Closing the Gap Between Stochastic and Hand-crafted LFG Grammars

Annette Hautli
Universität Konstanz
annette.hautli@uni-konstanz.de

Özlem Çetinoğlu Josef van Genabith
Dublin City University
{ocetinoglu,josef}@computing.dcu.ie

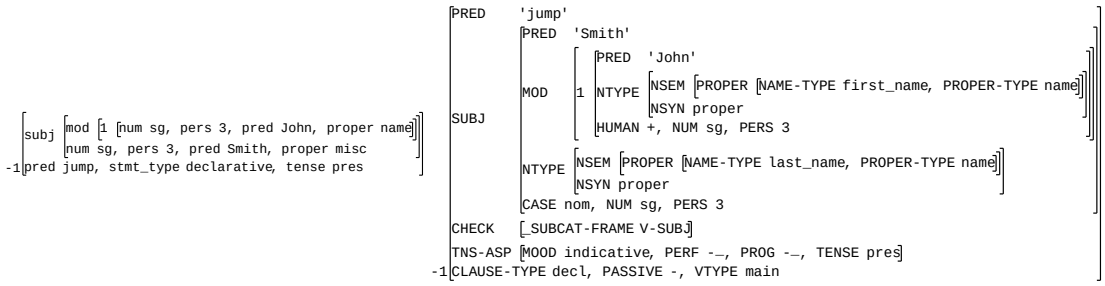
Developing large-scale deep grammars in a constraint-based framework such as Lexical Functional Grammar (LFG) is time-consuming and requires significant linguistic insight. This paper presents an approach to extend the stochastic DCU LFG annotation algorithm with more detailed f-structure information. It thereby reaches the feature detailedness of state-of-the-art hand-crafted grammars such as the English XLE grammar, while profiting from the robustness and the good coverage of stochastic grammars.

The DCU annotation algorithm (DCU grammar) (Cahill, 2004) comprises of the following parts: a stochastic parser (Charniak, 2000, Bikel, 2002) that creates trees in the Penn-II Treebank style (Marcus et al., 1993), a module that automatically annotates these trees with f-structure equations, a constraint solver that unifies the equations and produces f-structures, and a module that resolves long-distance dependencies. The system has been successfully evaluated on gold standards such as the PARC700 (King et al., 2003), outperforming the hand-crafted English XLE grammar by 2% (Cahill et al., 2008).

This provides an excellent basis for a further development of the DCU grammar. However, as the system was solely tuned to produce presentations restricted in detailedness with only some syntactic and semantic features, it lacks the detailedness of the hand-crafted English XLE grammar. In order to close the gap between the stochastic and the hand-crafted grammar, we need to extend the restricted feature space of the DCU grammar to get f-structures as detailed as XLE f-structures. Table 1 gives the original DCU features (34 in total) with the newly added f-structure features in bold (29 added). Extending the feature

F-structure feature space of the DCU grammar
adegree, adjunct, **adjunct-type**, **adv-type**, aquant, **atype**, **case**, **clause-type**, **common**, comp, conj, **coord**, coord-form, **deg-dim**, **degree**, **deixis**, **det**, det-form, **focus**, focus-int, **gend-sem**, **human**, **inf-type**, mod, **mood**, **name-type**, **nsem**, **nsyn**, **ntype**, num, number, number-type, obj, obj-th, obl, obl-ag, obl-compar, **part**, passive, pcase, perf, poss, prog, pron-form, pron-int, pron-rel, proper, **proper-type**, prt-form, **psem**, **ptype**, quant, **spec**, stmt-type, subj, subord-form, tense, **time**, **tns-asp**, topic-rel, **vtype**, xcomp, **xcomp-pred**

space includes renaming the already existing features and restructuring their representation in the f-structure. The following f-structures for *John Smith jumps.* exemplify how the tense/aspect paradigm is represented in the restricted and the extended DCU grammar. The existing TENSE feature is embedded into the TNS-ASP f-structure which now also contains the features MOOD, PERF and PROG. The extended f-structures also contain features that



distinguish first names from last names (as in *John Smith*). To add this information, the

relation of the tree nodes was taken into consideration. If two (or more) proper nouns are sisters and the lemma of the leftmost node can be found in a list of first names, we annotate it with the information that it is a first name. This is successively done until one of the sisters to the right is not a first name, annotating this node with the feature for last names. Therefore, names with a middle name are also detected (e.g. *John Adam Smith*).

A recent approach (Hautli and King, 2009) attempted to overcome the differences between the hand-crafted XLE grammar and the stochastic DCU grammar by using a set of ordered rewrite rules that added missing f-structure information, accepting the DCU system as a black box. Both architectures (extended DCU grammar vs. rewrite rules) were tested on a testsuite of 720 sentences, used to test the semantics of the English XLE grammar. The results show that the features can be reconstructed successfully in both scenarios, however extending the DCU grammar is more effective than using a set of rewrite rules, because we can allow for operations that are unavailable to the rewrite approach, such as taking into account the relation of nodes in the tree, as exemplified in annotating first and last names. For external validation, we took the PARC700 gold standard with a core feature

	Extended DCU grammar			**Set of rewrite rules**		
	precision	recall	f-score	precision	recall	f-score
sem_test	82.14	76.23	79.08	70.31	67.69	68.98

structure and evaluated the DCU grammar against it. This allowed us to check whether the extended system also performs well on other data than the testsuite, as in the case of PARC700, newspaper text of the Wall Street Journal. Performance with the development set of PARC700 (140 sentences), has the following results:

	precision	recall	f-score
development set PARC700	83.93	74.5	78.93

These initial experiments show that the gap between stochastic and hand-crafted grammars can be closed, which means that there is a possibility of generating deep LFG grammars on the basis of treebanks for other languages as well, benefiting from the aspect that the trees are automatically created with a robust parser and have a very good coverage for unknown text. As opposed to the earlier approach of employing the DCU grammar as a black box and using rewrite rules in addition, this new approach results in a single DCU grammar, which is more efficient and also has a higher accuracy due to the extra information that is available within the DCU grammar. By being able to take into account tree information which we previously could not, we allow for more LFG linguistic insight that can be captured in the final f-structure representation.

References

D. M. Bikel. Design of a multi-lingual, parallel-processing statistical parsing engine. In *The Proceedings of HLT 2002*, 2002.

A. Cahill. *Parsing with Automatically Acquired, Wide-Coverage, Robust, Probabilistic LFG Approximations*. PhD thesis, School of Computing, Dublin City University, 2004.

A. Cahill, M. Burke, R. O'Donovan, S. Riezler, J. van Genabith, and A. Way. Wide-coverage deep statistical parsing using automatic dependency structure annotation. *Computational Linguistics*, 34(1):81–124, 2008.

E. Charniak. A maximum entropy-inspired parser. In *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 132–139, 2000.

A. Hautli and T. H. King. Adapting stochastic lfg input for semantics. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG '09 Conference*. CSLI Publications, Stanford, 2009.

T. H. King, R. Crouch, S. Riezler, M. Dalrymple, and R. Kaplan. The PARC700 dependency bank. In *Proceedings of the EACL03: 4th International Workshop on Linguistically Interpreted Corpora (LINC-03)*, 2003.

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19(2):313–330, June 1993.