Book Reviews

A Grammar Writer's Cookbook

Miriam Butt, Tracy Holloway King, María-Eugenia Niño, and Frédérique Segond
(Universität Konstanz, Xerox Palo Alto Research Center, and Xerox Research Centre Europe)

Stanford, CA: CSLI Publications (CSLI lecture notes, number 95), 1999, xii+244 pp; distributed by Cambridge University Press; hardbound, ISBN 1-57586-171-2, $59.95; paperbound, ISBN 1-57586-170-4, $22.95

Reviewed by
Michael Maxwell
Summer Institute of Linguistics

1. Introduction

Grammar writers sometimes approach grammar writing as if the language being described were the only language in the world. In contrast, this book reports on the parallel development of computational grammars for three languages: English, French, and German. At the time the book was written, the "ParGram" (Parallel Grammars) project included researchers from the Xerox Palo Alto Research Center (California), the Xerox Research Centre Europe (Grenoble), and the Institut für Maschinelle Sprachverarbeitung (University of Stuttgart).

The theoretical approach is Lexical-Functional Grammar (LFG), a theory well suited to parallel development, in that it assumes two levels of grammatical representation:[1] "c(onstituent)-structure" is the traditional phrase-structure analysis, while "f(unctional)-structure" is a representation of argument structure (a surfacy kind of semantic representation). While the c-structures of analogous sentences in two languages may differ substantially, it is possible to analyze their f-structures as being much more parallel. For example, while languages may make more or less the same tense/aspect distinctions, they may differ in whether those distinctions are encoded in auxiliary verbs or in affixes. While such languages necessarily differ in their c-structure, it is possible to collapse those differences in f-structure by treating tense and aspect as morphosyntactic features rather than as lexical items or affixes.

Of course, not all differences between two languages can be collapsed at f-structure. English, French, and German, for example, differ in some of the properties of tense

1 For reasons of brevity, the descriptions given here are approximate. The authors briefly mention two other levels of structure: semantic structure, about which they have very little to say; and "morphosyntactic" structure, which refers to features that are neither functional nor semantic in nature (inflection classes, for example). The situations where the morphosyntactic level are required appear to be very few, and it is not clear in the text why it could not be collapsed into a slightly more articulated f-structure.

and aspect that they choose to represent or ignore. But differences of word order, or whether the encoding of grammatical relations is by prepositions or case-marking affixes or word order, can profitably be abstracted away from at f-structure.

2. Content

After an introduction to the goals of the project, and a brief overview of LFG (which should be comprehensible to anyone who has had a semester or two of syntax), the authors turn to the grammars developed in the ParGram project. This section is the "cookbook" part of the book, and is arranged by construction, with chapters for clauses, verbs and their complements and adjuncts, noun phrases, noun modifiers (determiners and adjectives), prepositional phrases, adverbials (the grab bag category), constituent coordination, and miscellaneous constructions, some commonly treated by traditional linguists (tag questions) and some not ("headers," e.g., newspaper headlines).

As for the depth of coverage in this section, the introduction (p. 1) states:

> We thus provide the potential grammar writer with a handbook in which sample analyses and their linguistic motivations can be looked up and used in the development of further grammars. To that end, we have tried to couch our solutions in terms that are sufficiently independent from the particular framework of LFG.

However, the analyses fall short of this goal. In general, the syntactic description is kept at a nondetailed level; it would not be possible to reconstruct the grammars from the descriptions alone, and probably not from the references cited either (most of which are taken from the LFG literature). To be fair, a true handbook for grammar writers would be a much larger volume than this. Later in this review, I will return to this section, viewing it from a linguist's perspective.

The second part of the book is given over to "grammar engineering," issues such as the user interface to the grammar development system; the measurement of performance; and the particular difficulties of trying to develop parallel grammars of different languages written by different linguists working in different locations. This section occupies less than a third of the book, and could easily have been expanded; too little has been said in the past about these practical issues.

For example, the authors briefly tell how they would debug the grammar if the parser returned a number of analyses for some sentence and it was not apparent whether the right one had been found. Their solution is to annotate the input with a partial c- or f-structure giving the expected analysis, and see whether the parser returns any parses satisfying those constraints. While this problem is not directly related to issues of parallel grammar development, given that the authors raise the issue, the discussion leaves many open questions. The annotation solution allows the grammarian to determine whether the desired analysis exists, but what of the remaining analyses: how do you compare them? My experience is that it is tedious to manually compare pairs of parses, and that comparing three or more parses by eye can be an exercise in frustration. Did the grammarians use a structure differencing utility? If so, how close could it come to characterizing the minimal difference(s) between two c- or f-structures? Also, given the parallel grammar approach, it seems likely that a cross-language f-structure differencing utility would have been useful; was one available?

Some more unanswered questions from this example: Why annotate the input to the parser with the expected structure? Why not build a filter on the output viewer that would show only the analyses corresponding to some expected structure? In case

the desired analysis was not found, was there a debugging utility for searching for partial analyses, or determining at what point the desired analysis failed?

Another topic in this section is how ideas from Optimality Theory (OT) might be integrated with rule-based grammars. What the authors really have in mind here is a sort of preference rating for various constructions. I suspect that adherents of OT might question whether this has anything to do with "real" OT, a theory in which rankings on constraints replace rules entirely. In any case, the use of preference weights on constructions is not new; see, for example, Harrison (1988, 192 ff.).

One chapter treats the use of finite-state technology for morphology and for multiword structures that may be uninteresting to a linguist but which are in practice quite important: for example, proper names, fixed expressions like French *afin que* 'so that', and compound nouns or technical terms that are unlikely to be ambiguous, such as French *arbre de transmission* 'drive shaft'. By preprocessing texts through a finite-state program that tags such multiword phrases as units, the job of the parser is greatly simplified. Of course, one runs the risk here of overdoing the preprocessing, thereby eliminating alternative analyses that may be correct under circumstances not foreseen by the writer of the preprocessing rules. Again, it would have been interesting to know how the grammar writers avoided this problem.

An appendix lists some of the morphosyntactic features that the project members standardized on. It strikes me that there could be a profitable interchange here among computational linguists working in various languages, and also between computational linguists and theoretical linguists. Indeed, the same could be said for the complete grammars developed in such projects, although considerations of proprietary development will doubtless hinder this.

3. A Linguistic Perspective

I have mentioned above a few of the "engineering" topics that could have been expanded, although most are not unique to parallel grammar development. In this section I will mention a few problematical points from a linguistic perspective, none fatal.

Linguists will be puzzled by some claims. For example, the authors state (pp. 96–97) that German has two distinct NP constructions consisting of a determiner followed by something that looks like an adjective. In one, the phrase is analyzed as a headless NP in which an adjective modifies a nonovert noun; in the other, the adjective is said to be nominalized, resulting in an ordinary NP headed by a noun. The text states that the distinction is based on the fact that the word in question is capitalized in the latter but not in the former. But an orthographic convention is never a safe basis for linguistic analysis, and the reader may be left wondering whether there is any real distinction.

Having once written a computational grammar of English, I was at times surprised by what was not accounted for by ParGram's English grammar: ellipticals, nonconstituent conjunction (a footnote on page 139 refers to a proposal here), and stylistic inversion involving auxiliary verbs (e.g., *Never have I seen an example like this*). But on reflection, it seems likely that these constructions are rare (particularly in the corpora dealt with by this project). This is precisely why theoretical linguists are interested in unusual constructions: the language learner rarely runs into them, yet we all have intuitions about them, and to a large extent our intuitions agree. This agreement is often seen as inexplicable unless grammar is partly innate.

My theoretical linguist side also found occasional fault with the chosen analyses. For example, so-called sentential subjects (*That John left surprises me*) are treated

as appearing in the subject position in c-structure. To be sure, a paper (dated 1996) questioning this is cited, but the discussion seems to imply that this is a new idea. The authors write further, "A valid alternative analysis may involve a structural topic position for sentential subjects. However, . . . the development of such an alternative analysis presupposes a large amount of linguistic research outside the scope of the ParGram collaboration" (p. 99). In fact, the shortcomings of the sentential subject analysis have been known at least since Koster (1978), who also proposed an alternative analysis. Do language engineers not read theoretical linguistics?

My practical side, though, had to recognize again that examples that count against the sentential subject analysis are rare—in fact, they may be nonexistent in real texts, because they are for the most part ungrammatical. The misanalysis may therefore be virtually harmless, since the computational grammar never needs to deal with the problem. Once more, this is the very reason theoretical linguists are so interested in these facts: the learner has no exposure to negative evidence, but we all agree on the unacceptability of nonsentences such as ∗*Why does for Bill to smoke bother you?* and ∗*Does that Bill smokes bother you?* If the embedded clauses really were subjects, the unacceptability of these examples would be inexplicable.[2] But how does the learner know this, given that in acceptable sentences, the sentences appear to be in subject position?

A final comment, concerning the overall idea of the ParGram project as a "proof of concept" for parallel grammar development in different languages: This proof would have been more convincing if more-divergent languages had been chosen. An ergative or polysynthetic language would certainly have made life more interesting. On the other hand, perhaps it was wise to save the challenging cases for later—and another book![3]

4. Conclusion

My quibbles are relatively minor, and I can therefore recommend this book to a variety of audiences. For teams of computational linguists developing grammars in multiple languages, it should be required reading. For formal linguists, the emphasis on structures that are actually found in corpora, and on structures commonly ignored by theoretical linguistics, will be enlightening. This book could also be used in a graduate course in syntactic processing, provided it was supplemented by more general texts in linguistic analysis, or grammars with deeper coverage of particular languages.

I hope the project members will continue to report on their experiences as they further develop their analyses and extend the work to other languages, and as they begin using the grammars in actual applications.

References

Harrison, Philip. 1988. *A New Algorithm for Parsing Generalized Phrase Structure Grammar*. Ph.D. dissertation, University of Washington.

Koster, Jan. 1978. Why subject sentences don't exist. Pages 53–64 in Samuel J. Keyser, editor, *Recent Transformational Studies in European Languages*. Linguistic Inquiry Monograph 3. The MIT Press, Cambridge, MA.

2 Compare the following acceptable alternatives that have pronominal subjects and extraposition: *Why does it bother you for Bill to smoke?* and *Does it bother you that Bill smokes?*

3 The web site for the ParGram project (http://www.parc.xerox.com/istl/groups/nltt/pargram/) states that Norwegian, Japanese, and Urdu grammars have now been added.

Michael Maxwell works in the development of computational environments for syntactic, morphological, and phonological analysis at SIL International, and has consulted on the analysis of a variety of indigenous languages. Previously, he developed a computational grammar of English at Boeing Computer Services. Maxwell's address is: 7809 Radin Rd, Waxhaw, NC 28173; e-mail: Mike_Maxwell@sil.org