from various news sources, around 20% is fiction, around 10% are legal texts, around 5% is spoken language (transcripts), and the rest is miscellaneous. As for the lexical units, our goal is to cover around 1,000 most frequently occurring verbs, calculated from the 10-million-word corpus. Since the most frequent verbs tend to be also the most polysemous, we expect that the number of lexical units (verb senses w.r.t. FrameNet and PropBank frames) will be larger – at least 1,500 units. Nevertheless, the frequently used verbs should have proportionally as many example sentences as possible, while a single sentence often exemplifies the usage of more than one target verb.

Paragraphs to be annotated are therefore selected based on verbs they contain, not randomly (see Figure 2), and curators are constantly updated on the current balance or imbalance of the corpus w.r.t. genres and verb frequencies.

This approach has several benefits:

- Text units are not isolated sentences, allowing for coreference annotation and discourse analysis within the paragraph boundaries. Although coreference resolution is rather irrelevant for the FrameNet and PropBank annotation itself, and sentence-boundary coreferences are sufficient for single-sentence AMR annotation, paragraph-boundary coreference resolution is required by the downstream NLU applications, in combination with named entity linking, to resolve semantic roles expressed by anaphoric references, as well as to connect semantic frames and AMR graphs within the paragraph scope.

- Text units are small enough, allowing for flexible adjustments and paragraph-scrambled distribution of the data set. The adjustments can be made regarding genres, target words and word senses, types and density of named entities, etc.

- The diversity of text authors and writing styles is achieved implicitly – due to the large number of text units selected randomly w.r.t. the authors.

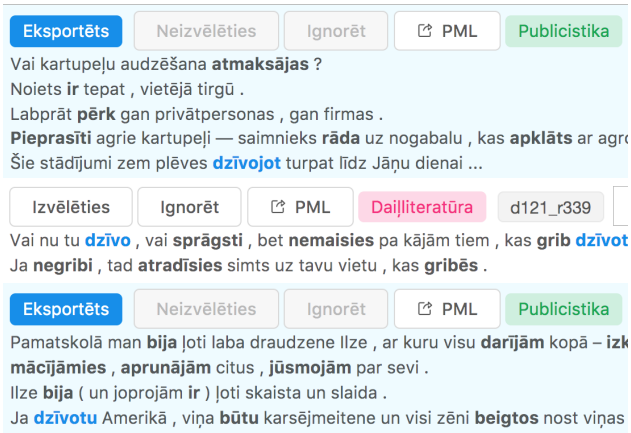

Figure 2: A screenshot of the paragraph selection tool. Candidate paragraphs are filtered by the given target verb 'dzīvot' ('to live', 'to reside' or 'to exist'); other verbs are also highlighted. Genre tags: journalism ('publicistika'), fiction ('daiļliteratūra'), etc.

Our decision about the data selection is justified also by the lessons learned in other treebanking and framebanking projects. For instance, Bick (2017) concludes that a sentence-randomized framebank (based on a sentence-randomized treebank) not only has a limited usage w.r.t. anaphoric relations and discourse analysis but also provides a limited coverage of lexical units.

3. Treebank

Latvian is an Indo-European language with rich morphology and relatively free word order, still many analytic forms are used as well. To capture the language-specific details and to accommodate the linguistic tradition on the one hand, and to meet the goal of the cross-lingual application on the other hand, the treebank annotation is provided in two complementary formalisms. First, the selected paragraphs are manually annotated according to our own hybrid grammar model developed with the linguistic tradition in mind. Second, the hybrid annotation is automatically converted to Universal Dependencies (UD) to achieve the cross-lingual compatibility, as well as to provide training data for efficient and robust parsers.

3.1. Annotation Using a Hybrid Dependency-Constituency Model

For the manual annotation, we use a hybrid dependency-constituency grammar formalism developed in the previous Latvian Treebank (LVTB) pilot project (Pretkalnina et al., 2011). LVTB uses a dependency based hybrid grammar model inspired by Tesnière's concept of syntactic nucleus (Barzdins et al., 2007; Nespore et al., 2010). The syntactic structure of the sentence is modeled by a dependency tree, however, dependencies can be linked not only between single words but also between words and more complex phrasal constructions. This allows for structural distinction between the dependant of a head word and the dependant of a whole phrase.

In LVTB, phrasal constructions are grouped into three classes: (i) coordination, (ii) punctuation mark attachment, and (iii) analytic constructions and other phrases with fixed of partially fixed word order, e.g. prepositional phrases, compound predicates, multi-word numerals, etc.

A sample sentence is shown in Figure 3, where the basic dependency links are brown, the constituency links of analytic forms are green, and the constituency links of the punctuation mark attachment are purple.

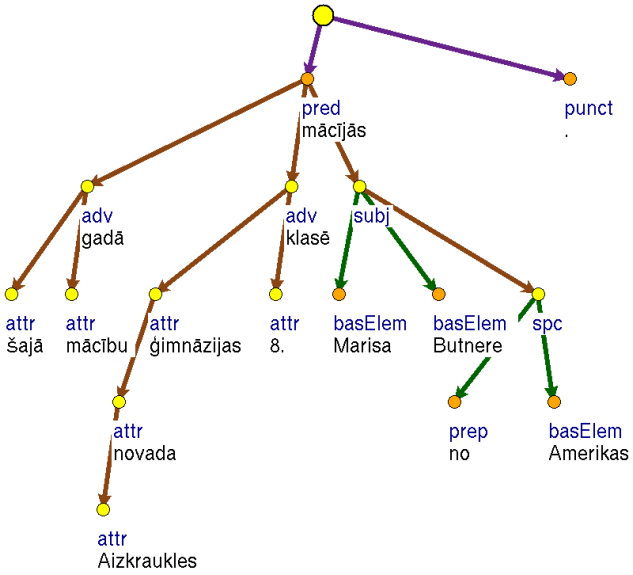

Figure 3: A sample sentence annotated according to our hybrid dependency-constituency grammar model. See Figure 6 and Table 2 for its linearization and enhanced UD representation.

The manual treebank annotation process is as follows:

1. The selected paragraph is automatically tokenized, lemmatized and morphologically tagged. A draft parse tree is automatically acquired using a basic dependency parser trained on LVTB, with limited rule-based post-conversion to the hybrid representation. While the morphological tagging is highly accurate (above 90%), the parser makes a lot of errors (well below 80%). Still, the post-editing of the draft trees is time saving.

2. The hybrid parse trees are post-edited using the TrEd tool (Pajas and Štěpánek, 2008) adapted for the hybrid model.

3. While most of the data is annotated by a single annotator, the perceived problem cases are regularly discussed among the annotators.

3.2. Conversion to Universal Dependencies

To obtain the UD representation from the hybrid representation, we have developed an automatic transformation procedure (Pretkalnina et al., 2016). The latest Latvian UD Treebank (LVUDTB) is periodically released though the UD repository.[2] It was also included in the CoNLL 2017 shared task on multilingual UD parsing (Zeman et al., 2017), being already classified as a relatively big treebank. Till UD v2.0, LVUDTB included only basic dependencies as specified by the UD guidelines. From UD v2.1 we are adding the enhanced dependencies. It is done automatically, extracting as much enhanced UD information as possible from the hybrid annotations and leaving some inaccuracies when there is not enough information available.

The transformation procedure is based on heuristics and on analytic comparison of the two representations. While most of the information necessary for the UD model can be derived from LVTB straightforwardly, it lacks some of the necessary distinctions. Most notably, LVTB does not indicate if a complement takes its own subject (*xcomp* and *ccomp* in UD). Also, since there are no articles or definite suffixes used in Latvian, the distinction between *DET* and *PRON*, and *det* and *nmod* is made heuristically by analyzing the tree structure and the pronoun agreement.

Since the enhanced dependencies were introduced quite after we started to contribute the UD treebank, our transformation was first build to construct the basic UD trees. It was later modified to produce also the enhanced UD graphs from the basic trees, consulting the original LVTB data as well. However, this leads to some inaccuracies, and we plan to rewrite the transformation so that the enhanced graph is built first as it closer follows the original hybrid representation, and then it is reduced to the basic dependencies.

The hybrid to UD transformation currently consists of the following steps:

1. **Tokenization**. The LVTB tokenization guidelines are tweaked to match the UD guidelines: abbreviations and numbers with spaces as group separators are treated as single tokens (e.g. 'u.c.', '1 000 000').

2. **Lemmatization**. Since UD has no strict guidelines on lemmatization, lemmas are taken as is from LVTB.

3. **POS tags and morphological features**. Most of the POS tags can be obtained directly from the LVTB tagset, but some tags (e.g. *DET*) are decided by analyzing the syntactic structure. The LVTB tagset ensures a good coverage of the morphological features as well.

[2] http://universaldependencies.org/#download

4. **Basic dependencies and ellipsis**. Basic dependencies are obtained via a bottom-up traversal through the original tree. Conveniently, phrasal constructions in the hybrid trees correspond to isolated subtrees in the UD trees. Thus, every phrase can be transformed by considering only its constituents and its parent (or the direct ancestor if the parent is a coordinated constituent). Basic dependencies between single words are used as is. Any dependency to/from a phrase can be transformed to a dependency to/from the root of the subtree representing the corresponding phrase.

   The transformation of roles is more complicated, as the relationship between the LVTB roles and the UD roles is mostly many-to-many. This is because LVTB uses a more semantically oriented role set than UD. For instance, while LVTB distinguishes between attributes (adjectives or nouns) and adverbial modifiers (adverbs or nouns), UD distinguishes between nominal modifiers, numeric modifiers and adverbs.

   Due to the fact that ellipsis are represented by empty nodes in LVTB, it is convenient to build an enhanced dependency "backbone" already in the same step. The backbone tree contains the relevant arcs from the basic dependency tree, the ellipsis nodes, and the links connecting the ellipsis nodes to the tree.

5. **Other enhanced dependencies**. Both coordination and compound predicates are annotated as phrasal constructions in LVTB. Since dependants of a phrasal construction (as a whole) are annotated structurally different from the dependants of head constituents of the phrase, obtaining links related to the coordination propagation and raised (controlled) subjects is relatively straightforward. Similarly, it is possible to obtain the case information from either the morphological tag of the preposition, or from the non-terminal phrase representing the prepositional construction.

For the enhanced dependencies, the UD guidelines suggest adding the following types of annotations:

- Null nodes for elided predicates – added.
- Additional subject relations for control and raising constructions – added.
- Propagation of conjuncts – added with rare inaccuracies. The current implementation can fail to obtain a correct enhanced dependency role for the second and further conjuncts, if the conjuncts belong to different parts-of-speech. This problem will be solved when the transformation is rewritten as planned.
- Modifier labels that represent the preposition, or other case-marking information – partially added. Case information is available for prepositional constructions and some types of subordinated clauses.
- Coreferences in relative clause constructions – not available. LVTB contains no information on coreferences. Coreference annotations will be eventually available from the coreference layer (see Figure 1 and Section 4.2.).

The necessity to transform the LVTB data to the UD representation also helps to review the hybrid annotation scheme from a different perspective. To acquire more accurate UD transformation results, we have improved the LVTB annotation guidelines, e.g. for more detailed annotation of compound predicates, and for revised annotation of comparison constructions. Overall, the hybrid annotation model contains more information compared not only to the basic but also to the enhanced UD dependencies, even though it lacks some of the distinctions which UD makes.

4. Named Entities

Named entities are essential for most NLU tasks, since they link the textual content to the real world, making the extracted facts (frames) meaningful for a knowledge base population task, for instance. From the multilayer corpus perspective, the AMR annotation heavily relies on named entity recognition and linking (as illustrated in Figure 7), and on within-sentence coreference resolution, using the reentrancy representation. This allows for connecting individual AMR graphs and subgraphs into a wider context.

4.1. Named Entity Annotation

In our corpus, we are using the following set of named entity categories: *person*, *organization*, *geopolitical entity* (GPE), *location*, *product*, *time* (relative or absolute date, time, or duration), *event*, and *entity* (entities of other categories that occur rarely but could be considered in future). We mostly follow the MUC-7 annotation guidelines (Chinchor, 1998) which we have extended for compatibility with the top-level named entity categories specified by the AMR guidelines.[3] For serialization of the named entity layer, we use a version of the CoNLL-2003 data format[4] on top of CoNLL-U (see Table 1).

In practice, many named entities contain references to other named entities. We do annotate nested (hierarchical) entities (see Figure 4), which enables us to exploit the annotations of either the outer or inner entities, or both, when annotating coreferences or deriving AMR graphs. This would not be possible with a flat annotation scheme. Furthermore, hierarchical annotations not only allow for the development of an automatic hierarchical named entity recognizer (NER) but also provide more training data for the development of a flat NER.

4.2. Named Entity Linking and Coreference Resolution

We approach coreference annotation in a pragmatic way, focusing on precision and annotating coreferences only within the paragraph boundaries. This allows to annotate a lot of various text units, and it makes the annotation process easier and less error prone. We are mostly following the English coreference guidelines used in the OntoNotes project.[5] We annotate pronominal and nominal noun phrases referring to real-word entities, and non-specific mentions if they are antecedents of pronouns. Bridging

[3] https://github.com/amrisi/amr-guidelines

[4] https://www.clips.uantwerpen.be/conll2003/ner/

[5] https://catalog.ldc.upenn.edu/docs/LDC2013T19/

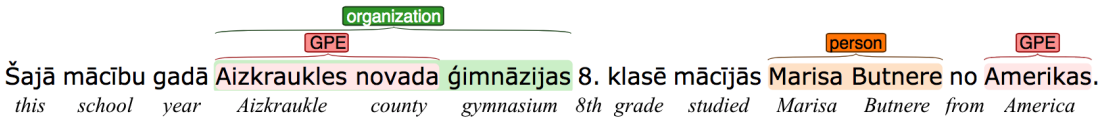

Figure 4: Hierarchical named entity annotation and wikification (behind the scenes) using WebAnno.

Table 1: A data format used to serialize the named entity layer of the corpus: a version of CoNLL-2003 based on CoNLL-U. Because of space restrictions, several less relevant CoNLL-U fields are excluded from the table. In addition to the CoNLL-2003 fields, we include extra fields for the inner entities and for the wikification of both outer and inner entities.

ID	FORM	LEMMA	UPOSTAG	XPOSTAG	**BIOTAG$_1$**	**BIOTAG$_2$**	**WIKI$_1$**	**WIKI$_2$**
1	Šajā	šis	DET	pd0fsln	O	_	_	_
2	mācību	mācība	NOUN	ncfpg4	O	_	_	_
3	gadā	gads	NOUN	ncmsl1	O	_	_	_
4	Aizkraukles	Aizkraukle	PROPN	npfsg5	B-organization	B-GPE	_	lv:Aizkraukles_novads
5	novada	novads	NOUN	ncmsg1	I-organization	I-GPE	_	_
6	ģimnāzijas	ģimnāzija	NOUN	ncfsg4	I-organization	_	_	_
7	8.	8.	ADJ	xo	O	_	_	_
8	klasē	klase	NOUN	ncfsl5	O	_	_	_
9	mācījās	mācīties	VERB	vmyisi330an	O	_	_	_
10	Marisa	Marisa	PROPN	npfsn4	B-person	_	_	_
11	Butnere	Butnere	PROPN	npfsn5	I-person	_	_	_
12	no	no	ADP	spsg	O	_	_	_
13	Amerikas	Amerika	PROPN	npfsg4	B-GPE	_	en:United_States	_
14	.	.	PUNCT	zs	O	_	_	_

relations, discontinuous expressions, split antecedents and zero anaphora are ignored.

In addition to annotating the named entity spans and categories, we also specify a corresponding Wikipedia identifier (URI) if one exists. First, we look for a corresponding article in the English Wikipedia. Second, if no article is found, we look for a corresponding article in the Latvian Wikipedia (see the different namespace prefixes in Table 1). For training a named entity linker (NEL), such corpus would be considered a very small one, but it will be helpful for evaluating a NEL. For this reason, we have specially included text units mentioning different persons with the same name, for instance. The manually verified Wikipedia identifiers are also useful when generating draft AMR graphs (see Section 5.3.).

5. Semantic Frames

The annotation of PropBank frames is relatively more simple if compared to FrameNet, since PropBank frames are less abstract, and their semantic roles directly follow from the syntactic verb argument structure. Creating the Latvian framebank, however, we start with annotating FrameNet frames, and we are deriving the PropBank annotations automatically from the FrameNet and UD annotations.

5.1. UD-Based Annotation of FrameNet

The creation of the FrameNet annotation layer is as follows. Paragraphs for which the manual treebank annotation is finalized and which have been successfully converted from the hybrid grammar to the UD representation are stored in a separate repository. While treebank, named entity and coreference annotation is done paragraph by paragraph, this is not a productive workflow for annotating semantic frames, especially in case of the highly abstract FrameNet frames. Instead, a concordance view is required, so that the linguist can focus on a target verb and its different senses (frames), without constantly switching among different sets of frames. This also improves the annotation consistency.

To provide such environment, we automatically extract all UD-annotated sentences from the finalized paragraphs containing the requested target verb, and we store the result in a separate temporary CoNLL-U file. When more paragraphs are finalized at the UD layer, they are considered in the next concordance queries. The acquired concordance files are imported in the WebAnno platform (Eckart de Castilho et al., 2016) which we have specifically configured for the FrameNet annotation. When the annotation is done, the finalized concordances are exported from WebAnno and are eventually reorganized back into paragraphs.

Figure 5 illustrates a sample concordance with the resulting FrameNet frame and frame element (FE) annotations (the UD annotations are hidden for the sake of simplicity). The actual annotation, however, is done on top of the UD layer, as illustrated in Figure 6 and Table 2. Such approach has a significant consequence: FEs are not annotated as spans of text – only the head word of a UD subtree is annotated; the whole span can be expanded automatically by traversing the respective subtree. This not only makes the annotation process more simple and the annotations more consistent, but it also facilitates the learning of automatic semantic role labeling, since it is easier to identify the syntactic head of a FE than a span of a string. Still, most FrameNet corpora are annotated in terms of spans, relying on syntactic parsing as a post-processing step.

The creation of the FrameNet layer is described in more detail by Gruzitis et al. (2018).

Table 2: A data format used to serialize the FrameNet layer of the corpus: a version of CoNLL-2009 based on CoNLL-U. Because of space restrictions, several CoNLL-U fields are excluded from the table.

ID	FORM	LEMMA	UPOSTAG	XPOSTAG	DEPS	**FILLPRED**	**PRED**	**APRED$_1$**
1	Šajā	šis	DET	pd0fsln	3:det	_	_	_
2	mācību	mācība	NOUN	ncfpg4	3:nmod:gen	_	_	_
3	gadā	gads	NOUN	ncmsl1	9:obl:loc	_	_	Time
4	Aizkraukles	Aizkraukle	PROPN	npfsg5	5:nmod:gen	_	_	_
5	novada	novads	NOUN	ncmsg1	6:nmod:gen	_	_	_
6	ģimnāzijas	ģimnāzija	NOUN	ncfsg4	8:nmod:gen	_	_	Institution
7	8.	8.	ADJ	xo	8:amod	_	_	_
8	klasē	klase	NOUN	ncfsl5	9:obl:loc	_	_	Level
9	mācījās	**mācīties**	VERB	vmyisi330an	0:root	Y	**Education_teaching**	_
10	Marisa	Marisa	PROPN	npfsn4	9:nsubj	_	_	Student
11	Butnere	Butnere	PROPN	npfsn5	10:flat:name	_	_	_
12	no	no	ADP	spsg	13:case	_	_	_
13	Amerikas	Amerika	PROPN	npfsg4	10:nmod:no	_	_	_
14	.	.	PUNCT	zs	9:punct	_	_	_

Table 3: A data format used to serialize the PropBank layer of the corpus: a version of CoNLL-2009 based on CoNLL-U. The PropBank annotations are semi-automatically derived from the FrameNet layer (see Table 2).

ID	FORM	LEMMA	UPOSTAG	XPOSTAG	DEPS	**FILLPRED**	**PRED**	**APRED$_1$**
1	Šajā	šis	DET	pd0fsln	3:det	_	_	_
2	mācību	mācība	NOUN	ncfpg4	3:nmod:gen	_	_	_
3	gadā	gads	NOUN	ncmsl1	9:obl:loc	_	_	AM-TMP
4	Aizkraukles	Aizkraukle	PROPN	npfsg5	5:nmod:gen	_	_	_
5	novada	novads	NOUN	ncmsg1	6:nmod:gen	_	_	_
6	ģimnāzijas	ģimnāzija	NOUN	ncfsg4	8:nmod:gen	_	_	_
7	8.	8.	ADJ	xo	8:amod	_	_	_
8	klasē	klase	NOUN	ncfsl5	9:obl:loc	_	_	AM-LOC
9	mācījās	**mācīties**	VERB	vmyisi330an	0:root	Y	**study.01**	_
10	Marisa	Marisa	PROPN	npfsn4	9:nsubj	_	_	A0
11	Butnere	Butnere	PROPN	npfsn5	10:flat:name	_	_	_
12	no	no	ADP	spsg	13:case	_	_	_
13	Amerikas	Amerika	PROPN	npfsg4	10:nmod:no	_	_	_
14	.	.	PUNCT	zs	9:punct	_	_	_

Figure 5: WebAnno screenshot: FrameNet-annotated occurrences of the target verb 'dzīvot' ('to live/reside/exist').

5.2. Conversion to PropBank

The semantic roles in PropBank are much more robust compared to FrameNet, and the overall PropBank annotation systematically follows the verb argument structure. Therefore the PropBank layer can be semi-automatically derived from the FrameNet and UD layers, by providing a mapping configuration from lexical units in FrameNet to PropBank frames (see Table 4), and a mapping configuration from FrameNet frame elements to PropBank semantic roles for the given pair of FrameNet and PropBank frames (see Table 5). We are building on the previous work on SemLink (Palmer, 2009) and Predicate Matrix (Lopez de Lacalle et al., 2016). We use the provided mapping between English FrameNet and English PropBank as a draft configuration. The linguistically intensive manual task is to map the lexical units from Latvian FrameNet to the semantic frames of English PropBank. The rest is a straightforward automation. Table 3 illustrates a PropBank-annotated sentence, where the annotation has been derived from the FrameNet and UD layers (Table 2). Note that the FrameNet annotation is semantically richer, as well as it might be non-projective w.r.t. the underlying UD tree. In the given example, the frame element *Institution* is not transfered to the PropBank layer because it is not a syntactic argument of the target verb.

5.3. Generation of AMR

So far we have conducted only limited preliminary experiments on the AMR annotation based on the underlying UD,

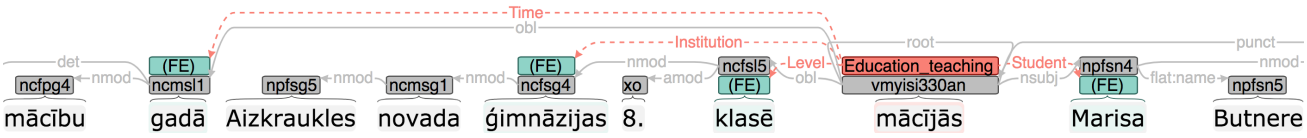

Figure 6: FrameNet annotation on top of a UD tree. Only head nodes are selected while annotating frame elements (FE). The FE spans can be acquired automatically by traversing the respective subtrees: [.. *school year*]$_{\text{Time}}$ [*Aizkraukle county gymnasium*]$_{\text{Institution}}$ [*8th grade*]$_{\text{Level}}$ ***studied***$_{\text{EDUCATION\_TEACHING}}$ [*Marisa Butnere* ..]$_{\text{Student}}$.

Table 4: Mapping from FrameNet frames to PropBank frames, taking the lexical units into account.

LEMMA	UPOSTAG	PRED$_{\text{FrameNet}}$	**PRED$_{\text{PropBank}}$**
mācīties	VERB	Education_teaching	study.01
mācīt	VERB	Education_teaching	teach.01
mācība	NOUN	Education_teaching	training.01
dzīvot	VERB	Residence	reside.01

Table 5: Mapping from FrameNet frame elements to PropBank semantic roles, taking the UD dependency relations into account.

PRED$_{\text{FN}}$	APRED$_{\text{FN}}$	DEPREL	PRED$_{\text{PB}}$	**APRED$_{\text{PB}}$**
Education_teaching	Student	nsubj	study.01	A0
Education_teaching	Student	obj	teach.01	A2
Education_teaching	Student	iobj	teach.01	A2
Education_teaching	Subject	obj	study.01	A1
Education_teaching	Subject	obj	teach.01	A1
Education_teaching	Teacher	obl	study.01	A2
Education_teaching	Teacher	nsubj	teach.01	A0
Education_teaching	Institution	obl	study.01	AM-LOC
Education_teaching	Institution	obl	teach.01	AM-LOC
Education_teaching	Level	obl	study.01	AM-LOC
Education_teaching	Time	obl	study.01	AM-TMP
Education_teaching	Time	obl	teach.01	AM-TMP

PropBank and FrameNet layers, as well as the auxiliary named entity and coreference layers. However, it seems feasible to systematically generate draft AMR annotations for manual post-editing, thus, boosting the productivity and acquiring more consistent AMRs. An illustrative example is given in Figure 7.

The auxiliary layers were not part of the initial work plan. The information they convey was planned to be added during the AMR annotation – only as part of the AMR graphs. However, we have introduced them as separate layers to facilitate the semi-automatic generation of AMR representations in addition to the utility of these auxiliary layers per se in practical applications.

Note that although FrameNet frames and frame elements are not explicitly integrated in AMR, FrameNet annotations still support the systematic construction of AMR graphs. In the above example, the *x8* instance is linked by the *location* relation instead of a more general relation thanks to the FrameNet frame element *Institution* which is not covered at the PropBank layer.

6. Conclusion

The consecutive treebank and framebank annotation workflow has turned out to be very productive and mutually beneficial. The dependency tree facilitates the annotation of semantic frames and roles, while the frame semantic analysis of verb valency often unveils some inconsistencies or bugs in the dependency annotation or in the morphological tagging. These issues are immediately fixed in the treebank. Similarly, the annotation of other layers helps to notice other kinds of bugs and inconsistencies in the underlying layers.

```
(x1 / study-01
  :ARG0 (x2 / person
    :name (x3 / name
      :op1 "Marisa" :op2 "Butnere")
    :source (x4 / GPE
      :name (x5 / name :op1 "Amerika")
      :wiki "United_States"))
  :location (x6 / klase
    :ord (x7 / ordinal-entity :value 8)
    :location (x8 / organization
      :name (x9 / name
        :op1 "Aizkraukles"
        :op2 "novada"
        :op3 "ģimnāzija")))
  :time (x10 / gads
    :mod (x11 / šis)
    :mod (x12 / mācība)))
```

Figure 7: A draft AMR graph to be generated, consulting all the underlying annotation layers (see Table 3, 2 and 1).

While the FrameNet annotation helps to eliminate certain types of syntactic annotation errors, the automatic conversion from the hybrid dependency-constituency grammar to the UD representation helps to unveil and fix other types of inconsistencies and bugs. The decision to use the more complex and rich hybrid representation for the manual annotation of the treebank has paid off even more: the enhanced UD dependencies can be derived without any additional annotation efforts, and there is more information to derive if necessary.

Although we are primarily focusing on verbs and verb frames that are the primary units making a sentence (or clause), the syntactic and semantic arguments in the sentence are not controlled only by verbal predicates. Deverbal derivatives, i.e. frame-evoking nouns and adjectives are often used as well, and they usually preserve the syntactic and semantic features of a verb, such as transitivity or the capability of taking nominal or verbal complements. Deverbal derivatives are inevitable in real-world full-text se-

mantic parsing, therefore we address these phenomena in a spin-off project which has been just launched and is building on the same data set.

The multilayer corpus is being gradually released on GitHub under the CC BY-NC-SA 4.0 licence.[6]

7. Acknowledgements

This work has received financial support from the European Regional Development Fund under the grant agreements No. 1.1.1.1/16/A/219 and No. 1.1.1.2/VIAA/1/16/188.

8. Bibliographical References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract Meaning Representation for Sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria.

Barzdins, G. and Gosko, D. (2016). RIGA at SemEval-2016 Task 8: Impact of Smatch extensions and character-level neural translation on AMR parsing accuracy. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, pages 1143–1147, San Diego, California.

Barzdins, G., Gruzitis, N., Nespore, G., and Saulite, B. (2007). Dependency-based hybrid model of syntactic analysis for the languages with a rather free word order. In *Proceedings of the 16th Nordic Conference of Computational Linguistics*, pages 13–20, Tartu, Estonia.

Bick, E. (2017). From Treebank to Propbank: A Semantic-Role and VerbNet Corpus for Danish. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 202–210, Gothenburg, Sweden.

Bos, J., Basile, V., Evang, K., Venhuizen, N., and Bjerva, J. (2017). The Groningen Meaning Bank. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, volume 2, pages 463–496. Springer.

Chinchor, N. (1998). Appendix E: MUC-7 Named Entity Task Definition. In *Proceedings of th 7th Message Understanding Conference (MUC)*, Fairfax, Virginia.

Eckart de Castilho, R., Mújdricza-Maydt, E., Yimam, S. M., Hartmann, S., Gurevych, I., Frank, A., and Biemann, C. (2016). A Web-based Tool for the Integrated Annotation of Semantic and Syntactic Structures. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities*, pages 76–84, Osaka, Japan.

Fillmore, C. J., Johnson, C. R., and Petruck, M. R. (2003). Background to FrameNet. *International Journal of Lexicography*, 16(3):235–250.

Gruzitis, N., Gosko, D., and Barzdins, G. (2017). RIGOTRIO at SemEval-2017 Task 9: Combining machine learning and grammar engineering for AMR parsing and generation. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, pages 924–928, Vancouver, Canada.

Gruzitis, N., Nespore-Berzkalne, G., and Saulite, B. (2018). Creation of Latvian FrameNet based on Universal Dependencies. In *Proceedings of the 11th International Conference on Language Resources and Evaluation*, Miyazaki, Japan.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: The 90% Solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA.

Lopez de Lacalle, M., Laparra, E., Aldabe, I., and Rigau, G. (2016). A Multilingual Predicate Matrix. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 2662–2668, Portoroz, Slovenia.

Nespore, G., Saulite, B., Barzdins, G., and Gruzitis, N. (2010). Comparison of the SemTi-Kamols and Tesniere's dependency grammars. In *Human Language Technologies – The Baltic Perspective*, volume 219 of *Frontiers in Artificial Intelligence and Applications*, pages 233–240. IOS Press.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 1659–1666.

Pajas, P. and Štěpánek, J. (2008). Recent advances in a feature-rich framework for treebank annotation. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 673–680, Manchester, UK.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Palmer, M. (2009). SemLink: Linking PropBank, VerbNet and FrameNet. In *Proceedings of the Generative Lexicon Conference*, pages 9–15, Pisa, Italy.

Pretkalnina, L., Nespore, G., Levane-Petrova, K., and Saulite, B. (2011). A Prague Markup Language profile for the SemTi-Kamols grammar model. In *Proceedings of the 18th Nordic Conference of Computational Linguistics*, pages 303–306, Riga, Latvia.

Pretkalnina, L., Rituma, L., and Saulite, B. (2016). Universal Dependency treebank for Latvian: A pilot. In *Human Language Technologies – The Baltic Perspective*, volume 289 of *Frontiers in Artificial Intelligence and Applications*, pages 136–143. IOS Press.

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., et al. (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.

[6] https://github.com/LUMII-AILab/FullStack