From LFG to Enhanced Universal Dependencies

Adam Przepiórkowski and Agnieszka Patejuk

{adamp,aep}@ipipan.waw.pl

Introduction Universal Dependencies (UD; Nivre *et al.* 2016) has recently become a *de facto* standard as a dependency representation used in Natural Language Processing (NLP). As most syntactic processing in NLP involves dependency structures, it is safe to say that it is becoming a standard for syntactic processing at large. Version 2.1 of UD treebank collection, released in mid-November 2017, consists of 102 treebanks representing 60 typologically diverse languages (`http://universaldependencies.org`).

The aim of this paper is to discuss linguistic issues in the process of converting LFG representations to dependency structures following the UD standard, specifically, its enhanced version 2. Conversion of LFG structures to dependency structures is not a new task (cf., e.g., Øvrelid *et al.* 2009 and Çetinoğlu *et al.* 2010), but apparently no attempt has been made so far to take advantage of the enhanced possibilities made available in UD v.2, and – with the notable exception of Meurer 2017 – previous attempts are only mentioned or very roughly outlined in the literature. Moreover, there is some disagreement about which syntactic level of representation – c-structure or f-structure – is the most natural basis for constructing dependency representations. While f-structure seems to be a natural candidate, Meurer 2017 sketches a conversion procedure based mainly on c-structure and consisting in step-wise transformations of the constituency tree into a dependency tree.

The approach presented here follows the more standard observation that f-structures provide the basis for dependency relations. Of course, c-structures cannot be ignored, as they provide the actual forms in the sentence (f-structure PRED values usually use lemmata as functors) and their order. We show that this information, and the ϕ correspondence between c-structure preterminals and f-structure components, is sufficient to perform the conversion, i.e., that the actual constituent structure may be completely ignored.

The empirical basis for the conversion is the manually disambiguated LFG parsebank of Polish consisting of over 17,000 sentences (almost 131,000 tokens). Since this is a parsebank, it only contains analyses successfully provided by the LFG parser of Polish (Patejuk and Przepiórkowski 2012) and selected by human annotators as correct. While this constrains the number and kinds of constructions present in the corpus, the underlying LFG grammar of Polish is currently one of the largest implemented LFG grammars, and it includes a comprehensive analysis of various kinds of coordination and its interaction with other phenomena, so there is no shortage of sentences which pose potential difficulties for the conversion.

From LFG to LFG-like dependencies Conversion is performed in two stages: from LFG structures to initial dependency structures directly corresponding to LFG representations, and from such initial dependency structures to final enhanced UD representations. The sole difficulty of the first stage stems from the fact that often multiple tokens map to the same functional structure, as exemplified in (1) and (3): of the 5 feature structures, 2 do not correspond to any token (they represent *pro*-dropped subjects), so the 9 tokens in the sentence (including punctuation) map to just the remaining 3 f-structures.

There are two problems to be solved: deciding which of the co-heads – tokens mapping to the same f-structure – is the true head, and deciding on the dependency labels from this true head to the other co-heads. As a result of solving the first problem, the backbone dependencies may be established on the basis of f-structure, as shown in (2). The true head is chosen mainly on the basis of part-of-speech information: in this case, the verb *daję* 'give' wins the competition with the two (sentence-initial and sentence-final) punctuation marks, and the verb *gniewam* 'be angry' wins with the negative marker *nie*, the inherent reflexive marker *się*, the complementiser *że* and the comma. In the case of coordination, the conjunction is selected as the head and set membership is translated as CONJ dependency. A complication of this step is the existence of asyndetic coordination, where the conjunction is often expressed with a comma: as there may be other commas mapping to the same coordinate f-structure, the trick is to choose the right one. On the other hand, the adopted solution of the second problem (dependency labels from true head to co-heads) is trivial: dependencies to co-heads are labelled with the names of the preterminals of these co-heads in c-structure. The result of this first stage of conversion is given in (4).

From LFG-like dependencies to enhanced Universal Dependencies In the simplest – but rare – case, in order to arrive at the final UD representation, it is sufficient to rename LFG dependency labels to UD labels, as shown in (5). However, in the usual case initial dependency structures must also be rearranged, for two main reasons. First, UD adopts the principle of the primacy of content words – rather than functional words – as heads. This means that, unlike in LFG representations, prepositional phrases are headed by nouns (even in the case of semantic prepositions, which contribute a PRED value), numeral phrases are headed by nouns (even though, for Polish, there are good arguments to the contrary), and auxiliaries and copulas are always dependents, rather than heads. This is not only a matter of reversing single dependencies: all dependencies originally targeting the functional head must now target the content head, and all outgoing dependencies from the functional head must now originate in the content head. Second, UD adopts a representation of coordination in which it is the first conjunct – not the conjunction – that is the head. All other conjuncts are direct dependents of this head and the conjunction is a dependent of the conjunct to its right.[1]

Both kinds of structural modification are illustrated with example (6) and its LFG-like and UD structures in (7)–(8). This example involves asyndetic coordination of a passive participial phrase, *wysoko zapięta pod szyję* 'buttoned up high to the neck', and a predicative adjectival phrase, *wysmukła jak kwiat* 'lean as a flower'. In the original LFG representation and the LFG-like dependency structure (7), the two conjuncts are dependents of the conjunction (here, the comma), and the whole coordinate structure is an XCOMP-PRED argument of *jest* 'be', which is the root of the whole sentence. On the other hand, in the UD representation (see the upper part of (8)), the coordinate structure is headed by the first conjunct, the passive participle, and *jest* 'be' is its dependent, so the passive participle is promoted to the status of the root of the sentence. Moreover, the structure of the two prepositional phrases, *pod szyję* 'up to the neck' and *jak kwiat* 'as a flower', is rearranged. Note that the relation between the passive participle *zapięta* and *jest* is `aux:pass`, so the whole sentence is represented as a passive construction. But this is contingent on the linear order of the two conjuncts: were it opposite, the main relation would be that from the adjective

[1] So this representation differs from that of Mel'čuk 1988, where the first conjunct is the head but other conjuncts and the conjunction form a chain.

wysmukła to *jest*, so it would be labelled as `cop`, and the whole sentence would in effect be analysed as a copular construction.[2] On the other hand, this dual nature of *jest* 'be' is expressed in the enhanced dependencies (see the lower part of (8)), which include one more dependency: the edge from the adjective *wysmukła* to *jest*, marked as `cop`. Similar secondary edges may be used in the enhanced representation to express control.[3]

Comparison Dependency representations, usually assumed to be simple trees rather than arbitrary directed graphs, seem to be much less expressive than LFG syntactic representations. However, as already shown in (8), enhanced UD representations are also directed graphs and may represent various kinds of re-entrancies characteristic of LFG (and HPSG) structures. This is more robustly shown in (9)–(11), where the two asyndetically coordinated verbs share three dependents: the subject, the object and an adjunct. There is no information loss between the LFG syntactic representation – schematically given in (10) – and the enhanced UD representation in (11).

However, it is still not the case that the two representations are similarly expressive. The full paper discusses, on the basis of the 17,000-sentence UD treebank of Polish, the extent to which information is lost during conversion from LFG to enhanced UD. The following are among the identified cases of information loss: **1.** There is no way to represent the difference between raising and control, or between functional control and obligatory anaphoric control. One way of dealing with this problem could consist in adding appropriate subtypes to the `xcomp` relation. **2.** There is no way to represent *pro*-dropped arguments; this results in very different representations of secondary predicates depending on whether they predicate of overt or dropped constituents. The current version 2 of enhanced UD allows for null nodes, but only in order to represent elided predicates, not dropped dependents of predicates. So, a possible extension of current UD guidelines is straightforward. **3.** It is illegal at the moment, even in enhanced dependencies, to have two different edges from token A to token B. The need for such a representation arises in those – admittedly rare – cases when the multi-functional reflexive marker *się* plays two roles at the same time (Patejuk and Przepiórkowski 2015), e.g., being a marker of an inherently reflexive verb (`expl:pv`, as in (5) and (11)) and being a part of an impersonal construction (`expl:impers`). A treebank example exhibiting this problem is (12): the first *się*, in *uczestniczyło się* 'one participated', is purely impersonal, and the second *się*, in *modliło się* 'one prayed', is impersonal and also an inherent part of the verb MODLIĆ SIĘ 'pray'.

(12) W Laskach w liturgii uczestniczyło się przez cały dzień i modliło się wszędzie.
in Laski in liturgy participate.IMPS RM for whole day and pray.IMPS RM everywhere
'In Laski, one took part in the liturgy for the whole day and one prayed everywhere.'

It seems that the ban on multiple edges could be lifted in the enhanced UD without any ill consequences. **4.** There is no way to distinguish between embedded coordination, with the first conjunct itself being a coordinate structure, and flat coordination. This problem is known to the UD community, but it is not clear how it could be solved. **5.** Some interactions of coordination and the way other dependencies are defined gives somewhat incoherent results. To give a simple English example, the sequence *president George W. Bush* would be analysed via the `flat` dependencies from *president* to the other three tokens (rather than as a chain), but in the case of *presidents George W. Bush and Barack Obama*, a partial chain would have to be formed – from *president* to *George* and from *George* to *W.* (and to *Bush*), so that coordination is also reasonably represented (as a `conj` dependency from *George* to *Barack*, etc.).

Conclusion The Polish enhanced UD treebank resulting from the conversion from LFG structures will be officially released with the next – 2.2 – version of UD treebank collection on 15th April 2018. Judging from the current state of the art, it may turn out to be the largest UD treebank taking substantial advantage of enhanced dependencies.

In the last – mid-November 2017 – release of UD treebanks only 2 (Finnish and Latvian) out of 102 made non-trivial use of enhanced dependencies; apparently dependency treebanks with structures going beyond basic trees are rare. The exercise described here shows that it is relatively easy to convert an LFG structure bank into a full-blown enhanced UD representation. Surprisingly little information is lost in the conversion from LFG to enhanced UD and – as discussed above – some of the deficiencies of current UD are easy to rectify, and other surface rarely. Given the close affinity between LFG and UD (e.g., their similar reliance on grammatical functions, which sets them apart from HPSG, etc.), LFG is uniquely poised to play an important role in the further development of the UD standard in general and treebanks for specific languages in particular.

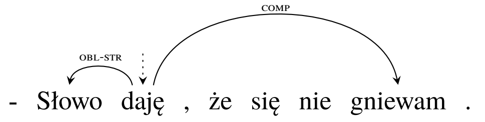
References ► Mel'čuk, I. (1988). *Dependency Syntax: Theory and Practice*. The SUNY Press. ► Meurer, P. (2017). From LFG structures to dependency relations. In V. Rosén and K. D. Smedt, eds., *The Very Model of a Modern Linguist*, pp. 183–201. University of Bergen Library. ► Nivre, J., de Marneffe, M.-C., 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 N. Calzolari, K. Choukri, T. Declerck, M. Grobelnik, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, eds., *Proceedings of the Tenth International Conference on Language Resources and Evaluation, LREC 2016*, pp. 1659–1666. ELRA, European Language Resources Association (ELRA). ► Patejuk, A. and Przepiórkowski, A. (2012). Towards an LFG parser for Polish: An exercise in parasitic grammar development. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012*, pp. 3849–3852. ELRA. ► Patejuk, A. and Przepiórkowski, A. (2015). An LFG analysis of the so-called reflexive marker in Polish. In M. Butt and T. H. King, eds., *The Proceedings of the LFG'15 Conference*, pp. 270–288. CSLI Publications. ► Çetinoğlu, Ö., Foster, J., Nivre, J., Hogan, D., Cahill, A., and van Genabith, J. (2010). LFG without c-structures. In M. Dickinson, K. Müürisep, and M. Passarotti, eds., *Proceedings of the Ninth International Workshop on Treebanks and Linguistic Theories (TLT 9)*, pp. 43–54. ► Øvrelid, L., Kuhn, J., and Spreyer, K. (2009). Cross-framework parser stacking for data-driven dependency parsing. *TAL*, **50**(3), 109–138.

[2] Similar cases of the conditioning of the label on the order of conjuncts occur also elsewhere in UD, e.g., in the case of unlike nominal/clausal coordination in the subject position, where the label of the subject dependency is either `nsubj` or `csubj`, depending on the first conjunct.

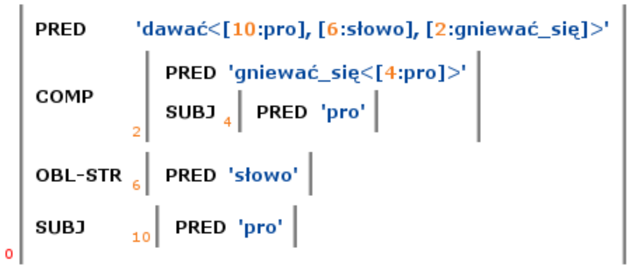
[3] The differences between the basic tree in the upper part and the enhanced representation in the lower part are shown in red. Note that the enhanced representation does not have to be a superset of the basic UD relations. In particular, enhanced dependencies allow for additional subtypes of some relations, namely, for the representation of the preposition in the relation, e.g., `obl:pod` in (8).

(1) - Słowo daję, że się nie gniewam.
word.ACC give.1.SG that RM NEG be_angry.1.SG
‘I give you my word that I am not angry.’

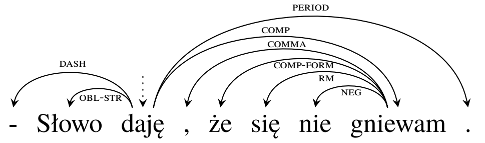
(2)



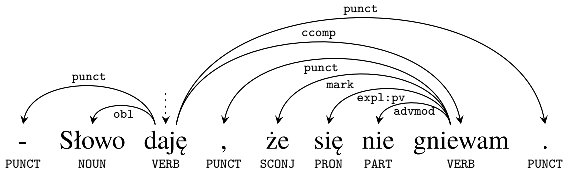
(3)


PRED 'dawać<[10:pro], [6:słowo], [2:gniewać_się]>'
COMP
PRED 'gniewać_się<[4:pro]>'
SUBJ 4 PRED 'pro'
2
OBL-STR 6 PRED 'słowo'
SUBJ 10 PRED 'pro'
0

(4)

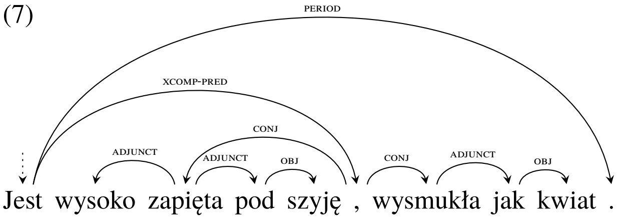


(5)

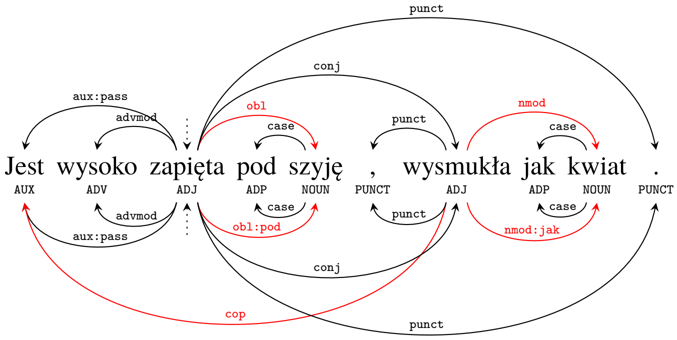

punct
ccomp
punct
mark
expl:pv
advmod
punct
obl
- Słowo daję , że się nie gniewam .
PUNCT NOUN VERB PUNCT SCONJ PRON PART VERB PUNCT

(6) Jest wysoko zapięta pod szyję, wysmukła jak kwiat.
is.3.SG highly buttoned_up.NOM.SG.F under neck.ACC lean.NOM.SG.F as flower.NOM.SG.M
‘She is buttoned up high to the neck, lean as a flower.’

(7)

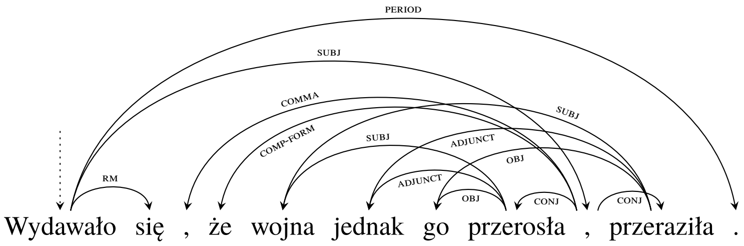


(8)

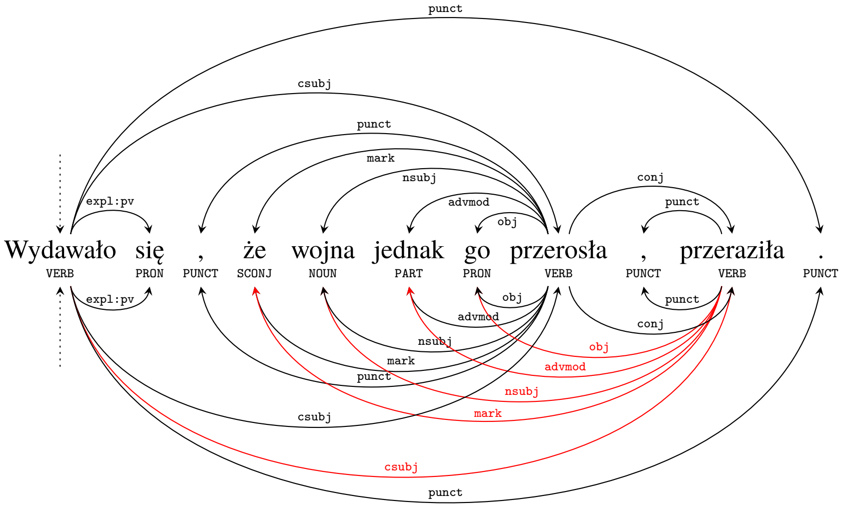

punct
conj
aux:pass
advmod
obl
case
punct
nmod
case
Jest wysoko zapięta pod szyję , wysmukła jak kwiat .
AUX ADV ADJ ADP NOUN PUNCT ADJ ADP NOUN PUNCT
advmod
case
obl:pod
punct
case
nmod:jak
aux:pass
conj
cop
punct

(9) Wydawało się, że wojna jednak go przerosła, przeraziła.
seemed RM that war.NOM.SG.F after all him.ACC overwhelmed.3.SG.F scared.3.SG.F
‘It seemed that after all the war overwhelmed and scared him.’

(10)



(11)


punct
csubj
punct
mark
nsubj
advmod
obj
conj
punct
expl:pv
Wydawało się , że wojna jednak go przerosła , przeraziła .
VERB PRON PUNCT SCONJ NOUN PART PRON VERB PUNCT VERB PUNCT
expl:pv
obj
advmod
punct
conj
nsubj
obj
mark
advmod
punct
nsubj
mark
csubj
csubj
punct