

**ON THE ‘SPIRIT OF LFG’ IN CURRENT
COMPUTATIONAL LINGUISTICS**

Jonas Kuhn
Universität Stuttgart

Proceedings of the LFG13 Conference

Miriam Butt and Tracy Holloway King (Editors)

2013

CSLI Publications

<http://csli-publications.stanford.edu/>

Abstract

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.

1 Introduction

In this position paper, I draw some parallels between some of the key “design principles” of LFG and recent developments in research on Natural Language Processing (NLP) and computational linguistics. Since the current computational work in which some original LFG design principles resurface is embedded in quite a different methodological context, one might 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 both in the LFG community and in the NLP community may lead to some new cross-fertilisation in the near future.

The paper is structured as follows: section 2 will very briefly review a development in the relation between theoretical and computational work in LFG over the past 10 years or so; in section 3, I will then point out some recent developments in data-driven computational linguistics, which are seemingly unrelated to LFG, but as I will claim display many features of the ‘spirit of LFG’. To be able to situate the various approaches in the same architectural coordinate system, I will introduce in section 4 a high-level scheme for classifying different ways of modeling the relation between linguistic forms and their interpretation as a function of the given context of utterance, which in turn serves as the basis for describing different types of modular interface architectures in section 5. Against this backdrop, I illustrate the claim advanced in section 3 for a particular study (specifically Seeker and Kuhn (2013b), of which I will outline the conceptually relevant points). In section 7, I point out the ways in which I think it is the ‘spirit of LFG’ that is particularly relevant for the point I am making and I conclude.

[†]The considerations in this contribution and the work from within my group that I refer to 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. I am indebted to my group for discussions and for contributing the computational and experimental work – in particular to Wolfgang Seeker who has influenced these considerations a lot – and to the collaborators from the SFB for ongoing exchanges about the representational and architectural conception of specification in context.

2 LFG and NLP: The past and the status quo

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 many computational analysis tasks (e.g., machine translation, semantic role labeling, 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 brought about many 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 by and large disappeared. 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. Method-

ologically, if language-specific hard constraints on the search space are used in some experiment, these are considered to be simplifying working assumptions that should ultimately be abandoned in favor of purely data-driven acquisition of all constraints. This is diametrically opposed to linguistic methodology in the generative tradition, which attempts to identify non-trivial generalizations or implications that hold across languages and thus help pre-structuring the search space for the language learner.¹

3 Recent developments

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:²

(i) The standard NLP approach to multi-level analysis (e.g., part-of-speech tagging, morphological analysis, syntactic constituent and/or dependency parsing, semantic role labeling, coreference resolution) is to assume a pipeline of separate input-output modules, each solving a single intermediate step the output of which is then fed into the next step. This is conceptually perspicuous, it avoids additional algorithmic complexity, and allows for module-specific supervised training. However it comes at the cost of error propagation. This has recently prompted considerable attention on “joint modeling”, i.e., effective ways of solving combined problems that span more than one step in the classical pipeline (for tractability, the joint modeling is often approximated using some flexible combination of modules). Examples of task combinations are morphological segmentation and parsing (Goldberg and Tsarfaty, 2008), part-of-speech tagging and parsing (Bohnet and Nivre, 2012), morphological disambiguation and parsing (Seeker and Kuhn, 2013a), syntactic and semantic parsing (Li et al., 2010), and, in the reverse direction, referring expression generation and surface realization (Zarrieß and Kuhn, 2013).

(ii) If some approximation of a joint model is assumed, how can the “candidate set” of intermediate results be best represented? One may for instance assume some (underspecified or packed) representation of the exhaustive list of candidates, or a k -best list according to some preliminary scoring, possibly combining candidates from different alternative modules. Björkelund et al. (2013) for instance use output from various parsers to populate a candidate set for ranking, achieving state of the art results for parsing across various “morphologically rich languages”. Depending on the data structure, one may even be in a position to combine partial analyses by a technique sometimes called “blending” (Sagae and Lavie, 2006; Hall et al., 2007).

¹Note however that since both frameworks are motivated by learning/learnability considerations, they *could* be related to each other at a substantial level – the differences can be argued to be mainly in prioritizing the step-by-step lifting of one’s idealizing working assumptions.

²The citations given in this listing are not intended to be exhaustive. Given that this is an individual position paper, there is a bias towards examples of work from my group and our department. This implies by no means that I think there is no other, more important work.

(iii) Related to the previous points, a question arises for applications involving only a level of analysis that is relatively far “downstream” in the pipeline: if the typical pipeline could build on alternative intermediate representations, which do not affect the outcome directly – how can one decide on the type that should be chosen? For example, should constituent or dependency parses, or both, be used for the task of coreference resolution (Björkelund and Kuhn, 2012); how should morphological segmentation be addressed in “morphologically rich” languages (Goldberg and 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 and Knight (2012), for instance, propose a probabilistic model for Machine Translation that uses a semantic feature structure as an intermediate representation, which is in the spirit of earlier LFG work on translation using f-structures (Kaplan et al., 1989; Riezler and Maxwell, 2006).

(iv) Can latent representations of intermediate levels be induced – e.g., for inducing semantic properties in a grounded learning scenario like in Richardson and Kuhn (2012) or for adjusting parsing models across languages, Titov and Henderson (2010)? If so, is the induced latent representation superior to an established intermediate representation, which can be trained directly, evaluated and tuned (to the extent that annotated resources are available)?

(v) Are there systematic linguistic constraints that can be exploited for improving a data-driven component, exploiting structural building blocks of linguistic expressions and detailed knowledge about the synchronization across (underspecified) interface representations? And can the relevant constraints be formulated in a way that they carry over across typologically different languages?

For the questions under (v) I will provide a relatively detailed illustration from the study in Seeker and Kuhn (2013b) in section 6 below.

Note that none of the approaches mentioned are modeled 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 one another (depending on the assumed architecture) has been growing as outlined above, questions about appropriate interface representations do however gain crucial importance. So, when it comes to deciding on a global model architecture spanning 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, as is sketched in Figure 1:

LFG	Data-driven NLP
c-structure categories	part-of-speech tags
morphological f-structure features	morphological analysis
c-structure trees	constituent syntax
f-structure embeddings (minus functional control)	dependency structure
a-structure (incl. functional control)	semantic role labeling
anaphoric control	coreference resolution

Figure 1: Rough correspondences across levels of representation

In addition, we can note that 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 modeling 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 gains importance: is there a combination of interface representations that allows for effective modeling of arbitrary languages? This does not seem to be all that different from linguistic research in the generative tradition.

4 The broader picture for interface representations in linguistic modeling

In this and the following sections, I take a few steps back to develop a high-level picture of the role of representations (and in particular interface representations assumed for interacting “modules”) that is broad enough for capturing linguistic work on the theory of grammar on the one hand and data-driven computational work in Natural Language Processing on the other hand.

This line of reasoning is closely connected to the Stuttgart collaborative research center SFB 732 *Incremental Specification in Context*, in which linguists and computational linguists from distinct research paradigms have been successfully cooperating. This SFB has been set up to depart from one of the most characteristic properties of natural language(s) and the human language faculty: the high

degree of ambiguity in linguistic expressions and the mostly effortless ability of speakers and hearers to deal with it when the expressions are contextually embedded.

Any model of language interpretation in the face of ambiguity will follow the general scheme in the top half of Figure 2; models of choice in language generation follow the same scheme in the reverse direction, as seen in the bottom half.

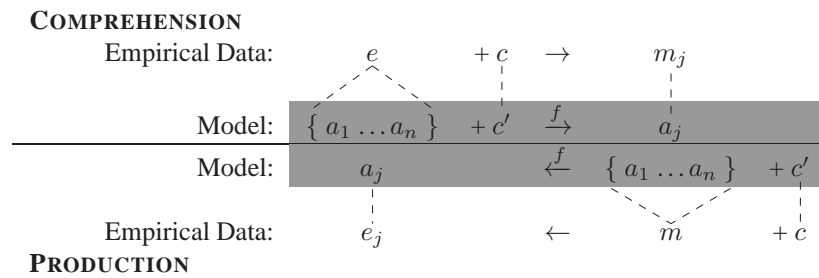


Figure 2: The general modeling scheme for specification in context

The observable empirical process in comprehension is a hearer's ability, given some linguistic expression e that she is confronted with in a particular context c , to decide which is the appropriate interpretation m_j among a large set of interpretations which e could have in different contexts. Any theoretical or computational model characterizes the input expression e in the empirical data by a set of alternative analyses of this input and assumes some appropriate representation c' of the empirically observed context c . At the core of the model is some function f which picks out one analysis a_j among the alternatives, given context c' . The form of representation of the competing analyses, and in particular of the target analysis a_j is chosen in such a way that a_j contains a representation of the hearer's interpretation m_j (e.g., $a_1 \dots a_n$ may be different syntactic trees for an observed string, and one of the trees reflects the structure that hearers find natural in the given utterance context). In the same way, the reverse process models a speaker's choice among possible expressions for realizing some underlying thought or message in a given utterance context.

The representations and functions assumed for a particular model process are chosen in a way that they satisfy certain meta-theoretical principles and allow for the prediction of some corpus of empirical language data. We will go into details of the modeling choices below, but note at this point that a wide-spread objective is to follow some principle of economy. For a process of specification in context, economical modeling can often be fleshed out as follows: Rather than assuming an explicit listing of the entire set of choices $\{a_1 \dots a_n\}$ prior to contextual disambiguation, the representation language is designed to provide a compact representation for this set – this is the widespread notion of *underspecification* in linguistic modeling, especially in its symbolic guise. Design decisions in probabilistic modeling are typically influenced by additional constraints, such as the attempt to exploit the information available in a given data sample/corpus in the best possible

way for deriving generalizations, without overfitting the model parameters to the training data.

In summary, the relationship between the two levels of representation is generally determined by meta-principles and a combination of considerations, which can have various forms depending on the theoretical framework.

Adopting a plain and simple common schematic core structure for all approaches to ambiguity in language is very useful for identifying the commonalities (and distinctions) between alternative approaches in the study of language – across disciplines, theoretical paradigms, and language families and languages. While the entities, representations and functions/processes under consideration may differ, the common scheme of specification in context makes it possible to pinpoint systematic similarities and differences – for instance the potential/justification for using underspecification in different modeling tasks.

5 The internal interface architecture of models of specification in context

The schematic process in Figure 2 captures the ordinary language notion of *ambiguity*: many natural language expressions can have various different interpretations or *readings*, but language users normally have the competence to pick a single one (or, more generally, reduce the set of choices) in a given context of usage.

In order to be able to model this process systematically, the relevant properties of expressions have to be accessible in some representation, and since various properties are known to interact in the process of context-sensitive specification, or disambiguation, the simple scheme requires some further explication. To capture different properties in the general case, each of the representations a_i from the set $\{a_1 \dots a_n\}$ of candidate analyses for some expression e can be thought of as a bundle $\langle \ell_i^1, \ell_i^2, \dots, \ell_i^k \rangle$ of properties – maybe at k different levels of linguistic representation, or layers, so ℓ_1^3 may be the constituent syntax representation for reading 1 of an utterance e , ℓ_2^3 the representation of a different reading of e , and ℓ_2^5 the corresponding representation at some more abstract linguistic level.

Since the cognitive process of picking a particular reading in context is extremely complex (and for instance involves extra-linguistic knowledge), it is common to focus attention on a subprocess with defined linguistic interface representations, typically relating two (or more) established levels of linguistic descriptions, such as syntactic constituent structure and grammatical functions, etc. The subprocess can then, quite conveniently, be seen as a small-scale version of the full process; and it suggests itself to construe the full process as a cyclic chain of formally similar subprocesses, as indicated in Figure 3.

The underlying assumption is that at each layer i , a specification process f^i reduces a set of possible alternatives $\{\ell_1^i \dots \ell_{n_i}^i\}$ for this layer to a particular choice $\ell_{j_i}^i$, which then again defines the choice of options for the next layer $i + 1$, and so on. Note that if we view the cascade as a series of contextually driven specification

COMPREHENSION

Empirical Data:

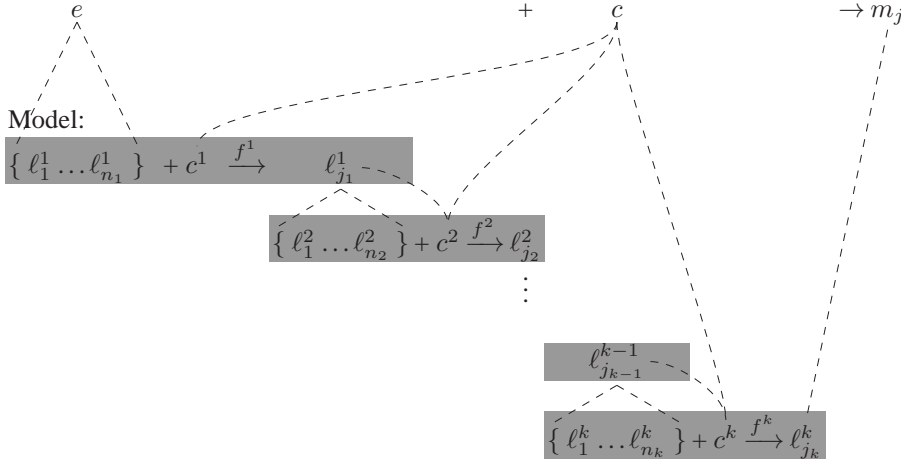


Figure 3: The cyclic (pipeline) model of specification in context

steps, the relevant context for each step is not just determined by the empirically observed (presumably largely extra-linguistic) context c , but each layer contributes highly relevant bits of information for the specification context at the next layer. For instance, layer 2 may be the level at which inflectional feature values such as number (on verbs with subject agreement and on nominal elements) are determined, and layer 3 may be the level at which the syntactic structure for this input string is determined. Then, due to agreement constraints, the feature values determined in layer 2 will affect the specification in layer 3.

Classical feature underspecification at intermediate levels of representation is typically motivated by the observation that certain choices stay open across layers at which the relevant feature type would normally be resolved. Clearly, the modeling decision for interface representations is intimately tied to the assumed sequence of cyclic specification decisions, i.e., the architectural design. Modeling alternatives can be decided on the grounds of economical considerations.

The cyclic specification sequence goes along with strong assumptions: growth of specificity has to follow the same sequence across layers for all analysis problems; in a classical pipeline architecture, specification decisions cannot normally be undone later. Often, the contextual clues at a particular layer give strong indications for a certain specification, but the decision can be overridden later. This effect cannot be modeled appropriately in a plain pipeline. While earlier work in Generative Linguistics (e.g., the GB model) was based on a clear concept of subsequent levels of information, more recent models (Minimalism and Distributed Morphology) have abandoned the idea of a step-by-step sequence of specification. Largely, problems of ambiguity are resolved at the interfaces with the articulatory and the perceptual system, respectively.

Despite the conceptual limitations tied to the strong implications for the sequence of specification decisions, pipeline models typically form the baseline systems in data-driven approaches in Natural Language Processing. Here, a layer corresponds to some analysis tool trained on annotated corpus data following the classical levels of linguistic representation. When applied on new input data, the tools make no strict choice of specification, but assign probability scores to the various options. In the typical pipeline setup, the highest-scoring prediction is passed on to the next layer, which may of course occasionally bring a subsequent layer in the situation where it can no longer make correct predictions – even though it may locally have strong evidence available.

As has become obvious, the pipeline architecture is not fully adequate to model situations where two independent linguistic subsystems interact in constraining the space of possibilities of further specification. An alternative abstract architecture is the joint model sketched in Figure 4, which does not pre-specify any particular sequence of subsequent specification, but posits a simultaneous decision, in principle allowing for arbitrary global interaction across layers.

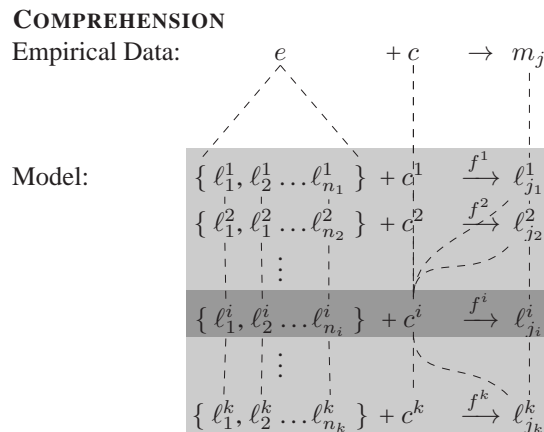


Figure 4: The joint model of specification in context

In the joint model, any subprocess of layer-specific specification can (at least abstractly) be informed by the output of any other subprocess; i.e., effectively the specification decisions mutually contribute context information for each other.

Of course, any concrete model following this idealized setup has to break up the circularity in its design. Moreover, complex model architectures can combine ideas from the pipeline and from the joint model, yielding a vast space of possible system architectures. This essentially characterizes the architectural status quo for several of the recent research questions in NLP, addressed in section 3.

Let me now come back to the question of representational interfaces. We note that despite considerable methodological differences, the various approaches tend to “meet at” common interface representations – mostly the classical levels of linguistic descriptions, such as segmental and prosodic phonological representations, representations of core aspects of morphological and syntactic structure, and mean-

ing representations of key notions of semantic interpretation.

This representational interfacing has been a key element for many of the successful examples of collaborations between linguistics and Natural Language Processing, providing hubs both for model combination across layers and for cross-paradigm comparison (or combination) of models addressing the same layer with alternative approaches. However, as research has proceeded to address advanced implications of modeling decisions, the fields are at a point where one can and should start lifting some of the simplifying assumptions – such as the assumption that the interface representations can be carried over without adjustment from one research paradigm/modeling approach to another. While it is convenient to use an existing treebank annotation for training a submodule, it is unlikely that the assumptions that informed the original annotation guidelines do actually hold in all respects for the context in which the trained submodule is currently supposed to be used.

The set of recently developing research questions (section 3) is a sign that a process of re-thinking the simplifying architectural and representational assumptions is happening. By looking at such activities with the coordinate system from classical linguistic modeling in mind, there is a (maybe somewhat unexpected) chance to take advantage of lessons learned from theoretical work in linguistics – or, as I am pointing out in this paper, the ‘spirit of LFG’.

To provide a more concrete illustration of how one should imagine this linguistically informed view on recent data-driven modeling, I will briefly review some high-level conclusions from the study by Seeker and Kuhn (2013b), which shows very clearly that it can be crucial to question the justification of particular representational decisions even in well-established standard NLP scenarios.

6 Questioning standard representations – an illustration

Data-driven parsing of text from the newspaper domain is one of the most established standard tasks in modern NLP. People have spent two decades on improving the processing pipeline to achieve the best possible results for syntactic analysis (and in particular contextual disambiguation) in a domain for which sizable amounts of hand-labeled training data, so-called treebanks, are available. The supervised scenario makes it possible for machine learning to exploit clues from various levels of representation for the disambiguation decision, possibly including non-linguistic statistical tendencies from the real-world domain – such as the fact that it is more likely for managers to employ people than for athletes.

Most recent advances in this task have thus been due to techniques that allow machine learning methods to capture more and more complex and potentially subtle constellations of contextual clues for the disambiguation decision. For instance, in statistical dependency parsing, the complexity of exploitable machine learning features has been subsequently increased to include combinations of two or three dependency arcs (some relevant contributions were Carreras (2007); Koo

and Collins (2010); Bohnet and Kuhn (2012)). Training a reranker among the most promising candidate analyses from an initial parser has been very successful in constituent parsing (the best known work here is Charniak and Johnson (2005), but the technique is widely used, and computational work in the LFG framework was among the first studies to this end (Riezler et al., 2000, 2002)).

Although it is acknowledged that the choice of appropriate linguistic representations is also important, most researchers assumed that after many years of tuning, no further improvements could be made by adjusting the representational assumptions behind the standard task (which depends on the initial decisions made in the original treebanking effort).

For the syntactic parsing task, the ultimate structural disambiguation decision depends on a constellation of morphosyntactic features (among other things): what are the inflectional person/number features of a verb, what is the case and number of a noun, etc.? The German sentence provided in figure 5 below for example is in OSV order, and it is not just selectional preferences of the verb that indicate this order; the inflectional marking of the initial determiner, in combination with the noun, makes it a clear accusative, and in addition, there is plural subject agreement on the verb.

Since the morphological features cannot be read off deterministically from the surface form (the two nouns in the example, by themselves, are ambiguous between nominative and accusative), data-driven parsing is typically split up into several subtasks: part-of-speech tagging, morphological tagging and structural dependency parsing. Although in morphological disambiguation, strict grammatical rule knowledge plays a more decisive role than in syntactic disambiguation (where it is obvious that data-driven techniques are needed to help with certain decisions), it has been established that training the inflectional disambiguator on the full sentences is important for achieving competitive results. One may intuitively imagine that the system can pick up corpus-specific statistical tendencies in the linearization of certain forms, so the morphological disambiguator effectively anticipates certain syntactic decisions.

Ideally, the syntactic parser should be able to override decision from an earlier morphological step, but the search space of all possible combinations of inflectional options for the words in a sentence is huge (at least for languages with a relatively rich inflectional morphology that displays syncretism). Even with a fairly large manually annotated treebank, a parser that is learning to make structural decisions and inflectional decisions jointly, without any side constraints, has too few indications from the corpus data to pick up some important patterns. Thus, it is more effective to break up the task in a pipeline of two subsequent decisions. The occasional errors in the earlier inflectional decisions are outweighed by the majority of cases where there are enough helpful clues available in the surface string; the second step, syntactic disambiguation can thus use the available training data very effectively for making decisions within the inflectionally constrained search space. Even the idea of allowing the syntactic parser to fall back on the second or third most likely morphological analysis of a word will typically lead to an overall

decrease in parsing performance (on the standard in-domain test data).

As the considerations and experiments in Seeker and Kuhn (2013b) show, there is however one point in the established standard setup for data-driven parsing that *can* be improved substantially, and this is where questioning the representations assumed at the interfaces between subprocedures is highly effective.

As just discussed, training a morphological disambiguator in the full sentence context is empirically superior to a purely symbolic approach that will feed the syntactic parser with all possible morphosyntactic options for each word. Conveniently, a syntactic treebank can be used as gold-standard data for this task too: the treebank annotators used their grammatical knowledge and understanding of the real-world context to determine exactly the right inflectional feature values for each word in the string. For instance, each word in the German NP *das kleine Auto* (“the small car”), when used in a treebank context like *the small car is parked in a side street* will receive an unambiguous marking for case, number and gender, although each form is ambiguous in several ways, and even the full NP – out of context – is ambiguous between nominative and accusative. Of course, the morphological disambiguator trained on such fully disambiguated data will not be perfect – it will occasionally make incorrect predictions for a syncretic inflectional form – but it can nicely exploit the tendencies mentioned above.

There seem to be just two alternatives: either feeding a subsequent syntactic parser with the word-by-word predictions of the statistically trained morphological disambiguator, or leaving all but the strictly symbolic decisions to the syntactic parser. This however ignores what in modern linguistics has been captured by the notion of underspecification, and cyclically increasing specification as more information becomes available: in the space of available analyses for the full sentence string, the adjective *kleine* as part of the NP *das kleine Auto* cannot be plural – although the word form certainly can. It is the local phrasal context that partially disambiguates the word forms. However, recognizing the three-word sequence as an NP still leaves the case feature for each of the three words open: they could be nominative or accusative (but all of them have to be the same).

The standard architectural scenario for input-output training of (token-based) machine-learning classifiers does not provide a basis for capturing this middle ground state of information half-way between the full set interpretation options for an ambiguous (syncretic) word form and the one contextually singled out interpretation. Note that it is not just a question of assigning different weight to potential interpretation options: a machine-learning classifier can quite well provide a probability distribution over the potential morphosyntactic feature values in the given string context. But this still does not capture the interdependencies across words (mediated through the syntactic structuring decision): if *kleine* ends up accusative, *Auto* will not end up nominative, unless there is some other syntactic structure in which they do not form an NP.

Of course it is not at all trivial to set up an architecture that is able to capture these very interdependencies, while at the same time still taking advantage of the statistical tendencies when predicting the distribution over possible inflectional

values/syntactic structures. But it is possible, using the meta-level framework of Integer Linear Programming (ILP) for navigating the search space of interrelated options, as Seeker and Kuhn (2013b) show. The present position paper is not the place to go into the technical details of how this works. The crucial point is that the frameworks allow a linguistically informed modeller to express interrelations across the choices for the various word-level and sentence-level decisions. The knowledge that NPs are a structural domain with important implications for the distribution of inflectional features within the sentence string is thus a priori knowledge that the machine learning system does not have to pick up from the available training data. Moreover, the way in which syncretic forms hold for several cells in the morphological paradigms for various inflectional classes can be stated explicitly as a quasi-underspecification of combinations of inflectional features. And finally, certain aspects of verb subcategorization can be enforced when assembling the predictions for a verb’s arguments – mainly what corresponds to LFG’s functional uniqueness principle: there cannot be two arguments that are *subject*.³

	Diesen	Unterschied	sehen	die	Versuchspersonen
	<i>this</i>	<i>distinction</i>	<i>sehen</i>	<i>the</i>	<i>experimental.subjects</i>
1.	ACC	NOM/ACC/DAT		NOM/ACC	NOM/ACC
2.		----- NP -----		----- NP -----	
3.	→	NOM/ACC/DAT		NOM/ACC	NOM/ACC
4.			→	NOM/ACC	NOM/ACC
5.		OBJECT		SUBJECT	

Figure 5: Illustration of the intuitive flow of information for disambiguation

The intuitive flow of information is sketched for a locally ambiguous (but globally unambiguous) German example in Figure 5, ignoring morphosyntactic features other than case. Both nouns in the sentence are case ambiguous by themselves. Since the dependency parser considers possible readings with the transitive main verb, it is clear that there must be two NPs. The NP *diesen Unterschied* is disambiguated by the demonstrative determiner. Because of functional uniqueness, this excludes the accusative/object reading for *die Versuchspersonen*, although it is locally ambiguous.

Figure 6 sketches more far-reaching interactions involving different inflectional paradigms for German adjectives and certain nouns, depending on the choice of determiner, plus the interaction with subject agreement on the finite verb. Note that the final noun has the same form in all instances (*Befragte*), but this is a syncretic form that could be nominative/accusative of feminine singular or plural.

In the transition from the second to the third variant of the example, changing only the word *viele* (“many”) to *die* (the definite article) turns an ambiguous sentence into an unambiguous one, because the article forces the de-adjectival noun

³Any more far-reaching subcategorization constraints tend to be empirically problematic, because of a fair degree of variation in real corpus data.

Stuttgarts	größten	Verein	erwähnten	viele	Befragte
<i>Stuttgart's</i>	<i>largest</i>	<i>club</i>	<i>mentioned (3pl)</i>	<i>many</i>	<i>interviewees</i>
----- OBJECT -----			--- SUBJECT ---		
Stuttgarts	größte	Vereine	erwähnten	viele	Befragte
<i>Stuttgart's</i>	<i>largest</i>	<i>clubs</i>	<i>mentioned (3pl)</i>	<i>many</i>	<i>interviewees</i>
----- ? -----			----- ? -----		
Stuttgarts	größte	Vereine	erwähnten	die	Befragte
<i>Stuttgart's</i>	<i>largest</i>	<i>clubs</i>	<i>mentioned (3pl)</i>	<i>the</i>	<i>interviewee</i>
----- SUBJECT -----			--- OBJECT ---		
Stuttgarts	größte	Vereine	erwähnte	die	Befragte
<i>Stuttgart's</i>	<i>largest</i>	<i>clubs</i>	<i>mentioned (3sg)</i>	<i>the</i>	<i>interviewee</i>
----- OBJECT -----			--- SUBJECT ---		

Figure 6: Illustration of subtle interactions between inflectional paradigms, NP-internal agreement and subject/verb agreement

Befragte to be a feminine singular rather than a plural. To complicate things some more, it is not the article in *die Befragte* alone that enforces this reading. With a singular subject agreement on the verb, as in the forth variant, the subject/object interpretation in the complete sentence is flipped, since all word forms in the NPs are syncretic for nominative/accusative. This illustration should give some intuitive indication that a syntactic parser that does not have to rely on some locally informed morphological prediction may have a real empirical advantage.

	Czech			German			Hungarian		
	NO-C	C	PRED-M	NO-C	C	PRED-M	NO-C	C	PRED-M
subject	85.41	87.23*	85.46	90.02	92.91*	90.59	85.05	87.67*	86.53
predicative	87.13	90.09*	87.11	72.86	80.70*	74.33	74.16	78.88*	74.79
obj (nom)	47.48	53.19*	38.74	–	–	–	–	–	–
obj (gen)	70.15	72.54	70.27	31.41	42.98	34.26	–	–	–
obj (dat)	79.99	80.42	79.54	65.21	77.78*	71.05	75.33	77.92*	73.49
obj (acc)	84.27	86.79*	84.12	83.74	87.96*	84.86	91.96	93.21*	92.53
obj (instr)	67.36	68.76	65.02	–	–	–	–	–	–
all arg funcs	84.33	86.37	84.21	86.27	90.11	87.24	86.87	89.04	87.78
all other	81.37	81.37	81.05	89.79	89.88	89.98	82.73	82.86	83.43

Table 1: Table from Seeker and Kuhn (2013b): Parsing results for the unconstrained (NO-C) and the constrained (C) ILP models, and the Bohnet parser with predicted morphology output (PRED-M) in terms of labeled attachment f-score.

Table 1 shows a summary of the results that Seeker and Kuhn (2013b) report for the different variants in the combination of data-driven syntactic and morphological models.⁴ The datasets considered were standard treebanks for Czech, Ger-

⁴* marks statistically significant differences when comparing the performance on a grammatical function for the C model to the PRED-M model.

man and Hungarian (three languages with relevant case marking in NPs). The ILP-constrained search (system output C) of the combination of predictions from the morphological disambiguator and a dependency parser leads to a significant improvement for the (case-marked) argument functions when compared with a parser that does not use the linguistically informed constraints (system output NO-C). The constrained combination is also superior to the state-of-the-art Bohnet parser (Bohnet, 2010), provided with morphological predictions as input to syntactic parser (system output PRED-M): Note the increase of more than two percentage points for the argument functions (“all arg funcs”) in Czech and German over the morphological prediction based parsers (84.21 → 86.37 and 87.24 → 90.11).⁵ For the other functions, which are independent of case marking, no decrease of performance is incurred. These overall results are quite remarkable, given that the baseline systems are very competitive parsers.

To conclude this section, we have seen an example of advanced NLP research that takes advantage of principled linguistic knowledge about the interaction across interface representations in the setup of a modeling architecture. Running the best available machine learning techniques on gold standard input/output pairs alone does not suffice. For best results, it is crucial to know about the role of intermediate interfaces and the status of the corresponding representations.

7 Conclusion: Is it the ‘spirit of LFG’ rather than the spirit of linguistic thinking more generally?

The interactions are subtle and the technical solution is quite involved – but I think the discussion of Seeker and Kuhn (2013b) in the previous section makes it clear that even in advanced statistical NLP modeling, targeted, linguistically informed constraining of the search space can have a very noticeable effect. Reaching the same effect in an exclusively data-driven way would be extremely hard, even when powerful general-purpose machine learning techniques are applied for picking up the constraints from training data: there is always just a limited amount of high-quality training data, and if an unconstrained model has to *learn* that case/number/gender feature agreement occurs inside of NPs, and person/number agreement holds between subjects and inflected verbs (but no agreement occurs in other configurations), the “signal” in the same data cannot be used to induce other important generalizations.⁶ In particular, the wide-spread syncretism in the morphosyntactic feature paradigms tends to blur many of the data points, so pre-structuring the space in terms of underspecified abstract representations makes the

⁵The advantage from the constraints is least pronounced for Hungarian, which has very few cases of syncretism in its inflectional paradigms.

⁶One might speculate that some of these configurational hard constraints reflect an aspect of Universal Grammar, but I do not want to go into this here. Note that the standard treebank-trained parser experiments may fail to reflect some bootstrapping scenario which human language learners are exposed to and which *does* allow for a more data-driven induction of the relevant constraints.

available data much more informative.

The critical reader will probably think, alright, this shows the importance of linguistic awareness about important interface representations even for heavily data-driven NLP – but is there really any specific point related to the ‘spirit of LFG’ that can be made? Or in other words, there *are* similarities for various LFG levels of representation and important interface representations in recent NLP work, as sketched in Figure 1 on page 6 – but these may reflect some rather unsurprising convergence which any empirical account has to undergo sooner or later, simply to capture the systematic patterns in the data!? After all, the affected levels also resemble traditional levels from descriptive grammar. So it would seem that similar parallels as listed in Figure 1 could be drawn for any other grammatical paradigm, especially constraint-based ones (such as HPSG or CCG) that share the conceptual view of simultaneous interaction across interface representations.

To a certain degree, this reservation is of course justified: if we look just at the representations at the established “hub” levels of grammatical analysis, and we ignore differences in the theoretical assumptions and mechanisms that different approaches assume to relate them to one another, we must expect structural similarities across all approaches at a relatively high level of abstraction (maybe with the exception of heavily derivational approaches).

However, I would nevertheless like to make a stronger point, and I think it is justified to argue that LFG *is* closer than other established grammatical frameworks to the emerging picture in NLP research sketched in section 3 and discussed in more detail in sections 5 and 6. I think there is an explanation for this circumstance at the level of sociology of science. Throughout its development, the LFG framework has been shaped in an interdisciplinary dialogue, involving theoretical linguists, descriptive linguists and computational linguists (as discussed in section 2). This circumstance can have an effect on the characteristics of the canonical interface representations that are being established: if one particular modeling goal dominates the design process, principles advocating formal uniformity and theoretical simplicity (which are of course important in any systematic approach) will have a stronger effect than in a multi-disciplinary setup. In the latter case, new uniformity assumptions about some level of representations will immediately prompt a debate if they are not compatible with the various points of view that the representation is relevant for. So, empirically grounded applicability of the interfaces ranks higher than aesthetic/theoretical considerations of cross-level uniformity (and as a side-effect, a multidisciplinary framework may be somewhat more conservative and keep up established assumptions). At points in time when it turns out that relevant interactions across levels are more complex than previously assumed (like in the examples discussion in section 3), this has the advantage that new accounts do not have to work around simplifications that are orthogonal to the issue under consideration.⁷

⁷To give an example, LFG has generally used relatively “flat” collections of features in the f-structure, whereas HPSG has established sophisticated, hierarchically organized feature structures bundling groups of features. Using inheritance hierarchies over sorted feature structures, the HPSG

Even if this explanation is not correct, it is a fact that LFG has consciously adapted an extensible projection architecture of heterogeneous representation structures. This allows users of the framework to consider alternative paths in the connection between representational layers. By assumption, all layers are in parallel correspondence, so there cannot be any non-monotonic effects that would strictly exclude more indirect cross-level effects (e.g., c-structure information becoming relevant for some decision that is normally made exclusively at f-structure). Yet, it is considered to be the scientific goal to identify the systematic, “direct” effects that are behind the empirical generalizations derived from the data.

And it is this architecture of parallel correspondence across formally heterogeneous representation structures represents that I would characterize as the ‘spirit of LFG’ in the present context. More than most other frameworks, LFG has avoided superimposing theory-internally motivated meta principles on the modeling approach and has thus kept an open eye on how empirically observable effects can help to make a design decision in one way or another. As a consequence, a considerable number of LFG contributions have looked at a network of levels of representations (the LFG projections) in an explorative way, trying to find arguments that help decide what are the most fundamental correspondence relations across levels, and what are the derived relations. As a matter of fact, with the availability of inverse projection functions and functional composition of projections, there are essentially no effects that cannot be modeled at least indirectly. Examples of relevant considerations are the questions under what circumstances the inverse of the ϕ projection (from c-structure to f-structure) is needed (e.g., Halvorsen and Kaplan (1988/1995); Bresnan (1995) and more recently Asudeh (2009)), or whether some local level of morphological structure is projected from c-structure or from f-structure (Butt et al., 1996; Frank and Zaenen, 2002). In computational work in the LFG framework, there have also been discussions of alternative approaches to the same underlying problem, e.g., disambiguation using symbolic constraint ranking (Frank et al., 2001) vs. data-driven training of discriminative models (Riezler et al., 2002); for disambiguation, it is a combination of the respective strengths of approaches that is effectively being applied in the large-scale ParGram grammars.

LFG’s way of *not* treating any particular architectural assumption as a strict given opens up the research paradigm to the possibility of what one might call conditional interface effects, i.e., seemingly incompatible cross-level effects that could not be explained by a single pipeline sequence of modules.⁸ As I also ar-

formalism can thus model theories about cross-feature interactions in an aesthetically appealing way, and more uniformly than LFG. It turns out however that it is extremely difficult, if not impossible, to establish a single hierarchical structuring of the feature geometry that is consensus among different points of view, as is reflected in long debates. In LFG, the *representations* of linguistic expressions were tentatively chosen in a way that does not reflect sophisticated theoretical assumptions (these are rather reflected in the constraints or *descriptions* of the objects, e.g., the mechanism of functional uncertainty).

⁸I would say that other constraint-based frameworks have made stronger commitments in terms of turning one particular assumption into a guiding representational principle – enforcing that other assumptions will be subordinate to it. (Of course, such statements are always somewhat subjective.)

gued in Kuhn (2007), it is a strength of constraint-based approaches to the theory of grammar that different systematic effects involving the same interface representation may quite well be based on different interface-to-interface correspondence, without enforcing the prediction that one of them is more fundamental than the other. Another way of stating this observation is that LFG never assumed a strict pipeline architecture, according to the classification from section 5 – which is of course what can be expected from a non-derivational approach, at a technical level, but the observation is also true at a more abstract conceptual level.

The benefit of being able to use “conditional interfaces effects” in the modeling extends quite naturally from classical symbolic modeling of the interface-to-interface relations, using some logic language to input-output modeling as it is done in current machine learning work.⁹ Hence, the observations made throughout this paper about more and more cross-level relations (seemingly) deviating from the step-by-step processing pipeline are not at all surprising from the LFG point of view of the architecture of grammar and interfaces.

So, in conclusion, as far as I can see, LFG’s architecture of parallel correspondence 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.

References

- Asudeh, Ash. 2009. Adjacency and locality: A constraint-based analysis of complementizer-adjacent extraction. In Miriam Butt and Tracy Holloway King (eds.), *Proceedings of the LFG09 Conference*, pp. 106–126, Stanford, CA: CSLI Publications.
- Björkelund, Anders, Çetinoğlu, Özlem, Farkas, Richárd, Müller, Thomas and Seeker, Wolfgang. 2013. (Re)ranking Meets Morphosyntax: State-of-the-art Results from the SPMRL 2013 Shared Task. In *Proceedings of the 4th Workshop on Statistical Parsing of Morphologically Rich Languages: Shared Task*, Seattle, WA.

For example, HPSG has decided to generalize the idea of systematic head-driven/intrinsic determination of the phrase-structure configuration – X bar theory and its extensions – in such a way that it stopped to provide any other way of specifying phrase-structure rules. CCG has given systematic principles in the lexicon (for which there is good empirical evidence) precedence over the entire grammatical theory. LFG is more pluralistic in this respect, it provides various points of entry for capturing some generalization.

⁹The same holds for a different target direction in which likewise the ‘spirit of LFG’ allows for a natural extension (which I did not consider in the present context as I focused attention on computational linguistics/NLP with a relatively technical motivation): this is the recent work on Probabilistic Grammar by Joan Bresnan and colleagues.

- Björkelund, Anders and Kuhn, Jonas. 2012. Phrase Structures and Dependencies for End-to-End Coreference Resolution. In *Proc. of COLING 2012: Posters vol.*, pp. 145–154.
- Bohnet, Bernd. 2010. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 89–97, Beijing, China.
- Bohnet, Bernd and Kuhn, Jonas. 2012. The Best of Both Worlds – A Graph-based Completion Model for Transition-based Parsers. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 77–87, Avignon, France.
- Bohnet, Bernd and Nivre, Joakim. 2012. A Transition-Based System for Joint Part-of-Speech Tagging and Labeled Non-Projective Dependency Parsing. In *Proceedings of EMNLP-CoNLL 2012*, pp. 1455–1465.
- Bresnan, Joan. 1995. Linear Order, Syntactic Rank, and Empty Categories: On Weak Crossover. In Mary Dalrymple, Ronald M. Kaplan, John T. Maxwell and Annie Zaenen (eds.), *Formal Issues in Lexical-Functional Grammar*, pp. 241–274, Stanford, CA: CSLI Publications.
- Butt, Miriam, Dyvik, Helge, King, Tracy H., Masuichi, Hiroshi and Rohrer, Christian. 2002. The Parallel Grammar Project. In *Proceedings of COLING-2002 Workshop on Grammar Engineering and Evaluation*, pp. 1–7.
- Butt, Miriam, Niño, María-Eugenia and Segond, Frédérique. 1996. Multilingual Processing of Auxiliaries within LFG. In D. Gibbon (ed.), *Natural language processing and speech technology: results of the 3rd KONVENS conference 1996*, pp. 111–122, Bielefeld: Mouton de Gruyter.
- Cahill, Aoife, Burke, Michael, O’Donovan, Ruth, Riezler, Stefan, van Genabith, Josef and Way, Andy. 2008a. Wide-Coverage Deep Statistical Parsing using Automatic Dependency Structure Annotation. *Computational Linguistics* 34(1), 81–124.
- Cahill, Aoife, Maxwell, John T., Meurer, Paul, Rohrer, Christian and Rosén, Victoria. 2008b. Speeding up LFG Parsing using C-Structure Pruning. In *COLING-2008: Proceedings of the workshop on Grammar Engineering Across Frameworks*, pp. 33–40.
- Carreras, Xavier. 2007. Experiments with a Higher-Order Projective Dependency Parser. In *Proc. of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pp. 957–961.
- Charniak, Eugene and Johnson, Mark. 2005. Coarse-to-Fine n-Best Parsing and MaxEnt Discriminative Reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL’05)*, pp. 173–180, Ann Arbor, Michigan.
- Church, Kenneth. 2011. A Pendulum Swung Too Far. *Linguistic Issues in Language Technology* 6(5), 1–27.
- Frank, Anette, King, Tracy Holloway, Kuhn, Jonas and Maxwell, John. 2001. Optimality Theory Style Constraint Ranking in Large-Scale LFG Grammars. In Peter Sells (ed.), *Formal and Empirical Issues in Optimality Theoretic Syntax*, Stanford, CA: CSLI Publications.
- Frank, Anette and Zaenen, Annie. 2002. Tense in LFG: Syntax and Morphology. In Hans Kamp and Uwe Reyle (eds.), *How we say WHEN it happens. Contributions to the theory of temporal reference in natural language*, Tübingen: Max Niemeyer Verlag.
- Goldberg, Yoav and Tsarfaty, Reut. 2008. A single generative model for joint morphological segmentation and syntactic parsing. In *Proceedings of ACL/HLT 2008*, pp. 371–379.
- Hall, Johan, Nilsson, Jens, Nivre, Joakim, Eryigit, Gülsen, Megyesi, Beáta, Nilsson, Mattias and Saers, Markus. 2007. Single Malt or Blended? A Study in Multilingual Parser Optimization. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pp. 933–939, Prague.

- Halvorsen, Per-Kristian and Kaplan, Ronald M. 1988/1995. Projections and Semantic Description in Lexical-Functional Grammar. In M. Dalrymple, R. Kaplan, J. Maxwell and A. Zaenen (eds.), *Formal Issues in Lexical-Functional Grammar*, pp. 279–292, CSLI Publications, reprint from: Proc. of the International Conf. on Fifth Generation Computer Systems, 1116–1122.
- Johnson, Mark. 2011. How relevant is linguistics to computational linguistics? *Linguistic Issues in Language Technology* 6(7).
- Kaplan, Ronald M., Netter, Klaus, Wedekind, Jürgen and Zaenen, Annie. 1989. Translation by Structural Correspondences. In *Proceedings of EACL 1989*, pages 272–281.
- King, Tracy H. 2011. (Xx*-)Linguistics; Because We Love Language. *Linguistic Issues in Language Technology* 6(9).
- Koo, Terry and Collins, Michael. 2010. Efficient Third-Order Dependency Parsers. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 1–11, Uppsala, Sweden.
- Kuhn, Jonas. 2007. Interfaces in Constraint-based Theories of Grammar. In G. Ramchand and C. Reiss (eds.), *The Oxford Handbook of Linguistic Interfaces*, pp. 613–649, Oxford: Oxford University Press.
- Li, Junhui, Zhou, Guodong and Ng, Hwee Tou. 2010. Joint syntactic and semantic parsing of Chinese. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 1108–1117.
- Quernheim, Daniel and Knight, Kevin. 2012. Towards Probabilistic Acceptors and Transducers for Feature Structures. In M. Carpuat, L. Specia and D. Wu (eds.), *Proc. of the 6th Workshop Syntax, Semantics and Structure in Statistical Translation*, pp. 76–85.
- Richardson, Kyle and Kuhn, Jonas. 2012. Light Textual Inference for Semantic Parsing. In *Proceedings of COLING 2012: Posters volume*, pp. 1007–1018.
- Riezler, Stefan, King, Tracy H., Kaplan, Ronald M., Crouch, Richard, Maxwell, John T. and Johnson, Mark. 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)*, pp. 271–278.
- Riezler, Stefan and Maxwell, John T. 2006. Grammatical machine translation. In *Proceedings of HLT-NAACL 2006*, pp. 248–255.
- Riezler, Stefan, Prescher, Detlef, Kuhn, Jonas and Johnson, Mark. 2000. Lexicalized Stochastic Modeling of Constraint-Based Grammars using Log-Linear Measures and EM Training. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL'00), Hong Kong*, pp. 480–487.
- Sagae, Kenji and Lavie, Alon. 2006. Parser Combination by Reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pp. 129–132, New York City, USA.
- Seeker, Wolfgang and Kuhn, Jonas. 2013a. The Effects of Syntactic Features in Automatic Prediction of Morphology. In *Proceedings of EMNLP*, pp. 333–344, Seattle.
- Seeker, Wolfgang and Kuhn, Jonas. 2013b. Morphological and Syntactic Case in Statistical Dependency Parsing. *Computational Linguistics* 39(1), 23–55.
- Titov, Ivan and Henderson, James. 2010. A Latent Variable Model for Generative Dependency Parsing. In H. Bunt, P. Merlo and J. Nivre (eds.), *Trends in Parsing Technology, Text, Speech and Language Technology Series*, Springer.
- Zarriß, Sina and Kuhn, Jonas. 2013. Combining Referring Expression Generation and Surface Realization: A Corpus-based Investigation of Architectures. In *Proc. of ACL*, pp. 1547–1557, Sofia.