Book Reviews

Natural language parsing: Psychological, computational, and theoretical perspectives (Studies in natural language processing)

Dowty, David R; Karttunen, Lauri; Zwicky, Arnold M (editors)

Cambridge University Press, 1985, xiii+413 pp.
ISBN 0-521-26203-8; $49.50 [20% discount to ACL members]

This book is the first volume in a series, *Studies in Natural Language Processing*, launched in 1984 under the sponsorship of the Association for Computational Linguistics. Aimed at a very wide audience, with background in formal linguistics, psycholinguistics, cognitive psychology or artificial intelligence, the series addresses a number of issues in the growing interdisciplinary field of computational linguistics. As a representative of these concerns, the inaugural volume succeeds quite well in setting the tone, by demonstrating the range of treatments that the notion of parsing has received from the perspectives of formal linguistics, computational analysis of language, and psycholinguistics.

The book is not yet another workshop or conference spin-off, even though earlier versions of several papers were originally presented at conferences on parsing in 1981 and on "Syntactic theory and how people parse sentences" in 1982. The individual contributions are of consistently higher quality than those in a number of other edited collections on a single topic within computational linguistics. The book manages to convey a feeling for the complexity of the phenomenon at its focus, as it has evolved in more recent studies of language and mental processes; it also indicates the diversity of research directions pursued within the general area of syntactic processing of natural language. Unfortunately, individual contributions stand pretty much on their own, as there are no immediately obvious connections between most of the papers in the volume. With one exception (Kay's "Parsing in functional unification grammar" and Karttunen and Kay's "Parsing in a free word order language", where a formalism introduced from a computational perspective in the first paper is used as a descriptive device for an analysis of Finnish word order in the second), the reader has to work hard to find common themes running through the rest of the papers.

The eleven chapters in the book can, on a first approximation, be grouped into several (overlapping) categories that reflect the structure suggested by the title. In particular, there are contributions concerned with a number of psychological and computational models of parsing, presentations of formal linguistic frameworks and evaluations of properties of syntactic theories, and linguistic studies, including a comparative analysis of the syntax and semantics of constituent questions in English, Swedish, and other Scandinavian languages. The volume contains a range of detailed reports on, and conclusions drawn from, psycholinguistic experimental work on human language comprehension.

On the whole, a number of papers seem to be concerned with seeking correlations between features of grammars and some aspects of human parsing performance. For instance, Crain and Fodor ("How can grammars help parsers?") seek to validate the claim that the human sentence processing mechanism is capable of applying all relevant grammatical information on an 'as needed' basis, as opposed to being viewed as a sequentially decomposable processor. Frazier ("Syntactic complexity"), while analysing sources of processing complexity in order to derive a general metric for it, raises the question of whether (and how) language, and grammars, might have evolved to facilitate the parsing task. Tanenhaus, Carlson, and Seidenberg ("Do listeners compute linguistic representations?") set out to study the manner in which syntactic theory and the human sentence parsing process are connected. Independently of the strength of their argument in favor of the modularity hypothesis, they focus on the aim of understanding "the relationship between the grammar and the general cognitive system". The bulk of Engdahl's paper "Interpreting questions" analyses a wide range of data from Swedish, Norwegian, German, and English that presents strong evidence in support of a correlation between the processes of extraction and wide scope interpretation. Ultimately, however, she proposes an account for this correlation by making a statement about the human sentence processing mechanism: the explanation rests on the same device (Cooper storage) underlying both processes.

It is, however, only at such level of generality that connections between separate papers can be perceived. If the book attempts to promote, explicitly, cooperation amongst researchers representative of the different areas within the general field of syntactic processing of language, then it should have given a clearer picture of the relationships between these areas. If the book sets out to convey the impression that a coordinated, multidisciplinary research programme on parsing is under way, such an impression is largely lost in the process of reading the individual contributions. This should not be regarded as a strong criticism of the book, as it is probably attributable to the fact that, particularly at the

time when this collection of papers was being compiled, such a programme simply did not exist. Nevertheless, and especially from the point of view of the aim of the series, a volume with explicit cross-references between the chapters and with a tighter introduction would have been particularly welcome (Engdahl is practically the only author here who references, on a fairly systematic basis, other chapters in the volume). As it stands, the opening chapter by Karttunen and Zwicky reflects the diversity and disconnectedness of the individual contributions: a succinct description of the notion of parsing, viewed from a number of different perspectives, provides context for outlining common concerns and points of contact; this is then followed by individual summaries of the eleven chapters, which, despite being very precise and informative, fail to improve on the predominantly flat structure of this collection as they view each chapter as a stand-alone entity.

In the remainder of this review I will, at the risk of doing injustice to those contributions which are written mostly from a psycholinguistic perspective, attempt to identify some of the central themes which may be of interest to the more computationally minded reader.

However, bear in mind that this volume is not a repository of parsing algorithms, ready to be translated into code. The inquisitive student, looking for insights into implementing parsers, better look elsewhere. Likewise, with the exception of Joshi's presentation of tree adjoining grammars (chapter 6), there is very little information concerning the complexity and power of formal systems. Still, that chapter offers a fairly detailed comparison of a number of recent syntactic theories as viewed from within formal language theory, as well as some interesting remarks.

This chapter is one of the two papers presenting formal linguistic frameworks; the other is Kay's "Parsing in functional unification grammar". Joshi's main concern, in addition to introducing tree adjoining grammars (TAGs) as a device for rendering an account of unbounded dependencies in a linguistically interesting way, is to study their linguistic adequacy from the point of view of the structural descriptions TAGs are capable of. In a larger context, this concern is reflected in the title of the chapter, "Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions?". Joshi demonstrates that TAGs characterise a class of grammars whose power is slightly beyond that of context-free grammars (he calls them "mildly context-sensitive") and argues that they are both linguistically adequate and parsable.

Kay's paper on functional unification grammar (chapter 7) raises questions about the role of linguistic formalisms in the study and analysis of language. He presents a formalism which is not "explanatory" in its own right, but "has been designed to accommodate functionally revealing, and therefore explanatorily satisfying, grammars". This formalism itself bears theoretical status; and Kay regards unification grammar as a "competence grammar" and does not expect a parsing procedure to make direct use of its rules. Indeed, neither tree adjoining grammars, nor functional unification grammar (FUG) have been used extensively for practical implementations of parsing systems; both formalisms have been applied to the task of generating natural language (Appelt 1983, McKeown 1985, McDonald and Pustejovsky 1985). The second half of the paper outlines a framework in which unification grammar can be used for parsing, by compiling a specific, and equivalent, representation suitable for use by a particular parsing formalism. In the sense of making a distinction between the grammar that a linguist might write and the grammar a parser can use, Kay's position predates the attitude taken by a number of contemporary parsing systems.

The following chapter, by Karttunen and Kay, shows how the formalism of FUG just presented can be used to give an account of Finnish word order (where complexity arises from the interplay of a relatively small number of constituent ordering rules involving both syntactic and discourse factors) and argue that this formalism provides "a firm basis for performance models and computational procedures". Equally important, particularly from the computational perspective, is the demonstration of how specific properties of a language affect its parsing, even if a generalised processing algorithm for a given formalism exists. In order to account for the relatively free word order of Finnish, a compilation procedure (along the lines discussed by Kay earlier) has to be further augmented and modified before standard parsing techniques can apply. From an equally practical point of view, the paper demonstrates a point emphasised by Kay in the previous chapter: "since the parsing and generation grammars do indeed describe exactly the same languages, so much of the work involved in testing prototype grammars can be done with a generator that works directly and efficiently off the competence grammar".

The other major thread, running through several papers and of particular relevance to computational linguists, concerns parsing strategies, and the papers by Pereira ("A new characterisation of attachment preferences") and Crain and Steedman ("On not being led up the garden path: the use of context by the psychological syntactic processor") are representative of two different attitudes to this issue taken from the perspectives of formal parsing and experimental psycholinguistics. Within the formal theory underlying bottom-up shift-reduce parsing, Pereira draws a precise model of the principles of Right Association and Minimal Attachment, introduced by psycholinguists to explain the preferential readings obtained from a certain class of structurally ambiguous sentences when presented out of context. He argues that much of the debate surrounding the formulation of such strategies can be resolved by their incorporation into a suitable, computationally trac-

table framework, since, as he then demonstrates, the two principles turn out to correspond to two precise rules in the model. The emphasis of Pereira's paper is not on the psychological plausibility of the model, and he is quite explicit in not making any claims about the human parsing mechanism. However, beyond the immediate content of the paper, his concern is with the rigorous specification of principles within a general framework capable of supporting the formulation of "precise falsifiable models".

The paper by Crain and Steedman is particularly interesting to read immediately after Pereira's proposal for precise formulation of parsing strategies within the framework of purely syntactic parsing, as it argues against any strategies for resolving local ambiguities on structural grounds alone. Crain and Steedman take the abundance of syntactic ambiguity which characterises natural languages as the starting point of their argument. Having introduced the notions of 'strong' and 'weak' interaction between a parser using structural guidance, and a semantic (in the widest sense of the word) interpreter with access to more general world knowledge, they draw together a number of related theoretical issues and a set of experiments which aim to monitor the effects of reference and context on the comprehension of garden path sentences. They conclude that "the primary responsibility for the resolution of local syntactic ambiguities in natural language processing rests not with structural mechanisms, but rather with the immediate, almost word-by-word interaction with semantics and reference to the context". The psycholinguist reading this article may, or may not, agree with such a position — after all, Frazier's chapter brings forth a number of references which argue in favour of the psychological reality of parsing strategies. The computer scientist, however, must pay due attention to Crain and Steedman's paper, as it addresses the important issue of the overall organisation of the language understanding system.

While I have not focused closely on all of them, the chapters describing experimental work on human language comprehension together convey a message of a slightly different nature to their individual topics. It is an important achievement of the book, particularly where the more computationally minded readers are concerned, that it manages to demonstrate, explicitly and conclusively, the importance, as well as the difficulty, of designing and controlling an appropriate and effective psycholinguistic experiment. Tanenhaus *et al* point out that experimental results based on some kind of conscious awareness "need to be interpreted with some caution" until the replication of such results using the same, or more specialised, experimental designs. Their chapter is a good example of precise experimentation.

The book also raises a number of issues concerning the methodology of psycholinguistic research. In particular, what comes across especially well is that a study of language should proceed from data to experimental analysis of theories of parsing, and not by equating general models or outlines of the natural language understanding system with non-refutable hypotheses. While as a methodological principle this may be well known in psycholinguistic circles, it is worth emphasising for the benefit of the larger audience at which the volume is aimed.

In conclusion, given its self-proclaimed interdisciplinary nature, this book is slightly unbalanced. It could have been made more cohesive by an introduction seeking to cluster the individual contributions together along a number of explicitly stated dimensions, identifying deeper-running threads and concerns common to a wide range of researchers. Even so, the book succeeds in presenting a lot of diverse material and demonstrating, albeit implicitly, why research in parsing can no longer be considered the exclusive province of practitioners within a single field.

This is not a textbook, nor is it a collection of papers that lends itself to casual browsing. It is a rich and thought-provoking volume which, given the attentive study it deserves, is going to reward its readers by offering a number of insights from neighbouring disciplines.

Branimir Boguraev
Computing Laboratory
University of Cambridge
Corn Exchange Street
Cambridge, England CB2 3QG

References

Appelt, D. 1983. TELEGRAM: A grammar formalism for language planning. *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, Germany, 595-599.

McDonald, D. and Pustejovsky, J. 1985. TAG's as a grammatical formalism for generation. *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, Chicago, Illinois, 94-103.

McKeown, K. 1985. *Text generation.* Cambridge University Press, Cambridge, UK.

TEXT GENERATION: USING DISCOURSE STRATEGIES AND FOCUS CONSTRAINTS TO GENERATE NATURAL LANGUAGE TEXT

(Studies in natural language processing)

McKeown, Kathleen R.

[Columbia University]
Cambridge University Press, 1985, x+246 pp.
ISBN 0-521-30116-5; $29.95 [20% discount to ACL members]

PLANNING ENGLISH SENTENCES

(Studies in natural language processing)

Appelt, Douglas E.

[SRI International]
Cambridge University Press, 1985, x+171 pp.
ISBN 0-521-30115-7; $32.50 [20% discount to ACL members]

The second and third entries in the ACL Cambridge

University Press series provide two excellent and very different approaches to text generation strategy. The overview of the work presented in these volumes will be familiar to anyone who has followed the annual meetings in artificial intelligence and computational linguistics, but up to now this material has not been available in anywhere near this much detail.

Kathleen McKeown's book, which appeared first of the two, takes a discourse-strategy approach. Beginning largely from the work of Grosz (e.g. 1977) and Sidner (1979) on discourse structures and focus issues in natural language understanding, and from the work of Mann and Moore (1979) and Weiner (1980) on textual organization, she constructs a system called TEXT for generating coherent extended answers to simple questions about information in a database.

The heart of the system consists of a set of discourse structures, formulated into schemata, which organize answers to questions of certain kinds. The theoretical basis for schemata of the kind developed here lies in work on rhetorical predicates beginning in the last century, but finding other recent applications in the coherence relations of Hobbs (1978) and Hirst (1981). By analyzing short expository texts for the kinds of rhetorical predicates occurring in them and the patterns in which they occur, the author developed a list of schemata, which amount to recursive templates for paragraphs. These schemata were then categorized by the situations in which they occurred; different schemata are used to achieve different goals, and depending on what information is available. Responses to questions are then generated by selecting an appropriate schema and filling it in, using focus to limit the knowledge pool from which information to fill in the schema may be selected (i.e. to determine relevance).

The knowledge base from which TEXT works is an entity-relation style database, enhanced with several kinds of hierarchical information and with automatically detected and flagged attributes that are particularly important or interesting to the system. For example, distinguishing descriptive attributes are those whose values partition a class; in Aristotelian terms, they are the differentia of classical genus-species-differentia definitions. The database has also been enhanced with a certain amount of meta-information, that is, information about what it knows. This information is used both to fill schema slots directly and help determine relevance.

The kind of request involved plays the major role in determining relevance. Here, it is important that McKeown's system deals with a very limited range of question types: essentially, it handles simple requests for information or definitions, and questions involving comparisons of objects of various kinds. Depending on the category of the question, relatively simple criteria involving the knowledge base features and hierarchy let the system determine a pool of relevant knowledge. The mechanism involved is moderately naïve, but effective for McKeown's purposes.

The schemata are implemented as ATNs (with some relatively minor extensions to the familiar formalism). The simplicity of this "discourse grammar" can be seen by noticing that backtracking ability was eliminated, as the author found it was never needed. Oddly enough, she chose not to use ATNs actually to produce the output; instead, the schema networks produce a chunk of representation, which is then handed to a Kay-style functional unification grammar (1979, 1981) to provide the actual sentences. McKeown makes no claims of sophistication for the tactical component; it was included simply to provide output so that there would be some practical way to tell that the strategic component was actually producing coherent discourse.

The primary weaknesses of this work arise from its specialization, which is evident in at least two different dimensions. First, the entire system presupposes very heavily that its knowledge base "started out life" as a relational database. Major portions of the strategic component rely essentially on enhancements to the database which may not even make sense for domains of other kinds. For example, if the information the system is dealing with does not split neatly into hierarchies of mutually exclusive subtypes with neat lines of distinction, distinguishing descriptive attributes may not be identifiable (or even make sense). In that case, a large portion of the mechanism falls apart. Similarly, the system presupposes that all information in the knowledge base is explicit; it is very unclear that the principles in use could be extended to a knowledge base that consisted of some version of axioms plus inference strategies.

Second, the system deals only with very limited kinds of expository text: texts that would be produced in response to questions like "what is an x?", "what is the difference between an x and a y?", or "what do you know about z?" To see how far this falls from the general case of producing expository text, consider how hard it would be to formulate in these terms a question about how x influences y. Theoretically, the model can handle such extensions by analyzing more texts, discovering more schemata, and writing their grammars; however, it is unclear that such new schemata could successfully be developed using the knowledge base design that is currently in place. Hence it is unclear to what extent the TEXT model is generalizable to generation in significantly different kinds of domains or contexts.

A further, extremely minor quibble with the book has to do with its physical presentation. For some reason, McKeown (who prepared camera-ready copy for the publisher) chose a proportionally spaced font without right justification. This leads to an extraordinarily ragged right margin, which actually gave my eye trouble in tracking the text. This choice is incomprehensible to me; every cheap desktop word-processor these days comes equipped with effective right justification, let alone the facilities that one would expect to be available at Columbia University. None of the other books in this

fine series share this property, a fact for which I am grateful.

Douglas Appelt's system KAMP takes an entirely different approach, based on speech act theory, intensional logic, and planning. Unlike McKeown, Appelt attacks the by-now traditional distinction between the strategic (what to say) and tactical (how to say it) tasks in generation, arguing that language is best viewed not as a process of packaging and transmitting predetermined concepts, but as a series of actions whose purpose is to cause some change in the (mental) state of some other agent. Planning an appropriate action, he argues, involves considering many kinds of constraints simultaneously, including those on what different kinds of utterance can do, those placed by current knowledge of the world, those placed by the other agent's knowledge and beliefs, those placed by the first agent's knowledge of the other agent, and those placed by grammar. Since the planning process actively involves all this information at once, a salient distinction in system structure between strategy and tactics cannot be maintained.

KAMP treats planning as a special case of reasoning. That is, the system embodies axioms representing the effects of various illocutionary acts under various circumstances and information about the world and the system's interlocutor. Planning an utterance then takes the global form of adding a statement about a goal and trying to infer a speech act that will realize it. The formalism used is Moore's modal logic (1980) with a Kripke-style possible worlds semantics. Almost all the "important stuff" (claims about results of performing actions, claims about knowledge, etc) actually resides in the metalanguage, where most of the important inference also takes place. The discussion of the logic involved is far more technical than anything in McKeown's work, and requires either a certain amount of expertise or a lot of persistence; but a dogged enough reader should be able to make sense of it regardless of background.

However, planning is not a simple matter of performing logical inferences on assertions. In fact, for the most part, the planner does not use the axioms for actions in terms of relations among possible worlds at all, although that information is available. Instead, each action is also given a STRIPS-like precondition/postcondition description, which the planner uses as a heuristic to construct what it thinks may be a good plan. These descriptions are fitted together into a Sacerdoti-style (1977) hierarchical planner, within which plans are represented by procedural networks. The inference mechanism is used to test what holds in the possible worlds represented by various network nodes, which arise as a result of performing actions in previous worlds. Critics monitor to ensure that interacting goals behave well. Like the discussion of the formalism, the discussion of planning builds strongly on technical results; a reader unfamiliar with the planning literature in A.I. may find it very heavy weather.

Within the resulting formalism, Appelt develops formal representations for illocutionary acts, including very precise axioms for their preconditions. Surface speech acts serve as an important way in which illocutionary acts can be expanded (though not the only way). Grammatical information is once again embodied in a functional unification grammar. To avoid the requirement that all semantic information be present from the outset, which would reintroduce the what/how distinction that Appelt wants to avoid, the unification process has been modified to let the system start with a minimal functional description, applying a unifier that knows when it does not have all the information it needs and calls the planner to get it. In this phase the planner handles both traditionally strategic issues like focus and traditionally tactical ones like pronominalization and compressing distinct information (or in this case, achieving multiple goals) into single utterances.

Appelt goes into far less implementation detail than McKeown, probably because the details of how implementation is achieved matter less to him than the formalism on which they are based. Still, the book is short enough that a chapter on the nitty-gritty of how he actually achieves his results would not have come amiss. Questions that are not answered include such basic issues as how he achieves inference, for instance. Since his axioms exercise most of the options of full modal logic and its metalanguage (they are not in any sort of normal form, and many probably cannot be stated as Horn clauses), this question is far from trivial.

More bothersome, however, is the question whether some of his assumptions undermine the point of his effort. For instance, he deals always with knowledge as opposed to belief, and makes some sweeping assumptions under that rubric. In particular, his axiomatization assumes that agents always know anything they have been told. Even assuming that the listener believes the speaker, this sweeps a host of issues under the rug; as every teacher knows, the fact that the students believed you, and even that they can repeat the words you uttered, does not entail that they know what you told them. The distinctions between believing, knowing, and understanding are crucial to speech act planning; although Appelt claims that their omission is not essential, it is not clear that this is so. Similar limitations affect the plausibility of many of his other axioms.

In summary, both these works are important contributions to a difficult problem in computational linguistics. McKeown's presentation is more accessible to general readers than Appelt's, but a determined reader can get through either one. The difficulties in reading Appelt's book result from one of its strongest virtues: a strong grounding in formal disciplines. Both books suffer to some degree from questions concerning their approach's generalizability; but at the current state of the art, their accomplishments are impressive indeed.

As a side note, both also read just a little too obviously like dissertations; but the worst defects associated with dissertations are absent. McKeown's is probably the better guide for those interested in actually building a system; Appelt's provides a stronger view of speech act theory; and both are important reading for anyone either working in or just trying to keep up with natural language generation.

Terry Nutter
Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

References

Grosz, B.J. "The representation and use of focus in dialogue understanding," SRI Technical Note 151, 1977.

Hirst, G. "Discourse-oriented anaphora resolution in natural language understanding: A review", *American journal of computational linguistics*, 7(2) 1981.

Hobbs, J. "Coherence and coreference," SRI Technical Note 168, 1978.

Kay, M. "Functional grammar," *Proceedings of the 5th Meeting of the Berkeley Linguistic Society*, 1979.

Kay, M. Unification Grammar, Xerox PARC Technical Report, 1981.

Mann, W.C. and J. Moore. "The computer as author — results and prospects." USC/ISI Technical Report, 1979.

Moore, R.C. "Reasoning about knowledge and action", SRI AI Center Technical Note 191, 1980.

Sacerdoti, E. *A structure for plans and behavior*, Amsterdam: North-Holland, 1977.

Sidner, C.L. *Towards a computational theory of definite anaphora comprehension in English discourse*. PhD Dissertation, MIT, 1979.

Weiner, J.L. "BLAH, A system which explains its reasoning", *Artificial Intelligence*, 15, 1980.

Phonology in the twentieth century: Theories of rules and theories of representations

Anderson, Stephen R.

Chicago:University of Chicago Press, 1985, x+373 pp.
Paperback, ISBN 0-226-01916-0, $18.50
Hardbound, ISBN 0-226-01915-2.

Computational linguistics is still a cross-disciplinary field; many people in computational linguistics have a limited formal background in either computer science or linguistics. For those out there who have a stronger computer science background and have been exposed to just one or two semesters of (typically parochial) phonology, Stephen Anderson's book is an opportunity to get an excellent, in-depth, review of many major issues and viewpoints in phonology in the last 90 years. Actually, even for people concentrating in phonology, Anderson's book can be informative and entertaining. Thus, though the book has no specific relevance to computation, a note on its existence and its contents is worthwhile.

Phonology in the twentieth century is organized into thirteen chapters, ten of which are focused on work of individuals. For example, Saussure has two chapters; Trubetzkoy, Jakobson, Sapir, and Bloomfield each have a chapter. The chapters typically start with a review of the principal's life and career, followed by about twenty pages covering his work and its relation to the competing approaches to linguistic description that Anderson has laid out in the first chapter: static "representations" of languages as sets of objects versus "rule"-based grammars. Anderson's biographical sections are fascinating, yet they seem very fair. He tells why Henry Sweet failed to secure a professorship at Oxford, he explains how Boas's valid point that "no particular language could furnish in itself an adequate framework for understanding all others" got warped into the American structuralist notion that "languages could differ from each other without limit", and he generally provides a framework for understanding how funding and academic departmentalization helped shape linguistics in the United States. Anderson represents most phonological theories as they can be found in the primary sources and he works hard to reconstruct their intellectual context without undue distortion to fit things into his rules and representations rubric.

Anderson's book gains structure and focus from tying so many discussions back to the tension between rule and representation, but the book misses a third 'R'— reality. Those traditions that emphasized the linguist's responsibility to fully portray the facts of how real people really speak are given short coverage; phonetics is out of focus, and historical and variationist work is untouched. Labov doesn't even make the index. Likewise, the phonology of tone and prosody is pretty much skipped.

Still, *Phonology in the twentieth century* gives the reader an amazingly clear and thorough view of what have probably been the main paths in twentieth century phonology. Beyond that, the book is well edited, nicely manufactured, and reasonably priced. Buy one before it goes out of print.

Jared Bernstein
SRI International
Menlo Park, CA 94025

Computational tools for doing linguistics

Gazdar, Gerald (editor)
[School of Social Sciences, University of Sussex]
[Special issue of *Linguistics*, 23(2), 1985, 185–360]
Berlin: Mouton
ISSN 0024-3949

Many CL systems aim to achieve language-processing tasks (such as translation or question-answering) by enabling computers to exploit expert linguistic knowledge embodied in some form of scientific language-description. The goal of this book is to survey systems of another kind, whose purpose is to help linguistic experts to create or improve their descriptions of languages whether these are carried out for practical or for

purely scholarly reasons. Not all chapters are equally relevant to this goal. Douglas Biber's (which discusses research on syntactic differences between linguistic genres, described more fully in the June 1986 issue of *Language)* concerns a mathematical technique, factor analysis, rather than a software system (though it is available in several standard statistics packages); and the "Lexicrunch" system presented by Andrew Golding and Henry Thompson seems intended more as a practical language-processing tool than as a generator of accurate analyses (it automatically finds a compact representation for a set of inflectional paradigms, eliminating the need to store inflected forms separately or work out morphological rules). But the other five chapters present software systems which can fairly be described as aids to the scientific linguist.

Three of these systems are grammar-development packages, which enable a linguist to explore the performance of his grammar by watching it parse test sentences step by step. ProGram (described by Roger Evans, Sussex) and GPSGP (John Phillips and Henry Thompson, Edinburgh) are both designed for the development of GPS grammars; ProGram gives the user greater control, allowing him to intervene at major choice-points, but on the other hand it seems to permit only a "toy" lexicon — GPSGP is not stated to suffer from a comparable limitation. PATR-II (described by Stuart Shieber of SRI International, California) copes with a wider class of grammar-types, including for instance lexical-functional and functional unification grammars as well as GPSG (the only systems identified as outside its scope are grammars requiring ordered application of rules, e.g. transformational grammars).

Since these systems are presented as tools of general usefulness, a relevant question is how available they are to the linguistic community. None of the writers addresses this question explicitly, but Evans implies that ProGram is being distributed internationally. GPSGP was funded by the Science and Engineering Research Council, which, I believe, implies that it will be generally available to UK researchers (I am not sure of the position with respect to the EC or the rest of the world); nothing is said about availability of SRI's PATR-II.

Jan Aarts and Theo van den Heuvel of Nijmegen describe two systems, both relating to corpus-based research. (It is interesting to find a book edited by a "mainstream" computational linguist devoting two out of seven chapters — this one and Biber's — to corpus linguistics, which has often been treated as virtually a taboo subject on essentially irrational grounds that are ably dissected by Aarts and van den Heuvel.) Their Linguistic Database is a sophisticated user interface to a corpus of parsed language material, permitting the linguist to search for patterns of many kinds in parse-trees via lucid graphic displays. Aarts and van den Heuvel write of "eliminat[ing] the need for much of the paperwork which has always dominated corpus research"; every corpus linguist will recognize the problem they allude to. The Linguistic Database is intended to be compatible with any kind of grammar, provided the analysis of a sentence is always a single (complexly) labelled bracketing. Aarts and van den Heuvel's other topic, the "Linguist's Workbench", is a tool (not complete at time of writing) for developing a grammar of their own brand (extended affix grammar) within a system which also contains a corpus, thus facilitating the business of testing grammar against diverse and complex examples.

As a tool for exploring the characteristics of a complex body of parse-trees, the "Linguistic Database" has impressed me when I have seen demonstrations; I would be glad to discover how well it works in practice. It is not stated here whether, or when, the system will be generally available. Indeed, Aarts and van den Heuvel do not make it as clear as they might just what their "Linguistic Database" really is: they sometimes suggest that its value lies in the information contained in it, but it should rather be seen as a novel means of accessing such information. Aarts and van den Heuvel believe that the English corpus which they were on the point of implementing on their Database when they wrote is the first extant corpus of parsed natural language, but this is not so: the machine-readable parsed subset of the Brown Corpus of American English described by Alvar Ellegård (1978), which is about the same size as the Nijmegen Corpus (c. 130,000 words), has been publicly available for years. (A copy was recently supplied to us at Leeds by Gudrun Magnusdottir of the Språkdata Institute, University of Gothenburg.) The UCREL group at Lancaster have by now gone a long way towards completing their goal of parsing the entire million-word LOB Corpus, and their product will presumably become available under SERC rules.

The remaining chapter, by Mark Johnson, describes a system which enabled a comparative dictionary of the Yuman languages, including reconstructed proto-Yuman forms, to be compiled and typeset automatically from records dealing with words of individual languages in their modern pronunciations. This is a readable account of what was clearly a valuable application of computing technology to a linguistic enterprise. Johnson does not claim that the software described has general applications.

Gazdar mentions other systems which he was not able to include: Lauri Karttunen's KIMMO morphophonemic package, and two further grammar-development systems. Nevertheless, what he has given us is a useful survey of an area of CL that is less well known than it might be.

Geoffrey Sampson
Department of Linguistics and Phonetics
University of Leeds
Leeds LS2 9JT
England

Reference

Ellegård, Alvar. 1978 *The Syntactic Structure of English Texts: A Computer-Based Study of Four Kinds of Text in the Brown University Corpus* (Gothenburg Studies in English, 43). University of Gothenburg.

Computers in linguistics

Butler, Christopher S.
[Department of Linguistics, University of Nottingham]
Oxford: Basil Blackwell, 1985, ix+266 pp.
ISBN 0-631-14267-3; £19.95

The first thing that's odd about *Computers in Linguistics* is its title. Most readers of this journal would expect a book of that title to be about applying computers as tools in research in theoretical linguistics, or about the contribution of computational linguistics to linguistic theory. That is, they would expect a book similar to the one by Gazdar reviewed above by Geoffrey Sampson. They may feel rather misled, therefore, to find Butler's to be almost nothing of the sort. Rather, *Computers in Linguistics* is about the field that has come to be known as literary and linguistic computing — the application of computers in the language-oriented humanities.

Granted, there are a couple of paragraphs on computational linguistics in the sense used in this journal, not to mention two pages on machine translation and one on corpus linguistics. However, Butler takes as his primary subject matter topics such as the computational analysis of literary style, the production of concordances, and computer-assisted teaching of second languages. There was a time when these topics were considered part of computational linguistics — witness articles in the early issues of this journal—but not any more. And they would be considered foreign matter by most members of departments of linguistics (at least in North America). These days, literary computing has its own society, The Association for Literary and Linguistic Computing, with its own journal, *Literary and Linguistic Computing*.*

The separation of the two areas is understandable; the computer scientists who seem to dominate ACL and this journal tend not to be interested (at least in their professional work) in things like studies of diachronic changes in the vocabulary size and punctuation style of the poetry of Sylvia Plath (to quote one example from Butler), even if computer analysis is involved. However, if one is to judge from Butler's book, the separation does seem to have been to the considerable detriment of the literary computing folks, for one gets the impression that their isolation has caused their field to largely become stuck in the stone age (that is, the early 1970s).

For example, one solid area in which computational linguistics has developed tools that could be applied in literary analysis is parsing. Concordances are useful, but so are structural analyses. For example, Cluett (1976) reports a number of stylistic studies on texts in which every word is tagged with its syntactic category. Working in the early 1970s, Cluett had to laboriously tag each word of his texts by hand. In the latter part of the 1980s, much of this could have been done automatically. But all Butler has to say about parsing is that it is difficult:

> In an automatic analysis, the computer must be able to deduce this information [grammatical function] simply from the sequence of letters, spaces, and punctuation marks. This is clearly an extremely difficult task, and it is not surprising that automatic parsing systems are often less than 100 percent accurate. (p. 15)

Two paragraphs follow showing some of the problems involved, such as the fact that prepositional phrases don't all have the same number of words, but there is not the slightest mention of such solutions as ATN grammars. The only subsequent discussion of parsing is in a description of the OXEYE analysis package, which contains a syntactic analyzer that is apparently rather naive.

I am, however, unfair to the field of literary computing if I take Butler's book as representative of its current state. Most of the references in the bibliography are to work published in the 1970s, or at best before about 1982. In fact, there are many members of the field who are keenly aware of what current computational linguistics can offer them, as is clear from some of the papers at recent conferences (see, for example, Lancashire 1986). It is unfortunate, therefore, that just at a time when there is a renewed possibility of cross-fertilization between the two fields, Butler's book should give such a short-sighted and pessimistic view.

I must now confess that I misled you a little when I told you that *Computers in Linguistics* is not about what its title suggests but rather is about literary computing. For in fact most of it isn't about that either. Only two chapters, totalling 67 pages, are on that topic. The rest of the book is a brief introduction to what a computer is (11 pages) and a primer on the SNOBOL4 programming language (168 pages). And that's another odd thing about the book: why SNOBOL4? Certainly, its powerful pattern-matching features made it the language of choice for literary computing for many years. But it is a language born in the early 1960s, based entirely on labels and go-tos. Development of the language ceased in 1969 (the standard guide to the final version of the language is Griswold et al 1971), and it is very poorly designed by today's standards. Happily, the originators of SNOBOL4 have developed a new language, ICON (Griswold 1983), which has the advantages of the old language but with modern control structures. It is a pity therefore that Butler chose to offer SNOBOL4 when he

* For more information on the ALLC and its journal, write to the society's secretary, Dr T Corns, Department of English, University College of North Wales, Bangor, Gwynedd LL57 2DG, U.K.

could have instead helped to promote the newer and better language.

At this stage, you might think the book is just plain out-of-date, and you'd be half right, which leads to another thing about the book that's rather odd: it's out-of-date and up-to-date at the same time. With regard to technical matters, such as personal computers and the like, the book seems to be an old manuscript that has been superficially modernized; it is up-to-date, but regards the past with fond nostalgia. For example, the section on input devices opens with a long and detailed paragraph on punched cards. It then cheerfully admits them to be obsolete and goes on to newer devices, such as terminals and Kurzweil optical scanners. One gets the impression, however, that the author had an old manuscript with a good paragraph about cards that he wasn't going to throw away, or even reduce to a cursory sentence at the end of the section, just because it was no longer applicable; a new sentence admitting that the preceding paragraph is almost completely useless was the preferred revision. Likewise, the bits-and-bytes hardware orientation of the introductory chapter is reminiscent of a textbook of the 1970s.

In summary, this is a book that does a disservice both to literary computing, the field that it describes, and to computational linguistics, the field that it implicates in its title but gives short shrift in the text.

Graeme Hirst
Department of Computer Science
University of Toronto
Toronto, Ontario
Canada M5S 1A4

References

Cluett, Robert. 1976 *Prose style and critical reading* (New humanistic research 3). NY: Teachers College Press.

Gazdar, Gerald (editor). 1985 Computational tools for doing linguistics. Special issue of *Linguistics*, 23(2), 1985, 185-360. See review in this issue.

Griswold, Ralph E.; Poage, J.F.; and Polonsky, I.P. 1971 *The SNOBOL4 programming language*, second edition. Englewood Cliffs: Prentice-Hall.

Griswold, Ralph E. and Griswold, Madge T. 1983 *The Icon programming language*. Englewood Cliffs: Prentice-Hall.

Lancashire, Ian (editor). 1986 *Conference pre-prints. Computers and the humanities: Today's research, tomorrow's teaching*. University of Toronto, April 1986.

Introduction to Natural Language Processing

Harris, Mary Dee

Reston, VA: Reston Publishing Company, 1985, xv+368 pp.
Paperback, ISBN 0-8359-3254-0

[Editor's note: Two different opinions of this book are presented, in reviews by Virginia Teller and Carole Hafner.]

A field or subfield of science can signal its emergence as a full-fledged discipline in its own right with the appearance of textbooks devoted to the subject, as opposed to edited books of readings, conference proceedings, or chapters in other, more general textbooks. Such is the case, for example, with the relatively new field of Cognitive Science, which has begun to spawn textbooks only in the last year or so (e.g. Stillings et al., 1987). Now that the Association for Computational Linguistics has just celebrated its twenty-fifth anniversary, it is interesting to note that until the last five years the discipline of natural language processing (nee computational linguistics) produced almost no textbooks. Since then the situation has changed dramatically. A number of offerings have been published with each author seeking his or her own niche vis a vis the competition. Besides the book under review here, there have been the primarily syntactic contribution of Winograd (1983), the cognitive science perspective of Moyne (1985), the terse but comprehensive overview provided by Grishman (1986), and most recently Allen (1987), which hasn't yet reached my desk.

Harris's approach reflects both her unique background and the audience for whom the book was written. A former associate professor of Computer Science in the Department of Mathematical Sciences at Loyola University in New Orleans, Harris now works for the Systems Research and Applications Corporation in Arlington, Virginia. She received degrees in Mathematics, German, and English Literature at Texas Tech University before completing a Ph.D. in English Literature and Computer Science at the University of Texas. An extended stint in industry with IBM added leavening to the years in academia. The book, which presupposes no knowledge of Artificial Intelligence or Linguistics and very little Computer Science, is aimed at two sorts of readers, the first being undergraduates or beginning graduate students with at least a basic programming course behind them. But it is also intended as a practical guide for programmers interested in adding natural language capabilities to software systems.

The text's ten chapters are divided into four sections entitled "An Introduction to the Study of Language", "Natural Language Input and Output", "Natural Language Structures and Algorithms", and "Natural Language Systems". Overall the balance between theoretical description and implementation considerations is good, and the liberal sprinkling of exercises and programming assignments throughout the text provides an ample supply of projects of varying difficulty. Although there are some surprising inclusions and a few startling omissions in coverage, Harris succeeds remarkably well in making the fundamentals of natural language processing accessible to a broad readership.

Part I contains a single chapter, "Basic Linguistics". This introduction is so clearly written and even-handed in its treatment of the major modern theories such as generative grammar, generative semantics, and case grammar that I have used it independently to teach students with no previous exposure to linguistics what

the field is all about. Harris demonstrates that she has mastered the skill of simplifying the original literature without introducing inaccuracies, for example, in describing trace theory. The two chapters in Part II are "Text Processing" and "The Lexical Phase". Unusual features of this section include discussions of text storage techniques (e.g. Huffman coding, the MARC format) and efficient methods for retrieving dictionary information (e.g. trie searching). Here and elsewhere throughout the text relevant data structures are reviewed (in this case linked lists), and the main algorithms are given in pseudo-Pascal.

The core of the book is in Part III. Its six chapters deal successively with transformational generative grammar, transition networks, case grammar, semantic networks, conceptual dependency theory, and the representation of knowledge. In each case, the primary literature is covered in some detail and approaches to implementation are presented. The first three of these chapters are more successful than the last three. As Harris moves from the major computational formalisms for *processing* natural language to the schemas for knowledge representation necessary for *understanding* natural language, the source material increasingly appears in a more direct and less digested form that becomes clumsy at times, partly because too much material outside the scope of the book is included. For instance, some of the diagrams borrowed from Brachman in the section on KL-ONE are completely mystifying.

In Part IV a final chapter called "Design of Natural Language Systems" outlines the design of a system capable of "understanding" natural language in the sense of accepting natural language input, extracting and storing relevant knowledge, drawing inferences, answering questions, and generating responses. Based on the conceptual dependency model (an unfortunate choice, in my opinion), the main modules include a parser, an understander, and a generator.

The book can be criticized on several grounds. At a superficial level, the limitations of Pascal as a vehicle for producing NLP systems become all too evident. There is also a tendency throughout to present a neutral or positive interpretation of source material rather than a critical evaluation of alternative or opposing points of view. Several important issues in the field are simply skipped entirely. Parsing, for example, is not considered as a separate topic. A reader relying on Harris as a sole source could come away not knowing the difference between top-down and bottom-up parsers or between backtracking and parallel processing. Although Harris acknowledges (p. 330) that "resolving pronomial reference is one of the major difficulties in natural language processing," the subject of anaphora is not dealt with except in passing, and problems associated with processing discourse units larger than single sentences are not discussed at all.

There are few typographical errors, but an incorrect subscript is given in an algorithm on p. 78, and the reproduction of a table from Lehnert's 1977 book on question answering (p. 330) uses the British form 'judgemental'. The bibliography of well over 100 items, all of them referenced in the text, provides a substantial overview of literature in the field.

Virginia Teller
Department of Computer Science
Hunter College and Graduate Center
The City University of New York
New York, N.Y. 10021

References

Allen, James 1987 *Natural Language Understanding*. Benjamin/Cummings, Menlo Park, California.

Grishman, Ralph 1986 *Computational Linguistics: An Introduction*. Cambridge University Press, Cambridge, England.

Moyne, John A. 1985 *Understanding Language: Man or Machine*. Plenum Press, New York, New York.

Stillings, Neil A. et al. 1987 *Cognitive Science: An Introduction*. The MIT Press / Bradford Books, Cambridge, Massachusetts.

Winograd, Terry 1983 *Language as a Cognitive Process. Volume 1: Syntax*. Addison-Wesley, Reading, Massachusetts.

In the Preface to this book, the author states that it is designed to be a text for upper-level undergraduate Computer Science students, and a guide for programmers interested in adding natural language interfaces to their software products. Unfortunately, the book is not adequate for either purpose, due to its weak presentation of the formal and computational aspects of the field.

The book is divided into four parts: "Introduction", "Natural Language Input and Output", "Natural Language Structures and Algorithms", and "Natural Language Systems". Part I contains a single chapter entitled "Basic Linguistics", which presents an overview of the linguistic concepts and theories (phrase structure, transformational grammar, case grammar, generative semantics) that are of most interest to Computational Linguists. It also gives the reader a basic working vocabulary of lexical categories, phrase types, and grammatical features to serve as examples.

Part II contains two chapters. The first, "Text Processing", covers ASCII coding, data compression, the WEBMARC format used for machine-readable dictionaries in the early 1970's, implementation of character strings using Pascal arrays, SNOBOL (as an example of a string processing language), and implementation of linked lists using Pascal records. The second chapter, "The Lexical Phase" begins with a discussion of morphology, including how one might program a morphological analyzer to remove affixes. The rest of the chapter, entitled "The Dictionary", covers binary search, indexing, trie searching, and (briefly) hashing. Notably missing from this chapter is a discussion of the internal structure of lexical entries. The selection of topics in Part II puzzled me; most of the material presented was not really part of Computational Linguistics, while more relevant work on lexical representation and lexical semantics was not mentioned at all.

Part III contains chapters called "Transformational Generative Grammar", "Transition Networks", "Case Grammar", "Semantic Networks", "Conceptual Dependency Theory", and "Representation of Knowledge". The two chapters on syntax are the weakest part of this text; they simply do not provide an adequate presentation of their topics. The chapter on transformational grammar begins by showing an example of a phrase structure grammar and associated phrase marker. It presents a Pascal representation for phrase markers as trees and develops general algorithms for tree traversal. It then describes how a phrase structure grammar could be implemented as a large case statement at the top level, with each constituent type encoded as a separate Pascal procedure. No examples are given to show how such a constituent-type-procedure would work. A brief section called "Transformational Rules" shows two examples (passive and *there*-insertion), and develops a Pascal data structure for representing transformational rules. This chapter is quite superficial in its discussion of transformational grammar — raising transformations, the need for a cycle in applying the rules, and the problems of rule interaction are not discussed.

The next chapter begins with a definition of transition networks and the Pascal data structures for representing them. A verbal description is given of a recognition algorithm for recursive transition networks. Next, a section on Augmented Transition Networks (ATNs) describes the ATN model and shows a small phrase structure grammar and its ATN equivalent. This section is clear and well-presented. The implementation of ATNs is explored by presenting a complete Pascal program for implementing the *particular* ATN grammar defined earlier in the chapter. The rest of the chapter discusses the process of writing a compiler to translate an ATN grammar into a tabular form that can be interpreted by a Pascal program. The operation of an ATN interpreter that uses this tabular grammar is verbally described.

The next four chapters, on natural language semantics and knowledge representation, are the strongest part of the book. The chapter on case grammar introduces the case-based analysis of a sentence, shows examples of different cases and modalities, and describes a data structure for representing a verb's case frame. It then explores the use of the case-based representation for generating and analyzing surface sentence forms, and for answering questions. It concludes by presenting another proposal for the set of deep cases. In this chapter, the lack of formality in describing how a particular process could or might work did not seem inappropriate, and the important ideas were well-presented.

The chapter on semantic networks begins by reviewing some early work, and then defines a Pascal record structure for representing nodes and links. A brief section entitled "Answering Questions from Semantic Networks" shows that the information is available for doing this, but does not discuss how to do it. The chapter then presents a rather detailed summary of three semantic network models: partitioned, procedural, and extended (propositional) networks. Each section cites the original sources on which it is based, so the student can pursue these topics further.

The chapter on conceptual dependency theory begins by defining the primitive acts and states, and the graph-structured representation of concepts. The concepts are illustrated with well-chosen examples. A section called "Implementing Conceptual Dependency" presents a Pascal record structure for representing states and actions, and then describes the conceptual analysis procedure from *Inside Computer Understanding* (Schank and Riesbeck 1981), which uses a request list to go from natural language directly into the conceptual dependency representation. Two more sections show Pascal data structures for representing requests in the lexicon and causation links between concepts.

The chapter entitled "Representation of Knowledge" begins with a discussion of the critical role of world knowledge in interpreting natural language; it then very briefly mentions some general issues in knowledge representation (e.g., declarative versus procedural knowledge). The next section, entitled "Frames and Scripts", provides a detailed introduction to these approaches, and the last section summarizes Brachman's original KL-ONE model for knowledge representation (Brachman 1979).

Part IV contains one chapter: "Design of Natural Language Systems", which describes a system called NLS (apparently a hypothetical system, since no discussion of its implementation nor examples of its behavior are presented). Sentences are translated by NLS into meaning representations (MREPs) which include the capabilities of frames, scripts, and semantic networks. A section on the details of lexical categories and features seems out of place here; it would have been more relevant in Chapter 3. The next section, entitled "Implementing NLS" describes four modes in which MREPs can be processed: paraphrase mode, inference mode, question-answering mode, and learning mode. A brief explanation of each mode describes, in a general way, what processing would need to be done (the one-page discussion of 'learning mode' does little more than show the need to add new descriptions and facts to a knowledge base in response to declarative input). This chapter does not discuss any specific, implemented NL systems, although there are several that have been developed and used for practical applications. It contains little (if any) concrete guidance for the programmer who wants to build a natural language interface.

In summary, this text has some very serious flaws:

(1) Many of the key ideas in natural language processing are missing. For example, the concepts of derivation and backtracking are not explained; the term "context free grammar" is used but never defined; and

top-down and bottom-up approaches to language analysis are not even mentioned. Predicate calculus is also omitted, although the section on propositional semantic networks includes a brief description of propositions and quantifiers. Worst of all, the problems of ambiguity — lexical, syntactic, and semantic — are not addressed. A person reading this book might not be aware that ambiguity is a problem for natural language processors.

(2) Sections that purport to present algorithms for various tasks do not. Often, there is a brief description of a task or method, followed by a detailed specification of the data structures required for the task, expressed in terms of Pascal declarations. But once the data structures are defined, nothing is done with them; we merely proceed to the next topic. At best, an algorithm may be explained verbally, followed by an exercise that says "implement the algorithm presented in this section". Although there is in-depth treatment of some formal concepts, such as linked lists, binary search, trees, and stacks, this material could have come directly from an introductory text on data structures.

On the positive side, there are good surveys of basic linguistics, case grammar, semantic networks, frames, scripts, and conceptual dependency. The critical importance of common-sense world knowledge and inference ability in natural language understanding is pointed out convincingly. However, because of the problems noted above, I cannot recommend this book as a general text or a reference on natural language processing.

Carole Hafner
School of Computer Science
Northeastern University
Boston, MA 02115

References

Brachman, R. J., 1979 On the Epistemological Status of Semantic Networks. In *Associative Networks: Representation and Use of Knowledge by Computers* 3-16. Edited by N. V. Findler. Academic Press, New York.

Schank, R. C. and C. K. Riesbeck, 1981 *Inside Computer Understanding*. Lawrence Erlbaum Associates, Hillsdale, NJ.

Logical Form in Natural Language

Lycan, William G.

Bradford Books / The MIT Press: Cambridge, MA, 1984, xii + 348 pp.
Hardbound, ISBN 0-262-12108-5, $27.50; Paperback, 0-262-62053-7, $10.95

In this book, William G. Lycan, Professor of Philosophy at the University of North Carolina, Chapel Hill, discusses a selection of important and difficult issues in linguistic and psychological research on natural language. The major theme of the book, as frequently stressed by the author, is a defense of truth-conditional semantics for natural language utterances. In this view, every well-formed sentence/utterance is assigned a semantic representation (logical form) whose truth conditions determine the meaning of the sentence/utterance (or at least the kernel of it). Pragmatics, the context-bound part of a linguistic process, is to be clearly separated from semantics and meaning: the context affects the surface form of an utterance, but it cannot interfere with its truth conditions. Lycan identifies and crushes a number of misconceived (in his opinion) attempts to infuse semantics with extra-semantic elements. The major thrust of his attack is directed against the possible-world semantics interpretation of attitudes, Strawson's notion of semantic presupposition, and existing explanations of the role of performatives and indirect forces in speech acts. Lycan is also taking on Quine's indeterminacy hypothesis, and eventually offers a "computational" model of the human speech center based on both Chomsky's transformational grammar and Davidson-style generative semantics.

The book starts with a lengthy exposition of the author's own view, which he identifies with those of Frege, Tarski, Davidson, and Grice. The first three chapters give a good and interesting presentation of some by-now classic results. A brief historical overview in Chapter 1 is followed by a discussion of the Tarskian theory of truth and its application to natural language in Chapter 2. Chapter 3, augmented with an appendix, discusses semantic representation for more-difficult parameters of a natural language utterance, including deitic elements, tense, and beliefs. The interpretation of belief sentences, as presented in the appendix is, however, far from satisfactory. Lycan is unhappy with the solution proposed within the possible-world paradigm, but he does not suggest anything more acceptable either. The introductory part of the book ends with Chapter 4, in which Lycan attempts a major assault on the Strawsonian notion of semantic presupposition (Strawson 1950). He vigorously attacks the notion of the truth-valuelessness of certain statements whose presuppositions fail. He maintains that the whole problem is the result of a misconception, and explains away a body of empirical data that had once led to it.

The argument is well prepared and forcefully supported by examples, and the reader is easily drawn into it. In the end, however, you feel that you've been cheated somewhere. It is somehow hard to accept that an utterance of "The present king of France is bald", where the definite noun phrase fails to refer to anything, is just simply false. If you cannot swallow this, you'll be presented with the notion of broad grammaticality, according to which the aforementioned sentence is simply ill-formed, and thus it does not make sense to talk about its truth conditions. Ultimately, the lack of a truth value is traded for undecidability upon broad-sense grammaticality (which notion must involve all aspects of a sentence's evaluation, including those context bound). Thus we come out virtually empty handed.

In Chapter 5, the discussion turns away from the

logical form and moves on to the problems of pragmatics and their influence on the surface form of produced utterances. Semantics and pragmatics contribute independently to the process of forming utterances (Lycan is looking at language generation only), with the latter explaining such phenomena as proper lexicalization ("he is X and Y" versus "he is X but Y"), performative prefaces ("I state X"), and indirect force. Chapter 5 is interesting but the discussion grows increasingly arcane and may prove irritating for a reader who is not an insider (especially Chapter 6). Numerous references to the author's other works do not help, and neither do a vast amount of notes placed at the end of the book.

Chapters 7 and 8 are among the best in the book. Chapter 7 starts with an overview of research by Gordon and Lakoff on conversational postulates regarding illocutionary force of natural language sentences. Lycan points out various weaknesses in this account and proposes a generalization in which he classifies all cases of indirect force into three types. Type 1 contains those sentences that in some circumstances can be used to convey indirect illocutionary meaning, and this property is relative to context of use ("It's cold in here"). Sentences of type 2 are normally used metaphorically, though in some situations may be taken as conveying literal meaning ("Have you lost your mind?"). Eventually, type 3 contains sentences that can be used only to communicate their conventional, indirect meaning ("Can you please be a little quieter?"). The reader will find this chapter a source of valuable information, even though there is no definite treatment proposed.

In Chapter 8, Lycan is back to his earlier discussion of truth conditions and meaning. He proposes to build a flow diagram of a computer program that would speak English. Among the few "obstacles" that remain to be to solved before such a program could be written, Lycan lists the problem of disambiguation in natural language understanding, which he classifies as "a special case of the vexing and vicious frame problem in Artificial Intelligence, . . . and not an especially aggravated instance of it". Well, perhaps things are just the other way around. As a computer scientist involved in natural language research, I find most of the author's claims of "computational" paradigms premature and misplaced. In Chapter 11 Lycan presents a schematic diagram (which he calls a flow diagram) of the human generative speech center, which, by any standard, is much too abstract to be considered a computational model.

In Chapter 9, Lycan takes on what he considers the most serious challenge to the truth-theoretic semantics, i.e., Quine's indeterminacy hypothesis (Quine 1960). The topic is of interest to anybody who thinks of automated natural language processing as a series of (possibly concurrent) transformations (or translations) from one representation to another in order to reach, eventually, an ultimate "logical form". But if you look for a theoretical foundation of a new "translation theory" you are heading for a disappointment. The discussion is somewhat confusing and the arguments lack proper force. Eventually you will feel totally at a loss from which you may never recover before the end of the book. It only remains to hope that all this stuff is important and relevant for philosophy, because I just cannot see the significance of the "truth versus V-truth (or even Vc-truth)" argument for computational linguistics or AI in general.

In summary, it is not clear what audience this book is addressed to. I guess it may be of interest in philosophy of language, psychology, and perhaps linguistics. To the AI and CL community it will be of moderate interest: Chapters 7 and 8 are especially worth noticing. The book is also not a primer, so I would not recommend it to somebody who has just entered the field. From this perspective, I think that the book is ultimately mistitled: "logical form" is a trendy catch-phrase that attracts attention and raises expectations which may prove difficult to fulfill.

Tomek Strzalkowski
School of Computing Science
Simon Fraser University
Burnaby, B.C. V5A 1S6
Canada

References

Quine, W. V. 1960. *Word and Object*. The MIT Press, Cambridge, Mass.

Strawson, P. F. 1950. On Referring. *Mind* 59: 320-344.

Understanding computers and cognition: A new foundation for design

Winograd, Terry and Flores, Fernando

Hardbound, Norwood, NJ: Ablex Publishing Corporation, 1986, xiv+207 pp,
ISBN 0-89391-050-3, $24.95
Paperback, Reading, MA: Addison-Wesley, 1987, 214 pp, ISBN 0-201-11297-3, $12.95

This is an important and exasperating book.

What do hermeneutics and Heidegger, autopoiesis and artificial intelligence, commitment and computers have in common? In their book, Winograd and Flores try to explain their own private views of the connections. They are mainly addressing the systems analysis, AI, and computational linguistics communities, warning them against embracing too closely the ways of mathematicians and the advocates of symbolic logic.

The authors perform a useful service by outlining the limitations of the approach that they call "rationalistic" and by calling attention to certain philosophical issues that might prove helpful in the future to establish new directions in computer design. From a mathematician's point of view, they are merely reviving and rerunning the morality play that has already finished its run in

mathematics under the banners of formalism, logicism, and intuitionism.

W&F are too emotionally involved with their subject matter to be mere formalists; they want more meaning than a formalist would be happy with. They explicitly reject the logicist line and urge us to listen to our intuitions and our personal, everyday, common experiences. Their path to the use of intuition and common sense experience is through phenomenology—more precisely, phenomenology of the Heideggerian variety. Although, on the surface, a phenomenologist computer science seems to be a contradiction in terms, the issues raised by W&F are worth careful consideration because the points they are making will be hard to ignore in the future.

Opposition to the rationalistic view

The authors' new vision is helped by focusing on what Maturana and others call *autopoietic systems*. These systems are self-organizing; they are not organized from the outside. Living organisms are the best examples of autopoietic systems. Computers that have to be programmed by an outside agent are not autopoietic. Further inspiration is provided by Maturana's work on frogs, where a lot of what one would be tempted to consider "cognitive" activity was found to be nothing more than biochemistry: peripheral devices, such as the eye of a frog, mechanistically performing their biological function.

Because the book is unusual in nature, it would be unfair not to let the authors state their main points in their own words.

> The key to much of what we [say] lies in recognizing the fundamental importance of the shift from an individual-centered conception of understanding to one that is socially based. Knowledge and understanding (in both the cognitive and linguistic senses) do not result from formal operations on mental representations of an objectively existing world. Rather, they arise from the individual's committed participation in mutually oriented patterns of behavior that are embedded in a socially shared background of concerns, actions, and beliefs. This shift from an individual to a social perspective — from mental representation to patterned interaction — permits language and cognition to merge. Because of what Heidegger calls our "thrownness", we are largely forgetful of the social dimension of understanding and the commitment it entails. It is only when a breakdown occurs that we become aware of the fact that "things" in our world exist not as the result of individual acts of cognition but through our active participation in a domain of discourse and mutual concern. (p. 78)

W&F's basic position centers on their opposition to what they call the "rationalistic approach", i.e., the view that "knowledge and understanding . . . [result] from formal operations on mental representations of an objectively existing world". This rationalistic view was very successful in effecting significant advances in computer work, and in the study of cognition and language. But as expectations about the performance level of computer systems increase, the rationalistic view is proving to be increasingly sterile. New approaches are needed that do not necessarily reject the rationalistic position, but go beyond it.

W&F criticize the rationalistic approach and would want to reject it altogether, rather than striving for a synthesis which incorporates the useful features, while supplementing them with new insights. Perhaps their uncompromising position is necessary in order to highlight the limitations of the older paradigm. But one can't help contrasting their single-minded opposition with more moderate, synthesis-oriented approaches such as, for example, George Lakoff's *experientialism* or *experiential realism*, as described in his recent book *Women, Fire, and Dangerous Things*.

In the authors' words:

> The rationalistic orientation can be depicted in a series of steps:
>
> 1. Characterize the situation in terms of identifiable objects with well-defined properties.
> 2. Find general rules that apply to situations in terms of those objects and properties.
> 3. Apply the rules logically to the situation of concern, drawing conclusions about what should be done.
>
> There are obvious questions about how we set situations into correspondence with systematic "representations" of objects and properties, and with how we can come to know general rules. In much of the rationalistic tradition, however, these are deferred in favor of emphasizing the formulation of systematic rules that can be used to draw logical conclusions. (p. 14-15)

What are the implications as far as computational linguistics is concerned?

> The rationalistic tradition regards language as a system of symbols that are composed into patterns that stand for things in the world. Sentences can represent the world truly or falsely, coherently or incoherently, but their ultimate grounding is in their *correspondence* with the states of affairs they represent. This concept of correspondence can be summarized as:
>
> 1. Sentences say things about the world, and can be either true or false.
> 2. What a sentence says about the world is a function of the words it contains and the structures into which these are combined.
> 3. The content words of a sentence (such as its nouns, verbs, and adjectives) can be taken as denoting (in the world) objects, properties, relationships, or sets of these. (p. 17)

The authors reject this view of language in favor of a very different view. Interestingly, the authors' concept of language is nowhere defined in the book. We don't even have a definition at the casual, snap-slogan level such as "a cultural construct", "a biological consequence", "ability to form utterances", "technology to rearrange mental models", "human activity defined by a grammar", or "conceptual aid for structuring reality". One may speculate whether this omission was inadvertent or deliberate.

Be that as it may, what is interesting about this omission is that it is rather hard to notice. It does not matter that language is not defined, because even in the discussion of linguistic matters, language assumes a very subordinate position. For the authors, natural language understanding and meaning are results of "listening" for "commitment".

> Language can work without any "objective" criteria of meaning. We need not base our use of a particular word on any externally determined truth conditions, and need not even be in full agreement with our language partners on the situations in which it would be appropriate. All that is required is that there be a sufficient coupling so that breakdowns are infrequent, and a standing commitment by both speaker and listener to enter into dialog in the face of a breakdown. (p. 63)

If there is no "breakdown", words are unnecessary. What "communicates" is not only what is specified, but also what does not need to be specified because of a shared background, and a grounding in physical and social reality.

Not only syntax, but even semantics fades into the background. The focus is on pragmatics alone. Natural language does not "bring about" understanding. The understanding is there as a background phenomenon. Natural language is called for—and becomes useful and relevant—when this background understanding breaks down. Use of language signals the lack of understanding. It is almost like a warning light which comes on when a malfunction is noticed. This is the exact opposite of the traditional view that we use natural language to create understanding. No. At best we re-establish understanding; at worst we merely signal that a breakdown in understanding has occurred. Hence understanding is not an act to perform, but a state to be in. If we have to talk, that means that this happy state of affairs has been disturbed.

Implications for computer design

How would W&F's insights about language be used to design and build a computer-based natural language understanding system? The authors do not say. They criticize the established approach which construes meaning as being *in* the message, rather than being *around* the message — in the text, and not in the context — but they do not quite get around to formulating new architectures. They are on the right track, but do not go far enough. The subtitle of the book promises "a new foundation for design". The appropriateness of the proposed new foundation cannot be evaluated without seeing at least a bit more of the design.

While on some subjects they do not go far enough, on one topic they do go somewhat overboard. Their dislike of the term *representation* is strong. What they mean by representation is often not very clear. They mention the dangers of assuming that the representation accurately reflects what is "out there" in a naive realist sense, and raise doubts about the view that cognition rests on the manipulation of symbolic representations. But then they go on to say that the representation is in the eye of the beholder (p. 86). This is a most valuable insight which could have been developed further. In general, we can express this type of insight in statements of the form "X is the representation of Y in the eyes of the beholder Z". This does not seem problematic even in the philosophical framework of the authors. One may guess that the term *representation* was condemned on the basis of guilt-by-association. Symbolic representations have been closely associated with the rationalistic tradition that the authors oppose. This is unfortunate, but some aspects of the symbolic representation concept are worth saving. For example, cognitive science is based on the mental models hypothesis, i.e., that people understand the world by forming mental models. This is a fairly recent view; we just got it and it would be a shame to abandon it so soon. What are we to replace it with? Are Winograd and Flores advocating a new school of neo-behaviorism, in which everything is embodied in hardware (wetware?), and there are no programs and no symbolic representation of world knowledge? How would we design and build such neo-behavioristic computers?

The authors' rejection of representation is not so much wrong-headed as unnecessary. Talking about representation as modeling of the world by a cogniting agent is quite harmless. The enemy is not representation; that is only a symptom. The enemy is naïve realism or objectivism, which blithely assumes, on the basis of one view, one version, one description, one glimpse from one perspective that the essence of an "objectively existing world" (p. 78) has been grasped, and that one knows exactly how this unique world is "really" constructed. In other words, on the basis of the existence of X, it assumes that Y also exists, as in the sentence "X is the representation of Y in the eyes of the beholder Z". Representations are fine, as long as they are construed to be no more than mental models of a publicly examinable kind. But models of what? The eyes of Z dominate the answer to that question. X exists, but Y may not, even if Z does not realize it. In fact, in spite of Z's limited vision, he might find X to be a most useful implement. He may be mistaken in the global sense, but still get the job done using X.

Building models is nothing objectionable, except that one should not attribute more verisimilitude to the model than is required by the modeler. A representation is a representation, and a model is a model, precisely because the observer, beholder, or modeler sees it as such. It is the attempts to escape from this perspectivist framework that create difficulties.

When we consider the following ordered list of statements:

1. "X is the representation of Y."
2. "X is the representation of Y 'out there'."
3. "Z considers X to be the representation of Y 'out there'."

4. "Z considers X to be a representation of Z's mental model Y' of Y 'out there'."

we might notice that the last sentence, although longer than is usually considered convenient for casual use, reflects a humble, modest, experientialist, and basically honest approach.

Computers as autopoietic systems

If computers are not to be programmed using representations of a "real world out there" how exactly are they supposed to function? W&F use Maturana's concept of autopoietic systems. Autopoietic computer systems somehow "self-organize", as opposed to having their programs inflicted upon them in an authoritarian manner by programmers. Certainly, an anthill is "autopoietic", and so is a free market economy. Can we object to programming because it is authoritarian and elitist, in the sense that it is an outsider who inflicts the program on the machine in a non-egalitarian way? What exactly is wrong with this approach?

> It assumes that the programmer (or "knowledge engineer") can articulate an explicit account of the system's coupling with the world—what it is intended to do, and what the consequences of its activities will be. This can be done for idealized "toy" systems and for those with clearly circumscribed formal purposes (for example, programs that calculate mathematical formulas). But the enterprise breaks down when we turn to something like a word processor, a time-sharing system, or for that matter any system with which people interact directly. No simple set of goals and operators can delimit what can and will be done.
>
> The person selects among basic mechanisms that the machine provides, to get the work done. If the mechanisms don't do what is needed, others may have to be added. They will often be used in ways that were not anticipated in their design. (p. 53)

Do Winograd and Flores have a genuinely novel approach to systems analysis? Yes. Their idea is to go beyond the verbal level in order to look at people's interactions with each other and to look for the commitment that underlies these interactions. Rather than going from the verbal level to Newell's "knowledge level", they try to go from the verbal level to an action/intention/commitment level. They claim that what matters is not what people say, but what they do, or intend to do, and the kind of commitment that they are ready to make. It should be repeated that this view of the task of the systems analyst is based upon considering the use of language as the performing of speech acts. This view agrees with the authors' notion of the natural language understanding process as a listening for commitment.

Would such an approach to systems analysis work in practice? That depends on the client. Some clients would feel that with such an approach the analyst is overstepping his mandate by appropriating to himself management functions. He transgresses the limits of his job category by investigating matters which are not within the bounds of his job description.

The limitations of the book are grounded in the experiential limitations of the authors. They are blind to the industrial and commercial domains of discourse, e.g. that computers are built to make money for the vendor, that it takes money to build computer systems, and whoever funds the work will expect in one form or another a return on his investment. Although they emphasize the importance of autopoietic systems being closely coupled to their environments, neither author seems to realize how messy the real world really is. Had they done so, this realization might have driven them back to the neat, well-ordered world of the rationalistic tradition that they criticize.

This reviewer would suggest that the authors should have been looking not at computers as single entities, but rather at the owner-computer complex as a structural unit. Looking at computers in isolation from ownership does not make sense. But this is a symptom of a larger deficiency. In general, it seems that the authors have no industrial experience. Maturana's frogs may be autopoeitic systems, but computers are not. From the industrial perspective, it's hard not to notice that no computer system is ever built unless someone pays for it. Computers, unlike frogs, have owners. It is the owner-computer complex that may be an autopoietic system. We should also note that programmers and analysts do not usually own the computers; they work for people or institutions who do.

The authors want to alter our vision. But they recommend corrective lenses, whereas radical eye surgery, and even some bionic aids, may be required. They are squeamish about money. They do not mention that computers are owned by owners, and that someone must pay for the construction of a computer system, and the person or institution who pays the designer has a lot to say about what kind of design is acceptable. They acknowledge that computers are structurally coupled to their environment, and that both this environmental context and the structural coupling are social in nature. They forget to mention the economics of the structural coupling. The seemingly dirty words of *money* and *ownership* are not prominently featured in the book.

The missing parts

Although W&F seem to be uncompromisingly bold and thorough in their analysis, and in their unflinching criticism of the shortcomings of the rationalistic position, it is curious that there are areas where they hesitate to go further. One of these areas has to do with discourse, and the domain of discourse, such as explored by Michel Foucault; the other area is conceptual analysis and Jacques Derrida's grammatology and deconstruction. Both of these omissions are puzzling, especially because Habermas and Gadamer are discussed. Roland Barthes is not mentioned.

Hermeneutics, or at least one type of the hermeneutical approach, does receive strong support, but phenomenologist social psychology and the sociology of knowledge as, for example, discussed by Abercromby, does not. It's an interesting guessing game to go through the book noting what the authors do, or do not, include, and try to guess the reason why.

Over the months, people have asked me if I like the book. I would answer that "liking" has nothing to do with the matter; there is something much more important at stake. The issues raised by the book are of fundamental importance, and should be kept in the forefront of public debate. Conceptually, the computer field is on the brink of radical changes. Systems analysis, application systems design, and knowledge acquisition are assuming new prominence. It would be desirable to have the changes aligned with larger, humanistic values, as opposed to narrow technical considerations. In that respect, Winograd and Flores light a candle while still cursing the darkness.

Stephen Regoczei
Computer Studies
Trent University
Peterborough, Ontario
Canada, K9J 7B8

Acknowledgement

I am grateful to Graeme Hirst for his comments on an earlier draft.

Reference

Lakoff, George. 1987 *Women, fire, and dangerous things: What categories reveal about the mind.* Chicago: The University of Chicago Press.

Semantics and syntax: Parallels and connections

James Miller
[Dept Linguistics, University of Edinburgh]
(Cambridge studies in linguistics 41)
Cambridge University Press, 1985, viii+262 pp.
ISBN 0-521-26265-8; $47.50

[Editor's note: This book is reviewed twice: by Bruce Nevin and by Barbara Brunson and Geoffrey Laker.]

Miller originally set out in this book to rehabilitate a theory of semantics known as Localism: the idea that everything we talk about either is an object located in space, or is spoken of metaphorically as though it were such an object, situated with respect to other such objects by means of familiar spatial relations.

Some examples give the flavor. *Over twenty students* expresses by its preposition the same spatial relation as *over the wall*, and "the sentence *The blacksmith beat out the horseshoe with a hammer* can be interpreted as presenting the blacksmith in the same location as the hammer, and *John went to the party with Mary* can be seen as presenting John as in the same location as Mary, albeit a changing location, as they travel from one location to another on their way to the party" (p. 123). *The mechanic got the car fixed* is derived from something like *The mechanic moved the car into a state of repair* (p. 174).

Echoes of this hoary notion reverberate from the Greek grammarians down to the terminology of traditional grammar, where for example a transitive relation 'carries' the 'action of the verb' from the subject to the object. There are obvious affinities to notions of case. Traditionally, case covers both syntactic relations, such as subject and object, and semantic notions that are clearly Localist, such as are expressed by the dative, ablative, and locative cases, with a rather foggy region of metaphoric extension between for things like the ablative absolute construction, in which one 'moves' from one action (expressed by a participle in the ablative case) to another. Miller would dispel the fog by extending this sort of metaphor boldly over the whole field of semantics, claiming (p. 119)

> that all constructions can be interpreted in terms of spatial expressions, that spatial expressions are the rock upon which the entire edifice of semantics is built.

It may be that he merely extends the fog.

In favor of Localism, we may look fondly on the relative tractability of physical relations and naive physics as compared with other dimensions of cognitive 'space'; point eagerly to the obvious importance of analogy and metaphor for cognition in general and language use in particular, feeling an understandable desire to get at some root of all analogizing; and cite numerous studies in the psychology and philosophy of cognitive development that advance or suggest some form of localism — for instance, Herskovits (1986) seems to cover some of the same conceptual ground.[1]

There are problems, of course. Miller confesses (p. 86) that while

> it would be convenient if there was a one-to-one correspondence between each [semantic entity] and a [word in the language] . . . language being as it is, a certain amount of vacillation is to be expected.

He illustrates a bit of this "vacillation" with a brief description of some difficulties getting the concepts

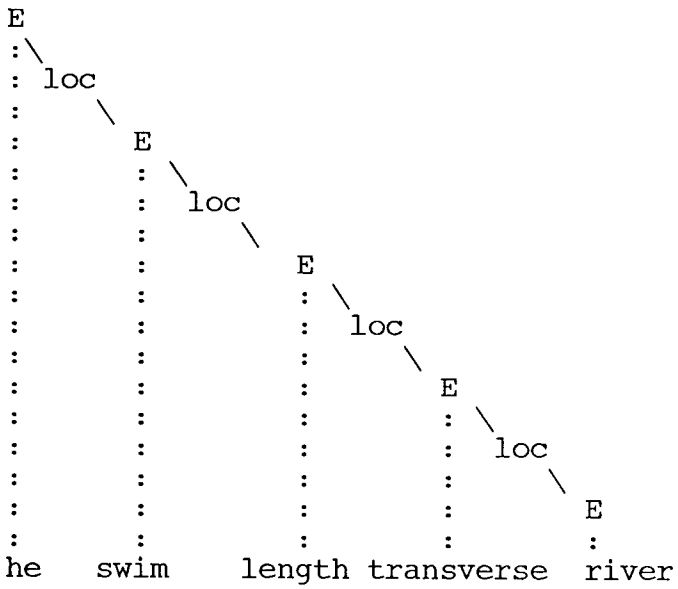
[1] Miller could make his case more effectively if he showed more familiarity with other work. His lexicalist treatment of morphology and syntax would benefit from unification techniques, but he is no computational linguist, and evinces no knowledge of recent CFL (context-free language) work, nor of the problems of knowledge representation (to which his work might well contribute). Even within linguistics, he makes no mention of Langacker's Space Grammar, recently renamed Cognitive Grammar. He opines (and I agree) that generativists would have avoided much needless ramification of theoretical blind alleys if they had followed the work of Zellig Harris more closely. It is a great pity that he himself apparently knows nothing of that work past 1957! Familiarity with Harris's more recent writings might have steered him clear of some unfortunate misinterpretations of the earlier work.

'front' and 'proximity' to behave as his theory predicts they ought. Another instance is the ambiguity of, *e.g.*, 'on' referring either to a top surface ('the book on the table') or a lateral surface ('the picture on the wall'). Quite apart from the spectacle of Miller indicting language for failure to measure up to hypothesis (hypostasy that would be comic were it not so commonplace), one is left with the inescapable impression that he has only, so to speak, scratched the surface.

In fact, application of these ideas to a particular situation is often not at all straightforward and obvious. This means that computer implementations would have to be both preceded and accompanied by intensive analysis of a very large range of linguistic and cultural data by linguists of Miller's persuasion. (We will return presently to the almost universally ignored need for a very large base of empirical data.)

Miller is here concocting a separate metalanguage, a metalanguage with its own dependency grammar and with its own distinctive Latinate vocabulary of 'entities' and 'relators'. The 'relators' are given (p. 146) as *loc*, *all*, and *abl* (for 'locative, allative, ablative', respectively). The list of entities for semantic analysis of English prepositions is given (p. 86) as *int, intspace, ext, extspace, sup, supspace, inf, infspace, ant, antspace, post, postspace, latus, latspace, circumspace*, and *proxspace*. There is evidently a large lexicon of other 'entities' (see below). No explanation is given of their origin or method of establishment, and no proposals as to their testability are offered.

The semantic structure for e.g. *He swam across the river* is given (p. 111) as:



Thus, 'he' is located in his swimming, which is located in the 'length' of the river, which is located in turn in its 'transverse', which finally is located in the 'river' itself. (E stands for 'Entity'.)

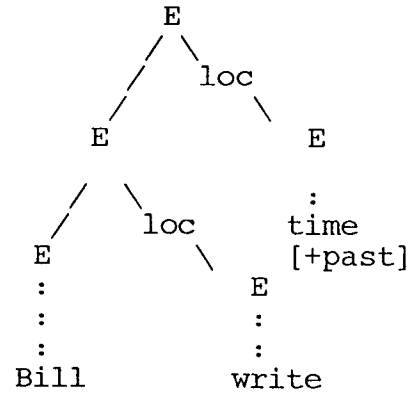
In keeping with Miller's utter partition of semantic from syntactic matters, this semantic metalanguage has no distinctions corresponding to syntactic categories such as verb, adjective, preposition, and adverb: all these are 'entities'. In addition to nouns like *bridge*, *children*, and *house*, which we might have assumed were entities, and verbs like *swim* and *run* which are at least familiar, we have less obvious entities like *transverse* and the list of neologisms given above. He does acknowledge (p. 84) a distinction between entities that correspond to objects ('river') and those that correspond to places ('transverse'). He demonstrates that not all languages make this distinction, or make it in the same way, in their morphology and syntax, and that a given entity may be represented in various ways in the morphology of different languages.

He conflates several 'parts of speech' into one that he, somewhat perversely, continues to call prepositions:

> The hypothesis to be applied is that prepositions — subsuming the traditional categories of preposition, particle and adverb — and verb prefixes correspond to entities in semantic structures and that these entities are the counterparts of nouns occurring in the surface syntax of various languages; e.g., *front, rear, proximity*. (p. 85)

It is principally the semantics of 'prepositions' that Miller uses for illustration of his purposes in this book.[2]

Tense is discussed only briefly (pp. 159–161). Miller's proposed semantic structure for *Bill wrote* suggests how he proposes to handle it:



The dependency grammar of the metalanguage is presented ostensively, rather than descriptively. For example, Miller does not explicitly note the evident fact that there can be entities with zero morphological realization, and that only right branches are labeled as relators.

Questions of formalism aside, there is a deeper problem that goes to the heart of linguistics and natural language research today. This is a *separate* metalanguage, outside and apart from the language it is supposed to be used to describe. But one of the characteristics of natural language that distinguishes it from

[2] For a systematic study of the syntax and semantics of prepositions with many examples Miller might well ponder, see Ryckman and Gottfried 1981.

formal languages is that it contains its own metalanguage as a sublanguage.[3]

If the metalanguage for semantics is separate, then what is its relation to this pre-existing, contained metalanguage? What is its relation to the metalanguage expressions in which the grammar is stated? Or is Miller claiming that there are distinct metalanguages for the different parts of his Firthian 'polysystemic' grammar? Finally, the metalanguage is, after all, a language. What is the semantic interpretation of this 'semantic' metalanguage, and in what form is that interpretation represented? For Miller's 'relator' and 'entity' words (the nodes in the above dependency trees) are not meanings, they are words that are supposed to represent meanings. These are questions that have been considered by few linguists or computational linguists.

Finally, this book stands as an exemplar of the great weakness of virtually all linguistic research in the past thirty years: its dependence on anecdotal evidence. In the relatively narrow scope of Miller's 'prepositions', this book gives us data drawn from an impressive variety of languages. The examples are interesting. Assuredly, they must be accounted for by anyone who wishes to make claims of universality. However, there is no reason to suppose that conclusions drawn for a fragment of semantics and syntax such as this may be extended to the whole of any single language, and judging from the history of attempts at synthesis of such fragments into a whole there is every reason to anticipate the success of Babel for such an enterprise. What is needed is an extensive base of empirical data covering the breadth of syntactic and semantic phenomena for each of several languages, such as the French lexicon-grammar work of Gross and his colleagues at the University of Paris. As Gross has pointed out (e.g. 1979), physicists or chemists would not be permitted to generalize from such a shallow and unsystematic sampling of empirical evidence as has become customary in linguistics.

The history of lingustics is replete with attempts to overgeneralize fragmentary successes over the whole of language. One remembers for example the enthusiasm for componential analysis following its successes with kinship terms and folk taxonomies. This essay in Localism appears to be another such instance.

Bruce E. Nevin
Bolt Beranek & Newman Inc.

[3] This follows from the observation that one may discuss anything in, say, English, including English itself; that ordinary sentences contain overt metalinguistic expressions, such as 'this' and 'say' in the preceding clause, or the cross references in the clause just ending; that native speakers and language learners do not have recourse to a separate, prior metalanguage for learning and understanding language (biologistical claims to the contrary); and that the infinite regress of metalanguages implies an unlearnably infinite set of metalanguage grammars unless the recursion is one of reference rather than of form.

References

Gross, Maurice. 1979 On the Failure of Generative Grammar. *Language* 55(4):859-885.

Harris, Zellig S. 1968 *Mathematical Structures of Language*. Wiley/Interscience.

Harris, Zellig S. 1986 Language and Information, the 28th annual Bampton Lectures in America, delivered at Columbia University October 7, 8, 14, and 15, 1986, to be published by Columbia University.

Herskovits, Annette. 1986 *Language and Spatial Cognition, An Interdisciplinary Study of the Prepositions in English*. Cambridge University Press.

Ryckman, Thomas and Michael Gottfried, 1981 Some Informational Properties of Prepositions, *Lingvisticae Investigationes* 1:169-214

It has long proven difficult to determine strict lines of division between traditional components of grammar. In this book, Miller proposes a revision to traditional notions of morphology, syntax, and semantics. He defines a model of descriptive grammar in which the representations of syntactic and semantic information are (by definition) distinct from one another.

Although his main focus is on syntax and semantics, Miller does briefly discuss his version of morphology, which embodies lexical subcategorizations and feature marking. Affixes, expressed as rules on lexical stems, are triggered by features of the stem and context (expressed as subcategorization).

Syntax in this model is severely depleted when compared to most other models of grammar. Syntax here involves the expression of dependency and constituency relations in a somewhat revised representation of phrase structure. Linear order has little relevance, being subsumed by lexical subcategorization frames.

Semantics is dealt with in considerably more detail. Semantic structures consist of entities and relators. Entities correspond to such things as concrete objects, actions, properties of objects or actions, and spaces. Entities are connected by relators that characterize the localist view of semantics. The localist hypothesis states that semantic structures are representations of the spatial relationships expressed in an utterance. Using traditional case names (e.g. ablative, allative, etc.) as labels for relators, semantic representations are constructed in accordance with localism. Miller devotes a considerable amount of discussion to the traditional theory of parts of speech, which is adhered to in the syntax, but is generalized to the entity/relator distinction in the semantics component.

Throughout the presentation of this model, a substantial amount of data is considered. This data covers a wide range of languages, and is both synchronic and diachronic. Unfortunately, it is sometimes unclear just how the data demonstrates the point being discussed. Irrelevant details of the data are often gone over at length, while the relevant facts are left for the reader to work out. Also, an inordinate amount of weight is given

to diachronic speculation in the development of the theory.

Miller displays an at times caustic distaste for generative linguistics. While some of his criticisms are apt and intriguing, others are unfair (or moot, at the very least). For example, in Miller's criticism of Lieber's morphology model (Lieber, 1981), he maintains that, by allowing affixes to bear category features, she makes no distinction between, for example, the distribution of the affix *-ness* (marked as a noun) and the noun *goodness*. However, Lieber *does* maintain the distinction between stems and affixes, thereby accounting for such distributional facts.

A major source of apprehension about this book stems from the ambitious nature of the attempted task. This results in an abundance of 'promissory notes' of topics and details to be explained in later sections, leaving the reader, at the end, with many unsatisfied expectations. For example, details of the semantic representation are discussed, but not the details of how to obtain such representations from arbitrary sentences. It is difficult to conclude whether the semantic structures are arbitrary and ad hoc or if they will prove to be generally applicable as a theory of semantics. Although Miller makes many strong claims about his theory, it is not at all clear that these claims are justified.

Barbara A. Brunson and Geoffrey K. Loker
Department of Linguistics
University of Toronto
Toronto, Ontario, Canada M5S 1A4

Reference

Lieber, Rochelle. 1981. On the Organization of the Lexicon. Ph.D. dissertation, MIT, Cambridge, Massachusetts.

Principles of Grammar and Learning

O'Grady, William
[Department of Linguistics, University of Calgary]

Chicago: The University of Chicago Press, 1987, xiii+233 pp.
Hardbound, ISBN 0-226-62074-3, $27.50

This book on theoretical linguistics is about linguistic competence and, in particular, language acquisition. What distinguishes this work from most others in the field is the assumptions that provide the basis for the syntactic analyses covering a wide set of phenomena.

In his research program, O'Grady assumes that the underlying principles and constraints of language are biologically determined, but similarities with the majority of other nativists end here. He proposes a dichotomy between special and general nativism, where special nativism, as exemplified by researchers in Government-Binding Theory, postulates that there is an innate language-specific faculty or mental organ. This is contrasted with general nativism, which states that language ability is based or grounded on principles independent of a language faculty. O'Grady, as a general nativist, adopts a categorial framework, and his most interesting innovations are the fundamental cognitive concepts from which he derives linguistic constructs and their constraining principles. The basic concepts, which include adjacency, continuity, dependency, and precedence, are conjectured (with little argument) to be part of a general conceptual base.

From these notions, O'Grady derives linguistic categories and conditions as he examines syntactic relations and processes such as thematic roles, extraction from phrases and clauses, anaphoric relations, extraposition, and quantifier movement. Analytic comparisons with Government-Binding Theory and occasionally with Lexical-Functional Grammar are made and O'Grady's approach is, in some instances, shown to be superior with respect to certain predictions.

As a linguistics or a cognitive science text, *Principles of Grammar and Learning* is important in that it contains a technical presentation of an alternative to special nativism. In general, the force of O'Grady's arguments could be enhanced by a discussion and study of various other cognitive domains incorporating the basic concepts. From a computational view, part of this work's significance is contingent upon whether the approach, the set of principles and conditions, and the resulting grammar can be transparently embedded within a natural language processor. In this way, the theory of competence that O'Grady describes can be shown to provide the integrated core of a performance-oriented natural language understanding system.

Robert J. Kuhns
The Artificial Intelligence Center
Arthur D. Little, Inc.
Cambridge, MA 02140

Aspects of Text Structure: An Investigation of the Lexical Organization of Text

Martin Phillips

(North-Holland linguistic series 52)
Amsterdam: North-Holland, 1985, xii+322 pp.
ISBN 0-444-87701-0; $51.75, Dfl 140.-

On the basis of a post-structuralist theory of language descending from Saussure and Foucault, and of Distributional Statistical Analysis (DSA) techniques devised by W. Moskovich and R. Caplan (1978), Martin Phillips proposes what he calls an "objective, statistical, computer-assisted methodology" for the knowledge-free analysis of non-linear lexical structures in large texts. With the help of Alan Reed's (1978) CLOC concordance-and-collocation generator and D. J. Wishart's (1978) CLUSTAN cluster-analysis program, he tests this method on eight books: five science textbooks, novels by Virginia Woolf and Graham Greene, and Christopher Evans's *The Mighty Micro*.

The Library of Congress calls Phillips's book "discourse analysis", but he chooses terms and phrases such as "text analysis", "conceptual structures", "macrostructure", "collocational patterning", "the typology of texts", "syntagmatic lexical networks", and especially "aboutness." If we could test Phillips's book with his own method, we would probably produce a set of content words such as these, whose associations were networked and represented in digraphs and dendograms. What Phillips's book is "about", then, would be contracted to an economical set of graphs where arcs or edges link points, each of which is labelled with a common "content" word from the text.

In computational science, research of this sort might be called content analysis or content scanning. Phillips's "macrostructures" capture the principal symbols of a text and their associations in a form that might be delivered, for instance, to an indexing system. He claims that he is studying semantic meaning, and I would agree *he* is, but the methodology he outlines has no natural-language understanding capabilities. There is no call here for a sophisticated lexicon, a morphological and syntactic processor, or a knowledge representation system. Phillips's "aboutness" machine systematically reorganizes the non-function words of a text, by reference to their mutual distance relations, into a formal language of graphs. The "words" of this language are empty of meaning except for what the reader can supply from personal experience or from a previous reading of the book. It is necessary to interpret the graphs, then, in order to ascertain semantic content, but in the act of doing so, Phillips intimates, we re-experience "the sensation of aboutness" (p. 26) that we feel on thinking about the meaning of a work after the sentences have all been read and the book put down.

Researchers in the humanities have employed such methods before, though not with Phillips's rigour, and have published in journals such as *Computers in the Humanities* and the various organs of the Association for Literary and Linguistic Computing. Representative text analysis of this kind can be seen in Alastair McKinnon's work (1977) on Kierkegaard, the studies of French fiction by Paul Fortier and Paul Bratley (1985), John B. Smith's critique (1978) of James Joyce's *Portrait of the Artist as a Young Man*, and Continental research by such as Christian Delcourt (Mersch and Delcourt 1979). Phillips certainly invites comparison with this tradition (although he appears unaware of it) because he assumes that the meaning of a content word can be established by the company it keeps. He identifies each such content word as a node, collects all non-function words collocating or co-occurring with each appearance of that node inside a span of four non-function words on either side, attempts to group all such nodes on the basis of these collocations, and displays this distributional network by a statistical technique known as cluster analysis.

Phillips's recipe for teasing macrostructures out of texts begins with extensive pre-processing of the text. First, syntax is ignored: function words are stripped from the text in advance. Distance relations alone, the context, are the basis for defining word associations. Because of computational complexity and memory limitations, many content words must also be passed over unanalysed. Researchers usually select nodes by a heuristic that makes sense to them, given the subject of the text. High-frequency words have claimed most attention, but researchers like McKinnon and Howard Iker (1975) have devised intelligent filters to ensure that nodes used distinctively in a work are included. To get his subset of nodes, Phillips further filters the text by reducing all content words to lemmata (root forms) by removing any morphological markers. Homographs are left as is and phrasal verbs (those distinguished by postpositions) are leveled. Then, to bring the data down to a total manageable by CLUSTAN (no more than 200 items), Phillips cuts away high-frequency and low-frequency lemmata — although these two limits are not stated nor is the choice of middle band of the distribution justified very clearly — and then randomly samples 50 to 80 of the remaining 300 to 500 lemmata.

After node-selection, Phillips begins analysis. For this he returns to the unreduced text with function words, and, identifying the chapter as text interval, then proceeds to "total over each occurrence of each node the number of its co-occurrences with each collocate within the span" (p. 64). The data then form a matrix, the rows identifying lemmata as nodes, the columns the lemmata as collocates, and in the cells, of course, the frequencies of collocation. Using Ward's hierarchical technique (rather than the Density Search technique), CLUSTAN computes a similarity coefficient summarizing the collocational behaviour of each node and outputs analysis in the form of a dendogram or inverted tree diagram.

Phillips's results for individual chapters in the five science textbooks appear in over 70 lexical networks depicted in digraph form. Each of these ties together between two and sixteen lemmata and is often structured in such a way as to reveal a central lemma linked to many outlying lemmata: these generously-collocating words Phillips calls "nuclear nodes". Only about five percent of the vocabulary of a text develops nuclear concepts in this way. (A large number of other lemmata without any obvious grounds for being associated are linked in a "rag-bag" network as the CLUSTAN program completes its analysis.) Networks, Phillips argues, establish the "meaning potential" of a nuclear node: they narrow down the range of meanings of the nuclear nodes as well as allow its new meanings to develop. Such networks literally construct text subject matter.

Phillips suggests that evidence for macrostructure appears where at least three lexical networks from two different chapters overlap sufficiently to be superimposed and where at least one of the identical lemmata in each pair of networks functions as a nuclear node.

Phillips thus develops second-order graphs, where nodes are lexical networks rather than lemmata, and third-order graphs, where nodes are chapter numbers rather than lexical networks. For each of the five science texts, Phillips is able to quote the author's own comments verifying that chapters were intentionally linked, or not linked, in just the way that his own networks show. The macrostructure for one text, Gareth Morris's *A Biologist's Physical Chemistry*, is particularly striking: Morris says his ten chapters are all self-contained, and Phillips's analysis only links three of them, and that in a way explicitly mentioned by Morris.

Phillips then turns to infer a theory of text structure from these findings. He begins by categorizing the possible relations one chapter may have to others: a "source chapter" links to one or more later chapters; a "goal chapter" links to one or more earlier chapters; a medial chapter links to one preceding chapter, and one or more later chapters; a pivot ties a preceding source to a chapter following the pivot; and an isolate stands alone. These in turn form three types of segments. Any work with a source, a goal and either a medial or pivot is "sequential". Anything with only a source and a goal is "synoptic". Isolates form segments all by themselves. Segmentation, in turn, allows us to characterize macrostructure itself in terms of unity and directedness. The more isolates, sources, and goals a text has, the more differentiated it is; the fewer, the more integrated. The number of pivots determines whether a work is uni-directional or "serial" (few pivots) or "parallel" (several or many).

From the perspective of others at work in the field, the final two chapters of Phillips's book, on a typology of texts, hold considerable interest. By contrasting science texts with non-science ones, he tries to split texts into either camp by determining to what degree they are capable of revealing macrostructure. If his conclusions are right, Phillips has contributed something new to the understanding of how language determines genre.

He tells us that literary texts are capable of no more than trivial lexical networking and so of macrostructural features. The results of his analysis revealed "virtually no evidence for macrostructure" in either Greene or Evans. Phillips explains this anomaly by pointing out that the three non-science texts (the third is Woolf) tolerate exceptional freedom in their choice of collocates: the total number of distinct collocates (the collocate-types) is high in comparison to the total number of times actual collocation takes place (the collocate-tokens; Phillips names this relationship the "range index"). The more different collocate-types there are, the less frequently will any given node-collocate pair occur; the fewer repeated pairs are found, obviously the fewer lexical networks there will be. Paul Fortier's use of synonym dictionaries to bring different content words under one canonical "thematic" form before proceeding to distributional analysis elsewhere tallies with Phillips's conclusion here.

Science texts, Phillips would have us believe, take on a macrostructural form in direct proportion to the leanness or conceptual integrity — a literary critic might be forgiven to add "poverty" — of their vocabulary. Any subject that has created a special lexicon to express its objects and their relations would tend to communicate just such macrostructures to books written on them.

Yet Phillips's scepticism about the existence of lexical macrostructures in non-science texts does not sit easily with the evidence he proffers. Nowhere are there examples of the lexical networks he finds in these three books, although he tells us, for example, that Woolf's *Mrs. Dalloway* has more nuclear nodes (272) than three of the science texts (186, 219, 219). He also complains that the apparently rich macrostructure of *Mrs. Dalloway* melts away once one excludes from analysis "the similarity of networks containing proper names and titles" (p. 204), although these — together with the lemma *said* — comprise many of the networks in both novels. It is unclear why Phillips thinks we should exclude proper names and titles as lemmata. Characters in novels function much in the same way as key concepts like *energy* and *force* do in scientific writing. Perhaps the "rich macrostructure" Phillips fails to show us says something valuable about the way Woolf related her characters.

The quirky treatment of networks in literary text undermines some of the large claims Phillips makes in the final two chapters of his book. He says that the meaning of literature cannot be described, only evoked, that literal meaning does not operate within the literary imagination, and that the truth of literature can only be verified by intuition. Science deals with "the actual world"; literature with "subjective experience". This "simpler nature of the relationship between text and reality in science text leaves its trace in the text as macrostructure" (p. 230).

Northrop Frye said in his *Anatomy of Criticism* (1957) that if literary critics like himself were not doing science, then they might as well not do criticism at all. There is some truth that we "appreciate" (value) literature only by experiencing emotionally its actions, characters and themes, but is that not also true of novices in the sciences? True disciples are won by "unfair means" like love and awe before they undergo the discipline of hypothesis, reason, and evidence, no less in analysing *Hamlet* than in learning chemical reaction engineering. One of the surest ways to misinterpret *Hamlet* is to assume that his world operates according to the laws of the subjective inner life which you live, a world that has little to do with the "actual" world of early 17th-century London, which many literary critics believe operates according to principles as verifiable as we may find anywhere.

Yet Phillips gives a lucid, intelligent explication of

what is surely one of the most interesting and solid experiments in distributional linguistics ever undertaken. His choice of large texts and of knowledge-free, parse-innocent methods seems antiquarian in a decade of limited-domain expert-systems and ATNs, but his good results are rather shocking. He has every reason to believe his book holds the seeds of a research programme into issues like "the formal identification of functional vocabulary classes". A restructured doctoral thesis, *Aspects of Text Structure* does credit to the distinguished linguistics research being done at the University of Birmingham.

Ian Lancashire
Centre for Computing in the Humanities
University of Toronto
Toronto, Ontario, Canada M5S 1A5

References

Fortier, Paul A., and Paul Bratley. 1985 Themes and Variations in Gide's *l'Immoraliste*. *Empirical Studies in the Arts* 3.2: 153-70.

Frye, Northrop. 1957 *Anatomy of Criticism*. Princeton University Press, Princeton.

Iker, Howard P. 1975 SELECT: A Computer Program to Identify Associationally Rich Words for Content Analysis. *Computers and the Humanities* 8: 313-19; 9: 3-12.

McKinnon, Alastair. 1977 From Co-occurrences to Concepts. *Computers and the Humanities* 11: 147-55.

Mersch, G. and Christian Delcourt. 1979 A new tool for stylistics: Correspondence analysis In: Ager, D. E., F. E. Knowles, and J. M. Smith, Eds., *Advances in Computer-Aided Literary and Linguistic Research*. University of Aston, Birmingham: 265-70.

Moskovich, W. and Caplan, R. 1978 Distributive-Statistical Text Analysis: A New Tool for Semantic and Stylistic Research. In: Altmann, G., Ed., *Glottometrika 1*. Quantitative Linguistics. Studienverlag Dr. N. Brockmeyer, Bochum.

Reed, Alan. 1978 *CLOC User Guide*. Birmingham University Computer Centre, Birmingham.

Smith, John B. 1978 Computer Criticism. *Style* 12: 326-56.

Wishart, D. J. 1978 *CLUSTAN User Manual*. 3rd ed. Program Library Unit, University of Edinburgh.

Revue québécoise de linguistique
14(2), 1985: Linguistique et informatique

Université du Québec à Montréal
ISSN 0710-0167; ISBN 2-89276-016-X
[Subscriptions: Service des publications, C.P. 8888, Succ. A, Montréal, Canada H3C 3P8. One year: $21.75; Two years: $40.00 (Canadian funds). Add $1.00 for postage outside Canada.]

This issue of the *Revue québécoise de linguistique* contains six articles — five in French and one in English — devoted to computational linguistics. Three of the articles describe work done at the Laboratoire d'automatique documentaire et linguistique (LADL), Paris, while the others concern research done at two Montreal universities. Because much of its content is introductory or has been published elsewhere, this collection of papers is rather disappointing.

The first article, "Théories syntaxique et théorie du parsage: quelques réflexions" ("Syntactic theory and parsing theory: Some reflections") (40 p.), by Jean-Yves Morin (Université de Montréal), is a review of the multiple aspects and dimensions of syntactic parsing — space/time complexity, ambiguity, determinism, relations between syntactic theory and parsing theory, linguistic coverage, choice of parsing strategy, etc. What Morin offers is primarily a linguist's view of the parsing problem, emphasizing aspects which are often paid too little attention in the literature, such as the problem of linguistic coverage or the relation between syntactic theory and parsing theory. As such, this is an interesting overview of the parsing problem, even though some of the issues that Morin considers (e.g. determinism) probably deserve a more detailed discussion. Perhaps the most important section of the paper, both in interest and length, is the one on ambiguity, in which Morin gives both a classification of ambiguity types and a brief review of the available options in dealing with PP-attachment ambiguities. Finally, the article includes a rich and interesting bibliography (over 130 entries).

In "Syntactic analysis and semantic processing" (16 p.) Morris Salkoff (LADL), discusses the role of syntactic analysis in "real world" sentence analysis and its relation to semantic analysis. Considering the problem of analyzing scientific articles (molecular biology), Salkoff argues that systems based on semantic networks are impractical because they require far too much domain-dependent knowledge. ("It is most unlikely that such a network will ever be set up for any scientific domain of interest." (p. 61)) He shows, on the other hand, that a system based on a very fine syntactic analysis, coupled with a detailed lexical subclassification, can save much time and effort, by eliminating many ambiguities. He suggests a two-step analysis in which sentences first undergo a strictly syntactic parse. The resulting structures, which are syntactically coherent, can undergo the second step, in which lexical sub-class information (including selectional restrictions) is used to eliminate many semantically incoherent analyses.

"Un survol des recherches en génération automatique" ("An overview of research in language generation") (38 p.), by Laurence Danlos (LADL), is a slightly modified version of the second chapter of Danlos (1985), offering a good introduction to some of the main issues in language generation. Danlos considers that the primary object of language generation is the production of texts from abstract semantic representation — as opposed to the production of text generation from syntactic structures, as used in most translation systems. From this perspective, she recognizes three classes of interacting problems and shows how they have been handled in recent work: (i) conceptual problems, i.e. what information must appear in the output text; (ii) linguistic problems, i.e. how to organize this information in sentences, how to select words and phrases; and finally (iii) syntactic problems, i.e. how to generate

sentences that are well-formed with respect to the rules of the grammar (word order, agreement, etc). The last section is devoted to a rather brief description of her own generation system, developed at LADL. Arguing that conceptual and linguistic choices cannot be taken independently of each other, Danlos proposes a system consisting of two modules. The first module takes conceptual as well as linguistic decisions based on a grammar of discourse that integrates conceptual and linguistic information. The second module, which is purely syntactic, converts sentence schemata into well-formed sentences.

"Un analyseur syntaxique du français" ("A syntactic analyzer for French") (16 p.), by Henri Labesse (LADL and Université de Paris-Sorbonne), is a short paper describing a small syntactic parser for French based on the "string grammar" formalism. Much of the paper is devoted to rather unexciting problems, such as punctuation, agreement, and elision. (The question of how to handle capitalized words receives more attention than word order!) For each of these problems, Labesse proposes the same kind of solution: enumerate all the possible strings in the grammar. It should therefore not come as a surprise that the grammar for the determiner system given in the appendix contains well above one hundred rules.

"La structure des données et des algorithms en Déredec" ("Data structures and algorithms in Déredec") (26 p.) by Pierre Plante (Université du Québec à Montréal) is a presentation of Déredec, a Lisp-based software environment for text analysis. According to Plante, Déredec is more appropriate than general purpose languages such as Lisp or Prolog as a programming environment for linguistic manipulations. This may well be true, but Plante's article, which reads too much like a programming manual, does not provide clear arguments in support of this claim.

"Un exemple d'exploration linguistique du français à l'aide de Logo" ("An example of linguistic investigation using Logo") (14 p.), by Louisette Emirkanian and Lorne H. Bouchard (Université du Québec à Montréal), discusses some aspects of a Logo program that converts non-reduced coordinate structures into their reduced forms, according to three schemata.

Eric Wehrli
Department of Linguistics
UCLA
Los Angeles, CA 90024

Reference

Danlos, Laurence. 1985. *Génération automatique de textes en langues naturelles*. Paris: Masson. [Translated as *The linguistic basis of text generation* (Studies in natural language processing) Cambridge, England: Cambridge University Press, 1987.]

Relevance: Communication and Cognition

Sperber, Dan and Wilson, Deirdre
[Centre National de la Recherche Scientifique and Université de Paris X, and University College London, respectively]

(The language and thought series)
Cambridge, MA: Harvard University Press, 1986, viii+279 pp.
ISBN 0-674-75475-1 (cloth), $25.00; ISBN 0-674-75476-X (paper) $8.95

Although Grice's work on the logic of conversation has been widely cited in the pragmatics literature, the ideas presented in this work were extremely vague and incomplete. Sperber and Wilson (S&W) set out to develop Grice's ideas into a more explicit theory of communication and cognition, with the notion of relevance at the heart of the theory. The book consists of four lengthy chapters entitled Communication, Inference, Relevance, and Aspects of Human Communication. As might be assumed from the titles of these chapters alone there is much in the book of interest to workers in pragmatics, discourse, and computational linguistics.

S&W present an inferential model of communication which they contrast with the traditional code model underlying most work within linguistics. They argue that a code model is insufficient on its own to account for the discrepancy between the semantic representations of utterances and the thoughts actually conveyed by utterances where some of the information conveyed is implicit rather than explicit. They then go on to develop a theory of non-demonstrative inference by means of which hearers create assumptions based on the communicator's ostensive behavior. This sort of inference, although involving the use of deductive rules, is unlike logical inference, where the results are guaranteed. Communication, they reason, is a matter of degree, so that a major challenge for any theory of communication is to give a precise description and explanation of its vaguer aspects. In this respect, non-demonstrative inference takes the form of "suitably constrained guesswork" (p.69).

There are two major components in S&W's theory of relevance:

1) contextual effects;
2) processing cost.

In order to be relevant, an utterance must have some contextual effects. Contextual effects involve the addition of new information to a context of old information, thus leading to a modification of the context. The result is that interpreting an utterance involves more than working out what assumptions are being conveyed; it also involves working out the consequences of adding this set of assumptions to a set of assumptions which have already been processed, i.e., seeing the contextual effects of the assumptions in a context determined partly by earlier acts of comprehension. An assumption with no contextual effects in a context is irrelevant in

that context. Thus having contextual effects is a necessary condition for relevance. The greater the contextual effects of an utterance, the greater its relevance.

Processing costs, on the other hand, reflect the amount of effort involved in working out the assumptions conveyed by an utterance. Here the greater the processing effort the lower the relevance. In other words, relevance is defined in terms of the conjunction of two conditions:

1) an assumption is relevant in a context to the extent that its contextual effects in this context are large;

2) an assumption is relevant in a context to the extent that the effort to process it in this context is small.

Having laid these foundations, S&W turn to the main issue in their theory — the role of relevance in communication. Relevance is discussed in terms of a guarantee from the speaker to the hearer that the communication is relevant. Thus each ostensive act of communication carries a guarantee of its own relevance and it is the task of the hearer to work out, by means of inferences, which assumptions the speaker is trying to convey. Though simple, this principle has far-reaching implications for the role of context in communication. It is usually assumed in the pragmatics literature (and also in AI work on natural language processing) that processing is done in the following order:

1) determine the context;

2) interpret the utterance;

3) assess the relevance of the utterance.

In S&W's theory, context is not given in advance of the interpretation of an utterance, but rather the determination of a particular context arises from the search for relevance. In other words, the order of processing for S&W is:

1) process the utterance in the hope that it is relevant;

2) select a context that will justify that hope.

Here relevance is treated as given and the context is the variable to be determined. As can be seen, the traditional method employed in much of AI, where a predetermined context constrains interpretation, is thus turned on its head, as it is argued that such a predetermined context would in principle have to make reference to such a vast amount of potentially relevant information that it would be psychologically implausible. Thus we have a contrast between what strategies are used to make a model work within a limited domain and the processes which are necessary for a psychologically realistic explanation of verbal communication — an issue central to the differences between AI and cognitive science.

In the final chapter, the theory of relevance is applied to issues such as poetic effect, style, metaphor, and irony. It is argued that these non-literal (and for some, perhaps peripheral) aspects of communication are really extensions of normal communicative processes and that they do not require any special interpretive procedures but are the natural outcome of general abilities used in verbal communication. In this way it is claimed that relevance theory can explain verbal communication as a whole.

Much of S&W's argument rings true and the arguments are presented clearly and logically, supported by numerous (invented) examples which are discussed in considerable detail. Many important issues in pragmatics, such as inference, the mutual knowledge hypothesis, explicature and implicature, and speech act theory, are covered in some detail. The book also provides a good account of differences between the code model of communication and the inferential model. For these reasons alone the book should be a valuable reference source for those interested in communication and language.

There are some, however, who will find difficulty with S&W's theory. Psychologists will no doubt be concerned that S&W's desire to construct a "psychologically realistic model" does not lead them to consider any experimental research findings in cognitive psychology. Their evidence is introspective, in keeping with their notion of a logically valid model. For the same reasons this approach will prove less than adequate to discourse and conversation analysts, as the authors totally avoid any reference to naturally occurring conversational data. This is unfortunate, as it has been shown that the sequential processes of conversation provide a basis for utterance interpretation. On the one hand, participants in conversations are engaged in a continual process of interpreting each other's utterances and (most crucially) of displaying these interpretations — often implicitly — in their subsequent talk. These interpretations are then available for inspection and if they run counter to what the speaker intended, for example, by not deriving the appropriate contextual effects, the speaker can repair this situation by drawing out the intended assumptions more explicitly. Furthermore, sequentially based approaches to the analysis of conversation point to the predictive implicativeness of utterances. Each utterance sets predictions as to what might legitimately occur next and what does actually occur is then interpreted in the light of these predictions. Thus following an utterance interpreted as a question, a next utterance will be inspected for its relevance and if it can be heard as an answer, then it will be heard in that way. Empirical studies of naturally occurring talk suggest that this is how conversation works.

S&W's conditions on relevance look as if they are pointing in the same direction. It is unfortunate therefore that a large body of potentially informative literature on verbal communication has escaped discussion and that the theory of relevance has not been illustrated and tested against naturally occurring data.

Finally, there is the question of what the theory has to say to computational linguists. The role of context in S&W's theory and how it contrasts with its role in much of current natural language processing has already been mentioned. Indeed, S&W present a much richer view of

context than has been addressed in current work in natural language processing, touching on many important issues which are only beginning to be recognized, such as the role of the conversational history of an utterance in its interpretation and the extent to which speakers maintain and update a model of their conversational partners. The theory also implies that interpretation is deterministic in that listeners do not search for all available interpretations and then decide which is the most reasonable one, but that they assume that the first interpretation selected which is consistent with the principle of relevance is the correct one. The principle of relevance thus constrains the search space of relevant interpretations.

As it stands, S&W's theory of relevance is far-reaching and provocative. At the moment, it is probably programmatic rather than programmable, and indeed it remains to be seen whether the vaguer aspects of communication, which are not a flaw but an inherent quality of communication between humans, will be amenable to computational modeling. It is to S&W's credit that they have laid the groundwork for further explorations within this important area of cognitive science.

Michael F. McTear
Department of Linguistics
University of Hawaii at Manoa
Honolulu, Hawaii 96822

From Simple Input to Complex Grammar

James L Morgan
[Institute of Child Development, University of Minnesota]

(The MIT Press series in learning, development, and cognitive change)
Cambridge, MA: The MIT Press, 1987, xiii+223 pp.
Hardbound, ISBN 0-262-13217-6, $22.50

From some points of view, natural language appears to be unlearnable, or nearly so: children cannot, it would seem, learn what they do learn, or only with substantially more difficulty that they seem to have. This unlearnability stems from learnability proofs that show that the language learner cannot correctly converge on a characterization of the grammar that generated the language being learned without various implausible assumptions about the learner's abilities or capacities. This book proposes that the input information available to the language learner is actually richer than assumed by prior learnability proofs, and that the additional input information is what renders learnable what was previously unlearnable. This information is "bracketing", and consists of information about the hierarchical syntactic structure of the input. The proposition that a child learning language receives bracketed strings of words as input rather than merely strings of words is termed the "bracketed input hypothesis". Previous research has provided evidence that children do, in fact, receive bracketed input, and has argued that it may facilitate language acquisition. The current work presents additional evidence that children receive bracketed input, and argues that bracketed input not only facilitates language acquisition, but is in fact necessary and is the key additional information which renders language learning possible for the child.

The first chapter of *From simple input to complex grammar* reviews the overall problem of language acquisition from the author's point of view, and sketches his solution. The second chapter summarizes the results of research into the formal problem of learnability of natural language as background. Chapter 3 presents a formal proof that transformational grammars are learnable given bracketed input. Since this result is meaningful for acquisition only if children actually receive bracketed input and can be shown to use it, Chapter 4 is devoted to a presentation of research results that support the proposition that acoustic bracketing information is available in the child's speech input. Chapter 5 presents research results indicating that children acquire certain constructions which could only be acquired if they used bracketing information during learning, and other results from the manipulation of bracketing information in "miniature language" learning experiments with adult subjects. These results seem to show that bracketing information is indeed used. At this point, the author's argument is complete:

(1) Natural language can be learned if bracketing information is included in the input (Chapter 3).

(2) Bracketing information is included in the input (Chapter 4).

(3) Children learn things that could only be learned if they use the bracketing information in the input (chapter 5).

Chapter 6 concludes by reviewing a number of open questions and areas of uncertainty.

The book is well thought-out and tightly argued, and appears to make an important contribution to questions of how children acquire language. As the author points out, the research presented is exploratory, rather than conclusive; nonetheless, a strong case is made. The author appears to have taken the formal learnability approach to studying language acquisition another step toward explaining how children could acquire language, and grounded that step in empirical research strongly suggesting that the author's explanation is correct. Thus, researchers active in this area and others with an interest in language acquisition will find this book valuable.

However, this reviewer finds the arguments presented in this book less than fully compelling. *From simple input to complex grammar* is based squarely within the transformational paradigm and within the formal learnability paradigm. Although an excellent contribution to these paradigms, if one does not share

them, then the significance of his book is reduced. Moreover, it is not clear what this book has to say about children. The author states, "Our intuitive understanding of what it means to succeed in acquiring language is . . . that, for any language, a child exposed to a sample of that language can induce a [transformational] grammar that will completely generate that language" (p. 3). From this reviewer's perspective, viewing the problem of language acquisition as that of acquiring a transformational grammar is an exceptionally limited one, since the scientific community has not yet come to the consensus that the transformational approach is correct. Further, language acquisition is a lot more than syntax acquisition, and all the components surely interact very closely; any approach which neglects meaning, inference, world knowledge and reasoning risks missing the central problems of language acquisition.

Thus, this book represents a strong contribution to a particular line of research on language acquisition, and those for whom this line of research is of interest will find this book important. Whether it represents a fundamental result in language acquisition, or merely an interesting approach of as-yet-undetermined overall relevance is not yet clear.

Mallory Selfridge
Department of Computer Science and Engineering
University of Connecticut
Storrs, CT 06268, U.S.A.

Computational Complexity and Natural Language

Barton, G. Edward; Berwick, Robert Cregar; and Ristad, Eric Sven
[Artificial Intelligence Laboratory, Massachusetts Institute of Technology]

(Series on computational models of cognition and perception)
Cambridge, MA: The MIT Press / Bradford Books, 1987, xii+335 pp.
Hardbound, ISBN 0-262-02266-4, $24.95

The disciplines of theoretical computer science and modern syntactic theory share origins in Chomsky's early work on mathematical linguistics, but have regrettably diverged since that time. The development and maturation of syntactic theories based in part on computational notions has occasioned a reversal of this trend. Barton, Berwick, and Ristad (henceforth BBR) have been contributors to the effort towards reversal, and this book is a synthesis of their recent results. Its concern is the application of complexity-theoretic lower bound techniques for decision problems — in particular, those results that show that no algorithm using a bounded amount of resources can solve all instances of the decision problem. The results are appealing, because one does not need to assume that any particular representations or algorithms are actually used in human linguistic processing, but only that the processing method is somehow represented in the brain as it would be on a conventional computer.

The question arises as to the relevance of such results in linguistics proper. Complexity theory is not about natural languages, but about computation. BBR's answer is that by focusing on inherent difficulties in linguistic processing, we can isolate those parts of a grammatical theory that would make it count as "natural": potentially part of either the human processing mechanism or part of grammatical competence. Those aspects which are inherently difficult to process should be at least considered as unnatural aspects. BBR study several problems in various theories, principally in generalized phrase-structure grammar (GPSG), lexical-functional grammar (LFG), and computational morphology.

It is difficult to tell whether the complexity considerations in the book actually inform us about naturality. BBR seem to be ambiguous on this point. They make no claim that complexity is a measure of empirical adequacy of a theory: the ability of the theory to account for the perceived regularities of structure. They do seem to acknowledge, at least implicitly, that processing considerations and limitations play a role in the construction of a theory. This seems a reasonable hypothesis to me. Most of the standard theories since transformational grammar can be stated in symbolic terms, and most of the substantive and universal constraints on grammar have been stated in ways such that, given a representation of linguistic structures, a computer can check the constraints. Processing — the reconstruction of structure from actual utterances — and generation — the inverse operation — are both typically modeled computationally, so that it makes sense to study the resources required.

The book consists of essentially four sections. The first section (Chapters 1–3) contains introductory material; the second (Chapter 4) treats LFG; the third (Chapters 5–6) treats computational morphology, especially the approach called the KIMMO system using finite state automata; and the last studies notions arising in GPSG. In order to discuss the points raised in these sections, a certain amount of technical vocabulary is necessary, so I will review this next.

The part of complexity theory used in the book is commonly called the *intrinsic complexity of language recognition*. A language recognition problem consists of trying to determine whether or not some string is in a set of strings specified in some formal way. It is crucial to understand that the term "string" need not refer to the representation of potential sentences as finite sequences of vocabulary items, but can refer to the linear encoding of entire grammars and other linguistic entities. This is important for BBR, because they distinguish the *fixed language recognition problem* (FLR) from the *universal recognition problem* (URP). The FLR fixes a grammar

G and asks whether or not a string w over the terminal symbols of G is in the language $L(G)$ generated by G. The URP, on the other hand, refers to the single set of strings $\{\langle G,w\rangle\} : w \in L(G)$, where G ranges over some class of grammars.

The term *intrinsic complexity* refers to the amount of time or space used by the best possible algorithm for telling whether or not a string is in a given language (set of strings). Thus, if a problem is undecidable, no algorithm will work correctly in all cases. If a recognition problem is in P, the class of polynomial-time solvable languages, then there is an algorithm which for a string of length n, will decide in time $p(n)$ whether or not the string is in the set, where p is some polynomial. The class P is generally conceded to be the most reasonable class of practically solvable decision problems. If a problem can be shown to be outside of P, then there is no hope of implementing a reasonable algorithm which will solve all instances of the problem.

Problems (languages) are generally classified by the amount of time or space required for their solution. Thus, the classes $DSPACE(s(n))$ and $DTIME(t(n))$ are the classes of problems which can be solved by deterministic Turing machines within space $s(n)$ and time $t(n)$ respectively, where the bounds $s(n)$ and $t(n)$ are nonnegative-valued functions of the length n of the input string. Thus P is the union over all polynomials $p(n)$ of the classes $DTIME(p(n))$. We also have the classes $NSPACE(s(n))$ and $NTIME(t(n))$, which are the problems that can be solved by nondeterministic Turing machines in the given bound. (A nondeterministic machine accepts the input if some sequence of choices leads to an accepting state.) Thus the class NP is the union of the nondeterministic time classes ranging over all polynomials.

Finally, we say a problem is *hard* for a class if any other problem in the class can be reduced to the solution of the given one. That is, for each of the other problems, there must be an efficient algorithm (generally working in polynomial time) such that instances of the other problems can be transformed by the algorithm into instances of the given one, in such a way that positive answers to the transformed instances exactly correspond to positive instances of the other problem. A problem is *complete* for a class if it is hard for the class and actually in the class itself. Cook's famous result is that the classes P and NP are equal if and only if the language SAT of all satisfiable Boolean formulas is in P. This can also be phrased by saying that SAT is NP-complete.

As we have noted, BBR treat the FLR and the URP for various classes of grammars. They show specifically that the URP for LFG is NP-hard; that the URP for unordered context-free grammars is NP-complete, and that the URP for GPSG is hard for the class $EXP-POLY$ of all languages recognizable in deterministic time $2^{p(n)}$, where p is a polynomial. This last result implies that the URP for GPSG is actually outside the class P, a result that is only conjectured to be true for NP-complete problems like SAT. They also show that the problem of GPSG category membership is polynomial-space complete, and that the general KIMMO recognition problem is NP-complete.

BBR provide a carefully motivated account of their techniques. The introductory chapters are a valuable source of information about complexity theory, and should be accessible to most formally inclined readers. In particular, they illustrate their general reduction strategy with the simple class of agreement grammars, an artificially chosen class which nevertheless shows how actual linguistic phenomena can combine to force computational intractability (assuming of course that P is not equal to NP). This example provides an extremely clear picture of the way general reduction techniques work, and those wishing to see how complexity techniques could be applied in their own work should definitely read this section.

These techniques are further developed in the remaining chapters; a typical application is the result that the universal recognition problem for LFG is NP-hard. BBR consider an extremely simple subclass of LFG grammars; the context-free rules are the same for every grammar in the subclass, and the only variation is in the lexicon and in the number of features that must be unified at each level of parsing. It is clear that any particular grammar in this subclass is weakly equivalent to a context-free grammar, so that any one fixed-language recognition problem would be solvable in polynomial time (in fact, in time proportional to n^3, where n is the length of the terminal string.) How, then, is it possible that LFG recognition is intractable? The answer lies in the fact that exponentially much information can be encoded in the feature-checking machinery of LFG. That is, n two-valued features can encode 2^n bits of information. BBR take advantage of this in reducing the satisfiability problem to the URP. Suppose that an instance of the satisfiability problem involves the propositional variables $p_1, \ldots, p_n$. There are 2^n different assignments of 0 or 1 to these variables. The essential trick in the reduction is to create a 0-1-valued feature for *each* of these variables in a grammar corresponding to the instance of SAT. Because the grammar is a parameter to the URP, the transformation of instances of SAT can affect the G component in the ordered pair $\langle G,w\rangle$, which is the typical instance of the URP. Then, the functional coherence principle of LFG can be used to guarantee that the same feature values, simultaneously for all n features (for which there are 2^n possibilities), occur at all nodes of the tree. It would need to have exponentially many nonterminal symbols in order to keep track of the same information.

For each problem proved intractable, BBR discuss the sources of intractability. In the case of LFG, for example, the intractability comes from allowing arbitrarily many different features in LFG grammars. In some cases, I feel that BBR do not show that the worst

cases of the URP will not arise in practice, and the LFG result is a case in point. The substantive principles of LFG will guarantee that there are only a fixed number of features to be passed up the c-structure tree by unification. These will be features, like agreement, chosen from a finite list. Thus the reduction will be blocked, at least in this case. For another problem, that of KIMMO recognition, a similar remark applies. In the reduction from satisfiability, propositional variables are encoded as lexical characters. The intractability of the KIMMO recognition problem thus crucially depends on there being arbitrarily many characters in an instance of the problem. If, however, this number of characters is fixed for all instances of the problem, then it would seem that the finite-state machinery of KIMMO recognition would yield polynomial-time processing, although, admittedly, a very large polynomial.

In general, it is not clear that the universal recognition problem is the problem that captures the notion of parsing complexity. BBR claim that we must consider the extra parameter of grammar size as a variable in parsing complexity, because, for instance, human grammars presumably change over time, as in language learning. However, the kind of free change allowed by the URP may not be the type of change that actually takes place in learning. Even in computational systems, changes to the lexicon and the addition of new rules do not always force recompilation of the grammar. And, in fact, it seems clear that other parameters (the discourse situation and semantic principles) should also be considered in complexity analyses; these parameters may actually significantly *reduce* parsing complexity. For these reasons, it is not wise to quote the complexity results proved here as evidence of the validity or non-validity of the linguistic theories in question.

I would have liked to have seen Government-Binding theory treated more fully in the book; one wonders if the same intractability problems in the other theories obtain in GB. This would, of course, require a precise statement of the URP for GB, but such a statement would be valuable information in itself. Since GB is predicated on substantive principles, the kinds of reductions allowed in LFG might be disallowed in GB, leading to efficient parsability. In fact, it would also have been a service to have considered upper bounds on complexity a little more fully. Is LFG recognition possible in *NP*?

Objections aside, I feel that BBR have made a significant contribution to the mathematical study of language in this book. It answers certain questions with sound technique, and, more importantly, it raises many others.

William Rounds
Center for the Study of Language and Information
Stanford University
Stanford, CA 94305

Communication Failure in Dialogue and Discourse: Detection and Repair Processes

Reilly, Ronan G (editor)
[Educational Research Centre, St Patrick's College, Dublin]

Amsterdam: North-Holland, 1987, xi+404 pp.
Hardbound, ISBN 0-444-70112-5, $59 / Dfl 175.00

It's early morning and facing you on your work desk is a bright new copy of *Communication Failure in Dialogue and Discourse*, edited by Ronan Reilly. You, the trusty book critic for *Computational Linguistics*, open the volume and proceed to examine the material to report on its value for the audience of computational linguistics researchers.

Inside you find a collection of papers, drawn from various sources, all on the topic of discourse and communication breakdown within discourse. The papers are organized into chapters, with headings. But there is no real attempt to integrate the material — there are no introductory sections to chapters, no comments on papers from other contributors. The road ahead is rough — you, the reader, must navigate through the book, finding the most interesting parts. Your sense of unity is also disrupted by the fact that the papers are both from computational linguistics and from disciplines outside computational linguistics: (psychology, sociology, etc.), and by the fact that most of the computational papers are North American, while most of those outside of computational linguistics are from the United Kingdom.

Do you throw your hands up, entering a brief, content-free 250-word review? Of course not. You find some worth in the book, through a careful reading of every single contribution. The papers outside of computational linguistics do have contributions — examples that suggest new input to process in a computational model, psychological evidence that suggests new processing strategies to try for these computational models.

A random sample of these worthy bits include:

(i) a number of new examples from Reilly, who also manages to reference almost all the computational linguistics researchers in the volume;

(ii) a study by McTear on developmental processing of communication failures;

(iii) a synopsis of processing observed in speech recognition of lexical failure by Harris;

(iv) a proposal by Anderson and Garrod that people use the same types of referring expressions through a session;

(v) evidence by Cahill and Mitchell that people do draw plan inferences while processing discourse.

But your careful reading also uncovers some frustration. There are examples of researchers simply unaware of the current efforts in discourse in computational linguistics (e.g. Egan speaks out against the plan-based approach, because it does not consider communicative

goals; in fact, both types of goals are handled by Litman (1985)). Then, there are numerous efforts to achieve the same goal (e.g. several taxonomies of errors), which make the results difficult to practically apply.

Scattered among the lesser known papers of the volume are contributions from familiar researchers in computational linguistics. Some of these papers have already appeared in *Computational Linguistics*. But, you convince yourself that it is useful to have several papers on the same subject together in one volume. Then, you find really useful deeper summaries of some work, to date available only as full Ph.D. theses or small conference papers (e.g. Carberry, Pollack, McCoy) (though the use of these contributions may soon be superceded by journal papers).

Donning your academic's hat, you discover several useful summaries of literature, within papers — Gardiner and Christie's survey of man-machine interfaces, Torrode's elaboration of Labov's work, Ferrari and Prodonof's summary of Allen's work, and Sharkey and Sharkey's review of connectionism. But you remind yourself that the book is still not appropriate as a course text — the material is too dispersed; the gems are hard to find.

At last, you sit back and conclude that there is some worth to the volume, for the undaunted reader. You particularly enjoy the fresh material from computational linguistics people outside of North America (e.g. Ferrari and Prodonof).

But your last impression is of the flaws in the book. Not all the papers seem relevant (e.g. Cater on metaphor). Still others seem to drift to a new topic, unannounced (e.g. Gardiner and Christie, Sheehy, which both discuss dialogue with gestures). And there are those irksome typos:

(i) on page 3, the first real page of the book, there is an indication of a footnote, with no footnote attached;

(ii) in the paper by Anderson and Garrod, Garrod's name is misspelled in a reference to a previous paper by the authors;

(iii) there is a reference in one of the papers to an interesting technical report, "A computational model for the analysis of arguments", by an author referred to as P.R. Cohen, the initials used by Phil — a rather strange mistake, at least to some people . . .

Robin Cohen
Department of Computer Science
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1

Reference

Litman, Diane 1985 Plan Recognition in Discourse Analysis: An Integrated Approach for Understanding Dialogues. TR 170, University of Rochester, Department of Computer Science

Readings in Natural Language Processing

Grosz, Barbara J; Sparck Jones, Karen and Webber, Bonnie Lynn (editors)

[Harvard University, University of Cambridge, and University of Pennsylvania, resp.]

Los Altos, CA: Morgan Kaufmann Publishers, 1986, xv+664 pp.

Paperbound, ISBN 0-934613-11-7, $26.95

This collection of 38 research papers is an extremely valuable resource for researchers, students, and teachers in the field of natural language processing (NLP). Its 664 pages provide extraordinary breadth and will be useful to old hands as well as newcomers. Although the readings span the time period of 1961 to 1985, only 8 of the 38 papers appeared before 1977, 19 were published from 1977 to 1981, and 11 from 1982 to 1985. The readings include 18 journal papers from 7 different journals, 9 conference proceedings papers from 5 different conferences, and 10 papers drawn from 9 other research collections.

The collection begins with an introduction, including a theoretical and historical overview of the field of NLP, and a discussion of the issues addressed in the six chapters that follow. The authors note that the chapter headings are broad categories "and should not be taken to imply either that we are adopting a particular position about the way processing . . . should be done, or that problems and solutions assigned to one category have no relevance elsewhere." Each of the six chapters also begins with an introduction describing the historical background and computational issues that gave rise to the papers in the chapter. These introductory sections, while short (3 to 5 pages), are specific and detailed enough to provide a context for the reader to appreciate the papers. They also include substantial bibliographies of important related work.

Chapter I: Syntactic models. Five different grammatical models are presented in this chapter (context-free grammar, augmented transition networks, Marcus's deterministic parser, definite clause grammar, and functional unification grammar). A discussion by Perrault on the generative power and computational complexity of grammatical formalisms, a description by Jane Robinson of a broad-coverage English grammar, and a 1962 paper by Kuno and Oettinger describing their "predictive analyzer" complete the section, which alone is worth the price of the book.

Chapter II: Semantic interpretation. This chapter is a diverse collection of nine papers about meaning representation and the process of translating natural language into a representation of meaning. The contributions include Schank on conceptual dependency and MOPs; Wilks on a machine translation system using preference semantics; Hendrix on the translation of English sentences into semantic networks; and Schubert and Pelletier describing an approach to semantic translation based on predicate logic. This chapter also includes two

well-known papers that could just as well have been placed in Chapter 6: Woods on the semantic component of the LUNAR question-answering system, and Winograd on the simulated blocks-world robot, SHRDLU.

Chapter III: Discourse interpretation. This chapter begins with a 1973 paper by Charniak discussing the need for knowledge about the events of everyday living and the ordinary motivations of people, in understanding children's stories. Following this are four papers (by Hobbs, Grosz, Sidner, and Webber) that describe computational models for interpreting pronouns and definite noun phrases, based on formal representations of discourse entities and discourse focus.

Chapter IV: Language action and intention. This chapter focuses on models of language as purposeful action. A short paper by Bruce motivates this work by showing how language is used to accomplish goals of requesting, informing, etc. Two papers follow (by Philip Cohen and Perrault, James Allen and Perrault) that develop a formal representation of speech act planning and show how it can be used to model generation and interpretation of utterances. The last paper (by Wilensky) describes the use of knowledge about plans and goals in understanding stories.

Chapter V: Generation. The three papers in this chapter (by McKeown, Appelt, and McDonald) are very recent contributions, the first two directed toward planning what information to communicate in an utterance, and the third describing a technique for realizing the chosen information as a grammatical text string.

Chapter VI: Systems. The collection concludes with eight papers describing systems for understanding natural language. It includes papers by Burton and Brown on the use of semantic grammar in the SOPHIE computer-aided instruction system; by Cullingford on SAM (the best paper I have read on script-based NLP); by Hendrix et al on the LADDER question-answering system; and a paper by Parkison, Colby and Faught on PARRY, a program that simulated paranoid thought processes. Taken together with the Woods and Winograd papers in Chapter 2, the collection provides the reader with a detailed picture of the experimental side of NLP research.

The most striking characteristic of the papers in this collection is their uniformly high quality of exposition. Each one is important, interesting, and readable. Readability is achieved, not by sacrificing technical detail and presenting a vague summary, but by illustrating the technical points with well-chosen examples. Thus, the papers in this collection are a pleasure to teach as well as a pleasure to read. *Readings in Natural Language Processing* represents an exercise of good literary judgement as well as good scholarship.

Carole D. Hafner
College of Computer Science
Northeastern University
Boston, MA 02115

Structured meanings: The semantics of propositional attitudes

Cresswell, M. J.

Cambridge, MA: Bradford Books / The MIT Press, 1985, x+202 pp.
ISBN 0-262-03108-6; $19.95

I have not made a scientific survey of the subject, but my hunch is that no philosophical logician has written more about the problem of the semantics of propositional attitudes than has M. J. Cresswell. Professor Cresswell (Professor of Philosophy at Victoria University, Wellington, New Zealand) has been chasing doggedly after this particular dragon for a good decade (and more); during that time he has canvassed and experimented with a variety of approaches and solutions (Cresswell 1972, 1975, 1980, 1982). Cresswell now thinks he has finally vanquished the dragon — or, at least, has it lying at his feet. I fear, however, that he has not succeeded in slaying the beast, and that it probably can't be slain with the weapons, and in accord with the rules of warfare, he adopts.

The book is divided into four parts. Part I, "Sense and Reference", lays out the problem and general criteria for the acceptability of solutions, and introduces the crux of the solution Cresswell proposes. Part II, "What Meanings Are", first argues against viewing meanings as linguistic entities; this is followed by an informal introduction of the technical framework within which the proposed solution is to fit. Part III, "Formal Semantics", presents this framework more formally and explicitly, shows how the proposed solution fits neatly within it, and furnishes additional details. Part IV, although not so labeled, is a bibliographical commentary; this consists of extensive notes on the literature, including presentations of, and arguments against, rival views.

I am already guilty of a serious, but useful, misrepresentation. The foregoing summary gives the impression that there is a single problem in the semantics of the propositional attitudes and that Cresswell addresses himself solely to it. Alas, there are a number of independent problems about the semantics of propositional attitudes and the book contains interesting and illuminating discussions of many of them. This review, however, will focus on the one that Cresswell himself considers central. The problem is that of necessary or, more narrowly, *logical* equivalence. As for other problems discussed by Cresswell, I shall ruthlessly ignore them all.

The Problem of Logical Equivalence

What is the problem of logical equivalence? Let us look at an example.

(a) Socrates is mortal.

(b) Either Socrates is mortal and Bruce Sprinsgsteen is The Boss or Socrates is mortal and it is not the case that Bruce Springsteen is The Boss.

These two sentences are necessarily equivalent, that is, if one of them is true, the other must also be true. There are no possible circumstances in which they can differ in truth value. Moreover, this equivalence is a matter of logic. So, if someone believes one, he must also believe the other, that is, the following two sentences must be true [false] together:

(a') Max believes that Socrates is mortal.

(b') Max believes that either Socrates is mortal and Bruce Springsteen is The Boss or Socrates is mortal and it is not the case that Bruce Springsteen is The Boss.

The problem, of course, is that it seems all too easy to imagine that *(a')* and *(b')* differ in truth value. It seems all too likely that someone could believe *(a)* without believing *(b)*. That is the problem of logical equivalence.

I assume it is clear that the above argument involves a massive non-sequitur. What further principles or assumptions must be added to the fact of the logical equivalence of *(a)* and *(b)* to close the gap(s) in the argument? The following are the assumptions that Cresswell—along with many others—seems to make:

- The meaning of a[n indicative] sentence Φ is that condition or set of conditions $Prop_{\Phi}$ under which Φ is true. Call such conditions the *truth conditions* of the sentence or *the proposition* expressed by the sentence.[1]
- Sentences with the same truth conditions — sentences which express the same proposition — have the same meaning.
- Logically equivalent sentences have the same truth conditions.
- The meaning of a complex expression, such as a sentence, is determined by the meanings of its parts and the way they are combined. (This is a rather vague and indeterminate expression of *compositionality*.)
- Sentences of the form α *believes that* Φ involve a transitive verb standing for a relation between a subject (a person) and the proposition expressed by the embedded sentence, that is, between a subject and $Prop_{\Phi}$.
- Thus, the embedded sentence Φ in sentences of the form α *believes that* Φ is a meaningful part of the total sentence and its meaning is its truth conditions, $Prop_{\Phi}$. That is, the meaning of the sentence when embedded in such constructions is identical to its meaning when it is not thus embedded.

We start from the premise that *(a)* and *(b)* are logically equivalent and infer that they have identical truth conditions and thus are identical in meaning. We further assume that that meaning is a part of the meaning of both *(a')* and *(b')*; it is the object of the relation denoted by the verb. Moreover the other parts of the two are identical, so the two sentences must be identical in meaning, hence must be identical in truth value, as well. Again, the problem is that *(a')* and *(b')* don't seem identical in meaning, indeed—to repeat—it is all too easy to imagine specific people for whom they differ in truth value.

The idea that the meaning of a sentence is the set of conditions in which it is true can take many forms. It will come as no surprise to those who have read other installments of the saga of *Cresswell v. The Propositional Attitudes* that the form of this idea that Cresswell adopts is a possible-worlds-based, model-theoretic semantics for λ-categorial languages.[2] Within this framework, the proposition expressed by a sentence is identified with the set of possible worlds in which the sentence is true. The notion of two sentences having the same meaning is identified with their being true or false in the same possible worlds. Thus, within this framework:

> The problem of the propositional attitudes arises in the following way. If the meaning of a sentence is just the set of worlds in which the sentence is true, then any two sentences that are true in exactly the same worlds must have the same meaning, or in other words must express the same proposition. Therefore, if a person takes any attitude (for instance, belief) to the proposition expressed by one of those sentences, then that person must take the same attitude to the proposition expressed by the other.
>
> Yet it seems easy to have sentences about believing, and about other attitudes, in which replacement of sentences that are true in exactly the same worlds turns a truth into a falsehood. (page 4)

So much for the problem. On to the proposed solution.

The Solution

The crux of the proposed solution is disarmingly simple. Let's look at a seemingly unrelated story: the relation between a certain complex numerical expression and the application of a monadic number-theoretic function, the factorial function, to a number, 3:

(3)!

In the complex expression displayed, we interpret the syntactic operation of enclosing a numeral, or, more generally, a number term, in parentheses and concatenating it with the '!' as correlated with the operation of applying the factorial function to the number denoted by the numeral (number term). Thus, the expression displayed is a complex term denoting the number 6. That's [functionally] compositional semantics in action! But so, too, would be the following: we correlate with the given syntactic operation the operation of pairing the function denoted by the function expression with the number denoted by the number term. This yields as value the ordered pair:

[1] I will ignore the phenomena of context-relativity and pretend that the sentences we are interested in are eternal sentences, that is, express the same proposition on all occasions of use.

[2] I shall have nothing more to say about the syntax of either English or λ-categorial languages. Cresswell provides a nice introduction to the latter in Chapter 11.

$$\langle \lambda x.factorial(x), 3 \rangle$$

This, too, is compositional semantics in action.[3] Note that

$$(3)! \neq \langle \lambda x.factorial(x), 3 \rangle.$$

What, you may ask, has this to do with the problem of propositional attitudes? Let us imagine that

(a') Max believes *that* Socrates is mortal.

but that it is not the case that

(b') Max believes that either Socrates is mortal and Bruce Springsteen is The Boss or Socrates is mortal and it is not the case that Bruce Springsteen is The Boss.

Given that the two embedded sentences above are logically equivalent — and hence have the same meaning — how could this be?[4]

Cresswell's simple solution: the operation of forming a complex noun phrase by concatenating *that* with a sentence is ambiguous. The crux of the solution is to associate a[n infinite] number of different semantic operations with the one syntactic operation. But this version of the solution would violate the injunction: **no semantic ambiguity without structural ambiguity.** Cresswell, instead, locates the ambiguity in the complementizer; the latter, in "deep structure," at any rate, is always there to serve the function; the complementizer *that* is infinitely ambiguous in English.

Sentence *(b')* is many ways ambiguous, its meaning depending on that of *that*. I shall use explicit grouping devices to illustrate:

(b'$_0$) Max believes $that_0$ (either Socrates is mortal and Bruce Springsteen is The Boss or Socrates is mortal and it is not the case that Bruce Springsteen is The Boss.)

This is the *that* that causes all the trouble. Syntactically, it converts a sentence into a name; semantically, it denotes the identity function on propositions, i.e., sets of possible worlds. So, since the set of possible worlds in which either Socrates is mortal and Bruce Springsteen is The Boss, or Socrates is mortal and it is not the case that Bruce Springsteen is The Boss, is the same as the set of possible worlds in which Socrates is mortal, *(b'$_0$)* really does mean the same as *(a')*.

A second reading is the following:[5]

(b'$_1$) Max believes $that_1$(ω(Socrates is mortal and Bruce Springsteen is The Boss), ω_{or}, ω(Socrates is mortal and it is not the case that Bruce Springsteen is The Boss))

Here we have a *that*, $that_1$, that is a symbol for a ternary function that takes as arguments a proposition, a binary function on propositions, and another proposition and yields the ordered triple whose first element is the binary function and whose next two are the two propositions:

$\langle \omega_{or}$, ω(Socrates is mortal and Bruce Springsteen is The Boss), ω(Socrates is mortal and it is not the case that Bruce Springsteen is The Boss)$\rangle$

One more case should (more than) suffice:

(b'$_2$) Max believes $that_2$ ((ω(Socrates is mortal), ω_{and}, ω(Bruce Springsteen is The Boss)), ω_{or}, ω(Socrates is mortal and it is not the case that Bruce Springsteen is The Boss)).

$that_2$ is a symbol for a ternary function that takes the following as arguments. First, an ordered triple whose first element is a binary function from pairs of propositions to propositions and whose next two elements are the arguments to that function; second, a binary function from pairs of propositions to propositions; third, a proposition. The triple named by the $that_2$ clause is the following:[6]

ω_{or}, $\langle \omega_{and}$, ω(Socrates is mortal), ω(Bruce Springsteen is The Boss), ω(Socrates is mortal and it is not the case that Bruce Springsteen is The Boss)$\rangle$

One could, of course, go on; but the reader has no doubt had quite enough to see what Cresswell's solution amounts to. Propositional attitude constructions are ambiguous; their ambiguity is localized in the complement construction. Indeed, we might as well say that it is located in the complementizer *that*, always present, if only in some underlying deep structure.[7] Different *that*s, applied to one and the same sentence, yield names of different *n*-tuples; these tuples consist of an *n*-ary function and its arguments, some of which might

[3] So, too, would be the association of that syntactic operation with pairing the same two items in the reverse order—number first, function second. What would not be functionally compositional semantics in action would be an account in which this one syntactic operation was associated with more than one semantic operation. We shall see that Cresswell's proposed solution indeed satisfies this constraint.

[4] Note that we are also assuming that in *(b)* the syntactic operations of forming *(i)* conjunctions of sentences (using the word *and* and sticking it between sentences), *(ii)* disjunctions (ditto, but with *or*), and *(iii)* negations (in logician's English, placing *it is not the case that* in front of a sentence) are associated with the operations of forming unions, intersections, and complements of sets from an underlying domain of possible worlds.

[5] ω is a symbol for the intension of the associated expression. If the latter is short, we shall subscript it. This notation is borrowed from Cresswell. I should also note that the system for indexing our different *that*'s is not Cresswell's, and is purely ad hoc. In his Chapter 11, Cresswell makes all this precise and systematic.

[6] Note that, unless *or* is to be infinitely ambiguous as to type, we will have a mismatch of types here. ω_{or} is a function from pairs of propositions to propositions, but the second element in this triple is not of the right type. Only sets of possible worlds, i.e., the intensions of sentences, are propositions. But we must keep in mind that all that the various *that*s, other than $that_0$, do is to form *n*-tuples; there is no requirement that the *tail* of such an *n*-tuple constitute an appropriate sequence of arguments for the first element of the tuple (which is always a function). The requirement is more complicated and is in terms of recursion on sublists — that is, application of the first element of a sublist to the other elements of that sublist.

[7] Actually Cresswell is suitably guarded about the relation between the syntax of the formal λ-categorial languages and that of English. As he himself notes, he used to be less guarded.

themselves be represented by tuples. All the tuples deriving from application of a *that* to a given sentence Φ will meet the following condition: the first element of the tuple, when applied to its arguments, yields $Prop_{\Phi}$—the set of possible worlds in which Φ is true.

The phenomenon of our reluctance to replace equivalents with equivalents — actually identicals with identicals — is illusory: the seeming equivalents [identicals] are not really equivalent. Inferences, like that from the logical equivalence of *(a)* and *(b)*, together with the truth of *(a')* to the truth of *(b')* run the danger of the fallacy of equivocation.

All this is set out in characteristically clear and crisp fashion. Moreover, Cresswell offers a fairly *resonant* system of nomenclature for the proposed solution. The *referent* of a sentence is its *intension*, that is, its referent is the set of possible worlds in which the sentence is true. The *sense* of an embedded sentence, governed by any *that* except $that_0$, is an *n*-tuple whose structure is tied to the type-index of the *that*. Such structures are Cresswell's *structured meanings*. So we have *sense* and *reference*, and, sure enough, Cresswell sees his theory as a successor to Frege's.

Comments on the Solution

I want first to contrast Cresswell's proposed solution to David Lewis's treatment of the problem of necessary equivalence (Lewis 1972).

Lewis's solution involves identifying *senses* with "semantically interpreted phrase markers minus their terminal nodes: finite ordered trees [which have] at each node a category and an appropriate intension."[8] So, as in Cresswell, *senses* are to be contrasted with *intensions*. In terms of this construction, Lewis is able to (re)define the basic semantic properties and relations, including, centrally, that of a *sense of a sentence being true or false (at an index i)*. This definition is a trivial reworking of the definition of truth and falsity (at *i*) for sentences. Thus, one selects the second of the ordered pair of the category and intension—that pair being the sense of a sentence. Cresswell can be seen as proposing roughly the same solution. With each [fully disambiguated] propositional attitude sentence is associated both an intension, a set of possible worlds, and a sense, a structured meaning. If you're interested in truth, select the first of these.

Why doesn't Cresswell simply accept Lewis's account? The main reason is that Cresswell has explicitly committed himself to working within a particular semantic framework. The semantic algebras for the languages to be treated are to be generated from collections of (possible) individuals, possible worlds, and times. *Particular* applications might require that the sentences of the object language, or all its expressions, together with [arbitrary] objects representing the various syntactic categories, be included in the collection of individuals. However, this cannot be a general requirement. Hence, the identification of senses (structured meanings) with semantically interpreted phrase markers requires going beyond the bounds of the semantic algebras set-theoretically generated out of arbitrary collections of individuals, worlds, and times. Lewis's solution involves reference to linguistic items. Thus it departs from the straight and narrow path of model-theoretic accounts based on possible worlds.[9]

The crucial points about Cresswell's solution can be put in terms of the essential characteristics of Cresswell's structured meanings:

1. There are many such *n*-tuples, and they are trivially distinguished from one another, as well as from sets of possible worlds.

2. Though the *n*-tuples correspond to Lewis's phrase markers, or labeled analysis trees, they are neither themselves linguistic entities, nor composed out of linguistic entities. They are, in fact, set-theoretic constructs derived from perfectly arbitrary collections of individuals, worlds, times; thus their existence is guaranteed by the specification of the semantic algebra for the semantic theory.

Cresswell's proposed solution, in its *letter*, does not stray from the straight and narrow. In spirit, however, it surely seems to. The solution has the feel of a "linguistic" account with a guilty conscience.[10] Faced with a problem for which his chosen framework seems inadequate, Cresswell takes a proposed solution, Lewis's, which goes beyond that framework and transmogrifies it into one that doesn't. Fair enough, but why should anyone not committed to the program of reducing all intensionality to set-theoretic constructions out of possible worlds and possible individuals care?

Note that Cresswell does not associate structured meanings with sentences in isolation; that is, it is only when embedded in propositional attitude constructions that sentences have senses.[11] Sentence *(b)* is ambiguous, and (on most of its readings) the embedded sentence is associated with a structured meaning; *(b)*, on the other hand, is not so associated. It simply has an intension, the same intension as *(a)*. But why not treat *(b)* as associated with a structure? Indeed, why not treat *(a)* as also associated with a structure—a different one from that associated with *(b)*—say, the ordered pair whose elements are the intension of "Socrates" and the

[8] Actually, Lewis speaks of *meanings*. I have changed terminology to conform with Cresswell's.

[9] So, too, does a quite different, nonlinguistic, account due to Thomason (1980), which, either instead of or in addition to a set of possible worlds and the associated intension functions, introduces a domain of *propositions*.

[10] I should note that there are propositional attitude constructions, accounts of which surely will involve reference to linguistic items. Cresswell discusses these in two (really three) chapters on indirect discourse and in a chapter on "Attitudes *De Expressione*".

[11] I'm assuming that we can ignore the trivial option of identifying the sense of a sentence with the one-tuple whose sole element is the intension of the sentence.

intension of "is mortal"? The former is Socrates himself; the latter is the property of being mortal. Moreover, one need not take this latter to be a function from possible worlds to possible individuals, or from possible individuals to sets of possible worlds. Surprisingly, one can take properties and relations to be just that, properties and relations. Having rejected the identification of propositions with sets of truth conditions, and of the latter with sets of worlds in which sentences are true, one might as well go back to basics and start with whatever variety of intensional objects one needs.

The suggestion, then, is to posit structured propositions and to identify the proposition expressed by a sentence with such a structured entity, *not* with the set of conditions in which the sentence would be true.[12] This suggestion leaves open the possibility that logically equivalent sentences can still differ in what proposition they express. In fact, this suggestion has the two essential features mentioned above: on any of a number of accounts, the structured propositions associated with *(a)* and *(b)* (even in isolation) will be distinct and each will be distinct from the set of conditions (possible worlds) in which the sentences are true.[13] Moreover, these structured propositions are nonlinguistic and again, on any of a number of different accounts, their existence will be guaranteed by the fixing of various parameters of the semantic set-up.

The Autonomy of Semantics

At this point, let me remind the reader of an opinion I expressed at the beginning. I said that I did not think that an adequate solution to the problem of logical equivalence could be devised that conforms to the criteria of adequacy adopted by Cresswell. What my claim, in effect, amounts to is that no adequate solution is possible within the framework of standard model-theoretic accounts based on possible worlds, a framework within which all intensionality is reduced to set-theoretic constructions out of possible worlds, possible individuals, and times. Obviously, I cannot *prove* this; but surely, we are by now justified in drawing such a conclusion from past, unhappy experience.

I do not, however, think that merely enriching the semantic algebras available to the theorist is sufficient. Other measures are called for. In various places, Cresswell has argued that:

> *If* it were not for the problem of propositional attitudes, semantics could be seen as an autonomous discipline not reducible to psychology or any other cognitive science.[14]

But I cannot see why any such *reduction* to psychology is threatened by the realization that an adequate *semantic* account of propositional attitude constructions requires one to ponder seriously the roles played in our lives by propositional attitude reports — that is, to think some about psychology and other cognitive sciences.

Return to sentences *(a')* and *(b')* and notice that Cresswell's solution does not, by itself, yield an answer to the question of why the speaker is not simply replacing equivalents with equivalents when, after all, it seems that he is. Moreover, it does not tell us which *that* it is that occurs in, e.g., *(b')*. It is clear that all these *that*s are around to capture some important dimension of difference among the uses to which a given propositional attitude construction can be put. The dimension might be couched as follows: lesser or greater degree of fidelity to the way in which the subject of the report conceives of the content of the report — the content, that is, of the embedded sentence. This way of putting things emphasizes the fact that the dimension in question is not a semantic one. But according to Cresswell, the ambiguity *is* a semantic one; it is located at the level of sentence types. This also explains why both the theory and Cresswell are largely silent on the questions posed at the beginning of this paragraph. Cresswell, after all, is doing semantics. Perhaps, we should think instead about the dimension of difference alluded to above — a difference among uses to which a [perhaps unambiguous] sentence might be put, a difference, that is, in the roles played in our lives by our uses of propositional attitude constructions.

A point made by Barwise and Perry (1983) is that, if one very simply and crudely divides this role in two, one can go a long way toward explaining a wide range of phenomena. Thus, we sometimes use propositional attitude reports — for example, belief reports — to communicate information about the world; at other times, we use them to explain or predict the behavior of the subject of the report. When using these reports in the first way, we are much less likely to balk at the replacements of equivalents with equivalents or identicals with identicals; when using them for the latter purpose, we are more reluctant to accept such replacements. If we add in recognition of the fact that our reports are not of *total* cognitive states of subjects, but only of components (parts) of such states, we can then go even further in explaining the phenomena. The net result of such reflections is to lessen the otherwise intolerable burden borne by any model-theoretic account of the attitudes that insists on isolation from all other parts of the cognitive sciences.

The moral here can be expressed as follows. The treacherous snake in the garden is the *title* — not the content, it should be emphasized — of the first of Montague's three papers on the semantics of natural languages: *English as a Formal Language*. Natural languages are not really formal languages; the latter,

[12] This suggestion is due, first, to Bertrand Russell (1956). It has, more recently, been taken up, in various forms, by Barwise and Perry (1985), Salmon (1986), and Soames (forthcoming).

[13] These latter may also be distinct, as they can be in the theory of Barwise and Perry (1983).

[14] p. 129. See also Cresswell 1982.

with the singular exception of programming and other computer languages, are studied, not used. Formal languages do not play the roles that natural languages do in our lives, although they can be used or misused as models of certain aspects of natural languages. It is quite likely that an adequate semantics of natural languages — in particular, an adequate semantics of propositional attitude reports — cannot be formulated in complete isolation from accounts of the creatures who use these languages and, in turn, are both the producers and the [primary] subjects of those reports. Surely this moral should not come as news to those interested in the design of *natural-language-using systems*.

David Israel
SRI International
Menlo Park, CA 94025
and
Center for the Study of Language and Information
Stanford University

Acknowledgement

This research was supported in part by a gift from the System Development Foundation.

References

Barwise, K. J. and J. Perry. 1983. *Situations and Attitudes*, Bradford Books / The MIT Press, Cambridge, MA.

Barwise, K. J. and J. Perry. 1985. Shifting Situations and Shaken Attitudes, *Linguistics and Philosophy*, 8: 105-161.

Cresswell, M. J. 1972. Intensional Logics and Logical Truth. *Journal of Philosophical Logic*, 1: 2-15.

Cresswell, M. J. 1975. Hyperintensional Logic, *Studia Logica*, 34: 25-38.

Cresswell, M. J. 1980. Quotational Theories of Propositional Attitudes, *Journal of Philosophical Logic*, 9: 17-40.

Cresswell, M. J. 1982. The Autonomy of Semantics, in *Processes, Beliefs, and Questions*, Peters and Saarinen, eds., Dordrecht, Reidel.

Lewis, D. K. 1972. General Semantics, in *Semantics of Natural Languages*, Davidson and Harman, eds., Dordrecht, Reidel.

Russell, B. 1956. The Philosophy of Logical Atomism, in *Logic and Knowledge*, R. C. Marsh, ed., George Allen and Unwin, London, 177-281.

Salmon, N. 1986. *Frege's Puzzle*, The MIT Press, Cambridge, MA.

Soames, S. Direct Reference, Propositional Attitudes, and Semantic Content, to appear in *Propositions and Attitudes*, N. Salmon and S. Soames, eds.,forthcoming.

Thomason, R. H. 1980. A Model Theory for Propositional Attitudes, *Linguistics and Philosophy*, 4: 47-70.