

English Word Grammar

Richard Hudson

(University College London)

Oxford: Basil Blackwell, 1990,

ix + 445 pp.

Hardbound, ISBN 0-631-16433-2, \$59.95

Reviewed by

Lynne J. Cahill

University of Sussex

The interaction between theoretical and computational linguistics, while not often acknowledged, undoubtedly had some influence in the relative failure of the previous incarnations of Hudson's Word Grammar (also known as Dependency Grammar and Daughter-Dependency Grammar) to capture the imagination in the same way that the context-free phrase structure grammars of the 1980s did. Who needed nice theories of syntax that had relationships between words expressed with pretty drawings when one could have the infinitely more computable phrase structure rules? But Hudson was undeterred, and perhaps in the vein of the mountain coming to Mohammed, he has valiantly attempted in this volume to bring his theory of Word Grammar (WG) closer to computational linguistics and AI. This is no mean feat for someone whose background is firmly in theoretical linguistics, and a lack of familiarity with computational matters is sometimes painfully clear. From the perspective of a computational linguist, some of the explanations and definitions are rather unnecessary, but it must be assumed that the book was primarily written with the theoretical linguist in mind.

The theory of WG originally set out to rival phrase-structure theories of grammar, placing the emphasis on the word (hence the name Word Grammar) rather than the phrase, clause, or sentence. Such concepts played no part in the theory at all, which relied on dependencies between words (hence the original name of Dependency Grammar). Thus, a phrase such as *black dogs eat cats* was analyzed as having three dependency relations: between *black* and *dogs*, between *eat* and *dogs* and between *eat* and *cats*.

The fact that Categorical Grammar has drawn much interest away from CF-PSGs may mean that Hudson's theory now has a more prominent role to play, and this book is certainly thought-provoking, if at times a little frustrating in the questions it asks without answering. The book is divided into two parts, the first seven chapters defining the theory, and the remaining seven providing an account of certain aspects of English, principally syntax with a little morphology and semantics.

The theory has moved on quite substantially from previous versions, not only in its form, but in its coverage. While the original theory was predominantly a theory of syntax, the current version purports to be a theory of knowledge, both linguistic and nonlinguistic. That over half of the book is given over to a (fairly substantial) grammar of English, and is thus very linguistically dominated, is only partially countered by the examples in the theory section relating linguistic phenomena to nonlinguistic concepts that seem to require similar apparatus. These examples are interesting, and the broad argument that the "bits of knowledge" that make up our linguistic competence are only (possibly specialized) cases of our other knowledge is one that deserves closer inspection. It goes without saying that in a book of this size it would be impossible to cover the whole area in enough detail to permit definite conclusions about the

feasibility of the general theory, but it is always disappointing when a work discusses such a wide area in fairly general terms and then goes into detail in only a small part of it.

The upshot of this approach is that the reader is left with a fairly clear idea of what the syntactic component of a WG would look like, but little else. Even the morphology is restricted to simple orthographical suffixation in English regular morphology, giving no clue as to how nonaffixational morphology, as found in sub-regular classes of English words, as well as many other languages, might be handled. Similarly, while examples of nonlinguistic applications of the type of constructions used to express linguistic phenomena are given, it is difficult to imagine what any substantial body of world knowledge might look like. And the total omission of an attempt at phonological examples (although the claim that such information should be encodable in the same way is made) seems a wise move, given that the orthographic rule that allows the doubling of a consonant before a vowel (e.g., *big/bigger*) is:

type of next of complementary of doubling-consonant = vowel

This brings us to the actual formalism used. It is very like English, which has never been known for its succinctness or easy computability. It therefore comes across as something which would appeal to someone who would rather not think in formal terms. The use, in addition, of the pronoun *it*, as in,

complementary of *doubling-consonant* = *it* + *it*

where *it* refers to the *doubling-consonant*, does seem to be taking things a little too far, and when Hudson starts defining the use in some syntactic constructions of the word *it* (in English as opposed to in his formalism) it is easy to get a little confused. I had always thought that the reason for having formalisms that are distinct from the language being analyzed was precisely to avoid this type of confusion.

The basis of the theory is that all knowledge can be defined in terms of a hierarchy (like a semantic net) within which default inheritance operates. The principal point at which it differs from other such theories (e.g., DATR, Evans and Gazdar 1989a, 1989b, 1990) is that inheritance is blocked not by means of a concept of “more general” or “more specific,” where values given to the more specific concept override those given to the more general, but by means of explicit blocking statements in the form of “NOT” propositions. For example, the propositions defining the morphology of the irregular verb *go* would include these two statements:

past-form of GO = <went>
NOT: past-form of GO = stem of *it* + *mEd*

(where *mEd* is the past-tense morpheme). The chief reason behind this is to enable multiple inheritance in cases such as the verb *dream*, which (for most speakers) has the two possible past-tense forms *dreamt* and *dreamed*. This would be handled in WG by the omission of the NOT proposition blocking the inheritance of the regular form. This is certainly something that deserves consideration. This sort of multiple inheritance is not possible in DATR, and it is clearly a problem, but the cost to Hudson’s theory of such a fix would appear to be excessive. The number of irregular forms that require NOT propositions would, I feel sure, exceed the number that need such multiple inheritance, although it is not possible to judge whether or not this is the case from this book, since at no point does Hudson give figures, nor does he give a listing of any substantial fragment of the grammar.

The whole work would have been more convincing had Hudson included a substantial part of the grammar in list form to give one an idea of the kind of thing our computer might have to deal with; yet even in the appendices he only gives a handful of examples (14 verb entries, 22 noun entries), although it must be said that these are reasonably representative and informative. The theory has, to some degree, been implemented (Hudson 1989; Fraser 1987, 1988a, 1988b, 1989), but no discussion of this enterprise is provided in this book, which covers just the theory. However, frequent references are made to the implementability of the theory, and Hudson does claim that although the grammar of English provided has not been completely implemented, he believes that there are no significant unsolved problems that would prevent this.

WG does not make use of any notion of phrase or clause—all syntactic information is about words, with the exception of the notion of “word-string,” which is needed for coordination. Hudson uses word-string to mean simply whatever needs to be coordinated, claiming that “because we have not used constituent structure elsewhere, we are free to use it in the treatment of coordination, without constraints imposed by other structures.” This may seem a plus to him, but to me it seems a distinct minus. What is inherently special about coordination that it should require such a radical departure from the apparatus needed to handle other constructions? Surely it is better to be able to handle coordination within the framework used for other constructions (albeit with some adjustments)?

The theory naturally has holes and is hindered to some degree by a formalism that appears somewhat clumsy, particularly if a computer implementation is intended, which it clearly is. It does, however, deserve consideration, in particular the arguments relating to the inextricability of linguistic and nonlinguistic knowledge, which hint at answers to many longstanding problems in the study of the higher levels of language. It would be impossible in a single volume such as this to cover in detail all the ground that is touched on, and Hudson, naturally enough for a linguist, sticks overwhelmingly with linguistic matters, in spite of his aggressive arguments against such a separation. It is to be hoped that someone (not necessarily Hudson himself) will take the theory further out of the realm of language to see if his claims can be substantiated.

References

- Evans, Roger, and Gazdar, Gerald (1989a). “Inference in DATR.” *Proceedings, 4th Conference of the European Chapter of the Association of Computational Linguistics*. Manchester, England. 66–71.
- Evans, Roger, and Gazdar, Gerald (1989b). “The Semantics of DATR.” *Proceedings, 7th Conference of the Society for the Study of Artificial Intelligence and Simulation of Behaviour*. Sussex, England. 79–88.
- Evans, Roger, and Gazdar, Gerald (1990). “The DATR papers.” *Cognitive Studies Research Paper No. CSR139*, University of Sussex.
- Fraser, N. (1987). “A word grammar parser. Progress report 1.” University College London.
- Fraser, N. (1988a). “A word grammar parser. Progress report 2.” University College London.
- Fraser, N. (1988b). “A word grammar parser. Progress report 3.” University College London.
- Fraser, N. (1989). “A word grammar parser. Progress report 4.” University College London.
- Hudson, Richard (1989). “Towards a computer-testable Word Grammar of English.” *UCL Working Papers in Linguistics*, 1, 321–339.

Lynne Cahill is a Research Fellow at the University of Sussex, where she is working on the parsing of ill-formed input and constructing computational lexicons. She received her D.Phil. from the University of Sussex for work on morphology. Cahill’s address is: School of Cognitive and Computing Sciences, University of Sussex, Falmer, Brighton BN1 9QH, U.K.; e-mail: lynneca@cogs.susx.ac.uk