Linguistic Annotation: from Links to Cross-Layer Lexicons

Jan Hajič, Zdeňka Urešová
Center for Computational Linguistics, Charles University, Prague
Malostranské nám. 25, 11800 Prague 1, Czech Republic
E-mail: {hajic,uresova}@ufal.mff.cuni.cz

1 Introduction

Lexicons have always been part of linguistic studies, the more in the era of computational linguistics. Complex linguistic annotation has emerged as an important research phenomenon relatively recently. Even though various annotation schemes ([10], [13], [15], [16], [17]) have been developed containing some sort of explicit or implicit reference to a "lexicon", none has presented a coherent and formal merge of structured linguistic annotation of a running text and lexicons. We believe that an explicitly described relation between lexicons and corpus annotation is necessary to facilitate both the analysis and generation of natural language sentences (when learned from the corpora) as well as a checking tool for annotation consistency.[1]

2 Motivation

The Prague Dependency Treebank (PDT, [2]) is a manually annotated million-word corpus of Czech texts with a rich, multilayered annotation scheme ([5]). Its multilayer architecture with a stress on minimal (if any) redundancy in the annotation makes it essential to mutually link the layers in a consistent manner. Three layers are used at present: the morphological layer, analytical layer (surface dependency syntax), and a tectogrammatical layer (underlying syntactico-semantic structure). The layers are independent in the sense that although they contain very different, non-redundant information, the original sentence should be in principle recoverable from any one of them.[2]

Even though one could design the system in such a way that there is no explicit (annotated) relation between the neighboring layers, it would be foolish to do so,

[1]This work has been supported by the Ministry of Education of the Czech Republic project LN00A063 and by the Grant Agency of the Czech Republic No. 405/03/0913.

[2]By "information" we mean information that a *human* needs to have to do so, not necessarily that we have already formalized the process explicitly for the *machine* to do so. We are talking about the "deepest", tectogrammatical layer here; it is almost trivial to reconstruct the original sentence from either the analytical or morphological layer.

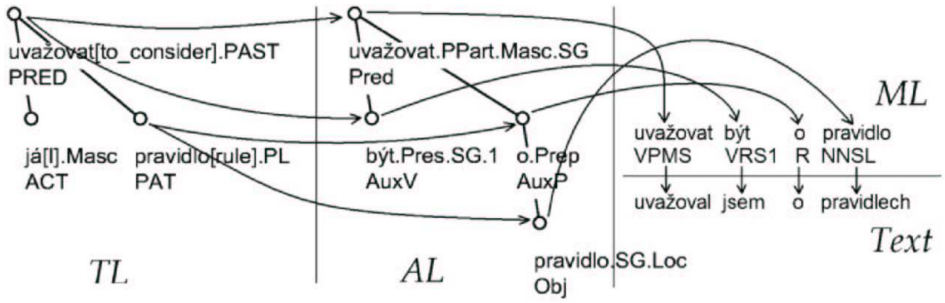

Figure 1: Links between units of annotation (*TL - tectogrammatical layer, AL - Analytical (surface syntax) layer, ML - morphological layer*)

lemma	*TAG*	*wordform*
být	VRS1	jsem
o	R	o
pravidlo	NNPL	pravidlech
uvažovat	VPMS	uvažoval

Figure 2: Fragment of an extracted morphological lexicon (*ML* ↔ *Text*)

since an important information for analysis and generation of a natural language text would be lost. These links can be relatively easily inserted while annotating the "upper" one of two neighboring layers (almost for "free", see also Sect. 7).

3 Linking the Layers

As can be seen in Fig. 1, links are often *1:1* (between the morphological and textual layer, and between the analytical and morphological layer), but they are more complex between the tectogrammatical and analytical layers (such as *1:2*, see *uvažovat (to_consider)* which is linked to two nodes: *uvažovat* proper and an auxiliary *být*, roughly corresponding to *"have considered"* in English; a *1:0* correspondence is also possible: personal pronoun *já ("I")* is dropped at the analytical layer, etc.).

4 From Links to Lexicons

Useful information can be derived from the annotated links in a corpus, in a statistical, machine-learning or similar manner (see Fig. 2 for an example of a **morphological lexicon** extracted from the sentence at Fig. 1). The annotation can then refer to entries of such a lexicon; in a fully annotated treebank, this is obviously redundant, but when facing the tasks of analyzing (parsing) and generating when only a single layer is available on input, the possibility to match and retrieve the lexical entries is invaluable for processing: the entries contain generalized information about the cross-layer relations.

Such a lexicon can be also created and extended manually even without additional data annotation. For example, it is unnecessary to annotate millions (and for inflective languages such as Czech or Russian even billions) of words to subsequently fully learn the morphology of such languages:[3] it is more efficient to create manually a (relatively complete) lexicon of morphological behavior of words directly ([1]). In such a case, human ability to generalize and regularize is more effective than that of the machine.

The situation is more complex between the structural layers of annotation, namely, between the tectogrammatical layer and the analytical layer. In both cases, the core annotation is expressed as a labeled (dependency) tree, the nodes of which do not necessarily correspond one-to-one across these two layers.[4] The **tectogrammatical lexicon**[5] thus cannot express its entries as a pair of related attribute and value pair sets, but it has to work with pairs of subtrees of varying complexity. Some relations (i.e., the lexicon entries) will be very simple (such as a tectogrammatical lemma vs. analytical lemma correspondence), while some will be much more complex. The definition of an entry as a tree-to-tree correspondence (regardless of its complexity) allows us to keep the lexicon as a set of entries of the same type. However, for historical as well as technical reasons, we refer to subsets of entries that share certain properties as (sub)lexicons (of the TL lexicon).

The most complex set of entries of the tectogrammatical lexicon is the valency (sub)lexicon ([3]). Its entries relate a one-level deep subtree of the tectogrammatical tree with its analytical counterpart, with full morphosyntactic subcategorization. They could be relatively easily extracted from the links between the tectogrammatical and analytical layers in the annotated data, but for many reasons (consistency being the first of them) it is again more effective to create it manually while using it already while annotating the data.[6] We will devote the rest of the paper to the valency lexicon, since it is the most complex and interesting.

5 The Valency Lexicon

In linguistics the term "valency" ([14]) indicates the capability of lexical units to bind ("meaning"-wise) other terms onto itself; their number and character is deter-

[3] Not considering methods that work off unannotated data, such as ([18]).

[4] Although not required in general, the correspondence between the nodes of the analytical layer and the tokens of the linear morphological layer *is* 1:1 (for Czech, but it might be different for other languages even if using a surface syntax description similar to ours).

[5] Please note that we name the lexicons based on the "higher" layer of annotation; i.e., a morphological lexicon refers to a lexicon relating the morphological and textual layers, and similarly, the tectogrammatical lexicon refers to a lexicon relating the tectogrammatical and analytical layers.

[6] The lexicons mentioned so far are so-called cross-layer lexicons: they relate information between two layers of annotation: the morphological lexicon relates forms and lemmas and tags; and the valency lexicon maps valency frames to their surface syntax form (as annotated at our analytical layer, with enough information to generate proper morphology). There are also more simple types of lexicons used in the PDT, such as a tag lexicon, function lexicon etc.; these are simple sets and we do not describe them in detail here.

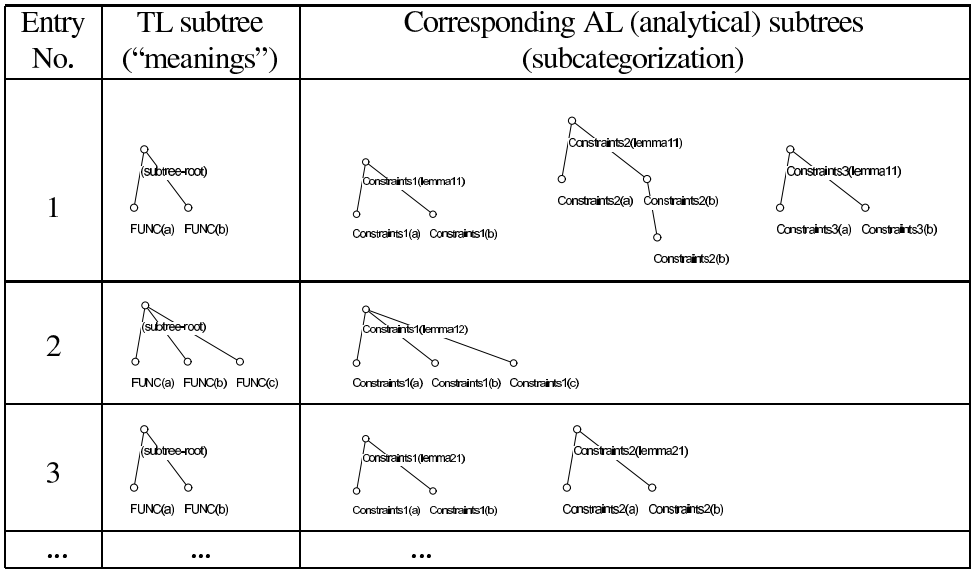

Figure 3: Structure of the valency (sub)lexicon (graphical representation).

mined for each word (or rather, word sense, or meaning) separately. Verbs have a valency frame, and so have many nouns and adjectives (and not only those derived from verbs directly); technically, every autosemantic word has one. For more on the valency lexicon used with the Prague Dependency Treebank, see ([3]) in this volume.

At the moment, we assume that each sense of a given word (i.e., one valency lexicon entry, see Fig. 3) contains one *valency frame*.[7] Each valency frame contains a fixed number of *slots*. Each slot contains a slot name (called *functor*, e.g. `ACT`, `PAT`) that effectively labels the corresponding dependency relation in a tectogrammatical tree (Fig. 3, 2^{nd} column: slots correspond to dependent nodes and are marked `FUNC(a)`, `FUNC(b)`, etc.). Moreover, a slot also refers to all the necessary information how to construct a surface dependency syntax representation (i.e. the one used at the analytical layer of annotation) for the given slot.[8] Such an information is in principle an underspecified analytical subtree (Fig. 3, 3^{rd} column), with morphosyntactic constraints of the form *attribute* = *value*.[9] Various globally defined default subtree transformations may also occur (in the case of passivization, for example).

Since a graphical representation is little less convenient to edit, and since the

[7]From the cross-layer lexicon point of view, it would not change anything if there are more valency frames per word sense, but for simplicity, we leave out the associated discussion whether this could happen, since it is rather theoretical anyway.

[8]Entries of a full valency lexicon ([9], [8]) contain additional information, such as the degree of optionality of the slots, links to semantic descriptions etc.

[9]For an example, see Fig. 5.

Entry No.	Slots			Lemma (AL)
	1	2	3	
1	$FUNC_a$(Constraints)	$FUNC_b$(Constraints)		lemma11
2	$FUNC_a$(Constraints)	$FUNC_b$(Constraints)	$FUNC_c$(Constraints)	lemma12
3	$FUNC_a$(Constraints)	$FUNC_b$(Constraints)		lemma21
...	...	...	...	...

Figure 4: Textual form of valency lexicon entries

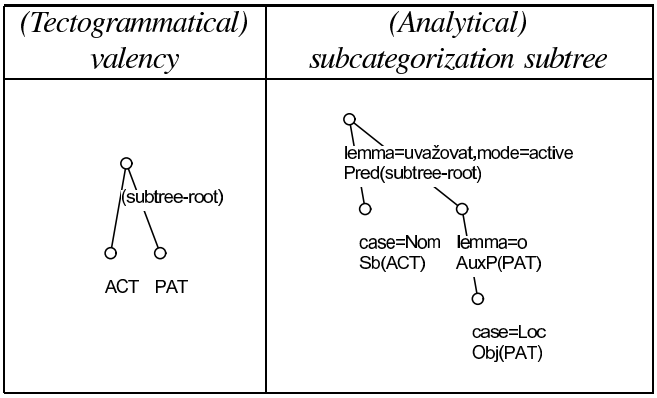

Figure 5: Valency frame for *uvažovat ("to_consider")* with its (underspecified) analytical form; cf. also Fig. 1 and Fig. 3.

information can be structured even more (at least for the majority of real life cases), we use a shorthand notation that can be represented and edited in textual form (Fig. 4); by showing the tree correspondences in Fig. 3 we wanted to stress that in general we cannot limit ourselves to the simple cases.

As we have stated previously, every entry in both Fig. 3 and Fig. 4 corresponds to a certain sense of some "word". We have intentionally left out the actual words out of the figures; it is sufficient to distinguish among them by the entry number. However, for easier editing, the annotators see a *tectogrammatical lemma* (a string resembling a classic printed dictionary entry heading, such as *to_consider*[1] for the sense 1 of the verb *to consider*), which is then internally mapped to a valency frame identifier.[10] Moreover, the valency lexicon contains fields for examples, comments and other notes to describe the sense and usage of the particular valency frame.

Fig. 5 shows an example of a valency frame (in the graphical form).

At each node of the analytical subcategorization subtree, a set of constraints expressed by attributes and their values effectively determines what are the combinations allowed in the fully specified analytical tree.[11] This information can be

[10]Since our sense inventory is not yet fixed, we in fact do not have unique sense numbers attached to the tectogrammatical lemmas, but the frames *are* uniquely identified.

[11]The creation of the auxiliary verb *"být" ("to_be")* (Fig. 1) is the result of a general rule for

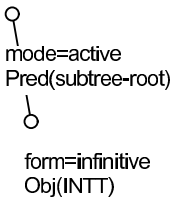

Figure 6: Example subcategorization: INTT as infinitive

used for sentence generation (from the tectogrammatical representation), as a supportive information for parsing (to the tectogrammatical representation) as well as for annotation checking (when building a treebank manually).

6 Types of Subcategorization

Various possible combinations of constraints have emerged during the annotation process and the parallel valency lexicon buildup, from a simple (morphological) case and *1:1* tectogrammatical-to-analytical node correspondence to complex relations for phraseological expressions. We thus distinguish the following types of correspondence between a valency frame slot and its form:

- *trivial subtree with a single analytical node:* Within this group, the following types of constraints are possible:
  - noun (phrase) head in a particular (morphological) case (in Fig. 5, it corresponds to the `[case=Nom / ACT]` node)
  - verb in infinitive (*we sent him to_learn*, for "Intention" (INTT)) (Fig. 6)[12]
  - relative subordinate clause without a conjunction (*know what comes*) (Fig. 7)
  - specific word, or only several words (usually idioms) (*be good*) (Fig. 8, in Czech: *dělat dobrotu*)
  - specific class of words, usually semantically related (*forward an application / a request / ...*) (Fig. 9)
  - functor-based-only constraint (functor LOC: *to come home*; there are other possibilities for expressing a location, such as a preposition and a noun, but they involve more than a single analytical tree node (to school / into a cave)) (Fig. 10)
  - and more.

[12] generation of a proper verb form from the "past perfect" attribute; it has nothing to do with valency

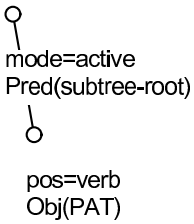

Figure 7: Example subcategorization: PAT as relative clause

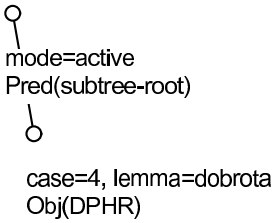

Figure 8: Example subcategorization: DPHR as a specific inflected lemma

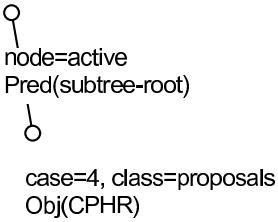

Figure 9: Example subcategorization: CPHR as a lemma from a class

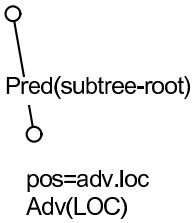

Figure 10: Example subcategorization: LOC as an adverb for location

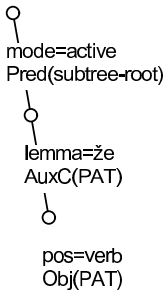

Figure 11: Example subcategorization: PAT as a subordinate clause with conjunction

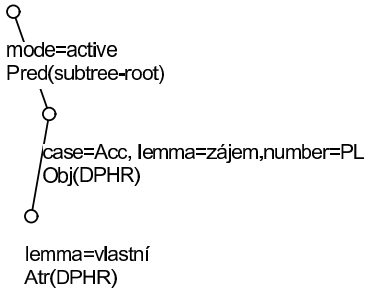

Figure 12: Example subcategorization: DPHR as a two-node, two-level subtree

- *rooted subtree with two nodes:* in a specified order:
  - preposition and its dependent in a particular (morphological) case (in Fig. 5, see the subtree with the preposition *"o"* and the `case=Loc` constraint at its dependent node)
  - subordinate conjunction and a dependent subordinate clause (*he discovered that it is true*) (Fig. 11)
  - specific word (conjunction) and infinitive (*no choice but come*)
  - two specific words (usually dependent on each other) (in Czech: *sleduje vlastní zájmy*, *"follows [his] own interests"*) (Fig. 12)
  - and more.
- other complex subtree topologies:

information.

[12] In Figs. 6-12, only the exemplified dependent node is shown; typically, there is also a node corresponding to the ACT of the subtree root, such as the one in Fig. 5.

- *1:3* correspondence of various shapes of the analytical tree, usually with specific words only (for certain types of idioms or phrasal expressions)

A combination of several of the above is also possible; for phrasal expressions, it is even possible that the phrasal part consists of two discontinuous subtrees.[13]

7 Annotating the Links

Superficially, it might seem that linking the layers of annotation is an unnecessary exercise - if we survey the available corpora with a similar structural annotation schemas, they do not explicitly do so, either. However, as we have said in the Motivation section (Sect. 2), we *do* want to have the links explicitly marked for *all* annotated units (nodes of syntactic trees as well as tokens) for lexicon extraction and checking purposes.[14]

In a (technically) single annotation scheme (such as the added annotation of Propbank predicate-argument structure to the Penn Treebank annotation, [7], [10], [11], [12]), the links are implicit (all the information is present in the chosen annotation structure, from predicates and arguments to POS tags). The associated PropBank lexicon does not formalize the form of the arguments, either: e.g., the prepositions used in the surface expression of the arguments are merely recorded as comments in the lexicon. Such a scheme is not very flexible - it allows only for direct correspondence of nodes to tokens, with difficulties to mark non-1:1 relations. On the other hand, it is clear that such an annotation does not add any extra burden to the annotators.

In our case, when the links are explicit, one could expect such an added burden. However, thanks to the programming capability of the annotation tool we use (TrEd, [4]), we were able to avoid any need for the annotators to mark those links explicitly. Only in some rare cases of annotation errors we have to restore those links manually during the annotation checking process.

For example, when the annotators of the tectogrammatical layer (or the preprocessing software) decide to create a new "tectogrammatical lemma" (e.g., for reflexive verbs, such as *ztratit_se "to disappear"*) from two nodes at the analytical layer (i.e. *ztratit* and *se*), they use a "macro" command that does that for them but it also (invisibly) inserts the appropriate double link (see Fig. 13). Similarly, macros exist for the collapsing of multi-node verb forms to a single tectogrammatical node, etc. The process is (auto)reversible so that the annotators can correct themselves without being stuck with the new links. In fact, most annotators are not even aware of the existence of the links.

[13] For example, *"mráz(PHR1) běhá(PRED) po_zádech(PHR2)"*, lit. *"(a) frost runs on (sb's) back"* (*a shiver runs down sb's spine*).

[14] One might even imagine that someone comes with a different idea of the types of lexicons to be extracted, especially between the tectogrammatical and analytical layers. The links will then allow her/him to do so.

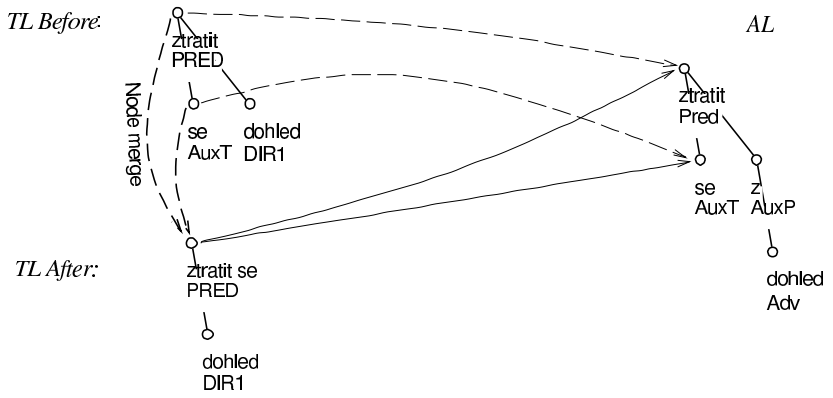

Figure 13: Links before and after an annotation action.

Technically, the links are implemented as ID $\leftrightarrow$ ID-reference pairs. The total number of them corresponds to the number of tokens times the number of layers.

8 Annotation and (Manual) Lexicon Creation

The valency lexicon is being created in parallel with the annotation. One could speculate to which extent the process can be automated or aided based on the increasing amount of annotated data during the annotation process. Unfortunately, it appeared that due to the relative novelty of the notions that drive the valency dictionary, inter-annotator disagreement on valency frames was as high as 20%, and therefore time-consuming reconciliation[15] was apparently needed to make the dictionary as consistent as possible.

Therefore, we have adopted a two-stage process. First, the annotators annotated the whole corpus (55k sentences), with the valency dictionary being merged and redistributed once every week (the annotators, working off-line for various reasons, could not share and access a single valency dictionary). Every annotator had the right to change, delete or insert any valency frame, with a log of changes preserved. Then, the fully merged dictionary was checked by a single person[16] to reconcile the entries. Finally, the data has been run against the resulting dictionary, nodes which were affected by the reconciliation changes were automatically marked and they will subsequently be manually re-annotated for valency.

An automatic procedure for valency annotation has been being developed in parallel already during the first phase ([6]), but it has been actually used only in the very late stages of annotation with some, but not substantial improvement.[17]

[15]The annotation, however, could not stop in the meantime because of administrative and budgetary reasons.

[16]More precisely, one for verbs and another one for nouns. Of course, they consulted with the authors of the affected entries and with others at regular meetings, but the final decisions were theirs.

[17]We believe that the main reason for the relatively small improvement only was the inconsistency

9 Conclusions

The formalization of lexicons in linguistic annotation could lead to better understanding of the relations between different depths of annotation, to less redundant annotation, and to more transparent usage. If such formalization exists and is complete, these lexicons can be either extracted automatically from manually annotated data, or, in case of insufficient amount of such data, created manually (and then used in the same way for annotation checking, and automatic parsing and generation).

References

[1] Jan Hajič: *Disambiguation of Rich Inflection (Computational Morphology of Czech)*, Karolinum Press, Prague, 324pp, 2003.

[2] Jan Hajič, Eva Hajičová, Petr Pajas, Jarmila Panevová, Petr Sgall, Barbora Vidová Hladká: *The Prague Dependency Treebank*, CDROM LDC2001T10, The Linguistic Data Consortium, Univ. of Pennsylvania, ISBN 1-58563-212-0, Philadelphia, PA, 2001

[3] Jan Hajič, Jarmila Panevová, Zdeňka Urešová, Alevtina Bémová, Veronika Kolářová, Petr Pajas: *PDT-VALLEX: Creating a Large-coverage Valency Lexicon for Treebank Annotation*. In Proceedings of TLT 2003. Växjö, Sweden, Nov. 14-15. This volume. 2003.

[4] Jan Hajič, Petr Pajas, Barbora Hladká: *The Prague Dependency Treebank: Annotation Structure and Support*. In Proceedings of IRCS Workshop on Linguistic Databases, pp. 105–114. Philadelphia, PA, Dec. 11–13 2001.

[5] Eva Hajičová, Jarmila Panevová, Petr Sgall: *A Manual for Tectogrammatic Tagging of the Prague Dependency Treebank*, ÚFAL/CKL Technical Report TR-2000-09, Charles University, Prague, 2000

[6] Václav Honetschläger: Using a Czech Valency Lexicon for Annotation Support. In V. Matoušek and P. Mautner (eds.), Proceedings of TSD'03, Českë Budějovice, Czech Republic, Sept. 8-12, LNAI 2807, pp. 120-125, Springer-Verlag Berlin Heidelberg. 2003.

[7] Paul Kingsbury, Martha Palmer: *From Treebank to PropBank*. In Proceedings of LREC 2002, Las Palmas, Canary Islands, Spain, 2002.

[8] Markéta Lopatková: *Valency in Prague Dependency Treebank: Building the Valency Lexicon of Verbs*, to appear in: Prague Bulletin of Mathematical Linguistics, Vol. 79, Charles University, 2003.

of the valency dictionary before its reconciliation. Obviously, we could not test it for "real" afterward since the corpus had already been fully annotated.

[9] Markéta Lopatková, Zdeněk Žabokrtský, Karolína Skwarska, and Václava Benešová: *Tektogramaticky anotovaný valenční slovník českých sloves*, ÚFAL/CKL TR-2002-15, Charles University, Prague, 2002

[10] Mitchell P. Marcus, Beatrice Santorini and Mary-Ann Marcinkiewicz (1993) Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics* 19, 313–330. [Reprinted in Armstrong, Susan (ed.) (1994) *Using large corpora*, pp. 273–290. Cambridge, MA: MIT Press.]

[11] Mitchell P. Marcus, Grace Kim, Mary-Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, Britta Schasberger (1994) The Penn Treebank: Annotating Predicate Argument Structure", In *ARPA Human Language Technology Workshop*.

[12] Margorzata Marciniak, Agnieszka Mykowiecka, Anna Kupść and Adam Przepiórkowski (2000) An HPSG-Annotated Test Suite for Polish. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC 2000)*.

[13] Kemal Oflazer, Bilge Say, and Dilep Hakkani-Tür (2000) A Syntactic Annotation Scheme for Turkish. In *Proceedings of 10th International Conference on Turkish Linguistics (ICTL-2000)*.

[14] Jarmila Panevová: *On verbal frames in functional generative description*, in *The Prague Bulletin of Mathematical Linguistics 22*, pages 3–40, 1974

[15] Louisa Sadler, Josef von Genabith, Andy Way (2000) Automatic F-Structure Annotation from the AP Treebank. In Butt, Miriam and Holloway King, Tracy (eds.) *Proceedings of the Fifth International Conference on Lexical-Functional Grammar*, The University of California at Berkeley, 19 July – 20 July 2000. Stanford, CA: CSLI Publications.

[16] Kiril Simov, Gergana Popova, Petya Osenova (2003) HPSG-Based Syntactic Treebank of Bulgarian (BulTreeBank). In A. Wilson, P. Rayson, T. McEnery (eds.) *A Rainbow of Corpora: Corpus Linguistics and the Languages of the World*, pp. 135-142. Munich: Lincom-Europa.

[17] Ulf Teleman (1974) *Manual för grammatisk beskrivning av talad och skriven Swedish.* Lund: Studentlitteratur.

[18] David Yarowsky, Richard Wicentowski (2000) *Minimally supervised morphological analysis by multimodal alignment*. In Proceedings of the 38th ACL, Hong Kong, October, p. 207-216. 2000.