On the 'spirit of LFG' in current Computational Linguistics

Jonas Kuhn
jonas.kuhn@ims.uni-stuttgart.de
IMS, University of Stuttgart, Germany

In this position paper, I take a look at some of the key "design principles" of LFG and draw some parallels to developments in research on Natural Language Processing (NLP) and Computational Linguistics over the past few years. A number of recent trends and findings in NLP research seem to have precedents in earlier LFG work in ways that have not received much attention so far. Since the current computational work in which some original LFG design principles resurface is embedded in quite a different methodological context, it is not clear to what extent implications from the earlier work will still apply. One might also argue that the parallels that can be drawn are at a level that is too abstract to make any point that is of scientific interest. I believe however that it is worthwhile taking a closer look and seeing whether the common aspects behind the original LFG ideas and the current computational research questions can be given a meaningful interpretation across frameworks. The hope is that an increased awareness in the LFG community may lead to some new cross-fertilisation in the near future.

Traditionally, the LFG community has been known to be a rare showcase for a continued and successful exchange between theoretical and computational linguistics. This has probably numerous reasons, but one is clearly that the representations used in the LFG formalism are an ideal common ground for exchanging thoughts about linguistic analyses of data from languages across the typological spectrum. The reflex of heavily theory-internal assumptions is carefully avoided in the representations; and for each relevant dimension of linguistic description, a formal structure is chosen for representation that displays the observed properties (trees for c-structure, set-based feature structures for f-structure etc.). These structurally straightforward representations allow both the theorists and the computationalists to anchor their respective systematic accounts – using a constraint-based and lexicalist approach. In what Johnson (2011: 3) calls the "golden age for collaboration and cross-fertilisation between linguistic theory and computational linguistics" – the 1980s – the connection was very obvious, but in the LFG community, the collaboration continued to be successful when the "empiricist" camp in NLP gathered momentum in the 1990s and statistical techniques were beginning to dominate research in computational linguistics (see Church, 2011). LFG has not only been the theoretical framework for one of the most successful attempts of engineering linguistically grounded broad-coverage grammars across languages (in the well-known ParGram project, Butt et al. 2002), but it also provided the representational framework for important work on treebank-based grammar acquisition (Cahill et al. 2008a), discriminative ranking models for parse disambiguation (Riezler et al. 2002), and statistical constituency-based pruning (Cahill et al. 2008b).

There is successful ongoing research work in the mentioned traditions; at the same time however, it has to be acknowledged that a lot of the computational analysis tasks (e.g., machine translation, semantic role labelling, coreference resolution) for which there was no doubt in the late 1980s that they would require carefully engineered knowledge sources, are quite successfully approached with cascades composed of statistical modules, each solving a structurally relatively simple input-output mapping. This is not to say that the importance of linguistic insights is not acknowledged in the field of NLP – the last few years have actually brought about a lot of occasions in which the relation between linguistics and language technology has been discussed (the 2011 *Linguistic Issues in Language Technology* on "Interaction of Linguistics and Computational Linguistics" is just one example; here King 2011 represents the LFG view); the occasionally hostile atmosphere between the camps from the 1990s has clearly ceased to exist. However it somehow seems that the common denominator across fields ended up less sophisticated than many would have hoped: linguistic insight is clearly needed for high-quality gold-standard corpus annotation; but most other ingredients for effective computational models seem to be taken from general-purpose machine learning that operates on this training data, avoiding any tailoring to peculiarities of the data representations (which is what linguistics in the generative tradition seems to be all about).

It is at this point that I would like to go into some recent developments: As the results for some of the standard NLP problems that can be addressed with supervised methods (such as treebank-trained constituent parsing or dependency parsing for English) are reaching a plateau, a new set of refined research questions comes up: (i) Moving away from input-output modules that solve a single intermediate step in a processing pipeline (e.g., part-of-speech tagging, morphological analysis, syntactic constituent and/or dependency parsing, semantic role labelling, co-reference resolution), what are effective ways of solving combined problems spanning more than one step in the pipeline (so-called "joint modelling", or effective approximations thereof)? Examples are Goldberg/Tsarfaty 2008, Li et al. 2010, Bohnet/Nivre 2012. If some approximation of a joint model is assumed, (ii) how can the "candidate set" of intermediate results be best represented? Related to these questions, (iii) what type of intermediate linguistic representations should be assumed where they don't affect the overall task directly? E.g., should constituent or dependency parses, or both, be used for some downstream task such as coreference resolution (Björkelund/Kuhn 2012); how should morphological segmentation be addressed in "morphologically rich" languages (Goldberg/Tsarfaty 2008)? Taking this question to the limit, one may ask what intermediate (linguistic?) representation to assume in end-to-end tasks like machine translation. (Quernheim/Knight 2012, for instance, propose a probabilistic, transducer-based approach to Machine Translation that is clearly in the spirit of earlier LFG work on translation.) (iv) Can some latent representation improve the performance across languages (e.g., Titov/Henderson 2010)? (v) Are there systematic linguistic

constraints that can be exploited, based on a concept of underspecified representation (e.g., Seeker/Kuhn 2013)? (vi) Can the assumptions be formulated in a way so they carry over across typologically different languages?

Note that none of the approaches mentioned are modelled in terms of an LFG grammar or sub-grammar. I would like to claim however that the methodology and the set of research questions is very much in the 'spirit of LFG': as mentioned, part of the long-term interdisciplinary success of LFG lies in the combination of (or: Parallel Correspondence across) relatively straightforward representational levels for which there are good empirical tests. So, typical high-level LFG research questions could be paraphrased as 'what are the primitives that should be assumed at the level of f-structure/a-structure – what effect do the possible choices have on the neighbouring levels of representation?'

Up until about five years ago, the data-driven paradigm in NLP was not questioning the input and output representations assumed in supervised approaches to particular analysis problems – the available datasets were taken for granted, and the challenge was to devise maximally general machine learning techniques. As the network of subtasks feeding and bleeding each other (depending on the assumed architecture) has been growing as outlined above, questions about appropriate interface representations *do* however gain crucial importance. So, at the level of the global architecture across subtasks, the field of NLP very much resembles the problem space that LFG theorists have been addressing all along. And indeed, most of the major interface representations under discussion in current NLP work can be argued to bear close resemblance to the LFG representations (part-of-speech labels: c-structure categories, constituent syntax: c-structure, dependency syntax: f-structure [minus functional control], PropBank/NomBank-style semantic role labelling: a-structure, pronominal co-reference: anaphoric control), and some of the more controversial parts of the NLP architecture, like the interplay of morphological segmentation and syntactic parsing, correspond to controversial parts of the LFG architecture (the morphology-syntax interface).

The major difference is that in classical LFG, the concrete modelling task for relating the various levels is solved in terms of the formulation of symbolic formal constraints describing the possible correspondence relations (and this task is addressed by the linguist or grammar writer), whereas in current "multi-level correspondence" NLP, the concrete pairwise (or larger) relation across levels is determined by machine learning methods operating on training data (possibly with latent intermediate representations). But as the character of the interface representations ceases to be fixed *a priori* in NLP work, the high-level search for the best possible set of interface representations that allows for the modelling of arbitrary languages doesn't seem to be all that different from linguistic research in the generative tradition.

And as far as I can see, LFG's architecture of parallel correspondence across formally heterogeneous representation structures seems to be closer to the current NLP situation than most other linguistic frameworks. This implies that there may be lessons to be learned from the LFG experience, and if the ultimate goal is to develop a satisfactory overall framework that makes sense both to linguists and to NLP researches working in the current paradigm, LFG's parallel correspondence architecture may be a good starting point. Such a framework would also provide the basis for assessing the implications of important developments in NLP work from a linguistic point of view, and thus revive the cross-fertilization between linguistics and computational linguistics.

Acknowledgement

The considerations in this contribution and the work from within my group that I mention have been carried out in SFB 732 "Incremental Specification in Context", funded by the German Research Foundation (DFG), in particular in projects D2 and D8.

References

Björkelund, Anders and Jonas Kuhn (2012). Phrase Structures and Dependencies for End-to-End Coreference Resolution. In Proceedings of COLING 2012: Posters volume, pp. 145-154.

Bohnet, Bernd and Joakim Nivre. A Transition-Based System for Joint Part-of-Speech Tagging and Labeled Non-Projective Dependency Parsing. EMNLP 2012.

Butt, Miriam, Helge Dyvik, Tracy H. King, Hiroshi Masuichi, and Christian Rohrer (2002). The Parallel Grammar Project. In Proceedings of COLING-2002 Workshop on Grammar Engineering and Evaluation. pp. 1-7.

Cahill, Aoife, Michael Burke, Ruth O'Donovan, Stefan Riezler, Josef van Genabith and Andy Way (2008a). Wide-Coverage Deep Statistical Parsing using Automatic Dependency Structure Annotation,.In: *Computational Linguistics*, 34 (1), 81-124.

Cahill, Aoife, John T. Maxwell, Paul Meurer, Christian Rohrer and Victoria Rosén (2008b). Speeding up LFG Parsing using C-Structure Pruning. In COLING-2008: Proceedings of the workshop on Grammar Engineering Across Frameworks, pp. 33-40.

Church, Kenneth (2011). A Pendulum Swung Too Far. In: Linguistic Issues in Language Technology, 6 (5), CSLI Publications.

Johnson, Mark (2011). How relevant is linguistics to computational linguistics? In: Linguistic Issues in Language Technology, 6 (7), CSLI Publications.

King, Tracy H. (2011). (Xx*-)Linguistics: Because We Love Language. In: Linguistic Issues in Language Technology, 6 (9), CSLI Publications.

Li, Junhui, Guodong Zhou, and Hwee Tou Ng (2010). Joint syntactic and semantic parsing of Chinese. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp.1108–1117.

Riezler, Stefan, Tracy H. King, Ronald M. Kaplan, Richard Crouch, John T. Maxwell and Mark Johnson (2002). Parsing the Wall Street Journal using a Lexical-Functional Grammar and Discriminative Estimation Techniques. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02), Philadephia, PA.

Seeker, Wolfgang and Jonas Kuhn (2013). Morphological and Syntactic Case in Statistical Dependency Parsing. In: *Computational Linguistics,* 39 (1):23-55.

Goldberg, Yoav and Reut Tsarfaty. 2008. A single generative model for joint morphological segmentation and syntactic parsing. In Proceedings of ACL-08: HLT, pp. 371–379.

Titov, Ivan, James Henderson (2010). A Latent Variable Model for Generative Dependency Parsing. In H. Bunt, P. Merlo and J. Nivre, editors, Trends in Parsing Technology, Text, Speech and Language Technology Series (Springer).

Quernheim, Daniel and Kevin Knight (2012), Towards Probabilistic Acceptors and Transducers for Feature Structures. In Marine Carpuat, Lucia Specia and Dekai Wu, (eds.), Proc. Of the 6th Workshop Syntax, Semantics and Structure in Statistical Translation, pp. 76-85.