

# Book Review

## New Editor's Note

Books reviewed in the *AJCL* will be those of interest to computational linguists; books in closely related disciplines may also be considered. The purpose of a book review is to inform readers about the content of the book and to present opinions on the choice of material, manner of presentation, and suitability for various readers and purposes. There is no limit to the length of reviews. The appropriate length is determined by its content.

If you wish to review a specific book, please contact me before doing so to check that it is not already under review by someone else. If you want to be on a list of potential reviewers, please send me your name and mailing address together with a list of keywords summarizing your areas of interest. You can also suggest books to be reviewed without volunteering to be the reviewer.

Lyn Bates, Book Review Editor

## Language as a Cognitive Process. Vol. 1. Syntax

T. Winograd

Addison-Wesley, Reading, MA, 1983.  
608 pp., \$23.95, ISBN 020108-571-2

### General

This book is probably the first ever comprehensive, authoritative, and principled description of the intellectual history of natural language processing with the help of computers. It is also a very thorough introduction into the craft of dealing with natural language in the framework of artificial intelligence or cognitive science, the disciplines that are interested in natural language theoretically only to the extent the latter sheds light on their main object of study: knowledge.

The book is multi-faceted. It is, first of all, a textbook; but it is also a reference book, a compendium of practical knowledge for grammar and parser writers. This practical knowledge is presented in a "digested" way – no small feat for the author – that is, is organized into conceptual groups and explained in a largely unified terminology. This approach is very welcome because "original research papers are often confusing, since a system is presented as a whole, with its unique features (rather than common ideas) emphasized, and with the important ideas mixed with the implementation details" (p. 358). Finally, it is a statement of the linguistic outlook of the author and has much to do

with artificial intelligence, computer science, cognitive science (sometimes referred to as theoretical artificial intelligence) and the philosophy of science in general.

These three objectives are present, to some extent, in all the chapters of the book, but one may find that the textbook material is concentrated in Chapters 2 through 6 – "Word Patterns and Word Classes", "Context-free Grammars and Parsing", "Transformational Grammar", "Augmented Transition Network Grammars", "Feature and Function Grammars" – as well as in Appendix X: "A Language for Describing Objects and Procedures". The material useful mostly as reference is contained in Chapter 7, "Computer Systems for Natural Language Parsing", and Appendixes B, "An Outline of English Syntax", C, "Current Directions in Transformation Grammar", and D, "An ATN Grammar for English". The methodological position and the linguistic credo of the author are explained in Chapter 1 (especially Section 1.3, "The computational paradigm"), which is the most theoretically significant part of the volume.

Another way to classify the book's material is to divide it into the theoretical versus implementational, descriptive versus operational, "what" versus "how to" parts. We shall follow this distinction in our discussion.

### DL

Winograd introduces the body of knowledge accumulated in the field over the past 20 years not as a chronicle, but rather as intellectual history: ideas have precedence over people. Thus, for example, the textbook starts not with the description of the early attempts at dealing with natural language, such as Raphael's SIR or Weizenbaum's ELIZA, but with the definition of the notion of patterns and pattern matching. These are the simplest and the least "intelligent" ways of dealing with natural language formally. The description, however, serves the double purpose of being the foundation for a discussion of more complex pattern matching (e.g., transition networks) at the same time providing a testbed for the introduction of DL, a notation used throughout the book for defining entities and describing algorithms and non-deterministic schemata. The decision to devise and use DL has obviously been a major one in the preparation of this book. There was a need for it since the prospective readers (as the participants in Winograd's courses) include linguists who do not speak any of the "standard" computer languages. Indeed, the book is saturated with DL, and a special 48-page-long appendix is devoted to DL language specification. The deci-

sion to use DL will probably prove the most controversial issue in the whole book, when viewed from the standpoint of classroom use.

The language will be initially disliked by the people with some background in computing, since it requires time to gain reading fluency.

### Textbook

The "how to" part of Chapter 2 contains definitions of simple (literal, open, and lexical) and variable patterns, as well as of basic procedures for matching and generating sentences from a pattern. The notions of regular expressions and transition networks (treated as extensions of patterns) are introduced, together with the non-deterministic procedures for recognition. The circle of problems connected with search is also addressed. Backtracking and parallel processing are discussed as techniques for traversing transition networks.

The "what" part of this chapter consists of the discussion of word classes, mostly in English. This is a bridge between lexical patterns and transition networks. The organization of dictionaries for computer systems is discussed. The word classifications and word class definitions are presented rather brusquely (cf. "anything that does not have another class gets called an adverb", p. 53), but they fill their purpose in providing terminology for discussions in further chapters.

The "what" part of Chapter 3 includes a discussion of the "final products" of syntactic analysis: the types of syntactic structures. The discussion covers the head and modifier approach, the immediate constituent and the slot and filler approach. The functional character of the latter is emphasized ("role names are different from phrase types", p. 79), the importance of which will be felt later. Section 3.3 introduces the notion of parsing (recognition + assignment of structure) and the first schematic representation of the components of a parsing system. Also, this section inaugurates a succession of very important subsections, scattered throughout the book, which deal with the issues of strategy, design, trade-offs and choices in building systems for the automatic processing of natural language. In the first such section Winograd discusses general issues in parser design: uniformity of processing, separate processing of levels and precision, procedural alternatives of sequential versus parallel, and top-down versus bottom-up analysis, and the choice of network nodes to be expanded. This book gives the reader a clear understanding that the alternatives are independent of the grammars chosen for a particular analysis, universally applicable and not specific to any system in which they may have been used.

The "how to" part of Chapter 3 includes the discussion of context-free grammars and derivation of sentences; context-free parsing; non-deterministic

schemata for the top-down and bottom-up recognition, as well as their realizations: both the backtracking and the parallel algorithms for top-down recognition and the parallel bottom-up one (the remaining algorithm is given as an exercise). Next the augmentation of a recognizer to a parser is discussed, and the chapter is crowned by the introduction of the active chart parser, a technique combining features of a top-down and a bottom-up parser. The material is presented in a concise and very efficient manner, and it is quite easy to understand the idea and the technical details of active chart parsing.

Chapter 4 discusses transformational grammar. There is no "how to" part. The context-free grammar rules cannot account for discontinuous components, subject-predicate agreement, etc. So, the rules of the grammar are generalized, and an hierarchy of grammar types is presented for the first time. The notions of the finite state (regular) and the context-sensitive grammar are defined. Also included are a procedure for producing a grammar from an equivalent network and some thoughts about the choice of the power of grammars for natural language processing. "In general, the motivations for using more powerful grammars go beyond *weak generative capacity* (the ability to specify a given language), but are based on a desire to have a grammar that is simple and that produces structures that correspond to our intuitions about other considerations, such as meaning" (p. 147).

Winograd goes on to give a brief explication of the Standard Theory (ST), striving to cover not only the technicalities of the application of transformational rules, but also the philosophy of ST's approach to the study of language (with its emphasis, in different contexts, on competence, deep structure and interpretive semantics). The transformations themselves are introduced in Winograd's customary lucid and formal way (here he borrowed Akmajian and Heny's (1975) notation); some refinements to ST (bracketing, variables, rule ordering, extensions to the base, etc.) follow the description of the basic transformational derivation. Additional developments in ST are covered in Appendix C.

Winograd criticizes transformational grammar for overemphasis on the role of syntax, complete disinterest in the problems of processing (and processes) and the resulting poor amenability of transformational grammar to computer studies of natural language.

Chapter 5 introduces the transition network equivalents of the context-free grammars and transformational grammar: recursive and augmented transition networks, respectively. The standard introductions to ATNs (the original paper by Woods (1970) and the excellent introduction by Bates (1978)), while being readable and useful, do not profit from the well-developed context built up by earlier chapters in this book. Winograd, predictably, does not use the tradi-

tional Lisp-like notation for ATNs, but opts for pictorial diagrams and a special notation with starting and ending state names in subscript around the arc name and conditions, actions and initializations specified in English. This approach should widen the circle of the readers of the book.

The discussion of the component parts of an augmented transition network is very explicit. The notions of the arcs and their classification, conditions and actions, initializations, feature dimensions and role names are lucidly defined. A separate section is devoted to the use of registers in ATN grammars, and is accompanied by specific examples of problems solved through the use of features under discussion. Winograd discusses in some detail a relatively wide range of syntactic phenomena of English and their treatment in ATN terms. A fuller outline of English syntax (though not in the ATN format) is given in Appendix B.

The section devoted to ATN implementation is more ideological than technical and includes the description of the notion of compilation that would be baffling to novices: "The ]concept of compiling ... can be loosely formulated as 'Once you know what is actually used, you can write a procedure that does it more efficiently than one designed to handle everything'" (p. 265). This is, however, one of very few flaws of metaphorical oversimplification in the text.

Chapter 6 deviates from the linear progression that established an ascending trend of complexity, and starts to discuss alternative grammar formalisms. The first such alternative is the systemic grammar of Halliday. Winograd's SHRDLU was built under the influence of systemic grammar, and thus there is a special relationship between the author and the approach. Systemic grammar has not found significant following among linguists; as Winograd himself mentions, its main reason lies in the sociological aspects of language. Maybe this is the reason why it lacks formality, accuracy and a unifying organizational principle: the authors of the grammar had no such intentions, their main audience, at least originally, being secondary school teachers. The emphasis on classification is too pronounced – designing taxonomies is the most superficial way of studying a phenomenon, even if functionality is declared as the general goal. The structure of the grammar is a loose conglomerate of very interesting issues which are not united by a common theoretical basis, and are thus relegated to the status of brilliant observations on the nature of language. It seems that the originality value of systemic grammar lies in the fact that its authors tried their best to produce something different from the "American transformationalist emanations". The interest to language function in systemic grammar ascends not simply to Firth and Whorf, but, more important, to the ideas of Elmslev and Prague Linguistic Circle. Incidentally, Sgall, Hajičová and Benešová, quoted by Winograd,

have but a geographical proximity to the Prague Linguistic Circle; Mathesius spoke about theme and rheme in the 1940's, not in 1961, as the date in the reference might suggest, thus making this idea roughly contemporary to Halliday's work; these notions, and the philosophy of systemic grammar, are part of the legacy of the structuralist paradigm and can be traced back to Saussure.

The introduction to systemic grammar has probably never been formulated as precisely and formally as in this book. It is my conviction that Winograd developed the systemic approach quite beyond its original level. Most of the ideas behind it are sound and appealing, and lack the apparent anti-semanticism of the transformational paradigm; the systemic approach presented a very good framework for developing computer programs for language understanding. One of the most attractive features of this approach for the computational linguists was its relative vagueness and pre-formal state, since this permitted quite diverse interpretations and further specification of the theory in the computational domain. Halliday's description of English clause types and "transitivity" must have especially attracted the designers of computer systems, since being little more than a list of distinct and real phenomena, it worked as a memory aid for recollecting the various language structures that had to be included into (or consciously excluded from) the sublanguage to be accounted for by the system under construction.

This impression is corroborated by the lack of formal definitions of rules for the systemic grammar and specifically by the observation that in computer programs that used some ideas of systemic grammars, such as Winograd's SHRDLU, the realization rules employed were "implicit", i.e., built into programs which were "complex and difficult to modify" (p. 310).

Next on the agenda are case grammars. The most important observation about this kind of grammar is that this approach permits one "to see the sentence structure as a window onto an underlying scenario... The grammar must provide a systematic way to carry out this mapping so the hearer will know ... what kind of scenario is intended. It could be argued that this problem is not properly a part of syntax and should instead be viewed as semantic interpretation" (p. 313).

Winograd gives a concise account of several case systems for English, including two proposals by Fillmore and the contributions by Simmons, Schank, Chase and Grimes. This account is much more principled and comprehensive than, for instance, the chapter on case grammars (by W. Samlowski) in Charniak & Wilks's "Computational Semantics" (1976) although Winograd defers all deliberations on semantics until

the second volume of this book, whereas no such restriction was present in the other textbook.

The "what" part of this section contains a discussion of "criteria for deciding on cases" and "integrating cases into formal syntax" along with a subsection on the relationship between case grammar and systemic grammar. Winograd argues that systemic grammar has all case grammar has to offer and more. It seems, however, that in view of the scientific paradigm of cognitive science the semantically oriented cases of, say, Schank are preferable to the taxonomy-minded clause-dependent cases (or transitivity patterns) of systemic grammar.

The last part of this chapter is devoted to functional and generalized phrase structure grammars. Both approaches are fairly new and have not yet resulted in the development of complete large-scale computer applications. These grammars emphasize nondirectionality (that is, they deal with both parsing and generation), correspondence with meanings and multiple dimensions of analysis. Thus, functional grammars consider the full analysis of a sentence to be made up of 1) constituent structure; 2) functional description; 3) feature description; 4) lexical content; 5) semantic structure; and 6) phonological structure. The goal is noble, but it is doubtful whether all these elements can be formally united, by a functional grammar, in one system – in order to implement a "blackboard"-type parsing arrangement. This discussion, together with the brief sketches on definite clause, slot, junction, cognitive and relational grammars, constitute a smooth transition from the textbook to the reference book part of Winograd's text. It is impossible to acquire more than a superficial knowledge of the grammar theories and mechanisms from the exposition; but this material was not meant to be a substitute, for one should deal with this material as a source of reference.

### Reference Book

Building computer systems that boast a measure of understanding natural language has become quite common and widespread. One feature almost universally present in such systems is a syntactic parser of natural language. Chapter 7 is an extended "quick reference card" for people who build syntactic parsers. The chapter contains discussions (in various levels of detail) of 52 systems spanning 19 years of effort in the field; the discussion proceeds along conceptual (and not historical or other) lines. After naming several application areas for such systems (machine translation, question answering, data base retrieval, theory justification, etc.), Winograd goes on to the section of the greatest practical importance: "Issues in the design of a parsing system". The following crucial issues stand out: a) the choice of the grammar formalism; b) the form of assigned structures; c) the search strategies used, and d) the degree of completeness of the

system (what size sublanguage it is supposed to take care of). The systems are classified and discussed according to the type of their grammar formalism (augmented phrase structures, transformations, charts, ATNs, pattern matching, or situation-action rules). The emphasis, predictably, is not on technical detail, but rather on the relative strengths and weaknesses of the approaches. One will not be able to implement a parser solely on the basis of the information in the book (this was not intended), but the chapter is an excellent source for choosing the approach best suited to one's individual needs and tastes.

Some "raw material" for use in a parsing system can be found in Appendix B: "An Outline of English Syntax". This is a digest of Quirk's English grammar (Quirk et al. 1972), set in a largely systemic terminology and framework. The appendix does not purport to give answers to all the grammatical problems of English. Many topics are not covered, many more are just sketched with pertinent examples. Some suggested solutions (one example: embedding constraints in dealing with long-distance dependencies) are transformationalist rather than systemic, and one could argue that it is next to impossible to reconcile the two philosophies, even without trying to incorporate them in one computer system. The question is whether researchers will be better off with this well-structured but rather tendentious digest than with a grammar like Quirk's, or Jespersen's, or Zaliznjak's "Nominal Form-Formation in Russian".

Appendix C contains a very concise survey of the post-1965 development of transformational syntax. It is a logical extension of Chapter 4. This material is not indispensable for the book, especially since the transformationalist approach has been shown not to be particularly applicable in building computer systems.

Appendix D is the shortest and the most immediately usable of all. It contains a DL definition of an ATN grammar and one such grammar for English. The network is the summary of results obtained in Chapter 5 and is a reasonable starting point for developing a practical parser. It contains 18 states and 51 arcs in three subnetworks.

### Methodology and Linguistic Theory

The methodological part of this book is the most important one. This seems to be the first forceful attempt at finding a substitute for the Chomskian transformationalist milieu in the field of linguistics. (Please note the absence of the describer "computational": Winograd significantly considers computational linguistics to be a linguistic paradigm, like the structural and the generative ones, not an application area of general theoretical linguistics.) The need for the philosophical and methodological justification of the largely application-minded efforts in AI has been realized for a long time but nobody had been eager to spend time

devising a meta-theoretical framework for the field. The necessity of the deviation from transformationalism is justified in this book in terms of a switch in the scientific paradigm within which research is being conducted – the notion was borrowed from the philosopher of science Thomas Kuhn. The reigning paradigm is generative. The computational paradigm is a rebel. Although the generative and the computational paradigms share an interest in the knowledge possessed by an individual who uses language and in formal symbol manipulation, they differ in the degree of attention to process organization (low in the generative paradigm) and the inclusion of non-linguistic knowledge into the sphere of interest of linguistics (liberal in the computational paradigm – cf. Raskin (forthcoming) and Schank et al. (1982) for two recent expositions of the positions of the adherents of the generative and the computational paradigm, respectively.)

The computational paradigm, of which Winograd is probably the best explicator, perceives language as “a communicative process based on knowledge.... Theoretical concepts of program and data can form the basis for building precise computational models of mental processing” (p. 13). The basic model of communicative processing is perceived in terms of fulfilling communicative goals (different sets for the speaker and the hearer) through ample use of the stored knowledge of 1) language, 2) world, and 3) situation. Winograd goes on to specify the model of processing done by a language user and to discuss the “nearly decomposable” stratified model of the knowledge of language used by a language user. This model contains three rubrics: stored knowledge, processes and assigned structures, each of which contains six parallel levels of rules, processes and structures, respectively (from phonology to, notably, pragmatics), while the stored knowledge also includes two kinds of dictionaries (a syntactic and a semantic one). The model is not discussed in the greatest possible detail simply because it is not yet a full-fledged theory, and also since the genre of the text precludes the undue emphasis on metatheory (however welcome such a discussion or theory may be to the field).

The chapter also gives an overconcise and excessively metaphorical account of the history of the linguistic science and an overview of the computer applications for natural language. The tone of the discussion here and throughout the book is refreshingly evenhanded and calm.

## Conclusion

This reviewer taught a one-term course based on Winograd's book in Spring 1983 to seniors in Computer Science. The course was very successful. The students expressed great enthusiasm about the topic and the way it was treated, although the course was by no means easy: the participants had 22 homework assignments, largely of a computational nature, including an active chart parser and an ATN parser for a small subset of English as two of the regular exercises, and a term project. In a very large measure, the course owed its success to the book under review, which was used as the textbook 75 percent of the time. A questionnaire distributed to the participants showed that the text was an unconditional success. Predictably, a majority of the students polled would have preferred Lisp or some other programming language to DL. There were no complaints about excessive difficulty, although the book is intended for graduate courses. In the course, we covered Chapters 1 through 5 and the case grammar part of Chapter 6. Chapter 7 was suggested for independent reading.

College teachers of computational linguistics should be very grateful to Terry Winograd for the amount of time and effort he devoted to this fundamental text. It is a beacon for the field. I have no doubt that this book will become a standard reference book for the developers of syntactic parsers. There are all reasons to believe that the forthcoming second volume, devoted to meaning, will as authoritative and comprehensive, and even more thought-provoking and stimulating.

Sergei Nirenburg, Colgate University

## References

- Akmajian, A. and Heny, F. 1975 *An Introduction to the Principles of Transformational Syntax*. MIT Press, Cambridge, MA.
- Bates, Madeleine. The theory and practice of augmented transition network grammars. In Bolc, L., Ed., *Natural Language Communication with Computers*. Springer, New York, NY: pp. 191-259.
- Charniak, E. and Wilks, Y., Ed. 1976 *Computation Semantics*. North Holland, Amsterdam.
- Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. 1972 *A Grammar of Contemporary English*. Seminar Press, New York, NY.
- Raskin, V. to appear On the boundary between linguistic and encyclopedic knowledge. *Quaderni de Semantica*.
- Schank, R., Birnbaum, L., and Mey, J. 1982 Integrating semantics and pragmatics. In *Preprints of the plenary session papers of the XIII International Congress of Linguistics*. Tokyo, pp. 129-140.
- Woods, W.A. 1970 Transition network grammars for natural language analysis. *CACM* 13 10, pp. 591-606.