THE LEGACY OF ZELLIG HARRIS II

AMSTERDAM STUDIES IN THE THEORY AND HISTORY OF LINGUISTIC SCIENCE

Volume 229

Bruce E. Nevin and Stephen B. Johnson (eds.)

The Legacy of Zellig Harris
Language and information into the 21st century
Volume 2: Mathematics and computability of language

THE LEGACY OF ZELLIG HARRIS

LANGUAGE AND INFORMATION INTO THE 21ST CENTURY

VOLUME 2: MATHEMATICS AND COMPUTABILITY OF LANGUAGE

Edited by

BRUCE E. NEVIN
Cisco Systems, Inc.

STEPHEN B. JOHNSON
Columbia University

JOHN BENJAMINS PUBLISHING COMPANY
AMSTERDAM/PHILADELPHIA

The paper used in this publication meets the minimum requirements of American National Standard for Information Sciences — Permanence of Paper for Printed Library Materials, ANSI Z39.48-1984.

Volume 2

The Legacy of Zellig Harris: language and information into the 21st century / Edited by Bruce E. Nevin and Stephen B. Johnson.
p. cm. -- (Amsterdam studies in the theory and history of linguistic science. Series IV, Current issues in linguistic theory, ISSN 0304-0763 ; v. 229)
Includes bibliographical references and index.
Contents: v. 2 Mathematics and computability of language.
ISBN 90 272 4737 4 (Eur.) / 1 58811 247 0 (US) (Vol. 2)

Library of Congress Cataloging-in-Publication Data for volume 1

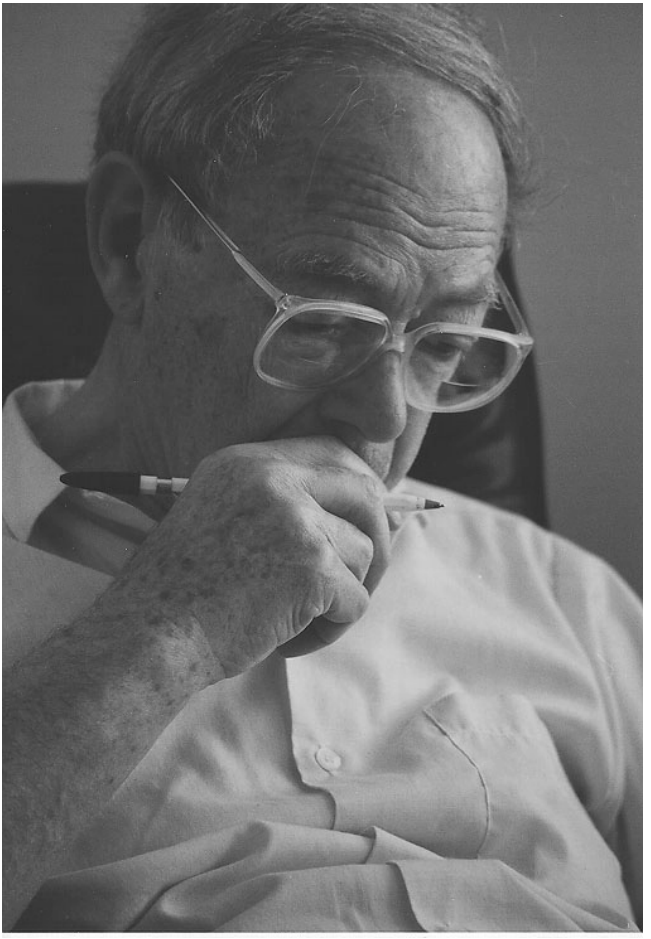
The Legacy of Zellig Harris: language and information into the 21st century / Edited by Bruce E. Nevin.
p. cm. -- (Amsterdam studies in the theory and history of linguistic science. Series IV, Current issues in linguistic theory, ISSN 0304-0763 ; v. 228)
Includes bibliographical references and index.
Contents: v. 1. Philosophy of science, syntax and semantics.
1. Harris, Zellig S. (Zellig Sabbettai), 1909- 2. Linguistics. I. Nevin, Bruce E. II. Series.
P85.H344 L44 2002
410--dc21 2002074704
ISBN 90 272 4736 6 (Eur.) / 1 58811 246 2 (US) (Vol. 1)

ISBN 90 272 4741 2 (Eur.) / 1 58811 316 7 (US) (SET VOLUME I+II)

John Benjamins Publishing Co. • P.O.Box 36224 • 1020 ME Amsterdam • The Netherlands
John Benjamins North America • P.O.Box 27519 • Philadelphia PA 19118-0519 • USA

Contents

Part 3 Computer applications



Professor Zellig Harris

Foreword

Stephen Johnson
Columbia University

The chapters in this volume reflect the impact of Zellig Harris on the study of language pertaining to formal systems, computability, and computer applications.

1. Development of the field

The effect of Harris's work in these fields stretches back almost 45 years, to 1957, when the first computer program to perform syntactic analysis of English sentences was developed on a UNIVAC computer. This parser, based on string theory, is described in Chapter 4 by Sager and Nhan, and in greater detail by Joshi in Chapter 5. Sager and Nhan also provide a detailed description of the Linguistic String Project, which began in 1965 and continues today. The LSP system inspired several others, including PROTEUS (Grishman & Hirschman 1986), PUNDIT (Hirschman et al.), KERNEL (Palmer et al.) and MedLEE (Friedman 1994), which continue to be active research projects.

As Harris's theories evolved over time, so did the range of computational devices inspired by them. These were naturally influenced by concurrent developments in mathematics and computer science. String grammar emerged around the time of finite state automata. The UNIVAC parsing program was based on a formalism that would today be called cascading finite state transducers (see Chapter 5). A different type of finite state automaton was able to determine sentence well-formedness using a cycling process of cancellations (Harris 1963).

With the expansion of work on formal languages in the 1960s, context-free grammars became central in natural language research, in large part because of their simplicity and tractability. However, these formalisms obscure some

properties of natural language that are directly available in string grammar. This led Sager to augment context-free rules with attributes and constraints written in a special-purpose restriction language. The resulting system also had the power to implement transformational grammar, since strings define the domains and ranges of transformations. Similar approaches are used in PROTEUS, PUNDIT, and KERNEL.

During this period, it was well established that individual word choices affect the acceptability of strings, and also restrict the ability of transformations to operate. The need to lexicalize context-free grammar, as well as a desire to combine the complementary strengths of context-free grammars and string grammars, led Joshi to develop tree-adjoining grammar, as described in Chapter 5. Maurice Gross investigated the lexicalization of string grammar itself, resulting in Lexicon Grammar (Gross 1984). In this approach, for each string a table is constructed in which the columns are the word types of the string and the rows are words of the language.

The phenomenon of sublanguage is one of Harris's major discoveries. A direct representation of sublanguage grammar is difficult, because of the large variation in surface forms. Early approaches pioneered by Sager dealt with the problem in stages: surface forms parsed by string grammar were transformed to elementary sentences, on which sublanguage constraints were applied. This led to systems in which syntax, semantics, and pragmatics were handled by separate 'modules', an architecture emulated by most other NLP systems. More recently, Friedman (1994) has pursued a different method, using a single grammar that parses sublanguage sentences directly, without the use of a general syntactic component.

The various formalisms for surface forms, transformations, sublanguage, and the lexicon can be related, but the resulting system is extremely complex. Harris devoted the last decades of his life to the creation of a unified theory of language, which culminated in operator grammar (1982, 1991). The theory is intrinsically lexical, driven by the argument requirements of each operator word. Surface forms are obtained through reductions, which are simpler than transformations and which have lexicalized domains. Sublanguage is recast using the dependency structures and reductions of the operator grammar framework. Operator grammar is discussed in several places in this volume: Pereira contrasts Harris's approach to 'mainstream' linguistics, Johnson presents the beginnings of a computational system based on this theory, Habert and Zweigenbaum explore automatic classification of words, and Smaby extends some of the methods to computer user interfaces.

2. Mathematical roots

The volume is introduced by André Lentin's essay, "Reflections on references to mathematics in the work of Zellig Harris". This brief piece articulates the mathematical philosophy that permeates Harris's work and informs subsequent work by others on formal systems, computability, and computer applications. Lentin describes the principal mathematical ingredients in the linguistic theory of Harris, and connects these to the giants of mathematics who influenced Harris's thinking (see also Harris's essay in Volume 1). Constructivism is seen in Harris's use of finite elements and explicitly stated operations. Type theory, linked to the work of Russell, appears in the typing of operators by the arguments to which they apply. Algebra was clearly the primary tool of choice for Harris, enabling definition and manipulation of symbolic structures. Finally, Lentin suggests that just as Gödel used metalanguage to find cracks in the foundations of mathematics, Harris showed the impossibility of an external foundation for language.

Lentin indicates that Harris's approach was to select from mathematics the appropriate tools for examining language. This philosophy leads to creation of new mathematical systems, rather than forcing language into the confines of an existing system. Moreover, there is no 'best' or 'true' system, but instead multiple ways of viewing the same mathematical object.

3. Mathematics and formal systems

The section on mathematics and formal systems provides a clear illustration of Lentin's observations. The three chapters explore three very different formal systems. Pereira reviews Harris's theory of operator grammar and relates it to learning theory. Oehrle focuses on formalization of the morphology of Semitic words. Langendoen looks at the mathematical properties of sequences.

Fernando Pereira's "Formal grammar and information theory: together again?" examines the great divide between information theory (in the Shannon tradition) and linguistics (in the Chomsky tradition). He asks whether Harris's theory can provide the bridge between these disciplines, and notes that Harris's principle of least grammar is not only a methodological constraint for the linguist, but also a crucial component of language acquisition. Harris discussed this point (1991: 404-405), but other than hints about the use of classifiers for analogic extension of selection domains he left open exactly

how language learners generalize from a limited sample. Recent work on learning theory shows that models can generalize from a finite sample, e.g. to assign probabilities to unseen events. This phenomenon is essential for the creative aspect of language use. Such models must represent the internal state of the learner, e.g. as 'hidden variables' that capture uncertainty about events observed so far.

These techniques provide a way of quantifying Harris's principle of grammatical simplicity. Pereira also points out that other aspects of Harris's theory are helpful in developing models of learning. For example, Harris's focus on lexicalized elements makes it easier to factor interactions among the hidden variables.

Pereira reminds us that language learning is grounded in sensory experience, which greatly constrains the learning process. He notes as well that the linguistic environment (e.g. neighboring sentences) provides an additional set of constraints on interpretation. This discourse restriction goes beyond the likelihood constraint in Harris's theory. Pereira suggests that 'language understanding' can be recast in this framework, by relating linguistic events to one another, e.g. to judge whether a sentence follows from a discourse, or to answer a question about the discourse.

In "Logics for intercalation", Richard Oehrle develops a mathematical approach to discontinuities. He examines the problem of intercalative morphology of Arabic, in which a pattern of vowels is inserted into a sequence of consonants (the lexical root). Oehrle presents a multi-modal categorial grammar that allows one to escape the simple concatenation operations of 'classical' categorial grammar. Categorial grammar is mentioned in Harris's work (1991: 60), and also in this volume in the chapters by Pereira and by Johnson.

The basic idea is to assign each vowel pattern to a pattern of modal operators, which controls how the vowels get 'shuffled' into the sequence of consonants. Oehrle creates discontinuous structures through a variety of distributivity postulates, which enable an element to distribute through another expression, just as multiplication distributes over addition in arithmetic: $x \star (y + z) = x \star y + x \star z$. The distributivity postulates are combined with a pattern of modal operators, resulting in a 'controlled shuffle' of one list of elements into another.

While the focus of the article is on morphology, Oehrle supplies many examples in English syntax. He shows how controlled discontinuity can be used to analyze structures such as the relative clause, establishing a relation between the relative pronoun and the 'missing' argument.

D. Terence Langendoen in "Sequence Structure" defines a language as a set of morpheme sequences. There is a similarity to Harris's methods in that the formalism deals only with relations between morphemes. For a given sequence of morphemes, a 'subsequence' consists of some of the morphemes in the same order. Because a subsequence does not have to be continuous, there are some rough parallels to the discontinuous structures examined in the preceding chapter by Oehrle.

Langendoen defines the 'sequence structure' of a language as a partial ordering induced by the subsequence relation. The 'conjunction' of two sequences is their greatest lower bound in the partial order. He uses this operation to obtain structural analyses about sequences. For example, a sequence is ambiguous if there is more than one way to obtain it by conjunction. He shows several examples of sequence structures applied to morphology and to syntax. In particular, he shows how his approach can account for both local and unbounded dependencies.

4. Computability of language

The central section, "Computability of Language", considers how to implement mathematical theories as computer programs. Sager and Joshi each describe formalisms based on Harris's earlier theories (strings, transformations, and sublanguage), while Johnson considers the later operator grammar. In "The computability of strings, transformations and sublanguage" Naomi Sager and Ngô Thanh Nhàn describe research of the Linguistic String Project (LSP). The original motivation for this research was to assess the computability of language structure. The success of the approach led to practical applications of natural-language processing for patient care and related activities.

The project began with the computerization of String Grammar. The LSP system was implemented using context-free grammar, represented in Backus-Naur Format. Sager notes that CFG resembles immediate constituent analysis, and obscures the underlying character of linguistic strings. This limitation was addressed by assigning rules of the grammar to string types. The key issue for computability lies in the fact that grammatical restrictions can be encapsulated within defined strings. (Joshi also addresses the property of local constraints in Chapter 5.) In terms of implementation, each restriction can be associated with a grammar rule, and the restrictions can be expressed using string types and operations on strings.

Harris represented the transformational decomposition of sentences as a semilattice of operations. Because the vast details of the transformations and the lexicon impose, as Sager observes, 'formidable' requirements for implementation, the LSP system performs transformations using the same operations that were employed to express restrictions.

Sublanguage was a major focus of LSP research. A chief use of sublanguage was to reduce syntactic ambiguities; the medical domain required several thousand patterns used in selection restrictions. Sublanguage classes were determined by manual analysis of patterns, and words were assigned to classes using 'diagnostic frames'. A discovery process was not automated because it relied on obtaining a syntactic parse and transformations, which rely on sublanguage constraints. The research identified interesting phenomena, such as unique syntactic usage ('sublanguage idioms'), and fragments, which could be described using familiar components of grammatical sentences. This approach yields a 'grammar of the ungrammatical'.

Aravind Joshi addresses similar issues in "Hierarchical structure and sentence description". He describes three formalisms based on Harris's theories: cascaded finite state automata (Uniparse), string grammar, and tree adjoining grammar. Joshi identifies their common methodology as the separation of a finite set of elementary units from the process of derivation that relates these units. The elementary units encapsulate the dependencies among word categories (and words), making all dependencies 'local'. The derivation process is recursive, and creates what Joshi terms the 'hierarchical structure' of sentences.

In the Uniparse system and string grammar, this hierarchy is implicit. Context-free grammar can describe the hierarchical structure, but cannot encapsulate dependencies because each rule describes only one level of the hierarchy. Moreover, context-free grammars cannot be lexicalized while preserving the strong generative capacity of a grammar. To address these shortcomings, tree adjoining grammar is presented as a formal description of sentences that can define multi-level dependence structures (elementary trees). Substitution and adjunction operations are applied recursively to elementary trees to generate derived trees.

Stephen Johnson considers the linguistic theory that followed the work on strings and transformations in "The computability of operator grammar". In place of strings, the theory has simple dependency structures in which operator words predicate on argument words. Instead of transformations, reductions map the predication structures into compact forms. Johnson observes that the complexity of the reduction system is a significant barrier to computerization,

a point echoed by the statement of Sager and Nhan that operator grammar analysis is "deeper than what we are in a position to compute today."

Johnson notes (along with Pereira) that Operator Grammar makes a radical departure from other theories by establishing semantics on a statistical model rather than through interpretation in a logical model. His paper proposes a synergy of several different grammatical formalisms into a new type of grammar: Categorial grammar provides a system of types; rewrite grammars formalize the process of sentence generation; lexicon grammar manages the vast body of details relating lexical items to syntactic forms; information theory enables description of predication as a mathematical distribution; and fuzzy logic provides a foundation for semantics based on these distributions. Johnson suggests that the resulting grammar can be implemented in a variety of ways, with the possibility of extremely efficient algorithms using finite state transducers.

5. Computer applications

The final section, "Computer applications", examines several different applications of the method and formalisms described in the previous sections: building dictionaries for corpus linguistics, automatic discovery of sublanguage categories, generation of sublanguage texts, and analysis of user interfaces.

"Distributional syntactic analysis and valency: Basic notions, procedures, and applications of the Pronominal Approach" by Karel van den Eynde, Piet Mertens, Sabine Kirchmeier-Andersen, and Lene Schøsler presents work on the development of dictionaries (readable by machines as well as by humans) and their application in corpus linguistics. Their methodology, called the Pronominal Approach, is related to Harris's substitution procedures, his concept of pro-forms, his string analysis, and his proposals regarding the use of classifier vocabulary to relate novel sentences to other sentences whose likelihood or acceptability (in a given subject-matter domain) is known. They discuss their methodology and its roots in the linguistics of Harris and of Togeby, and in the constructivism of Goodman (1951). Then they describe two dictionaries being developed along these lines, the PROTON dictionary of French verbs (Leuven, Belgium) and the Odense Valency Dictionary (OVD) of Danish verbs. They conclude with proposals for further refinements and applications for the methodology.

In "Contextual acquisition of information categories: what has been done and what can be done automatically?" Benoit Habert and Pierre Zweigenbaum examine the problem of determining a semantic classification of lexical items. Semantic classes are important for the scientific study of sublanguage, for developing natural language processing systems, and for performing a wide variety of practical applications in information retrieval. Habert and Zweigenbaum firmly ground the problem in Harris's theory of operator grammar: in general language, the selection of an operator for its argument is fuzzy, while in sublanguage selection is crisp. This theory offers the possibility of automatic classification in sublanguage using distributional methods.

Two major projects (Sager *et al.* 1987, Harris *et al.* 1989) developed classifications for several biomedical domains, but this work was largely manual. While classes could in theory be obtained solely on a distributional basis, in practice external resources (e.g. thesauri and expert opinion) were required to expedite the process. As discussed by Sager in this volume, automatic classification is extremely difficult due to the need for high-quality parsing and transformation of sentences to obtain underlying operator-argument pairs for analysis.

Habert and Zweigenbaum review the literature on machine learning and statistical approaches for semantic classification, and conclude that robust parsers that produce partial analyses are preferable to more complete parsers that frequently fail. In particular, they find that useful results can be obtained by examining a local 'window' of words around a given word. However, none of the current approaches succeeds in fully automating the method described by Hirschman *et al.* (1975). They suggest that a new paradigm is required to implement and evaluate distributional semantics.

Since most of the contributions in this volume are concerned with analysis of text, Richard Kittredge chose to focus on its synthesis in "Text generation within sublanguages". Kittredge observes that despite a great deal of research on sublanguages and their restricted grammars, few have been found amenable to generation. Success has been achieved with stereotypical technical reports, such as weather reports, summaries of financial market activity, and descriptions of sporting events. For each of these, there is an alternative representation of the information, such as numerical measurements. However, for scientific articles and even for instructional materials generation is difficult because there is no independent representation of the source knowledge. Information formats (described by Sager and Nhan) can be used as an interlingua to represent the information to be generated. This work raises

intriguing issues about the relation between information and language.

In contrast with text analysis, text generation must focus on whole texts instead of single sentences. Kittredge describes distributional analysis carried out at the sentence level, which assigns sentences to classes. Members of a class are informationally equivalent; each can be an answer to the same question. This approach is reminiscent of the discourse dependencies invoked by Pereira in the first section of this book. Recent work by Kittredge employs meaning-text theory (Mel'chuk 1987). Although Kittredge does not go into detail, this theory has some interesting similarities to Harris's operator grammar, such as the use of dependency representations and paraphrastic operations.

In the final chapter, "A distributional semantics applied to computer user interfaces", Richard Smaby goes beyond traditional natural-language applications. He examines the interaction between humans and computers using multiple modalities, such as graphical displays, pointing devices (e.g., the mouse), keyboard, speech input, and speech output. He shows that interaction with a graphical user interface of a software application such as a word processor can be represented as sequences of discrete 'events'. Collections of sequences form a corpus suitable for distributional analysis. The corpus is similar to that employed in studies of dialog, in mixing user events with application events (computer responses).

In Smaby's distributional analysis of the interaction data, he identifies the dependence of the occurrence of a sequence X on the occurrence of another sequence Y. For example, X may be a sequence of user operations that select a portion of text in a document, and Y a sequence of computer operations that applies some command to the selected text, such as making the font bold. Using distributional techniques, the structure X Y can be shown to have a linguistic interpretation, in which Y is an operator that predicates on the argument X.

Smaby finds that user interfaces have many linguistic properties, whether the modality is mouse clicks or speech and whether the lexical elements are icons or words. Interfaces have features reminiscent of pronouns, paraphrase (e.g., several ways to select a piece of text), and conjunction (e.g., multiple operations performed on one piece of text).

Smaby describes a user interface as a form of sublanguage that pares language down to the minimum needed to interact effectively. This approach is intriguing because the sublanguage must be learnable by the user through feedback from the system. Smaby concludes that a distributional approach can help with the design of effective user interfaces.

References

Friedman Carol, Phil Alderson, John Austin, James Cimino, & Stephen Johnson. 1994. "A general natural language text processor for clinical radiology". *Journal of the American Medical Informatics Association* 1(2): 161–174.

Grishman, Ralph & Lynette Hirschman. 1986. "PROTEUS and PUNDIT: Research in Text Understanding". *Computational Linguistics* 12 (2), 141–145.

Gross, Maurice. 1984. "Lexicon-Grammar and the Syntactic Analysis of French". *Proceedings of the 10th International Conference on Computational Linguistics* (COLING'84), Stanford, California.

Harris, Zellig S. 1963. "A Cycling Cancellation-Automaton for Sentence Well-Formedness". Transformations and Discourse Analysis Papers; 51. (Repr. as National Bureau of Standards #6320427; in *International Computation Centre Bulletin* (1966) 5.69–94; and in Zellig S. Harris, *Papers in Structural and Transformational Linguistics.* Dordrecht/ Holland: D. Reidel 1970, pp. 286–309.)

Harris, Zellig S., Michael Gottfried, Thomas Ryckman, Paul Mattick Jr, Ann Daladier, T.N. Harris, and S. Harris. 1989. *The Form of Information in Science: Analysis of an immunology sublanguage*, Dordrecht, The Netherlands: Kluwer Academic. Boston studies in the philosophy of science, 104.

Harris, Zellig S. 1991. *A Theory of Language and Information: A mathematical approach.* Oxford: Clarendon Press.

Hirschman, Lynette, Ralph Grishman & Naomi Sager. 1975. "Grammatically-based automatic word class formation". *Information Processing and Management* 11: 39-57.

Hirschman, Lynette, Martha Palmer, John Dowding, Deborah Dahl, Marcia Linebarger, Rebecca Passonneau, Francois-Michel Lang, Catherine Ball, & Carl Weir. 1989. "The PUNDIT NLP System". In: *AI Systems in Government Conference*, Computer Society of IEEE, 27–31 March, 1989.

Mel'chuk, Igor & Nikolai Pertsov. 1987. *Surface Syntax of English: A formal model within the meaning-text framework.* Philadelphia: John Benjamins.

Palmer, Martha, Rebecca Passonneau, Carl Weir, & Tim Finin. 1994. "The KERNEL text understanding system". In Pereira, Fernando C. N. & Barbara J. Grosz, eds., *Natural Language Processing*, MIT Press, Cambridge, Massachusetts.

Sager, Naomi, Carol Friedman & Margaret Lyman, eds. 1987. *Medical Language Processing: Computer management of narrative data.* Reading, Mass: Addison Wesley.

Shapiro, P. A. 1967. "ACORN — an automated coder of report narrative". *Methods of Information in Medicine* 6(4): 153–162.

Acknowledgements

The initiator of this project was Harris's friend and one-time student, William M. Evan, without whose encouragement and helpful suggestions at certain trying junctures it might have foundered. I am also most grateful to the other members of my advisory board, Henry Hiz, Henry Hoenigswald, and Hilary Putnam, for their counsel and assistance in many matters.

Stephen Johnson has played an extremely important role as an advisor and reviewer. Of signal importance were the suggestions, sometimes critical, always helpful, of many people, including especially Naomi Sager, Michael Gottfried, Tom Ryckman, Paul Mattick, and Jim Munz. My repeated efforts to persuade Noam Chomsky to contribute proved fruitless, but the process helped to sharpen the argument in portions of the Foreword. Anne Daladier provided the English original of Harris's essay, which introduces Volume 1, at a time when a copy had not been found here; and for this, and indeed for her role in encouraging Harris to write it in the first place, we all owe her a great debt of gratitude.

But the greatest acknowledgement and thanks must go to the contributors to these two volumes, and to others not directly represented here, who have carried forward the various lines of research initiated by Harris. If Zellig Harris were still with us, he would doubtless express his appreciation to them, and to his colleagues, especially those in the University of Pennsylvania Department of Linguistics, which he founded in 1946. He would probably also express his appreciation to generations of his gifted students, and acknowledge how discussions with them helped him to develop and clarify his ideas.

Finally, he would undoubtedly express appreciation to his wife, Bruria Kaufmann, for her many substantive contributions over the years, and to his brother, Tsvee Harris and his wife Susannah, whose knowledge of the field of immunology guided the research that resulted in *The Form of Information in Science*.

Reflections on references to mathematics in the work of Zellig Harris*

André Lentin
Université Paris V

Language bears, on the face of it, the promise of mathematical treatment.
— Z. Harris

The idea that language, by its very nature, requires the aid of mathematics for its study appeared as a leading idea very early in the work of Zellig Harris. As far back as 1946, in "From Morpheme to Utterance", by formalizing 'expansions' he introduced a hierarchical system of equations between linguistic categories, and he sketched the formulation of a grammar in terms of partially ordered homomorphisms. This was not the habitual way of seeing things at that time.

This leading idea only became stronger in subsequent writings. Recall that the adjective *mathematical* appears in the very titles of three major works: *Mathematical Structures of Language* (1968), *A Grammar of English on Mathematical Principles* (1982a), and *A Theory of Language and Information: A mathematical approach* (1991). The aphorism cited as an epigraph above is the beginning of a recent essay found in *Mélanges Schützenberger* (Lothaire 1990), entitled precisely "On the Mathematics of Language".

We see here then nearly fifty years during which, to realize the program that he established very early, Zellig Harris searched and found in mathematics some of his supports. This merits closer attention, and it is doubtless advisable to consider it without shutting it into the reductive box of "possible applications of mathematics to linguistics." Is not the question rather "how could a little mathematics transmute itself into linguistics?"[1]

---

*This is a revision of an essay that appeared in Daladier (1990:85–91). Translated by Bruce Nevin with assistance of the author.

1. Harris subsequently affirmed the felicity of this expression of his aim (letter to the author, 6 Feb. 1991).

* * *

Mathematics — taking the term in a broad sense, that is, including logic — the mathematics with which Harris nourished his thinking was, quite simply, that of his times: no need to quest for a great ancestor.

From the beginning of the 20th century, logicians made an effort to give a solid foundation to the edifice of mathematics, which had been weakened by the discovery of many paradoxes. Now — chance or necessity? — this problem of *the foundations of mathematics* was more topical than ever just at the time when Harris took charge of the 'homologous' enterprise of establishing linguistics on a clear basis. Thus, the idea could be to attempt to transfer from one field to the other, certainly not methods put into practice, but their spirit, and, chief among these, the spirit of *finitism* and *constructivism*.

For the study of formal mathematics, Hilbert recommended the use of a *finitary* arithmetic as a metamathematical instrument. In this sort of arithmetic, one considers only a definite number of objects and functions that may be thought of and manipulated in an immediate and concrete manner. Because the severe restrictions imposed by finitism (this is not the place to go into technical detail) made an instrument that was too weak to reach the intended ends, subsequently the expanded point of view of *constructivism* was adopted (in which finitism appears as a special case).

In order to progress only on sure ground, constructive mathematics considered only entities constructed by means of explicitly stated rules and such that the existence of the entities in question could be held as intuitively assured. Intuitively — there you have it! The *intuitionism* of L.E.J. Brouwer (developed equally by A. Heyting) occupied among the constructivist schools a place that was important yet a bit apart from the others because it refused to define constructivity *a priori*, holding to a certain reference to intuition. Brouwer's (Kantian?) reference to intuition seems so obscure, only a philosopher would be able to examine whether Harris's thought owes something to this aspect of Brouwerian doctrine. On the other hand, it very well seems that Harris made his own the golden rule of intuitionism: its rejection of the unthinking use of *tertium non datur* (the principle of excluded middle) and its limitation to finite systems.

There is another direction of research that deserves special mention, that which was opened by the theory of types. Russell had given a first version of this as far back as 1903 in *Principles of Mathematics*. A second version, much restructured, appeared in 1910 in *Principia Mathematica*, written in collaboration with A.N. Whitehead. This theory was perceived as difficult to read, for

good reason, with many commentators denouncing its faults and obscurities. However this may be for their implementation there, the ideas on which the theory was founded are solid and of real worth. They aim to prevent the paradoxes that appeared whenever one allowed oneself to apply "anything to anything else". They set up rules. For example, it is thus that a function necessarily requires an argument of a certain type. In other words, the arguments for which a function takes values — its domain of signification — lead to a characterization by a type. A type is a logical category. And since the values of such a function can serve as arguments for another, one knows that there must exist a hierarchy of types, and that this is not 'vertical' but 'branching' (whence an order). A theory of types, to be effective, must therefore provide an effective procedure for the calculus of types (and it is in regard to the details of this calculus of types that Russell's construction did not seem always satisfactory[2]).

The most difficult aspect of developing a fruitful interconnection of the two 'problems of foundations' seemed, *a priori*, to be in the notion of metalanguage. In fact Harris knew how to make good use of the very conflicts involved. Independently of Gödel, and somewhat before him, logicians of the Polish school (Łucasiewicz, Tarski) had elaborated an infinite series stacking up language (of mathematics) and metalanguages, where every step contains the syntax for the step immediately below. How could one avoid being enmeshed in a system of this sort? On what ground could one stand?
Inversely (if one could say that), Gödel, taking ordinary arithmetic as a language, succeeded in formulating in the language *certain procedures of the metalanguage*. This is how he was able to demonstrate the famous theorem which in the 1940s scintillated in its strange novelty.

One can imagine Harris meditating on, among other things, this sort of 'polarity', and drawing from it for his program this leading idea: to use in a positive way, and even in technical details, the fact that in linguistics one cannot conceive of a metalanguage situated outside of languages.

---

2. Russell's theory of types occasioned the glosses of innumerable commentators, among them Gödel himself, who reproached him for the lack of a rigorous statement of the syntax of the formalism. One of the more recent studies is found in Rouilhan (1996). Philippe de Rouilhan has had the merit, the courage, and the patience to invest his competencies in a difficult enterprise: he has entirely rewritten the theory of types in a rigorous language, reforming the letter while scrupulously respecting the spirit of the text. The Master emerges from the test cleansed of many accusations, such as that of having horribly confused type and order.

But there can be no doubt that Harris reflected most on algebra.

Born in 1909, Harris was the exact contemporary of many great algebraists, his peers, such as G. Birkhoff and S. MacLane in the United States or A.I. Mal'tsev in the Soviet Union: it was in this sphere of influence that he found 'his' algebra.

Algebra, as we know, can be defined as the science of *symbol manipulation*. In *classical* algebra, the symbols represent numbers, real or complex, whereas in the algebra not long ago called *modern*, they represent diverse axiomatically defined non-numeric entities.

Moderne Algebra, yes, that was the title (in German) of the two-volume work appearing in 1932 where B.L. Van der Waerden vigorously synthesized the achievements of the past and opened the way to new developments. Again, we must recall that this *Great Initiatory Book* dealt essentially with 'noble' structures: infinite commutative rings and their ideals, fields, etc. In the 1950s more 'common' structures appeared: lattices, for example. Bourbaki at the time refused to accept them, but many textbooks, of which the most famous remains *A Survey of Modern Algebra*, by the same Birkhoff and MacLane (1941), have brought them into higher education, granting them in the same stroke full citizenship. And subsequently the movement grew. In this way, during the 1950s, the notion of *algebraic system*[3] was born, which would become progressively richer in the 1960s. With this, Harris was able to express certain of his views of language in a framework familiar to mathematicians as well as to linguists.

What is an algebraic system? Perhaps it can be agreed to give the most elaborated definition (due to Birkhoff), although Harris did not *explicitly* make use of this definition.[4]

We give ourselves a family of disjoint sets, the *phyla* of the system, and a set of operations, nullary, unary, binary, tertiary (possibly partial). The nullary operations distinguish certain remarkable elements. The others each send their Cartesian product of phyla into such a phylum. Finally, we give ourselves a

---

3. It was probably Mal'tsev who coined the term *algebraic system*. It is attested as long ago as 1953 in his article "Ob odnom Klasse algebraiceskih sistem". To designate such an object, many authors wrote 'algebra' (homogeneous or heterogeneous) — implying 'in the sense of Universal Algebra'. For the concept, see, for example, G. Birkhoff, "The Role of Algebra in Computing", *Computers in Algebra and Number Theory*, SIAM-AMS Proceedings, Vol. IV, Providence, Rhode Island, 1971.

4. But see e.g. Harris (1991: 305 n. 16).

certain number of binary, tertiary, etc. relations. An algebraic system is called *homogeneous* if it contains just one phylum, otherwise heterogeneous. Finally, *relational system* is the preferred term for an algebraic system that involves only relations.

To firm up these ideas, we consider from this point of view an object familiar to everyone, constituting an oriented graph G. Classically, we would consider G to be a homogeneous relational system with binary relation(s) whose points are contained by the unique phylum S. But we could also define it as a heterogeneous system including a phylum V of vertices, a phylum E of edges, and two unary operations, $\alpha : E \to V$, $\omega : E \to V$, which provide each edge e with an origin $\alpha(e)$ and an extremity $\omega(e)$. This is not all, again we could consider G to be (among others!) a heterogeneous system with three phyla, understanding E and V as above and $T = \{-1, 0, 1\}$ with the function $\phi : E \times V \to T$ such that $\phi(e, s) = -1$ if S is the origin of e, 1 if it is the extremity, otherwise 0. This definition is particularly well adapted to algebraic topology. Rudimentary though it is, this example will suggest the idea that there are in general many very different ways to 'see' the same object as an algebraic system (the *same* object at least if one thinks that a mathematical object exists in its own right, independently of the procedures employed to grasp it).

We can see also how the notion of type could be introduced to clarify rules for calculation in a heterogeneous algebraic system: certain primitive phyla receive simple types. In fact, the types exist 'virtually' and they may be amenable to being made explicit, or not.

As we have seen, the conceptual framework of algebraic systems has nothing about it of a procrustean bed: when it is a matter of a given language, the linguist remains the absolute master of the choice of phyla, operations, and relations. But this extreme liberty is available to the linguist only at the cost of some extremely difficult epistemological problems. Suppose in effect that a linguist, studying a given language L, proposes first a system S_1, and then a system S_2. Are the systems in question different only by their way of defining one and the same 'mathematical reality' proposed as a model in one case or the other? Or on the contrary do they translate a change of model corresponding to a profound change of the linguistic 'content'?

In this matter, the mathematician can shed no more light on the debate than the following observation. The notion of isomorphism (thus also of automorphism) — meaning a biunique correspondence compatible with the operations and relations — this precious and fecund notion can no longer be applied when S_1 and S_2 have different structures, even if one has reasons to

think that these systems proceed from some 'same mathematical object'.

Let us say, with Birkhoff, that when S_1 and S_2 proceed effectively from a 'same object', as in our example above, we pass from one to the other by a *cryptomorphism*. Here's the right expression: *e bene trovato!* Suffice for explanation that cryptomorphisms come under the theory of categories (algebraic functors). Some mathematicians draw the conclusion that while we are doing this it would be better immediately to undertake a formalization in the categorial framework.

So far as this concerns the linguist — but do we have the right to speak in his name? — without denying the epistemological problem, he perhaps says that a passage from S_1 to S_2 by 'simple' cryptomorphism could reflect, if necessary, a method that is rich and interesting from a linguistic point of view. For, even if S_2 is 'cryptically' equivalent to S_1, it can give to L a representation 'less tortured' than that which S_1 proposes for it, more natural and more apt to favor the progress of research. In brief, the specific task of the linguist now includes the determination of 'good' cryptomorphisms.

* * *

Harris wanted to give, even here, the leading thread appropriate to guide those who one day with a sense of responsibility will undertake to write the history of his work. Thinking of these future historians, one would suggest that mathematical considerations such as these which have just been discussed could have some usefulness. In effect, the evolution of transformational methods, when we attempt to follow it in its detail, turns out to be so complex that one must not neglect any instrument of analysis.

While awaiting the hoped-for grand essay, it seems possible to present a few introductory remarks.

The evolution of Harris's work answers to the classical schema of the spiral: it returns periodically, not of course to the point of departure, but each time to a corresponding point aligned above it.

Moreover, when Harris presented diverse theories in succession, he did not think that the latest necessarily outdated and excluded the first. In his evaluation, rather, all these theories were complementary in that they offered various points of view.

Finally, it is advisable to note that for Harris the study of numerous and diverse basic properties of sentences, together with the possibility of using selected ones of them as a central method for analyzing languages, enabled him to consider a theory of language without making appeal to a grammar of

logical forms (see on this subject the end of Section 1 of "Transformational Theory" (1965)).

Is it improper to say that this 'articulation' is situated in linguistic theory at a level that is 'homologous' with that which the articulation of cryptomorphisms occupies in algebraic theory?

Besides, what does the notion of transformation cover? Nothing has evolved more in the course of time than the allocation of the stock of 'linguistic operations' between *transformations* on the one hand and *operations* on the other, the latter tending more and more to be understood as the action of some OPERATOR.

In this evolution, we see two great periods separated by an 'intermission'. The first period, which favored a set of transformations, culminated in the system proposed in *Mathematical Structures of Language* (1968). The 'intermission' is *Report and Paraphrase* (1969), the least 'mathematical' (in appearance at least) of Harris's works. The second period opens with *Notes du Cours de Syntaxe* (1976c); it strengthens the pairs (operator, operation) and ends with the specialization of transformations as reductions, proposed in *A Grammar of English on Mathematical Principles* (1982), with further developments in *Language and Information* (1988a) and *A Theory of Language and Information* (1991), not to mention *The Form of Information in Science* (1989).

But why this evolution? What prevented Harris from being satisfied with *Mathematical Structures* (1968)? Why did he pose problems again? Doubtless there were many reasons for this — such as these, among others, that in his fashion the mathematician sees.

The *abstract system* — these are the author's terms — which *Mathematical Structures* (1968) proposed was not defined canonically as an algebraic system, but only as an ordered 6-tuple comprising a set of base N and five kinds of symbols of functions. The normalization of this object into a heterogeneous algebraic system seemed not to present difficulties *in principle* but required great care in execution. The main task evidently would be the correct determination of adequate phyla. But even amended in this way, the system would still suffer from a defect due to the PARTIAL character of the functions (in fact: semi-functions). The linguistic reasons that explain this partial character had no counterpart that was more or less satisfactory, much less elegant, in pure mathematics.

What to do so that a semi-function found its arguments in a natural way, mathematically speaking?

It is in this connection that it would turn out to be useful to have read — as Harris had read — the theory of types of B. Russell. From *Notes du Cours de*

Syntaxe (1976c) there begin in effect to appear typed operators. The mathematicization moves toward an applicative calculus controlled by types permitting the definition of the structure of a sentence as a clearly defined partially ordered structure (see for example the *Grammar of English on Mathematical Principles* (1982)).

The way is opened to new developments in the future.

Extending what has just been said, from here it is possible to enrich and to specify the combinatorics of types and to make the applicative calculus benefit from the rich experience recently acquired in the domain of lambda-calculus. It is equally conceivable to reconsider the heterogeneous system considered above while bringing to bear the system of types. Perhaps one could arrive at an object cryptically linked with a certain typed lambda calculus. In this case, and for every other formalism that may be considered, it is evidently appropriate to judge as a last resort from the point of view of pertinence and linguistic transparency.

To conclude, it is perhaps appropriate to prevent a possible misunderstanding. Historical considerations have suggested a correspondence between two 'foundation problems', but it would be false to pursue the parallel too far. In the eyes of the mathematicians of today, the hope of *founding* mathematics seems to be a chimaera: mathematics is not to be founded. Does this mean that the immense efforts put out for the sake of resolving a problem now held to be impossible were all for nothing? Certainly not! Thanks to these efforts, we understand better what is and what can be the activity of mathematicians, our freedom and our responsibility, and the obligation to question ourselves periodically.

What are the integers, really? What is the continuum today? The idea prevails henceforth that the mathematician must work on the basis of *temporary agreement*, confronting the *horizon of the moment*.

In linguistics, the situation is necessarily different because — a banal observation — the facts that this science deals with are in space-time, partaking of the 'real world'. A banal observation from which the different schools draw different conclusions, even those that agree in placing a certain confidence in the utility of mathematics.

Harris, for his part, rejects any method that would fit the facts of linguistics into prefabricated formalisms, to which it is then necessary only to make a few adjustments. He believes on the contrary in the validity of mathematical structures progressively extricated from observables: such is his manner of 'founding'.

In brief, Harris does not require of mathematics anything off the shelf, but rather, as in the excellent title recalled above, *principles*.

References

Birkhoff, G. 1971. 'The role of algebra in computing'. *Computers in Algebra and Number Theory*, SIAM-AMS Proceedings, Vol. IV, Providence, Rhode Island.

Birkhoff, Garrett & Saunders MacLane. 1941. *A survey of modern algebra*. New York: Macmillan.

Harris, Zellig S. 1946a. "From morpheme to utterance". *Language* 22: 3.161–183.

Harris, Zellig S. 1965. "Transformational Theory". *Language* 41: 3.363–401.

Harris, Zellig S. 1968. *Mathematical Structures of Language*. (=*Interscience Tracts in Pure and Applied Mathematics*, 21.) New York: Interscience Publishers, John Wiley & Sons.

Harris, Zellig S. 1969. "The two systems of grammar: Report and paraphrase". Transformations and Discourse Analysis Project (TDAP) 79. Philadelphia: The University of Pennsylvania.

Harris, Zellig S. 1976c. *Notes du cours de syntaxe*. Translated and presented by Maurice Gross. Paris: Éditions du Seuil.

Harris, Zellig S. 1982a. *A Grammar of English on Mathematical Principles*. New York: John Wiley & Sons.

Harris, Zellig S. 1988a. *Language and Information*. (=Bampton Lectures in America, 28.) New York: Columbia Univ. Press.

Harris, Zellig S., Michael Gottfried, Thomas Ryckman, Paul Mattick, Jr., Anne Daladier, Tzvee N. Harris, & Suzanna Harris. 1989. *The Form of Information in Science: Analysis of an immunology sublanguage*. Preface by Hilary Putnam. (=Boston Studies in the Philosophy of Science, 104.) Dordrecht/Holland & Boston: Kluwer Academic Publishers.

Harris, Zellig S. 1990. "On the mathematics of language". In M. Lothaire (ed), *Mots (Mélanges offerts à M.-P. Schützenberger*). Paris: Éditions Hermes.

Harris, Zellig S. 1991. *A Theory of Language and Information: A mathematical approach*. Oxford & New York: Clarendon Press.

Mal'tsev, A.I. 1953. "Ob odnom Klasse algebraiceskih sistem" ["On one class of algebraic systems"]. *Uspehi Matematiceskih Nauk* VIII.1(53):165–171.

Rouilhan, Philippe de. 1996. *Russel et le cercle des paradoxes*. Paris: Presses Universitaires de France.

Waerden, B. L. van der. 1932. *Moderne Algebra*. Berlin: Springer.

Whitehead, Alfred North, and Bertrand Russell. 1910–1913. *Principia Mathematica*, Vol. 1, 1910, Vol. 2, 1912, Vol. 3, 1913. Cambridge: University Press.

Part 1

Mathematics and formal systems

Chapter 1

Formal grammar and information theory: together again?*

Fernando Pereira
University of Pennsylvania

1. The great divide

In the last forty years, research on models of spoken and written language has been split between two seemingly irreconcilable points of view: formal linguistics in the Chomsky tradition, and information theory in the Shannon tradition. Chomsky's famous example signals the beginning of the split:

> (1) Colorless green ideas sleep furiously.
> (2) Furiously sleep ideas green colorless.
> . . . It is fair to assume that neither sentence (1) nor (2) (nor indeed any part of these sentences) has ever occurred in an English discourse. Hence, in any statistical model for grammaticalness, these sentences will be ruled out on identical grounds as equally 'remote' from English. Yet (1), though nonsensical, is grammatical, while (2) is not. (Chomsky 1957: 16)

---

*This is a revised version of of a paper originally published in the *Philosophical Transactions of the Royal Society*, Series A 358.1769: 1239–1253. I would like to thank Gerald Gazdar and Karen Spark Jones for their careful reading of the original version and illuminating comments; Bruce Nevin for correcting mistakes in my reading of Harris, detailed comments, and extraordinary editorial patience; Ido Dagan, Lillian Lee, Larry Saul, Yves Schabes, Yoram Singer, Amit Singhal, and Tali Tishby, for the joint research that helped shape these ideas; Yoav Freund, Michael Kearns, and Rob Schapire, for guidance on learning theory; and Steve Abney, Hiyan Alshawi, Michael Collins, Don Hindle, Mark Johnson, Aravind Joshi, John Lafferty, David McAllester, Glyn Morrill, Michael Moortgat, Hinrich Schütze, Stuart Shieber, and Ed Stabler for many conversations on these topics over the years. I am sure that each of them will have good reasons to disagree with some of my arguments and interpretations, but nevertheless their help was invaluable in this effort to reconcile the two rich traditions in the study of language that most of my work derives from.

Before and after the split, Zellig Harris had advocated a close alliance between grammatical and information-theoretic principles in the analysis of natural language (Harris 1951; Harris 1991). Early formal-language theory provided another strong link between information theory and linguistics. Nevertheless, in most research on language and computation, those bridges were lost in an urge to take sides that was as much personal and ideological as scientific.

Today, after many years in the defensive, the information-theoretic view is again thriving and has led to practical successes in speech recognition, information retrieval, and, increasingly, in language analysis and machine translation. The exponential increase in the speed and storage capacity of computers is the proximate cause of these successes, allowing the automatic estimation of the parameters of computational models of language by counting occurrences of linguistic events in very large bodies of text and speech. However, vastly increased computer power would be irrelevant if automatically derived models or linguistic data were not able to *generalize* to unseen data. I will argue below that progress in the design and analysis of such models is not only playing a central role in those practical advances but also carries the promise of fundamentally deeper understanding of information-theoretic and computational-complexity constraints on language acquisition.

2. Harris's program

The ascent of Chomskian generative linguistics in the early 1960s swept the focus of attention away from distributional views of language, especially those based on the earlier structuralist tradition. In that tradition, Zellig Harris developed what is probably the best-articulated proposal for a marriage of linguistics and information theory. This proposal involves the three constraints of *partial order*, *likelihood* and *reduction* as well as a *linearization* process (Harris 1988):

— Partial order: "[. . .] for each word [. . .] there are zero or more classes of words, called its arguments, such that the given word will not occur in a sentence unless one word [. . .] of each of its argument classes is present." (Harris 1988: 12)

  There is a strong similarity between the argument class information for a word as suggested by Harris and its type in categorial grammar, or subcategorization frames in other linguistic formalisms. However, traditional

categorial grammar (Lambek 1958) conflates function-argument relationships and linear order, whereas Harris factors out linear order explicitly. It is only more recently that categorial grammar has acquired the technical means to investigate such factorizations (Morrill 1994; Moortgat 1995). It becomes then clear that Harris's partial order may be formalized as the partial order among set-theoretic function types. However, unlike modern categorial grammar, Harris's partial order constraint specifies only the basic configurations corresponding to elementary clauses, while complex clauses are a result of applying another constraint, reduction, to several elementary clauses after linearization.

— Likelihood: "[. . .] each word has a particular and roughly stable likelihood of occurring as argument, or operator, with a given other word, though there are many cases of uncertainty, disagreement among speakers, and change through time." (Harris 1988: 16)

Using current terminology, one might interpret the likelihood constraint as a probabilistic version of selectional restrictions. However, Harris makes a sharp distinction between general language, in which likelihoods for fillers of argument positions represent tendencies, and technical sublanguages, in which there are hard constraints on argument fillers, and which, thus, correspond more closely to the usual notion of selectional restriction.

— Reduction: "It consists, for each language, of a few specifiable types of reduction [. . .] what is reduced is the high-likelihood [. . .] material [. . .]; an example is zeroing the repeated corresponding words under *and*." (Harris 1988: 20)

The reduction constraint accounts both for morphological processes like contraction, and for processes that combine elementary clauses into complex clauses, such as relativization, subordination and coordination. In each case, Harris claims that high-likelihood material may be elided, although it would seem that additional constraints on reduction may be necessary. Furthermore, connections between reduction-based and transformational analyses (Harris 1965; Chomsky 1965) suggest the possibility of modeling string distributions as the overt projections of a hidden generative process involving operator-argument structures subject to the likelihood constraint and transformations. Recent work linking transformational and categorial approaches to syntax makes this possibility especially intriguing (Stabler 1997; Cornell 1997).

— Linearization: "Since the relation that makes sentences out of words is a

> partial order, while speech is linear, a linear projection is involved from the start." (Harris 1988: 24)
>
> The linearization process introduces opportunities for reduction by creating predictable configurations such as the occurrence of a noun in a main clause and in a adjoined clause, which allows the replacement of the second occurrence by a relative pronoun, or even its complete elision in some cases.

Thus, linguistic events involve the generation of basic configurations — unordered simple clauses — whose structure is determined by the partial order constraint and whose distribution follows the probabilities associated with the likelihood constraint. Predictable linearized configurations are then subject to reduction (compression from the information-theoretic perspective).[1] As I will discuss in Section 7, though, the likelihood constraint as stated by Harris, or its current version, leaves out dependencies on the broader discourse context that strongly affect the likelihoods of linguistic events.

For the present discussion, the most important feature of Harris's constraints is how they explicitly link linguistic structure with distributional regularities involving the relative frequencies of different operator-argument configurations. For Harris, the critical phenomena to be explained are the *departures from equiprobability* (Harris 1991: 30–31) in possible combinations of linguistic units. These departures support procedures for discovering higher-level units (Harris 1991: 32):

> [. . .] when only a small percentage of all possible sound-sequences actually occurs in utterances, one can identify the boundaries of words, and their relative likelihoods, from their sentential environment [. . .]

However, such procedures are underspecified in the absence of constraints on the form of the combination of lower-level elements into higher-level ones (Harris 1991: 33):

> If then, we have for a language different descriptions, adequate to characterize its utterances, but with different amounts of departures from equiprobability ascribed at different points in the course of the descriptions, we opt for the one with least such departures, since that one has clearly added least to the inherent departures in the language being described.

---

1. Certain kinds of reduction, for instance VP ellipsis, might however be better understood as operating at the partial-order level (Dalrymple et al. 1991).

For Harris, minimizing redundancy in linguistic description is a methodological requirement on the linguist that constrains the form of the description:

> The effect of the least-redundancy test [. . .] is a grammar with fewest possible and most independent objects [. . .], fewest and least-intersecting classes of objects, fewest and most independent rules [. . .] on the objects, fewest differences in domain for the rules, and finally, fewest abstract constructs. (Harris 1991: 33)

As we shall see, learning theory advances this line of argumentation by shifting from constraints on the form of descriptions to combinatorial constraints on the kinds of distinctions the grammar can make. This also allows us to extend these information-theoretic arguments beyond the methods of the linguist to computational constraints on language users. While the linguist and the language user perform very different tasks, they both face the task of building from finite evidence a description of language — explicit for the linguist, implicit for the language user — that generalizes to new linguistic events. Therefore, information-theoretic constraints on such inductive processes are relevant not only to Harris's program but also to the analysis of language acquisition.

3. Generalization

While Harris in the 1950s discussed the functional role of distributional regularities in language, he proposed no specific mechanisms by which language users could take advantage of those regularities in language acquisition and use. In particular, it is not obvious that language users can acquire stable distributional information, let alone the lexical and grammatical information required by the partial-order, reduction and linearization constraints, from the limited evidence that is available to them from their linguistic environment. This question created a great opening for Chomsky's rationalist critique of empiricist and structuralist linguistics, of which the 'green ideas' quote above is an early instance.

Chomsky concluded that sentences (1) and (2) are equally unlikely from the observation that neither sentence or 'part' thereof would have occurred previously (Abney 1996). From this observation, he argued that any statistical model based on the frequencies of word sequences would have to assign equal, zero, probabilities to both sentences. But this relies on the unstated assumption that any probabilistic model necessarily assigns zero probability to unseen events. Indeed, this would be the case if the model probability estimates were

just the relative frequencies of observed events (the *maximum-likelihood* estimator). But we now understand that this naïve method badly *overfits* the training data.

The problem of overfitting is tightly connected with the question of how a learner can generalize from a finite training sample. The canonical example is that of fitting a polynomial to observations. Given a finite set of observed values of a dependent random variable Y for distinct values of the independent variable X, we seek an hypothesis for the functional dependency of Y on X. Now, any such set of observations can be fitted exactly by a polynomial of high-enough degree. But that curve typically is a poor predictor of any new observation because it matches exactly the peculiarities of the training sample. To avoid this, one usually *smoothes* the data, using a lower-degree polynomial that may not fit the training data exactly but that is less dependent on the vagaries of the sample. Similarly, smoothing methods can be used in probability models to assign some probability mass to unseen events (Jelinek & Mercer 1980). In fact, one of the earliest such methods, due to Turing and Good (Good 1953), had been published before Chomsky's attack on empiricism, and has since been used to good effect in statistical models of language (Katz 1987).

The use of smoothing and other forms of *regularization* to constrain the form of statistical models and ensure better generalization to unseen data is an instance of a central theme in statistical learning theory, that of the *sample complexity* relationship between training sample size, model complexity, and generalization ability of the model. Typical theoretical results in this area give probabilistic bounds on the generalization error of a model as a function of model error on training data, sample size, model complexity, and margin of error (Vapnik 1995). In qualitative terms, the gap between test and training error — a measure of overfitting — grows with model complexity for a fixed training sample size, and decreases with sample size for a fixed model complexity.

To quantify the trade-off between training set accuracy, generalization to new data, and constraints on the model, we need a rigorous measure of model complexity. In the polynomial example, the usual intuition is that complexity is measured by the degree of the polynomial (the number of tunable coefficients in the model), but intuitions are harder to come by for model classes without a simple parametric form. Furthermore, even in the polynomial case, the common-sense complexity measure can be misleading, because certain approaches to polynomial fitting yield much smaller model complexity and thus better generalization ability (Vapnik 1995).

The definition of model complexity is also intimately tied to assumptions regarding the distribution of the data presented to the learner. In statistics, it is usual to assume that the data has a distribution of known form but unknown parameters. Statistical learning theory commonly adopts the *distribution-free* view, in which the data is generated by a fixed but arbitrary distribution (Valiant 1984). Finally, one may adopt an *on-line* view, in which the data is not assumed to be generated by a fixed distribution, and in fact may be generated by a malicious adversary. In this view, the goal of the learner is to be *competitive* with a given set of *experts*, that is, to do almost as well on any observation sequence as the expert that performs best on that sequence, given only the sequence so far and the past performance of the experts (Littlestone & Warmuth 1994; Freund & Schapire 1997).

A crucial idea from the distribution-free setting is that model complexity can be measured, even for an infinite model class, by combinatorial quantities such as the *Vapnik-Chervonenkis* (*VC*) *dimension* (Vapnik & Chervonenkis 1971), which roughly speaking gives the order of a polynomial upper bound on how many distinctions can be made between samples by models in the class, as a function of sample size.

Returning to the debate between empiricism and rationalism, the relationships between model complexity, sample size and overfitting developed in learning theory may help clarify the famous argument from *poverty of the stimulus* (APS). Reacting to empiricist and especially behaviorist theories, Chomsky and others have argued that general-purpose learning abilities are not sufficient to explain children's acquisition of their native language from the (according to them) very limited linguistic experience that is available to the learner. In particular, they claimed that linguistic experience does not provide negative examples of grammaticality, making the learner's task that much harder. Therefore, they conclude, a specialized innate language faculty must be involved. The 'green ideas' comparison is an early instance of the same argument, asserting that statistical procedures alone cannot acquire a model of grammaticality from the data available to the learner.

The APS does not just require restrictions on model classes to ensure effective generalization from finite data, which would be unobjectionable from a learning-theoretic viewpoint. In its usual form, the APS also claims that only a learning mechanism developed specifically for language could generalize well from limited linguistic experience. The flaw in this argument is that it assumes implicitly that the only constraints on a learner are those arising from particular *representations* of the learner's knowledge, whereas we now know that the

informational difficulty of learning problems can be characterized by purely combinatorial, representation-independent, means. Statistical learning theory gives us the tools to compute empirically-testable lower bounds on sample sizes that would guarantee learnability for given model classes, although such bounds can be very pessimistic unless they also take into account computational constraints on the model fitting procedure (Freund 1998). Nevertheless, it is unlikely that the debate over the APS can become empirically grounded without taking into account such calculations, since the stimuli that APS supporters claimed to be missing are actually present with significant frequency (Pullum 1996).

The APS reached an extreme form with Chomsky's principles-and-parameters theory, according to which learnability requires that the set of possible natural languages be generated by the settings of a finite set of finitely valued parameters (Chomsky 1986: 149). But this extreme constraint is neither necessary, since infinite model classes of finite VC dimension are learnable from an information-theoretic point of view, nor sufficient, because even finite classes may not be *efficiently* learnable, that is, the search for a model with good generalization may be computationally intractable[2] even though the information is in principle available (Kearns & Valiant 1994).

4. Hidden variables

Early empiricist theories of linguistic behavior made themselves easy targets of critiques like that of Chomsky (1959) by denying a significant role for the internal, unobservable, state of the language user. Thus, in a Markov model of language, all the state information would be represented by the externally observable sequence of past linguistic behavior. However, even in this case, the empiricist position was being caricatured. If we consider a language user that updates internal expectations and probable responses according to statistics collected from past experience, those expectations and response propensities, however represented, are a part of the user state that is not directly available to observation. Furthermore, the behavior of language users may give valuable

2. I use 'intractable' in this chapter in the usual sense from theory of computation of a problem that has been proven to belong to one of the standard classes believed to require more than polynomial time on a deterministic sequential computer, for instance the NP-hard problems.

information about the power of their experience-encoding mechanisms. For instance, a language user that maintains statistics over pairs of consecutive words only (bigram statistics) might be less effective in anticipating and reacting appropriately to the next word than a user that keeps statistics over longer word sequences; in other words, the bigram model may have higher entropy. This example, related to finite-state text compression, may seem simplistic from the point of view of linguistics, but it is a convenient testbed for ideas in statistical modeling and constraints on model structure, and introduces the idea of a hidden modeling state in a very simple form.

Hidden *random* variables in a language user's state, or rather statistics involving their joint values, represent the user's uncertainty about the interpretation, and best response to, events observed so far. Such uncertainty may not be just over the interpretation of a particular course of events, but also over which particular model in a class of models is a best compromise between fitting the experience so far and generalizing to new experience. When the best choice of model is uncertain, Bayesian *model averaging* (Willems et al. 1995) can be used to combine the predictions of different candidate models according to the language user's degree of belief in them, as measured by their past success. Model averaging is thus a way for learners to hedge their bets on particular grammars, in which the initial bets represent a *prior* belief on particular grammars and are updated according to a regularizing procedure that balances fit to immediate experience with predictive power over new experience. The prior distribution on grammars can be seen as a form of innate knowledge that implicitly biases the learner towards 'better' — in particular, less complex — grammars. In particular, any infinite class of grammars can be given a *universal* prior distribution based on the number of bits needed to encode members of the class, which favors the least complex grammars compatible with the data (Solomonoff 1964; Horning 1969). Universal prior distributions over grammars have a close connection with compression and the *minimum description length* principle (Rissanen 1989), according to which the best grammar minimizes the number of bits required to compress the grammar itself and the raw linguistic data given the grammar.

Universal prior distributions and minimum description length provide possible quantitative interpretations of Harris's notion of grammatical simplicity. However, those concepts do not provide a way of quantifying the relationship between a prior distribution over grammars, training sample size, and generalization power, and in any case seem to have been ignored by those interested in language acquisition and the APS. Recent advances in statistical

learning theory (McAllester 1999) may provide new theoretical impetus to that research direction, since they show that a prior distribution over models can play a regularizing role similar to that of a combinatorial complexity measure.

The other role for hidden variables, capturing uncertainty in the interpretation of particular experience, becomes especially interesting in modeling ambiguity. Going back to Harris's theory, each of the constraints involves covert choices by the language user: assignment of types — positions in the partial order — to lexical items, lexical choice according to selection probabilities, reduction choices according to the distributional statistics of predictability, and linearization choices. More generally, any model of language that appeals to non-observables, for instance any model that assigns syntactic analyses, will require hidden variables.

Hidden variables representing uncertainty of interpretation can also be used to create *factored* models of joint distributions that have far fewer parameters to estimate, and are thus easier to learn, than models of the full joint distribution. As a very simple but useful example, we may approximate the conditional probability $p(x, y)$ of occurrence of two words x and y in a given configuration as

$$p(x,y) = p(x)\sum_c p(y \mid c)\ p(c \mid x)$$

where c is a hidden "class" variable for the associations between x and y in the configuration under study. For a vocabulary of size V and C classes, this model uses $O(CV)$ parameters rather than the $O(N^2)$ parameters of the direct model for the joint distribution and is thus less prone to overfitting if $C \ll V$. In particular, when $(x,y)=(v_i, v_{i+1})$ we have an *aggregate* bigram model (Saul & Pereira 1997), which is useful for modeling word sequences that include unseen bigrams. With such a model, we can approximate the probability of a string $p(w_1 \ldots w_n)$ by

$$p(w_1 \ldots w_n) = p(w_1)\prod_{i=2}^{n} p(w_i \mid w_{i-1}).$$

Using this estimate for the probability of a string and an aggregate model with $C=16$ trained on newspaper text using the expectation-maximization (EM) method (Dempster et al. 1977), we find that

$$\frac{p(\text{Colorless green ideas sleep furiously})}{p(\text{Furiously sleep ideas green colorless})} \approx 2 \times 10^5$$

Thus, a suitably constrained statistical model, even a very simple one, can meet Chomsky's particular challenge.

A plausible and well-defined model of the statistical dependencies among the hidden variables is however not in general sufficient, since the problem of setting the corresponding conditional probabilities from observable linguistic material is in most cases computationally intractable (Abe & Warmuth 1992). Nevertheless, those intractability results have not precluded significant algorithmic and experimental progress with carefully designed model classes and learning methods such as EM and variants, especially in speech processing (Baum & Petrie 1966; Baker 1979). In particular, the learning problem is easier in practice when interactions between hidden variables can be approximately factored in terms of the observed variables.

5. Lexicalized models

Harris's model of dependency and selection is *lexicalized* in the sense that all postulated relationships are between the words in (precursors for) sentences, rather than the relationships between structures postulated in generative grammar. From the points of view of distributional modeling and machine learning, an important property of lexicalized models is that they anchor analyses in observable cooccurrences between words, rather than in unobservable relationships among hypothetical grammatical structures.[3] In a probabilistic setting, a way to state this more precisely is that lexicalization makes it easier to factor the interactions between the hidden variables by conditioning on the observed sentence.

Even lexicalized models will involve hidden decisions if they allow ambiguous interpretations. As noted in the previous section, hidden-variable models are computationally difficult to learn from evidence involving the observable variables alone. An alternative strategy is to constrain the hidden variables by

3. This property could well make lexicalized models less rather than more palatable to Chomskian linguists, for whom structural relationships are the prime subject of theory. But notice that Chomsky's more recent "minimalist program" (Chomsky 1995) is much more lexically-based than any of his theories since "Aspects" (Chomsky 1965), in ways that are reminiscent of other lexicalized multistratal theories, in particular lexical-functional grammar (Bresnan 1982), HPSG (Pollard & Sag 1994), and certain varieties of categorial grammar (Morrill 1994; Moortgat 1995; Cornell 1997).

associating sentences with disambiguating information. At one extreme, that information might be a full syntactic analysis. In this case, which is very interesting from computational and applications perspectives, recent work has shown that lexicalized probabilistic context-free grammars can be learned automatically that perform with remarkable accuracy on novel material (Charniak 1997; Collins 1998). Besides lexicalization, these models factor the sentence generation process into a sequence of conditionally independent events that reflect such linguistic distinctions as those of head and dependent and of argument and adjunct. That is, the models are in effect lexically-based *stochastic generative grammars*, and the conditional independence assumptions on the generation process are a particular kind of Markovian assumption. Crucially, these assumptions apply to the hidden generative decisions, not to the observable utterance, and thus allow for analysis ambiguity.

The learning algorithms just discussed need to be given the full correct syntactic analysis of each training example, and are thus not realistic models of human language acquisition. One possible direction for reducing the unrealistic amount of supervision required would be to use instead additional observables correlated with the hidden variables, such as prosodic information or perceptual input associated with the content of the linguistic input (Siskind 1996; Roy & Pentland 1999). More generally, we may be able to replace direct supervision by indirect correlations, as I now discuss.

6. The power of correlations

How poor is the stimulus that the language learner exploits to acquire its native language? Linguistic experience is not just a string of words, but it is *grounded* in a rich perceptual and motor environment that is likely to provide crucial clues to the acquisition, interpretation and production processes, if for no other reason than for the functional one that much of the linguistic experience is *about* that non-linguistic environment. But this points to a fundamental weakness in much of the work discussed so far: both in formal grammar and in most computational models of language, language is taken as a completely autonomous process that can be independently analyzed.[4] Indeed, a simplistic

4. I include under this description all the work on formal semantics of natural language, since logical representations of the meanings of sentences are as unobservable as syntactic analyses, and thus equally artificial as inputs to a language acquisition process.

use of information theory suffers from the same problem, in that the basic measures of information content of a signal are intrinsic, rather than relative to the correlations between a signal and events of interest (the meaning(s) of the signal). Fortunately, information theory provides a ready tool for quantifying *information about* with the notion of *mutual information* (Cover & Thomas 1991), from which a suitable notion of compression relative to side variables of interest can be defined (Tishby et al. 1999).

Given the enormous conceptual and technical difficulties of building a comprehensive theory of grounded language processing, treating language as an autonomous system is very tempting. However, there is a weaker form of grounding that can be exploited more readily than physical grounding, namely grounding in a *linguistic* context. Following this path, sentences can be viewed as evidence for other sentences through inference, and the effectiveness of a language processor may be measured by its accuracy in deciding whether a sentence entails another, or whether an answer is appropriate for a question.

Furthermore, there is much empirical evidence that linguistic grounding carries more information than it might seem at first sight. For instance, all of the most successful information retrieval systems ignore the order of words and just use the frequencies of words in documents (Salton 1989) in the so-called *bag-of-words* approach. Since similar situations are often described in similar ways, simple statistical similarity measures between the word distributions in documents and queries are effective in retrieving documents relevant to a given query. In the same way, word senses can be automatically disambiguated by measuring the statistical similarity between the bag of words surrounding an occurrence of the ambiguous word and the bags of words associated with definitions or examples of the different senses of the word (Schütze 1997).

In both information retrieval and sense disambiguation, bag-of-words techniques are successful because of the underlying coherence of purposeful language, at syntactic, semantic, and discourse levels. The *one sense per discourse* principle (Gale et al. 1992) captures a particular form of this coherence. For example, the cooccurrence of the words "stocks", "bonds" and "bank" in the same passage is potentially indicative of a financial subject matter, and thus tends to disambiguate those word occurrences, reducing the likelihood that the "bank" is a river bank, that the "bonds" are chemical bonds, or that the "stocks" are an ancient punishment device. These correlations, like the correlations between utterances and their physical context, allow a language processor to learn from its linguistic environment with very little or

no supervision (Yarowsky 1995), and have suggested new machine-learning settings such as *co-training* (Blum & Mitchell 1998).

Both lexicalized grammars and bag-of-words models represent statistical associations between words in certain configurations. However, the kinds of associations represented are rather different. The associations in lexicalized grammars are mediated by a hidden assignment of dependency relationships to pairs of word occurrences in an utterance. Many such assignments are potentially available, leading to great structural ambiguity, as discussed in Section 5. In contrast, the associations in bag-of-words models and many other statistical models (for instance, Markov models) are defined over very impoverished but unambiguous overt structures. Furthermore, effective lexicalized models must make drastic statistical independence assumptions between parts of the underlying structure, thus missing the global coherence correlations that bag-of-words models capture.

7. Local structure and global distribution

Current stochastic lexicalized models, with their lexically-determined local correlations, capture much of the information relevant to Harris's partial-order and likelihood constraints. However, unlike Harris but like dependency grammar and other monostratal grammatical formalisms, they conflate linearization with the argument structure given by the partial-order constraint.

In asserting the 'rough stability' of the likelihood of a given argument of a given operator, Harris assumed implicitly a generative model in which dependents are conditionally independent of the rest of an analysis given the head they depend on. Existing lexicalized models use similar Markovian assumptions, although they typically extend lexical items with additional features, for instance syntactic category (Charniak 1997; Collins 1998). However, Harris's information-theoretic arguments, especially those on reduction, refer to the overall likelihood of a string, which involves the global correlations discussed in the last section. But such global correlations are precisely what the Markovian assumptions in generative models leave out.

Thus Markovian generative models are not able to model the potential correlations between the senses assigned to occurrences of "stocks" and "bonds" in different parts of a paragraph, for example. This problem may be addressed in two main ways. The first is to preserve Markovian assumptions, but to enrich lexical items with features representing alternative global coher-

ence states. For instance, lexical items might be decorated with sense features, and local correlations between those would be used to enforce global coherence. Those features might even be other lexical items, whose cooccurrence with the given items as operators or arguments may disambiguate them. The difficulty with this approach is that it introduces a plethora of hidden variables, leading to a correspondingly harder learning problem. Furthermore, it relies on careful crafting of the hidden variables, for instance in choosing informative sense distinctions. The second approach is to adopt ideas from random fields and factor probabilities instead as products of exponentials of indicator functions for significant local or global *features* (events) (Della Pietra et al. 1997; Ratnaparkhi 1999; Abney 1997; Rosenfeld 2000), which can be built incrementally with 'greedy' algorithms that select at each step the most informative feature.

8. From deciding to understanding

Models based on information-theoretic and machine-learning ideas have been successful in a variety of language processing tasks, such as speech recognition and information retrieval. A common characteristic of most of those tasks is that what is sought is a decision among a finite set of alternatives, or a ranking of alternatives. For example:

1. A newswire filter classifies news stories into topics specified by training examples.
2. A part-of-speech tagger assigns the most likely tags to the words in a document.
3. A Web search engine ranks a set of Web pages according to their relevance to a natural-language query.
4. A speech recognizer decides among the possible transcriptions of a spoken utterance.

In each case, the task can be formalized as learning a mapping from spoken or written material to a choice or ranking among alternatives. As we know from the earlier discussion of generalization, we need to restrict our attention to a class of mappings that can be actually be learned from the available data. Computational considerations and experimental evaluation will narrow further the mapping classes under consideration. Finally, a suitable optimization procedure is employed to select from the class a mapping that minimizes some measure of the error on the training set.

A potential weakness of such task-directed learning procedures is that they ignore regularities that are not relevant to the task. Yet, those regularities may be highly informative about other questions. While language may be redundant with respect to any particular question, and a task-oriented learner may benefit greatly from that redundancy as discussed earlier, it does not follow that language is redundant with respect to the set of all questions that a language user may need to decide. Furthermore, one may reasonably argue that a task-oriented learner does not really 'understand' language, since it can decide accurately just one question, while our intuitions about understanding suggest that a competent language user can decide accurately many questions pertaining to any discourse it processes. For instance, a competent language user should be able to answer reliably 'who did what to whom' questions pertaining to each clause in the discourse.

We are drawn thus to the question of what kinds of learning tasks may involve 'understanding' but do not force us to attack frontally the immense challenges of grounded language processing. Automatically-trained machine translation (Brown et al. 1990; Alshawi & Douglas 2000) may be such a task, since translation requires many questions about a text to be answered accurately to produce a correct output. Nevertheless, it is easy to find many other reasonable questions that can be left unanswered while still performing creditably on the task. Indeed, there is no single 'understanding' task, but rather a range of tasks whose difficulty can be measured by the uncertainty — information-theoretically, the entropy — of the output in the absence of any information about the input. The objective of a learner is then to acquire a function that can reduce that uncertainty by exploiting the mutual information between inputs and outputs (Tishby & Gorin 1994). Tasks (1-4) above are listed roughly in order of increasing output entropy, with machine translation being possibly even more difficult.

The theoretical representations postulated by formal linguistics — constituent structure, functional and dependency structures, logical form — can also be understood as codified answers to particular kinds of questions pertaining to the text, with their own degrees of information-theoretic difficulty. For instance, different assignments of arguments to thematic roles lead to different correct answers to 'who did what to whom' questions. From this point of view, the task of the learner is to acquire an accurate procedure for deciding whether a simple sentence follows from a discourse, rather than the more traditional tasks of deciding grammaticality or assigning structural descriptions. Structural descriptions may still play an important role in such a theory, but now as

a technical language for informational relationships between linguistic events instead of end-products of the theory. We thus return to Harris's original insight that departure from equiprobability is the only source of knowledge of language in the absence of an external metalanguage.

9. Summary

While researchers in information retrieval, statistical pattern recognition, and neural networks kept developing theoretical and experimental approaches to the problem of generalization, that work was ignored by formal linguistics for both cultural and substantive reasons. Among the substantive reasons, possibly the most important was that the models proposed, even if successful in practice, failed to capture the productive, recursive nature of linguistic events.

Recent advances in machine learning and statistical models are starting to supply the missing ingredients. Lexicalized statistical models informed by linguistic notions such as phrase head, argument, and adjunct specify how complex linguistic events can be generated and analyzed as sequences of elementary decisions. Machine learning suggests how rules for the elementary decisions can be learned from examples of behavior, and how the learned decision rules generalize to novel linguistic situations. Probabilities can be assigned to complex linguistic events, even novel ones, by using the causal structure of the underlying models to propagate the uncertainty in the elementary decisions.

Such statistical models of local structure are complemented by the models of larger-scale correlations that have been developed in information retrieval and speech recognition. These models have proven quite successful in learning automatically how to rank possible answers to a given question, but it is still unclear how they may combine with lexical models in a unified account of the relationship between linguistic structure and statistical distribution.

Furthermore, we have barely touched the question of what such models may say about human language acquisition. Although statistical learning theory and its computational extensions can help us ask better questions and rule out seductive *non sequiturs*, their quantitative results are still too coarse to narrow significantly the field of possible acquisition mechanisms. However, some of the most successful recent advances in machine learning arose from theoretical analysis (Cortes & Vapnik 1995; Freund & Schapire 1997), and theory is also helping to sharpen our understanding of the power and limitations of informally-designed learning algorithms.

All in all, while much remains to be done, we may well be seeing the beginning of a new version of the Harris program, in which computational models constrained by grammatical considerations define broad classes of possible grammars, and information-theoretic principles specify how those models are fitted to actual linguistic data.

References

Abe, N. & M. Warmuth. 1992. "On the computational complexity of approximating distributions by probabilistic automata." *Machine Learning* 9: 205–260.

Abney, S. 1996. "Statistical methods and linguistics." In J. L. Klavans & P. Resnik (eds.), *The balancing act.* Cambridge, Massachusetts: MIT Press.

Abney, S. 1997. "Stochastic attribute-value grammars." *Computational Linguistics* 23.4:597–618.

Alshawi, H. & S. Douglas. 2000. "Learning dependency transduction models from unannotated examples." *Philosophical Transactions of the Royal Society* Series A 358.1769: 1357–1372.

Baker, J. K. 1979. "Trainable grammars for speech recognition." In J. J. Wolf & D. H. Klatt (eds.), *97th Meeting of the Acoustical Society of America.* Cambridge, Massachusetts.

Baum, L. E. & T. Petrie. 1966. "Statistical inference for probablistic functions of finite state Markov chains." *Annals of Mathematical Statistics* 37: 1554–1563.

Blum, A. & T. Mitchell. 1998. "Combining labeled and unlabeled data with co-training." *Proceedings of the 11th Annual Conference on Computational Learning Theory.* New York: ACM Press.

Bresnan, J. (ed.). 1982. *The Mental Representation of Grammatical Relations.* Cambridge, Massachusetts: MIT Press.

Brown, P., J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, & P. S. Roossin. 1990. "A statistical approach to machine translation." *Computational Linguistics* 16.2: 79–85.

Charniak, E. 1997. "Statistical parsing with a context-free grammar and word statistics." *Fourteenth National Conference on Artificial Intelligence*, 598–603. AAAI Press/MIT Press.

Chomsky, N. 1957. *Syntactic Structures.* The Hague: Mouton.

Chomsky, N. 1959. "Review of Skinner's *Verbal Behavior*." *Language* 35: 26–58.

Chomsky, N. 1965. *Aspects of the Theory of Syntax.* Cambridge, Massachusetts: MIT Press.

Chomsky, N. 1986. *Knowledge of Language: Its nature, origin, and use.* New York: Praeger Publishers.

Chomsky, N. 1995. *The Minimalist Program.* Cambridge, Massachusetts: MIT Press.

Collins, M. 1998. *Head-driven statistical models for natural language parsing.* PhD dissertation, University of Pennsylvania.

Cornell, T. 1997. "A type-logical perspective on minimalist derivations." G.-J. van Kruijff & R. Oehrle (eds.), *Formal Grammar '97.* Aix-en-Provence.

Cortes, C. & V. N. Vapnik. 1995. "Support vector networks." *Machine Learning* 20: 273–297.

Cover, T. M. & J. A. Thomas. 1991. *Elements of information theory.* New York: John Wiley.
Dalrymple, M., S. M. Shieber, & F. Pereira. 1991. "Ellipsis and higher-order unification." *Linguistics and Philosophy* 14: 399–452.
Della Pietra, S. A., V. J. Della Pietra, & J. D. Lafferty. 1997. "Inducing features of random fields." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.4: 380–393.
Dempster, A. P., N. M. Laird, & D. B. Rubin. 1977. "Maximum likelihood from incomplete data via the em EM algorithm." *Journal of the Royal Statistical Society* 39.1: 1–38.
Freund, Y. 1998. "Self bounding learning algorithms." *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, 247–258. ACM.
Freund, Y. & R. Schapire. 1997. "A decision-theoretic generalization of on-line learning and an application to boosting." *Journal of Computer and System Sciences* 55.1: 119–139.
Gale, W. A., K. W. Church, & D. Yarowsky. 1992. "One sense per discourse." *Proceedings of the 4th DARPA Speech and Natural Language Workshop*, 233–237). San Francisco, California: Morgan Kaufmann.
Good, I. J. 1953. "The population frequencies of species and the estimation of population parameters." *Biometrika* 40.3/4: 237–264.
Harris, Z. S. 1951. *Structural Linguistics.* Chicago, Illinois: University of Chicago Press.
Harris, Z. S. 1965. *String Analysis of Sentence Structure.* The Hague, Netherlands: Mouton & Co.
Harris, Z. S. 1988. *Language and Information.* New York: Columbia University Press.
Harris, Z. S. 1991. *A Theory of Language and Information: A mathematical approach.* New York: Clarendon Press.
Horning, J. J. 1969. *A Study of Grammatical Inference.* PhD dissertation, Stanford University.
Jelinek, F. & R. L. Mercer. 1980. "Interpolated estimation of Markov source parameters from sparse data." *Proceedings of the Workshop on Pattern Recognition in Practice.* Amsterdam: North Holland.
Katz, S. M. 1987. "Estimation of probabilities from sparse data for the language model component of a speech recognizer." *IEEE Transactions on Acoustics, Speech and Signal Processing* ASSP-35.3: 400–401.
Kearns, M. J. & L. G. Valiant. 1994. "Cryptographic limitations on learning boolean formulae and finite automata." *Journal of the ACM* 41.1: 67–95.
Lambek, J. 1958. "The mathematics of sentence structure." *American Mathematical Monthly* 65: 154–170.
Littlestone, N. & M. Warmuth. 1994. "The weighted majority algorithm." *Information and Computation* 108: 212–261.
McAllester, D. A. 1999. "PAC-Bayesian model averaging." *Proceedings of Twelfth Annual Conference on Computational Learning Theory*, 164–170. New York: ACM Press.
Moortgat, M. 1995. "Multimodal linguistic inference." *Bulletin of the Interest Group in Pure and Applied Logics* 3.2/3: 371–401.
Morrill, G. V. 1994. *Type Logical Grammar: Categorial logic of signs.* Dordrecht, Holland: Kluwer Academic Publishers.
Pollard, C. & I. A. Sag. 1994. *Head-Driven Phrase Structure Grammar.* Chicago: The University of Chicago Press.

Pullum, G. K. 1996. "Learnability, hyperlearning, and the poverty of the stimulus." In J. Johnson, M. L. Juge & J. L. Moxley (eds.), *Proceedings of the 22nd Annual Meeting*: General Session and Parasession on the Role of Learnability in Grammatical Theory, 498–513. Berkeley, California: Berkeley Linguistics Society.

Ratnaparkhi, A. 1997. "Learning to parse natural language with maximum entropy models." *Machine Learning*, 34: 151–175.

Rissanen, J. 1989. *Stochastic Complexity in Statistical Inquiry.* Singapore; Teaneck, New Jersey: World Scientific.

Rosenfeld, R. 2000. "Incorporating linguistic structure into statistical language models." *Philosophical Transactions of the Royal Society* Series A 358.1769: 1311–1324.

Roy, D. & A. Pentland. 1999. "Learning words from natural audio-visual input." *International Conference on Spoken Language Processing* (Vol. 4, 1279–1283). Sidney, Australia.

Salton, G. 1989. *Automatic Text Processing: The transformation, analysis and retrieval of information by computer.* Reading, Massachusetts: Addison-Wesley.

Saul, L. & F. Pereira. 1997. "Aggregate and mixed-order Markov models for statistical language processing." In C. Cardie & R. Weischedel (eds.), *Proceedings of the second conference on empirical methods in natural language processing*, 81–89. Association for Computational Linguistics, Somerset, NJ. Distributed by Morgan Kaufmann, San Francisco, CA.

Schütze, H. 1997. *Ambiguity Resolution in Language Learning: Computational and cognitive models.* Stanford, California: CSLI Publications.

Siskind, J. M. 1996. "A computational study of cross-situational techniques for learning word-to-meaning mappings." *Cognition* 61: 39–91.

Solomonoff, R. J. 1964. "A formal theory of inductive inference." *Information and Control* 7: 1–22, 224–254.

Stabler, E. 1997. "Derivational minimalism." In C. Retoré (ed.), *Logical Aspects of Computational Linguistics*, 68–95. Berlin & New York: Springer-Verlag.

Tishby, N. & A. Gorin. 1994. "Algebraic learning of statistical associations." *Computer Speech and Language* 8.1: 51–78.

Tishby, N., F. Pereira, & W. Bialek. 1999. "Extracting relevant bits: The information bottleneck method." In B. Hajek & R. S. Sreenivas (eds.), *Proceedings of the 37th Allerton Conference on Communication, Control and Computing.* Urbana, Illinois.

Valiant, L. G. 1984. "A theory of the learnable." *Communications of the ACM* 27.11:1134–1142.

Vapnik, V. N. 1995. *The Nature of Statistical Learning Theory.* New York: Springer-Verlag.

Vapnik, V. N. & A. Y. Chervonenkis. 1971. "On the uniform convergence of relative frequencies of events to their probabilities." *Theory of Probability and its Applications* 16.1: 264–280.

Willems, F., Y. Shtarkov & T. Tjalkens. 1995. "The context tree weighting method: Basic properties." *IEEE Transactions on Information Theory* 41.3: 653–664.

Yarowsky, D. 1995. "Unsupervised word sense disambiguation rivaling supervised methods." *33rd Annual Meeting of the Association for Computational Linguistics*, 189–196. Association for Computational Linguistics.

Chapter 2

Logics for intercalation

Richard T. Oehrle
YY Technologies

One of Zellig Harris's signal contributions to linguistics was to focus theoretical scrutiny on challenging properties of natural languages. A central example is his recognition of the importance of Semitic morphology to theoretical accounts of language structure. While not surprising, given Harris's connections with the rich tradition of Semitic philology, Harris's focus on Semitic morphology presaged the influential role that Semitic morphology has assumed and continues to play in formal theories of language.

In Harris's early paper 'Linguistic Structure of Hebrew' (Harris 1941: 156f.), we find such examples as *la-k͟túb* 'to write', *kat͟bí·* 'write! f. sg.', *ko·t̲é·b̲* 'writing', *ko·t̲á͟bt* 'writing f.', related by the consistent presence of the consonantal sequence *ktb* (abstracting away from automatically conditioned variants such as *t*/*t̲*). Drawing on the long and notable history of Hebrew grammatical scholarship, Harris distinguishes three sorts of morphemes (Harris 1941: 152):

> In Hebrew there are three types of phoneme sequences which make up morphemes. Some morphemes, called roots, consist of consonants only: *šbr* '(relating to) break(ing).' Others, called patterns, consist of vowels, vowels plus ·, or vowels plus an affix (a consonant at beginning or end): _ _*u* _ [footnote omitted; see below] 'active action,' _*a*_*i*·_ 'object having a particular quality,' _*a*_ · *a*_ 'transitive intensive action,' *n*_*a*_*i*_ 'middle action.' Still other morphemes are successions of consonants and vowels: *mí·* 'who?,' _*tím* 'you m. pl.,' *balt'i·* 'without.'

In the footnote, Harris states: 'The dash _ indicates the presence of some phoneme, usually a consonant, in close juncture.' But how should the relation between root and pattern be formally characterized? Harris says that roots are 'intercalated in patterns' (Harris 1941: 160). I don't find in Harris's paper an explicit statement — an algorithm — characterizing the notion 'intercalation'. And while it may seem clear enough what the intended result is in particular cases, it is possible to characterize the intercalative combination involved in

such cases in fundamentally different ways. The goal of the present paper is to explore this question from the type logical perspective of multi-modal categorial grammar.

At first glance, it may seem that categorial grammar is an unlikely framework in which to study intercalative morphology of the kind found in Semitic. Indeed, classical categorial grammar, as described for example by Ajdukiewicz (1935) and Bar-Hillel (1953), allows only a single mode of combination: concatenation. And intercalative morphology is beyond the reach of any system of categorial grammar based solely on concatenation. But categorial grammars need not in principle be restricted in this way. To explain this remark, we shall first provide some background material on the structure of categorial grammars. We shall then examine a number of ways in which non-concatenative modes of combining grammatical resources can be introduced in such systems, with the goal of finding a way of modeling intercalative morphology compatible with the categorial setting.

1. Multi-modal categorial grammar

From the point of view adopted here (Morrill 1994, Moortgat 1997, Oehrle 2002), categorial grammar provides a deductive setting in which to investigate **grammatical composition**, the relation that holds (relative to a formal or cognitive system of grammatical analysis G) between the global properties associated with a linguistic entity and the correlative properties associated with its component parts. We think of the expression as a whole as being built up by various *modes of composition* from a set of atomic components, each assumed to be associated with a grammatical category (or type), possibly labeled with terms representing properties in various linguistic *dimensions* — phonological or semantic properties, for example. If T is a type and ϕ and σ represent phonological and semantic terms appropriate to T, we may label the type with these terms, writing $\phi:\sigma:T$. If E is an expression built up from atomic components $w_1, \ldots, w_k$ and we represent the compositional structure of E by $C(w_1, \ldots, w_k)$, then we represent the claim that E can be assigned to the labeled type $\phi:\sigma:T$ by the *sequent*

$$C(w_1, \ldots, w_k) \rightarrow \phi:\sigma:T$$

Whether we regard such a sequent as valid in a particular system of categorial deduction depends on a variety of factors: the meaning of the deducibility

relation $\rightarrow$, the properties of the modes of composition involved in the structured antecedent configuration, general logical properties of the type-constructors corresponding to these modes. We briefly describe the contribution of each of these factors.

1.1 Sequent and inference

If we interpret the deducibility relation $\rightarrow$ as meaning that any instance of the antecedent configuration is an instance of the succedent type, then we see at once that certain sequents should count as valid.

First, any sequent whose antecedent configuration coincides with the succedent type should be valid. This is the Identity Axiom. Suppressing labels for simplicity, it takes the form:

$$\frac{}{A \longrightarrow A}\ \textit{Identity}$$

Second, if $C[A]$ is a structured configuration involving a distinguished occurrence of type A and $\Gamma \longrightarrow A$ is a valid sequent, then the structured configuration Γ is an instance of A and the result of replacing the distinguished occurrence of A in $C[A]$ by Γ, which we denote by $C[\Gamma]$, is an instance of $C[A]$. In other words, the following inference figure, which allows us to pass from valid premisses of the form given above the line to the conclusion below the line, is sound.

$$\frac{\Gamma \longrightarrow A \quad C[A] \longrightarrow B}{C[\Gamma] \longrightarrow B}\ \textit{Cut}$$

1.2 Types and inference

Types correspond to modes of composition. We started above with the assumption that linguistic objects can be put together by a family of different 'modes of composition'. This has long been a standard assumption in linguistics: one thinks of the different modes of morphological combination signaled by the variety of boundary symbols in the phonological system of *The Sound Pattern of English* (Chomsky & Halle 1968), or the distinct roles played in some theories of $\overline{\text{X}}$-theory by 'spec-head' combination versus 'head-complement' combination. Thus, the idea of multiple modes of combination is linguistically familiar in concrete cases. Our approach is more abstract. We assume first that modes can be of different arities: a mode of arity 1 acts on a

single object (example: an intonational contour assigned to an expression), a mode of arity 2 acts on a pair of objects (example: concatenation), . . ., a mode of arity *n* acts on a sequence of *n* objects.

Now, let $\mathcal{A}$ be a set of atomic types (or categories). Then, for each distinct mode of compositon of arity *k*, we introduce a product type-constructor of arity *k*. We use the symbol $\diamond$, possibly with subscripts, to denote unary products, and symbol $\bullet$ to generically denote binary products. We further assume that for each product type-constructor there is a family of corresponding implicational type-constructors — a single implication $\Box$ corresponding to a unary product, a pair of implications /,\ corresponding to a binary product, etc. — and, crucially, that each implication is adjoint to the product it corresponds to. What this means for a unary product/implication pair is the following:

$$\diamond A \longrightarrow B \quad \textit{if and only if} \quad A \longrightarrow \Box B$$

That is, if the sequent on the left is valid, so is the sequent on the right, and vice versa.

For binary products, there are apparently two cases:

$$\begin{array}{ll} C \bullet A \longrightarrow B & \textit{if and only if} \quad A \longrightarrow C \backslash B \\ C \bullet A \longrightarrow B & \textit{if and only if} \quad C \longrightarrow B / A \end{array}$$

But it is easy to see that we can regard both of these as instances of the unary adjointness relation above. In the first case, for example, applying the binary type constructor $\bullet$ to its left argument *C* yields a unary operator $C \bullet -$. We can think of this unary operator as a composite $\diamond$. Playing the role of the corresponding composite $\Box$ is the unary operator $C \backslash -$. From this point of view, then, we see that this binary case reduces to the unary case.

$$\begin{array}{rll} C \bullet A \longrightarrow B & \textit{if and only if} & A \longrightarrow C \backslash B \\ \boxed{C \bullet} A \longrightarrow B & \textit{if and only if} & A \longrightarrow \boxed{C \backslash} B \\ \diamond A \longrightarrow B & \textit{if and only if} & A \longrightarrow \Box B \end{array}$$

The second binary case is the same, taking $- \bullet A$ as the composite $\diamond$ and $-/A$ as the corresponding composite $\Box$. The same technique works for products and their implications corresponding to modes of composition of higher arity.

Now, we already have $C \backslash B \longrightarrow C \backslash B$ and $B/A \longrightarrow B/A$ by *Identity*. Taking these on the right hand side of the binary adjointness laws, we have as the left

$$C \bullet C \backslash B \longrightarrow B \quad \textit{and} \quad B/A \bullet A \longrightarrow B$$

These are the *modus ponens* rules for the directionally sensitive implications \ and /. Starting with the identity $\Box B \longrightarrow \Box B$ yields the unary analogue of *modus ponens*:

$$\Diamond \Box B \longrightarrow B$$

The adjointness laws may be formulated within a deductive system in remarkably many ways. Here we use the sequent formulation of Natural Deduction. Along with *Identity* and *Cut*, we have, for each type-constructor, rules governing its introduction and elimination in the sequent succedent. Some fundamental cases are exhibited below. We use upper-case Greek letters for structured antecedent configurations, '$\circ$' for the mode of composition corresponding to $\bullet$, and $\langle - \rangle$ for the mode of composition corresponding to $\Diamond$:

unary product:
$$\frac{\Gamma \longrightarrow A}{\langle \Gamma \rangle \longrightarrow \Diamond A} \Diamond I \qquad \frac{\Gamma \longrightarrow \Diamond B \quad \Delta[\langle B \rangle] \longrightarrow A}{\Delta[\Gamma] \longrightarrow A} \Diamond E$$

unary adjoint:
$$\frac{\langle \Gamma \rangle \longrightarrow B}{\Gamma \longrightarrow \Box A} \Box I \qquad \frac{\Gamma \longrightarrow \Box B}{\langle \Gamma \rangle \longrightarrow B} \Box E$$

binary product:
$$\frac{\Gamma \longrightarrow A \quad \Delta \longrightarrow B}{\Gamma \circ \Delta \longrightarrow A \bullet B} \bullet I \qquad \frac{\Gamma \longrightarrow A \bullet B \quad \Delta[A \circ B] \longrightarrow C}{\Delta[\Gamma] \longrightarrow C} \bullet E$$

binary adjoints:
$$\frac{\Gamma \circ B \longrightarrow A}{\Gamma \longrightarrow A / B} / I \qquad \frac{\Gamma \longrightarrow A / B \quad \Delta \longrightarrow B}{\Gamma \circ \Delta \longrightarrow A} / E$$

$$\frac{\Gamma \longrightarrow A \backslash B}{A \circ \Gamma \longrightarrow B} \backslash I \qquad \frac{\Gamma \longrightarrow A \quad \Delta \longrightarrow A \backslash B}{\Gamma \circ \Delta \longrightarrow B} \backslash E$$

It is not difficult to see that if we make the notational change of replacing the symbols for modes of composition '$\circ$' and '$\langle - \rangle$' by the symbols for the corresponding products $\bullet$ and $\Diamond$, that these rules are interderivable with the adjointness rules as stated earlier. They yield many further consequences as well.

As an example, take the sentence *every student seems to have been there*. If we assign *every student* the quantifier type $s/(np\backslash s)$ and assume that *seems to have been there* has type $(s/(np\backslash s))\backslash s$ — that is, takes a quantifier to its left to make a sentence — then the basic logic characterized above provides two proofs:

PROOF 1.

$$\dfrac{\text{every student}\longrightarrow s/(np\backslash s) \qquad \text{seems-to-have-been-there}\longrightarrow (s/(np\backslash s))\backslash s}{\text{every student} \circ \text{seems-to-have-been-there}\longrightarrow s}\ \backslash E$$

PROOF 2.

$$\dfrac{\text{every student}\longrightarrow s/(np\backslash s) \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{np\longrightarrow np}\,Iden \quad \dfrac{}{np\backslash s\longrightarrow np\backslash s}\,Iden}{np \circ np\backslash s\longrightarrow s}\,\backslash E}{np\longrightarrow s/(np\backslash s)}\,/I \qquad \text{seems-. . .-there}\longrightarrow (s/(np\backslash s)}{np \circ \text{seems-. . .- there}\longrightarrow s}\,\backslash E}{\text{seems-. . .-there}\longrightarrow np\backslash s}\,\backslash I}{\text{every student} \circ \text{seems-to-have-been-there}\longrightarrow s}\ /E$$

To bring out the distinctness of these two proofs, we may turn our inference system into a system of labeled deduction, in line with the Curry-Howard correspondence. We assume each type is labeled with a λ-term representing its interpretation. We then expand each proof rule in a way that makes explicit how the λ-term associated with the active types of the conclusion can be specified in terms of the λ-terms of the active types of the premisses. For the rules of • Introduction and Elimination, we use pairing and projection. The rules $/E$ and $\backslash E$ are associated with application: the term of the major premise (that is, the one containing the occurrence of / or \ removed in the rule) is applied to the term of the minor premise. Correlatively, the rules $/I$ and $\backslash I$ are associated with λ-abstraction. Labeling *every student* with the λ-term $\lambda P.\forall x\ (student\ (x\ ) \Rightarrow P\ (x\ ))$ and *seems to have been there* with the λ-term $\lambda \mathbf{Q}.(seem\ (\mathbf{Q}(\lambda z.have\text{-}been\text{-}there\ (z))))$, the Curry-Howard correspondence between proofs and λ-terms associates *Proof 1* above with a term that normalizes to

$$seem(\forall x(student(x) \Rightarrow have\text{-}been\text{-}there(x)))$$

The term associated with *Proof 2* normalizes to

$$\forall x(student(x) \Rightarrow seem(have\text{-}been\text{-}there(x)))$$

These terms correspond to intuitively different interpretations of the sentence whose compositional structure we are attempting to model. Interestingly enough, the system obtains both readings relative to unambigous lexical assumptions and in the absence of any special rules of scoping.

Although the deductive system just described is rich in consequences, it is quite restricted in comparison to some closely related systems of propositional logic. For example, the classical/intuitionistic conjunction operator $\wedge$ can be formulated using exactly the logical rules that we have used for the binary type-constructor $\bullet$. Yet conjunction satisfies many properties whose analog for $\bullet$ are not provable in the system described above. In the representative examples below, we write $\Gamma \not\longrightarrow A$ to indicate that the sequent $\Gamma \longrightarrow A$ is not provable.

	$\wedge$	$\bullet$
Commutativity	$A \wedge B \longrightarrow B \wedge A$	$A \bullet B \not\longrightarrow B \bullet A$
RAssociativity	$(A \wedge B) \wedge C \longrightarrow A \wedge (B \wedge C)$	$(A \bullet B) \bullet C \not\longrightarrow A \bullet (B \bullet C)$
Idempotence	$A \longrightarrow A \wedge A$	$A \not\longrightarrow A \bullet A$
Projection	$A \wedge B \longrightarrow A$	$A \bullet B \not\longrightarrow A$

If the logical rules for $\wedge$ and $\bullet$ are the same, there must be other factors which affect the provability of sequents. Gentzen (1934), who introduced this division of labor, called the other factors 'structural rules'.

1.3 Modes and inference

Different modes of the same arity may have different structural properties. And these are inherited by the corresponding type-constructors. For example, if the mode $\circ$ corresponding to the type-constructor $\bullet$ is commutative, then we can prove $A \circ B \longrightarrow B \bullet A$. If we wish to take $\circ$ to be commutative, we add the inference rule:

$$\frac{\Gamma[B \circ A] \longrightarrow C}{\Gamma[A \circ B] \longrightarrow C}\ \mathit{Com}$$

With this rule available, we have a proof of $A \circ B \longrightarrow B \bullet A$:

$$\dfrac{\dfrac{\dfrac{}{B \longrightarrow B}\ \mathit{Iden} \quad \dfrac{}{A \longrightarrow A}\ \mathit{Iden}}{A \circ B \longrightarrow B \bullet A}\ \bullet I}{B \circ A \longrightarrow B \bullet A}\ \mathit{Com}$$

For associativity, there are two rules, one in each direction:

$$\frac{\Gamma[(A \circ (B \circ C))\ ] \longrightarrow D}{\Gamma[((A \circ B) \circ C)\ ] \longrightarrow D}\ \mathit{RAssoc} \qquad \frac{\Gamma[((A \circ B) \circ C)] \longrightarrow D}{\Gamma[(A \circ (B \circ C))] \longrightarrow D}\ \mathit{LAssoc}$$

Similarly, to prove idempotence in the direction A⟶A • A, we may add the Contraction rule, which in effect allows a premise to be deductively used more than once:

$$\frac{\Gamma[(A \circ A)] \longrightarrow B}{\Gamma[A] \longrightarrow B}\ Con$$

One may prove idempotence in the other direction by adding the inverse of the Contraction rule or its generalization, Weakening:

$$\frac{\Gamma[A] \longrightarrow C}{\Gamma[(A \circ B)] \longrightarrow C}\ Weak$$

There is a second form of Weakening, which introduces the new element as the left argument of ∘, though in the presence of Commutativity, this second form is a direct consequence of the first.

Since the mode ∘ plays a prominent role in the logical rules for the implicational connectives as well as in the logical rules for the corresponding product connective •, its properties affect the logical powers of the implicational connectives as well. For example, in the presence of commutativity, we have $A/B \longrightarrow B\backslash A$ and its converse. In the presence of the associativity rules, we have first-order composition $(A/B) \circ (B/C) \longrightarrow A/C$ and the recursive form of composition represented by the 'division arrow' $A/B \longrightarrow (A/C)/(B/C)$.

Stepping back, we see that these structural principles represent independent and fundamental structural principles — Order, Grouping, Multiplicity. To a great extent these themes are formally independent of one another, and when they are, different subsets of them determine different logical systems. The conjunctive-implicational fragment of Intuitionistic Logic enjoys all the structural rules mentioned above (and Classical Logic gives them greater scope by allowing them freer play in the succedent space); Relevant Logic omits Weakening; the multiplicative fragment of Linear Logic omits both Weakening and Contraction; the associative Lambek Calculus allows only the two associativity rules; the non-associative Lambek Calculus allows none of these structural rules. This hierarchy of logical systems is built up from the parameters that are the focus of the structural rules.

Yet moving from one system to another represents an absolute decision: one accepts or rejects totally the power of a given structural option and each step increases deductive power, yielding more provable sequents, but at the same time, bringing in its wake a loss in structural discrimination. Natural languages are typically more selective and subtle in how they provide access to

such structural manipulations. We can attain a better model of natural language composition by moving to a system that allows controlled access to structural reasoning.

In fact, the multi-modal system that we have described above already instantiates these possibilities. Different modes, both binary and unary, live compatibly in a single system. Different modes may have distinct properties, of course. But bringing a variety of different modes of composition into a single system does more than just collapse different single-mode systems together: new possibilities arise through the way these distinct modes interact. In the next section, we give an example focusing on forms of distributivity. This will lead us directly to the study of intercalation.

2. Controlled discontinuity

Suppose we start with the system **NL**, the single-mode system with one binary mode $\circ$, its corresponding product $\bullet$ and implications / and \. Let's introduce some lexical assumptions that will allow us to build up simple sentences with an intransitive verb (*whistled*), with a transitive verb (*phoned*), and with some more complicated verbal constructions involving prepositions and embedded sentences:

Kim$\longrightarrow np$	Sandy$\longrightarrow np$	Maria$\longrightarrow np$	Mario$\longrightarrow np$
whistled$\longrightarrow np\backslash s$	phoned$\longrightarrow (np\backslash s)/np$	to$\longrightarrow tpp/np$	about$\longrightarrow abpp/np$
introduced$\longrightarrow ((np\backslash s)/tpp)/np$		talked$\longrightarrow ((np\backslash s)/abpp)/tpp$	
announced$\longrightarrow (np\backslash s)\bar{s}$		that$\longrightarrow \bar{s}/s$	

We wish to extend this by adding constructions with simple embedded questions (type: *eq*). First, let *wondered* be assigned to type $(np\backslash s)/eq$. Our next step is to find a type or a finite set of types for *who* so that such sentences as those in the table below are derivable. For each case, we give a type for *who* which would suffice.

Kim wondered . . .	
who whistled	who $\longrightarrow eq/(np\backslash s)$
who Sandy phoned	who $\longrightarrow eq/(np\bullet(np\backslash s)/np)$
who Sandy introduced Maria to	who $\longrightarrow eq/(np\bullet(((np\backslash s)/tpp\bullet tpp/np))$
who Sandy introduced to Maria	who $\longrightarrow eq/(np\bullet((((np\backslash s)/tpp)/np)\bullet tpp))$
who Sandy talked to about Mario	who $\longrightarrow eq/(np\bullet((((np\backslash s)/abp)/tpp)\bullet (tpp/np)\bullet abpp)$

who Sandy announced that Maria phoned . . .

$$\begin{array}{cc} \vdots & \vdots \end{array}$$

If *who* is assigned all of the types in the righthand column, then the sentences in the lefthand column are all derivable. Conversely, if the sentences in the lefthand column are derivable in this system, *who* is assignable to all the types in the righthand column. But this set of type assignments cannot be a lexical matter. Since this system is recursive, the table can be made as large as we please. Thus, since we assume that lexical type assignment associates each lexical element with a finite set of types, we must find a non-lexical means of associating *who* with each of the types in the righthand column.

One possibility would be to admit the structural rules of Associativity and Commutativity. Then all the types are derivable from the single lexical assumption $who \longrightarrow eq/(np\backslash s)$. Here is one case, for example. (The proof is split typographically into two parts, connected by the vertical ellipsis marks, so that the upper subproof attaches to the lower to form the upper right subproof of the whole.)

$$\dfrac{\dfrac{}{np \longrightarrow np}\,Iden \qquad \dfrac{\dfrac{}{np\backslash s \longrightarrow np\backslash s}\,Iden \quad \dfrac{}{np \longrightarrow np}\,Iden}{((np\backslash s)/np) \circ np \longrightarrow np\backslash s}\,/E}{\dfrac{\dfrac{\dfrac{\dfrac{np \circ (((np\backslash s)/np) \circ np) \longrightarrow s}{(np \circ np((np\backslash s)/np)) \circ np \longrightarrow s}\,Assoc!}{np \circ (np \circ ((np\backslash s)/np)) \longrightarrow s}\,Com!}{np \circ ((np\backslash s)/np \longrightarrow np\backslash s}\,\backslash I}{\vdots}\,\backslash E$$

$$\dfrac{\dfrac{}{eq \longrightarrow eq}\,Iden \qquad \dfrac{\dfrac{}{(np \bullet (np\backslash s)/np) \longrightarrow (np \bullet (np\backslash s)/np)}\,Iden \quad \vdots}{(np \bullet (np\backslash s)/np) \longrightarrow np\backslash s}\,\bullet E}{\dfrac{eq/(np\backslash s) \circ (np \bullet (np\backslash s)/np) \longrightarrow eq}{eq/(np\backslash s) \longrightarrow eq/(np \bullet (np\backslash s)/np)}\,/I}\,/E$$

But adding Commutativity and Associativity to derive the cases we wish to derive makes inevitable the derivability of many cases we wish not to derive. In particular, we can no longer grammatically distinguish among an expression and any of its permutations.

This difficulty can be easily circumvented, however. It isn't that we wish to be able to change the relative order and grouping of every expression — the only type that needs properties of this kind is the hypothetical *np* argument introduced in the type $eq/(np\backslash s)$. We can achieve this in two steps:

1. replace *np* in $eq/(np\backslash s)$ by the modally-decorated $\Diamond_x\Box_x np$, so that we have the lexical type assignment:

$$\text{who}\longrightarrow eq/((\Diamond_x\Box_x np)\backslash s)$$

2. introduce a recursive package of modally-controlled structural rules which allow the modally-decorated resource to find the 'missing *np*' position:

$$\begin{array}{ll} \Diamond_x A\bullet(B\bullet C)\longrightarrow B\bullet(C\bullet\Diamond_x A) & \text{P0}\\ (B\bullet C)\bullet\Diamond_x A\longrightarrow(B\bullet\Diamond_x A)\bullet \mathrm{C} & \text{P1}\\ (B\bullet C)\bullet\Diamond_x A\longrightarrow B\bullet(C\bullet\Diamond_x A) & \text{P2}\end{array}$$

Now, given these rather fine-grained, modally-controlled forms of commutativity and associativity, a proof of *who* $\circ\,\phi[\ ]\longrightarrow eq$, with the brackets in $\phi[\ ]$ indicating schematically the position of the 'missing *np*', will take the following form:

$$\dfrac{\vdots \atop \phi[np]\longrightarrow \mathrm{s}}{\phi[\Diamond_x\Box_x np]\longrightarrow \mathrm{s}}\;\textit{unary MP}$$

$$\boxed{\text{zero or more applications of P0, P1, P2}}$$

$$\dfrac{\dfrac{}{\text{who}\circ\phi[\ ]\longrightarrow eq}\;lex \qquad \dfrac{\Diamond_x\Box_x\, np\circ\phi[\ ]\longrightarrow s}{\phi[\ ]\longrightarrow(\Diamond_x\Box_x\, np)\backslash s}\;\backslash I}{\text{who}\longrightarrow eq/((\Diamond_x\Box_x\, np)\backslash s)}\;\backslash E$$

All the examples of embedded questions given above are derivable in this way. Example:

$$\dfrac{\dfrac{}{\text{who}\longrightarrow eq/((\Diamond_x\Box_x\, np)\backslash s)}\;\text{lex} \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{sandy}\longrightarrow np \qquad \dfrac{\text{phoned}\longrightarrow(np\backslash \mathrm{s})/np \quad np\longrightarrow np}{\text{phoned}\circ \mathrm{np}\longrightarrow np\backslash s}\;/E}{\text{Sandy}\circ(\text{phoned}\circ np)\longrightarrow s}\;\backslash E}{\text{Sandy}\circ(\text{phoned}\circ\Diamond_x\Box_x np)\longrightarrow s}\;\textit{unary MP}}{(\text{Sandy}\circ\text{phoned})\circ\Diamond_x\Box_x np\longrightarrow s}\;P2}{\Diamond_x\Box_x\, np\circ(\text{Sandy}\circ\text{phoned})\longrightarrow s}\;P0}{\text{Sandy}\circ\text{phoned}\longrightarrow(\Diamond_x\Box_x\, np)\backslash s}\;\backslash I}{\text{who}\circ(\text{Sandy}\circ\text{phoned})\longrightarrow eq}\;/E$$

It is worth emphasizing the role of the modal operator $\diamond_x$ in the postulate rules $P0$, $P1$, $P2$. The $\diamond_x$ decoration gives a modally-decorated type of the form $\diamond_x A$ a passport to cross a limited set of boundaries. And if it likes a certain spot, a type of the form $\diamond_x\Box_x A$ can throw away its passport, using the unary Modus Ponens rule $\diamond_x\Box_x A \longrightarrow$A, and continue life in its new location as a citizen of type A. Modally-undecorated types, such as *np*, are not entitled to this passport and thus, in the ordinary course of events, are not entitled to travel. This sorting of types on purely logical grounds is an example of how the multi-modal setting allows controlled structural reasoning. It arises as soon as we have a single binary mode and a single unary mode living together in a single system, but is beyond the reach of any single-mode binary system. Let us now turn to the study of intercalation, where we will find once again an example of controlled structural reasoning.

3. Distributivity

3.1 Weak versus strong

The rules $P0$, $P1$, $P2$ of the postulate package introduced above are examples of *weak distributivity*. Rules of weak distributivity are linear in their treatment of resources: input types and output types stand in a one-to-one correspondence. Rules of *strong distributivity*, such as the familiar rule

$$x \cdot (y + z) = (x \cdot y) + (x \cdot z) \quad (Dist)$$

connecting multiplication and addition are not linear (in this terminology!), since we have two occurrences of x on one side and only a single occurrence of x on the other. This basic distinction in resource management connects directly with linguistic properties of intercalation, as we shall see below.

3.2 Distributivity and recursion

Another interesting aspect of the arithmetical rule distributing multiplication over addition is its recursive character. On the right side of the distributivity rule *Dist* above, if $y = y_1 + y_2$ or $z = z_1 + z_2$, then *Dist* is again applicable. Thus, this rule is recursive in the sense that its application may produce environments in which it is again applicable.

Rules of distributivity need not be recursive. For example, it is possible to

write a single distributive rule which will combine a list of three elements and a list of two elements to produce a list of five elements:

$$(A_1 \bullet (A_2 \bullet A_3)) \odot (B_1 \bullet B_2) \longrightarrow A_1 \bullet (B_1 \bullet (A_2 \bullet (B_2 \bullet A_3)))$$

This is a rule of very limited scope: it only treats a single pattern and it only gives rise to a single case. For ranges of data involving a complex combination of patterns, many rules of this kind are required — one for each identifiable pattern of combination. We would like to see what kinds of complexity arise when structural postulates introduce dynamic interactions, rather than simply providing static correspondences. *A* good place to start is with simple cases of recursive distributivity.

3.3 Mapcar

Mapcar is a LISP functor which takes a function f and a list $m_1 \bullet (m_2 \bullet (\ldots \bullet (m_{n-1} \bullet m_n) \ldots)$ and returns the list

$$(f \bullet_a m_1) \bullet ((f \bullet_a m_2) \bullet (\ldots \bullet ((f \bullet_a m_{n-1}) \bullet (f \bullet_a m_n)) \ldots))$$

in which f is applied to each member of the list (with application of f to m written $f \bullet_a m$). This can be characterized recursively by a rule of strong distributivity:

$$A \bullet_a (B \bullet C) \longrightarrow (A \bullet_a B) \bullet (A \bullet_a C)$$

3.4 Mapshuffle

Mapshuffle is a simple variant on on *mapcar*. Suppose that instead of starting with a single function and a list of arguments, one starts with two lists of the same length: a list of functions and a list of arguments. We want to define a notion of 'application' which works by applying the first element of the first list to the first element of the second list, the second element of the first list to the second element of the second list, etc. Thus, we want the following:

$$\begin{array}{rcll} f \bullet_a m & \longrightarrow & f \bullet_a m & \text{GIVEN!} \\ (f_1 \bullet_l f_2) \bullet_a (m_1 \bullet_l m_2) & \longrightarrow & (f_1 \bullet_a m_1) \bullet_l (f_2 \bullet_a m_2) & \\ (f_1 \bullet_l (f_2 \bullet_l f_3)) \bullet_a (m_1 \bullet_l (m_2 \bullet_l m_3)) & \longrightarrow & ((f_1 \bullet_a m_1) \bullet_l ((f_2 \bullet_a m_2) \bullet_l (f_3 \bullet_a m_3))) & \\ \vdots & & \vdots & \end{array}$$

This behavior can be characterized by the recursive, linear distributivity postulate:

$$(A \bullet B) \bullet_a C \bullet D) \longrightarrow (A \bullet_a C) \bullet (B \bullet_a D) \quad M0$$

Now, this step already gives us a form of intercalation. Take the first list to be a list of consonants $(c_1 \bullet (c_2 \bullet c_3))$ and take the second to be a list of vowels $(v_1 \bullet (v_2 \bullet v_3))$. Then with the distributivity postulate above for $\bullet_a$, we have:

$$c_1 \bullet (c_2 \bullet c_3)) \bullet_a (v_1 \bullet (v_2 \bullet v_3)) \longrightarrow (c_1 \bullet_a v_1) \bullet ((c_2 \bullet_a v_2) \bullet (c_3 \bullet v_3))$$

Thus, from a list of 'consonants' and a list of 'vowels', as given on the left, we construct a list of 'syllables' consisting of 'consonants' and 'vowels' intercalated among one another. But this form of intercalation is a very special case: for example, it depends implicitly on the assumption that the two lists to be intercalated are the same length; moreover, it is not flexible enough to deal with patterns of gemination or spreading commonly found in relevant natural language examples.

3.4.1 *Rightward Spreading*

'Spreading' — re-use of a list member — represents an intrusion of non-linearity, requiring non-linear postulates of strong distributivity rather than postulates that are linear in their resource manipulation. Suppose the final element of a list can be re-used as needed. In the programming languages Lisp and Prolog, one recognizes a list containing a single element A by identifying this list with the result of appending A to the empty list *nil* (Lisp) or [ ] (Prolog). We could regard $A \bullet nil$ as a way of placing A inside the scope of a modal operator (written on the right rather than the left), but we will instead make the simplifying assumption that we can recognize the end of a list by encountering a type atom. We use lower-case letters as variables ranging over type atoms; upper-case letters stand for atomic or non-atomic types. On these assumptions, consider the consequences of adding the rules $R1$ and $R2$ below to the rule $M0$ stated above:

$$\begin{array}{ll} a \bullet_a (B \bullet C) \longrightarrow (a \bullet_a B) \bullet (a \bullet_a C) & R1 \\ (A \bullet B) \bullet_a a \longrightarrow (A \bullet_a \mathrm{a}) \bullet (B \bullet_a \mathrm{a}) & R2 \end{array}$$

In terms of consonants and vowels, applying $R1$ to $c_1 \bullet_a (v_1 \bullet v_2)$ yields $(c_1 \bullet_a v_1) \bullet (c_1 \bullet v_2)$: the consonant spreads. $R2$ works symmetrically, taking $(c_1 \bullet c_2) \bullet v_1$ to $(c_1 \bullet_a v_1) \bullet (c_2 \bullet_a v_1)$, spreading the vowel.

3.4.2 *Medial Spreading*

There is an alternative rule package which induces medial spreading:

$$(A \bullet B) \bullet_a C \bullet d) \longrightarrow (A \bullet_a C) \bullet (B \bullet_a C \bullet d)) \qquad R1m$$
$$(A \bullet b) \bullet_a C \bullet D) \longrightarrow (A \bullet_a C) \bullet ((A \bullet b) \bullet_a D) \qquad R2m$$

3.5 Distributivity: summary

By systematically varying resource-sensitive aspects of distributivity, we refract this general process into a large family of special cases. It is far from clear, however, that the mere enumeration of possibilities offers an explanatory framework for the combination of variability and stability found in natural language examples involving intercalation of root and pattern, perhaps because the resource-sensitive parameters of variation are embedded intrinsically into individual postulates. Moreover, a family of special cases can hardly provide a model of Harris's intuitive notion of intercalation, a notion which he intended to be applicable in a general way to a variety of roots — roots containing 2, 3, or 4 consonants (Harris 1941:160) — and clearly distinguishable patterns.

In the next section, we look at the problem from a different perspective — one involving some basic product structures, a set of distributivity postulates, and a control structure which gives recursive access to these postulates in a discriminating way. This mixture of logic with control pursues the same theme we saw earlier in our discussion of English embedded questions.

4. The shuffle product and its variants

A *shuffle* of two lists is a third list whose elements stand in a one-to-one correspondence to the disjoint union of the first two lists, in such a way that the mapping from each of the two input lists to the third is order-preserving. For example, take the input lists to be [*a*,*b*] and [*c*,*d*]. Then each of the lists below is a shuffle of these two inputs:

[*a*, *b*, *c*, *d*], [*a*, *c*, *b*, *d*], [*a*, *c*, *d*, *b*], [*c*, *a*, *b*, *d*], [*c*, *a*, *d*, *b*], [*c*, *d*, *a*, *b*]

But [*b*, *c*, *a*, *d*] is not a shuffle of [*a*, *b*] and [*c*, *d*], because *a* precedes *b* in [*a*, *b*], but follows *b* in [*b*, *c*, *a*, *d*].

Taking lists to be built up by a non-commutative, non-associative binary product $\bullet_l$, we can characterize a shuffle product ※ by the following recursive scheme:

$$\frac{B \longrightarrow B}{A \between B \longrightarrow A \bullet_l B}\ \textit{Shuf0 L} \qquad \frac{B \longrightarrow B}{B \between A \longrightarrow A \bullet_l B}\ \textit{Shuf0 R}$$

$$\frac{B \between C \longrightarrow D}{(A \bullet_l B) \between C \longrightarrow A \bullet_l D}\ \textit{Shuf1 L} \qquad \frac{B \between C \longrightarrow D}{B \between (A \bullet_l C) \longrightarrow A \bullet_l D}\ \textit{Shuf1 R}$$

For example, here is a proof that $c \bullet_l (a \bullet_l (d \bullet_l b))$ is a shuffle of $a \bullet_l b$ and $c \bullet_l d$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{b \longrightarrow b}\ \textit{Iden}}{b \between d \longrightarrow d \bullet_l b}\ \textit{Shuf0 R}}{(a \bullet_l b) \between d \longrightarrow a \bullet_l (d \bullet_l b)}\ \textit{Shuf1 L}}{(a \bullet_l b) \between c \bullet_l d) \longrightarrow c \bullet_l (a \bullet_l (d \bullet_l b))}\ \textit{Shuf1 R}$$

And here is a proof that the Arabic verb *katab* 'write' is a shuffle of the triliteral root *ktb* and the vowel sequence *aa* (suppressing the product symbol $\bullet_l$ for readability):

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathrm{b} \longrightarrow b}\ \textit{Shuf1 L}}{\mathrm{b} \between \mathrm{a} \longrightarrow \mathrm{ab}}\ \textit{Shuf0 R}}{\mathrm{tb} \between \mathrm{a} \longrightarrow \mathit{tab}}\ \textit{Shuf1 L}}{\mathrm{tb} \between \mathrm{aa} \longrightarrow \mathit{atab}}\ \textit{Shuf1 R}}{\mathrm{ktb} \between \mathrm{aa} \longrightarrow \mathit{katab}}\ \textit{Shuf1 L}$$

In a way, it is not too surprising that we can show that *katab* is a shuffle of the triliteral root *ktb* and the vowel sequence *aa*: there are many shuffles of these two: *ktbaa*, *ktaba*, *ktaab*, *katba*, *katab*, *kaatb*, *aktba*, *aktab*, *akatb*, *aaktb*. So simply shuffling root and vowel sequence overgenerates: it gives back 10 cases where we want only one. And for empirical purposes, not only does it give back too much, it gives back too little. As defined by the inference rules above, the shuffle operator is *linear* in resource management. But in Arabic intercalative morphology, there are forms in which vowels or consonants are repeated: in *ktabab* we find two occurrences corresponding to the single *b* of the root *ktb*; in *kattab* we find two occurrences of the single root consonant *t* (or a lengthened instance of *t*); in *katab* itself, we might perceive two occurrences of a single input vowel *a*.

5. A glance at Arabic

John McCarthy, who pioneered the application of the methods of autosegmental phonology to the analysis of such problems (McCarthy 1981), illustrates the situation more completely with a table whose rows consist of the fifteen conjugations or *binyanim* of the Arabic triliteral roots and whose columns represent active and passive representatives of the categories *Perfective*, *Imperfective*, and *Participle*. Below, we reproduce a portion of this table.

	Perfective		Imperfective		Participle	
	Active	Passive	Active	Passive	Active	Passive
Triliteral roots						
I	katab	kutib	aktub	uktab	kaatib	maktuub
II	kattab	kuttib	ukattib	ukattab	mukattib	mukattab
III	kaatab	kuutib	ukaatib	ukaatab	mukaatib	mukaatab
IV	ʔaktab	ʔuktib	uʔaktib	uʔaktab	muʔaktib	muʔaktab
V	takattab	tukuttib	atakattab	utakattab	mutakattib	mutakattab
VI	takaatab	tukuutib	atakaatab	utakaatab	mutakaatib	mutakaatab
⋮	⋮	⋮	⋮	⋮	⋮	⋮
IX	ktabab		aktabib		muktabib	
⋮	⋮	⋮	⋮	⋮	⋮	⋮

All of the forms in this table can be factored into a triliteral root, a vocalic pattern, zero or more affixes, and a mode of combination. We shall show how this mode of combination can be realized as a modally-controlled shuffle product. But we emphasize that our goal is not to focus on the empirical details of the data represented above (already somewhat idealized, as McCarthy emphasizes), but rather to model qualitative aspects of its patterning in a deductive setting.

6. Shuffles with modal control

The deductions in this system will have the basic shape exemplified below, just as in our earlier discussion:

$$\dfrac{\dfrac{\text{ktb} \longrightarrow \textit{3root} \qquad \textit{ui} \longrightarrow \textit{3root} \setminus \mu(I(\textit{perf}, \textit{pass}))}{\vdots}}{\text{kutib} \longrightarrow \Box I(\textit{perf}, \textit{pass})}$$

We use '*3root*' as the name of the category of triliteral roots; '*I(perf,pass)*' is the category for perfective passive members of binyan I; μ stands for a modal prefix whose structure will provide the control of our shuffle; finally, the □ in the conclusion provides a way to launch the proof upward from the end-sequent and to distinguish the type of the completed shuffle from intermediate stages of a shuffle in progress. We need a way to formally distinguish consonants and vowels, and a way to formally distinguish the back vowels *a,u* from the non-back vowel *i*. We will think of consonants for present purposes as labeled types: $\text{k} : c$, $\text{t} : c$, $b : c$. Vowels in general will be *v*, but we will decorate the vowels with a modal feature to distinguish the front vowel *i* from the back vowels, yielding: $\text{i} : \Diamond_{-b} v$, $\text{u} : \Diamond_{b} v$, $\text{a} : \Diamond_{b} v$.

We will assume that sequences of segments are structured by a binary product operator. In premises, products associate to the right; in conclusions, they associate to the left. The shuffle process maps one to the other.

Finally, it will be convenient to make use of the identity element 1 as a place-holder for the empty list. This constant is governed by the postulates below, where lower case variables range over types built up from atoms using the unary operators only:

$$\dfrac{\Gamma \longrightarrow 1 \quad \Delta \longrightarrow a}{\Gamma, \Delta \longrightarrow a}\, 1EL \qquad \dfrac{}{\longrightarrow 1}\, 1I \qquad \dfrac{\Gamma \longrightarrow a \quad \Delta \longrightarrow 1}{\Gamma, \Delta \longrightarrow a}\, 1ER$$

These rules allow one to prove $a \longrightarrow 1 \bullet a$, $(1, a) \longrightarrow a$, $a \longrightarrow a \bullet 1$, and $(a, 1) \longrightarrow a$. Thus we have:

$$\dfrac{(a,1) \longrightarrow \text{a} \quad \Gamma[\text{a}] \longrightarrow B}{\Gamma[(a,1)] \longrightarrow B}\, \textit{Cut}$$

And:

$$\dfrac{\text{a} \longrightarrow 1 \bullet a \quad \Gamma[(1,a)] \longrightarrow B}{\Gamma[a] \longrightarrow B}\, \textit{Cut}$$

As a result, we can replace *a* by $a \bullet 1$ and $1 \bullet a$ by *a*, a possibility that we make use of below.

Now let us see how the schematic proof above can be fleshed out. There are four phases. It is perhaps helpful to examine them in isolation before putting them together into a single deduction. The four phases are:

6.1 Phase 1

The initial phase combines root and vowel as the two arguments of a distinguished binary operator ⁂, our 'shuffle product type constructor'. The combination is assigned a category within the scope of a modal prefix. This phase, which depends only on standard reasoning with the binary operators, has the form:

$$\dfrac{(\mathrm{k}:c,(\mathrm{t}:c,\mathrm{b}:c)) \longrightarrow \mathit{3root} \quad (\mathrm{u}:\Diamond_b v,\mathrm{i}:v) \longrightarrow \mathit{3root} \backslash\!\!\ast (\Box_{sh} \ldots \Box_{sh} I\,(\mathit{perf},\mathit{pass})}{((\mathrm{k}:c,(\mathrm{t}:c,\mathrm{b}:c)) \ast\!\!\ast (\mathrm{u}:\Diamond_b v,\mathrm{i}:v)) \longrightarrow \Box_{sh} \ldots \Box_{sh} I\,(\mathit{perf},\mathit{pass})} \backslash\!\!\ast E$$

$$\vdots$$

6.2 Phase 2

In the second phase of the proof, all the □ operators decorating the succedent type shift successively to the left, as unary modes taking the antecedent structure (successively) in their scope. This again is simply standard unary reasoning, using $\Box E$ repeatedly:

$$\dfrac{\dfrac{((\mathrm{k}:c,(\mathrm{t}:c,\mathrm{b}:c)) \ast\!\!\ast (\mathrm{u}:\Diamond_b v,\mathrm{i}:v)) \longrightarrow \Box_{sh} \ldots \Box_{sh} I\,(\mathit{perf},\mathit{pass})}{\vdots}\,\Box E}{\dfrac{\left\langle \ldots \langle ((\mathrm{k}:c,(\mathrm{t}:c,\mathrm{b}:c)) \ast\!\!\ast (\mathrm{u}:\Diamond_b v,\mathrm{i}:v))\right\rangle^{sh} \ldots \right\rangle^{sh} \longrightarrow (\mathit{perf},\mathit{pass})}{\vdots}\,\Box E}$$

The sequence of □ operators filling the ellipsis in the sequence $\Box_{sh} \ldots \Box_{sh}$ governs the pattern of consonants and vowels in the result. Our initial example *kutib* instantiates the pattern *cvcvc*. Accordingly, the succedent type for the topmost sequent in the proof above has the form

$$\Box_{sh}\,\Box_c\,\Box_v\,\Box_c\,\Box_v\,\Box_c\,\Box_{sh}\,I\,(\mathit{perf},\mathit{pass})$$

In this case, then, the second phase consists of seven steps of $\Box E$, one for each □. (To increase legibility, we replace each unary mode $\langle - \rangle^i$ by the corresponding $\Diamond_i$ and collapse consonants and vowels into more readable lists.)

$$\vdots$$

$$\frac{(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_{\mathrm{b}}\, \mathrm{v}\ \Diamond_{-b} v \longrightarrow \Box_{sh}\Box_c\Box_v\Box_c\Box_v\Box_c\Box_{sh}\, I(perf, pass)}{\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_c\Box_v\Box_c\Box_v\Box_c\Box_{sh}\, I(perf, pass)}\ \Box_{sh}E$$

$$\frac{\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_c\Box_v\Box_c\Box_v\Box_c\Box_{sh}\, I(perf, pass)}{\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_v\Box_c\Box_v\Box_c\Box_{sh}\, I(perf, pass)}\ \Box_c E$$

$$\frac{\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_v\Box_c\Box_v\Box_c\Box_{sh}\, I(perf, pass)}{\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_c\Box_v\Box_c\Box_{sh}\, I(perf, pass)}\ \Box_v E$$

$$\frac{\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_c\Box_v\Box_c\Box_{sh}\, I(perf, pass)}{\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_v\Box_c\Box_{sh}\, I(perf, pass)}\ \Box_c E$$

$$\frac{\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_v\Box_c\Box_{sh}\, I(perf, pass)}{\Diamond_v\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_c\Box_{sh}\, I(perf, pass)}\ \Box_v E$$

$$\frac{\Diamond_v\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_c\Box_{sh}\, I(perf, pass)}{\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_{sh}\, I(perf, pass)}\ \Box_c E$$

$$\frac{\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow \Box_{sh}\, I(perf, pass)}{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\,(\mathrm{k(tb)} : c(cc) \between \mathrm{ui} : \Diamond_b v\ \Diamond_{-b} v \longrightarrow I(perf, pass)}\ \Box_{sh}E$$

$$\vdots$$

This completes the second phase of the proof.

6.3 Phase 3

In the third phase of the proof, the two arguments of the antecedent ⁕ product are shuffled together in a way controlled by the unary modes flanking them. Each mode carries out a characteristic role, working from the inside out:

— the inner occurrence of $\langle - \rangle^{sh}$ initiates the shuffle, using the identity element 1 as a placeholder for the result, removing itself in the process;
— an inner occurrence of $\langle - \rangle^{c}$ governs the shift of the left-most element of the left-argument of the shuffle product ⁕ — the input consonant list — to the righthand end of the shuffled list, removing itself as a side effect;
— an inner occurrence of $\langle - \rangle^{v}$ governs the shift of the left-most element of the right-argument of the shuffle product ⁕ — the input vowel list — to the righthand end of the shuffled list, removing itself as a side effect;
— continuing in this way, if we can remove all the inner unary mode operators until we reach the final occurrence of occurrence of $\langle - \rangle^{sh}$, we can terminate the shuffle if both the consonant list and the vowel list arguments of ⁕ are empty (as indicated by the occurrence of the identity element 1).

We now formulate each of these steps as a modally-governed structural rule. For readability, we write these structural rules in the format:

$$\Diamond_{\mathrm{i}} A \longrightarrow B$$

In proofs, the antecedent type appears as a substructure of the antecedent structure of the conclusion of an inference step, the succedent appears as a substructure of the antecedent structure of the premise, as shown on the left below. This can be justified by the Cut shown on the right:

$$\frac{\Gamma[B] \longrightarrow C}{\Gamma[\langle A\rangle^i] \longrightarrow C} \qquad \frac{\langle A\rangle^i \longrightarrow \mathrm{B} \quad \Gamma[B] \longrightarrow C}{\Gamma[\langle A\rangle^i] \longrightarrow \mathrm{C}}\ \mathit{Cut}$$

The structural rule package for the shuffle consists of the following rules:

$$((A \divideontimes B) \ltimes 1) \longrightarrow \Diamond_{sh}(A \divideontimes B) \qquad (\mathit{Shuf\text{-}Start})$$
$$(A \divideontimes B) \ltimes (C \bullet D) \longrightarrow \Diamond_{c}(((D \bullet A) \divideontimes B) \ltimes C) \qquad (\mathit{Shuf\text{-}Cons})$$
$$((A \divideontimes B) \ltimes (C \bullet D)) \longrightarrow \Diamond_{v}((A \divideontimes (D \bullet B)) \ltimes C) \qquad (\mathit{Shuf\text{-}Vwl})$$
$$\Diamond_{sh} A \longrightarrow \Diamond_{sh}((1 \divideontimes 1) \ltimes A) \qquad (\mathit{Shuf\text{-}End})$$

Note that in the absence of the rules for 1, we would have to state several additional rules to initiate and terminate the shuffling process. And note as well that apart from the role played by the constant 1, the system is completely *linear* in resource management: each occurrence of a type on the left of each arrow corresponds to exactly one occurrence of the same type on the right of the arrow.

The diagram below illustrates the process of the shuffle, focusing only on the elements of the lists themselves — that is, leaving the modal controllers out.

$$\vdots$$

$$\frac{\ldots \text{k(tb)} : c(cc) \circledast \text{ui} : ((\diamond_b v)v) \ldots \longrightarrow I(perf, pass)}{\ldots ((\text{k(tb)} : c(cc) \circledast \text{ui} : ((\diamond_b v)v)) \ltimes 1) \ldots \longrightarrow I(perf, pass)}\ \textit{Shuf-Start}$$

$$\frac{\ldots ((\text{k(tb)} : c(cc) \circledast \text{ui} : ((\diamond_b v)v)) \ltimes 1) \ldots \longrightarrow I(perf, pass)}{\ldots ((\text{tb} : cc \circledast \text{ui} : ((\diamond_b v)v)) \ltimes (1, \text{k} : c)) \ldots \longrightarrow I(perf, pass)}\ \textit{Shuf-Cons}$$

$$\frac{\ldots ((\text{tb} : cc \circledast \text{ui} : ((\diamond_b v)v)) \ltimes (1, \text{k} : c)) \ldots \longrightarrow I(perf, pass)}{\ldots ((\text{tb} : cc \circledast \text{ui} : ((\diamond_b v)v)) \ltimes \text{k} : c) \ldots \longrightarrow I(perf, pass)}\ \textit{Left Iden}$$

$$\frac{\ldots ((\text{tb} : cc \circledast \text{ui} : ((\diamond_b v)v)) \ltimes \text{k} : c) \ldots \longrightarrow I(perf, pass)}{\ldots ((\text{tb} : cc \circledast \text{i} : v) \ltimes \text{ku} : c(\diamond_b v)) \ldots \longrightarrow I(perf, pass)}\ \textit{Shuf-Vwl}$$

$$\frac{\ldots ((\text{tb} : cc \circledast \text{i} : v) \ltimes \text{ku} : c(\diamond_b v)) \ldots \longrightarrow I(perf, pass)}{\ldots ((\text{b} : \text{c} \circledast \text{i} : v) \ltimes (\text{ku})\text{t} : (\text{c}(\diamond_b v))c) \ldots \longrightarrow I(perf, pass)}\ \textit{Shuf-Cons}$$

$$\frac{\ldots ((\text{b} : \text{c} \circledast \text{i} : v) \ltimes (\text{ku})\text{t} : (\text{c}(\diamond_b v))c) \ldots \longrightarrow I(perf, pass)}{\ldots (((\text{b} : c, 1) \circledast (\text{i} : v, 1)) \ltimes (\text{ku})\text{t} : (c(\diamond_b v))c) \ldots \longrightarrow I(perf, pass)}\ \textit{Right Iden}\ (2\times)$$

$$\frac{\ldots (((\text{b} : c, 1) \circledast (\text{i} : v, 1)) \ltimes (\text{ku})\text{t} : (c(\diamond_b v))c) \ldots \longrightarrow I(perf, pass)}{\ldots (((\text{b} : c, 1) \circledast 1) \ltimes ((\text{ku})\text{t})\text{i} : ((c(\diamond_b v))c)v) \ldots \longrightarrow I(perf, pass)}\ \textit{Shuf-Vwl}$$

$$\frac{\ldots (((\text{b} : c, 1) \circledast 1) \ltimes ((\text{ku})\text{t})\text{i} : ((c(\diamond_b v))c)v) \ldots \longrightarrow I(perf, pass)}{\ldots ((1 \circledast 1) \ltimes (((\text{ku})\text{t})\text{i})\text{b} : (((c(\diamond_b v))c)v)c) \ldots \longrightarrow I(perf, pass)}\ \textit{Shuf-Cons}$$

$$\frac{\ldots ((1 \circledast 1) \ltimes (((\text{ku})\text{t})\text{i})\text{b} : (((c(\diamond_b v))c)v)c) \ldots \longrightarrow I(perf, pass)}{\ldots (((\text{ku})\text{t})\text{i})\text{b} : (((c(\diamond_b v))c)v)c \ldots \longrightarrow I(perf, pass)}\ \textit{Shuf-End}$$

$$\vdots$$

In this way, the shuffle of the triliteral root and the vocalic sequence is initiated, carried out, and terminated — all under the control of the modal operators introduced on the goal type of the categories that the vocalic sequence is assigned to. These modal operators are omitted above to allow a focus on the process itself, rather than its control, but they appear in the full proof below.

6.4 Phase 4

In the final phase of the proof, the unary operator $\langle - \rangle^{sh}$ which terminates the shuffle shifts back to the right as a $\square_{sh}$ decorating the succedent type:

$$\frac{\diamond_{sh}((((\text{ku})\text{t})\text{i})\text{b} : (((\text{c}(\diamond_b v))c)\diamond_{-b} v)c) \longrightarrow I(perf, pass)}{(((\text{ku})\text{t})\text{i})\text{b} \longrightarrow (((c(\diamond_b v))c)\diamond_{-b} v)c \longrightarrow \square_{sh} I(perf, pass)}\ \begin{matrix}\square_{sh} \\ I\end{matrix}$$

As noted earlier, this provides a way to distinguish the succedent type of the result of a successful shuffle from the succedent type of the intermediate steps.

6.5 Putting the phases together

To construct the complete proof, we append the phases to one another, top to bottom:

$$\dfrac{\mathrm{k(tb)} : c(cc) \longrightarrow \mathit{3root} \quad \mathrm{ui} : \Diamond_b v\, \Diamond_{-b} v \longrightarrow \mathit{3root} \backslash \Box_{sh}\Box_c\Box_v\Box_c\Box_v\Box_c\Box_{sh}\, I(\mathit{perf}, \mathit{pass})}{\mathrm{k(tb)} : c(cc) \divideontimes \mathrm{ui} : \Diamond_b v \Diamond_{-b} v \longrightarrow \Box_{sh}\Box_c\Box_v\Box_c\Box_v\Box_c\Box_{sh}\, I(\mathit{perf}, \mathit{pass})}\ \backslash E$$

$$\Box E$$

$$\vdots$$

$$\Diamond_{sh}\Diamond_c\, \Diamond_v\Diamond_c\Diamond_v\Diamond_c\Diamond_{sh}\, \mathrm{k(tb)} : c(cc) \divideontimes \mathrm{ui} : \Diamond_b v \Diamond_{-b} v \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Shuf\text{-}Start}$$

$$\Diamond_{sh}\Diamond_c\, \Diamond_v\Diamond_c\Diamond_v\Diamond_c\, ((\mathrm{k(tb)} : c(cc) \divideontimes \mathrm{ui} : \Diamond_b v \Diamond_{-b} v) \ltimes 1) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Shuf\text{-}Cons}$$

$$\Diamond_{sh}\Diamond_c\, \Diamond_v\Diamond_c\Diamond_v\, ((\mathrm{tb} : cc \divideontimes \mathrm{ui} : \Diamond_b v \Diamond_{-b} v) \ltimes (1, \mathrm{k} : c)) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Left\ Iden}$$

$$\Diamond_{sh}\Diamond_c\, \Diamond_v\Diamond_c\Diamond_v\, ((\mathrm{tb} : cc \divideontimes \mathrm{ui} : \Diamond_b v \Diamond_{-b} v) \ltimes \mathrm{k} : c) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Shuf\text{-}Vwl}$$

$$\Diamond_{sh}\Diamond_c\, \Diamond_v\Diamond_c\, ((\mathrm{tb} : cc \divideontimes \mathrm{i} : \Diamond_{-b} v) \ltimes \mathrm{ku} : c(\Diamond_b v)) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Shuf\text{-}Cons}$$

$$\Diamond_{sh}\Diamond_c\, \Diamond_v\, ((\mathrm{b} : c \divideontimes \mathrm{i} : \Diamond_{-b} v) \ltimes \mathrm{(ku)t} : (c(\Diamond_b v))c) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Right\ Iden}\ (2\times)$$

$$\Diamond_{sh}\Diamond_c\, \Diamond_v(((\mathrm{b} : c,1) \divideontimes (\mathrm{i} : \Diamond_{-b} v,1)) \ltimes \mathrm{(ku)t} : (c(\Diamond_b v))c) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Shuf\text{-}Vwl}$$

$$\Diamond_{sh}\Diamond_c\, (((\mathrm{b} : c,1) \divideontimes 1) \ltimes \mathrm{((ku)t)i} : ((c(\Diamond_b v))c)\Diamond_{-b} v) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Shuf\text{-}Cons}$$

$$\Diamond_{sh}\, ((1 \divideontimes 1) \ltimes \mathrm{(((ku)t)i)b} : (((c(\Diamond_b v))c)\Diamond_{-b} v)c) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \mathit{Shuf\text{-}End}$$

$$\Diamond_{sh}(\mathrm{(((ku)t)i)b} : (((c(\Diamond_b v))c)\Diamond_{-b} v)c) \longrightarrow I(\mathit{perf}, \mathit{pass}) \quad \Box_{sh} I$$

$$\mathrm{(((ku)t)i)b} : (((c(\Diamond_b v))c)\Diamond_{-b} v)c \longrightarrow \Box_{sh} I(\mathit{perf}, \mathit{pass})$$

6.6 Non-linear shuffling

The example above, *kutib*, is an instance of a *linear* shuffle: each element of either input list occurs exactly once in the result. Famously, not all the Arabic cases have this linear character. For example, in the form *ktaabab*, the single vowel *a* of the vocalic pattern occurs more than once and the final root consonant spreads to the right. To treat the consonant case, we simply allow the final consonant of the root to be copied to the end of the shuffle in progress, in addition to being moved from the root list to the shuffle list. This takes the following form:

$$(((1, a) \divideontimes B) \ltimes (C, a)) \longrightarrow \Diamond_c\, (((1, a) \divideontimes B) \ltimes C) \quad (\mathit{Copy\text{-}Cons})$$

A rule of the same form applies to the back vowels *a* and *u*, which are picked out by the decoration $\Diamond_b$.

$$((B \divideontimes (1, \Diamond_b a)) \ltimes (\mathrm{C}, \Diamond_b a)) \longrightarrow \Diamond_c((\mathrm{B} \divideontimes (1, \Diamond_b a)) \ltimes C) \quad (\mathit{Copy\text{-}Vwl})$$

We can represent the shuffle producing *ktaabab*, then, as follows.

$$\vdots$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_v\Diamond_c\Diamond_c((\text{ktb} \divideontimes \text{a}) \ltimes 1) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_v\Diamond_c((\text{tb} \divideontimes \text{a}) \ltimes \text{1k}) \longrightarrow XI(perf,act)}\ \textit{Shuf-Cons}$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_v\Diamond_c((\text{tb} \divideontimes \text{a}) \ltimes \text{1k}) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_v\Diamond_c((\text{tb} \divideontimes \text{a}) \ltimes \text{k}) \longrightarrow XI(perf,act)}\ \textit{Left Iden}$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_v\Diamond_c((\text{tb} \divideontimes \text{a}) \ltimes \text{k}) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_v((\text{b} \divideontimes \text{a}) \ltimes \text{kt}) \longrightarrow XI(perf,act)}\ \textit{Shuf-Cons}$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v\Diamond_v((\text{b} \divideontimes \text{a}) \ltimes \text{kt}) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v((\text{b} \divideontimes \text{a}) \ltimes \text{kta}) \longrightarrow XI(perf,act)}\ \textit{Copy-Vwl!}$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c\Diamond_v((\text{b} \divideontimes \text{a}) \ltimes \text{kta}) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c((\text{b} \divideontimes \text{a}) \ltimes \text{ktaa}) \longrightarrow XI(perf,act)}\ \textit{Copy-Vwl!}$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v\Diamond_c((\text{b} \divideontimes \text{a}) \ltimes \text{ktaa}) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c\Diamond_v((\text{b} \divideontimes \text{a}) \ltimes \text{ktaab}) \longrightarrow XI(perf,act)}\ \textit{Copy-Cons!}$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v((\text{b} \divideontimes \text{a}) \ltimes \text{ktaab}) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c\Diamond_v((\text{b1} \divideontimes \text{a1}) \ltimes \text{ktaab}) \longrightarrow XI(perf,act)}\ \textit{Right Iden}$$

$$\dfrac{\Diamond_{sh}\Diamond_c\Diamond_v((\text{b1} \divideontimes \text{a1}) \ltimes \text{ktaab}) \longrightarrow XI(perf,act)}{\Diamond_{sh}\Diamond_c((\text{b1} \divideontimes 1) \ltimes \text{ktaaba}) \longrightarrow XI(perf,act)}\ \textit{Shuf-Vwl}$$

$$\dfrac{\Diamond_{sh}\Diamond_c((\text{b1} \divideontimes 1) \ltimes \text{ktaaba}) \longrightarrow XI(perf,act)}{\Diamond_{sh}((1 \divideontimes 1) \ltimes \text{ktaabab}) \longrightarrow XI(perf,act)}\ \textit{Shuf-Cons}$$

Notice that the structural rule Copy-Vwl is restricted to back vowels. The front vowel *i* is apparently governed by linear resource management principles: any occurrence of *i* in the input vowel list corresponds to exactly one occurrence of *i* in the resulting shuffled form. But when *i* is final, a preceding back vowel may occur multiple times, as exemplified by such forms as *tukuutib* (*VI*(*perf,pass*)) and *mutakaatib* (*VI*(*part,act*)). This justifies a third copying rule:

$$((B \divideontimes (\Diamond_b a, \Diamond_{-b} b)) \ltimes (C, \Diamond_b a)) \longrightarrow \Diamond_v((B \divideontimes (\Diamond_b a, \Diamond_{-b} b)) \ltimes C) \quad (Copy/\_i)$$

6.7 Additional processes

There are additional inflectional processes at work in the Arabic verbal paradigm: gemination, infixation, affixation. These are intriguing. But since our goal here is to illustrate how intercalative morphology can be treated by a simple and quite general combinatory process (the shuffle product) constrained by the control of modal operators, we leave the treatment of these empirical details to another occasion.

7. Summary

The treatments of intercalation discussed above bring together two broad ideas: distributivity and controlled recursion. In particular, the idea that gram-

matical forms of intercalation can be regarded as a controlled shuffle of two input lists combines the two: the structural rules carrying out the shuffle are distributivity postulates.

A benefit of this general approach is that it offers a more abstract formal setting for investigating distributivity and the processes it can be used to model. And for the root-and-pattern morphology of Hebrew and Arabic, the essential ideas involve the isolation of the root, the representation of the vocalism, and the control structure mediating the combination of the two.

In Harris's practice, the control aspects are always integrated into the pattern through the use of dashes. And morphophonemic rules (Harris 1941: 156f.) are invoked so that roots always combine with commensurate patterns. In this way, it is possible to *define* intercalation as a form of linear, recursive distributivity. This definition applies uniformly to the particular roots and patterns Harris discusses, independently of how many consonants the root contains. And this interpretation accords better with Harris's unitary idea that 'roots are intercalated in patterns' than would a family of non-recursive distributivity rules (a different one for each distinct mode of root/pattern combination).

The model of intercalation as a controlled recursive process as suggested above also shares many properties with McCarthy's 'prosodic' theory. In particular, the prosodic template or CV-skeleton of his model plays much the same role in his account as the modal control mechanism plays in our account above. But the underlying view of how these factors interact is very different. For example, McCarthy's system allows segments to be linked to the skeleton and subsequently delinked. Such operations play no role in the deductive system described here. Moreover, McCarthy's system also rests on general, global well-formedness conditions, including the condition that links from morphemes to skeleton must be order-preserving (embodied in the famous slogan 'association lines do not cross'). This global condition plays no overt role in the system described here: its consequences are enforced intrinsically.

The controlled shuffle which forms the basis of the deductive approach above is an instance of a simple recursive process characterizing a relation (here a ternary relation between the two input lists and the resulting shuffle) which becomes a function when coupled with a controller. This point of view has natural affinities with work in finite-state morphology. Beesley & Karttunen's 'Finite-State Non-Concatenative Morphotactics' (2000) shows how a recursive extension of standard finite-state techniques — embodied in

their 'compile-replace' algorithm — leads to an elegant account of intercalation of the kind discussed here. On their account, consonantal roots and the vocalic sequences of Harris's patterns are both represented as simple finite-state networks; in addition, a third network of arcs labeled *C* or *V* provides a template specifying the remaining details. An operation called *merge* associates the elements of the first two networks with the third. The architecture of this system clearly displays the interaction of a simple logical engine (*merge*) with a controller governing the combination of consonants and vowels.

Controlled processes of this kind can be used to model other grammatical phenomena as well. For example, it is straightforward to construct a recursive rule for copying a list. Using the identity 1 as above to mark the end of a list, we have:

$$\begin{array}{ll} \Diamond_d(a \bullet a) \bullet \Diamond_2 A \longrightarrow \Diamond_2(a \bullet A) & (\textit{start}) \\ \Diamond_d((A \bullet b) \bullet (A \bullet b)) \bullet \Diamond_2 B \longrightarrow \Diamond_d(A \bullet A) \bullet \Diamond_2(b \bullet B) & (\textit{continue}) \\ A \longrightarrow \Diamond_d A \bullet \Diamond_2 1 & (\textit{done}) \end{array}$$

Thus, to copy a form completely, simply decorate it with $\Diamond_2$ and use the rules above as a recursive structural rule package. To copy a form partially, as in many forms of reduplicative morphology, one needs a controller. As in the case of intercalation studied above, a linguistic form can provide its own intrinsic controller, through selected and salient aspects of its structure.

It is perhaps anachronistic to ask whether one or another implementation of a given formal intuition corresponds precisely to Harris's intentions. I hope that the design solution discussed above for intercalation is one that Harris would have found congenial. In any case, the empirical phenomena in language that caught his eye continue to repay revisitation.

Acknowledgements

This paper had its origin in materials prepared for a course at the XIth European Summer School on Logic, Language, and Information, held in Utrecht in August 1999, jointly taught with Michael Moortgat, who I wish to thank for many helpful discussions on the topics treated here. In addition, I am grateful to Bruce Nevin and anonymous reviewers for very constructive suggestions.

References

Ajdukiewicz, Kazimierz. 1935. "Die syntaktische Konnexität". *Studia Philosophica*, 1: 1–27. English translation in Storrs McCall (ed.) *Polish Logic: 1920–1939*, 207-231, Oxford: Oxford University Press, 1967.

Bar-Hillel, Yehoshua. 1953. "A quasi-arithmetical notation for syntactic description". *Language*, 29: 47–58. Repr. in (Bar-Hillel 1964: 61-74).

Bar-Hillel, Yehoshua. 1964. *Language and information*. Addison-Wesley, Reading, Massachusetts.

Beesley, Kenneth R. & Lauri Karttunen. 2000. "Finite-state non-concatenative morphotactics". In *Proceedings of the Fifth Workshop of the ACL Special Interest Group in Computational Phonology*, pp. 1–12.

Chomsky, Noam & Morris Halle. 1968. *The sound pattern of English*. New York: Harper & Row.

Gentzen, Gerhard. 1935. "Untersuchungen über das logische Schliessen". *Mathematische Zeitschrift*, 39: 176–210, 405–431. English translation in (Gentzen 1969: 68-131).

Gentzen, Gerhard. 1969. *The collected papers of Gerhard Gentzen*. M. E. Szabo, ed. Amsterdam: North-Holland.

Harris, Zellig S. 1941. "Linguistic structure of Hebrew". *Journal of the American Oriental Society*, 61: 143–167.

McCarthy, John J. 1981. "A prosodic theory of nonconcatenative morphology". *Linguistic Inquiry*, 12.3: 373–418.

Moortgat, Michael. 1997. "Categorial type logics". In J. van Benthem & A. ter Meulen (eds.) *Handbook of Logic and Language*. Amsterdam: Elsevier.

Morrill, Glyn. 1994. *Type logical grammar*. Dordrecht: Kluwer.

Oehrle, Richard T. 2001. "Multi-modal type-logical grammar". In R. Borsley and K. Börjars (eds.) *Non-Transformational Syntax*. Oxford: Blackwell.

Chapter 3

Sequence structure

D. Terence Langendoen
University of Arizona

1. Introduction

Harris (1946) anticipated current interest in the avoidance of unnecessary constructs in linguistic theory when he wrote:

> [T]here is an advantage in avoiding [constructs such as 'morphological levels'] if we can achieve the same results by direct manipulation of the observable morphemes. The method described in this paper will require no elements other than morphemes and sequences of morphemes, and no operation other than substitution, repeated time and again.

Assuming with Harris that a language consists of morphemes and sequences of morphemes, I investigate the structures of morpheme sequences in a language as determined by their relation to their subsequences which also belong to that language.

2. Fundamental notions

In this section, I define and illustrate the notions of subsequence, maximal (proper) subsequence, sequence structure, and conjunction of sequences.

2.1 Subsequences

Let L be a set of sequences of morphemes.[1] Then $s \in L$ has as its parts any subsequence $r \in L$, where the subsequence relation $\geq_S$ is defined as in (1). $\geq_S$

1. For our purposes, a single morpheme counts as a morpheme sequence. However, L does not contain the 'empty' sequence ε, though it can contain morphemes with 'zero allomorphs'.

is reflexive, antisymmetric, and transitive; i.e., it is a partial order.

(1) For all r, $s \in L$, r is a **subsequence** of s in L ($s \geq_S r$) if and only if there is an $n > 0$ such that there are sequences $r_1, \ldots, r_n \in L$, and $x_0, \ldots, x_n$ such that $r = r_1 \ldots r_n$ and $s = x_0 r_1 x_1 \ldots r_n x_n$.

In this chapter, I make three additional assumptions. First, I assume that L contains every morpheme that appears in any sequence in L. Without this assumption, one could consider L to be a set of words, possibly polymorphemic, which does not contain some of the morphemes (i.e. the affixes and bound roots) that appear in those words. Second, I assume the stronger version of the subsequence relation $\geq_Q$ in (2).

(2) For all r, $s \in L$, r is a **strict subsequence** of s in L ($s \geq_Q r$) if and only if there is an $n > 0$ such that there are sequences $r_1, \ldots, r_n$, $x_1, \ldots, x_{n-1} \in L$, and x_0, x_n either $\in L$ or empty, such that $r = r_1 \ r_n$ and $s = x_0 r_1 x_1 \ldots r_n x_n$.

By requiring that $x_1, \ldots, x_{n-1}$ also belong to L, $\geq_Q$ is not transitive over every set of morpheme sequences. For example, let $L_0 = \{a, b, c, d, ad, bd, cd, abd, acd, bad, bcd, cad, cbd, abcd, acbd, bacd, bcad, cabd, cbad\}$. Then $abcd \geq_Q acd$, and $acd \geq_Q ad$, but $abcd \not\geq_Q ad$, since $bc \in L_0$. Hence $\geq_Q$ is not transitive and therefore not a partial order on L_0. My third assumption is that the sets of morpheme sequences of linguistic significance include only those for which $\geq_Q$ is a partial order. Henceforth by 'subsequence' I mean 'strict subsequence'.

2.2 Sequence structures

Let $L_1 = \{cure, able, ity, curable, curability\}$, where *cure*, *able*, and *ity* are morphemes. Each morpheme has only itself as a subsequence, whereas *curable* has *cure* and *able*, as **proper** subsequences, and *curability* has *curable*, (and hence also *cure* and *able*) and *ity*, as proper subsequences. A proper subsequence r of a sequence s is **maximal** if s has no other proper subsequence of which r is a proper subsequence. For example, *curable* is a maximal proper subsequence of *curability*, but *cure* and *able* are not.[2] and it induces over L_1 the inverse hierarchical structure Q_1 diagrammed in Figure 1, in which reflexive arcs and

2. Henceforth I drop the term 'proper' in describing proper maximal subsequences.

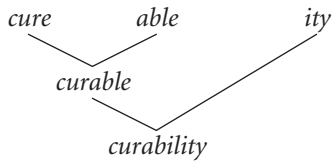


Figure 1. Sequence structure Q_1

arcs derivable from transitivity are omitted. I call such a structure a **sequence structure**.

2.3 Conjunction of sequences

Since $\geq_Q$ is a partial order, the greatest lower bound of any two sequences in a sequence structure, if it exists, is their **conjunction**. For example, the conjunction of *cure* with *able* in Q_1 is *curable*, and the conjunction of *curable* with *ity* is *curability*. In these examples, the conjunction operator is equivalent to concatenation (in a certain order). However for other examples, it is not; for example, the conjunction of *curable* with *able* is *curable*, of *cure* with *ity* is *curability*, and of *able* with *ity* is also *curability*.

The sequence structure Q_1 is that of L_1 as a whole, not just of the root morpheme sequence *curability*. To see this more clearly, let us add to L_1 the morphemes *prove*, *sane*, and *mouse*, and the morpheme sequences *sanity*, *provable* and *provability*. The sequence structure Q_2 of the resulting set L_2 is shown in Figure 2. In Q_2, the conjunctions of the elements in Q_1 remain the same, except that of *able* with *ity*, which have no conjunction.[3] Both *curability* and *provability* are candidates (both have *able* and *ity* as subsequences), but neither is a subsequence of the other (i. e. there is no **greatest** lower bound for

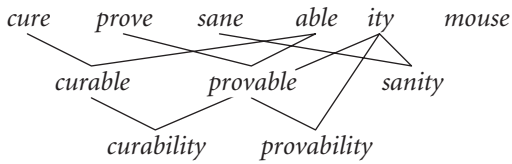


Figure 2. Sequence structure Q_2

3. However, if the sequence *ability* is added to L_2, then the conjunction of *able* with *ity* is *ability*.

able and *ity* in L_2). In addition, *sane* and *able* have no conjunction in Q_2, there being no sequence which has both *sane* and *able* as subsequences.[4] Q_2 is a multiple (or multiply rooted) inverse hierarchy.

Sequence structures can also involve classes of morphemes. An example is Q_3 in , which is based on L_3 = {*V*, *A*, *N*, *A**V*, *N**A*, *V A**V*, *A N**A*, *V A**V N**A*}, derived from L_2 by setting *V* = {*cure*, *prove*}, *A* = {*sane*}, *N* = {*mouse*}, *A**V* = {*able*}, and *N**A* ={*ity*}. Sequences with the same potential for entering into longer sequences can be established by cancellation in the usual manner, so for example the sequences *A* and *V A**V* are combinatorially equivalent, both combining with *N**A*, etc.

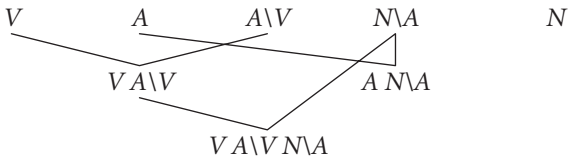


Figure 3. Sequence structure Q_3

2.4 Mergers of sequences

The relation $\geq_Q$ can be generalized to allow for any number of antecedent sequences. For this purpose, I define a **merger** of two sequences as in (3).

(3) The sequence *u* is a merger of sequences *s*, *t* in *L* if $u \geq_Q s$, $u \geq_Q t$, and every morpheme in *u* also appears in either *s* or *t*.

Then $s_1, \ldots, s_n \geq_Q r$ if *r* is a subsequence of a merger of $s_1, \ldots, s_n$. It follows that if *u* is a merger of *s* with *t*, then *u* is also the conjunction of *s* with *t* provided that there is no other sequence *v* in *L* which is a merger of *s* with *t*.

2.5 Subsequences versus substrings

The sequence structures Q_1 through Q_3 are compatible with a stronger mereological relation than subsequence: the substring relation $\geq_R$ defined in (4).

4. For justification of this view of conjunction, see Koslow (1992).

(4) For all $r, s \in L$, r is a **substring** of s ($s \geq_R r$) if and only if there are sequences x_0, x_1 such that $s = x_0 r x_1$.

However, let $L_4 = \{$*the*, *only*, *point*, *the point*, *the only point*$\}$. In L_4, *the point* occurs as a subsequence of *the only point*, but not as a substring. Assuming that *the point* is correctly analyzed as a part of *the only point*, as in the sequence structure Q_4 in Figure 4, then the weaker subsequence relation is the desired mereological relation for linguistic analysis. In Q_4, *the only point* is the conjunction of *the point* with *only*.

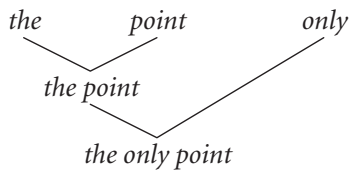


Figure 4. Sequence structure Q_4

3. Sequential ambiguity

In this section, I define a notion of sequential ambiguity which can be made to match that of structural ambiguity as defined for constituent structures. A sequence s is **sequentially unambiguous** in Q if it is the conjunction of at most one pair $\{t, u\}$ of maximal subsequences in Q; otherwise it is **sequentially ambiguous**. Every sequence in Q_1 through Q_4 is sequentially unambiguous. Similarly, every sequence in the sequence structure Q_5 of $L_5 = \{$*very*, *nice*, *result*, *very nice*, *nice result*, *very nice result*$\}$ in Figure 5 is sequentially unambiguous, including *very nice result*, since it is the unique conjunction of its maximal subsequences *very nice* and *nice result*. The fact that *very nice*

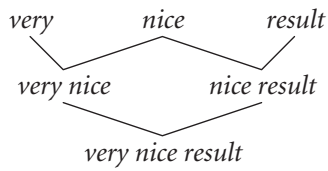


Figure 5. Sequence structure Q_5

result is also the conjunction of *very* with *nice result* and of *very nice* with *result* is irrelevant, since these pairs are not maximal subsequences of *very nice result*.

On the other hand, the sequence *old clothing store* in the sequence structure Q_{6a} of L_{6a} = {*old*, *clothing*, *store*, *old clothing*, *clothing store*, *old store*, *old clothing store*} in Figure 6 is sequentially ambiguous, since it is the conjunction of three different pairs of its maximal subsequences *old clothing*, *old store* and *clothing store*.

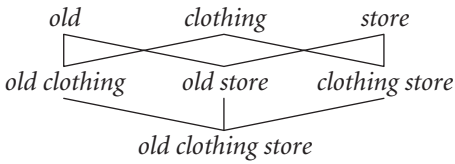


Figure 6. Sequence structure Q_{6a}

Viewing the sequence *old clothing store* as the conjunction of the maximal subsequence *old clothing* with *clothing store* corresponds to the constituent structure [[*old clothing*] *store*], whose interpretation is that the clothing in the store is old, but not necessarily the store. Viewing it as the conjunction of *old store* with *clothing store* corresponds to the structure [*old* [*clothing store*]] whose interpretation is that the store is old, but not necessarily the clothing in it. However the conjunction of *old clothing* with *old store* does not correspond to any interpretation of *old clothing store* in English. If it did, it would be to one in which both the clothing and the store (not necessarily a clothing store) are old. To account for the fact that *old clothing store* is only two ways, and not three ways sequentially ambiguous in English, we observe that the members of L_{6a} are part of a language that also includes the sequences *store clothing* and *old store clothing*, with the sequence structure Q_{6b} in Figure 7. In Q_{6b}, *old clothing* and *old store* have two different mergers, *old clothing store* and *old store clothing* (the latter shown with dashed lines), so that neither one is the conjunction of

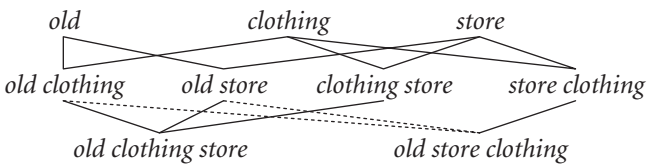


Figure 7. Sequence structure Q_{6b}

old clothing with *old store*. The sequence *old clothing store* is therefore the conjunction of only the two pairs of maximal subsequences, *old clothing* and *clothing store*, and *old store* and *clothing store*, as desired.[5]

This correlation between sequential ambiguity and structural ambiguity extends to more complex cases, such as that of Q_{6d} of L_{6d}, which includes all of the sequences in L_{6b} together with those in L_{6c} = {*fine*, *fine clothing*, *fine store*, *fine clothing store*, *fine store clothing*, *fine old clothing*, *fine old store*, *old fine clothing*, *old fine store*, *fine old clothing store*, *fine old store clothing*, *old fine clothing store*, *old fine store clothing*}. Figure 8 shows that in Q_{6d} the sequence *fine old clothing store* is four-ways sequentially ambiguous, being the conjunction of the following four pairs of maximal subsequences: (1) *fine old clothing* and *fine clothing store* (corresponding to the constituent structure [[*fine* [*old clothing*]] *store*]); (2) *fine old clothing* and *old clothing store* (corresponding to the constituent structure [[[*fine old*] *clothing*] *store*]); (3) *fine old store* and *fine clothing store* (corresponding to the constituent structure [*fine* [*old* [*clothing store*]]]); and (4) *fine old store* and *old clothing store* (corresponding to the constituent structure [[*fine old*][*clothing store*]]). It is not the conjunction of *fine clothing store* with *old clothing store*, since *old fine clothing store* is also a merger of those maximal subsequences (shown with dotted lines). Nor is it the conjunction of *fine old clothing* with *fine old store*, since *fine old store clothing* is also a merger of those maximal subsequences (shown with dashed lines). On interpretations (1) and (2), the clothing is fine and old, but not necessarily the store; on interpretations (3) and (4), the store is fine and old, but not necessarily the clothing. The sequence lacks an interpretation corresponding to the constituent structure [*fine* [[*old clothing*] *store*]], in which

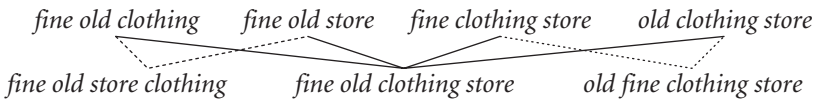


Figure 8. Part of sequence structure Q_{6d}

5. In Q_{6b} the conjunction of *clothing* with *store* is also undefined, since both *clothing store* and *store clothing* are mergers of *clothing* with *store*. However, both *clothing store* and *store clothing* are interpretable sequences in Q_{6b}, for the simple reason that there is no conjunction at all of their maximal subsequences *clothing* and *store*. It is only if a sequence has an analysis as the conjunction of maximal subsequences that nonconjunctive mergers of maximal subsequences are ignored.

the store but not necessarily the clothing is fine, and in which the clothing but not necessarily the store is old. But this is correct, since that interpretation is only possible if there is a clear juncture between *fine* and *old*, and again between *clothing* and *store*, i.e. if the sequence contains at least one additional 'juncture' morpheme.

3.1 The need for nonstandard morpheme sequences

In order for subsequence ambiguity to match structural ambiguity in some cases, certain morpheme sequences that are not standardly assumed to be part of a language must be posited. For example, the structural ambiguity of the sequence *old men and women* in English is not matched by sequential ambiguity in the sequence structure Q_{7a} of L_{7a} = {*and*, *men*, *women*, *old*, *and men*, *and women*, *old men*, *old women*, *and old men*, *and old women*, *men and women*, *women and men*, *old men and women*, *old women and men*, *men and old women*, *women and old men*}. Figure 9 shows that *old men and women* is sequentially unambiguous in Q_{7a}; it is the conjunction of the maximal subsequences *old men* and *men and women* only.

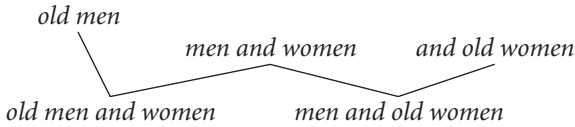


Figure 9. Part of Q_{7a} showing lack of sequential ambiguity of *old men and women*

To get the result that *old men and women* is sequentially ambiguous, we must replace in L_{7a} the sequences {*and men*, *and women*, *and old men*, *and old women*} that have been standard in most recent analyses of English with the nonstandard sequences {*men and*, *women and*, *old men and*, *old women and*}. Figure 10 shows that in the resulting sequence structure Q_{7b}, *old men and women* is sequentially ambiguous, being the conjunction of the maximal subsequence *old men and* with *men and women* (the interpretation being that of the constituent structure [[old men] and women], in which the men are old, but not necessarily the women), and of *old men and* with *old women* (the interpretation being that of the constituent structure [old [men and women]],

in which both the men and the women are old). However, *old men and women* is not the conjunction of *old women* with *men and women*, since both *old men and women* and *men and old women* are mergers of those maximal subsequences (the latter shown with dashed lines).

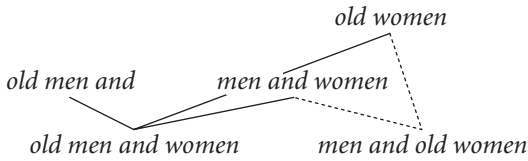


Figure 10. Part of Q_{7b} showing sequential ambiguity of *old men and women*

3.2 Sequential vs. morphemic (lexical) ambiguity

Sequential ambiguity must be distinguished from morphemic (lexical) ambiguity just as structural ambiguity must. In the sequence structure Q_8 in Figure 11, based on L_8 = {*un*^{1}, *un*^{2}, *ed*, *pack*, *un*^{2}*pack*, *packed*, *un*^{1}*packed*, *un*^{2}*packed*}, neither *un*^{1}*packed* nor *un*^{2}*packed* is sequentially ambiguous. The former is the conjunction of the maximal subsequences *un*^{1} and *packed*; the latter of *un*^{2}*pack* and *packed*.

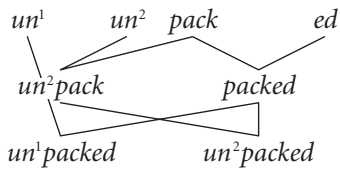


Figure 11. Sequence structure Q_8

The ambiguity of expressions like *Irish grammarian* in Q_9 of L_9 = {*Irish*^{1}, *Irish*^{2}, *grammar*, *ian*} in Figure 12 is similarly explained. On the interpretation "grammarian who is Irish", it is the conjunction of the maximal subsequences *Irish*^{1} (referring to a person) and *grammarian*; whereas on the 'bracketing paradox' interpretation "student of Irish grammar", it is the conjunction of the maximal subsequences *Irish*^{2} *grammar* and *grammarian* (*Irish*^{2} referring to a language).

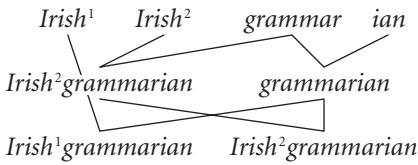


Figure 12. Sequence structure Q_9

3.3 Repetition of morphemes

A morpheme can occur more than once in a morpheme sequence, as in *I think I drink* in L_{10a} = {*I*, *think*, *drink*, *I think*, *I drink*, *I think I drink*}, in which the morpheme *I* occurs twice. In order to represent the sequence structure of sets containing such sequences, the two occurrences of the morpheme must be distinguished, say by indices, with the convention that subsequent occurrences have higher indices. For example, L_{10a} may be replaced by L_{10b} = {I_1, I_2, *think*, *drink*, I_1 *think*, I_2 *think*, I_1 *drink*, I_2 *drink*, I_1 *think* I_2 *drink*}, which has the sequence structure Q_{10b} in Figure 13. The latter is the conjunction of the maximal subsequences I_1 *think* and I_2 *drink* only, and so is sequentially unambiguous.

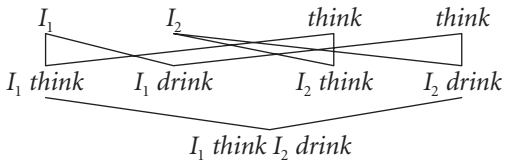


Figure 13. Sequence structure Q_{10b}

4. Modeling transformational relations using sequence structures

Sequence structures can also model mereological relations for which movement transformations have been thought necessary.

4.1 Apparent 'local movement': subject–auxiliary inversion in English

For example the sequence *can tan* is assumed to be part of *can Fran tan* in English, but is not considered a constituent of it. Given that it is a constituent of *Fran can tan*, the sentence *can Fran tan* is generally considered to be derived from it by the 'movement' of the auxiliary verb *can* around the subject noun *Fran*. However *can tan* is a subsequence of *can Fran tan* just as much as it is of

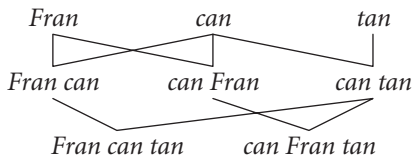


Figure 14. Sequence structure Q_{11}

Fran can tan. No movement of *can* around *Fran* is necessary to identify *can tan* as part of *can Fran tan* in the appropriate sequence structure. Specifically in Q_{11}, the sequence structure of L_{11} = {*Fran*, *can*, *tan*, *Fran can*, *can Fran*, *can tan*, *Fran can tan*, *can Fran tan*} in Figure 14, *can Fran tan* is the conjunction of the maximal subsequences *can Fran* and *can tan*, while *Fran can tan* is the conjunction of the maximal subsequences *Fran can* and *can tan*.

4.2 Apparent 'unbounded movement': *wh*-movement in English

The 'movement' of *can* in *can Fran tan* does not reorder the subsequences of *can tan*, so that *can tan* is, as has already been observed, a subsequence of *can Fran tan*. However the 'movement' of *what* (an instance of *wh*-movement) in *what will Phil mill* does reorder the subsequences of *mill what*, which allegedly occurs as part of *what will Phil mill*. As a result, *mill what* is not a subsequence of *what will Phil mill*, so that subsequence structure does not provide a model of *wh*-movement in English as it is usually analyzed in generative grammar.

The model of *wh*-movement that emerges from sequence structure analysis is one in which the 'moved' *wh*-element appears at the left edge of successively longer sequences, including eventually the verb of which it is an argument (if it is one), as in the sequence structure Q_{12} of L_{12} = {*Phil*, *will*, *mill*, *what*, *Phil will*, *what will*, *will Phil*, *will mill*, *mill what*, *Phil will mill*, *will Phil mill*, *will mill what*, *what will Phil*, *Phil will mill what*, *what Phil will mill*,

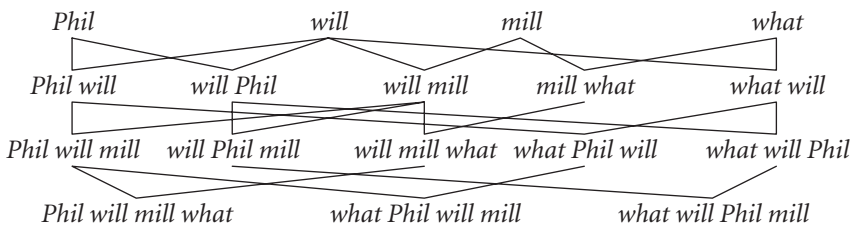


Figure 15. Sequence structure Q_{12}

what will Phil mill} in Figure 15. In Q_{12}, the shortest sequence containing 'moved' *what* is *what will*. The next longer sequences *what Phil will* and *what will Phil* are the conjunctions of *what will* with *Phil will* and *will Phil* respectively. Finally the sequences *what Phil will mill* (a subsequence of *I know what Phil will mill*, in an extension of L_{12}) and *what will Phil mill* are the conjunctions of *what Phil will* with *Phil will mill* and of *what will Phil* with *will Phil mill*, respectively.[6]

In cases in which a *wh*-element is 'extracted' from a complement, that element will be part of a sequence expressing exactly that complement, so that the grammatical relation of the 'extracted' element to other elements in the complement can be determined in a sequence structure. For example, the sequence *who may Fay say Phil will thrill* is analyzed in the appropriate sequence structure Q as the conjunction of the maximal subsequences *who may Fay say* and *who Phil will thrill*, so that the grammatical relation of *who* with *thrill* in *who may Fay say Phil will thrill* can be determined. In addition, the fact that certain elementary sequences containing two *wh*-elements do not belong to the sequence structure directly accounts for the absence of longer such sequences, without appeal to 'constraints' on movement. For example, the absence of the sequence **who what will drill* in Q accounts for the absence of **who may Fay say what will drill*, since the latter would have to be analyzed as the conjunction of the maximal subsequence *who may Fay say* with *who what will drill* in Q. On the other hand, if the sequence *what who will drill* belongs to Q, then we may expect that the sequence *what may Fay say who will drill* will also belong to Q, since the latter is the conjunction of *what may Fay say* with *what who will drill*.

5. Determining the interpretations of sequence structures

The interpretation of a sequence s is fully determined by its place in a sequence structure Q. If s is a single morpheme m, then its interpretation is simply that which is assigned to m. If s is a sequence of two or more morphemes, then either it is (1) not the conjunction of any pair of maximal subsequences $\{t, u\}$, (2) the conjunction of exactly one such pair, or (3) the

6. Again, certain nonstandard sequences must be postulated in order for the analysis to work.

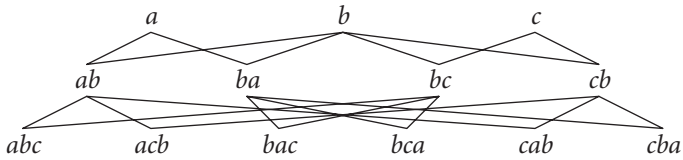


Figure 16. Sequence structure Q_{13}

conjunction of more than one such pair. Case (1) arises when for every pair $\{t, u\}$ of maximal subsequences of s, there is another sequence r in Q which is a merger of $\{t, u\}$. In this case, the interpretation of s is a function of the interpretation of each pair of its maximal subsequences. The simplest subcase of case (1) arises when $s = tu$ and $r = ut$, as in Q_{6d}, where $s =$ *clothing store*. In Q_{6d}, the sequences of which s and r are maximal subsequences do not fall under case (1); i.e. in Q_{6d}, the property of failing to be the conjunction of maximal subsequences is not inherited by the sequences of which the sequences with that property are maximal subsequences. (Both *old clothing store* and *old store clothing* are conjunctions of pairs of maximal subsequences in Q_{6d}.) The property is only inherited under very specific circumstances, as in the hypothetical sequence structure Q_{13} in Figure 16. In Q_{13}, neither *ab* and *ba*, nor *bc* and *cb* are conjunctions of their maximal subsequences *a* and *b*, and *b* and *c* respectively. In addition, neither *bac* nor *bca* are conjunctions of their maximal subsequences *ba* and *bc*; i.e., both *bac* and *bca* inherit the property. On the other hand, *abc*, *acb*, *cab*, and *cba* are conjunctions of their respective maximal subsequences; i.e., they don't inherit the property.

In addition, I am not aware of any natural language examples in which a sequence is the merger of more than one pair of maximal subsequences, and that sequence is the conjunction of none of those pairs. If that restriction indeed holds, then all instances of case (1) are sequentially unambiguous.

Case (2) has two subcases: (a) when a sequence has exactly one pair of maximal subsequences, and (b) when it has more than one such pair, but it is the conjunction of the members of exactly one of them. The sequence *very nice result* in Q_5 is an instance of case (2a); no instances of case (2b) are given here.

Case (3) also has two subcases: (a) when a sequence is the conjunction of every pair of its maximal subsequences (there being at least three such pairs), and (b) when it is the conjunction of more than one pair of its maximal subsequences, but not of every pair. Case (3a) does not arise in any of the examples we have considered, and I believe does not arise at all in natural languages. The sequences *old clothing store* in Q_{6d} and *old men and women* in

Q_{8b} are instances of case (3b). In this case the sequence has as many interpretations as the number of pairs of maximal subsequences of which it is the conjunction; in each case the interpretation is a function of the interpretations of those maximal subsequences.

The degree of sequential ambiguity of any case (3b) sequence is determined simply by the number of pairs of maximal subsequences of which it is the conjunction. The sequential ambiguity of those maximal subsequences does not enter into the calculation. For example, on one of its interpretations, the sequence *fine old clothing store* in Q_{7d} is the conjunction of the pair of maximal subsequences *fine old clothing* and *old clothing store*. The fact that the latter is itself sequentially ambiguous does not matter. In other words, given an analysis of a sequence into a pair of maximal subsequences of which it is the conjunction, each member of that pair makes a univocal contribution to the interpretation of that sequence, even if it is itself sequentially ambiguous.

6. Concluding remarks

For the past several years, I have been exploring the applicability to linguistic analysis of Koslow's (1992) notion of an 'implication structure,' which consists solely of a set *S* and an implication relation (a partial ordering) defined over *S*. It had occurred to me that the substring relation induces an implication structure over a set of strings, but that structure turns out not to have quite the right properties for general linguistic analysis. At first, when I considered the passage in Harris (1946) quoted at the beginning of this essay, I did not realize that the subsequence relation is distinct from the substring relation. It was when I realized that they are distinct, and that the subsequence relation is the weaker of the two, that I discerned that the structure that the subsequence relation induces over a set of sequences (or strings, it does not matter what you call them) does have properties suitable for linguistic analysis. I hope to explore the matter further in future work.[7]

7. I thank Richard Oehrle and the editor, Bruce Nevin, for helpful comments and discussion of this chapter, and Arnie Koslow for many exchanges of ideas over the years.

References

Harris, Zellig S. 1942. "Morpheme alternants in linguistic analysis". *Language* 18:169–180.
Harris, Zellig S. 1946. "From morpheme to utterance". *Language* 22:161–183.
Koslow, Arnold 1992. *A Structuralist Theory of Logic*. Cambridge: Cambridge University Press.

Part 2

Computability of language

Chapter 4

The computability of strings, transformations, and sublanguage

Naomi Sager and Ngô Thanh Nhàn
New York University

1. Introduction

Zellig Harris's work in linguistics placed great emphasis on methods of analysis. His theoretical results were the product of prodigious amounts of work on the data of language, in which the economy of description was a major criterion. He kept the introduction of constructs to the minimum necessary to bring together the elements of description into a system. His own role, he said, was simply to be the agent in bringing data in relation to data.

Outsiders could see the genius and great insight into the workings of language that guided the application of rigorous methods of analysis, leading as they did to the formulation of grammatical systems, and ultimately to a penetrating theory of language and information (Harris 1982, Harris 1991). But it was not false modesty that made Harris downplay his particular role in bringing about results, so much as a fundamental belief in the objectivity of the methods employed. Language could only be described in terms of the placings of words next to words. There was nothing else, no external metalanguage. The question was how these placings worked themselves into a vehicle for carrying the 'semantic burden' of language.

Yet Harris's work did not start with a big question and search directly for the answer. His commitment to methods was such that it would be fair to say that the methods were the leader and he the follower. His genius was to see at various crucial points where the methods were leading and to do the analytic work that was necessary to bring them to a new result.

The close relation of Harris's grammatical descriptions to the real data of language invited the possibility of computation, and the close relation of the described structures to the information content of sentences suggested that

such computations could lead to the performance by computer of practical informational tasks.

Harris himself had an interest in computation. A number of the procedures that he manually carried out were virtually dry runs of what a computer could be programmed to do. One example is the determination of morpheme boundaries in a phonemically represented utterance by noting peaks in the successive counts of possible next phoneme in utterances that share the same initial segment up to the point of counting (Harris 1968, Section 3.2). On the syntactic level, the cycling-cancellation automaton for sentence well-formedness (Harris 1962) was described in sufficient detail so that it could be implemented from its description and used to analyze medical documents (Shapiro 1967).

2. Linguistic string computation

First, we survey the early computational approaches to syntactic analysis.

2.1 The UNIVAC program

The first computer program to perform syntactic analysis of English sentences was developed by a group under the direction of Harris at the University of Pennsylvania in the period from 1957 to 1959. It ran on the UNIVAC I and successfully analyzed a short scientific text (Harris 1959).

The algorithm of the UNIVAC program incorporated the major constructions of English grammar in considerable detail. While the dictionary was small, lexically ambiguous words were multiply classified (i.e. assigned category symbols corresponding to their different parts of speech, e.g. *walk* noun *N* and verb *V*), with provision in the algorithm for recognizing these as potential sources of alternative analyses. Idioms were included, with provision in the algorithm for certain permitted interruptions in the textual occurrence of the idiom.

The UNIVAC sentence analyzer was not a toy program, nor was it specifically tailored for the sample text. Its generality was demonstrated again 40 years later when the program was reconstituted at the University of Pennsylvania and shown to be effective in computing sentence structure (Joshi & Hopely 1997). It was noted in published comments on this reconstruction that "many of the currently popular techniques for robust parsing are already

present, fully articulated in the 1959 UNIVAC parser from UPenn" (Karttunen 1997).

The 1959 UNIVAC program used a grammatical formulation that was termed at the time 'substring analysis'. This was later generalized to 'axiomatic string theory', described along with a brief summary of the UNIVAC program in (Harris 1962a).

2.2 The NYU linguistic string program

A parsing program based on linguistic string analysis, with subsequent extensions to perform transformational and sublanguage analysis, underwent continuous development at New York University from 1965 to 1998. The system came to be known as the LSP (Linguistic String Project) system. The remainder of this chapter summarizes some of the experience of this effort.

The LSP parsing algorithm and grammar grew out of an attempt to solve a problem left over from the 1959 UNIVAC program, namely, how to obtain not just one valid analysis of a sentence, but all possible analyses consistent with the grammar embodied in the program, i.e. how to treat syntactic ambiguity.

The UNIVAC program performed multiple scans of the sentence, recognizing first the 'first order strings' such as noun phrases and prepositional phrases, then the 'second order strings' or 'verb-containing strings' of which the first order strings could be elements. The program left markers at points where decisions were made among alternative lexical categories or alternative ways of continuing the substring analysis.

After some study of how a changed decision at a point of ambiguity affected further processing, several conclusions could be drawn:

— Greater clarity regarding grammatical alternatives would result from separating the grammar from the analysis procedure.
— The elimination of levels in the definition of substrings ('first order' and 'second order') used in different stages of processing would make it easier to correlate a choice made at one point with a dependent choice made at another point.
— The definition of strings as composed solely of category symbols, and the definition of substring relations solely in terms of the possibilities of inserting given types of substrings into other substrings, would make possible a single left-to-right analysis procedure and allow for keeping track of decisions in an orderly way.

In particular the observation which led to the LSP algorithm was that if the grammar was constituted, as above, of elementary strings composed of category symbols, grouped into classes according to the points in other strings at which they have permission to occur, then as one proceeded from left to right through the sentence representation (the sequence of category symbols corresponding to the words of the sentence), each successive word's category symbol (one or more) was either the continuation of a string already begun in the analysis or the beginning of a string permitted to occur at that point. Whenever a category symbol of the current sentence word matched more than one category symbol of the grammar, an alternative analysis path through the sentence would be opened. Keeping track of the opening and closing of paths could be done in various ways (Sager 1960, 1967).

2.3 Implementation of the LSP string parser

The approach taken in the first implementation of the 1960 single-scan left-to-right procedure was to develop a fairly general, language-independent processor, with the grammar definitions and input sentences represented as list structures (Morris 1965). The parse procedure was top down, syntax driven, keeping the analysis in the form of a tree, with ability to back up and obtain another analysis when a branch failed or when the end of the sentence was reached with a successful parse. To apply linguistic constraints to the parse tree, the grammar writer called upon operators for navigating the tree and performing logical operations, and procedures for applying the tests (called 'restrictions') to the parse tree nodes or the sentence representation. On encountering a conjunction, the parser dynamically generated coordinate conjunction strings. As candidate definitions, it used copies of those that were used to analyze the immediately preceding words as properly nested string occurrences. Subsequent implementations have followed a similar approach. A computer grammar of English was written in this style (Sager 1981). The grammar was also adapted to process medical documents in French (Nhàn 1989), German (Oliver 1992), and Dutch (Spyns et al. 1996).

A parse tree obtained in the above manner is not transparently a linguistic string analysis. For one thing, the points of optional string insertion, before or after particular category symbols (e.g. before *N*, after *V*, or at stated inter-element points) become elements of the grammar definitions (Figure 1), hence are seen as nodes in the record of the analysis in the form of a parse tree (Figure 2). Thus, the position to the left of *N* at which a left adjunct of *N* has permis-

<ASSERTION> ::= <SA> <SUBJECT> <SA> <TENSE> <SA> <VERB> <SA> <OBJECT> <SA-LAST> .

<LNR> ::= <LN> <NVAR> <RN> .

<SA> ::= <*NULL> | <SAOPTS> <SA> .

<SAOPTS> ::= <PDATE> | <SUB11> | <SUB9> | <SUB12> | <SUB0> | <PN> | <PD> | <LDR> | <VENPASS> | <VINGO> | <NSTGT> | <RNSUBJ> | <RSUBJ> | <SUB5> | <SUB1> | <SUB2> | <SUB3> | <SUB8> | <TOVO> | <PVINGO> | <PWHERES>.

Grammar definitions are written in Backus Naur Form (BNF):
<X> <Y> means <X> AND <Y>; <X> | <Y> means <X> OR <Y>.

SA	sentence adjunct position;
SAOPTS	options of SA (modifiers of entire ASSERTION);
SA-LAST	options of SA in string-final position;
LNR	noun N with left and right adjuncts;
LN	left adjuncts of N;
NVAR	local variants of N;
RN	right adjuncts of N;
PDATE	preposition + date-form;
SUBn	subordinate conjunction strings;
PN	prepositional phrase;
PD	preposition + locative LDR;
LDR	adverb with left and right adjuncts;
VENPASS	passive string;
VINGO	gerund string;
NSTGT	noun string of time;
RNSUBJ	post-object adjuncts of subject N;
RSUBJ	roving adjuncts of subject;
TOVO	infinitive string;
PVINGO	Preposition + VINGO;
PWHERES	Preposition + WHERE string.

Figure 1. Definitions in the LSP string grammar

sion to occur is seen as a node of the tree, *LN*, whose value may be a string of the type 'left adjunct of *N*'. Similarly, the position of sentence adjunct occurrence, before or after any element of an elementary verb-containing string, appears as a node SA in the parse tree. In Figure 2, the first SA node represents the position where the sentence adjunct *Today* had permission to insert itself into the elementary assertion string *she has cough*.

In the example parse tree in Figure 2, only non-empty elements of the grammar definitions are shown, except for the ordered adjunct positions of LN:

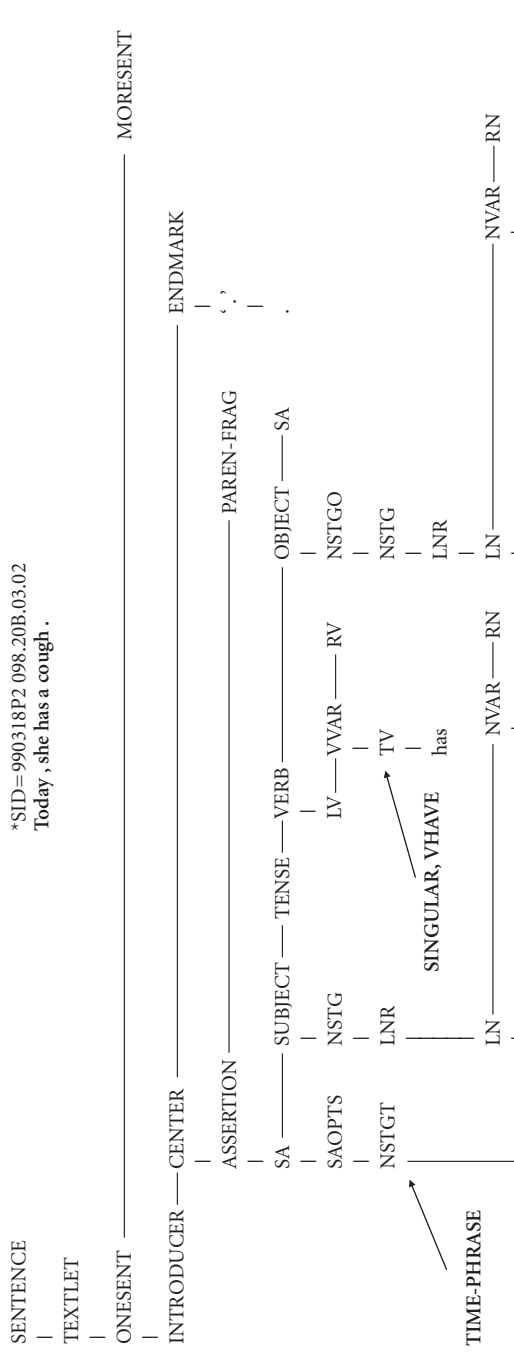


Figure 2. Linguistic string parse tree

TPOS	article position
QPOS	quantity position
APOS	adjective position
NPOS	left compound noun position

VERB is classed as an LXR-type node. English lexical attributes are SINGULAR, NOMINATIVE, FEM, NTIME2, and VHAVE. TIME-PHRASE is a computed node attribute, and H-PT and H-NEG are Healthcare-sublanguage lexical attributes.

The definition of optional insertion points as elements of the computer representation may seem like a simple accommodation for efficiency of implementation, but it masks the linguistic string character of the underlying grammar by giving the same form to a linguistic relation as to a position of word occurrence in the sentence. Thus, the linguistic string parse tree looks like a tree formed by an immediate constituent grammar, but in essential respects it is not.

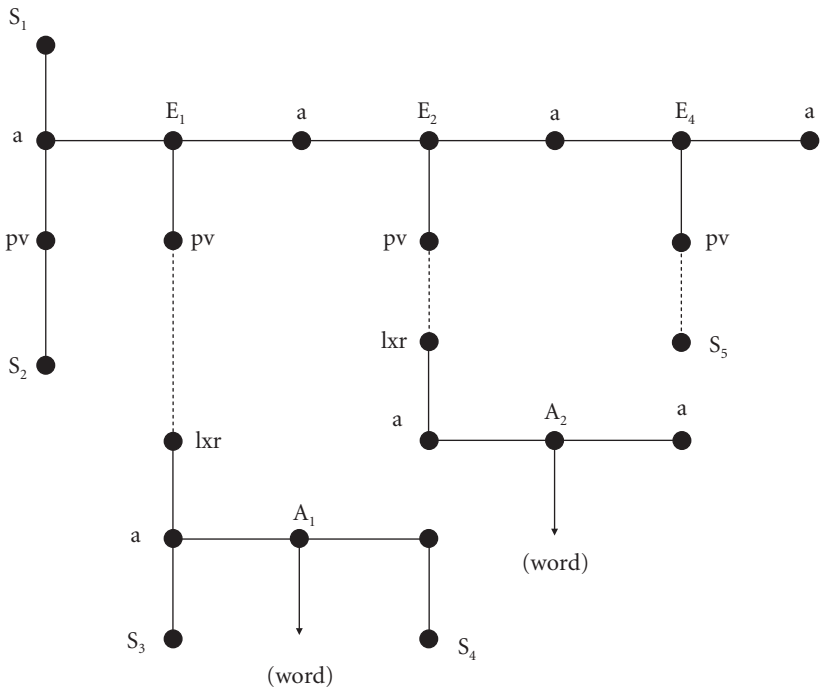
String analysis is better suited for computation than immediate constituent analysis is. One of the reasons is that linguistic constraints, whether grammatical (e.g. number agreement of subject and verb) or selectional (e.g. semantic compatability of a noun and its modifier), apply only to words occurring as coelements within an elementary string or as elements of strings related by string adjunction. (There is also a special case of noun replacement that accounts for subject and object strings.) Computationally, this means that the arguments of a test that is to realize a linguistic constraint (the words to be tested) can always be located in the parse tree based on their string relation. In an immediate constituent parse tree, it is not as straightforward to point to words that have a co-dependence.

To retain this advantage of string analysis for computation, grammar definitions in the form used by the parser (in later implementations written in Backus Naur Form, BNF, as seen in Figure 1) are divided into types according to their role in representing string grammar. The type *STRING* covers all definitions corresponding to the elementary strings of axiomatic string theory. In the BNF representation, the *STRING* elements are usually not category symbols but named sets of positional variants that terminate in the category symbols of elementary strings, as discussed above.

The type *LXR* covers all definitions consisting of a category symbol *X* preceded by the set of its left adjuncts *LX* and followed by the set of its right adjuncts *RX*, where *LX* and *RX* each has a null option to express the

optionality of adjunct occurrence. The 'core' of an *LXR* node in the parse tree is uniformly its central category symbol *X* (or the value of *XVAR*, its local variants), which is also the core of the string element that lies above it in the parse tree and of any intermediate positional variants. Thus, in Figure 2, *PRO* (*she*) is the core of *LNR* under positional variant *NSTG* under the element *SUBJECT* of the string *ASSERTION*. In *ASSERTION*, the elementary string *N tV N* (*she has cough*) is the sequence of the cores of the elements *SUBJECT*, *VERB*, *OBJECT*.

Because navigation routines (*CORE*, *ELEMENT*, *COELEMENT*, etc.) are written in terms of the definition types (node types in the parse tree, Figure 3), they can locate the arguments of restrictions as though they existed in a simpler tree composed solely of string-related category symbols.



Node types

S	string	pv	positional variant
a	adjunct set	lxr	atom x with its adjuncts
E	element	A	atomic node

Figure 3. Generalized linguistic string parse tree

Over the years significant features have been added to the string parser. These include:

- The *Restriction Language*, a programming language for stating the restrictions on the parse tree (Sager & Grishman 1975).
- Procedures for checking the semantic well-formedness of the parse tree in terms of the co-occurrence of word subclasses in particular syntactic relations — for example, to check whether the words occurring as the noun-preposition-noun (*N P N*) relation are in compatible sublanguage word classes for the given subject area.
- Procedures for rearranging and augmenting the parse tree in accordance with established linguistic transformations — for example, to expand conjunction constructions.
- Mappings to different forms of output depending on further regularities observed in the data or the needs of particular applications.

2.4 From strings to transformations

Harris recognized the validity of different methods of analysis. In Section 1.4 of (Harris 1962a:18-19), he compared string analysis to transformational analysis, and these in turn to immediate constituent analysis:

> If we consider all three types of analyses, we note first that string analysis is intermediate between the other two: It isolates one elementary sentence out of each sentence; constituent analysis isolates no sentence; while transformational analysis reduces the whole sentence to elementary sentences (with primitive adjuncts) and constants [. . .].
>
> Nor does the difference lie in the power of the three to characterize different sets of sentences [. . .]. For each of these types of analysis can describe all the sentences of a language (though at very different cost in complexity of the description) [. . .].
>
> The difference is rather in how the three analyses interrelate the sentences and sentence-segments of the language: For each characterization of a sentence relates that sentence to its decomposition products and also to other sentences having a similar decomposition. Thus, constituent analysis shows to what extent the sentences can all be viewed as sequences of two constituents, subject and predicate, with sentence adjuncts deployed around them. String analysis relates all sentences having the same elementary sentence, the same adjuncts, etc. Transformational analysis goes far beyond either in bringing together the sentences which we feel should be brought together. Thus it relates *He is slow in learning* with *He learns slowly*; and *He began to speak* with *He spoke*; and *He seems young* with *He is young*; and *whom I saw* adjoining *man* with *I saw the man*; whereas neither

> constituent analysis nor string analysis shows direct relation between the members of each pair [. . .].
>
> Nevertheless, though transformational analysis is the most refined, all three analyses are relevant, for language has the properties of all three.

A transformational analysis of a sentence is more refined than other grammatical analyses in one respect in particular: it is closer to an informational decomposition of the sentence. It displays the component individual statements that were combined into one larger informational package, the sentence. This suggested strongly that the path from string parsing to informational applications would lead through transformations.

It was clear from the start that linguistic strings were closely related to transformations. The sentence forms of the transformational kernel set were virtually the same as the elementary center strings of linguistic string analysis, and many of the elementary adjunct strings could be described as 'deformed' elementary sentences, e.g. the adjective left adjunct of the noun in *A N* could be said to be a 'deformation' of *N is A* obtained by dropping the *is* and permuting *A* to before *N*. Thus, many linguistic strings can be seen as the form an elementary sentence takes as a result of an information-preserving form change that makes it available to be a component of a larger sentence.

3. Transformational computation

Transformational analysis brought with it new challenges for computation.

3.1 Initial considerations

As transformational analysis evolved from a relation among sentence forms to a theory of grammar (Harris 1968, Ch. 4), it was possible to base transformational computation on one or another of its formulations. Because a transformational decomposition of a sentence makes explicit how every element of meaning enters the sentence and the changes of form this entails, there was interest in finding a formal (and computable) representation of this process. Harris provided a representation in the form of Decomposition Lattices (Harris 1967, also Harris 1970), in which each node corresponds to a transformational operation and the lattice displays the order of their operation.

The requirements for an implementation of a decomposition-lattice analysis of sentences are formidable. A large number of detailed transformations must be formulated and formalized; a correspondingly detailed lexicon must be developed, in which derivational affixes are treated (e.g. the *ly* in *slowly* in Harris's example above: *He is slow in learning* ↔ *He learns slowly*). Unfortunately these imposing requirements have prevented such a computer program from being developed.

Without going so far as to do a complete transformational decomposition, it is possible to use the transformational relations among sentence forms to bring into alignment such segments as carry the same or similar information, somewhat in the spirit of transformations as a tool for discourse analysis (Harris 1952, Harris 1963). For example, in one form of output of the LSP system, mapping the output to a relational database, transformations are used implicitly by placing in the same column the words that would have been aligned linguistically by transformations. Thus, *she broke her ankle*, *broken ankle*, *ankle break*, *a break in the ankle bone*, will all have *ankle* in a column of the database table labeled *BODYPART*, and *broke*, *broken*, *break* in a column labeled *SYMPTOM*, without having rearranged the parse tree in accord with the applicable linguistic transformations.

3.2 Implementation of transformations in the LSP system

Some transformations in the LSP system are implemented as changes to the parse tree and some transformations are utilized rather than implemented. One example of the latter was given above. For another example, the passive transformation N_2 *is Ven by* $N_1 \leftrightarrow N_1$ *tV* N_2 need not be executed on the parse tree in order for selectional (word choice) compatibility in a passive construction to be checked, based on a list of acceptable subject-verb-object patterns stated for N_1 *tV* N_2. Similarly, it is not necessary to reconstruct *N is A* from an *A N* occurrence in a sentence in order to check the compatibility of the adjective and noun in this relation. There is some advantage in retaining the original word order of the sentence unless the goals for the representation or the application require the rearrangement of sentence parts.

The transformations that change the parse tree primarily serve to obtain complete, or relatively complete, informational units of the *ASSERTION* type from the more diverse adjunctive and conjunctive forms in the original sentence. Coordinate conjunction constructions are expanded up to the *ASSERTION* level, i.e. the 'understood' or 'zeroed' elements are copied from

Original sentence from an anonymized patient document:

Today, she has no cough, chest pain, or shortness of breath.

is transformed into single information units:

Today, she has no cough
, today, she has no chest pain,
and today, she has no shortness of breath.

where:
– Time word *today* is distributed to the basic statements;
– Negative word *no* is distributed to every object;
– *or* in distributed negative statement is transformed to *and*.

Figure 4. Transformation of a parsed sentence into information units

the full form into the reduced form in the positions dictated by parallel construction. Thus, *Extremities revealed clubbing and cyanosis* becomes *Extremities revealed clubbing and [extremities revealed] cyanosis*. In the case of a construction of the type *NO X OR Y*, the negative is distributed and the conjunction is changed to *AND*: *Extremities revealed no clubbing or cyanosis* ↔ *Extremities revealed no clubbing and [extremities revealed] no cyanosis*. The antecedent of the bound pronoun in a *WH* construction likewise is supplied: *A peripheral neuropathy workup was initiated which revealed normal folate levels* ↔ *A peripheral neuropathy workup was initiated which [workup] revealed normal folate levels*. Other modifiers are similarly expanded with the goal of obtaining elementary *ASSERTION* units that are informationally relatively complete. Figure 4 illustrates the expansion process.

4. Sublanguage computation

This application of transformational analysis computationally to texts led naturally to a more detailed consideration of domain-specific word relations, i.e. to sublanguage grammar.

4.1 The sublanguage method

Natural Language Processing (NLP), so named to distinguish it from the processing of computer languages, needs to arrive at a representation of the

content of texts in order to provide further procedures that depend upon it, such as information extraction and word-pair indexing. Some attempts have been made to move directly to semantic characterization without syntactic analysis, and even for those who believe that syntax is part of the information there is a recognition that there is another part. The particular words that occur in the given syntactic relations are what convey the specific information.

Linguists had not been unaware of the role that word choice plays in language. Leonard Bloomfield discussed this as the phenomenon of 'selection' (Bloomfield 1933:164–169, 190–199, 229–237). However, no rules could be imposed as to which word choices make acceptable as opposed to unacceptable sentences. The flexibility of language that enables it to accommodate nonsense, fairy tales, untruths, and so on, leaves it to the speaker to choose whichever words seem suitable as long as they are assembled into an understandably grammatical sentence.

It was Harris's work that first brought word choice into the realm of grammar, albeit in this case as a criterion, not a rule: the definition of the transformational relation between sentence forms, based on the similarity (on some scale) of the acceptability of the word choices in the candidate forms. However, when Harris introduced the notion of sublanguage grammar, particularly with regard to science sublanguages (Harris 1968, Section 5.9.1), the door was opened to extending the rules of grammar into the realm of selection. In a science sublanguage, some types of sentences are possible while others are simply outside the subject area or are such combinations of sublanguage words as are simply not sayable within the science. To use Harris's example (1968:152), in the language of biochemistry, contrast the possible (1) *The polypeptides were washed in hydrochloric acid*, with (2) *Hydrochloric acid was washed in polypeptides*, which if it ever occurred would not be in the discourse of biochemistry.

What was immediately appealing about sublanguage was its methodology. Word classes of semantic specificity could be established objectively based on their sublanguage co-occurrence properties, and in terms of these word classes, sublanguage statement types could be defined to serve as templates to house the information in sublanguage texts.

Experimentally, it was possible to show that the semantically relevant word classes of a particular biomedical sublanguage could be established on purely distributional grounds, using a clustering program (Hirschman *et al.* 1975). In

the same vein more recently, a computer program (named ZELLIG in honor of its co-occurrence basis) was developed to obtain semantic classes for French medical documents by applying distributional criteria to noun phrases in parsed documents. The classes obtained corresponded well to the major term types of an established medical terminology (Nazarenko *et al.* 2001).

Frequently co-occurring sublanguage word classes in particular syntactic relations lead to the formulation of a very large array of detailed sublanguage statement types that can be grouped for convenience in different ways. Harris et al. (1989) developed a formulaic representation of the sentence types in a science sublanguage. The purpose of the work, as stated by Harris in the Foreword, was

> [. . .] to develop a formal tool for the analysis of science, and more generally of information [. . .]. In respect to the history of science, the formulaic representation of research done over a period shows, for example, changes in the way words for the objects of the science co-occur with words for the processes, changes which exhibit the actual development of the science.

Another form for grouping related statement types was termed an 'information format' (Sager 1978). This form proved convenient for computer operation on the data. As applied to clinical documents (the Healthcare sublanguage), statement types with a common feature (e.g. the occurrence of a treatment-type word class, or a laboratory-type word class) were combined into one information format that covered the occurrence of all the statement types of that class (Sager *et al.* 1987).

Since the concept of sublanguage was introduced, it has proved especially fruitful in language computation, as attested by chapters in this volume and other publications (e.g. Kittredge & Lehrberger 1982, Grishman & Kittredge 1986, Marsh & Friedman 1985).

4.2 A medical sublanguage

Illustrations of sublanguage computation will be drawn from the LSP treatment of the sublanguage of clinical reporting, i.e. narrative accounts of patients' conditions and treatments as recorded primarily in hospital discharge summaries and visit reports. Reports have been drawn from the areas of Cardiology, Restricted Airways Disease (RAD, mainly, asthma), Rheumatoid Arthritis, Epilepsy, Sickle Cell Disease, Orthopedics, and to a lesser degree from a variety of other specialties. There has been some experience with other

types of documents, such as imaging reports, pathology reports, and surgical reports, each of which employs some specialized vocabulary and usages related to the techniques employed. The French experience was with texts in Digestive Surgery. Portions of a patient visit report are shown in Figure 5.

4.2.1 *Syntax of the healthcare sublanguage*

The first thing that strikes one about most free text clinical documents (once they are typed or otherwise made legible) is their seemingly wild departures from normal syntax. Some 'sentences' are series of noun phrases and other forms, punctuated only by commas. Others are grammatical but endlessly long, as though stopping to form a new sentence would compromise the information. Single-word sentences are not uncommon, where all the words that make the one word into a statement are understood.

HISTORY DIAGNOSIS: Stage I left breast cancer, diagnosed February 19xx.

INTERVAL HISTORY: Ms. XXX returns for her semi-annual visit approximately one month earlier than scheduled. In the last week, she has had tenderness in the mid to lower right axilla as well as in 2 or 3 spots in her right breast including laterally at about the 9:00 position and inferiorly along the inframammary fold. She has not been able to palpate any specific lumps in these areas although she thought she could at 1 point feel a lymph node in the underarm.

On review of systems, the patient has hip pain which is from degenerative joint disease. She under the care of Dr. YYY of ZZZ Dept. of Orthopaedics. She is also recently recovering from a upper respiratory infection felt to be bronchitis. She is taking the last day of an Azithromycin long-acting schedule. She has had improvement in symptoms in the last 1-2 days.

REVIEW OF SYSTEMS: She denies headaches or visual symptoms. Today, she has no cough, chest pain, or shortness of breath.

PHYSICAL EXAMINATION:
Vitals: weight 58.2 stable, pulse 98, BP 131 / 73, temp 36.4, resp 16 unlabored.
The patient appears well.
HEENT: Head atraumatic and normocephalic.
Fundi: benign.
Mouth and throat: clear.
Neck: supple
. . .

Figure 5. Portions of a patient visit report

Table 1. Shortened sentence forms in the healthcare sublanguage

[N V] N	Stiff neck and fever
N [*be*] A | Ving | Ven	Brain scan negative
	Patient complaining of increased breathlessness
	No growth seen
[N *be*] A | Ving	Positive for heart disease and diabetes
	Feeling better
[N] tV O	Has Paget's disease
[N *be*] Ven O	Treated for meningitis
[N *be*] to V O	To be followed in Pain Clinic
[N *be*] P N	On folic acid

The key to this lack of grammaticality is to realize that in most cases what is observed is the residue of a properly formed sentence after all words that would be obvious to another clinician are dropped, or rather are still present but reduced to zero form ('zeroed' in Harris's term). Sometimes this relies on an understood *the patient*, the default subject of all manner of clinical observations (*Fever.* ↔ *Patient has fever.*). It is interesting that for the most part the reduced sentence forms (Table 1) are strings that occur otherwise in English string grammar, similarly also often involving the zeroing of the verb *be*. For example, compare *Brain scan negative*, in Table 1, with *They pronounced the brain scan negative*, in which the same shortened sentence form occurs grammatically as an object string.

Thus, it is possible to write a grammar of the ungrammatical, by observing that the departures from grammaticality are not arbitrary, but follow patterns of reduction that are for the most part already familiar. The BNF part of the Healthcare sublanguage grammar contains a definition *FRAGMENT* whose options are definitions that also occur in the English computer grammar on which the sublanguage grammar is based.

4.2.2 *Word classes of the healthcare sublanguage*

Word classes of the Healthcare sublanguage have been developed manually, first by studying texts for patterned occurrences, then by defining diagnostic frames for further classification of vocabulary (Sager *et al.* 1987). The word classes in current use are listed with examples in Table 2.

Table 2. Word classes of the healthcare sublanguage

Medical Classes	Description	Examples in English and French
*** PATIENT AREA		
H-PT	references to patient	*she, candidate, Mrs. XXX, patient*
H-PTAREA	anatomical area	*edge, left, surface, rebord, gauche*
H-PTDESCR	patient description	*American, homeless, works*
H-PTFUNC	physiological function	*BP, auditory, appetite, tonalité*
H-PTLOC	location relation	*branching, radiating, localisé*
H-PTMEAS	anatomical measure	*height, bulk, depth, corpulence*
H-PTPART	body part	*arm, adrenal, carotid, liver, foie*
H-PTPALP	palpated body part	*abdomen, liver, foie*
H-PTSPEC	specimen from patient	*sample, scraping, frozen section*
H-PTVERB	verb with patient subj	*complain, endure, suffer*
*** TEST / EXAM AREA		
H-TXCLIN	clinical exam procedure	*Babinski, palpation, auscultation*
H-TXPROC	diagnostic procedure	*MRI, xray, ultrasound,*
H-TXSPEC	test of specimen	*CBC, immunoassay, urinanalysis*
H-TXVAR	test variable	*Iodide, iron, glucose, GB*
*** TREATMENT AREA		
H-TTGEN	general medical mgmt	*follow-up, admit, discharge, soins*
H-TTMED	treatment by medication	*aspirin, clamoxyl*
H-TTSURG	surgical interventions	*excise, hysterectomy*
H-TTCOMP	complementary therapy	*bedrest, repos, physiothérapie*
*** RESULT AREA		
H-AMT	amount or degree	*much, partly, total, sévère*
H-DESCR	neutral descriptor	*amber, amorphous, amphoric,*
H-DIAG	diagnosis	*asthma, diabetes mellitus*
H-INDIC	disease indicator word	*fever, swelling, pain, thrombose*
H-NORMAL	non-problematical	*within normal limits, bon état*
H-ORG	organism	*renibacterium, rickettsial, rod*
H-TXRES	test/exam result word	*gram-negative, positive, positif*
H-RESP	patient response	*better, improve, relief*
H-CHANGE-MORE	quantity increase	*peak, rise, increase, spikes*
H-CHANGE-LESS	quantity decrease	*lower, recede, reduce, taper*
H-CHANGE-SAME	quantity constant	*keep, remain, same, maintain*
H-CHANGE	indication of change	*alteration, changing, drift, modify*
*** EVIDENTIAL AREA		
H-NEG	negation of finding	*no, not, cannot, denied, ne pas*
H-MODAL	uncertainty of finding	*probable, seems, suspicion*

(Table 2 *cont.*)

Medical Classes	Description	Examples in English and French
*** CARE ENVIRONMENT AREA		
H-FAMILY	family, friends, . . .	*she, sister, father, boy friend*
H-INST	doctors, institutions, . . .	*Dr. XXX, hospital, social service*
*** TIME AREA		
H-TMBEG	beginning	*onset, new, dévelope, apparition*
H-TMEND	termination	*end, terminal, discontinue, arrêt*
H-TMLOC	location in time	*current, previous, actuelle, post-op*
H-TMPER	duration	*brief, persistent, constant*
H-TMREP	repetition	*frequently, intermittent, habituelle*
H-TMPREP	time preposition	*during, after, since, après, depuis*
*** CONNECTIVE AREA		
H-BECONN	classifier verb	*is (a), represent, resemble, est (un)*
H-CONN	connects 2 IF's	*due to, along with, secondaire à*
H-SHOW	connects test & result	*shows, confirmed, notable for*

One might ask why manual as opposed to automatic methods of sublanguage word class generation have been used. For one reason, the frequently occurring reduced sentence forms in this sublanguage deprive the automatic procedure of the explicit syntactic relations upon which the clustering depends. In addition, in complete sentences, syntactic ambiguity can muddy the results. Without constraints that utilize the very sublanguage classes one is trying to generate, too many syntactically valid parse trees are generated for clear co-occurrence patterns to emerge. Bootstrapping approaches could probably be developed.

Another group of reasons is special to the medical domain and the intended applications. There is a very large special vocabulary of medicine. Co-occurence analysis of free text input could probably determine major classes, but it is far more efficient to use medical dictionaries and text elicited from experts. Morphological analysis of the Latinate medical vocabulary can result in automatic classification of many medical words, and this has been done (Wolff 1984). Finally, the quality of the output of language processing depends crucially on the quality of the dictionary used in the processing. The standards for the delivery of information to support healthcare are particularly high so that whatever human or computer means are used to create the dictionary, human quality control is an absolute necessity.

To build the Healthcare sublanguage dictionary, words are coded for their syntactic properties according to the scheme described in Appendix 3 of (Sager 1981), to which are added the appropriate medical classes as additional attributes, relying in large part on the contexts in which the words occur. The English Healthcare dictionary currently numbers about 51,000 words, supplemented by lists derived from published sources, e.g. drug lists.

4.2.3 *Creation of new connectives*

Harris envisioned that sublanguage analysis would stimulate the definition of new connectives.

> Even the small classes that fill the role of transformational constants, such as prepositions and conjunctions, which have always been considered to be unextendable objects in grammar, can receive new members in particular subsets of sentences, thus increasing the grammar for these sentences. The creation of new members of prepositions *P* and conjunctions *C* is possible because certain grammatical sequences of morphemes have the same neighbors within a sentence form as do *P* or *C*. (Harris 1968: Section 5.9.2)

In the course of developing the Healthcare sublanguage grammar and applying it to texts, the issue of what constituted an information unit arose. Some prepositions (e.g. *with*) when occurring between two nouns of the same predicate-type subclass (e.g. H-INDIC) could be seen as a connective between two reduced-sentence-form units of information, e.g. *headache with fever* similar to *headache and fever*. Extending this process, similar sublanguage environments became the criterion for defining many new idiom prepositions and some new subordinate conjunctions. A partial list of idiom prepositions in the Healthcare sublanguage dictionary is shown in Table 3.

4.3 Healthcare sublanguage processing

The overall sequence of procedures in the Medical Language Processor, or MLP, as it has come to be called, is shown in Figure 6. In practice, the processing of clinical documents requires a number of preliminary procedures, which are not specifically linguistic in character but are necessary if the documents are to be parsed. Examples include recognizing names, determining section heads, finding sentence boundaries, treating abbreviations, and normalizing number, date, and unit formats. These and other operations are combined into a preprocessing stage. After preprocessing, every sentence carries a sentence identifier (SID), which locates it as an element of a document set, a

Table 3. Idiom prepositions in the healthcare sublanguage

accompanied by	free of	prior to
according to	halfway up	regardless of
accounting for	improved by	relieved by
aggravated by	in absence of	remarkable for
akin to	in anticipation of	resulting from
along with	in association with	resulting in
alternating between	in between	s / p
alternating with	in competition with	secondary to
apart from	in contrast to	significant for
as a consequence of	in light of	similar to
as a result of	in regard to	situated in
as distinct from	in spite of	situated on
as exemplified by	in terms of	specific for
as part of	in the absence of	status post
associated with	in the course of	subsequent to
at the time of	in view of	such as
because of	inconsistent with	suggestive of
bounded by	independent of	suspicious for
characterized as	instead of	suspicious of
characterized by	located in	tolerant of
close to	made worse by	triggered by
compatible with	manifest as	typical of
confined to	mediated by	unassociated with
consistent with	more than	up to
consisting of	notable for	w / o
down to	notable only for	with and without
due to	on basis of	with involvement of
evolving to	on the basis of	with regards to
except for	on top of	with respect to
exemplified by	other than	without evidence of
followed by	out of	worsened by
free from	precipitated by	

particular document, a section of the document, a paragraph in the document and a sentence in the paragraph.

MLP dictionaries include the basic Healthcare sublanguage dictionary described above, along with outside sources and special subarea dictionaries that add special terms and alternative definitions in case of conflict. The parsing engine provides for dictionary lookup to obtain the parts of speech and syntactic and sublanguage attributes of document words, calls on the parsing grammar to obtain the syntactic analysis of the sentence, and applies

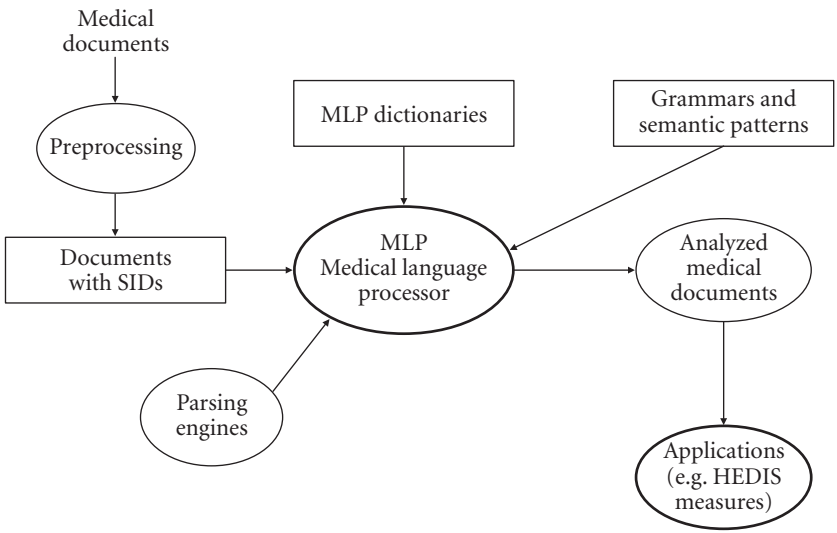


Figure 6. The MLP system overview

the sublanguage (semantic) patterns to resolve syntactic and lexical ambiguity. It then applies the transformational grammar and the information formatting procedures, after which the output can be mapped into the desired form. An overview of the MLP system is given in (Sager *et al.* 1994).

4.3.1 *Sublanguage constraints in parsing*

A parsing grammar that contains most of the constructions found in English sentences, plus reduced sentence forms, is very likely to produce multiple analyses of an input string. To constrain the number of analyses and, hopefully, arrive at the intended one, the grammar must be further restricted, and this is the primary role of sublanguage in parsing. Some of the more interesting situations are noted here.

Conjunctional equivalence

For the MLP to end up with correctly segmented and characterized information units, it is important that coordinate conjunction strings be composed of 'like' elements, not any parsable *N CONJ N*. In sublanguage terms, the conjoined *N*s should be in the same or similarly occurring sublanguage classes, e.g. all H-PTPART words, or an H-INDIC word with an H-DIAG word. This problem can arise even in a straightforward medical sentence, such as

> The concurrent weight loss raises a concern in regard to malignancy of the stomach, pancreas, colon, and female organs.

Structural definitions (the BNF component) in the Healthcare sublanguage grammar would generate (among others) a parse showing *malignancy* and *pancreas* conjoined. Compare the syntactically similar sentence in which *malignancy* and *ulcer* are conjoined:

> The concurrent weight loss raises a concern in regard to malignancy of the stomach, or benign gastric ulcer.

To prevent inappropriate conjoinings, the Healthcare sublanguage grammar contains lists of subclasses that are compatible in conjunction constructions. For example, two sublists of the list CONJ-EQUIV-CLASSES from the grammar are:

> (H-TTSURG, H-TXCLIN [*refused surgery or workup*]),
> (H-TTSURG, H-INDIC, H-DIAG [*Past medical history includes hypertension, left hip arthroplasty and Perth's disease*]),

A restriction checks conjuncts using these lists. If the test fails, it is likely that conjoining will succeed if the conjunct is detached from its current position in the parse tree and re-attached to another available host.

Computed attributes

When sublanguage conjunction constraints are applied, it becomes apparent that testing core *N*s is not always effective, because in some contexts it is the semantic value of the *N + adjunct* that enters into conjunction equivalency. For example, in *fatigue and swollen ankles* the subclasses H-INDIC (*fatigue*) and H-PTPART (*ankles*) are not in a CONJ-EQUIV-CLASS sublist, but if we allow the *N + LN* (*swollen ankles*) to take on the 'computed attribute' H-INDIC (from *swollen*), then the conjoining will be approved.

In applying the conjunction equivalency test, numerous situations have to be accounted for. For example, in *Fatigue and swollen ankles and knees*, the implicit computed attribute for *swollen knees* must be inferred in order for the triple conjunct to be accepted.

4.3.2 *Selection using sublanguage co-occurrence patterns*

By far the greatest source of syntactic ambiguity is the situation in which an adjunct string can be parsed as adjunct to different candidate hosts, especially in the ubiquitous *N PN PN* sequences. This problem can be compounded by

the presence of conjunctions. The approach taken by the LSP has been to collect well-formed patterns of *host + adjunct*, specified with regard to the syntactic relation and the sublanguage word classes that occur correctly in that relation, and to use these authenticated patterns as 'filters' to reject occurrences that do not conform.

For example, in the parse tree for *Rash over abdomen over past week*, the final analysis will show both *PN*s with *P = over* adjoined to *rash* (H-INDIC) in the parse tree, since there is no stored *N + PN* pattern (for *P = over*) corresponding to *abdomen over past week*, i.e. a host *N* of class H-PTPART with a time expression as adjunct. It should noted that 'host *N*' here refers to the core *N* as carrier of node attributes, so that if the core *N* carries a 'computed attribute' it is that attribute that will be used in the filtering test. Thus, *Swollen abdomen over past week* will pass the test, because *abdomen* in this case carries the computed attribute H-INDIC (from *swollen*), which can be adjoined by a time adjunct.

The computed attribute is another instance of employing a transformational relation without carrying out the transformation. In a transformational analysis of the above example, one step would be: *swollen abdomen over past week* ↔ *abdomen was* (or *has been*) *swollen over past week*, where the time phrase adjoins the predicate. Another step might take *swollen* to its verbal source *swell*, where the result would assert that the swelling occurred over the past week.

Several thousands of patterns are stored in a compact notation in 'Selection Lists' that are used in selection restrictions (the filtering tests). Selection patterns are stored for each individual preposition. Some entries from the stored authenticated pattern occurrences for *P = over* are shown in Figure 7.

Selection patterns are also helpful in resolving lexical ambiguity such as occurs when a word has several sublanguage class assignments in the dictionary, e.g. *discharge* H-TTGEN/H-INDIC (*discharge from hospital* vs. *discharge from nose*). There is a stored pattern H-INDIC *from* H-PTPART, but no stored pattern H-INDIC *from* H-INST, so in an occurrence of *discharge from nose*, *discharge* will be stripped of its H-TTGEN class, and *discharge* will be treated by the information formatting procedure as an H-INDIC word.

4.3.3 *Forms of output*

Figure 8 shows the principal output of the information-formatting component of the MLP. This output represents the results of converting the parse tree to a medical representation composed of Information Format (IF) occurrences and connectives. Each IF occurrence corresponds to a statement type of the

SUBLANGUAGE CO-OCCURRENCE TABLE
Approved HOST-P-N for preposition "OVER"

Layout of table:
— Column 1: Pattern **name** and frequencies [*n:m*],
— *n* : frequency of same exact word cooccurrences in row;
m : frequency of sublanguage class cooccurrences in row.
— Columns 2-3-4: Words and their sublanguage classes;
— Column 5: Sentence ID and source text.

Pattern	HOST	P	N	Sentence ID
HOST-P-N [1:10]	**spiders** N:H-INDIC	**over** P:OVER	**extremities** N: H-PTPART	*SID=CPRIS 007.01D.01.06 there were very few **spiders over** the upper **extremities** .
HOST-P-N [1:1]	**centered** VEN: H-PTLOC	**over** P:OVER	**pubis** N: H-PTPART	*SID=991121 098.36F.01.06 she is to return again 11/19/1999 for her six month follow up , with ap pelvis **centered over** the **pubis** , and ap and lateral of the left hip .
HOST-P-N [1:1]	**inversion** N:H-TXRES	**over** P:OVER	**precordium** N: H-PTPART	*SID=CABG1 051B.1.07 there was t wave **inversion over** the anterior **precordium** and t wave flattening laterally which was new compared to an electrocardiogram done approximately one month earlier .
HOST-P-N [1:1]	**syncope** N:H-INDIC	**over** P:OVER	**winter** N: H-TMLOC	*SID=CPRIS 006.01E.01.03 due to his rhythm problems , as well as a history of near **syncope over** the **winter** , we will admit him to the hospital for further evaluation of his arrhythmia and the need for possible permanent pacemaker placement .
HOST-P-N [1:2]	**recover** V:H-RESP	**over** P:OVER	**five to ten minutes** QN: NTIME1	*SID=MGHPT 005A.02.02 at that time , without warning , she would fall and have generalized tonic-clonic movements with accompanying loss of consciousness from which she would **recover over** the next **five to ten minutes** .

Figure 7: Approved selection pattern occurrences

```
**************************************
**************************************
*SID=990318P2 098.20B.03.02
[ HISTORY-OF-PRESENT-ILLNESS ] Today , she has no cough , chest pain , or
shortness of breath .

(CONNECTIVE (CONJOINED (CONN = , <',':()> )))

(PATIENT-STATE-IF
   (PT-DEMOG (GENDER = [FEMALE] <GRAM-NODE:(FEM)> ))
   (SUBJECT = she <PRO:(H-PT)> )
   (VERB = has <TV:(VHAVE)>
      (EVENT-TIME (REF-PT = Today <N:(NTIME2)> , <',':()> ))
      (TENSE = [PRESENT] <GRAM-NODE:(H-VTENSE)> ))
   (PSTATE-DATA
      (S-S = cough <N:(H-INDIC)>
         (MODS (NEG = no <T:(H-NEG)> ))))
   (TEXTPLUS = ))

(CONNECTIVE (CONJOINED (CONN = AND <'AND':()> )))

(PATIENT-STATE-IF
   (PT-DEMOG (GENDER (GENDER = [FEMALE] <GRAM-NODE:(FEM)> ))
   (SUBJECT = = she <PRO:(H-PT)> )
   (VERB = has <TV:(VHAVE)>
      (EVENT-TIME (REF-PT = Today <N:(NTIME2)> , <',':()> ))
      (TENSE = [PRESENT] <GRAM-NODE:(H-VTENSE)> ))
   (PSTATE-SUBJ (PTPART = chest <N:(H-PTPART)> ))
   (PSTATE-DATA
      (S-S = pain <N:(H-INDIC)>
         (MODS (NEG = no <T:(H-NEG)> ))))
   (TEXTPLUS = ))

(PATIENT-STATE-IF
   (PT-DEMOG (GENDER (GENDER = [FEMALE] <GRAM-NODE:(FEM)> ))
   (SUBJECT = = she <PRO:(H-PT)> )
   (VERB = has <TV:(VHAVE)>
      (EVENT-TIME (REF-PT = Today <N:(NTIME2)> , <',':()> ))
      (TENSE = [PRESENT] <GRAM-NODE:(H-VTENSE)> ))
   (PSTATE-DATA (S-S = shortness of breath <N:(H-INDIC)>
      (MODS (NEG = no <T:(H-NEG)> ))))
   (TEXTPLUS = ))
```

Figure 8. Output of the MLP system

sublanguage and constitutes a basic unit of healthcare information.

In the parenthesized information-format tree display in Figure 8, only non-empty elements of the definitions are shown. The node names that are not obvious are:

*SID=	A unique sentence identification number
[HISTORY-OF-PRESENT-ILLNESS]	A section reference
CONNECTIVE	A node that connects two following IFs (Polish notation)
CONJOINED	A type of connective
CONN	A connective word
PATIENT-STATE-IF	An information format type, in this case, Patient-State
PT-DEMOG	Patient demographic information referred to in the sentence
GENDER	The gender of patient
SUBJECT	A grammatical subject (if not otherwise assigned)
VERB	A grammatical verb (if not otherwise assigned)
EVENT-TIME	A chronology modifier of the reported event
REF-PT	A time reference point
TENSE	The tense of the sentence verb
PSTATE-DATA	Data of the patient state
S-S	Signs and symptoms
MODS	Modifiers
NEG	A negative modifier
PTPART	A body part
TEXTPLUS	Words not included in IF

English parts of speech (or generated placeholder GRAM-NODE) and Healtcare-sublanguage lexical attributes are indicated by angle brackets: <GRAM-NODE:(FEM)>, <PRO:(H-PT)>, <TV:(VHAVE)>, <GRAM-NODE:(H-VTENSE>, <N:(NTIME2)>, <N:(H-INDIC)>, <N:(H-PTPART)>, <T:(H-NEG)>. Values generated by the MLP grammar are [PRESENT] (from verb *has*), and [FEMALE] (from pronoun *she*).

Depending on the type of applications, the MLP output is converted from the IF form into a simple table or XML trees, as follows:

– *A simple 2-dimensional table.* Each row corresponds to one IF occurrence and has the following 35 fields: the sentence SID (1 field), the section of the document (1 field), the number of this IF in this sentence (1 field), how it is

connected to other IFs in the same sentence (3 fields), the NIMPH marking for this IF (1 field) (see 4.3.4, below), and a flat layout of the major data points of the IF (remaining fields). For example, the 3 IFs from Figure 8 are presented in 3 rows. The symptom phrases (e.g. *no cough*, *no pain* and *no shortness of breath*) are housed in the fields Negation (NEG = *no*) and fields Sign-Symptom (S-S = *cough*, S-S = *pain*, and S-S = *shortness of breath*). Studies such as Healthcare Quality Assurrance, (5.1 below) were done using the database management systems INGRES and Informix, and web-based HTML (HyperText Markup Language) (Sager *et al.* 1996).

– *XML-trees*. This is another variation of the IF trees (Figure 9), fully equivalent to the ones in Figure 8. XML (eXtensible Markup Language) is a representation formalism which is part of a web-based 'family of technologies' (see W3C:XML 1999). XML promises flexibility in representation and presentation of information. Using XML, the original text after MLP is tagged with lexical and syntactic information. However, this is not just another variation of the IF trees. It is a richer representation where each node is now capable of housing attribute information.

In the XML representation, each node in the IF is represented as one tagged item (opening with '<tag>' and closing with '</tag>'); each unit of lexical information at a terminal node is represented as a triple consisting of one category tag, followed by sublanguage word class tags, followed by the word (where 'word' here stands for the word or phrase at the terminal node). For example, the phrase *no cough* in the IF tree is represented as follows:

<S-S>
<NEG><T><(H-NEG)>*no*</(H-NEG)></T></NEG>
<N><(H-INDIC)>*cough*</(H-INDIC)></N>
</S-S>

Here, <S-S>, <N>, etc. are opening tags, and </S-S>, </N>, etc. are closing tags.

Furthermore, it allows an application to extract data by scanning the MLP IF output. For example, the extraction of sign-symptom information in the first XML IF-tree of Figure 9 is accomplished by scanning from left to right and picking up everything between <S-S> and </S-S>, i.e. *no cough*, within the context of one IF, that is, between <PATIENT-STATE-IF> and </PATIENT-STATE-IF>.

This technology allows the designer to embed any number of tags that need not be seen by the user but can direct the retrieval and display of content

```
*SID=990318P2 098.20B.03.02
[ HISTORY-OF-PRESENT-ILLNESS ] Today , she has no cough , chest pain , or shortness of breath .

<CONNECTIVE><CONJOINED>
                    <CONN><',>,</',></CONN></CONJOINED></CONNECTIVE>
<PATIENT-STATE-IF>
   <PT-DEMOG><GENDER><GRAM-NODE><(FEM)>
                                        [FEMALE]</(FEM)></GRAM-NODE></GENDER></PT-DEMOG>
   <SUBJECT><PRO><(H-PT)>she</(H-PT)></PRO></SUBJECT>
   <VERB><TV><(VHAVE)>has</(VHAVE)></TV>
      <EVENT-TIME><REF-PT>
         <N><(NTIME2)>Today</(NTIME2)></N> <',>,</',></REF-PT></EVENT-TIME>
      <TENSE>
         <GRAM-NODE><(H-VTENSE)>[PRESENT]</(H-VTENSE)></GRAM-NODE>
         </TENSE></VERB>
   <PSTATE-DATA><S-S><N><(H-INDIC)>cough</(H-INDIC)></N>
               <MODS><NEG> <T><(H-NEG)>no</(H-NEG)></T></NEG></MODS>
            </S-S></PSTATE-DATA>
   <TEXTPLUS></TEXTPLUS></PATIENT-STATE-IF>
<CONNECTIVE><CONJOINED>
                  <CONN><'AND'>AND</'AND'></CONN></CONJOINED></CONNECTIVE>
<PATIENT-STATE-IF>
   <PT-DEMOG><GENDER><GRAM-NODE><(FEM)>
                                        [FEMALE]</(FEM)></GRAM-NODE></GENDER></PT-DEMOG>
   <SUBJECT> <PRO><(H-PT)>she</(H-PT)></PRO></SUBJECT>
   <VERB> <TV><(VHAVE)>has</(VHAVE)></TV>
      <EVENT-TIME><REF-PT>
         <N><(NTIME2)>Today</(NTIME2)></N> <',>,</',></REF-PT></EVENT-TIME>
      <TENSE>
         <GRAM-NODE><(H-VTENSE)>[PRESENT]</(H-VTENSE)></GRAM-NODE>
         </TENSE></VERB>
   <PSTATE-SUBJ><PTPART>
         <N><(H-PTPART)>chest</(H-PTPART)></N></PTPART></PSTATE-SUBJ>
   <PSTATE-DATA><S-S> <N><(H-INDIC)>pain</(H-INDIC)></N>
            <MODS> <NEG><T><(H-NEG)>no</(H-NEG)></T></NEG></MODS>
         </S-S></PSTATE-DATA>
   <TEXTPLUS></TEXTPLUS></PATIENT-STATE-IF>
<PATIENT-STATE-IF>
   <PT-DEMOG><GENDER><GRAM-NODE><(FEM)>
                                        [FEMALE]</(FEM)></GRAM-NODE></GENDER></PT-DEMOG>
   <SUBJECT><PRO><(H-PT)>she</(H-PT)></PRO></SUBJECT>
   <VERB> <TV><(VHAVE)>has</(VHAVE)></TV>
      <EVENT-TIME><REF-PT>
         <N><(NTIME2)>Today</(NTIME2)></N> <',>,</',></REF-PT></EVENT-TIME>
      <TENSE>
         <GRAM-NODE><(H-VTENSE)>[PRESENT]</(H-VTENSE)></GRAM-NODE>
         </TENSE></VERB>
   <PSTATE-DATA><S-S><N><(H-INDIC)>shortness of breath</(H-INDIC)></N>
               <MODS><NEG> <T><(H-NEG)>no</(H-NEG)></T></NEG></MODS>
            </S-S></PSTATE-DATA>
   <TEXTPLUS></TEXTPLUS></PATIENT-STATE-IF>
```

Figure 9. XML output of the MLP system

It has made it possible to add medical knowledge to the MLP output, as described in Section 5.2 below.

4.3.4 *Quality control of MLP*

One of the bars to the use of NLP is the recognition that the very flexibility that gives language its widespread utility makes it difficult to ensure that a computer representation arrived at via NLP has captured the intended meaning. At the least, a control of the output in relation to the target representation is essential. To that end, in the case of medical language processing, the LSP-MLP system includes an error-detection program that is applied to each Information Format and Connective in the MLP output. The program is called NIMPH for the 5 types of problems it monitors: *N* for possible mis-analysis of Negation; *I* for Ill-formed semantic output (wrong assigning of subclass occurrence to Information Format slot); *M* for possible mis-analysis of a Modal word; *P* for Partial parse (a correct analysis of an *ASSERTION* or *FRAGMENT* up to a point in the sentence, not the end); *H* for total HangUp (no parse returned).

After each processing run, a report is issued that includes the NIMPH numbers as well as a breakdown of problems by component. In the case of failures of Selection filters, a separate report is issued so that the failures can be evaluated. A Selection failure may be due to the absence of a pattern that should be added to the grammar; it may be due to a mistake in the classification of a word (dictionary error); or it may signal some other problem in the processing.

5. Validation and application

Different objectives can motivate the development of computer programs for language analysis. One objective might be to test the validity of a theory of grammar. For this, one develops a parsing program and writes a grammar, with associated dictionary, based on the theory. If a representative sample of sentences is correctly parsed by such a system, one can claim that up to some level of detail incorporated in the grammar, language structure is 'computable' using this theory.

The initial motivation for developing the Linguistic String Analysis program was of this type. In the grammar of (Sager 1981) great attention was paid to many forms, particularly those involving deep nesting and zeroing,

that would not likely occur in most texts but are possible in the English language. The goal was to 'prove' that Linguistic String Analysis was an effective grammatical formulation for the analysis of English sentences.

Harris's theories of language structure do not need computer programs to validate them. His was a different style wherein the theory emerged from a great deal of sentence analysis in which problems were anticipated and dealt with in great detail. And in later work, such as the grammar in (Harris 1982), the analysis is far deeper than what we are in a position to compute today. The string analysis experiment was fitting at a time when there were serious claims that natural language (even just syntactic parsing of sentences) was beyond the reach of machine analysis.

Another motivation for developing computerized language processing is practical. Assuming that such computer programs can be written, can they be made to provide some useful service? This might be considered another type of validation of linguistic analysis, the 'proof of the pudding' type. Whether or not applications are seen as validations of the theory that underlies the linguistic processing, they have their own standing in the larger world. The goal of developing practical applications has driven much of the work in NLP since the early days.

In particular, work on the medical sublanguage by the LSP group has been strongly motivated toward finding useful applications in patient care and related activities. Two examples are given here.

5.1 Healthcare quality assurance

The need to monitor the quality of healthcare that is delivered to patients has been recognized for a long time, but with the recent radical changes to the U.S. healthcare delivery system the issue has become prominent. One of the obstacles to such monitoring is the difficulty of obtaining the data it requires, and, as a prerequisite to that, the specification of what data are required. A step in that direction was made by the National Committee for Quality Assurance by defining a minimal set, called the Health Plan Employer Data and Information Set (HEDIS), for a number of medical conditions.

One of the HEDIS measures concerned whether patients who had suffered a heart attack (acute myocardial infarction, AMI) received beta blocker medication, which was considered desirable unless they had a contraindication as specified in the measure ("Beta blocker treatment after a heart attack", HEDIS 3.0/1998, Volume 2).

To test whether MLP applied to hospital discharge summaries could extract data pertaining to the HEDIS Beta blocker measure, an experiment was performed in which 95 discharge summaries that had been coded by a particular hospital for a diagnosis of AMI were processed by the MLP. The output was mapped to a relational database table (one information format to one row) and retrieval queries were written to extract the rows with pertinent data.

Figure 10 summarizes the experiment and the retrieval results. Figure 11 shows a portion of the combined table of results for the following two queries:

Was the patient given a beta blocker medication?
Did the patient have any contraindications?

HEDIS MEASURE
"Beta blocker treatment after a heart attack (AMI)"

- 95 discharge summaries of patients whose diagnosis had been coded by the hospital as **Acute Myocardial Infarction** (AMI), ICD-9-CM code 410.01 - 410.91
- These discharge summaries had been divided into Sections, such as
  HISTORY OF PRESENT ILLNESS
  PAST MEDICAL HISTORY
  PHYSICAL EXAMINATION
  LABORATORY DATA
  HOSPITAL COURSE
  DISCHARGE STATUS, etc.
- These discharge summaries were analyzed by the Medical Language Processor. Retrieval was performed on the MLP output.
  1. Was the patient given a beta blocker medication?
  2. Did the patient have any contraindications?
- Summary of Retrieval Results:

  – TOTAL NUMBER OF PATIENTS: 95
  – Patients not considered:
    – Under 35: |009|024|
    – Incomplete Documents: |048|093|
  – Results from database queries:

	Beta Blocker Given	Beta Blocker Not Given	Total
With contra-indications	42	19	61
Without contraindications	28	2	30
Total	70	21	91

Figure 10. HEDIS retrieval from MLP output

HEDIS MEASURE
"Beta blocker treatment after a heart attack (AMI)"
from database queries over data
structured by Medical Language Processing

QUERY 2A: **Patients given beta blockers who have contraindications**
Number of patients: 42
List of patients: 003 005 006 008 010 015 016 018 020 027 030 033 034 036 037 038 041 042 043 045 049 051 055 059 061 062 063 064 068 069 070 071 072 073 076 079 084 085 089 091 092 094

Sent ID	Beta Blockers	Contraindications	Time
003B.06.02 5	LOPRESSOR		
003D.04.01 1		SINUS BRADY-CARDIA AT 55	
003F.01.03 1	LOPRESSOR		
005B.04.02 1	TENORMIN		
005E.01.04 1		MILD RIGHT VENTRICULAR DIASTOLIC DYSFUNCTION	
005E.02.02 5	WITH # LOPRESSOR #		
005E.03.01 4	WITH # LOPRESSOR #		
005F.03.04 1	LOPRESSOR		
006B.02.08 4	LOPRESSOR		
006B.04.03 1		COMPLETE HEART BLOCK	SUBSEQUENTLY #
006B.04.04 2		SINUS BRADY-CARDIA	
006B.05.02 1	LOPRESSOR		

Figure 11. 'Snapshot' of HEDIS retrieval output

It may be considered surprising that of the 91 patient documents that qualified for review, 42 indicated that patients received beta blocker even though they had contraindications. Many of these contraindications (in 29 patients) were congestive heart failure. It was reported during 1997, just one year before the edition of the HEDIS measures available for the experiment, that beta blockers reduce deaths from congestive heart failure. Possibly clinical practice was ahead of the measure.

5.2 Access to narrative data

One of the key problems facing clinicians is access to the right information, at the right time, organized in the optimal way for management of the specific clinical question to be addressed. Effective, high quality care depends on the ability to access, review, and interpret a large amount of information on a given patient as part of the decision making process. Due to cost and time constraints, attempts to have clinicians structure their clinical documentation in order to facilitate this process have been largely unsuccessful, despite the apparent benefits. Consequently, the vast majority of clinical information has remained locked within dictated medical notes, unavailable for retrieval and efficient review. The use of MLP, enriched with medical knowledge, may help to address this problem.

5.2.1 *Adding medical knowledge to MLP*

Currently, there is under development an XML-based medical terminology which can be used to enrich the medical representation obtained by the MLP. The Structured Health Markup Language (SHML) is an organized, highly specialized set of tags that are aimed at describing the medical content of terms encountered in medical text. More than 40 distinct SHML categories have been created, each a description of medical content in patient documents, and each with multiple subcategories. Thus, conceptually, the phrase *pneumonia, right lower lobe, superior, due to Klebsiella* is tagged in XML-based SHML format as

<diagnosis> *Pneumonia* ,
    <location> *right lower lobe* </location> ,
    <position> *superior* </position> ,
    <link> *due to*
        <org> **Klebsiella** </org>
    </link>
</diagnosis>

Table 4. SHML tag system–correspondence of the anatomic structure and body region hierarchies

Description	SHML Tag Class	Tag Class	Description
anatomic system	a-s	b-r	body region
neurologic system	a-s_nr	b-r_h-n_h	head-neck head
central nervous system	a-s_nr_cns	b-r_h-n_h	head-neck head
brain	a-s_nr_cns_brn	b-r_h-n_h	head-neck head
cardiovascular system	a-s_cv	b-r	body region
heart	a-s_cv_hrt	b-r_tk_thx_msty	mediastinum
chest	a-s	b-r_tk_thx	trunk thorax
respiratory system	a-s_rsp	b-r_tk_thx	trunk thorax
upper respiratory tract	a-s_rsp_u-r	b-r_tk_thx	trunk thorax
lower respiratory tract	a-s_rsp_l-r	b-r_tk_thx	trunk thorax
lung	a-s_rsp_l-r_lng	b-r_tk_thx	trunk thorax
stomach	a-s_gi_gi-tr_u-gi_stm	b-r_tk_thx	trunk thorax

SHML defines several vectors of description of a term found in medical text. Major vectors include sign-symptoms, diagnoses, procedures, organisms, allergies, social behaviors, activities, medications, chemicals, persons, demographics, etc., besides time (frequency, repetition, event-time,[. . .]), links (connective, preposition,[. . .]), modifiers (certainty, negation, changes, amounts,[. . .]).

A term in SHML contains several hierarchical vectors, the first of which is the principal tag, and two of which are always anatomic structure and body region, as shown in Table 4. Thus, terms like *cough* and *shortness of breath* as *N* (noun) and H-INDIC are tagged as

<s-s><a-s_rsp><b-r_tk_thx>
cough
</b-r_tk_thx></a-s_rsp></s-s>

<s-s><a-s_rsp><a-s_cv_hrt><b-r_tk_thx>
shortness of breath
</b-r_tk_thx></a-s_cv_hrt></a-s_rsp></s-s>

This says that

– *Cough* is a sign-symptom, associated with respiratory system, and thorax (in body region trunk).
– *Shortness of breath* is a sign-symptom, associated with both the respiratory system and the [cardiovascular] heart, and the thorax (in body region trunk).

An MLP-SHML correspondence dictionary has been established which currently numbers over 64,000 row entries. Each entry in this dictionary is a row, which is currently defined by one unique MLP triple consisting of a term (word, or several words treated as an idiom), one of its MLP categories, and one of its MLP sublanguage classes. A term having more than one MLP category is represented in more than one row; a term having more than one MLP sublanguage class is represented in more than one row. Thus, every MLP lexical ambiguity is made explicit so that the SHML tag corresponding to each meaning can be unambiguously assigned. Each entry contains:

– the term
– two fields: an MLP category and an MLP class
– SHML tags in 4 fields, laid out as a multi-vector description of the term

SHML is here used as an extension to the MLP in which each triplet of term, MLP category, and MLP sublanguage class defines one unique entity (i.e. one entry).

5.2.2 *A browser for medical narrative data*

The combined MLP-SHML representation of clinical narrative supplies a richly textured clinical data store obtained by linguistic processing and medical tagging of free text patient documents. It remains to make the results selectively viewable by the clinical (or administrative) user. To provide this function, a prototype browser has been developed by InContext Data Systems, Inc. using a relational database system, and HTML and XML web technologies. This is an attempt to integrate different technologies into a system for flexible access to pertinent medical data (Figure 12).

Input to the relational database includes only a preprocessed source medical text, its SHML-tagged MLP output, and an administrative section of the source text. All interchanges between the MLP and the browser are done in ASCII format. The information format (IF) generated by the MLP, now enhanced with medical knowledge from SHML classes, is called a health information unit, or HIU.

The HIU table is then indexed for major SHML tags, such as Signs and Symptoms, Diagnoses, Vital Signs, Labs, Procedures, Medications, Patient Social Behaviors, etc. which can be further sorted by Anatomic System, Body Region, Chemical Classes, and other categories.

To illustrate how the user might access analyzed narrative patient data using the Browser, Figure 13 shows a snapshot of the Browser using the 'Signs

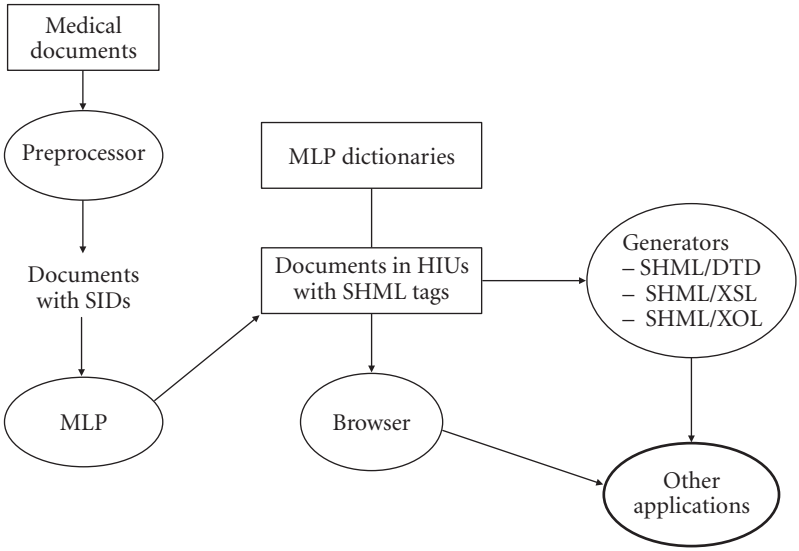


Figure 12. MLP and SHML linkage

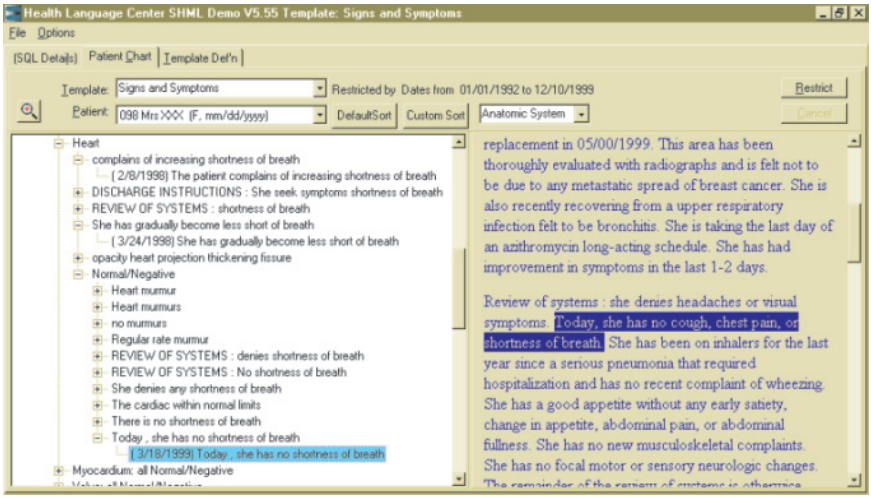


Figure 13.

and Symptoms' template, custom sorted by 'Anatomic System', to present the 'Patient Chart' for Patient 098, Mrs. XXX, female, born *mm/dd/yyyy*, for whom there are 36 documents in the system. There are 544 HIUs found, each tagged with the date of visit. Looking under 'Heart' and then under 'Normal/Negative' subbranch, we find the HIU *Today , she has no shortness of breath*. This HIU is highlighted together with the source text of the sentence, the same sentence as shown in Figure 8 and Figure 9. Note that the HIU containing *shortness of breath* is shown here, correctly, because *shortness of breath* has the SHML anatomic tag <as_cv_hrt> (i.e. 'heart' of the anatomic cardiovascular system). The MLP-SHML tagged form of this HIU is shown in Figure 14.

```
<SID id="990318P2 098.20B.03.02">
<!-- Row 3: Today , she has no shortness of breath . -->

<PATIENT-STATE-HIU id="990318P2 098.20B.03.02"
                   sect="REVIEW OF SYSTEMS" row="3" ARG="2" CONNid="2">
  <PT-DEMOG><GENDER>[FEMALE]</GENDER></PT-DEMOG>
  <SUBJECT> <per><_SD14065>she</_SD14065></per> </SUBJECT>
  <VERB><li><li_vhv><_SD7168>has</_SD7168></li_vhv></li>
    <EVENT-TIME>
      <REF-PT>
        <tm><tm_loc><_SD4152>Today</_SD4152></tm_loc></tm>
        ,
        </REF-PT></EVENT-TIME>
    <TENSE>[PRESENT]</TENSE></VERB>
  <PSTATE-DATA>
    <SIGN-SYMP>
      <s-s><a-s_rsp><a-s_cv_hrt><b-r_tk_thx><_SD7917>
        shortness of breath
          </_SD7917></b-r_tk_thx></a-s_cv_hrt></a-s_rsp></s-s>
        <MODS><NEG>
          <md><md_ng><_SD3440>no</_SD3440></md_ng></md>
        </NEG></MODS>
      </SIGN-SYMP>
  </PSTATE-DATA>
</PATIENT-STATE-HIU>
```

Figure 14. An SHML-tagged health information unit

According to the correspondence of the anatomic structure and body region table (Table 4), the HIU *Today, she has no chest pain* is also retrieved as a 'Normal/Negative' finding related to heart. In this case *pain* is a non-specific symptom, and *chest* is in a body region thorax, which contains the heart (<a-s_cv_hrt>).

By contrast, if one selects 'Custom Sort' by 'Body Region', the display area will show 544 HIUs organized under 'Body Region'. We will find under 'Thorax' and then under 'Normal/Negative' subbranch, the three HIUs shown in Figure 8 and Figure 9, because all three terms *cough*, *chest pain*, and *shortness of breath* have the 'supporting' SHML tag <b-r_tk_thx> (for the thorax in the trunk body region).

In Figure 13, two tabs are concealed by the 'Patient Chart' tab: 'Template Def'n' and '(SQL Details)'. The 'Template Def'n' tab displays two subwindows. The left window presents the current SHML tag set and their hierarchies; the right window is a template building window. By dragging tags from the left window to the right one, the user can build new queries. Retrievals of these queries are displayed on the 'Patient Chart' tab. The '(SQL Details)' tab, for debugging purpose, displays SQL database queries translated from the right 'Template Def'n' subwindow.

The Browser, using SHML-tagged MLP formatted output of original natural language text, enables physicians to (a) create templates best suited for their particular view of patient information from actual documents, (b) see the selected units of information in the context of the original documents for verification, and (c) study patterns across the entire set of patient documents.

6. Summary and conclusion

Harris's string analysis, transformations and the sublanguage method provide a sound basis for language computation, particularly as the basis for representing the information content of scientific and other fact-reporting texts.

In this chapter we have summarized an experience of building upon this basis to arrive at an operational 'real world' system, a medical language processor that can help healthcare workers obtain the data they need from narrative reports.

This effort has been singular in several respects which may not recur. Much of the linguistic input (e.g. the dictionary) was developed manually, demanding a great amount of human resources. We were fortunate that the

project began in a period when the Federal government was still supporting long-range development efforts, and funding was forthcoming from the National Science Foundation and the National Institutes of Health. We were also fortunate in having highly skilled labor contributed on a voluntary basis by persons who believed in the goals of the project.

At the same time, because of the early origin and long history of this work, computer tools that could lighten the burden were not always available as they are now in many places. In general, as the computer field advances, new ways of doing old, still needed, tasks are developed and new tasks for new goals emerge. It is likely that the need for information that is recorded in natural language will not disappear, so there is hope that the methods of language analysis that marked Harris's oeuvre will find their application in the future of language technology, along with their proper place in the history of the field of linguistics.

Acknowledgements

First and foremost we wish to acknowledge the essential contribution of Margaret Lyman, M.D., to the development of the medical language processor. In addition to her clinical practice, from 1977 until her death in November 2000, Dr. Lyman was undaunted in her dedication to this effort. She believed in the importance of the freely dictated medical report and in the possibility of using computers to organize and make accessible its content for the betterment of patient care and the advancement of medical knowledge. Dr. Lyman was a tireless worker and one who inspired all of us toward greater effort and dedication. We cannot match her in either of these, but we hope to realize at least some of her goals. Dr. Leo Tick provided technical support for Dr. Lyman's activities and also reimplemented, improved, and maintained the parser; he added much to the development of the medical language processor.

Over the years of the Linguistic String Project many coworkers participated in its activities. It is not possible to acknowledge them all, but some (in alphabetical order) are: Barbara Anderson, Jeff Bary, Beatrice Bookchin, Shiun Chen, Emile Chi, Pascale Claris, Judith Clifford, Eileen Fitzpatrick, Carol Foster, Carol Friedman, Dan Gordon, Ralph Grishman, Lynette Hirschman, Stephen Johnson, Michiko Kosaka, Joyce London, Catherine MacLeod, Elaine Marsh, James Morris, Neal Oliver, Morris Salkoff, Richard Schoen, Guy Story, Susanne Wolff, and Su Yun.

References

Bloomfield, Leonard. 1933. *Language*. New York: Holt, Rinehart & Winston.

Gleitman, Lila R. 1959. "Word and word-complex dictionaries". Transformations and Discourse Analysis Project (TDAP) 16. Philadelphia: Department of Linguistics, The University of Pennsylvania.

Grishman, Ralph & Richard Kittredge (eds.). 1986. *Analyzing Language in Restricted Domains: Sublanguage description and processing*. Hillsdale, NJ: Lawrence Erlbaum.

Harris, Zellig S. 1952. "Discourse analysis". *Language* 28.1: 1–30.

Harris, Zellig S. 1959. "Computable syntactic analysis". Transformations and Discourse Analysis Project (TDAP) 15. Philadelphia: Department of Linguistics, The University of Pennsylvania.

Harris, Zellig S. 1962a. *String Analysis of Sentence Structure*. (= *Papers on Formal Linguistics*, No. 1.) The Hague: Mouton, 70 pp. (Repr., 1964 and 1965.) [Revised version of Harris 1959.]

Harris, Z.S. 1962b. "A cycling cancellation-automaton for sentence well-formedness". Transformations and Discourse Analysis Project (TDAP) 51. Philadelphia: Department of Linguistics, The University of Pennsylvania. [Also in *International Computation Centre Bulletin* 5(1966): 69–94. Reprinted in Harris (1970: 286–309).]

Harris, Zellig S. 1963. *Discourse Analysis Reprints*. The Hague: Mouton & Co.

Harris, Zellig S. 1967. "Decomposition lattices". Transformations and Discourse Analysis Project (TDAP) 70. Philadelphia: Department of Linguistics, The University of Pennsylvania. [Reprinted in Harris (1970: 578–602).]

Harris, Zellig S. 1968. *Mathematical Structures of Language*. New York: John Wiley/ Interscience Publishers.

Harris, Zellig S. 1970. *Papers in Structural and Transformational Linguistics*. Dordrecht: D. Reidel Publishing Company.

Harris, Zellig S. 1982. *A Grammar of English on Mathematical Principles*. New York: John Wiley & Sons.

Harris, Zellig S. 1991. *A Theory of Language and Information: A mathematical approach*. Oxford: Clarendon Press.

Harris, Zellig S., Michael Gottfried, Thomas Ryckman, Paul Mattick, Jr., Anne Daladier, Tzvee N. Harris, & Suzanna Harris. 1989. *The Form of Information in Science: Analysis of an immunology sublanguage*. Preface by Hilary Putnam. (= *Boston Studies in the Philosophy of, Science*, 104). Dordrecht: Kluwer Academic Publishers.

Hirschman, Lynette, Ralph Grishman, & Naomi Sager. 1975. "Grammatically-based automatic word class formation". *Information Processing and Management II*, pp. 39–57.

Joshi, Aravind K. 1960. "Recognition of local substrings". Transformations and Discourse Analysis Project (TDAP) 18. Philadelphia: Department of Linguistics, The University of Pennsylvania.

Joshi, Aravind K. & Philip D. Hopely. 1999. "A parser from antiquity: An early application of finite state transducers to natural language parsing", in *Extended Finite State Models of Language* (Proceedings of the ECAI 96 Workshop), ed. by Andras Kornai, 6–15. New York & London: Cambridge University Press.

Kaufmann, Bruria. 1959. "Right-order substrings and wellformedness". Transformations and Discourse Analysis Project (TDAP) 19. Philadelphia: Department of Linguistics, The University of Pennsylvania.

Karttunen, Lauri. 1997. "Comments on Joshi", in *Extended Finite State Models of Language* (Proceedings of the ECAI 96 Workshop), edited by Andras Kornai. New York & London: Cambridge University Press.

Kittredge, Richard, & Jack Lehrberger. 1982. *Sublanguage: Studies of language in restricted semantic domains*. Berlin: Walter de Gruyter.

Marsh, E., and Friedman, C. 1985. Transporting the Linguistic String Project system from a medical to a Navy domain, *ACM Transactions on Office Information Systems* 3:2, pp. 121–140.

Morris, J. 1965. The IPL string analysis program, in *First Report of the String Analysis Programs*, Philadelphia: Department of Linguistics, The University of Pennsylvania.

Nazarenko, A., P. Zweigenbaum, B. Habert, & J. Bouaud. To appear. "Corpus-based extension of a terminological semantic lexicon", in *Recent Advances in Computational Terminology*, ed. by D. Bourigault, C. Jacquemin, & M.-C. L'Homme. Amsterdam: John Benjamins.

Nhàn, Ngô Thanh, Naomi Sager, M. Lyman, L. J. Tick, F. Borst, & Y. Su. 1989. "A medical language processor for two Indo-European languages". *Proceedings of the 13th Annual Symposium on Computer Application in Medical Care* (L.C. Kingsland, ed.), 554–558. Washington, D.C.: IEEE Computer Society Press.

Oliver, N. 1992. *Sublanguage Based Medical Information Processing System for German*. Ph.D. thesis, New York University.

Sager, Naomi. 1959. "Elimination of alternative classification". Transformations and Discourse Analysis Project (TDAP) 17. Philadelphia: Department of Linguistics, The University of Pennsylvania.

Sager, Naomi. 1960. "Procedure for left-to-right recognition of sentence structure", in Transformations and Discourse Analysis Project (TDAP) 27. Department of Linguistics, The University of Pennsylvania.

Sager, Naomi. 1967. "Syntactic analysis of natural language". *Advances in Computers 8*, 153–188. New York: Academic Press.

Sager, Naomi, & Ralph Grishman. 1975. "The restriction language for computer grammars of natural language". *Communications of the ACM 18*, pp. 390–400.

Sager, Naomi. 1978. "Natural language information formatting: The automatic conversion of texts to a structured data base". In *Advances in Computers* 17, edited by M. C. Yovits, 89–162. New York: Academic Press.

Sager, Naomi. 1981. *Natural Language Information Processing: A computer grammar of English and its applications*. Reading, Massachusetts: Addison-Wesley.

Sager, Naomi, Carol Friedman, M. S. Lyman, MD, & members of the Linguistic String Project. 1987. *Medical Language Processing: Computer management of narrative data*. Reading, Massachusetts: Addison-Wesley.

Sager, Naomi, M. S. Lyman, C. Bucknall, N. T. Nhàn, & L. J. Tick. 1994. "Natural language processing and the representation of clinical data", *Journal of the American Medical Informatics Association*, 1.2:142–160.

Sager, Naomi, N.T. Nhàn, M.S. Lyman, & L.J. Tick. 1996. "Medical language processing with SGML display". *Proceedings of the 1996 AMIA Annual Fall Symposium*. Hanley & Belfus. Pp. 547–551.

Shapiro, P.A. 1967. "Acorn", in *Methods of Information in Medicine* 6:153-162.

Spyns, P., N.T. Nhàn, E. Baert, N. Sager, & G. De Moor. 1996. "Combining medical language processing and mark-up technology: An experiment applied to Dutch", in *Proceedings of MIC96*, edited by C. Sevens & G. De Moor, pp. 69–77. Brussels.

Wolff, S. 1984. "The use of morphosemantic regularities in the medical vocabulary for automatic lexical coding". *Methods of Information in Medicine*, 23:195–203.

The World Wide Web Consortium (W3C). 1999. "W3C technologies: XML". (http://www.w3.org). [See under XML, XML technical reports, XML Base, XML Encryption, XML Protocol, XML Query, XML Schema, XML Signature, etc.]

Chapter 5

Hierarchical structure and sentence description

Aravind K. Joshi
The University of Pennsylvania

1. Introduction

In many of his writings, Zellig Harris pursued the strategy of eschewing as much hierarchical structure as possible in describing sentence structure. In this chapter I will describe some research efforts that are in this general spirit. These are (1) the use of cascaded finite state transducers for sentence analysis, (2) subsequent work on string grammars, and (3) tree-adjoining grammars. The first two were part of the Transformations and Discourse Analysis Project (TDAP) directed by Zellig Harris from 1959 until about 1973. I have pursued the study of tree-adjoining grammars since about 1973. I will describe (1) and (3) in some detail and (2) only briefly. String grammars have been actively pursued by Naomi Sager with respect to their use in computational linguistics and information extraction. Formally, all these efforts share one aspect, namely, they can be seen as addressing (along with many other important issues, of course) the issue of the minimal hierarchical structure necessary for sentence description. The framework of tree-adjoining grammars allows us to cast the issue of minimal hierarchical structure necessary for sentence description in terms of the issue of minimal structure necessary for the elementary structures of the tree-adjoining grammars.

2. Cascaded finite state transducers for sentence analysis

A parsing program was designed and implemented at the University of Pennsylvania during the period from June 1958 to July 1959. This program was part of the Transformations and Discourse Analysis Project (TDAP) directed by

Zellig Harris.[1] The techniques used in this program, besides being influenced by a particular linguistic theory, arose out of the need to deal with the extremely limited computational resources available at that time. The program was essentially a cascade of finite state transducers (FSTs). To the best of our knowledge, this was the first application of FSTs to parsing. This program was reimplemented in 1997. The discussion below is based on Joshi & Hopely (1999), which describes this faithful reimplementation (named Uniparse).

The relevance of this program to the present topic is that Uniparse provides a relatively flat structure to a sentence. More precisely, each clause has a flat structure and modifiers (adjuncts) are not explicitly attached. Use of cascading finite state transducers allows the computation of embedded clauses (along with some other structures as will be clear later). The hierarchical structure between the embedding and embedded clauses is implicit. The minimal clauses are flat and the hierarchical structure at the clause level (if any) is kept separate (factored away) from the hierarchical structure implicit in the embeddings. This aspect of keeping the minimal clause structure separate from the structure of the modifiers and embeddings, recursion in general, is a characteristic feature of all the efforts mentioned above.

The program consisted of the following phases:

1. Dictionary look-up.
2. Replacement of some 'grammatical idioms' by a single part of speech.
3. Rule-based part of speech disambiguation.
4. A right-to-left FST composed with a left-to-right FST for computing 'simple noun phrases.'
5. A left-to-right FST for computing 'simple adjuncts' such as prepositional phrases and adverbial phrases.
6. A left-to-right FST for computing simple verb clusters.
7. A left-to-right 'FST' for computing clauses.

In Section 3 we will describe the different phases of the parser in some detail and also briefly discuss several aspects of the parser that have a close relation-

1. See Transformations and Discourse Analysis Project (TDAP) Reports, University of Pennsylvania, Reports, available in the Library of the National Institute of Science and Technology (NIST) (formerly known as the National Bureau of Standards (NBS)), Bethesda, MD. Relevant to this early work are reports 15–19: Harris (1959), Gleitman (1959), Sager (1959), Joshi (1959), and Kauffman (1959).

ship to some of the recent work on finite state transducers. An illustrative example in Section 4 shows the output of each phase.

3. Some details of uniparse

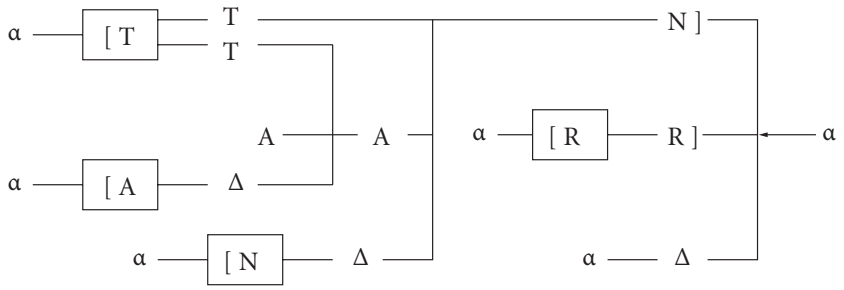
We will describe here the various phases of the parser, illustrated with a few examples for each phase.

Phase 1: Each word is assigned one or more parts of speech (POS). If a word is assigned more than one POS, then sometimes these are ranked, the less frequent POS first and then the next. Thus, for example, *show* has *N* ranked before *V* and *book* has *V* ranked before *N*. There are about 14 subcategorization frames for verbs. Since the Prepositional Phrases (PPs) are marked with the specific prepositions, there are effectively over 50 verb subcategorizations. The parser does not handle unknown words.

Phase 2: A 'grammatical idiom' such as *of course* is replaced by a single POS for adverb, *per cent* by POS for noun, etc. Certain 'grammatical idioms' such as *per cent* were replaced by a single POS with certainty. Others, such as *of course*, were replaced by a single POS with an indication that this replacement is tentative because, for example, *of* and *course* may belong to different phrases. For each 'grammatical idiom', one word of the idiom is marked as an index into the idiom-dictionary that specifies the local environment (words to the left and to the right of the index word). Phase 2 operations consist of finite transductions.

Phase 3: Rule-based disambiguation techniques are used for POS disambiguation. There are about 14 tests: *N*-eliminating tests, *V*-eliminating tests, etc. If the POS for a word are ordered, for example, for *show N* before *V*, then the *N*-eliminating tests are applied first. If they fail, then the *V*-eliminating tests are applied. If these also fail then the ambiguity remains. Most tests look for bounded contexts to the left and to the right; thus these are finite transductions. Some tests use contexts specified by simple (i.e., no Kleene-* over another Kleene-*) regular expressions and thus they are also finite-state transductions. The ordered set of tests, X - eliminating tests for a POS X, are cycled until no further disambiguations can be made.

Phases 4, 5 and 6: The strings (phrases) in these phases are called first-order strings as they do not involve proper nesting, as compared to the strings (clauses) computed in Phase 7 that are called second-order strings as they may involve proper nestings. The computation in Phase 7 is, strictly speaking, not a finite-state computation.



N: noun, R: pronoun, T: article, A: adjective,

Δ: any symbol other than those that appear above Δ.

Figure 1. A highly simplified diagram of an RL FST for simple noun phrases (original notation)

In Phase 4 a right-to-left (RL) FST is composed with a left-to-right (LR) FST for computing simple noun phrases. A highly simplified diagram of a RL FST for simple noun phrases in the original notation is shown in Figure 1. The sentence is scanned from right to left. A closing bracket] is placed as soon we meet a symbol (a POS) that allows us to enter the graph in Figure 1. The possible entrance points are N and R. Once we enter the graph, control flows according to the symbols encountered while scanning from right to left. If the graph terminates on a box then we place the opening bracket [; if the graph terminates on a POS symbol, we loop back to the place on the graph where that POS appeared before for the first time. Note that the longest-path strategy is built into the graph, in the sense that once we enter the graph we try to absorb as many of the POS symbols encountered as possible while traversing the graph. This longest-path strategy is also used (not shown in Figure 1) if a word is encountered with ambiguous POS. In this case the POS selected is the one that allows continuation of the phrase; however, information about the alternatives is left behind for later processing. Thus some unresolved POS ambiguities from Phase 1 are temporarily resolved during the computation of these first-order strings. Similarly, if a conjunction is encountered and the phrase can be continued by looking at one symbol to the left of the conjunction (for the RL FST) then it is continued.

If we represent the graph in Figure 1 in terms of the modern notation for sequential FST we obtain an RL FST as shown in Figure 2. This FST is strictly

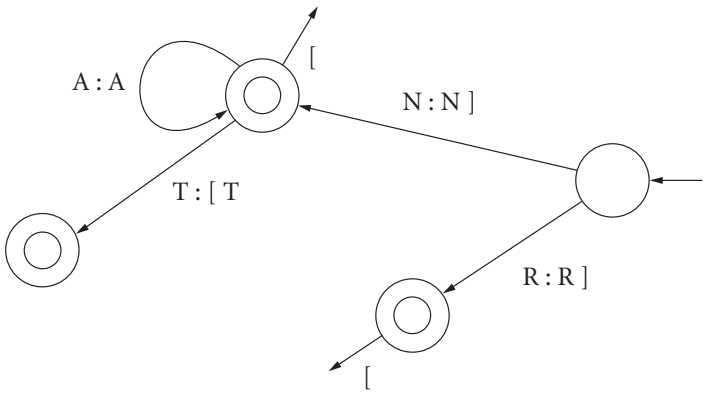


Figure 2. RL subsequential FST corresponding to Figure 1

not sequential. It is subsequential because upon exiting from some final states the FST produces an output.

This RL FST computes simple noun phrases, enclosed in [ ], whose rightmost end is determined. This transduction is followed by an LR FST that computes simple noun phrases such as *the rich* where the leftmost end is reliably determined. Of course, if the actual string was *the rich man* the first RL FST would have already enclosed this string in [ ], thus making the string *the rich* ignorable in the second LR FST. The total number of states in the RL FST is approximately 160 and in the LR FST approximately 50.

In Phase 5 a LR FST computes simple adjuncts such as prepositional phrases and adverbial phrases, enclosed in ( ), for example:

(*very clearly*)
(*rapidly*)
(*to* [*date*])
(*in* [*increased production*])

In (*in* [*increased production*]) the LR FST cannot place *(* before *in* unless it is followed by a noun phrase [ ]. However, by composing this FST with an RL FST (for simple noun phrases discussed earlier), where both FSTs are sequential, this can be easily accomplished. The total number of states in this FST is approximately 40.

In Phase 6, a LR FST computes simple verb clusters. If simple adjuncts, enclosed in ( ), are encountered during the computation, they are skipped over. Explicit attachment of adjuncts is not shown in the output. Simple verb

clusters include verbal and infinitival complements of verbs. This FST checks to see that these complement requirements are satisfied. Simple verb clusters are enclosed in { }. This LR FST has 65 states approximately. Some examples are:

{*went*}
{*has gone fishing*}
{*may have been published*}
{*may have been* (*already*) *published*}
{*have been observed and reported to be*}
{*wants to leave*}

but not

{*wants* [*the man*] (*from* [*Philadelphia*]) *to leave* }

which is bracketed as

{*wants*} [*the man*] (*from* [*Philadelphia*]) {*to leave*}

Phase 7: The computation in this phase is strictly speaking not a finite-state computation. The strings (phrases) computed in this phase are called second-order strings as they may involve nesting. In the original parser there were two implementations of this phase. The first implementation is equivalent to a pushdown automaton. In the second implementation we look for the most embedded clause, compute this with a LR FST and then repeat this process, starting from the beginning of the sentence, until the main clause has been computed. Clauses already computed are skipped while computing clauses that embed them. In either case, as far as strings enclosed in ( ) are concerned, they are either skipped as adjuncts or they are used as complements if appropriate. Attachment of clauses as well as attachment of adjuncts is not shown explicitly in the output.

While looking for verb complements during the computation of clauses the longest complement is preferred — that is, the longest path strategy is followed. The end of a complement is marked by +. Complements of a verb cluster are taken as the complements of the last verb in the cluster. Clauses are enclosed in <>. Clauses headed by *that* and sentential subjects and complements are enclosed in /\. The main clause is not enclosed in any brackets. Some examples are:

[*Those*] <*who* {*read*} [*newspapers*] + > {*waste*} [*their time*] +
<*Since* [*the conviction*] {*was based*} (*on* [*evidence*]) + > . . .

The FSTs described in Phases 1 through 6 were made 'effectively' deterministic by (1) choosing the direction of the scan (left-to-right or right-to-left) and adopting the longest-path strategy, (2) cascading right-to-left and left-to-right transducers, and (3) using the delimiting characters to allow for some minimal nondeterminism. These aspects of the program have a close relationship to some of the recent work on FSTs such as subsequential machines (Mohri (1997) and Schützenberger (1977)), decomposition of a FST into two sequential FSTs (Elgot & Mezzi (1965) and Mohri (1997)) and the work on 'directed replacement' (Karttunen (1996)). The parsing style itself has a resemblance to Abney's chunking parser (Abney (1991)). The first-order strings in Phases 4, 5, and 6 (enclosed in [ ], ( ), and { } respectively) are 'chunks' in Abney's sense.

4. A detailed example

The overall objective of the program was to prepare the text for tasks such as abstracting. However, the 1958–59 program only performed the parsing task. Besides parsing a large number of test sentences (about 200), the program processed about 25 sentences from a journal paper in biochemistry. Although the FSTs compute more structure, the final output shows relatively flat structures. Adjuncts are never explicitly attached. Here is an example from the original set of sentences:

(1) We have found that subsequent addition of the second inducer to either system after allowing single induction to proceed for 15 minutes also results in increased reproduction of both enzymes

There are no grammatical idioms in this example. In Phase 3, results (N/V) is resolved to V. After the first three phases, in Phase 4 the first (right-to-left) FST identifies the following simple NPs in this example, enclosed in [ ].

(2) [We] have found that [subsequent addition] of [the second inducer] to [either system] after allowing [single induction] to proceed for [15 minutes] also results in [increased reproduction] of [both enzymes]

The next (left-to-right) FST does not identify any new simple NPs in this example. It would have found NPs such as [the rich]. Then the next (left-to-right) FST (Phase 5) identifies the following simple adjuncts in this example, enclosed in ( ).

(3) [We] have found that [subsequent addition] (of [the second inducer]) (to [either system]) after allowing [single induction] to proceed (for [15 minutes]) (also) results (in [increased reproduction]) (of [both enzymes])

The final (left-to-right) FST (Phase 6) identifies the following simple verb clusters in this example, enclosed in { }.

(4) [We]{ have found } that [subsequent addition](of [the second inducer]) (to [either system]) after { allowing } [single induction] {to proceed} (for [15 minutes]) (also) {results} (in [increased reproduction]) (of [both enzymes])

Then the left-to-right scan (strictly, not finite state) of Phase 7 identifies the clauses, enclosed in <> and /\. The main clause is not enclosed in any brackets. + indicates the end of a complement. Thus the final output is as follows:

(5) [We] {have found} / that [subsequent addition](of [the second inducer])(to [either system]) < after {allowing} [single induction] to proceed + > (for [15 minutes]) (also) {results} (in [increased reproduction]) + > \ + (of [both enzymes])

In each one of these phases the longest-path criterion is used. This results in maximal extensions of simple NPs, simple adjuncts, and simple verb clusters and clauses. While looking for verb complements the longest complement is preferred.

5. Historical notes

The original program was implemented in the assembly language on UNIVAC 1, a single-user machine. The machine had acoustic (mercury) delay line memory of 1000 words (100 mercury channels, 10 words/channel) with input/output tape speed, 60,000 bits/sec and add/subtract speed, 1 million bits/sec. Each word was 12 characters/digits; each character/digit was 6 bits.

Lila Gleitman, Aravind Joshi, Bruria Kauffman, & Naomi Sager, and, a little later, Carol Chomsky, were involved in the development and implementation of this program (Transformations and Discourse Analysis Project (TDAP) Reports 15-19: Harris (1959), Gleitman (1959), Sager (1959), Joshi (1959) and Kauffman (1959)). A brief description of the program appears in

Joshi (1961) and a generalized description of the grammar appears in Harris (1962). This program is the precursor of the string grammar program of Naomi Sager at NYU, leading up to the current parsers of Ralph Grishman (NYU) and Lynette Hirschman (formerly at UNISYS, now at Mitre Corporation). Carol Chomsky took the program to MIT and it was used in Green's question-answering program, BASEBALL (Green 1961). At Penn, it led to a program for transformational analysis (kernels and transformations) (Joshi (1962)) and, in many ways, influenced the formal work on string adjunction (Joshi, Kosaraju, & Yamada (1972)) and later tree-adjunction (Joshi, Levy, & Takahashi (1972), Joshi (1985), Joshi & Schabes (1997), Vijay-Shanker (1987), and Weir (1988)).

6. String grammars

A string grammar is a grammatical formalization of the grammar implicit in Uniparse. A string grammar consists of a finite set of elementary strings of terminal symbols and possibly the nonterminal S. A subset of these elementary strings is called center strings. These are elementary 'sentential' strings. Each elementary string has points of adjunction, to the left and to the right of each symbol in an elementary string.[2] If an elementary string has one or more occurrences of the nonterminal S then this symbol can be replaced by one of the elementary center strings. The derivation starts with a center string and continues by adjunctions and replacements. A detailed description appears in Harris (1962) and a study of the formal properties appears in (Joshi, Kosaraju, & Yamada (1972). We will give two simple illustrative examples.

(1) The (DET) people (N) who (WH) read (V) evening (N) newspapers (N) waste (V) precious (A) time (N)

In (1) for each word we have the corresponding part of speech.
Elementary strings:

e1: N VN
e2: WH V N
e3: DET

2. Adjunctions to the whole elementary strings are also allowed. However, we will skip this detail here.

e4: A
e5: N

Center strings:

e1

Derivation: Begin with e1. e3 is adjoined to the left of the first N in e1 and e2 is adjoined to the right of the same N. e5 is adjoined to the left of N in e2 and e4 is adjoined to the left of the second N in e1. We have defined this derivation in a top-down manner. It can also be defined in a bottom-up manner.

(2) John (N) knows (V) Bill (N) left (V)

Elementary strings:

e1: N VS
e2: N V

Center strings:

e1, e2

Derivation: Begin with e1. Replace S by e2.

In a string grammar each elementary string has a flat structure. The hierarchy implicit in the derivation is decoupled from the structure of the elementary strings. Two or more elementary strings can be adjoined to a symbol on the same side. In this case the one closest to the symbol enters last in the derivation. The surface order of these adjoined strings can, of course, indicate their scope relations.

7. Tree-adjoining grammars

Tree-adjoining grammar (TAG) is a formal tree-rewriting system. TAG and Lexicalized Tree-Adjoining Grammar (LTAG) have been extensively studied with respect to both their formal properties and their linguistic relevance. TAG and LTAG are formally equivalent, but from the linguistic perspective LTAG is the system we will be concerned with in this chapter. We will often use these terms TAG and LTAG interchangeably.

The motivations for the study of LTAG are both linguistic and formal. The elementary objects manipulated by LTAG are structured objects (trees or directed acyclic graphs) and not strings. Using structured objects as the

elementary objects of the formal system, it is possible to construct formalisms whose properties relate directly to the study of strong generative capacity (i.e., structural descriptions), which is more relevant to the linguistic descriptions than the weak generative capacity (sets of strings).

Each grammar formalism specifies a domain of locality, i.e., a domain over which various dependencies (syntactic and semantic) can be specified. It turns out that the various properties of a formalism (syntactic, semantic, computational, and even psycholinguistic) follow, to a large extent, from the initial specification of the domain of locality.

7.1 Domain of locality of CFGs

In a context-free grammar (CFG) the domain of locality is the one-level tree corresponding to a rule in the CFG (Figure 3). It is easily seen that the arguments of a predicate (for example, the two arguments of *likes*) are not in the same local domain. The two arguments are distributed over the two rules (two domains of locality) — S → NP VP and VP → V NP. They can be brought together by introducing a rule S → NP V VP. However, then the structure provided by the VP node is lost. We should also note here that not every rule (domain) in the CFG in Figure 3 is lexicalized. The four rules on the right are lexicalized, i.e., they have a lexical anchor. The rules on the left are not lexicalized. The second and the third rules on the left are almost lexicalized, in the sense that they each have a preterminal category (V in the second rule and ADV in the third rule), i.e., by replacing V by *likes* and ADV by *passionately* these two rules will become lexicalized. However, the first rule on the left

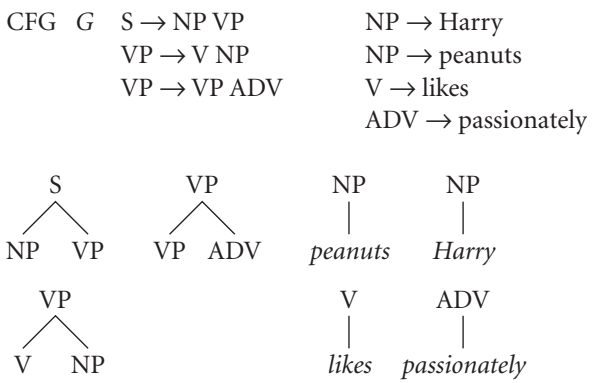


Figure 3. Domain of locality of a context-free grammar

(S → NP VP) cannot be lexicalized. Can a CFG be lexicalized? That is, given a CFG, G, can we construct another CFG, G', such that every rule in G' is lexicalized and $T(G)$, the set of (sentential) trees (i.e., the tree language of G) is the same as the tree language $T(G')$ of G'? It can be shown that this is not the case (Joshi and Schabes (1997)). Of course, if we require that only the string languages of G and G' be the same (i.e., they are weakly equivalent) then any CFG can be lexicalized. This follows from the fact that any CFG can be put in the Greibach normal form where each rule is of the form $A \rightarrow w\, B_1\, B_2\, B_n$ where w is a lexical item and the Bs are nonterminals. The lexicalization we are interested in requires the tree languages (i.e., the set of structural descriptions) to be the same, i.e., we are interested in the 'strong' lexicalization. To summarize, a CFG cannot be strongly lexicalized by a CFG. This follows from the fact that the domain of locality of CFG is a one-level tree corresponding to a rule in the grammar. Note that there are two issues we are concerned with here — lexicalization of each elementary domain and the encapsulation of the arguments of the lexical anchor in the elementary domain of locality. The second issue is independent of the first issue. From the mathematical point of view the first issue, i.e., the lexicalization of the elementary domains of locality, is the crucial one. We can obtain strong lexicalization without satisfying the requirement specified in the second issue (encapsulation of the arguments of the lexical anchor). Of course, from the linguistic point of view the second issue is very crucial. What this means is that among all possible strong lexicalizations we should choose only those that meet the requirements of the second issue. For our discussions in this chapter we will assume that we always make such a choice.

7.2 Lexicalization of CFGs

Now we can ask the following question. Can we strongly lexicalize a CFG by a grammar with a larger domain of locality? Figure 4 and Figure 5 show a tree

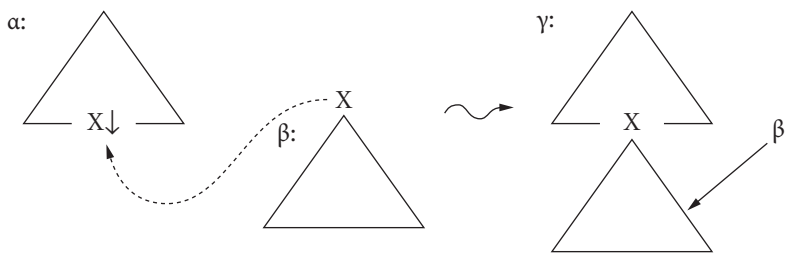


Figure 4. Substitution

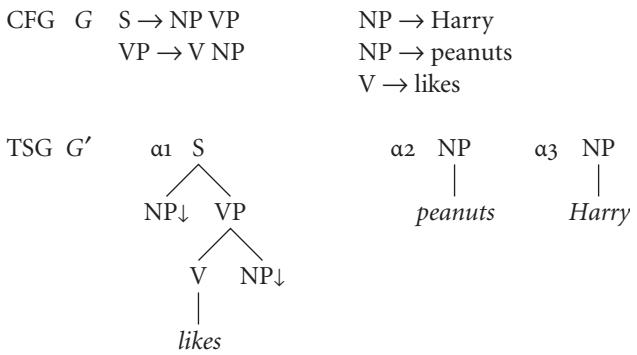


Figure 5. Tree substitution grammar

substitution grammar where the elementary objects (building blocks) are the three trees in Figure 5 and the combining operation is the tree substitution operation shown in Figure 4. Note that each tree in the tree substitution grammar (TSG), G' is lexicalized, i.e., it has a lexical anchor. It is easily seen that G' indeed strongly lexicalizes G. However, TSGs fail to strongly lexicalize CFGs in general. We show this by an example.

Consider the CFG, G, in Figure 6 and a proposed TSG, G'. It is easily seen that although G and G' are weakly equivalent they are not strongly equivalent. In G', suppose we start with the tree a_1 then by repeated substitutions of trees in G' (a node marked with a vertical arrow denotes a substitution site) we can grow the right side of a_1 as much as we want but we cannot grow the left side. Similarly for a_2 we can grow the left side as much as we want but not the right side. However, trees in G can grow on both sides. Hence, the TSG, G' cannot strongly lexicalize the CFG, G (Joshi and Schabes (1997)).

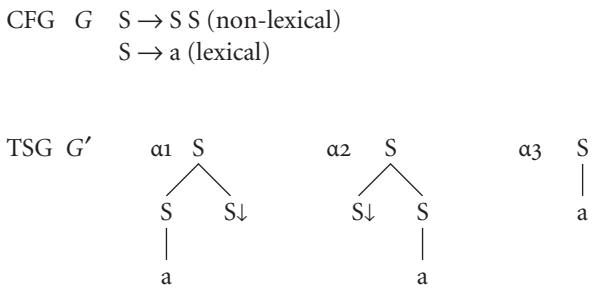


Figure 6. A tree substitution grammar

We now introduce a new operation called 'adjoining' as shown in Figure 7. Adjoining involves splicing (inserting) one tree into another. More specifically, a tree β as shown in Figure 7 is inserted (adjoined) into the tree α at the node X resulting in the tree β. The tree β, called an auxiliary tree, has a special form. The root node is labeled with a nonterminal, say X, and on the frontier there is also a node labeled X called the foot node (marked with *).There could be other (terminal or nonterminal) nodes on the frontier of β. The nonterminal nodes will be marked as substitution sites (with a vertical arrow). Thus if there is another occurrence of X (other than the foot node marked with *) on the frontier of β it will be marked with the vertical arrow and that will be a substitution site. Given this specification, adjoining β to α at the node X in β is uniquely defined. Adjoining can also be seen as a pair of substitutions as follows: The subtree at X in α is detached, β is substituted at X and the detached subtree is then substituted at the foot node of β. When a tree-substitution grammar is augmented with the adjoining operation, it is called a tree-adjoining grammar (a lexicalized tree-adjoining grammar, because each elementary tree is lexically anchored). In short, LTAG consists of a finite set of elementary trees, each lexicalized with at least one lexical anchor. The elementary trees are either initial or auxiliary trees. Auxiliary trees have been defined already. Initial trees are those for which all nonterminal nodes on the frontier are substitution nodes. It can be shown that any CFG can be strongly lexicalized by an LTAG (Joshi and Schabes (1997)).

In Figure 8 we show a TSG, G', augmented by the operation of adjoining, which strongly lexicalizes the CFG, G. Note that the LTAG looks the same as the TSG considered in Figure 6. However, now trees α_1 and α_2 are auxiliary trees (marked with *) that can participate in adjoining. Since adjoining can insert a tree in the interior of another tree, it is possible to grow both sides of the tree α_1 and tree α_2, which was not possible earlier with substitution alone. In summary, we have shown that by increasing the domain of locality we have

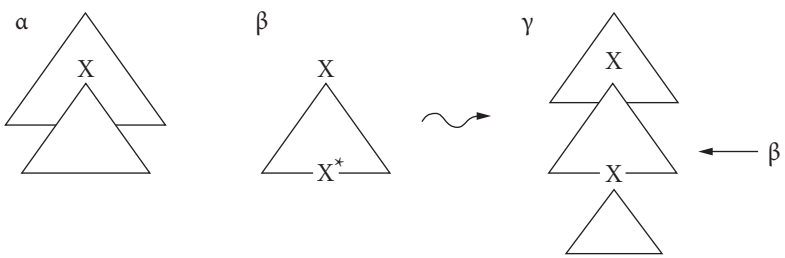


Figure 7. Adjoining

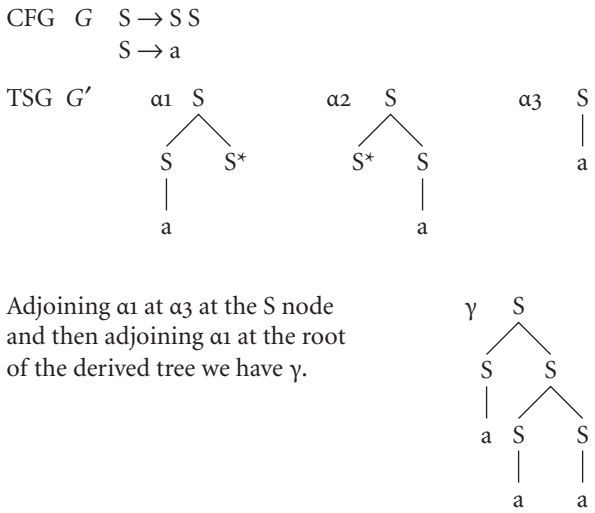


Figure 8. Adjoining arises out of lexicalization

achieved the following: (1) lexicalized each elementary domain, (2) introduced an operation of adjoining, which would not be possible without the increased domain of locality (note that with one-level trees as elementary domains adjoining becomes the same as substitution since there are no interior nodes to be operated upon), and (3) achieved strong lexicalization of CFGs.

7.3 Lexicalized tree-adjoining grammars

Rather than giving formal definitions for LTAG and derivations in LTAG, we will give a simple example to illustrate some key aspects of LTAG. We show

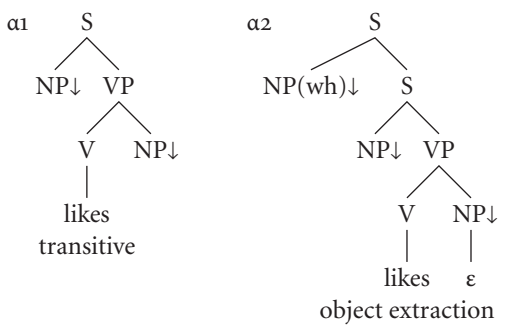


Figure 9. LTAG: elementary trees for *likes*

some elementary trees of a toy LTAG grammar of English. Figure 9 shows two elementary trees for a verb such as *likes*. The tree α_1 is anchored on *likes* and encapsulates the two arguments of the verb. The tree α_2 corresponds to the object-extraction construction. Since we need to encapsulate all the arguments of the verb in each elementary tree for *likes*, for the object-extraction construction, for example, we need to make the elementary tree associated with *likes* large enough so that the extracted argument is in the same elementary do main. Thus, in principle, for each 'minimal' construction in which *likes* can appear (for example, subject extraction, topicalization, subject relative, object relative, passive, etc.) there will be an elementary tree associated with that construction. By 'minimal' we mean when all recursion has been factored away. This factoring of recursion away from the domain over which the dependencies have to be specified is a crucial aspect of LTAGs as they are used in linguistic descriptions. This factoring allows all dependencies to be localized in the elementary domains. In this sense, there will, therefore, be no long-distance dependencies as such. They will all be local and will become long-distance dependencies on account of the composition operations, especially adjoining.

Figure 10 shows some additional trees. Trees α_3, α_4, and α_5 are initial trees and trees β_1 and β_2 are auxiliary trees with foot nodes marked with *. A derivation using the trees in Figure 9 and Figure 10 is shown in Figure 11. The trees for *who* and *Harry* are substituted in the tree for *likes* at the respective NP nodes, the tree for *Bill* is substituted in the tree for *think* at the NP node, the tree for *does* is adjoined to the root node of the tree for *think* (adjoining at the root node is a special case of adjoining), and finally the derived auxiliary tree (after adjoining β_2 to β_1) is adjoined to the indicated interior *S* node of the

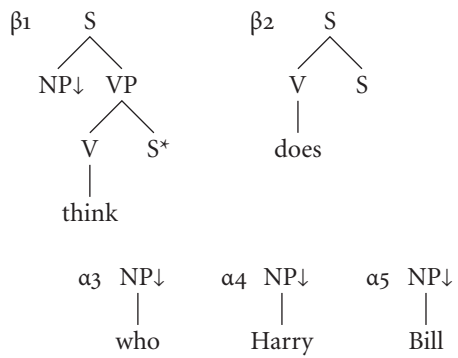


Figure 10. LTAG: sample elementary trees

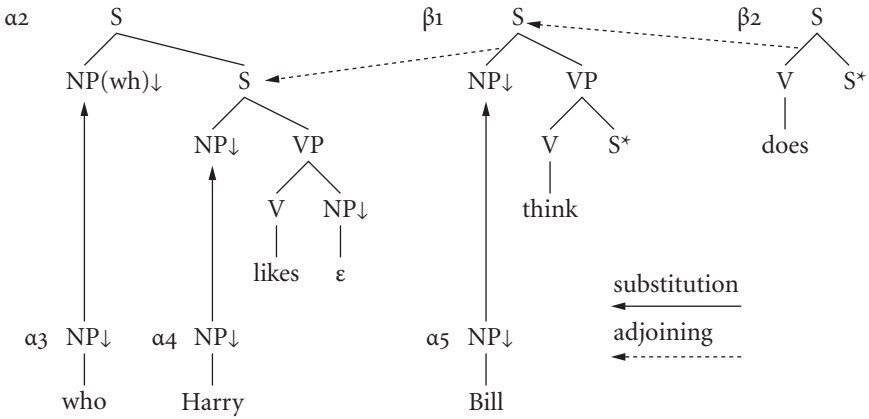


Figure 11. LTAG derivation for *who does Bill think Harry likes*

tree α_2. This derivation results in the derived tree for *who does Bill think Harry likes*, as shown in Figure 12. Note that the dependency between *who* and the complement NP in α_2 (local to that tree) has been stretched in the derived tree in Figure 12. This tree is the conventional tree associated with the sentence.

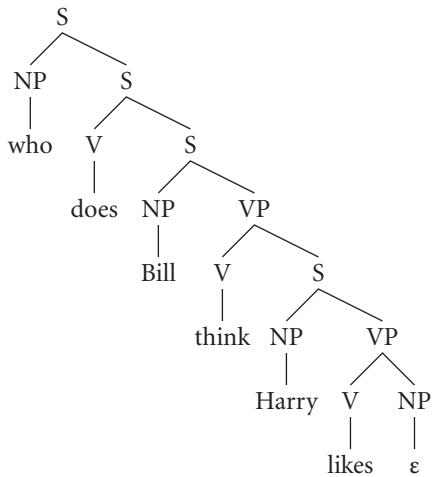


Figure 12. LTAG derived tree for *who does Bill think Harry likes*

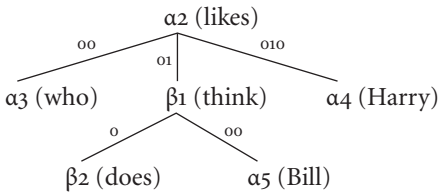


Figure 13. LTAG derivation tree

However, there is also a derivation tree in LTAG, the tree that records the history of composition of the elementary trees associated with the lexical items in the sentence. This derivation tree is shown in Figure 13. The nodes of the tree are labeled by the tree labels such as α_2 together with the lexical anchor.[3]

The derivation tree is the crucial derivation structure for LTAG. We can obviously build the derived tree from the derivation tree. For semantic computation the derivation tree (and not the derived tree) is the crucial object. Compositional semantics is defined on the derivation tree. The idea is that for each elementary tree there is a semantic representation associated with it and these representations are composed using the derivation tree. Since the semantic representation for each elementary tree is directly associated with the tree there is no need to reproduce the internal hierarchy in the elementary tree in the semantic representation (Joshi & Vijay-Shanker (1999)). This allows the so-called 'flat' semantic representation and also helps in dealing with some non-compositional aspects, as in the case of rigid and flexible idioms.

8. Some important properties of LTAG

The two key properties of LTAG are (1) extended domain of locality (EDL) (for example, as compared to CFG), which allows (2) factoring recursion from the domain of dependencies (FRD), thus making all dependencies local. All other properties of LTAG (mathematical, linguistic, and even psycholin-

3. The derivation trees of LTAG have a close relationship to dependency trees. There are some crucial differences, but the semantic dependencies are the same.

guistic) follow from EDL and FRD. TAGs (LTAGs) belong to the class of so-called mildly context-sensitive grammars (Joshi (1985)). Context-free languages (CFL) are properly contained in the class of languages of LTAG, which in turn are properly contained in the class of context-sensitive languages. There is a machine characterization of TAG (LTAG), called embedded push-down automaton (EPDA) (Vijay-Shanker (1987)), i.e., for every TAG language there is an EPDA which corresponds to this (and only this) language and the language accepted by any EPDA is a TAG language. EPDAs have been used to model some psycholinguistic phenomena, for example, the processing of crossed dependencies and nested dependencies has been discussed in (Joshi (1990)). With respect to formal properties, the class of TAG languages enjoys all the important properties of CFLs, including polynomial parsing (with complexity $O(n^6)$).

Large-scale wide-coverage grammars have been built using LTAG, the XTAG system (LTAG grammar and lexicon for English and a parser) being the largest so far (for further details see (The XTAG Research Group (2002)). In the XTAG system, each node in each LTAG tree is decorated with two feature structures (top and bottom feature structures), in contrast to the CFG-based feature-structure grammars. This is necessary because adjoining can augment a tree internally, while in a CFG-based grammar a tree can be augmented only at the frontier. It is possible to define adjoining and substitution (as it is done in the XTAG system) in terms of appropriate unifications of the top and bottom feature structures. Because of FRD (factoring recursion from the domain of dependencies), there is no recursion in the feature structures. Therefore, in principle, feature structures can be eliminated. However, they are crucial for linguistic descriptions. Constraints on substitution and adjoining are modeled using these feature structures (Vijay-Shanker (1987)). This method of manipulating feature structures is a direct consequence of the extended domain of locality of LTAG.

The relevance of LTAG to the topic of this chapter is that LTAG provides a framework that allows us to view the issue of the minimal hierarchical structure necessary for sentence description in terms of the minimal structure necessary for the elementary trees of LTAG. Viewing the problem in this way leads to factoring the contribution to the hierarchical structure for a sentence due to recursion from the hierarchical structure of the elementary trees of LTAG. The hierarchical structure of the elementary trees is the appropriate structure to consider when investigating the issue of the minimal hierarchical structure necessary for describing sentence structure.

References

Abney, S. 1991. "Parsing by chunks". In Berwick, R. et al. (eds.), *Principle-based Parsing*, 257–278. Dordrecht, Holland & Boston: Kluwer Academic Publishers. [Originally appeared in: Carol Tenny, editor, The MIT Parsing Volume, 1988–89. Center for Cognitive Science, MIT.]

Elgot, C.C. and J.E. Mezzi. 1965. "On relations defined by generalized finite automata". *IBM Journal of Research and Development* 9.

Gleitman, Lila R. 1959 "Word and Word-Complex Dictionaries". *Transformations and Discourse Analysis Project* (TDAP) 16. The University of Pennsylvania.

Green, B.F. et al. 1961. "Baseball: An automatic question-answerer." Proceedings of the Western Joint Computer Conference (WJCC) 219.

Green, B., A. Wolf, C. Chomsky, & K. Laughery, 1961. "BASEBALL: An automatic question answerer", In: Proceedings of the Western Joint Computer Conference 19, 219–224. American Federation of Information Processing Societies. [Reprinted in B.J. Grosz, K. Spärck-Jones, & B.L. Webber (editors), *Readings in Natural Language Processing*, 545–549. California: Morgan Kaufmann (1986).]

Harris, Zellig S. 1959. "Computable syntactic analysis". Transformations and Discourse Analysis Project (TDAP) 15. The University of Pennsylvania.

Harris, Zellig S. 1962. *String Analysis of Sentence Structure*. The Hague: Mouton

Joshi, A.K. 1961. "Computation of syntactic structure". *Advances in Documentation and Library Science*, vol. II, part 2. New York: John Wiley/Interscience Publishers.

Joshi, A.K. 1962. "A procedure for transformational decomposition". Transformations and Discourse Analysis Project (TDAP) 42, The University of Pennsylvania.

Joshi, A.K., S.R. Kosaraju, & H.M. Yamada. 1972a. "String adjunct grammars: I. Local and distributed adjunction". *Information and Control*, 21.2:93–116.

Joshi, A.K., S.R. Kosaraju, & H.M. Yamada. 1972b. "String adjunct grammars: II. Equational representation, null symbols, and linguistic relevance". *Information and Control*, 21.3:235–260.

Joshi, A.K., Levy, L.S. and Takahashi, M. 1975. "Tree adjunct grammars". *Journal of Computer and System Sciences*, 10:1.

Joshi A.K. 1985. "Tree-adjoining grammars: How much context sensitivity is required to provide reasonable structural descriptions?" In D. Dowty, L. Karttunen, and A. Zwicky, editors, *Natural Language Parsing*, 206–250, Cambridge University Press.

Joshi, A.K. 1999. "A parser from antiquity: an early application of finite state transducers to natural language parsing". In A. Kornai, editor, *Extended Finite State Models of Language*, 6–15, Cambridge University Press.

Joshi, Aravind K. 1959. "Recognition of local substrings". Transformations and Discourse Analysis Project (TDAP) 18. The University of Pennsylvania.

Joshi, A.K. and Schabes, Y. 1999. "Tree-adjoining grammars". In G. Rosenberg and A. Salomaa, editors, *Handbook of Formal Languages*, 69–123. Berlin: Springer Verlag.

Karttunen, L. 1996. "Directed replacement". *Proceedings of the 34th Annual Meeting of the ACL*, Santa Cruz, CA.

Kauffman, Bruria. 1959. "Second-order substrings and wellformedness". Transformations and Discourse Analysis Project (TDAP) 19. The University of Pennsylvania.
Miller, P. H. 1999. *Strong Generative Capacity*. Stanford: CSLI Publications.
Mohri, M. 1997. "Finite state transducers in language and speech processing". *Computational Linguistics* 23.2: 269–311.
Sager, Naomi. 1959 "Elimination of alternative classification". Transformations and Discourse Analysis Project (TDAP) 17. The University of Pennsylvania.
Schützenberger, M. P. 1977. "Sur une variante des fonctions séquentielles". *Theoretical Computer Science* 4:47–57
The XTAG Research Group 2002. "A lexicalized tree-adjoining grammar for English". Technical Report 02–18, Institute for Research in Cognitive Science, Philadelphia.
Vijay-Shanker, K. 1987. *A study of Tree-Adjoining Grammars*. Ph.D. Dissertation, University of Pennsylvania.
Weir, D. 1988. *Characterizing Mildly Context-Sensitive Grammar Formalisms*. Ph.D. Dissertation, University of Pennsylvania.

Chapter 6

The computability of operator grammar

Stephen B. Johnson
Columbia University

1. Introduction

When encountering Operator Grammar for the first time, one is first struck by its incredible elegance: a universal theory of language based on simple predication. The theory states that each sentence of a language has an underlying normalized structure in which operator words predicate on argument words, and that this structure carries the information of the sentence. This assertion has profound consequences for information science and computational linguistics. However, deeper exploration of the theory reveals an overwhelming volume of linguistic data that must be managed: the tantalizing hope of a pure normal form is obscured by a vast amount of structural variation.

The representation of Operator Grammar in a form amenable to computation is an elusive task for a number of reasons. First, Harris provides no formalization of his descriptions of Operator Grammar (Harris 1982, 1991). Although the grammatical coverage of this work is staggering, linguistic structures and processes are described entirely in prose. Harris used mathematical descriptions in his early work and was certainly aware of existing developments in linguistics, so the complete avoidance of formalism is significant. His restraint may be partly due to a philosophy that seems to permeate his writings, that no formal system can provide the best characterization of language, and that each system is a tool that is useful for some purposes, and less useful for others. Another important methodological consideration for Harris was the search for a "least grammar"; any notation or formalism requires additional definition and explanation (Harris 1991:39).

The most radical departure of Operator Grammar from other linguistic theories is the representation of meaning in a statistical model rather than through interpretation in a logical model: the information carried by a

sentence is given by the probability distributions of arguments under their operators. Integration of this perspective with conventional symbolic representations of grammar is a daunting task. In particular, it suggests that any serious investigation into the computational aspects of the theory would require access to a large body of linguistic data in order to understand the actual distributions of operators and their arguments. Corpora with formats sufficient for this kind of analysis have only become available very recently, and appropriate tools are still lacking.

Operator Grammar describes language in terms of three constraints: Dependency, Likelihood, and Reduction. Dependency creates the structure of operators predicating on arguments. The relative likelihood of operators with respect to their arguments gives rise to the information carried by the structure. Reductions simplify the structure by shortening or removing low-informational material. Each of these constraints has an affinity to particular linguistic formalisms. The most obvious of these related approaches is Categorial Grammar, which is the only formalism mentioned by Harris in his later work. Categorial Grammar is well suited to representing simple strings of operators and their arguments and their formation into predication structures using functional application.

It is intriguing that when Harris comes to describe the Reduction constraint he switches to a process model in which words progressively 'enter' into the structure. This perspective differs enormously from the functional world of Categorial Grammar. In particular, reductions are described as ordered rules that apply as words enter. This component of the theory is most similar to early transformational systems, or 'rewrite' grammars.

Individual operators and arguments provide the informational conditions under which reduction can take place. This implies that the lexicon plays a central role in the theory. An immediate parallel can be found in Lexicon Grammar (Gross 1984), which is based on the later transformational theory of Harris (1968), just before operator grammar was formulated. Transformations can be expressed as rewrite rules that establish mappings in the set of known category sequences of a language (strings). A Lexicon Grammar of a language specifies, for every lexeme, the strings in which the lexeme can occur. The resulting system thereby indicates which lexemes participate in a given transformation. Operator Grammar requires a similar mechanism to specify which reductions affect which lexical items. However, because reductions usually involve two or more lexemes, a different representation is required.

These are symbolic approaches. The Likelihood constraint has an entirely different character with much more in common with work in language engineering and statistical natural language processing. Probabilistic models have been integrated with symbolic grammars, but only as practical methods for addressing syntactic ambiguity in parsing. Relevant work can be found in corpus-based linguistics in which semantic models are developed using statistical methods such as clustering. Some methods investigate the relative informational contribution of operators and arguments, which can provide the formal conditions for Reduction (Resnik 1993).

Individually, each of the formalisms discussed above characterizes aspects of Operator Grammar with various degrees of success. This chapter presents a method that combines features of the different approaches into a single formalism. Section 2 reviews the main categories of Operator Grammar and formalizes the Dependency constraint using Categorial Grammar. Section 3 expresses this grammar using a rewrite system, which is then used to represent the Reduction constraint. Section 4 captures the relationship between reductions and individual lexemes using a tabular approach similar to Lexicon Grammar. Section 5 attempts to extend this approach to model the Likelihood constraint in conjunction with corpus-based methods. Section 6 discusses different techniques for implementing the tabular grammar as algorithms that recognize and generate sentences.

2. Categories and base sentences

The categories of Operator Grammar can be formalized as *O*, *N*, *I* and *C*. Operators (*O*) include primitive verbs, adjectives, prepositions and relational nouns. Noun arguments (*N*) consist of primitive nouns not derivable from other words. Indicators (*I*) mark words as arguments, marking nouns usually with prepositions, and operators with the words *that*, *whether*, *to* and the suffixes *-s* and *-ing*. Carriers of tense morphology (C) include *be* for all non-verb operators (*is big*, *is on the table*, *is a father*) and *do* for verb operators (*does go*).

The base sentences of Operator Grammar consist of sequences of the above categories. In this Section all morphemes are treated as independent words, even affixes like *-s* and *-ing*. Morphology is covered by the reductions described in Section 3. Operators have one, two or three arguments, which may be of category *N* or *O*. In English, the normal linear sequence of base

sentences is first argument, operator, second argument, and then third argument. The indicators *that*, *whether*, and the prepositions precede their arguments. The indicators *-s*, *-ing*, and *to* immediately precede the operator. Carriers (*be* and *do*) occur between the operator and its indicator.

The base sentences can be easily described using Categorial Grammar (Oehrle, Bach & Wheeler 1988, Carpenter 1998). The categories of Operator Grammar are shown in Table 1, with their equivalent representations in Categorial Grammar. The first column presents a notation based on that used by Harris. The third column lists examples of some lexemes for each category. The fourth column supplies the semantics for the first word in each list using the lambda calculus.

Table 1. Categories in operator grammar and categorial grammar

Operator Grammar Category	Categorial Grammar Category	Lexeme Examples	Semantic Example
N	N	John, Mary, Fido	*John*
O	O	*John gives Fido to Mary*	$give^{s}(John,Fido,Mary^{to})$
C	$(O\backslash N)/((O\backslash N)/(O\backslash N))/(O\backslash N)$	*be, do*	$\lambda x\lambda y.y^{x}$
I	N/N	*to, with, on*	$\lambda x.x^{to}$
I	$(O\backslash N)/(O\backslash N)$	*-s, -ing, to*	$\lambda x.x^{s}$
I	O/O	*that, whether*	$\lambda x.x^{that}$
O_N	$O\backslash N$	*big, bark, fall*	$\lambda x.big(x)$
O_{NN}	$O\backslash N/N$	*eat, pet, feed*	$\lambda y\lambda x.eat(x,y)$
O_{NN}	$O\backslash N/N$	*depend*	$\lambda y\lambda x.depend(x,y^{on})$
O_{NNN}	$O\backslash N/N/N$	*give*	$\lambda y\lambda z\lambda x.give(x,y,z^{to})$
O_O	$O\backslash O$	*true, fact*	$\lambda x.true(x)$
O_{ON}	$O\backslash O/N$	*please, with*	$\lambda y\lambda x.please(x^{ing},y)$
O_{NO}	$O\backslash N/O$	*want, know*	$\lambda y\lambda x.want(x,y^{to})$
O_{OO}	$O\backslash O/O$	*cause*	$\lambda y\lambda x.cause(x^{ing},y^{to})$
O_{NNO}	$O\backslash N/O/N$	*ask, tell*	$\lambda y\lambda z\lambda x.ask(x,y,z^{to})$
O_{NOO}	$O\backslash N/O/O$	*wonder*	$\lambda y\lambda z\lambda x.wonder(x,y^{whether},z^{or})$
O_{NNN}	$O\backslash N/N$	*Give Fido*	$\lambda z\lambda x.give(x,Fido,z)$
O_{NNN}	$O\backslash N$	*Give Fido to Mary*	$\lambda x.give(x,Fido,Mary^{to})$
O_{NO}	$O\backslash N$	*want Fido to eat*	$\lambda x.want(x,eat^{to})$

Categorial Grammar uses slashes to indicate the categories of required arguments, while Operator Grammar uses subscripts. In Categorial Grammar, the

slant of the slash specifies the side of the operator on which the argument must appear. For example, the category *O\N/O* requires a noun argument on its left and an operator (sentence) on its right. The category *O* represents a complete sentence of a language, i.e., a morpheme sequence in which all requirements are satisfied. The semantic form of each operator consists of a predicate word, one or more arguments, and indicators of operators and arguments represented as features (superscripts in this notation). Features are commonly used in Categorial Grammar to handle phenomena such as agreement (Steedman 1996).

The analysis of a sentence in Categorial Grammar is accomplished by assigning the appropriate categories to each lexeme, then combining adjacent categories until only the category *O* remains. As categories combine, the lambda expressions are used to determine the resulting semantic form. For example, the sentence *John gives Fido to Mary* is analyzed as follows:

N	$(O\backslash N)/(O\backslash N)$	$O\backslash N/N/N$	N	N/N	N
John	$\lambda x.x^{s}$	$\lambda y\lambda z\lambda x.give(x,y,z)$	*Fido*	$\lambda x.x^{to}$	*Mary*
					N
					$Mary^{to}$
		$O\backslash N/N$			
		$\lambda z\lambda x.give(x,Fido,z)$			
		$O\backslash N$			
		$\lambda x.give(x,Fido, Mary^{to})$			
	$O\backslash N$				
		$\lambda x.give^{s}(x,Fido, Mary^{to})$			
		O			
		$give^{s}(John,Fido, Mary^{to})$			

Additional morpheme sequences recognized by this grammar include the following:

Fido -s bark.
Fido -s be big.
John -s pet Fido.
Fido -s depend on John.
Mary -s ask Sam for John to feed Fido.
John -s wonder whether Fido -s bark or Fido -s bite.
John -ing feed Fido -s please Mary.

The compact notation of Operator Grammar is very convenient, and will be used in the remainder of the chapter. Some additional notation is required such as the intermediate categories formed during the derivation. For example, *give Fido* has category $O\backslash N/N$ and *give Fido to Mary* has category $O\backslash N$. Harris did not provide a notation for operators with satisfied requirements, and so these will be indicated by underlining the specified arguments as shown in Table 2: $O_{N\underline{N}N}$ indicates that the second argument is satisfied, and $O_{N\underline{NN}}$ indicates that the second and third arguments are satisfied.

It is also convenient to use categories with variables to refer to first, second and third arguments. This notation makes it possible to describe operators more generally, without having to specify the exact number or type of arguments. The variables X, Y, and Z indicate unsatisfied requirements, while underlined variables $\underline{X}$, $\underline{Y}$, and $\underline{Z}$ indicate a requirement that has been satisfied. The following categories can be used to match operators in various states of satisfaction:

$O_{X\ldots}$	Only first argument is unsatisfied
$O_{\ldots Y\ldots}$	All arguments are unsatisfied
$O_{\ldots Z}$	First and third arguments are unsatisfied
$O_{\underline{X}\ldots}$	All arguments are satisfied (same as category O by itself)
$O_{\ldots \underline{Y}\ldots}$	Second argument is satisfied
$O_{\ldots \underline{Z}}$	Second and third arguments are satisfied

3. Processes and reduced sentences

Although the categories of Operator Grammar appear very similar to those of Categorial Grammar, Harris does not make use of functional application or the lambda calculus in analyzing sentence structure. Instead, he describes sentences as resulting from a generative process in which operators and arguments progressively enter to form a dependency structure. It is during this process that reductions apply, creating more compact forms.

The generative process of sentence construction is most naturally captured in rewrite rules — one of the earliest formalisms to be used in mathematical linguistics. Early transformational approaches employed simple word classes such as noun, verb, and adjective. Although Operator Grammar does not have classes like these, the categories developed in Section 2 prove to be extremely helpful in describing both the base sentences and the reduced sentences

derived from them. In fact, the process of generating base sentences can be described in three rules:

$$I\ O_{\underline{X}\ldots} \rightarrow I\ X\ I\ C\ O_{X\ldots} \quad \text{(ARG1)}$$
$$O_{\ldots\underline{Y}\ldots} \rightarrow O_{\ldots Y\ldots}\ I\ Y \quad \text{(ARG2)}$$
$$O_{\ldots\underline{Z}} \rightarrow O_{\ldots Z}\ I\ Z \quad \text{(ARG3)}$$

Rule ARG1 states that an operator predicating on its first argument ($O_{\underline{X}\ldots}$) and marked by an indicator (I) can be linearized into a sequence of morphemes in which a carrier word (C) immediately precedes the operator lacking its first argument ($O_{X\ldots}$), and the argument (X) precedes the carrier. The first occurrence of category (I) on the right side of the rule is filled by indicators *that*, *whether*, and *for*, while the second occurrence is filled by indicators *–s*, *–ing*, and *to*.

The rule ARG2 states that an operator predicating on its second argument ($O_{\ldots\underline{Y}\ldots}$) can be linearized into a sequence of morphemes in which the operator lacking its second argument ($O_{\ldots Y\ldots}$) precedes the indicator (I) marking the second argument, which in turn precedes the argument (Y). Rule ARG3 has a similar interpretation for the linearization of the third argument.

These linearization rules form a grammar that can either generate the base sentences of a language, or recognize (parse) such sequences. (The grammar is a context-sensitive grammar because the ARG1 rule has two categories on the left side of the rule.) For example, the sentence *John gives Fido to Mary* can be generated as follows:

	I	$O_{\underline{X}\ldots}$							(satisfied operator, with argument *-s*)
	-s								
→	I	O_{NNN}							(choose operator *give*)
	-s	*give*							
→ I	N	I	C	$O_{N\underline{NN}}$					(apply rule ARG1)
	John	*-s*		*give*					
→ I	N	I	C	$O_{N\underline{NN}}$		I	N		(apply rule ARG3)
	John	*-s*		*give*		*to*	*Mary*		
→ I	N	I	C	O_{NNN}	$I\ N$	I	N		(apply rule ARG2)
	John	*-s*		*give*	*Fido*	*to*	*Mary*		

This example illustrates a number of important points. First, morphemes such as *-s* occur as independent words (morphology is handled below). Each category in the derivation becomes paired with a lexeme. Because of the variables *X*, *Y* and *Z* in the rules, the choice of one lexeme affects others. For example, the choice of the word *give* determines that there will be three arguments, and that these arguments belong to category *N*. Finally, carriers and indicators are not always present. The verb *give* does not require a carrier, and there are no indicators on first argument *John* or second argument *Fido*. The third argument *Mary* is marked by the indicator *to*.

Most sentences of a language are generated by reductions. These rules fall into three groups: reduction of free morphemes to affixes (morphological rules); reduction of free morphemes to null (zeroing); and alternative linearizations of base categories (transpositions). The following reductions provide a sense of the variety of reductions in Operator Grammar:

N	$\rightarrow a\ N$	(ART)
$I\ C\ O_{X\ldots}$	$\rightarrow O_{X\ldots}{+}I$	(OPIND)
X	$\rightarrow X\ [\ I\ O\ ]$	(INT)
$X\ [\ X\ I\ C\ O_{X\ldots}]$	$\rightarrow X\ which\ I\ C\ O_{X\ldots}$	(WH)
$X\ which\ I\ C\ O_{X\ldots}$	$\rightarrow X\ O_{X\ldots}$	(MOD)
$X\ O_X$	$\rightarrow O_X\ X$	(ADJ)
$X\ I\ C\ O_{XO}\ X\ I\ C\ O_{X\ldots}$	$\rightarrow X\ I\ C\ O_{XO}\ I\ C\ O_{X\ldots}$	(REP1)
$O_{\ldots Y\ldots}\ I\ Y$	$\rightarrow O_{\ldots Y\ldots}$	(INDEF2)

The ART rule introduces the indefinite article on nouns: *a dog*. This is a morphological rule, because Harris treats the indefinite article as a discontinuous part of the noun morpheme, and not an independent word. The OPIND rule attaches the operator indicator as a suffix on the verb when the carrier is null: *barks*. When the carrier is not null, additional rules create morphological forms such as *is big*, and *does bark*. Additional reductions create plural and tensed forms (*barked, will bark, was big, were big*, etc.). In this grammar notation, spaces convey the separation between free morphemes, while the plus sign indicates attachment of affixes.

The INT rule allows a sentence to interrupt another sentence at any point: *A dog — that dog was really big — barked*. Orthographically, the interrupting sentence is typically set off by dashes, but in the notation square brackets are used for clarity. Because this construction is so rare, it may seem peculiar to include this rule in the grammar (it is likely that no computerized grammar represents this structure). Interruption is vital as the source of all modifiers in

language. This becomes clearer in the WH rule, which changes the interruption into a relative clause by reducing an argument to a relative pronoun (*which*): *A dog which was really big barked.*

The variable *X* in the WH rule serves two purposes. First, it enforces that the category of the argument meets the requirement of its operator ($O_{X\ldots}$). Second, the variable forces the argument in the embedded sentence to be identical to the argument in the sentence being interrupted. When used in this way, the variable *X* represents the category (e.g., *N*) paired with the lexeme (e.g., *dog*). This notation provides a formal means of representing the metalinguistic 'sameness' operator used by Harris (1982: 87–97).

The MOD rule shows the deletion of material in the embedded sentence (*which*, indicator *-s*, and carrier *be*), creating a right modifier of the noun (e.g., prepositional phrase). This rule is the motivation for the existence of the carrier category (*C*) in the grammar: complete sentences require the presence of a carrier, while modifiers lack a carrier. The ADJ rule moves the modifier O_X to the left of the host *X*: *A really big dog barked.*

The REP1 rule provides an example of argument zeroing. In this case, an argument is a repetition of another argument of a higher operator. For example, *John wanted to go* is derived from *John wanted John to go*, in which the subject of *go* repeats the subject of *want*. The lower argument can be zeroed because it conveys no additional information. Here again, the variable *X* is used to represent the repetition of the subjects for both operators, and to ensure that the arguments belong to the category required by those operators. The INDEF2 rule shows a different type of zeroing, in which verbs zero their direct objects when these are indefinite nouns. For example, *John eats things* reduces to *John eats*.

When the above rules are combined with the argument rules of Section 2, the grammar can generate a much greater variety of sentences. For example, the following is a derivation of *A big dog wants to eat*. For brevity, the words paired with each category are omitted:

	$I\ O_{\underline{NO}}$	
→	$N\ I\ \underline{C}\ O_{\underline{NO}}$	(ARG1)
→	$N\ I\ C\ O_{\underline{NO}}\ I\ O$	(ARG2)
→	$N\ I\ C\ O_{\underline{NO}}\ N\ I\ C\ O_{\underline{NN}}$	(ARG1)
→	$N\ I\ C\ O_{\underline{NO}}\ N\ I\ C\ O_{\underline{NN}}\ I\ N$	(ARG2)
→	$N\ I\ C\ O_{\underline{NO}}\ N\ I\ C\ O_{\underline{NN}}$	(INDEF2)
→	$N\ I\ C\ O_{\underline{NO}}\ I\ C\ O_{\underline{NN}}$	(REP1)
→	$a\ N\ I\ C\ O_{\underline{NO}}\ I\ C\ O_{\underline{NN}}$	(ART)

$\rightarrow$ *a* N [I O] I C O_{NO} I C O_{NN} (INT)
$\rightarrow$ *a* N [N I C O_N] I C O_{NO} I C O_{NN} (ARG1)
$\rightarrow$ *a* N *which* I C O_N I C O_{NO} I C O_{NN} (WH)
$\rightarrow$ *a* N O_N I C O_{NO} I C O_{NN} (MOD)
$\rightarrow$ *a* O_N N I C O_{NO} I C O_{NN} (ADJ)
$\rightarrow$ *a* O_N N O_{NO}+I I O_{NN} (OPIND)
a big dog wants to eat

Additional sentences generated by this grammar include:

Fido — Fido is a dog — barks.
A man pets a large white dog.
A dog which has a white spot runs.
John feeds Fido which pleases Mary.
Fido barks which causes John to feed Fido.
A dog which wants to eat barks.

While this example is presented in terms of generation, the rules can be applied in reverse order as a recognition procedure: Morphological rules map complex lexemes into free morphemes; modifiers are expanded into independent interrupting sentences. Zeroed arguments are reconstructed; and arguments are combined with their operators. The process terminates when only a single, satisfied operator remains, marked by the present singular indicator (*-s*).

4. Lexical preferences and reduction

The production rules of Section 3 characterize a general relationship between the base sentences and the reduced sentences. However, individual word choices are not taken into account. For example, the ARG1 rule does not specify which operators require a carrier and which do not (*to be big* vs. *to bark*) while the ARG2 rule does not specify which indicators are imposed on the argument (*depend on John* vs. *near to John*). A greater concern is that the reductions apply equally to any words having the appropriate categories. The ART rule must be restricted to count nouns, while the ADJ rule should not apply to O_O operators such as *fact*.

The rewrite rule formalism does not lend itself to representing such restrictions. In contrast, Categorial Grammar enables each lexeme to specify only the appropriate forms in which it may occur. Table 1 gives examples of restrictions on indicators. For example, the operator *depend* imposes the

indicator *on*, while the operator *eat* does not. The rule formalism must be extended to achieve a similar linking of syntactic forms and lexemes.

One method would be to list applicable rules for each lexeme. For example, the lexeme *dog* (a count noun) would list the indefinite article rule ART, while *sand* (a mass noun) would not. This representation is rather clumsy, and does not succeed in making the link between syntax and individual lexemes explicit. An alternative approach is to use a tabular representation (Nevin 1985, pc 1987). This direction is suggested by Lexicon Grammar (Gross 1984), which organizes the lexemes of a language into tables in which lexical items are rows and category sequences (strings) are columns. The cells of each table contain Boolean values indicating whether a lexeme occurs in a given string or not. For example, *eat* occurs in the string *N V* and in *N V N*, while *wear* occurs only in *N V N*. The clever insight of this approach is to represent transformations as relations between pairs of columns in a table. For example, the above pair of columns is a transformation that allows omission of direct objects. Words like *eat* that occur in both columns participate in the transformation, while verbs like *wear* do not.

This technique is not immediately applicable to Operator Grammar for several reasons. First, Operator Grammar does not yield a small, regular set of category sequences like the strings of transformational grammar. Strings are the products of multiple reductions and so cannot be used without factoring. A deeper difficulty is that most reductions involve at least two words, not one — the reduction takes place when both are present. This prohibits a representation in which there is only one lexeme per row.

The tabular approach can be adapted to Operator Grammar by constructing a table for each rewrite rule. For example, Table 2 demonstrates how different lexemes instantiate the ARG1 rule from Section 3. The first column of the table shows various fragments of sentences for an O_N verb *fall*, an O_N adjective *big*, an O_{ON} verb *surprise* and an O_O adjective *possible*. The second two columns show the left side of the rule: the indicator (I), and the operator when it is combined with its first argument $(O_{\underline{X}\ldots})$. The remaining columns show the right side of the rule: indicator word on the operator (I), indicator on the first argument (I), first argument (X), indicator suffixes on the operator (I), carrier (C), and the operator without its first argument $(O_{X\ldots})$.

In a complete grammar, the rows in Table 2 would have to be expanded to cover all the operators of English. In this sense, the table is equivalent to a lexicon that specifies the characteristics of each operator, such as the types of the arguments, the indicators imposed by it, and the need for a carrier word

Table 2. Tabular representation of ARG1 rule

Example	I	$O_{\underline{X}\ldots}$	I	X	I	C	$O_{X\ldots}$
It falls	*-s*	*fall*	–		*-s*	–	*fall*
It does fall	*-s*	*fall*	–		*-s*	*do*	*fall*
It falling	*-ing*	*fall*	–		*-ing*	–	*fall*
Its falling	*-ing*	*fall*	*-'s*		*-ing*	–	*fall*
That it falls	*that*	*fall*	*that*		*-s*	–	*fall*
Whether it falls	*whether*	*fall*	*whether*		*-s*	–	*fall*
For it to fall	*to*	*fall*	*for*		*to*	–	*fall*
It is big	*-s*	*big*	–		*-s*	*be*	*big*
It being big	*-ing*	*big*	–		*-ing*	*be*	*big*
Its being big	*-ing*	*big*	*-'s*		*-ing*	*be*	*big*
That it is big	*that*	*big*	*that*		*-s*	*be*	*big*
Whether it is big	*whether*	*big*	*whether*		*-s*	*be*	*big*
For it to be big	*to*	*big*	*for*		*to*	*be*	*big*
That it falls surprises us	*-s*	*surprise*	*that*		*-s*	–	*surprise*
Its falling surprises us	*-s*	*surprise*	*-ing*		*-s*	–	*surprise*
For it to fall surprises us	*-s*	*surprise*	*to*		*-s*	–	*surprise*
That its falling surprises us	*that*	*surprise*	*that*		*-s*	–	*surprise*
Whether its falling surprises us	*whether*	*surprise*	*whether*		*-s*	–	*surprise*
For its falling to surprise us	*to*	*surprise*	*for*		*to*	–	*surprise*
That it falls is possible	*-s*	*possible*	*that*		*-s*	*be*	*possible*
Its falling is possible	*-s*	*possible*	*-ing*		*-s*	*be*	*possible*
For it to fall is possible	*-s*	*possible*	*to*		*-s*	*be*	*possible*
That its falling is possible	*that*	*possible*	*that*		*-s*	*be*	*possible*
Whether its falling is possible	*whether*	*possible*	*whether*		*-s*	*be*	*possible*
For its falling to be possible	*to*	*possible*	*for*		*to*	*be*	*possible*

(*be* or *do*). Because the column labeled X in Table 2 is blank, any words of the appropriate category are allowed as first argument, e.g. any word of category N can be the first argument of the O_N verb *fall*. However, if an operator has a very restricted set of arguments, each could be listed explicitly. This technique is similar to the handling of idiomatic expressions in Lexicon Grammar (Gross 1984).

The tabular representation is particularly useful for restricting a reduction to apply to a particular set of lexemes. For example, verbs like *want* allow the subject of the complement sentence to be null (*John wanted to go*), while verbs such as *observe* do not (**John observed to go*). This reduction is formalized by the rule REP1 in Section 3. The rule must be restricted to apply only to operators like *want* and *eager*, but not to operators like *observe*. In addition,

the reduction only takes place when the indicator *to* is imposed on the complement sentence (**John wanted going*, **John wanted that goes*). These details can be captured in a tabular form, as shown in Table 3.

Table 3. Tabular representation of rule for subject zeroing

Example	X	I	C	O_{XO}	X	I	C	$O_{X\ldots}$	X	I	C	O_{XO}	I	C	$O_{X\ldots}$
John wants to sleep		*-s*	–	*want*		*to*	–			*-s*	–	*want*	*to*	–	
John wants to be happy		*-s*	–	*want*		*to*	*be*			*-s*	–	*want*	*to*	*be*	
John is eager to sleep		*-s*	*be*	*eager*		*to*	–			*-s*	*be*	*eager*	*to*	–	
John begins to sleep		*-s*	–	*begin*		*to*	–			*-s*	–	*begin*	*to*	–	
John begins to be happy		*-s*	–	*begin*		*to*	*be*			*-s*	–	*begin*	*to*	*be*	

Using this representation, the grammar of a language consists of three tables for the argument linearization rules (ARG1, ARG2, and ARG3), and a table for each reduction of the language. Thus, the grammar would have tables for morphological rules (e.g., ART, WH, OPIND), zeroing rules (e.g., MOD, REP1, INDEF2), and transposition rules (e.g., ADJ).

5. Likelihood and interpretation

The tables of Section 4 model the relationship between lexical items (morphemes) and grammatical forms in a crisp manner: the lexeme either participates in the rule or it does not. This is the correct model of the Dependency constraint: each morpheme either belongs to one of the categories or it does not. The vast majority of morphemes belong to only one category (Harris 1991:34–35). The tables are also a good model for capturing many reductions — the conventions of each language establish many absolute rules about the forms in which specific morphemes can participate. These restrictions in the tables are immediately applicable to idiomatic expressions by algorithms that parse or generate sentences.

The arguments of a given operator do not occur with equal probability, and conversely the operators over a given argument are not equiprobable. This Likelihood constraint has practical consequences for the computability of Operator Grammar. Without appropriate constraints on word choices, parsing algorithms will consider a vast number of implausible structures and generation algorithms will produce extremely unnatural constructions. If data are

available about the relative likelihood of operator words co-occurring with particular argument words, probabilities can be assigned to dependency structures. The set of possible choices can be pruned by dropping very low-probability word combinations. This technique can reduce ambiguity when parsing or improve appropriateness when generating.

The likelihood of a given pair of operator and argument words co-occurring can be estimated from a large corpus, e.g. by using measures such as mutual information (Hindle 1990). Because the number of possible pairs is enormous, even a very large corpus will assign zero probability to many pairs that are in fact acceptable. Various techniques have been developed to address the problem of sparse data: smoothing techniques (Church 1991), taxonomies (Resnik 1993), clustering (Lee 1993, 1997), and similarity metrics (Dagan 1993).

The three tables for base sentences described in Section 4 provide a natural place to store data about relative likelihood. A column could be added to each of the three tables to specify the estimate of likelihood for particular word combinations in the operator and argument positions. An extreme approach would be to maintain tables of several billion rows that enumerate each possible pair, with scores estimated by smoothing techniques. Another method would be to store only rows for observed pairs, and estimate scores for unobserved pairs based on a measure of similarity to other words. If clusters or taxonomic classes are used, scores can be assigned not to pairs of particular words, but to operator words and some representation of a cluster. For example, the operator *eat* might have a row in the ARG2 table representing a high score for words belonging to the 'food' cluster.

Likelihood data are also necessary to provide a more accurate model of reductions. Using the tabular approach of Section 4, it is necessary to list all verbs that can zero their direct objects (the INDEF2 rule of Section 3). This set includes *eats* and *reads*, but not *needs* and *wears*. The set of objects that can be zeroed differs for each verb, e.g. *eats* can zero words belonging to the class 'food'. The strength with which an operator selects its arguments can be measured using relative entropy, by comparing the distributions of the arguments with and without the operator (Resnik 1993). Operators that select strongly for their arguments tend to be those that can zero their arguments, and the zeroed arguments are those that contribute most to the overall measure.

Measures of an operator's selection strength can be calculated from the likelihood scores stored in the base tables. Selection strength can then be used to suggest or block entries in the table that represents the INDEF2 rule for

object omission. Verbs whose selection strength exceeds a given threshold would be considered by the linguist for inclusion, and those falling below would be flagged as potential errors. Similar techniques are applicable to many other reductions. For example, the verb *expect* zeros the operator *come* in the complement sentence: *I expect John* is derived from *I expect John to come*. By examining the distribution of *expect* it is possible to find other operators with similar behavior and establish a reduction with broader applicability.

In this manner, the reduction tables can be populated in a principled manner, based on the informational properties of operators. This approach provides linguists with a new set of mathematical tools to explore the phenomena of reductions. The mathematical criteria for each reduction can be stated in a precise manner using distributional measures. For generation algorithms this approach supports formal techniques for making sentences compact, while for recognition algorithms, it enables guessing the identity of zeroed material based on probability distributions.

The Likelihood constraint of Operator Grammar has implications far beyond the construction of more efficient language-processing algorithms. The availability of detailed data about argument likelihood opens a new paradigm for semantics. Predication can be modeled using distributions rather than the discrete truth model of conventional logic. This can be interpreted as a predicate being 'more sayable' of some arguments than of others. Higher-order operators (*believe, deny, if, could*, etc.) can be modeled as functions that alter the distributions of their arguments in distinct and important ways. This approach has a strong foundation in Fuzzy Logic, which generalizes the discrete values of logic to all values in the interval between 0 and 1. Various mathematical functions have been investigated for modifying and combining fuzzy predicates, e.g., logical conjunction and disjunction are generalized using minimum and maximum functions (Klir 1988).

In most work on Fuzzy Logic, distributions of predicates like *tall* and *young* are typically constructed by intuition. Operator Grammar offers a method whereby the distributions of predicates can be estimated empirically, using a sufficiently large corpus pertaining to the domain of discourse of interest. Logical systems based on this approach would be computable to the extent that data are available for estimating a model. For example, fuzzy rules for diagnosing diseases from symptoms could in principle be constructed from an enormous body of medical articles and textbooks. Development of such rules would, of course, be an enormous undertaking — parsing the texts, determining argument frequencies, and smoothing the data. Above all, this

approach requires a much deeper understanding of the different types of distributions operators can have, and in particular how higher-order operators effect these distributions, e.g. in establishing a model of implication in the given domain. Practical systems of deduction will be feasible only in narrow domains such as science and technology.

6. Implementation

The tables of Operator Grammar described in Section 4 can be used to implement generation and recognition algorithms in several ways. The most direct method is to employ a traditional transformational approach, in which the tables are used as rewrite rules. However, unlike the rules presented in Section 3, the entries in the tables constrain the applicability of rules to particular lexemes. Although these lexical constraints prevent many absurd derivations, the combinatorial problems of phenomena such as conjunction and modifier attachment remain as intractable as in other computational grammars. The likelihood data described in Section 5 can be used to improve efficiency of rule-based approaches by pruning derivations that have low probability.

An interesting option is to translate the tables into a Categorial Grammar. The basic logical form of each lexeme could be determined from the tables for the base sentences (ARG1, ARG2 and ARG3). These representations would look much like those in Table 1. Many versions of Categorial Grammar include a set of 'lexical rules' that derive additional forms of a lexeme from the basic forms (Carpenter 1998). For example, a lexical rule is required to establish a relationship between an adjective that occurs before the noun (*a big dog*) to an adjective that occurs after the noun (*a dog is big*). This is accomplished by deriving the category and lambda expression of one form from the other. Similarly, a lexical rule could generate the lambda expression for verbs such as *want* that allow subject zeroing. Lexical rules could be constructed from the reduction tables presented in Section 4. Lambda expressions for words on the left side of the rule must be mapped to expressions that account for the reduced elements on the right side.

The tabular approach could be quite effective in constructing large, detailed Categorial Grammars by organizing relationships in a more inspectable format and managing the idiosyncrasies of individual lexemes. The enormous breadth of reduction phenomena require the most powerful

varieties of Categorial Grammars, which have developed many extensions over the years. Unfortunately, parsers for these grammars (e.g., Lambek grammar) currently exhibit exponential behavior (Hepple 1999).

An extremely appealing approach has emerged with recent interest in finite-state formalisms. Simple finite-state transducers can be combined to create very complex grammars that handle phenomena usually addressed by context-free grammars (Gross 1989, Gross 1997, Roche 1997). Transducers are developed to transform 'local' syntactic variations into more regular forms. Separate transducers are created to represent the unique properties of each lexeme. This method is quite compatible with the tabular representation described in Section 4. Essentially, each row of each table would be translated into a finite state transducer. Although the resulting finite state machine for a given language is enormous, parsing and recognition procedures have extremely efficient implementations.

7. Conclusion

The tabular approach to Operator Grammar provides an organizing framework in which several very different grammatical approaches can be combined. The columns of the tables are similar to the type systems employed in Categorial Grammar. The grouping of the columns corresponds to the left and right sides of classic rewrite rules, which can be used for either generation or recognition of sentences. The rows capture lexical idiosyncrasies as in Lexicon Grammar, and constrain the application of reductions to particular words.

Tables representing the relationship between operators and their arguments provide a natural place to store measurements of relative likelihood. Likelihood scores can be used in several ways: to improve the efficiency of parsing and generation algorithms; to formalize the conditions under which reductions apply; and to support a model of semantic interpretation based on Fuzzy Logic.

The resulting grammar system makes relationships between lexical items and syntactic forms explicit. This approach integrates naturally with corpus-based techniques and therefore can support research on very large grammars with complex behavior. The formalism fits well with current work on finite state transducers, facilitating the implementation of extremely efficient language-processing algorithms.

References

Carpenter, Bob. 1998. *Type-Logical Semantics (Language, Speech, and Communication)*. Cambridge, Massachusetts: MIT Press.

Church, Kenneth William Gale. 1991. "A comparison of the enhanced Good-Turing and deleted estimation methods for estimating probabilities of English bigrams". *Computer Speech and Language*, 5: 19–54.

Dagan, Ido, Shaul Markus, & Shaul Markovitch. 1993. "Contextual word similarity and estimation from sparse data". In *Proceedings of the 30th Annual Meeting of the ACL*, 164–171.

Gross, Maurice. 1984. "Lexicon-Grammar and the Syntactic Analysis of French". In *Proceedings of the 10th International Conference on Computational Linguistics (COLING'84)*, Stanford, California.

Gross, Maurice. 1989. *The Use of Finite Automata in the Lexical Representation of Natural Language*. Lecture Notes in Computer Science, 377. Heidelberg: Springer-Verlag.

Gross, Maurice. 1997. "Local grammars". In Emmanuel Roche & Yves Schabes (editors), *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts. pp. 330–354.

Harris, Zellig S. 1968. *Mathematical Structures of Language*. New York: John Wiley & Sons.

Harris, Zellig S. 1982. *A Grammar of English on Mathematical Principles*. New York: John Wiley & Sons.

Harris, Zellig. 1991. *A Theory of Language and Information: A mathematical approach*. Oxford: Clarendon Press.

Hepple, Mark. 1999. "An Earley-style predicative chart parsing method for Lambek grammars". In *Proceedings of the 37th Annual Meeting of the ACL*, 465–475.

Hindle, Donald. 1990. "Noun classification from predicate-argument structures". In *Proceedings of the 28th Annual Meeting of the ACL*, 268–275.

Klir, George, Tina Folger. 1988. *Fuzzy Sets, Uncertainty, and Information*. Englewood Cliffs, New Jersey: Prentice Hall.

Lee, Lillian, Fernando Pereira, & Naftali Tishby. 1993. "Distributional clustering of English words". In *Proceedings of the 31st Annual Meeting of the ACL*, pp. 183–190.

Lee, Lillian. 1997. *Similarity-Based Approaches to Natural Language Processing*. Doctoral Dissertation, Harvard University,

Nevin, Bruce. 1985. "Constructive Lexicon-Grammar". Bolt Beranek & Newman Artificial Intelligence Seminar, Cambridge, Massachusetts, September 13, 1985.

Oehrle, Richard, Emmon Bach, & Deirdre Wheeler (Eds.). 1988. *Categorical Grammars and Natural Language Structures*. Studies in Linguistics and Philosophy, Vol 32. Dordrecht/ Holland: D. Reidel.

Resnik, Phillip. 1993. *Selection and Information: A class-based approach to lexical relationships*. Doctoral Dissertation, University of Pennsylvania. (Institute for Research in Cognitive Science report IRCS-93-42).

Roche, Emmanuel. 1997. "Parsing with finite state transducers". In Emmanuel Roche & Yves Schabes (editors). *Finite-State Language Processing*. Cambridge, Massachusetts: MIT Press.

Steedman, Mark. 1996. *Surface Structure and Interpretation*. Linguistic Inquiry Monograph 30. Cambridge, Massachusetts: MIT Press.

Part 3

Computer applications

Chapter 7

Distributional syntactic analysis and valency

Basic notions, procedures, and applications of the Pronominal Approach

Karel van den Eynde
Katholieke Universiteit Leuven

Sabine Kirchmeier-Andersen
Copenhagen Business School

Piet Mertens
Katholieke Universiteit Leuven

Lene Schøsler
University of Copenhagen

1. Introduction

This chapter introduces an approach to valency analysis extending Harris's distributional procedures, and presents its application to the French and the Danish verbal valency dictionaries, PROTON and OVD. This method is called the Pronominal Approach (PA)[1] because of the role it assigns to paradigms of pronouns that are proportional to constituents containing lexical (i.e. non-pronominal) elements. (The relation termed *proportionality* is defined below.). The PA first examines the set of 'pronominal sentences' and their internal relations. It thus is based on language-immanent observations such as those to which Harris's procedures are applied. We will also show how it furnishes a

1. The PA was first presented in Eynde & Blanche-Benveniste (1978). For a detailed account see Blanche-Benveniste et al. (1984) or the special issue of *International Journal of Lexicography* (1994, vol 7 no. 2).

constructive proof[2] of basic syntactic concepts, and in particular reinforces Harris's string analysis (Harris 1965).

Whereas traditionally syntax and lexicon were viewed as two independent domains of grammar, in modern syntactic theories syntactic information is almost completely integrated in the lexicon. Consequently, the investigation of the relations between syntactic constructions has gained interest during the last decade. This led to many interesting insights into the syntax-semantics interface (Levin 1993, Levin & Rappaport 1996) and to proposals for a highly structured and hierarchical design of the lexicon (Pustejovsky 1995).

Harris exploited a pair-wise equivalence relation between constructions (e.g. Harris 1969a [1970:615]). Whiteley (1960) used *entailment* between constructions as a criterion for establishing construction groups or networks leading to a syntactico-semantic typology.

Recent work in the PA combines both aspects: it establishes relations between two or more constructions of a verbal predicator, based on the observation that terms are shared in those constructions; moreover relationships are classified according to criteria such as the following: is syntactic function of terms preserved or not, is the relationship predicator-specific or not, is one construction implied by the other. This results in a network of relations between constructions, some of which may be combined. For an application to French, see Eynde, Mertens, & Swiggers (1998).

In Harris's and Whiteley's approaches, construction slots are usually filled by lexical phrases. In Whiteley's approach this contains a risk: the relations of implication or entailment might be established only for particular lexical items on a more or less ad hoc basis, and therefore would not prove the existence of systematic relationships (of logical implication) between the constructions as such, independently of the lexical items filling the slots. Whiteley reduces the risk of establishing unjustified implications between constructions by using variables representing referents of the constructions (Whiteley 1960:150). The

---

2. For a definition of constructivism in mathematics, see Goodman (1951), and the introductory definition in Bishop (1967, 1985). For its application outside mathematics see Goodman (1951), Bronowski (1978), Hofstadter (1980), Quine (1989), Harris (1990). The fundamental idea is that all concepts should be defined as the result of the application of a specific set of procedures (whatever the length of the proof: it is important that the concept be *procedurally provable*). Langacker rightly claimed as a basic need the conceptual clarification of fundamental issues: "The vital problems of current linguistic theory are not of a formal nature, but lie instead at the level of conceptual foundations" (1987:1).

PA takes this approach one step further by using pronominal paradigms instead of variables. In this way it applies the principles underlying the following quotation from Quine (1964:13):

> [T]o be is to be in the range of reference of a pronoun. Pronouns are the basic media of reference; nouns might better have been named pro-pronouns.[3]

In order to obtain a workable definition of the pronominal paradigms for syntactic analysis, the notion of *proportionality*[4] is essential for the establishment of construction groups. This concept is procedurally defined by establishing the ratios between the lexical (sub)syntagms of a sentence and their pronominal counterparts (cf. Section 2.2). Harris's String Analysis provides a series of systematic tests for the identification of syntactic elements, among which the excision or *omissibility* test, but as shown in Eynde (1998:147-150) this test in fact presupposes the foreknowledge of the correct structure, based upon the proportionality with minimal referentials.

Contrary to such concepts as NP, VP . . ., to syntactic functions (subject, object, . . .), or to roles (agent, patient, . . .), pronouns are elements that belong to language itself and pronominal sentences are directly observable: they are open to judgment of grammaticality or acceptability.[5] Moreover, the inventory of pronouns is finite and constitutes a closed set.

These observations constitute the basis of the PA: when establishing the valency of a predicator, we exploit this proportionality in order to reduce the huge number of combinations between lexical elements to a much smaller

---

3. See also Quine's 'dictionary' (1987) under "Variables", where he attributes the last sentence to C.S. Peirce.

4. Proportionality is a relation of equivalence between **ratios**. The proportionality relation

boat		plane		train			(1,a)		(2,a)		(3,a)
------	÷	------	÷	-------	results in the feature definition		-----	÷	-----	÷	-----
water		air		earth			(1,b)		(2,b)		(3,b)

one feature common to the (three) numerators, another common to the (three) denominators, and another common to the elements of a given ratio. This provides five syntactico-semantic features: *a* means of transport, *b* environment, *1* aquatic, *2* aerial, *3* terrestrial. We put more specified terms as numerator, less specified ones as denominator. We have a simplified use of the term "proportional(ity)" when applying it to a single ratio, leaving it up to the reader to supplement missing ratios.

5. The terms "subject, object, indirect object, . . ." will be used for practical reasons; a distributional definition of the syntactic functions will be given later.

number of combinations with pronouns. In other words, the finite nature of the list(s) of pronouns makes it possible to examine their combinations with predicators (in our dictionaries the *full verbs*, i.e. lexical as opposed to function verbs such as auxiliaries or modals) in a systematic and exhaustive way, without having to appeal to particular semantico-interpretative features. The possibility or impossibility of a particular pronoun appearing is indeed meaningful: the pronouns reveal the *primary* (i.e. basic) characteristics a predicator imposes on its dependents.

Sections 2 to 4 present the basic notions of the PA progressing from the basic elements, the pronouns, to paradigms of pronouns and relations between paradigms and predicators forming constructions. Section 5 considers relations between constructions with a view to verb classification. Examples from French will illustrate the basic notions, whereas examples from Danish will be used in Section 4.

It will be clear that pronouns and pronominal paradigms can be used in many different contexts such as discourse analysis, prosodic analysis, and text cohesion. However, we will concentrate exclusively on aspects related to valency and verb classification.

2. Procedures for defining construction frames

In the first part of this chapter, we describe how the basic elements, the pronouns, are identified, delimited from lexical elements, and further classified (cf. Section 2.1). Next we present the organization of pronouns in paradigms (cf. Section 2.3). Furthermore, we describe how paradigms are related to predicators, thus forming constructions (cf. Section 2.4), and finally how relations between different constructions of the same predicator can be established (cf. Section 2.5).

2.1 Referents and types of referents

2.1.1 *Degree of referential specification*

The degree of specification distinguishes pronouns from lexicalized constituents, i.e. constituents containing lexical elements. The degree of lexicalization coincides with the degree of specification: the pronouns are generic elements with minimal reference; the lexicalized forms are necessarily more specific. Thus, (1) below is more specific than (2) which is more specific than (3):

(1) Our French readers buy Harris's books
(2) Our readers buy his books
(3) They buy them

Put differently, the denotation of a pronoun (the set of objects it may refer to) is larger than the denotation of a lexical item. As opposed to nouns which may function as hyperonyms or generic concepts such as *man* or *human*, pronouns can be delimited as those elements which have the largest set of lexicalization possibilities.

The group of pronouns thus defined can be further subdivided according to their degree of referential specification. If the referential specification is suspended we use the term suspensive pronouns, which correspond roughly to the traditional category *interrogative pronoun* and *adverb*. These pronouns are the ones that have the smallest or at least a smaller set of syntactico-semantic features. Furthermore, there exists a category of paranouns (e.g. *someone*, *something*, *everywhere*, *never*, . . .), involving extra features of quantification. Finally, even more specific are assertive pronouns such as *he*, *she* or *these* characterized by the presence of a number of syntactico-semantic features labelled as *number*, *gender*, *sex*, etc. (cf. Section 5.1). For a more detailed account of feature analysis of French pronouns, see Eynde et al. (1988). Since proportionality is most appropriately observed between constructions, we will focus on constructions instead of isolated forms.

2.1.2 *Clitic — non-clitic*

The Pronominal Approach presupposes a complete inventory of the pronominal system which reveals different properties for each language because the specific properties of pronouns are language specific. In the analysis of French, we have to make a prosodic distinction between clitic pronouns, which are never stressed, and non-clitic pronouns, which may be stressed.[6] This distinction is important for the establishment of more general (reduced) paradigms

6. Clitic pronouns do not receive the final accent, except when, in case of inversion, they appear in final position in the intonation group (in that case the lexical accent of the verb shifts to the last full syllable of the group). A clitic which is not in final position of the intonation group may only bear the initial accent, which is different from the final accent, which is preserved, cf. Mertens (1993, 1997). Some non-clitic pronouns are univocal (14a), others like *elle*, *elles*, *nous*, *vous*, *lui*, . . . have homophonous clitics (14b). Suspensive pronouns, except *que*, and paranouns all are non-clitic.

(cf. Section 2.3) and for the analysis of constructions with double marking in French (cf. 2.6). Appendix 1 shows the pronoun inventory of French.

2.2 Proportionality between constructions

The term *construction* is used in its broadest sense referring to a verbal, adjectival or nominal *predicator* (cf. section 2.4) and its dependents, whether in pronominal or lexical form. In the following, however, we will focus on constructions with verbal predicators.

The relations between constructions with lexical elements are prefigured by the relations between the corresponding proportional sentences containing only the predicator and pronominal referents. Sentence (1) is proportional to (3) since *our French readers* is proportional to *they*, and *Harris's books* is proportional to *them*, and so forth.

The relation of proportionality differs from the concept of substitution, commonly used in linguistics, cf. Eynde et al. (1988: 178). First, it disregards the linear order of the elements in the construction, as seen in the French translations of (1) and (3) given in (4) and (5) below:

(4) Nos lecteurs français achètent les livres de Harris.

(5) Ils les achètent.

Next, proportionality is more specific than substitution: in (6) the constituent *the baker* may be replaced by the string *he* but there is no proportionality between *he* and *the baker's wife* cf. (7). Proportionality indeed implies the *unification* with a subset of the (morpho-syntactic) features of the related strings, which is not the case with substitution. The concept of proportionality is well compatible with the concept of unification as it is used in current unification grammars (cf. Shieber (1986), Abeillé (1993)).

(6) the baker offers the boy a sweet ÷ he offers him a sweet

(7) the baker's wife offers the boy a sweet ¬÷ he offers him a sweet

2.3 Paradigms — reduction of paradigms

Starting from the list of pronominal constructions of a verbal predicator, we set up the paradigm of pronouns that appear in particular positions relative to the predicator, and this for all attested positions. For instance, from (8) we

obtain (9), and consequently, disregarding morpheme inflection, we obtain (10) where the positions are labeled P0, P1, P2, . . . (zero elements, noted ø, indicate optionality of the dependent).

(8) il le leur donne (he+it(non-fem)+them+gives = he gives it to them)
il le lui donne (he+it(non-fem)+him/her+gives = he gives it to him/her)
elle la lui donne (she+it(fem)+him/her+gives = she gives it to him/her)
on le lui donne (one(indef)+it(non-fem)+him/her+gives = I/we/you/someone gives it to him/her)
je les donne (I+them+give = I give them)
je t'en donne (I+you+thereof+give = I give you some)
il me les donne (he+me+them+gives = he gives them to me)

(9) {il, elle, je, on, ..} {le, la, les} {lui, leur} donner

(10)

P0	P2	P1	P2′	P1′	V
je	me	le	ø	ø	donner
tu	te	la			
il	se	les			
elle	nous	ø		en	
on	vous				
nous	ø		lui		
. . .			leur		

Pronominal homonyms, such as *nous* in paradigms P0 and P2, are distinguished by their membership in distinct paradigms: the *paradigmatic solidarity* of *nous* in P0 is distinct from that of *nous* in P2.

On the basis of the observations in (11) and (12) we define the complementary distribution between classes P2 and P2′, and between P1 and P1′. The complete paradigm for each position containing all pronouns which may possibly appear with some predicator is called a *maximal paradigm*. The paradigm of pronouns encountered for a particular predicator is always a subset of the maximal paradigm. Reduction of the complementary classes in (11b) and (11c) and (12b) produces the table in (13):

(11) a. il {me, te, se, nous,vous} {l'/le, l'/la, les} {donner, propose}
b. *il vous les lui donne
c. *elle vous leur donne

(12) a. il en donne
b. *il nous la en donne

(13)

P0	P1	(P2)	V
je	le	ø	donner
tu	la	me	
il	les	te	
elle	en	se	
on		nous	
nous		vous	
. . .		lui	
		leur	

The French paradigms so far have been established with clitics. The examples in (14) show that clitic and non-clitic paradigms are complementary. Sentence (15) is grammatical only if the constituent *à celle-là* "to that one" (fem) doubles the pronoun *lui*. Evidence for this *double marking* (cf. Section 2.6), is the fact that both are responses to a single suspensive pronoun *à qui* "to whom", and the fact that in (15) *à celle-là* "to that one" (fem) necessarily has the appendix intonation (a very low level tone, cf. Eynde et al. (1998)).

(14) a. je donne {ceci, cela, ça, celle-ci} {à celle-là, à celui-là,. . .}
b. je lui donne {ceci, cela, ça, celle-ci}
c. je le donne {à celle-là, à celui-là,. . .}

(15) je lui donne ceci à celle-là

A sentence like *il la leur donne* resembles an algebraic formula (or an expression in a formal language) where the pronouns correspond to variables and the lexical elements to constants. More precisely, the lexical element (here the constructing verb) corresponds to the functor of a structure, and the pronouns to arguments of that structure: DONNER(*il, la, leur*). The formula, while mentioning the set of possible pronouns instead of a single pronoun, provides a specification of the construction characteristics of the predicator. This specification is termed *formulation*.

The way it is used in the PA, the term *pronoun* covers many more elements than is usual in the grammatical tradition (cf. *infra*, § 4.5).

As defined for the PA a paradigm contains at least two pronouns and is proportional to a suspensive pronoun such as *à qui* (2.1.1).

2.4 Predicators

The procedure for identifying the predicator of a sentence *S* is as follows:

1. Identify constituents that are elements of pronominal paradigms, or lexical counterparts which are proportional to pronominal paradigms. Set them aside with the status of *referents*.

S =	(John)	surely did not hand	(the book)	over	(to his friend)
	who/he		what/it		to whom/him
	ref		*ref*		*ref*

2. Construct the minimal question *Q* to which the sentence *S* is an answer. This minimal question is:

Q=	Did	(he)	hand	(it)	over	(to him)
		ref		*ref*		*ref*

3. Identify constituents in *S* that may be omitted in the minimal interrogative sentence. These *adjuncts* are non-referential. To distinguish them from optional referents, the non-referential adjuncts will be called *appendages*.

(John)	(surely)	did (not)	hand	(the book)	over	(to his friend)
who/he				what/it		to whom/him
ref	*adjunct*	*adjunct*		*ref*		*ref*
	appendage	*appendage*				

4. Identify modal and auxiliary verbs (temporal, interrogative, epistemic, etc.). The details of this step are necessarily language-specific.

(Did)	(he)	hand	(it)	over	(to him)
aux	*ref*		*ref*		*ref*

5. The remainder is the predicator: *hand over*. This predicator may be simple (*give*), discontinuous (*hand over*), or complex (*kick the bucket*, *take the French leave*). Moreover, it may be a proverb (*la nuit, tous les chats sont gris*; *a rolling stone gathers no moss*). No referents nor appendages can be identified in such sayings.

The verbal predicator is the most common construction kernel, but predicators can be adjectival too. The adjectival predicator consists of a predicative adjective together with its copula, e.g. *to be eager for*. The valency of the adjectival predicator may differ from the valency of the homophonous attributive adjective.

2.5 Formulations

A formulation is the distributional table of the pronouns appearing with a particular predicator. More precisely, it is the set of paradigms of predicator-specific elements, each paradigm containing the total set of pronouns (clitic or non-clitic) withheld after elimination of the restrictions concerning linear order and after reduction of the complementary paradigms (cf. 2.3 and 2.6). This set provides part of the definition of the valency of the predicator: it makes explicit restrictions on the nature of the paradigms (16) (e.g. pseudo-paradigms), on the number and syntactic function of the dependents (17), and states whether they are compulsory or optional (18).

(16) a. *il pleut*
b. *{*elle,on*} *pleut*

(17) a. *je lui offre celui-là*
b. **il lui pleut* (**it+him+rains*)

(18) a. *je les lui offre, je les offre*
b. **j'offre*

Example (19) shows the compulsory presence of an element of paradigm P1 for the predicator *comporter*.

(19) a. *ça en comporte* (*autant*)
it+thereof+contains+(that many)
b. *ça les comporte*
it+them+contains
c. **ça comporte*
*it+contains

Note that the selectional restrictions do not stem from individual pronouns, but from the combination of proportional pronouns (the *pronominal cluster*), and, consequently, from its complement of non-proportional pronouns (i.e. in comparison with the elements of the *maximal* paradigm as observed for other

predicators). If the paradigm P1 conforms with the expression *÷le, ÷me, ÷te, ÷celui-ci ¬÷ceci* (i.e. accepts the pronouns *le, me, te, celui-ci*, but refuses *ceci*), the referent unifies with the feature [+anim] (20). If it unifies with *÷le, ¬÷me, ¬÷te, ÷ceci, ÷celui-ci*, the referent unifies with the bundle [-anim, +concr] (21). Finally, the configuration *÷le, ¬÷me, ¬÷te, ¬÷ceci, ¬÷celui-ci, ÷ça* unifies with [–anim, –concr] (22).

(20) a. *il* {*le,me,te*} *engendre* (*engender*)
b. **il engendre ceci*

(21) a. *je l'embranche* (*join up*)
b. **je t'embranche*

(22) a. *je le pense, je pense ça* (think)
b. **je pense celui-ci*

Some predicators accept a union of feature values for a particular slot, as in P1 in (23).

(23) a. *je le vois*, {*lui, ça, celui-là*}
I+it/him+see,+(him, that, that one)
b. *je le vois, l'homme*
I+him+see+,+the man
c. *je le vois, ce miroir*
I+it+see+,+that mirror

Formulations should not be confused with constructions, as there may be several constructions corresponding to a single formulation. The following section describes the general syntactic relations between unmarked and marked constructions.

2.6 Dispositives and double marking

Constructions in (24), (25), and (26) are proportional to one another. The function of the referents is constant despite the variation of the surface syntax. We distinguish three types of constructions: (Blanche-Benveniste *et al.* 1984: 37, 66–67, 152–155): the direct dispositive (24), the binarization or *extraction* dispositive (25), and the double marking (26).

(24) *ils chargent le camion*
"they are loading the truck"

(25) a. *c'est le camion qu'ils chargent*
"it is the truck they are loading"
b. *ce sont eux qui le chargent*
"it is they who are loading it"

(26) a. *moi, je charge le camion*
"me, I'm loading the truck"
b. *je charge le camion, moi*
"I'm loading the truck + me"
c. *le camion, ils le chargent*
"the truck, they are loading it"
d. *ils le chargent, le camion*
"they are loading it + the truck"

The binarization dispositive and the double marking are general *mechanisms*, systematic and thus predictable: they apply to dependents of all predicators, whether predicator-specific or not. Moreover, they may be combined (27).

(27) *Jacques, c'est à midi qu'il l'a retrouvé, son livre.*
"Jacques, it is at noon that he recovered it, his book"

The verbal form *est* in the extraction dispositive *c'est . . . qu . . .* in (27) is neither the governing verb nor the predicator, since the dependent in extraction is selected by the predicator in the part of the construction introduced by *que*. For a more detailed analysis for Dutch and integration with the tonal morphemes, see Eynde & Van Dooren (1983).

3. The predicator and its dependents: Internal analysis of formulations

3.1 The PA as valency grammar

Dependency grammars conceive the sentence as a sequence of elements that are linked by dependency relations, i.e. *connections*. This dependency network hinges on a *construction kernel* that selects a precise number of *dependents* and imposes their basic features. A verbal *predicator* imposes particular features on its complements; the same goes for the adjectival predicator.[7] A *dependent* in

7. The terminology chosen here to describe dependency relations ("imposes features") does not contradict the bottom-up procedural approach which we advocate in the PA.

turn can be the *construction kernel* of its own *dependents*, and so forth, recursively, from element to element, till all sentence elements have been integrated in the network.

As a first approximation we may say that the *valency* corresponds to the set of predicator-specific dependents (called *actants* by Tesnière), as opposed to the non-specific dependents (Tesnière's *circonstants*), which may be adjoined to the whole category of predicators (e.g time, place or manner syntagms that may be adjoined to any verb).

On the basis of these observations and that of the pronominal paradigms, we can identify 3 different types of dependents.

3.2 Typology of dependents

The (dependent) elements in a sentence will be classified in different classes according to the following criteria.

Tesnières terms *actant* and *circonstant* are used here for convenience, to indicate resp. predicator-specific vs. predicator-general dependents; both terms of Tesnière are concise and well established in French linguistics. The term *appendage* is used because, to our knowledge, no term is available to indicate the class of non-referential adjuncts, such that they may be distinguished from the larger set of adjuncts.

Proportional to a paradigm of pronouns	Predicator-specific	Obligatory	Terminology
+	+	±	actant
+	−	−	circonstant
−	−	−	appendage

Examples for actant (28), circonstant (29), and appendage (30) are:

(28) *He takes the book*

(29) *They are eating in the garden*

(30) *All things considered, it was the best solution*

The remaining part of this chapter focuses on the elements important to valency analysis, i.e. the predicator-specific dependents (*actants*).

3.3 Typology of actants

The distributional analysis of the actants for French produces an inventory of ten paradigms. The following table shows the set of clitic pronouns and a representative subset of non-clitics and suspensives.[8]

Table 2. Classification of pronouns

	Clitics	**Non-clitics**	**Suspensives**
P0	je tu il elle ils elles on $nous_0$ $vous_0$ ça(ce) il. . .en_0(Q)	ÇA ceci moi toi LUI ELLE eux VOUS NOUS ELLES	qui que..il
P1	le la les me_1 te_1 $nous_1$ $vous_1$ en_1(Q) se_1	ÇA ceci moi toi LUI ELLE eux VOUS NOUS ELLES	qui, quoi que
P2	y_2[à_ça] lui_2 $leur_2$ me_2 $nous_2$ $vous_2$ se_2	(à + non-clitic)	à + qui/quoi
P3	en_3 [de_ça] lui_3 $leur_3$ me_3 $nous_3$ $vous_3$ se_3	(de + non-clitic)	de + qui/quoi
PL	y_L[là]	là ici	où
PD	en[de_là]	de_là d'ici	d'où
PT		alors	quand
PM		comme_ça	comment
PQ		autant	combien
(PP)			PREP + qui/quoi

8. Note for Table 2: Capitals indicate stressable non-clitic pronouns that have clitic homophones.

(i) NOUS, nous travaillons. NOUS, on travaille.

An ellipsis (. . .) joins the parts of a discontinuous morpheme. In these cases *il* is not a pronoun, but just a function index (P0 — *subject*) of the accompanying element. It is also found in the lexicalizations by *il . . . que*+sentence or *il . . . de*+(construction with) infinitive.

(ii) a. *qu'arrive-t-il ? que..il* "what is happening"
b. *il manque ça/ceci*
il..ça/il..ceci "that/this is missing"
c. *il arrive que des gens changent leur façon de vivre d'un jour à l'autre*
÷ ça arrive
"it happens that people suddenly change their way of living"

In *y*[*à_ça*] etc. the pronominal forms between square brackets are the non-clitic proportional forms that are necessary for the disambiguation of some clitics with multiple functions.

The paradigms P0, P1, and P2 correspond roughly to the subject, direct object, and indirect object of traditional grammar. Example (31) compares how four different predicators with P2 are proportional with particular pronouns. This illustrates how the pronominal paradigms contribute relevant information by using pronoun clusters, rather than isolated pronouns.

(31)	a.	*parler*	(to speak)	**y*	**à ça*	*lui*	(*à lui*)[9]
	b.	*penser*	(to think)	*y*	*à ça*	**lui*	*à lui*
	c.	*appartenir*	(to belong to)	*y*	*à ça*	*lui*	(*à lui*)[10]
	d.	*remédier*	(to remedy)	*y*	*à ça*	**lui*	**à lui*

Paradigm P3 (32) corresponds to a non-clitic complement introduced by the preposition *de*, the whole being proportional to the clitic *en*. In this list, PL(ocation), PD(irection), PT(ime), PM(anner), and PQ(uantity) refer to the *obligatory*, predicator-specific dependents.

(32) *il s'en souvient, de lui; de cette époque de sa vie*
"he remembers him; he remembers that period of his life"

(33) *il* y_{PL} *va*, [$_{PL}$ *au musée*], [$_{LOC}$ *à Paris*]
"he goes to the museum, in Paris"

(34) *j'en reviens, du magasin*
"I return from the store"

(35) *les événements se déroulent à Paris en 1789*
"the events take place in Paris in 1789"

(36) *on dénomme cette plante ainsi*
"this is what this plant is called"

(37) *ses frais de déplacement lui coûtent 200 francs*
"his travelling expenses amount to 200 francs"

Some predicators also select one or more prepositional complements (PP). These are characterized by the preposition or by the set of prepositions they select.

9. The form *à lui* only appears in environments that exclude the clitic: *je ne parle qu'à lui, c'est à lui que je parle.*
10. See fn. 9.

(38) *je vous désapprouve dans/*comme votre comportement*
"I disapprove of your behavior"

(39) *ils l'ont désigné comme/*dans organisateur de la fête*
"they designated him as the organizer of the party"

The restrictions on combination with different types of actants are one of the main characteristics of the predicator. However, one observes other types of restrictions on constructions imposed by the predicator, such as different types of passive constructions and alternations. The systematic analysis of these restrictions, which is necessary in order to give a complete picture of the valency of the predicator, will be described in the following Section.

4. Relations between formulations: External analysis of formulations

4.1 Reformulations

The different referents in construction (40a) also appear in construction (40b), although with a different syntactic function (and many times in a different order). The paradigm P1 in construction (a) is coreferent with paradigm P0 of construction (b). Between these two formulations exists a relation of *reformulation*: construction (a) implies construction (b), and vice versa. Some verbs (41) have an additional reflexive passive with the same relation of implication between (a), (b), and (c). This (logical) equivalence relation is considered fundamental. Therefore, when the implication is unidirectional, the label *reformulation* is not applied.

(40) a. *je*$_i$ *le*$_j$ *retrouve facilement*
"I find it/him easily"

b. *il*$_j$ *est retrouvé par moi*$_i$
"it/he is found by me"

(41) a. *ils chargent le camion*
"they load the truck"

b. *le camion est chargé par eux*
"the truck is loaded by them"

c. *le camion se charge par les conducteurs*
"the truck is loaded by the drivers"

The passive formulation is not considered a *transformation* of the corresponding active formulation. Not all active formulations with P1 have a corresponding passive formulation: the existence of reformulations is a characteristic of the individual predicator (42).

(42) a. [active] *il double*$_1$ *cette Peugeot*
"he overtakes this Peugeot"
passive: *la Peugeot a été doublée*$_1$ *par lui*
"the Peugeot is overtaken by him"
b. [active] *il double*$_2$ *cette année*
"he redoes this school year"
[*passive] **cette année a été doublée*$_2$ *par lui*
"this school year is redone by him"

4.2 The formulation group

The set of formulations that are considered reformulations of each other is called the *formulation group*. Constructions (43a) and (43b) belong to the same formulation group.

(43) a. *ils chargent le camion de charbon*
"they load the truck with coal"
b. *le camion est chargé de charbon par eux*
"the truck is loaded with coal by them"

From here on the referents that (may) appear in different formulations with different syntactic functions (i.e. different P-paradigms) will be coined *valency terms* (44). These terms are indicated as T1 and T2 in the examples below.

(44) a. *ils chargent le camion* — *ils*=P0=T1, *le camion*=P1=T2
b. *le camion est chargé par eux* — *le camion*=P0′=T2, par eux=PP=T1
c. *le camion se charge par les conducteurs* — *le camion*=P0′=T2

This implies the (partial) dissociation of the content of the referential paradigm from its syntactic function in a particular formulation: a term thus is defined by (a) the set of its possible positional paradigms (function features) and (b) the (non-function) features of its pronominal cluster(s).

4.3 Linked formulation groups

As mentioned above, a formulation consists of the predicator and the set of paradigms it governs, each paradigm being marked for its compulsory or optional nature and its basic referential features. These features are indicated by the list(s) of pronouns in the paradigm. But all formulations are also characterized by the relation of reformulation that links them with other formulations. This is the case for the two formulations in (45), as it is the case for the two formulations in (46). But whereas there is an equivalence relation between (45a) and (45b), and also an equivalence relation between (46a) and (46b), there is only a *unidirectional implication* between the formulation set of (45) and the set of (46): each of the formulations of (45) implies (each of) the formulations of (46), but not vice versa. In the cases where such single implicational relations do exist, we use the term *linked formulation groups* (cf. Eynde & Eggermont (1990)).

(45)	a.	*ils chargent le camion de charbon*	active formulation I
		le camion est chargé de charbon par eux	passive formulation I
(46)	a.	*ils chargent le charbon sur le camion*	active formulation II
	b.	*le charbon est chargé sur le camion par eux*	passive formulation II

These examples have lexicalized constituents. The procedure of testing equivalence and implication between formulations has to be applied on pronominal sentences. The absence of such a link between (47) and (48) is not obvious. This is the reason why we can not establish this type of link between

(47) *le soleil sèche la chemise*
"the sun dries up the shirt"

(48) *la chemise sèche au soleil*
"the shirt is drying in the sun"

However, the absence of a controllable implication here becomes obvious when we compare their pronominal counterparts in (49) and (50).

(49) *il la sèche*
"it dries it up"

(50) *elle sèche là-bas*
"it is drying there"

In such cases it is not the syntactic structure, but the happenstance of lexical

resemblances that creates the implication. Such cases are instances of the *lexical pitfall*: mistaking lexical links for constructional links.

The importance of these properties for the syntactico-semantic description of the predicator has been documented in work on verb typology during recent decades (Levin 1996, Levin & Rappaport 1993), but so far no formal procedure or methodology for the registration of these properties has been laid out. In Section 5.3 we will return to this matter and give some examples of the results that have been achieved working with verb typology within the framework of the PA.

4.4 Valency

The above procedures refine the concept of valency, the set of predicator-specific dependents, as it was sketched in the introduction. To characterize these predicator-specific constituents correctly, it is necessary to specify in a single formulation their number, their syntactic function (subject, direct or indirect object, various complements), and their syntactic nature (prepositional or not), but this is not sufficient. Because of the *equivalence* (mutual implication) between the formulations of the same group, it is also necessary to specify the actants for *all* the formulations of the group (Eynde *et al.* (1998)).

4.5 Syntactic typology of pronouns and paranouns in French

Table 1 (see appendix) shows the inventory of clitic, non-clitic, and suspensive pronouns in French. It also shows the paranouns of French (termed indefinites by Harris). These classes are then subdivided along another dimension on the basis of the syntactic status of the elements, in accordance with the classification proposed in Eynde (1995).

— The element is of the *pro-syntagm* type if it appears exclusively in a particular paradigm because it then directly inherits the function feature of the paradigm. The pronouns *je, tu, il, . . .* have the built-in feature of subject reference. The existence of homonymous pronouns as *nous, vous, se, . . .* that appear in multiple paradigms does not impede the classification: they are disambiguated through the solidarity relation with the other paradigm-specific pronouns. They are given an index indicating their syntactic function.
— The element is of the *pro-referent* type if it lacks a (built-in) feature indicating its particular syntactic function but otherwise covers the whole referent

of the paradigm. Its function may be indicated by the preposition preceding the element, or by the (significant) absence thereof. For example, the non-clitic pronoun *toi* may designate a referent as subject, direct or indirect object, etc. but if preceded by *à* the subject interpretation will be automatically discarded.[11]

— The element is of the type *pro-kernel + determiner of identification* if it is proportional to a complex referent consisting of a kernel determined by a demonstrative, a possessive, a relative construction or a prepositional syntagm.

(51) *lequel* (which one) ÷ *ce bureau* (this desk)
lequel ÷ *mon bureau* (my desk)
lequel ÷ *le bureau où tu travailles* (the desk where you work)
lequel ÷ *le bureau près de la fenêtre* (the desk next to the window)

– The element is of the type *pro-kernel + determiner of connection* if it is proportional to a complex referent consisting of a kernel determined by a possessive.

(52) *le mien* (mine) ÷ *mon bureau* (my desk)

– The *pro-determiners of identification* (53) and the *pro-determiners of connection* (54) are necessary for the definition of the valency of complex predicators with support verbs. Otherwise they mainly serve the definition of the nominal valency.

(53) *prendre la responsabilité de . . .* "take responsibility for . . ."
en prendre la responsabilité "thereof + take + responsibility"
prendre cette responsabilité "take that responsibility"

(54) *mettre à ma/ta/sa/. . . disposition* "place at my/your/his disposal"
mettre à la disposition de . . . "place at the disposal of . . ."

This classification is reminiscent of Harris (1957 [1970:409-433]), where he proposes the notion of pro-morpheme, with different sub-types: pro-*S*, pro-*V*, pro-*A*, pro-*N*, pro-*NP*. On the one hand, the way it is used and defined in the

11. Prepositions are functionally ambiguous: a complement introduced by *à* may indicate an indirect object, a temporal, locative or manner syntagm, a prepositional complement. The pronoun *ça* is a pro-referent, but *contre ça* and *pour ça* are pro-syntagms.

PA, the term *pronoun* covers many more elements than is usual in the grammatical tradition.[12] On the other hand, some elements that are considered as pronouns in many a tradition will be qualified as *pseudo-pronouns*, either because they do not form any paradigm with other elements (as in the case of the so-called impersonal pronouns) (55) or because they form a discontinuous unit of the predicator (56a) and cannot be lexicalized, i.e. replaced by a proportional lexical string (55) and (56).

(55) a. *il neige*, **elle neige*, **ils neigent*
"it is snowing"
b. *il y a un petit problème*, **elle y a un petit problème*
"there is a little problem"
c. *c'est un truc* **elle est un truc*
"it's a trick"

(56) a. ***en*** *vouloir à quelqu'un* *de + inf.*
"to resent someone doing sth."
b. *je lui* ***en*** *veux de m'avoir trompée*
"I resent him/her having cheated on me"
c. **je lui veux de ça/de m'avoir trompée*

The list of pronouns is given in Table 1 (see appendix). This list does not contain pronouns the occurrence of which can be predicted from other pronouns already included in the list, e.g. the pronoun *je* "I" implies the occurrence of pronouns *tu* "you" and *nous* "we", but not vice versa.

4.6 The Proton project

The Proton project (1986–1992) directly led to the construction of verbal valency dictionaries for French and Dutch. The French lexicon contains 8600 formulation groups covering 3700 morphological verbs. For each entry, i.e. for each formulation group, the structured database specifies

1. Number, type and content of the paradigms, as a list of accepted pronouns,[13] including the possible nature of their lexicalizations (subordinate clause, infinitive, . . .).

12. In some publications based on the PA the term pronoun was replaced, following Hiż (1969), by the term *referential*.
13. The list omits pronouns whose occurrence can be predicted from other pronouns, cf. supra.

2. Paradigm preposition (if any).
3. Reformulations (5 different types of passive).
4. Links to related formulation groups.

The Proton database was integrated in the Siemens Metal system for automatic translation. A similar but smaller database has been implemented for Dutch (Van Langendonck *et al.* 1990), Russian (Soldatjenkova 1996), and Japanese (Cormo 1998). The same method has been applied to Chinese (Zhao 1995, Daugaard 1995).

5. Applications of the pronominal approach: verb classification in Danish

This section describes the approach to verb classification for Danish that has been based on the Odense Valency Dictionary (OVD). The OVD was a project at the University of Odense (1994–1998) with the aim of developing the first valency dictionary for Danish verbs to be used for human inspection as well as in computational applications. To compile the dictionary, the Pronominal Approach developed for French by the Proton project in Leuven was adapted to Danish (Schøsler & Kirchmeier-Andersen 1997b). At present the OVD contains approximately 4,000 verb senses corresponding to 1,900 Danish verbs.

The OVD records different types of syntactico-semantic information about the verb and its combinatorial potential. In the following we shall take a closer look at some of this information and investigate the way in which it may contribute to a semantic classification of Danish verbs.

5.1 Establishing the pronominal paradigms for the OVD

Since a complete distributional account of Danish pronouns and proforms was not available, the development of the OVD began with the establishment of a complete inventory of Danish pronouns and their distributional properties. Using the suspensive forms as a point of departure, the morpho-syntactic properties of the pronouns were related to each other in a two-dimensional grid, as illustrated in Table 3 for the group of pronouns which stand in a proportional relation to the suspensive form *hvem* "who".

Table 3. Morpho-syntactic properties of proforms

Features		Feature values								
referentiality		+speaker		−speaker		−speaker				
addressability		−addressable		+addressable		−addressable				
nb[14]		+sg	−sg	+sg	−sg	+sg		−sg		
sex						+fem	−fem			
case	+subj	jeg	vi	du	I	hun	han	de/De		
		I	*we*	*you*	*you*	*she*	*he*	*they*/*You*		
	-subj	mig	os	dig	jer	hende	ham	dem/Dem		
		me	*us*	*you*	*you*	*her*	*him*	*them*/*You*		

Within the proportionality relation to *hvem* further correspondences could be established, e.g. that *jeg* , *vi*, *du* etc. were related by virtue of occurring in the same relation to verbal predicators (i.e. the subject position) and that they contrasted to the forms occurring in a different relation (i.e. in non-subject position) (cf. Section 2.3). In the paradigmatic dimension the pronouns were grouped according to the type of entities they could denote, i.e. singular or plural entities, masculine or feminine, and some of their discourse properties, i.e. referentiality (e.g. could they be used to refer to the speaker) and addressability (e.g. could they be used to address discourse participants directly). The result is that each pronominal form corresponds to a unique combination of feature values in the grid.

We can use Table 3 to specify a feature structure for each pronoun. The pronoun *du* "you" for example carries the features: referentiality=−speaker, addressability=+*addressable*, case=+*subject* (P0), sex=unmarked, number=+*singular*. The pronoun *hvem* itself, however, is not marked for any of these features although each one of them is acceptable in answers to a question phrased with *hvem*. Thus, *hvem* can be described as representing a feature potential which can be instantiated by any of the forms in Table 3. We may say that a part of the feature structure of *hvem* subsumes the feature structures associated with the related pronouns.[15]

14. The singular–plural distinction does not apply to addressable and speaker-referential pronouns in exactly the same way as to others. For the forms *jeg* and *vi* for instance the distinction would be "speaker only" vs. "speaker + other referents".

15. The subsumption is only partial as it does not concern the discourse features of *hvem*. Thus, the feature ±*suspensive* is not a feature shared with the proportional pronouns.

The labeling of the properties with features and their values is of secondary importance. The primary focus is on observable relations of proportionality between the pronominal forms.

Progressing as described above it became possible to classify 94 pronominal forms for Danish on the basis of 26 different features.

In the next step a set of semantic labels could be defined formally on the basis of the syntactico-semantic features (e.g. the label *+human* can be assigned to elements in a paradigm characterized by the feature *+speaker*, whereas *−human +concrete* can be assigned to elements in a paradigm characterized by the feature *++near*)[16] and as such the traditional semantic features constitute a secondary classification. This is obvious from the semantic hierarchy in Figure 1 below derived from the singular forms of the pronouns in paradigmatic relation with *hvem* "who" and *hvad* "what", with the semantic labels added in parentheses.[17]

The hierarchy shows that distinctions traditionally made on the basis of the semantic labels correspond to the syntactico-semantic features derived from the pronominal forms.

In the OVD the semantic labels are computed automatically from the syntactico-semantic feature descriptions of the formulations in order to improve readability and inspectability of the database after the completion of the coding process.

The formulations in the Proton dictionary are based on individual pronominal forms, but those in the OVD are based on clusters of pronouns. This clustering of pronouns shows that not all features related to pronominal forms are relevant for the description of valency properties and for the distinction of different senses of a predicator.

For instance, grammatical gender in Danish (neuter vs. uter)[18] or certain features related to the attributive use of a pronoun do not seem to play a crucial role in the establishment of valency properties and the distinction of verb meanings and have consequently been ignored. The same applies to certain distinctions which can only be observed with a very limited number of

16. We have observed three deictic dimensions in Danish: *den* "that" (−near), *denne* "this" (+near), *denne her* "this one here" (++near).

17. The feature *proposition* is assigned on the basis of the observation of a paradigm consisting of the pronoun *det* "it" and different forms of subordinate and infinitive clauses. The feature *near* is a deictic feature addressing the distance between the speaker and the object referred to.

18. Uter combines the former masculine and feminine genders.

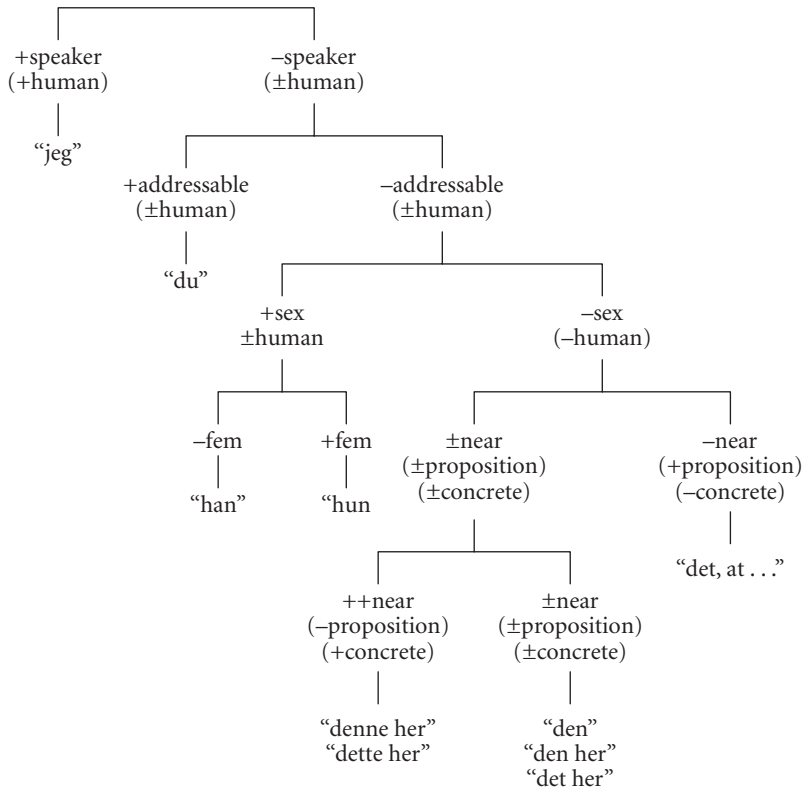


Figure 1. Semantic hierarchy

predicators, such as biological sex. Once the valency-irrelevant features have been taken out of consideration, a number of pronouns turn out to have identical values for the remaining features. This holds for the pronouns *det* "it", *nogen* "some", *et eller andet* "one or the other" (neut), *en eller anden* "one or the other" (utr), *ingen* "none", and others. As shown in Table 4 below, a reduction can be achieved by grouping pronouns with identical feature structures together.

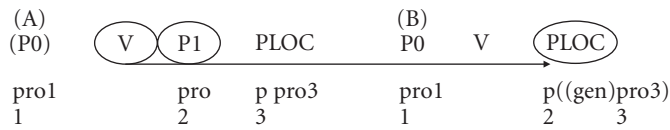
Table 4. Grouping of pronouns with identical feature structures

	loc	dir	tim	dur	man	rea	qua	sub	obl	sg	pl
det	–	–	–	–	–	–	–	+	+	+	–
nogen	–	–	–	–	–	–	–	+	+	+	–
et eller andet	–	–	–	–	–	–	–	+	+	+	–
en eller anden	–	–	–	–	–	–	–	+	+	+	–
ingen	–	–	–	–	–	–	–	+	+	+	–

5.2 Linked formulation groups and argument alternations

In Schøsler and Kirchmeier-Andersen (1997a) we have shown that it is possible to establish fairly homogeneous semantic groupings of Danish verbs if we apply the criterion of linked formulation groups, a concept most commonly referred to as argument alternations (Levin 1993). A systematic description of argument alternation in Danish is given in Schøsler & Kirchmeier-Andersen (1997b). Within the PA, three main linking patterns have been defined. Here we use only the *merging* pattern, with its three subtypes (Schøsler and Kirchmeier-Andersen 1997a). The *permutation* pattern and the *permutation and reduction* pattern (Kirchmeier-Andersen 1997a) are not considered here. The basic properties of the merging relation are illustrated in Figure 2.[19]

The concept of linked formulation groups corresponds very closely to the argument-alternation criterion that Levin (1993) used for the classification of English verbs. However, the account provided by the PA has a formal basis. Approximately 50% of the alternations which can be accounted for in Danish differ from the ones identified by Levin for English.



(A) *Hun* (P0) *stikker ham* (P1) *der* (PLOC)/*i denne her* (pro)
Mary stikker John i maven
"Mary stabs John in the stomach."

(B) *Hun* (P0) *stikker der* (PLOC)/*i denne her* (pro)
Mary stikker i Johns mave
"Mary stabs in John's stomach."

Figure 2. Merging pattern

19. Note that while *permutation* and *permutation and reduction* patterns can be accounted for on the pronominal level, the *merging* pattern cannot, since there is a merge of two distinct forms (*him there*) to one form (*it* o*r there*) which, however, contains something resembling the two forms at a lower level (*his* X). While we can argue for a relation between *him* and *his*, there is no provable relation between *there* and *X* as *X* is necessarily always lexicalised. The number of lexical items which can appear in both constructions is, however, limited to items referring to things which can be inalienably possessed.

5.3 Classification criteria in the OVD

In this section, we will discuss only four of the eleven distinctive features used in the OVD. The proposed classification limits itself to lexical verbs as opposed to function verbs (cf. Section 1) and is based on the properties of their actants, i.e. complements which are proportional to verb-specific pronominal paradigms. The four features constitute the core description of the combinatory potential of a verb, as they focus on:

1. the formulations of the predicator: the number and nature of dependents.
2. the pronoun clusters: the choice of preposition and syntagmatic class (phrase type) within the formulation.
3. the pronoun clusters: the selectional restrictions within the formulation.
4. the nature of the linked formulation groups: different construction possibilities.

In the OVD the relations between linked formulation groups are expressed as a relation between two or more entries. There is a progression from very general properties of the dependents to more and more specific features, all derived from pronoun distribution. The classification of the verbs is based on the properties of the dependents because this relation is basic and does not presuppose any of the other features.

The following paragraphs present examples of the verb classes which can be extracted from the OVD using search criteria constructed from the four features. The features are organized in a hierarchy (Figure 3). This helps us to identify the features used to specify the distributional class of a verb, and makes it easier to compare the features of different verb classes. The formulation pattern of the predicator is the first criterion to be considered. Criteria 2 and 3 focus on different properties revealed by pronominal clusters. Since they depend on each other it seems convenient to combine the syntactic and semantic features of the dependents in the same branch of the hierarchy. We need further research in this domain before we will be able to propose a more hierarchical ordering of the main criteria. The list of features is not closed since further criteria remain to be investigated.

The hierarchy in Figure 3 is based on monotonic inheritance, i.e. the lower types inherit all the information of the higher, less specific types. For example, the feature +*human* as a value of the pronominal cluster type of the P1 inherits the feature specifications of its supertypes. Thus, the full description of the constituent type P1 is: +proportional/−proposition/+concrete/+human.

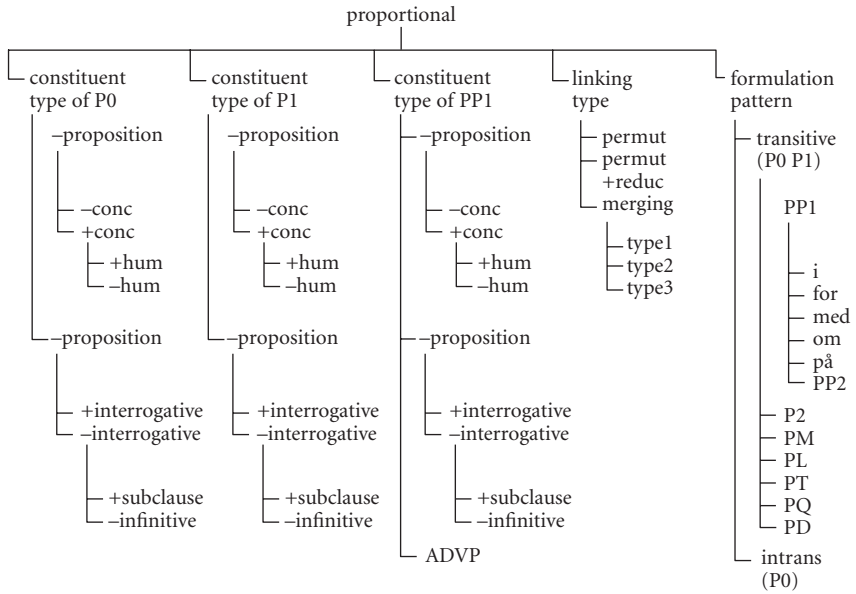


Figure 3. Hierarchy of features

5.3.1 *Criterion 1: formulation patterns*

Our starting point for a further classification based on formulation patterns is verbs, which combine with at least a P0 and a P1 paradigm. This corresponds to the search profile: formulation pattern: P0_P1, followed or not by additional dependents. Out of 3114 lexical verbs in the database, 1422 match this criterion (46%). Table 5 shows the distribution of these verbs over different formulation patterns.

Table 5. Distribution of verbs across formulation patterns

1	P0_P1	907
2	P0_P1_P2	33!
3	P0_P1_PP1	335
4	P0_P1_PP1_PP2	12!
5	P0_P1_PL	37!
6	P0_P1_PT	0
7	P0_P1_PD	40!
8	P0_P1_PQ	4!
9	P0 P1 PM	54!
	Total	1422

There are two very large classes. 907 entries have the pattern P0_P1 and 335 verb readings have the pattern P0_P1_PP1. These two classes are internally heterogeneous, and further criteria are necessary in order to make a sensible sub-classification (see the following subsections). However, within the smaller classes (marked (!)) verbs with clear similarities can be identified already at this point. For instance, P0_P1_P2 contains two types of verbs: verbs of donation and verbs of information transfer. These two types are distinguished by another criterion, namely the pronominal cluster type of the P1, i.e. *−proposition* for verbs of donation and *+proposition* for verbs of information transfer.

(57) *Hun gav ham bogen*
"She gave him the book"

(58) *Hun fortalte ham historien*
"She told him the story"

The list of verbs with the pattern P0_P1_PP1_PP2 contains only verbs indicating change of state or change of location from one site to another:

(59) *Han oversætter teksten fra dansk til fransk*
"He translates the text from Danish into French"

(60) *Han flytter stolen fra køkkenet til badeværelset*
"He moves the chair from the kitchen into the bathroom"

The difference between these two classes of verbs is clarified if we consider the selectional restrictions on the P1 which is –concrete for change of state verbs and +concrete for change of location verbs.

The class P0_P1_PM contains verbs of personal evaluation or verbs of utilization:

(61) *Han betragter ham som sin ven*
"He considers him his friend"

(62) *Han bruger sin sko som hammer*
"He uses his shoe as a hammer"

In this case we have not yet found distributional features that might confirm our intuition of the existence of two groups. In Table 5 we also find an empty set of verbs, i.e. the pattern P0_P1_PT. We have not yet encountered verbs of this type, but we cannot exclude the possibility that there may be some.

5.3.2 *Criterion 2: choice of preposition and syntagmatic class (phrase type)* Some of the large verb groups can be further subclassified according to the type of the pronominal clusters in the different paradigms of the formulation. For example, the pattern P0_P1_PP1 was found with 335 verb senses, not constituting a homogeneous class. To test whether this criterion groups together verbs which have similar or related meanings, subclasses of verbs with identical cluster types were extracted. The verbs were first extracted on the basis of the form of the preposition heading the PP1. In the group where the preposition introducing the third dependent was *om* the verbs turned out to be verbs of communication.[20] This group can be further subdivided into three groups, determined by the type of pronominal clusters headed by the preposition *om* in the PP1.

— Verbs of asking permission: *om* is followed by a subordinate clause or an infinitive clause.
— Verbs of information: *om* is followed by pronouns which can be grouped together with *det* (a cluster denoting nominal constituents) or a subordinate clause introduced by the complementizer *at* "to".
— Verbs of interrogation: *om* is followed by a nominal constitutent or a suspensive subclause.

Syntactic properties	Examples	Semantic paraphrase
PP1 = *om* + inf/subclause	*anmode om* (request), *ansøge om* (apply for) *bede om* (ask for)	ask permission
PP1 = *om* + *at* + subclause	*minde om* (remind of), *underrette om* (notify of),	give information
PP1 = *om* + **at* + subclause	*spørge om* (ask about), *udspørge om* (question about),	question

The examples show how Danish and English verbs use different mechanisms for encoding the difference between these verbs. In English, the form of the

20. The Danish preposition *om* corresponds to different prepositions in English, including *in*, *for*, *of* and *about*.

preposition introducing the PP1 seems to be a sufficient criterion, whereas in Danish we also have to consider the constituent type.

5.3.3 *Criterion 3: selectional restrictions*

The second very large group of verbs — those with the formulation pattern P0_P1 — can also be subdivided on the basis of the constituent type of the dependents. In what follows we will discuss an example where selectional restrictions can be used to make further subclasses. The number of verbs extracted using the criterion *–proposition* leaving us with only nominal constituents is still very large, and the verbs do not seem to share any obvious common characteristics. Therefore, a new set of extractions was made using the following more specific extraction profiles:[21]

Profile 1

Formulation pattern:	P0 _ PP1
CLUSTER TYPE:	P0 = NP, P1 = NP
SELECTIONAL RESTRICTIONS:	P0 = +HUM, P1 = +HUM
Argument alternation:	no

It turns out (not surprisingly) that the verbs extracted using profile 1 express exclusively human interaction, i.e. what only humans do to other humans. This excludes e.g. verbs of sensation, verbs of feeling, verbs of physical aggression, which are common to e.g. animals and humans. Verbs of this type mainly concern social activities like the verbs *ansætte* "employ" and *afskedige* "dismiss" with an extraordinary number of synonyms. The class also includes verbs of psychological aggression like *diskriminere* "discriminate" as well as verbs like *bisætte* "perform a funeral service".

In many cases the most satisfactory results are obtained by comparing verbs which do not differ in any other feature except the selectional restrictions on one of the dependents. This is illustrated by the extraction profiles 2 and 3 where the verbs only differ in the set of selectional restrictions placed on the P0.

Profile 2

Formulation pattern :	P0 _ P1
CLUSTER TYPE:	P0 = NP, P1 = NP
SELECTIONAL RESTRICTIONS:	P0 = +HUM, P1 = +CONC
Argument alternation:	no

21. NP in this context means non-sentential

Profile 3

Formulation pattern:	P0 _ P1
CLUSTER TYPE:	P0 = NP, P1 = NP
SELECTIONAL RESTRICTIONS:	P0 = +CONC, P1 = +CONC
Argument alternation:	no

Extraction profile 2 brings out verbs which we may call *crafting verbs* such as *stive* (*en skjorte*), *bukke* (*et rør*), *lade* (*en kanon*), . . . "starch (a shirt), bend (a tube), load (a cannon), . . .". Extraction profile 3 is less restrictive as it accepts P0 proforms which may refer to both human and non-human concrete entities. These verbs denote activities which are common to humans and non-humans, particularly animals. We may call them *body-function verbs* as they include verbs like *indånde* (*gas*), *synke* (*et æg*), *udskille* (*sved*) . . . "inhale (gas), swallow (an egg), secrete (sweat) . . .".

5.3.4 *Criterion 4: linked formulation groups*

In Section 5.2 we described argument alternation of the merging type. The linked formulation groups P0_P1_PP1 ~ P0_P1_PL and P0_P1 ~ P0_PL characterize verb classes which are related in this way. Homogeneous subclasses can be extracted from these verb classes. Aggression verbs are in subgroup P0_P1_PL ~ P0_PL (merging type 1) and caressing verbs in subgroup P0_P1_PL ~ P0_P1 (merging type 2). Further subclassification of the last group is possible on the basis of the inherent aspectual properties of the verbs. In the subgroups P0_P1 ~ P0_P1_PP1 (merging type 3a and 3b), verbs of obstruction and verbs of support have the preposition *i* "in" introducing the PP1. Verbs of obstruction and support can be distinguished from verbs of reproach and appraisal characterized by the preposition *for* "for" introducing the PP1. The OVD-database does not offer criteria for further subclassification within the two groups.

Merging type 1: aggression verbs:
Hun stikker ham i maven
"She stabs him in the stomach"
P0 P1 PL
Hun stikker i hans mave
"She stabs in his stomach"
P0 PL

Merging type 2: caressing verbs:
Hun kysser ham på kinden
"She kisses him on the cheek"
P0 P1 PL
Hun kysser hans kind
"She kisses his cheek"
P0 P1

Merging type 3a: verbs of obstruction or support
Hun afbryder/støtter ham i aktiviteten
"She interrupts/supports him in the activity"
P0 P1 PP1
Hun afbryder/støtter hans aktivitet
"She interrupts/supports his activity"
P0 P1

Merging type 3b: verbs of appraisal or reproach:
Hun roser/kritiserer ham for indsatsen
"She praises/criticizes him for the effort"
P0 P1 PP1
Hun roser/kritiserer hans indsats
"She praises/criticizes his effort"
P0 P1

In all three cases the dependents involve an agent, a possessor and a possessum. The difference in the constructions lies in the way the same constituents are expressed in the parallel constructions. In the first case the possessor is expressed as the P1 and the possessum is expressed as a locative (PL). They are merged together into a locative in the alternative construction. In the second case the possessor and the possessum are merged in a P1 instead of a locative in the alternative construction, whereas in the third case, the possessum is not expressed as a locative but as a prepositional complement (PP1) which is expressed as a P1 in the alternative construction.

We may conclude that the criterion of linked formulation groups in the cases described above provides clearly delimited groups.

So far, only a limited portion of the OVD has been investigated by means of extraction profiles like the ones described above, and in many cases semantically homogeneous classes could be established. There are, however, exceptions. For instance, in order to distinguish verbs of appraisal from verbs of

reproach, criteria which specify an antonymic relation need to be found. Furthermore, some extraction profiles reveal classes of verbs which do not seem to be semantically related but which turn out to have other features in common, for instance the same derivational pattern.

6. Conclusion

We have shown that based on a systematic analysis of pronominal features it is possible to obtain clear and consistent categorizations and typologies of predicators and dependents in French and Danish unbiased by traditional categorizations or intuitively established classifications (e.g. the lexical pitfall). The analyses of the language-immanent system based on the Pronominal Approach reveal new and important aspects even of well-described languages such as French, and become of crucial importance for less well-described languages such as Danish. Moreover, by not imposing so-called universal theories, one avoids the danger of neglecting important aspects of the individual language. Because of space limitations, we prefer not to make a comparison of the procedure of the PA and the equivalent procedures based on linear distributional criteria. We are aware that these criteria yield the same result, but the proportionality criterion seems less laborious.

For us, the question was whether it is possible to find a non-intuitive, non–a-prioristic way to obtain a structure taking into account a third relational axis, the syntax-semantics interface.

The two traditional dimensions of sentence structure (the syntagmatic and the paradigmatic axis) are insufficient as well for the understanding of the message as for its analysis. Deciphering a message necessarily implies taking into account the *hidden* dimension(s): the hearers, as well as the speakers, have at their disposal their knowledge of the multi-dimensional structure. The hearer projects the missing dimension(s) in the linear chain so as to restore, to re-create the complete set of features that make the message *understood*.

Just as distinct sound units have to be treated on two levels (phonemic and morphophonemic), the pronouns have to be treated first as single distinctive reference units, and afterwards — because of the way they appear in paradigms — have to be reanalyzed as units (maybe homophonous) of particular clusters of pronouns. It is these clusters of pronouns, together with the (clusters of) paradigms, that will provide the ultimate syntactico-semantic features.

Our methodology for syntactic analysis corresponds to Harris's (1951) methodology for phonemes and morphemes. In parallel with Harris's justification of the distinction between phoneme and morphophoneme, we might consider the isolated pronouns as having merely morphemic status, and clusters of them in particular constructions having a "syntagmo-morphemic" status. The common distinctive features of the pronominal clusters serve as minimal syntactico-semantic features that build the grammatical semantics.[22]

In a first stage, the distribution of morphemes in pronominal sentences is studied exhaustively, paralleling Harris's distributional basis for Immediate Constituent Analysis. When this distributional structure is projected upon the lexicalized sentences, the proportionality criterion provides a procedural shortcut for the recognition of elements in the main syntactic skeleton. It is easy then to identify 'left' or 'right' adjuncts to the sentence, to the predicator, to any of the valency or rection paradigms or subsyntagm paradigms, or, once these have been eliminated, as adjuncts to one of the lexicalizing elements. In this respect, the PA parallels most closely Harris's String Analysis.

Our treatment of 'reformulations' differs in an important way from Harris's transformational analysis. Whereas the Harrisian notion of 'transform' covers all kinds of distributionally definable relations between sentences or even between sentences and their nominalizations, we introduced a distinction between four types of 'transform' relations. A first type groups pronominal constructions in which the same (or equivalent) paradigms have identical syntactic function (e.g. clefting or syntagm anteposition). These are termed *dispositives*. A second type is found where different formulations of the same predicator have a (logical) equivalence relationship. Since each of these is a 'reformulation' of the other one, this was termed the *(re)formulation group*. Where we found a unidirectional implication between two (or more) distinct formulation groups of the same predicator, we distinguished a third type, termed the *linked formulation group*. The fourth type covers relations between constructions that have no common predicator, e.g. nominalizations, or where the construction appears to determine one of its own syntagms or referents, e.g. the relative, which is necessarily embedded in the construction of another predicator. For the most part, only the second and third types are important for our valency dictionaries. The only exception so far is for Japanese, where

22. cf. Togeby's "sémantique de la grammaire", which he considers anterior to the lexical morpheme semantics. See also Eynde & Dehaspe (1991).

nominalizations, members of the fourth type, were required to disambiguate homophonous predicators.

As is the case in constructive mathematics, this does not imply the refutation of all traditional concepts. On the contrary, many of them are methodologically relevant since they have become procedurally *provable*, whereas other concepts will have to be replaced by more adequate (and provable) concepts, in such a way that the one-to-one relationship between form and meaning is restored. The constructivist cannot accept, as others apparently do, that the formal structure of the language and the semantic structure are different.

References

Abeillé Anne. 1994. *Les Nouvelles Syntaxes. Grammaires d'unification et analyse du francais.* Paris: Armand Colin.

Bishop, Errett. 1967. *Foundations of Constructive Analysis.* New York: MacGraw-Hill.

Bishop, Errett, & Bridges Douglas. 1985. *Constructive Analysis.* Berlin: Springer Verlag.

Blanche-Benveniste, Claire, José Deulofeu, Jean Stefanini, & Karel van den Eynde. 1984. *Pronom et Syntaxe. L'approche pronominale et son application au français.* Paris: Selaf.

Bronowski, Jacob. 1978. *Magic, Science, and Civilization.* New York: Columbia University Press

Cobuild-Collins. 1996. *Grammar Patterns.* Vol. 1: *Verbs.* London: Harper Collins.

Cormo, Christiane 1998. *On Valency and Sense Distinction of Japanese Verbal Predicators: A constructivist approach.* Ph.D. Thesis presented at the Katholieke Universiteit Leuven, Departement Linguistiek.

Daugaard, Jan (ed.). 1995. *Valency: The Pronominal Approach applied to Danish, Russian and Chinese.* Odense Working Papers in Language and Communication No. 8. Odense: Odense University.

Davidsen-Nielsen, Niels (ed.). 1996. *Sentence Analysis, Valency, and the Concept of Adject.* Copenhagen Studies in Language 19.

Durme, Karen van (ed.) 1997. *The Valency of Nouns.* Odense Working Papers in Language and Communication. Odense: Odense University.

Durme, Karen van & L. Schøsler (eds.) *Studies in Valency IV: Valency and verb typology.* Odense: Odense University Press. (=RASK Supplement Vol. 8.)

Engberg-Pedersen, Elisabeth, L. Falster Jakobsen, & L. Schack Rasmussen (eds). 1994. *Function and Expression in Functional Grammar.* Berlin: Mouton de Gruyter.

Eynde, Karel van den & Claire Blanche-Benveniste. 1978. "Syntaxe et mécanismes descriptifs: Présentation de l'approche pronominale." In: Cahiers de Lexicologie XXXII 1:3–27.

Eynde, Karel van den 1998. "From verbal to nominal valency: Some methodological reflections." In Durme & Schøsler (eds.) (1998: 147–57).

Eynde, Karel van den, Eric Broeders, Carmen Eggermont, & Ludo Vangilbergen. 1988. "Coordination and pronominal feature analysis in French: A computational treatment of ET, MAIS and OU". *Computers and Translation* 3:177–213.

Eynde, Karel van den & Luc Dehaspe. 1991. "The pronominal approach to verbal valency. A formal description of SPEAK, SAY, TELL, and TALK". *Betriebslinguistik und Linguistikbetrieb.* 24. Linguistisches Kolloquium, Bd I, Tübingen, Niemeyer, 273–280.

Eynde, Karel van den & Carmen Eggermont. 1990. "A pronominal basis for computer assisted translation, the PROTON project". *Proceedings of the Maastricht Session of the 1990 Maastricht-Łódż Duo Colloquium on Translation and Meaning in Practice*, Maastricht, 4–6 January 1990, 1–14.

Eynde, Karel van den, Carmen Eggermont & Eric Broeders. 1990. *Dictionnaire Automatisé des Valences des Verbes Français.* Leuven, Departement Linguistiek, Katholieke Universiteit Leuven.

Eynde, Karel van den, Piet Mertens & Pierre Swiggers. 1998. 'Structuration segmentale et suprasegmentale en syntaxe. Vers un modèle intégrationniste de l'écrit et de l'oral'. In M. Bilger, K. van Den Eynde & F. Gadet (eds.), *Analyse linguistique et approches de l'oral. Recueil d'études offert en hommage à Claire Blanche-Benveniste*, *Orbis Supplementa.* Leuven & Paris: Peeters.

Eynde, Karel van den & Karin van Dooren. 1983. "Intonation and syntactic structure in Dutch". *I.T.L. Review of Applied Linguistics*, 60–61:27–42.

Eynde, Karel van den & Karen van Durme. 1998. "A verb typology on a distributional basis. General typology and classification of modals". In Durme, Karen & Schøsler (1998: 9–32).

Goodman, Nelson. 1951. *The Structure of Appearance.* Third edition. Dordrecht/Boston: Reidel.

Goossens, Louis. 1994. "Transitivity and the treatment of (non)prototypicality in Functional Grammar". In Engberg-Pedersen et al. (1994:65–80).

Gross, Maurice (ed.). 1993. "Sur le passif". *Langages* no 109. Paris. Larousse.

Harris, Zellig S. 1951a [1946]. *Methods in Structural Linguistics.* Chicago: Univ. of Chicago Press. (Repr. 1960 as *Structural Linguistics*, Phoenix Books; 7th impression, 1966; 1984.)

Harris, Zellig S. 1962a. *String Analysis of Sentence Structure.* (=Papers on Formal Linguistics, 1.) The Hague: Mouton, 70 pp. (2nd ed., 1964; repr., 1965.)

Harris, Zellig S. 1970. *Papers in Structural and Transformational Linguistics.* Dordrecht, Holland: D. Reidel.

Harris, Zellig S. 1990. "La genèse de l'analyse des transformations et de la métalangue". *Langages* No.99 (Sept. 1990) 9–19. [Ed. by Anne Daladier; translation into French by A. Daladier of most of the essay that introduces Vol. 1 of the present work.]

Heltoft, Lars. 1995. "Danish predicative adjectives and adverbials as valency bearers. In Schøsler & Talbot (1995:211–235).

Hiż, Henry. 1961. "Steps towards grammatical recognition". *Advances in Documentation and Library Science*, vol. III, part 2.

Hiż, Henry. 1969. "Referentials". *Semiotica*, 1.2: 136–166.

Hofstadter, Douglas R. 1980. *Gödel, Escher, Bach: An eternal golden braid.* New York: Random House Inc., Vintage Books.

Kirchmeier-Andersen, Sabine. 1997a. *Lexicon, Valency and the Pronominal Approach.* Ph.D. Dissertation. Odense University. Odense.

Kirchmeier-Andersen, Sabine. 1997b. "Valency, sense distinction and inheritance in different types of nominalisations". In Durme (1997: 59–88).

Levin, Beth & Malka Rappaport Hovav. 1996. *Unaccusativity: At the syntax-lexical semantics interface.* Cambridge: MIT.

Langacker, Ronald 1987. *Foundations of Cognitive Grammar: Theoretical prerequisites.* Stanford University Press.

Langendonck, Willy van, Luc Dehaspe & Erik Broeders. 1990. *Geautomatiseerd Valentiewoordenboek van de Nederlandse Werkwoorden.* Linguistics Department, Katholieke Universiteit Leuven

Levin, Beth. 1991. "Introduction to special issue of *Cognition* on lexical and conceptual semantics". *Cognition*, 41: 1–7.

Levin, Beth.1993. *English Verb Classes and Alternations.* Chicago: The University of Chicago Press.

Melis, Ludo. 1998. "Les relations syntaxiques entre constructions verbales: propositions pour une notation systématique". In Geisler, Hans & Jacob, Daniel (ed) (1998) *Transitivität und Diathese in Romanischen Sprachen.* Tübingen: Max Niemeyer Verlag, pp. 5–19.

Mertens, Piet. 1993. "Accentuation, intonation et morphosyntaxe", *Travaux de Linguistique* 26: 21–69.

Mertens, Piet. 1997. "De la chaîne linéaire à la séquence de tons", *T.A.L.* (*Traitement Automatique des Langues*) 38.1: 27–51. Paris: Klincksieck.

Navarretta, Costanza.1997. "Encoding Danish Verbs in the PAROLE Model". In R. Mitkov, N. Nicolov & N. Nikolov (eds.) *Recent advances in Natural Language Processing*, Tzigov Chark, Bulgaria, pp. 359–363.

Pustejovsky, James. 1995. *The Generative Lexicon.* Cambridge: MIT

Quine, Willard van Orman. 1964. *From a Logical Point of View.* Harvard University Press.

Quine, Willard van Orman. 1970. "Methodological reflections on current linguistic theory". In *Synthese* (October 1970), 21(3–4) & 22(1–2): 386-398. Repr. in Donald Davidson and Gilbert Harman, eds., *Semantics of Natural Language* (Dordrecht: Reidel, 1972), pp. 442–454.

Quine, Willard van Orman. 1987. *Quiddities: An intermittently philosophical dictionary.* Harvard University Press.

Rappaport, Malka & Beth Levin. 1988. "What to do with θ-roles". In Wilkins W. (ed.) *Syntax and Semantics 21*: *Thematic relations* (New York: Academic Press), pp. 7–36.

Schøsler, Lene & Mary Talbot. 1995. *Studies in Valency I*, RASK Supplement Volume 1, Odense: Odense University Press.

Schøsler, Lene & Karen van Durme. 1996. "The Odense valency dictionary: An introduction". *Odense Working Papers in Language and Communication* 13. Odense: Institute of Language and Communication. Odense University.

Schøsler, Lene & Sabine Kirchmeier-Andersen. 1997a. "Valency and inalienable possession". In Baron, Irene & Michael Herslund. 1997, *Possessive Structures in Danish*, Fagling-rapport no. 3, 45–77, Handelshøjskolen i København.

Schøsler, Lene & Sabine Kirchmeier-Andersen. 1997b. *Studies in Valency II: The pronominal approach applied to Danish*. RASK Supplement Volume 5, Odense: Odense University Press.

Shieber, Stuart M. 1986. *An Introduction to Unification-Based Approaches to Grammar*. CSLI Lecture Notes 4, Chicago: University of Chicago Press.

Soldatjenkova, Tatjana. 1995. "A syntactic approach to the valency of Russian verbs". In Daugaard (1995: 89–126).

Soldatjenkova, Tatjana. 1996. *De werkwoordelijke valentie in het Russisch: een constructivistische benadering*. Ph.D. thesis presented at the Katholieke Universiteit Leuven, Departement Linguistiek. (English translation to appear. Leuven/Paris: Peeters.)

Tesnière, Lucien. 1959. *Eléments de Syntaxe Structurale*. Paris: Klincksieck.

Togeby, Knud. 1965. "Grammaire, lexicologie et sémantique". *Cahiers de Lexicologie* 6: 3–7.

Whiteley, Wulf H. 1960. "Some problems of the syntax of sentences in a Bantu language of East Africa". *Lingua* IX.2: 148–174.

Zhao, Yilu. 1995. *Distributional Criteria for Verbal Valency in Chinese*. Leuven & Paris: Peeters.

Appendix

		Clitic	Suspensive	Non-clitic			Paranoun	
PRO-SYNTAGM	P0	*je* tu *il* elle *ils* elles *on* *nous*$_0$ vous$_0$ *ça(ce) il . . .en*$_0$(Q)	*que..il*	il. . .ça	il. . .ceci		il..quelque_chose il..quelqu'un	il..ne..rien il..ne..personne
	P1	*le la les* me$_1$ *te*$_1$	*que*					
	P2	nous$_1$ *vous*$_1$ *en*$_1$(Q) *se*$_1$ *y*$_2$[à_ça] *lui*$_2$ *leur*$_2$ me$_2$ nous$_2$ vous$_2$ *se*$_2$						
	P3	*en*$_3$ [de_ça] *lui*$_3$ *leur*$_3$ me$_3$ nous$_3$ vous$_3$ *se*$_3$						
	PL	*y*$_L$ [là]	*où*	*là*	*ici*	partout	quelque_part	nulle_part
	PD	*en*$_L$[de_là]	*d'où*	*de_là*	*d'ici*	de_partout	de_quelque_part	de_nulle_part
	PT		*quand*	alors		toujours	quelquefois	jamais
	PM		*comment*	*comme_ça*				
	PQ		*combien*	autant			quelque_peu	rien
PR0-REFERENT ±preposition			quoi *qui*	ÇA *LUI* toi ELLE eux VOUS ELLES	*ceci* *moi* NOUS	tout tout_le_monde	quelque_chose quelqu'un	rien personne
PRO-KERNEL + DET of identification			lequel laquelle lesquels lesquelles	celui-là celle-là ceux-là celles-là	*celui-ci* celle-ci *ceux-ci* celles-ci			
PRO-KERNEL + DET of connection		le/la/les + ...		sien tien mien (ne)(s) (ne)(s) (ne)(s) leur (s) votre(s) notre(s)				
PRO-DET of identification			quel quelle quels quelles	ce(t)..là cette..là ces..là	ce(t)..ci cette..ci ces..ci			
PRO-DET of connection				son ton mon sa ta ma ses tes mes leur votre notre leurs vos nos				

Chapter 8

Contextual acquisition of information categories*

What has been done and what can be done automatically?

Benoît Habert
LIMSI — CNRS

Pierre Zweigenbaum
Assistance Publique — Hôpitaux de Paris

Language is clearly and above all a bearer of meaning. (Harris 1991:321)

1. Distributional semantics in the light of current language engineering

Semantic category acquisition and extension is a growing subfield in natural language processing (NLP) and information retrieval (IR), as the use of the Web increases demand for content-based access to textual data.[1]

We are both involved in (sub)language analysis for practical applications (Zweigenbaum & Consortium MENELAS 1994, Bourigault & Habert 1998). With J. Bouaud and A. Nazarenko, we developed software and methods for semantic category acquisition and extension (Nazarenko et al. 2001) which turned out to be akin to Harris's approach.[2] We were influenced as well by the work done on medical language by Naomi Sager and her team.[3]

---

*We thank B. Nevin and two anonymous referees as well as A. Borillo, D. Leeman, J.-M. Marandin, and N. Sager for their remarks on a draft version of this chapter.

1. Cf. Manning & Schütze (1999: ch. 8). An important related task is *word sense disambiguation* (Ide & Véronis 1998, Manning & Schütze 1999, Jurafsky & Martin 2000, Kilgarriff & Palmer 2000).

2. Hence the name Zellig for the software (Habert & Fabre 1999).

3. We are deeply grateful to Naomi Sager for her participation on the boards for our 'Habilitations à diriger des recherches', for many (but too few) fruitful discussions and for her "satiable curtiosity", as in Kipling's "The Elephant's Child".

Our aim in this chapter is to try and appraise the Harrisian framework[4] in contextual acquisition of information categories from a natural language processing (NLP) viewpoint. In our discussion of Harris's analysis of sublanguages (Section 2), we put emphasis on the interplay between technical choices and theoretical commitments. In Section 3, we survey the current automatic methods for semantic knowledge acquisition. They can benefit from robust parsers yielding numerous but non-canonicalized dependencies. Their results provide empirical backing for the language vs sublanguage distinction. In Section 4, we address issues which, in our view, are important for the future of distributional semantics: the variation between registers, the trade-off between quality and quantity when choosing the type of context to rely upon, the need for a priori semantic knowledge, and, finally, the challenge of validating results in distributional semantics.

2. 'Semantic grammars' for sublanguages: the Harrisian framework

We present in Section 2.1 the foundational hypotheses: selection relationships, leading to meaning distinctions, are objectively investigable; they have sharp boundaries for sublanguages and 'fuzzy' ones for language. Parsing a relevant corpus and canonicalizing the underlying dependencies allows for inducing semantic categories and patterns (Section 2.2). These categories and patterns mirror the perceived world, its evolution, and its important relationships (Section 2.3).

2.1 Foundational hypotheses

Meaning not as an input but as a result. For Harris (1988:60), ". . . there is no usable classification and structure of meanings per se, such that we could assign the words of a given language to an a priori organization of meaning". He gives a striking example (ibid., p. 62): "The operator *divide* has virtually the same meaning as the operator *multiply* when its argument is a cell name: for a cell, to divide is to multiply". It is therefore not possible to rely on a priori meanings of words. Harris does not give up semantic analysis altogether.[5] On the contrary, he claims that dependency relationships between a

4. Our discussion of Harris's work relies as well on Sager's results.

5. In France, the lexicon-grammar of M. Gross (1996), a student of Harris, excluded almost all semantic analysis for thirty years.

word and its relevant operators and/or operands, are "objectively investigable and explicitly statable and subdividable" (Harris 1991:332) and lead to meaning distinctions:

> Characterizing words by their selection allows for considering the kind and degree of overlap, inclusion, and difference between words in respect to their selection sets — something which might lead to syntax-based semantic graphs (e.g. in kinship terms), and even to possibilities of decomposition (factoring) for particular sets of words. Such structurings in the set of words are possible because in most cases the selection of a word includes one or more coherent ranges of selection . . . The effect of the coherent ranges is that there is a clustering of particular operators around clusterings of particular arguments, somewhat as in the sociometric clusterings of acquaintance and status (e.g. in charting who visits who within a community). (Harris 1991:329-330)

The above quotation could seem close to Firth's position on collocation ("you shall know a word by the company it keeps", Firth 1957:11) and to the more recent research on semantic category acquisition based on context similarities (see Section 3.1). However, Harris focuses on (recursive) dependencies instead of (frequent) co-occurrence:

> . . . the structural property is not merely co-occurrence, or even frequent co-occurrence, but rather dependence of a word on a set: an operator does not appear in a sentence unless a word — one or another — of its argument set is there (or has been zeroed there). When that relation is satisfied in a word-sequence, the words constitute a sentence. (Harris 1991:332) The word classes are . . . defined by their dependence on word classes which are in turn defined by the same dependency relation. (Harris 1991:17)

Selectional regularities in language as a whole nevertheless differ from those in *sublanguages*.

Selection: likelihood in language vs sharp boundaries for sublanguages. An empirical 'method' of gathering *sublanguage* discourse is provided by Harris and by Sager:

> The sublanguages are found . . . simply by using the existing methods on bodies of data whose provenance is restricted in the real world. (Harris 1991:272)

> Informally, we can define a sublanguage as the language used by a particular community of speakers, say, those concerned with a particular subject matter or those engaged in a specialized occupation. (Sager 1986:2)

According to Sager (1986:3) and (Harris 1988:16–17, 1991:338), the selection of arguments under a word or of operators over it is a matter of likelihood in language but is nearly boolean for sublanguages. For language as a whole, ". . . in many cases there may be uncertainty as to whether a particular

operator or argument has selectional frequency for the given word or is a rarer co-occurrent which does not affect its meaning" (Harris 1991: 329- 330), and there may be ". . . disagreement among speakers, and change through time" (Harris 1988: 16–17) respecting likelihood of a word occurring as argument or operator with a given other word. When no quantitative information about collocation is available, the 'fuzzy' nature of selection makes it difficult to isolate operator-operand dependencies.[6] However, given the very restricted selection in sublanguages, it is possible to devise procedures to discover not only operator-operand dependencies, but also, in clusters of these dependencies, something like a 'semantic grammar' of the sublanguage, in the form of relevant tuples of operator- and argument-classes.

2.2 Sublanguage analysis method

Sublanguage sentences from a corpus are parsed. Underlying dependencies are canonicalized, so as to facilitate the induction of 'semantic grammars'.

Corpus selection. The results published in (Harris *et al.* 1989) rely on a selection of 25 scientific papers in immunology written between 1935 and 1966 (Ryckman 1990: 34). The hints of specialists in immunology were instrumental in the choice. Harris had a brother who was an immunologist. This brother contributed to (Harris *et al.* 1989) in providing with S. Harris a survey of the problem at hand (*The cellular source of antibody: a review*, pp. 192–215). Some of his papers were part of the corpus. Sager *et al.* (1987) analyze discharge summaries.

In both cases, the resulting corpus is small, at least by our current standards. Within the chosen domain, highly specialized texts are selected (technical or scientific data, as opposed, for instance, to popular science). A single 'genre' unifies each corpus. By *genre*, we mean a textual unit with structural, lexical, and grammatical constraints, such as, in academic life, a paper, a thesis, a technical report, a seminar, a project submission, etc. The *document type descriptions* in the SGML/XML world as well as the *document models* within current word processors instantiate some of these constraints.

6. This is however the approach followed by G. Gross (1994), who tries to discover for French 'object classes' according to their relevant sets of operators.

These further restrictions on the texts that are chosen to be 'representative' of the domain complement the empirical definition of a sublanguage by Sager or by Harris (see above).

Parsing: canonicalizing dependencies. Classes of operands are in principle obtained through corpus analysis. Sager's original work on clinical documents was done manually, through distributional analysis. Sager and her colleagues then showed that such classes could be obtained by applying automatic clustering techniques (Hirschman *et al.* 1975) to pre-parsed texts. Later work also built on existing medical resources, such as the *International Classification of Diseases* and the *Systematized Nomenclature of Medicine* (Sager 1986:6, London 1987), as a complementary source of words. There can be therefore a subtle interplay in experiments between corpus-based distributional analysis and recourse to external resources.

The medical corpus is automatically parsed, thanks to Sager's LSP (Sager 1981), whereas the immunology corpus is manually processed. Nevertheless, in order to bring the selectional regularities to light, further manual analysis is required. Transformations are applied,[7] so as to regularize the discourse: 'reversing' passivizations and nominalizations for instance.

From elementary sentence regularities to 'information formulas'. According to Sager (1986:7): "It was then straightforward for a program to substitute class names[8] for class-member occurrences in the sentence trees and to make a table of the operator-argument tuples, sorted alphabetically by operator or by the sublanguage class of each argument." The *interpretation* of the tuples yields what Sager identifies as sublanguage *kernel-types*, in that case about forty, or what Harris calls *information formulas.* A sublanguage kernel-type "is defined as a sublanguage operator class and its argument classes in terms of about a dozen sublanguage noun classes" (Sager 1986:9).

Note that the focus of the intended result is more on 'semantic grammars',[9] defined as associations between classes of operators and classes of operands, than on semantic classes of words as such.

---

7. Possibly with an a posteriori control from domain specialists (Ryckman 1990:34). Daladier (1990:82) insists on the need for experts, as acceptabilities and transformations depend on the domain.

8. Operand classes, we assume.

9. Our terminology.

2.3 Status of the acquired semantic knowledge

There is a surface contradiction in the possible status of these sublanguage semantic grammars. On the one hand, they are presented as a kind of interlingua underlying texts of a given domain in different languages, such as English, French, etc. On the other hand, sublanguage analysis reveals operator/operand evolutions mirroring notional changes in the field. The first viewpoint corresponds to a short-term span of analysis, whereas the second one is adequate for long-term drifts. A common hypothesis is used in both cases: ". . . the dependence relation of words reflects what one might consider dependencies within man's perceivable world . . . word meanings and co-occurrence selections express a categorizing of perceptions of the world" (Harris 1991: 347).

'Information formulas' as an interlingua. In (Harris *et al.* 1989), Daladier applies the same methodology to what is called in corpus linguistics a *comparable corpus*[10] for French. For Ryckman (1990: 35), the sublanguage grammar, with about 15 word categories and a dozen information formulas, is shared by French and English.

The next step is to generalize these experiments on *two* languages:[11]

> . . . for a given subscience, the reports and discussions written in one language satisfy much the same special grammar as do papers in the same field written in other languages. The structure of each science language is found to conform to the information in that science rather than to the grammar of the whole language. (Harris 1988: viii)

The resulting *formulaic language* — it "has some of the properties of mathematical notation" (Harris 1991: 4) — is not valid for a whole language. Harris makes the hypothesis that there is one such 'special grammar' for each given subscience. Each grammar "is a structural representation of the knowledge and opinions in the field" (Harris 1991: 20).

Selectional changes mirror notional evolution. On the other hand, in the long run, language is always in process of changing (Harris 1988: 92, Ryckman 1990: 37). Language change obviously affects selection:

---

10. Sets of documents, here in English and in French, following the same constraints of 'genre', domain, date, etc.

11. See also Ryckman (1990: 35, note).

> The most general factor in the varied and changing meanings of words is simply the constant though small change in likelihood — what words are chosen as operators and arguments of other words, and how frequently they are thus used. (Harris 1991: 327)

Tracking these changes is even easier for sublanguages, as ". . . the known change of information through time is seen in the change of word subclasses and sentence types in the successive articles of [the] period" (Harris 1991: 286). It is therefore possible to reveal disagreements and changes in a subfield, as illustrated for immunology in (Harris *et al.* 1989), thanks to the diachronic nature of the corpus (1935–1966).

In contrast to the synchronic analysis, whose aim was a canonical representation of formulas written in a structured notation, this diachronic viewpoint pays more attention to the social and historical dimensions of meaning, which for instance explain the necessary role of 'fuzziness': "In certain situations there is need for imprecision, when one is dealing with unsettled questions and with areas where concepts are not fixed because the operations or relations of the science are not adequately understood" (Harris 1991: 297).

As a matter of fact, in France, from the end of the 1960s to the beginning of the 1980s, Michel Pêcheux initiated a School of Discourse Analysis whose principles are close to those introduced in this section, but for language as a whole, and especially for political discourse, rather than for science sublanguages. Pêcheux's processing methods[12] are based explicitly on Harris's distributional methodology and immediate constituent analysis.[13] The sentences are manually parsed so as to get elementary units.[14] These elementary units make up a graph connected by the original relationships between them (connectors, subordination links, etc.). Pêcheux also relies on computer processing to identify similarities between units. The similarities are triggered by the sharing of relationships between units and depend on weights given to the syntactic categories of the components within each unit.[15] We must make

12. Detailed in (Pêcheux 1969) and summarized in (Maingueneau 1991: 91–105).

13. However, according to Catherine Fuchs and to Jean-Marie Marandin, who both worked with M. Pêcheux, he was not aware of the Harrisian work on sublanguages, which was not really known in France anyway at that time.

14. Pêcheux seemed to believe that some parts, such as the determiners, were too difficult to be automated anyway.

15. This computing method is close to Harris's emphasis on likelihood for language, except that the weights are on relationships between categories.

clear some important differences between Pêcheux's approach and Harris's. First, the operator/operand relationship is not central. Secondly, like Harris, Pêcheux claims that it is not possible to isolate stable semantic classes for language in general, and that any appearance of having done so is an illusion. On the contrary, relating language to ideology, Pêcheux intends to show that the semantic categories his method yields can have different meanings according to the ideological interpretation of the discourse they appear in.

3. Automatic semantic acquisition in language and sublanguages

The explosion of the quantity of on-line documents is now a commonly quoted observation. Publicly readable Web pages (more than two billion[16]) and documents managed in large intranets are the usual inputs of information processing. Words or even 'stems' are unsatisfactory indexing terms for such a volume of documents: the relevant synonyms or translations of the query components should be used in order to increase *recall*, that is, the proportion of relevant documents in the retrieved set. What is needed therefore is a 'semantic access' to documents (hence the project of a semantic Web launched by the World Wide Web consortium — http://www.w3.org).

That is the reason why, whereas previously syntax played a major role in NLP, nowadays, the emphasis of more and more research is on 'semantics', though the word refers to a wide range of 'products' and models: text classification according to domain or subdomain categories, cross-lingual information retrieval, thesaurus-based text retrieval, etc. Because of the overwhelming need to access documents through meaning, the scope of semantics is becoming surprisingly broad. However, most of this research aims at 'discovering' or at building up semantic classes or even 'ontologies', in order to tag words and documents. The interplay between these classes and the syntactic patterns they enter are less investigated. That is a major contrast with Harris's approach.

In Section 3.1, we present the current approaches to corpus-based acquisition of semantic information: non-supervised and supervised category acquisition, supervised and combined acquisition of semantic classes and of restricted patterns, as well as the underlying paradigm shifts. Robust parsers which are

16. More than two billion web pages were indexed by http://www.google.com in January 2002.

now available yield numerous but non-canonicalized dependencies for category acquisition (Section 3.2). The language/sublanguage opposition gains a certain amount of support from the observable division of labor within NLP, as well as from corpus linguistics (Section 3.3).

3.1 From knowledge representation to machine learning

Since the beginning of the 1960s until the early 1990s, NLP was mainly symbolic and oriented to artificial intelligence.[17] The systems and the underlying theories aimed to represent human knowledge, whether morphological, syntactic, semantic, or pragmatic, as precisely as possible.

Three causes of change can be identified: first, the success of probabilistic approaches in natural language engineering, such as hidden Markov models (HMM) in voice recognition, secondly the difficulties encountered by the symbolic approaches to scale up their 'toy systems', and finally the development of the Web, with its demand for language-based information retrieval from diverse and dynamically changing sources.

(Linguistic) knowledge acquisition tends to replace knowledge representation. This approach combines symbolic and numerical methods (Manning & Schütze 1999). As it is often based on machine learning techniques[18] (Mitchell 1997), it depends heavily on the availability, on the size, and on the quality of annotated linguistic resources (tagged or parsed corpora, machine readable dictionaries), as *training sets*.

Two families of machine learning techniques are being used for semantic category acquisition. In the *supervised* approach, the relevant categories are known beforehand. The aim is to assign new data to these categories, using e.g. classification algorithms. For example, to extend an existing semantic lexicon without modifying the range of semantic types, each unknown word must be linked with an existing semantic type. In the *non-supervised* approach, which does not define a priori which classes should be found, the result of the analysis, by e.g. clustering techniques, is a set (or a graph) of categories.

---

17. An influential synthesis, (Gazdar & Mellish 1989), was written in 1989 simultaneously for three languages used in developing AI systems: Lisp, Prolog, and Pop-11. A lengthy introduction to mathematical models for NLP, (Partee *et al.* 1990), does not present statistics and probabilities *at all* in 663 pages.

18. These new trends are suggested by the titles of both the paper and the workshop for (Hatzivassiloglou 1994).

Harris's method can be compared with a non-supervised machine learning process.[19]

3.1.1 *Non-supervised semantic category acquisition*

Grefenstette (1994a) distinguishes three types of semantic affinities between words and three steps in 'discovering' semantic categories:

> First-order techniques examine the local context of a word by attempting to discover what can co-occur with that word within that context. Second-order techniques derive a context for each term and compare these contexts to discover similar words or terms. Third-order techniques compare lists of similar words or terms and group them along semantic axes.

Grefenstette shows that, at each step, various approaches can be used, yielding different categorizations.

In a syntactic analysis, the attributes of a word are the words which are related to it syntactically. For instance, a noun is associated with the verb upon which it depends as an argument (as subject or object). It is also associated with the adjectives which modify it. For instance, for automatic thesaurus discovery Grefenstette (1994b) relies on the dependency relationships produced by his robust parser, Sextant. However, even within syntax-based contexts, there is still room for 'choices'. Firstly, if a *partial parser* is used, it is limited to certain types of constituents.[20] Secondly, 'transformations' can be available to a limited extent.[21]

An alternative is to use the rather crude context provided by a 'window' before and after each pivot word. One can prune the non-content words (determiners, conjunctions, adverbs) from this context. The main parameter here is the type of window. The window can correspond to a genuine text unit, whether a sentence, a paragraph or a whole text. The limit of the window

---

19. Our work belongs mainly to the non-supervised approach, but we experimented as well with supervised machine learning (Nazarenko *et al.* 2001).

20. For instance, we considered a terminology acquisition system, Lexter (Bourigault 1993), as a noun phrase extractor. No sentence parser was available to us at that time. Because our syntactic contexts were limited to nominal phrases, we were not able to rely on noun/verb co-occurrences to cluster nouns.

21. For instance, the software we developed automatically reduces the numerous and complex NPs provided by Lexter to elementary dependency trees, since they readily exhibit the fundamental binary relations between content words. However, our system does not 'undo' nominalizations or tag the relations between content words within these elementary trees.

Table 1. Varying context around the 'pivot' *developed*

Sentence	
*Patient **developed** severe respiratory distress from bilateral pneumonia, possible pulmonary infarct, and pleural effusion*	
Syntax-based context	
whole sentence, heads of constituents	{patient-Subj}{distress-Obj, infarct-Obj, effusion-Obj}
Window-based contexts	
4 words around pivot	{patient, severe, respiratory, distress, from}
– 'bag of words'	{patient}{severe, respiratory, distress, from}
+ content-words only	{patient}{severe, respiratory, distress, bilateral}
+ tagging	{patient-N}{severe-A, respiratory-A, distress-N, bilateral-A}

can also be more arbitrary: for instance Schütze (1992, 1993) clusters the words in a text within a 1,001-character window (Ide & Véronis 1998: 16), that is, about 100 words, 2 or 3 sentences. The second parameter, which is correlated to the first one, is the size of the window. A large window (Schütze's, a paragraph, even a sentence) is likely to yield topical information about the domain and its lexicon. A narrow window can correspond roughly to a syntactic context: within a span of 4 words before and after a verb, the nouns remaining after the removal of grammatical words have a heightened probability of being its subject and object. The third parameter is the order of the words within the window and with respect to the pivot: the context words can be considered as a 'bag of words' or their positions can be taken into account. The fourth parameter is the tagging and the 'stemming' of the words. The relationships between the context words and the pivot word are refined when a part-of-speech tagger has been applied and when each word form is replaced with its root.

In Table 1, we show the consequences of varying the definition of the context for the verb *developed* in a sentence from a sample discharge summary (Lyman 1987: 27).

3.1.2 *Supervised semantic category acquisition*

A priori knowledge for specifying semantic classes can be provided under several forms, such as 'seeds' vs 'full' semantic lexica, and patterns associating semantic and syntactic constraints.

Seeds. A few words can be specified as initial exemplars of given semantic classes and used as 'seeds' from which the collection of distributionally similar words is bootstrapped. For instance, Roark & Charniak (1998) choose seed words among the most frequent ones in the corpus: e.g., *murder*, *crime*, *killing*, *trafficking*, *kidnapping* for the *Crimes* class. They collect the distribution of words in a few selected syntactic constructs (conjunctions, lists, and appositions) and determine which words co-occur most significantly with the seed words. Following Riloff & Shepherd (1997), by iteratively considering these as additional seed words, they incrementally build classes of semantically close words, which they rank for category membership. For the *Crimes* class above, the top-ranked words are *terrorism*, *extortion*, *robbery*, *assassination*, *arrest*, *disappearance*, *violation*, *assault*.

'Full' semantic lexica. Semantic resources, whether general or specialized, nowadays exist in abundance, at least for English, often at no cost for research purposes. They provide many examples of various semantic classes. Whatever their size, they must be adapted and completed for more specific tasks and domains. Many methods have been proposed for supervised categorization of unknown words into existing classes, such as (Nazarenko *et al.* 2001). We mention here some well-known examples of large semantic resources.

The ancestor of general semantic networks is Roget's *Thesaurus* (1852). An up to date electronic version is available (http://mthwww.uwc.edu/wwwmahes/files/thesaur.htm). In 1985, Miller's team in psycholinguistics at Princeton University designed an electronic semantic network, WordNet (Fellbaum 1998). Today, it comprises more than 168,000 words or idioms, with 345,000 semantic relationships between them. Hyperonymy (classifier or *is-a* hierarchy) and synonymy play a major role in it. Similar semantic networks have been devised for the major European languages, within the project EuroWordNet (Vossen 1998), with links between languages to facilitate multilingual access to documents.

Terminologies and thesauri have been compiled in many domains,[22] such as medicine (Cimino 1996), agriculture, aeronautics, etc.[23] In medicine, the US

22. With the advent of the Web, it has become easier to identify the domains for which terminologies exist, and often these terminologies are also now easier to obtain. Some terminology bodies maintain web pages with links to many such resources, e.g., Termisti at http://www.termisti.refer.org/ or Infoterm at http://www.infoterm.org/.

23. See for instance http://clicnet.swarthmore.edu/dictionnaires.html.

National Library of Medicine (NLM) launched in 1989 a project to collect and link medical terminologies: the Unified Medical Language System (UMLS, Lindberg *et al.* 1993). The UMLS for the year 2002 contains over 60 biomedical terminologies, resulting in about 800,000 'concepts' and 2.1 million terms. Each concept is tagged with one or more of 134 semantic types organized in a semantic network. Its companion Specialist Lexicon includes over 163,000 lexical entries. Lexical tools for inflection and derivation are also provided.

3.1.3 *Patterns associating semantic and syntactic constraints*

Because words of a given semantic class occur in restricted contexts, some initial knowledge in the form of 'lexico-syntactic templates' can guide the collection of words that belong to a common semantic class or which obey the same semantic relationship, such as hyperonymy (Hearst 1992, Borillo 1996). Conversely, the distribution of a set of words with the same semantic class helps to detect sublanguage patterns.

This approach has gained strength rather recently in NLP since the development of information extraction (IE). In the IE task, only specific items of information are to be spotted and extracted from each text of a collection. For instance, in newswire dispatches about joint ventures (MUC6 1996), the required items are the names of the original companies and that of the new one, the place where it is based, etc. One possible method for collecting this information is to design a limited grammar subset where only the patterns that deal with these information items are present. The rationale is that if such information items are sparse enough in the text, it is more economical and still relevant to concentrate on the more restricted constructs that only cover these items than to develop a grammar for the whole texts.

Methods have therefore been designed to acquire targeted patterns instead of a full grammar. For instance, the system designed by Riloff & Jones (1999) learns both how to populate semantic classes and what the authors call 'extraction patterns' from a corpus, given a parser and a few initial 'seed' words for each semantic class (e.g., *bolivia, city, colombia, district, guatemala, honduras, neighborhood, nicaragua, region, town* for the class *Terrorism Location* in the MUC-4 corpus of terrorism news articles). Note that words are decapitalized. This system relies on 'mutual bootstrapping': "extraction patterns can generate new examples of a semantic category which in turn can be used to identify new extraction patterns." The seed words serve to initialize the process. With those above, the top-ranked patterns provided by the parser for the class *Terrorism location* are the following: *living in* <*x*>, *traveled to* <*x*>,

become in <x>, *sought in <x>*, *presidents of <x>*, *parts of <x>*; the precision of word categorization (the ratio of correctly categorized words to the total number of categorized words) ranges from .63 to .67 depending on the number of iterations.

Many authors have built on the interplay of these two facets of selection to induce both semantically related words and specific patterns (Morin 1998, Poibeau 2000). An advantage is that the process can be iterated, capitalizing on the previously acquired patterns to collect more class members which in turn will help identify supplementary patterns.

3.1.4 *Conclusions*

Unsupervised learning of word classes according to distributional similarities provides clusters of words that *require* a posteriori human 'pruning', interpretation, and labeling. Some clusters are spurious artefacts and must be removed, as must some words which are clearly 'outsiders' in otherwise consistent clusters. Clusters can contain words that are related by various semantic relations, such as meronymy (*racine*/*pommier*: *root*/*apple tree*) or process/entity (*enracinement*/*système racinaire*: *rooting*/*root system*). Even with hierarchical clustering, the resulting clusters are coarse-grained as compared with the ones described by Harris or by Sager.

The same conclusion holds for supervised acquisition: the result cannot be used 'as is'. For instance, as one can judge from the example in (Roark & Charniak 1998) given above, most of the acquired words are intuitively good instances of crimes, but some of the words (e.g., *arrest(s), disappearance(s)*) are probably not valid members of the class (although *arrest* is strongly linked to *crime*, an arrest is rarely a crime, in very specific contexts). For that reason, a common practice is to integrate human input in the end or at each round of iteration to control the acquisition spiral (Morin 1998). As it is difficult to train a classifier for too many categories at once, supervised acquisition seems as well to work better with a limited number of semantic classes: five for instance in (Riloff & Shepherd 1997) or (Roark & Charniak 1998).

Even with the interplay between patterns and semantic classes, the restricted goal-driven approach to grammar in information extraction does not yield sublanguage grammars, as it is severely restricted to a very limited range of 'semantic relationships', which cover a small portion of the corpus. What's more, the term 'pattern acquisition' is a bit misleading: so far, there is very little generalization from these patterns (Soderland 1999). For instance, several thousand patterns are acquired in (Riloff & Jones 1999). On the contrary, the

view of sublanguage grammar that is construed in Harris's work is that of a grammar with complete coverage of every sublanguage sentence — and also a grammar of texts.

3.2 Current robust parsers offer numerous non-canonicalized dependencies

Current robust parsers do yield a large quantity of dependency information from which semantic acquisition can proceed. Moreover they perform limited transformations which could be used to increase operator–operand associations.

3.2.1 *Numerous but partial dependency relationships*

While the 1980s saw the creation or extension of powerful, formal linguistic theories for grammar description, partial but more pragmatic approaches have been advocated in the late 1980s and in the 1990s.

The former class of grammatical theories include Lexical-Functional Grammar (Bresnan 1982a), Generalized Phrase Structure Grammar (Gazdar *et al.* 1985), Head-driven Phrase Structure Grammar (Pollard & Sag 1987) and Tree Adjoining Grammar (Joshi 1987). They place a great emphasis on predicate–argument relationships, so that constructions such as passivization, control, or raising can be handled precisely. They rely on large, complex grammars and on detailed lexical information to drive syntactic analysis. The development of a large-coverage lexicon and grammar for a language in such formalisms is consequently a knowledge-intensive task (e.g., Butt *et al.* 1999). Furthermore, parsing complexity for these formalisms is at best cubic (GPSG), if not in $\mathcal{O}(n^6)$ (TAG), with respect to sentence length.[24] This prohibits practical application of such systems to unrestricted, large corpora, which may have sentences as long as 100 words, because the parser can fail or produce too many analyses.

Therefore, computationally less complex formalisms have been advocated and designed. They are geared towards more robust processing of running text (Vergne 2000). Instead of systematic recursive processing, the general trend has been to focus first on a deterministic parse of local structures that can reliably be determined without ambiguity. On the basis of these 'chunks' (Abney 1991)

24. See e.g., Joshi *et al.* (1991) for a discussion of a complexity class that is interesting for natural languages.

or 'non-recursive syntagms' (Vergne 2000), larger structures are then identified, using a limited number of specialized iterations rather than a generalized recursive process. Grammatical dependencies between structures are also computed. This sequence of steps can be implemented using a cascade of finite-state transducers (Roche & Schabes 1997) or dependency-oriented non-recursive rules (e.g., Grefenstette 1994b, Vergne 2000). Another robust approach, constraint grammars, uses a set of constraints on the distribution of morphosyntactic categories, and includes a shallow dependency-oriented functional description (Voutilainen & Heikkilä 1993). More recent research has strengthened this dependency representation (Tapanainen & Järvinen 1997).

Because of their nearly linear complexity, these pragmatic methods can be applied to huge and 'noisy' corpora. These approaches are indeed inherently robust: they can produce useful grammatical representations even when sentences are long or possibly ill-formed. The tradeoff for this robustness is partial, shallow analysis: the grammatical representations produced may not wholly cover each input sentence, or may not be as detailed as one might wish. The small loss of input clues because of partial parsing may only slightly decrease the recall of these tools, and still disclose sufficient dependencies for distributional analysis.

3.2.2 *Little progress in automatic canonicalizing of dependencies*

Harris's work makes extensive use of *transformations* to reduce utterances to minimal operator-argument constructions. Such transformations were performed manually in his work, as well as in early computer-aided experiments in acquisition of semantic categories (Hirschman *et al.* 1975). The current situation is roughly the same. Even though specific rewriting devices are now sometimes available, this is still usually a manual step.

Some of these transformations are addressed by the above-mentioned grammatical theories. For instance, passivization is handled by LFG (Bresnan 1982b) in such a way that the structure that links the verb predicate (operator) to its subject and object arguments is the same for active and passive constructions — in other words, the passive transformation can be seen as being undone in that representation. Relations across grammatical categories, such as verb nominalizations, seem to be less integrated in the core theories. The reason may have been insufficient interest in morphology. Another reason may have been unavailability of large, detailed lexical resources, but they are available today, and actual parsers relying on these theories still do not implement such relations.

Generally partial parsers do not include devices for handling such transformations either. For instance, the GREYC parser (Vergne 2000) does not explicitly mark the subjects of passive verbs as having a particular status: assigning the surface subject to the second argument of the predicate is left to subsequent processing external to the parser.

However recent corpus-based work has partially tackled this problem. For instance, Jacquemin & Tzoukermann (1999) identify morpho-syntactic variants of controlled vocabulary terms through the application of precise syntactic rules. They show the following examples, where the second expression of each line has been identified to be a variant of the first:

échange d'ion (ion exchange) / échange ionique (ionic exchange)
lessivage des sols (soil leaching) / sol lessivé (leached soil)
clonage des gènes (gene cloning) / cloner les gènes (to clone genes)

Evaluated on a thesaurus and corpus in the agriculture domain, their method reaches 86% precision and 76% recall for morpho-syntactic variants. Royauté (1999) studies the relation between verb subcategorization frames and that of the corresponding nominalizations. Relying on Jacquemin & Tzoukermann (1999), he automatically extracts such verb-noun couples from a corpus then uses these to align, for each lexical item, verbal and nominal subcategorization frames (with precision 79–91%), thus relating transformed structures.

3.3 Empirical backing for the language-sublanguage distinction

The distinction between language and sublanguage is recurrent in Harris's work. The recent tidal wave of resource-based NLP and the 'rebirth' of corpus linguistics provide indirect empirical backing for this distinction, although they do not always add more precise insights about it.

3.3.1 *Specialized and reference corpora*

The renewed interest in corpus linguistics comes from the merging of two lines of research: nearly four decades of 'unassuming' descriptive linguistics, and about one decade of resource-based language engineering trying to scale up systems and models.[25]

25. As far as language engineering is concerned, the year 1993 seems to represent the 'official' beginning of the 'corpus era'. Cf. Jurafsky & Martin (2000:10–14) for a more precise history. For instance the Association for Computational Linguistics that year

Reference corpora are easily available. They fulfill the aim of providing a set of textual data whose production and reception conditions are precisely defined and which are representative of a great variety of communication situations. There are now 'mega-corpora', such as the *British National Corpus*:[26] 100 million tagged words (about 1,000 medium-size novels), comprising 10 million words of transcribed spoken English as well as written language (fiction from 1960 onwards and 'informative' texts from 1975 onwards).[27]

These balanced 'samples' make it possible to test on a wide range of discourses Harris's findings about collocation and selectional preferences and about distinction between language and sublanguages, or to study the selection of arguments under a word or of operators over it according to their strata (spoken/written, topic, etc.).[28]

On the other hand, within a particular community of speakers, the nature of selection can be precisely determined, as corpora are easily gathered and processed (electronic versions of newspapers, theses and scientific papers, Web sites, etc.).

Significant syntactic variations are evidenced between domains and corroborate to a certain extent Harris's proposals about sublanguages. Biber (1993: 223) uses reference corpora divided into different domains to show that the probability of occurrence of a morpho-syntactic category is a function of the domain. Biber also shows (ibid., p. 225) how the sequence of probabilities of morphosyntactic categories (bigrams as in HMM) varies across domains. Similarly, collocation is shown to differ from one domain to another (for instance for *sure* and *certain*), which should correspond to a variation in selection.

Sekine (1997) gives other indications of the syntactic homogeneity of a text 'genre'. He uses 8 genres of the Brown corpus (documentaries, editorials,

published two special issues of its journal on the topic and has since then been holding an annual workshop on very large corpora.

26. http://info.ox.ac.uk/bnc/

27. However the huge size of current corpora does not give access to language as a whole: we have no way so far to eliminate *bias error*. Bias error occurs when one or several characteristics of a sample are systematically different from those of the population that the given sample is supposed to represent. As a matter of fact, we cannot so far evaluate precisely the relationship between the huge 'document bases' which are available and the set of all discourses.

28. See Manning & Schütze (1999: ch. 2 & 8).

hobbies, learned, fiction, western, romance, novels). He examines the performance of a probabilistic parser when it is trained on the same genre as is used for the test, when it is trained on all genres, when it is trained on the fiction 'class' (fiction, western, romance novels) or on the non-fiction 'class' (documentaries, editorials, hobbies, learned). The different performances are in decreasing order: training corpus genre equals test corpus genre, training corpus and test corpus genres belong to the same class, training corpus and test corpus are composed of extracts from all genres. Training the parser on one class (fiction, for instance) and using it on the other class (non-fiction) yields the worst results.

3.3.2 *Division of labor within NLP*

We consider here a partition of NLP work which parallels Harris's distinction between language and sublanguage, namely, the processing of unrestricted texts vs. texts of circumscribed domains and genres.

One task that unarguably addresses unrestricted text is information retrieval (IR) on large and heterogeneous collections such as the Web (Baeza-Yates & Ribeiro-Neto 1999). Companion tasks of IR, such as query expansion, thematic segmentation, text summarization, and text categorization, also apply to the same kinds of documents as IR. In contrast, NLP tasks such as information extraction (IE) or knowledge acquisition from texts, e.g. term acquisition (Jacquemin & Bourigault 2002), attack homogeneous collections of texts. In parallel, one can distinguish two classes of corpus-based acquisition of semantic knowledge: (i) learning word associations, and (ii) learning more precise linguistic or knowledge structures. Word associations are useful for identifying words that are semantically linked to each other. These associations show some degree of flexibility: there is a gradation from unrelated words to strongly related words. Associated words can provide the basis for, e.g., query expansion or text summarization. Most of the corresponding tests have been conducted on news articles with a variety of topics (finance, computers, terrorism, etc.).

Learning to disambiguate syntactic attachments in a corpus by identifying which attachments occur in non-ambiguous configurations is a fruitful method in robust parsing (Bourigault 1993, Grefenstette 1994b). Learning more precise semantic structures, typically involving selectional restrictions, is the aim of much research around information extraction: porting an IE system to a new domain means acquiring again local patterns or semantic grammars for identifying specific information items. Among these works are Riloff & Jones

(1999), Roark & Charniak (1998), presented above. All the systems in the latter stream of work have been applied to selected text collections. That is, they seem to have been confined to specialized languages. In contrast, the systems in the former stream address more diverse texts, for instance newspaper articles, which can be considered closer to 'general language'. But what they obtain from these texts is less specified association relations between words. What we observe in this picture is congruent with Harris's observation that selectional restrictions are stronger in sublanguages and weaker in language as a whole (Harris 1991:338): it may be that the NLP community has arrived at the distinction between language and sublanguage by self-organization.

4. Proposals for automatic distributional semantics

We are certainly not in a position to stand back from the drift of semantic acquisition research and to provide guidance about it. We can only state the conclusions we draw from our own experiments in supervised and non-supervised category acquisition. First, register variation is a factor which must be thoroughly controlled (Section 4.1). Secondly 'windows' of words are often a sufficient estimate of syntactic contexts (Section 4.2). Thirdly, we do not believe that semantic acquisition 'from scratch' is really an option (Section 4.3). Lastly, semantic acquisition techniques and methods should be evaluated in the same ways that other NLP areas are (Section 4.4).

4.1 Controlling register variation within a domain

The insistence on sublanguage as a coherent system can lead to overlooking existing variation within a domain which can be related to the presence of various registers.[29] The existence of these registers, with their different styles, may produce heterogeneity in an apparently homogeneous domain. For instance, the precision of the taggers for French which have been evaluated within the framework of GRACE (Adda *et al.* 1998), measured in relation to the manually tagged reference corpus, shows significant variations depending on the part of the corpus which is under examination (Illouz 1999). This 100,000-word corpus comprises extracts from the newspaper *Le Monde*

29. But see Kittredge (1982) and Zwicky & Zwicky (1982) for a contrary view.

(2 extracts) and from literary texts: memoirs (2 extracts), novels (6 extracts), and essays (2 extracts). Thus there are important positive and negative variations among the taggers for an extract from memoirs, with respect to extracts from other registers. (The registers in the literary domain are memoirs, novels, and essays). In other words, the presence of memoirs in the literary domain is a source of heterogeneity.

However, even though a native speaker (and writer) has some intuitions about registers and styles, (s)he is often at a loss when it comes to giving precise features for them. We lack a meta-language for such correlations, as evidenced by the rather ad hoc way in which socially recognized 'genres' (essay, review, scientific paper, etc.) are transmitted more than taught. (We may be said to have mastered a 'genre' when we are able to do a pastiche of it). Text types seem to derive from a correlation of morphological and syntactic features. Although this has so far resisted formalization, a multi-dimensional statistical approach, e.g. a correlation of text types with registers and 'genres' (Biber 1995), can provide a 'handle' on this phenomenon. Biber uses 67 features corresponding to 16 different categories (verb tense and aspect markers, interrogatives, passives, etc.). He examines their distribution in the first 1,000 words of 4,814 contemporary English texts from reference corpora. The identification of the 67 features in the corpus is done automatically on the basis of a preliminary morpho-syntactic tagging but is manually checked. The sets of correlated features (the 'dimensions') are obtained through a multivariate statistical technique (factor analysis). Each dimension consists of 2 complementary groups of features which can be interpreted as positive and negative poles. When one group of features occurs in a text, the other group is far less frequent or is avoided. Any new text can then be placed in the space of *n* dimensions previously identified. The location of the new text along each dimension in this space is determined by the frequency count in it of each of the features associated with that dimension. Clustering methods are then used to group texts in terms of their locations in this space. The resulting clusters are 'types of texts' which correspond directly neither to text 'genres' nor to language styles or registers.

These experiments show the need to become well acquainted with the registers which are used in a domain. It is likely that within a domain, some registers are more suited than others to the acquisition of information categories. For instance, in the rather specialized domain of coronary diseases, we found that discharge summaries and handbooks were more favorable for semantic category acquisition than letters between doctors about their pa-

tients, even though the lexicon in these letters was highly specialized as well. In other words, for purposes of contextual acquisition of information categories, Harris's hypotheses about the sublanguage for a domain must be restricted to a subpart of a domain, some registers only. In a way, this control of register variation parallels and specifies the selection of relevant sentences within the corpus by Harris.

4.2 Syntactic contexts are not always more informative than 'windows' of words

We illustrated in Section 3.1.1 the alternative between syntactic analysis and windows of words for the acquisition of semantic information. To compare the two techniques on nouns, Grefenstette used Roget's thesaurus as a benchmark (*gold standard*), "checking whether the 'similar words' discovered by each technique are placed under the same heading in the thesaurus" (Grefenstette 1996: 206). His method is the following:

> Given a corpus, use the similarity extraction method to derive similarity judgements between the words appearing in the corpus. For each word, take the word appearing as most similar. Examine the human compiled thesaurus to see if that pair of words appears under the same topic number. If it does, count this as a hit.

The corpus gathered for the experiment was made from sentences containing *Harvard* or one of the 30 hyponyms found under the word *institution* in WordNet — *institution*, *establishment*, *charity*, *religion*, etc. — from *Grolier's Encyclopaedia*: 3.9 megabytes of text, that is, more than 400,000 words, or four medium-size novels.

For the syntactic approach, a noun's context becomes all the adjectives, nouns, and verbs that enter into syntactic relations with it. For the window-based approach, all nouns, adjectives, or verbs on either side of the pivot noun within ten words and within the same sentence represent the context. Once the context is extracted for each noun, the contexts of every pair of nouns are compared for similarity using a weighted Jaccard measure.[30]

---

30. The non-weighted Jaccard measure between two words follows the equation $\frac{a}{a+b+c}$, where a is the number of shared contexts, b the number of contexts used only by the first word, and c the number of contexts used only by the second word (Losee 1998: 43–62). Weighting this measure amounts in this case to taking into account the number of occurrences of each context.

According to Grefenstette, the results depend on the range of frequencies which is considered:

> . . . the performance of the partial syntactic analysis based technique is better for the 600 most frequently appearing nouns, which may be considered as the characteristic vocabulary of the corpus.
>
> . . . the window co-occurrence techniques give more hits for less frequently occurring words, after the 600th most frequent word.

The reason for this variation of accuracy can be explained by the number of attributes associated with each technique. The syntactic approach prunes the context dramatically, leaving enough to cluster frequent words, but less frequent words need a larger context to get enough 'attributes' to be compared with other words.

In the light of this experiment and of the literature, we believe that a narrow window around pivot words often seems to constitute a sufficient estimate of syntactic contexts in order to discover semantic categories. It is not always necessary to resort to parsing. For that task as well, it is better to let well enough alone.

4.3 The need for a priori semantic knowledge

For studying language, Harris assumes no prior knowledge apart from informant judgments. This is linked to the methodological motivation of the absence of an independent metalanguage for describing language. The study of a sublanguage, though, may rely on external knowledge: that of the language. It should not however assume the prior availability of specific knowledge about the sublanguage: such knowledge is precisely to be obtained through distributional analysis.

When we consider the recent automated methods that aim at producing semantic categories or semantic patterns from a corpus (see Section 3.1), we observe that all are provided with some linguistic knowledge to begin with. A general-purpose syntactic parser is used in most of these methods. As an embodiment of grammatical knowledge about the language, this is in line with the Harrisian methodology. A general semantic lexicon, such as WordNet, also pertains to language as a whole. 'Pure' distributional analysis of a sublanguage may therefore rely on such knowledge sources, i.e., follow a non-supervised acquisition method. However, syntactic or semantic knowledge specifically for the sublanguage domain cannot be provided within this paradigm.

As a matter of fact, none of the non-supervised experiments we are aware of has succeeded in automatically producing useful semantic categories from scratch, that is, in fully automatizing the method of Hirschman *et al.* (1975). The more successful experiments provide their systems with selected seed words or more sizable domain lexica, or with specific lexico-syntactic templates: they assume prior semantic or syntactic knowledge about the domain or sublanguage. This a priori knowledge seems to be necessary, at least in the current state of technology, to run semantic distributional methods.

4.4 A challenge: validating distributional semantics results

For a long time, NLP modules were research prototypes. More and more of them are now industrial products (ViaVoice, DragonDictate, etc.) or are freely available software components (such as Brill's (1995) tagger). The actual use of NLP treatments for very large corpora and real-life tasks as well as the availability of several approaches and 'products' for a given task led to the need for evaluation techniques (Spärck Jones & Galliers 1996).

For nearly every type of task (tagging, parsing, word sense disambiguation, anaphora resolution, information extraction, etc.), evaluation initiatives are organized along similar lines. First, an annotated corpus is provided to the participants in order to adapt (train or tune) their systems. For instance, in word sense disambiguation (Kilgarriff & Palmer 2000), the words which need disambiguating are manually tagged with the meaning that is relevant in context. Secondly, an unannotated corpus is 'tagged' by every participant. It is then possible to compare their results with the reference version of this second corpus, which is provided as a benchmark. This is often a 'black box' evaluation: instead of basing an explanation upon the internal characteristics of the respective systems (as in a 'glass box' evaluation), metrics are provided for comparison with the benchmark, such as *recall* (proportion of the overall correct answers which was returned by the system) and *precision* (proportion of correct answers among the given answers).

Contextual acquisition of information categories should be submitted to such evaluations. For instance, an existing terminology can be used as a touchstone to evaluate the quality of acquired contextual information categories for a given sublanguage. We followed that approach in (Bouaud *et al.* 2000). A general semantic network can have the same role, as in (Grefenstette 1996) — see Section 4.2. It is however more difficult to devise evaluation criteria for non-supervised category acquisition.

The research done in (Harris *et al.* 1989) on immunology sublanguage evolution concurs with more recent work. For instance, Bourigault & Slodzian (2000) emphasize that terminologies are normalized snapshots of the concepts and terms of a domain for a given activity: they must keep track of changes as new texts are produced in these domains. This sheds some doubt on the mere possibility of reaching stability in the information formulas of a sublanguage. The actual status of general semantic networks such as WordNet is probably dubious as well: like any other dictionary, WordNet creates 'spurious ambiguities' when it mixes meanings that are rooted in very different language uses. It may be the case that the relationships and categories they offer are an artefact which can nevertheless help in discovering sublanguage structure.

Bourigault & Slodzian also insist that using terminologies for any purpose other than that for which they were designed, if possible at all, generally requires non-trivial adaptation. This means that the above-described evaluation paradigm does not transpose directly to non-supervised semantic category acquisition, short of a suitable gold standard: distributional semantic results require the emergence of a new perspective for their evaluation.

References

Abney, Steven P. 1991. "Parsing by chunks". *Principle-Based Parsing: Computation and Psycholinguistics* ed. by Robert C. Berwick, Steven P. Abney & Carol Tenny, 257–278. Dordrecht & Boston: Kluwer Academic Publishers.

Adda, Gilles, Josette Lecomte, Joseph Mariani, Patrick Paroubek, & Martin Rajman. 1998. "The GRACE French part-of-speech tagging evaluation task". *First International Conference on Language Resources and Evaluation*, (LREC), Granada, ed. by Antonio Rubio, Navidad Gallardo, Rosa Castro & Antonio Tejada, ELRA vol. 1, 433–441.

Baeza-Yates, Ricardo &Berthier Ribeiro-Neto. 1999. *Modern Information Retrieval.* New-York: Addison-Wesley.

Biber, Douglas. 1993. "Using register-diversified corpora for general language studies". *Computational Linguistics* 19:2. 243–258.

Biber, Douglas. 1995. *Dimensions of register variation: A cross-linguistic comparison.* Cambridge: Cambridge University Press.

Borillo, Andrée. 1996. "Exploration automatisée de textes de spécialité: repérage et identification de la relation lexicale d'hyperonymie". *LINX* 34–35. 113–124.

Bouaud, Jacques, Benoît Habert, Adeline Nazarenko & Pierre Zweigenbaum. 2000. "Regroupements issus de dépendances syntaxiques sur un corpus de spécialité: catégorisation et confrontation à deux conceptualisations du domaine". *Ingénierie des connaissances: évolutions récentes et nouveaux défis* ed. by Jean Charlet, Manuel Zacklad, Gilles Kassel, & Didier Bourigault, chap. 17, 275–290, Paris: Eyrolles.

Bourigault, Didier & Benoît Habert. 1998. "Evaluation of terminology extractors: principles and experiments". *First International Conference on Language Resources and Evaluation* (LREC), Granada, ed. by Antonio Rubio, Navidad Gallardo, Rosa Castro & Antonio Tejada, ELRA vol. 1, 299–305.

Bourigault, Didier & Monique Slodzian. 2000. "Pour une terminologie textuelle". *Terminologies Nouvelles* 19.

Bourigault, Didier. 1993. "An endogeneous corpus-based method for structural noun phrase disambiguation". *Proceedings of the 6th Conference of the European Chapter of the Association for Computational Linguistics (EACL'93)*, Utrecht, 81–86.

Bresnan, Joan, ed. 1982a. *The Mental Representation of Grammatical Relations*. Cambridge, MA: MIT Press.

Bresnan, Joan. 1982b. "The passive in lexical theory". Bresnan 1982a, chap. 1, 3–86.

Brill, Eric. 1995. "Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging". *Computational Linguistics* 21.4: 543–565.

Butt, Miriam, Stefanie Dipper, Anette Frank, & Tracy Holloway King. 1999. "Writing large-scale parallel grammars for English, French, and German". *Proceedings of the LFG99 Conference*, Stanford, Calif. ed. by Miriam Butt & Tracy Holloway King.

Cimino, James J. 1996. "Coding systems in health care". *Yearbook of Medical Informatics '95 — The Computer-based Patient Record* ed. by Jan H. van Bemmel & Alexa T. McCray, 71–85. Stuttgart: Schattauer.

Daladier, Anne. 1990. "Aspects constructifs des grammaires de Harris". *Langages* 99. 57–84. Ed. by A. Daladier.

Fellbaum, Christiane, ed. 1998. *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.

Firth, John Rupert. 1957. *Papers in Linguistics, 1934–1951*. London: Oxford University Press.

Gazdar, Gerald & Chris Mellish. 1989. *Natural Language Processing in Lisp*. Reading, MA: Addison Wesley.

Gazdar, Gerald, Geoffrey Pullum Ewan Klein, & Ivan Sag. 1985. *Generalised Phrase Structure Grammar*. Oxford: Blackwell.

Grefenstette, Gregory. 1994a. "Corpus-derived first, second and third order affinities", *EURALEX*, Amsterdam.

Grefenstette, Gregory. 1994b. *Explorations in Automatic Thesaurus Discovery*. Dordrecht & Boston: Kluwer Academic Publishers.

Grefenstette, Gregory. 1996. "Evaluation techniques for automatic semantic extraction: Comparing syntactic and window based approaches". *Corpus Processing for Lexical Acquisition* ed. by Branimir Boguraev & James Pustejovsky, chap. 11, 205–216. Cambridge, MA: MIT Press.

Grishman, Ralph. 1997. *Proceedings of the Fifth Conference on Applied Natural Language Processing*, ACL, Washington, DC.

Gross, Gaston. 1994. "Classes d'objets et description des verbes". *Langages* 115: 15–30.

Gross, Maurice. 1996. "Lexicon grammar". *Concise Encyclopedia of Syntactic Theories* ed. by Keith Brown & Jim Miller, 244–258. Cambridge: Pergamon.

Habert, Benoît & Cécile Fabre. 1999. "Elementary dependency trees for identifying corpus-specific semantic classes". *Computers and the Humanities* 33.3: 207–219.

Harris, Zellig S. 1988. *Language and Information.* New York: Columbia University Press.

Harris, Zellig S. 1991. *A Theory of Language and Information. A mathematical approach.* Oxford: Oxford University Press.

Harris, Zellig S., Michael Gottfried, Thomas Ryckman, Paul Mattick Jr., Anne Daladier, T.N. Harris, & S. Harris. 1989. *The Form of Information in Science: Analysis of an immunology sublanguage.* Boston Studies in the Philosophy of Science 104. Dordrecht & Boston: Kluwer Academic Publishers.

Hatzivassiloglou, Vasileios. 1994. "Do we need linguistics when we have statistics? a comparative analysis of the contributions of linguistic cues to a statistical word grouping system". *The Balancing Act: Combining symbolic and statistical approaches to language*, ed. by Judith L. Klavans & Philip Resnik, Cambridge, Mass.: MIT Press, pp. 67–94.

Hearst, Marti A. 1992. "Automatic acquisition of hyponyms from large text corpora". *Proceedings of the 14* th *COLING* ed. by Antonio Zampolli, 539–545. Nantes, France.

Hirschman, Lynette, Ralph Grishman & Naomi Sager. 1975. "Grammatically based automatic word class formation". *Information Processing & Management* 11. 39–57.

Ide, Nancy & Jean Véronis. 1998. "Introduction to the special issue on word sense disambiguation: the state of the art". *Computational Linguistics* 24.1: 1–40.

Illouz, Gabriel. 1999. "Méta-étiqueteur adaptatif: vers une utilisation pragmatique des ressources linguistiques". *Proceedings of TALN 1999 (Traitement automatique des langues naturelles)* ed. by Pascal Amsili, 185–194. ATALA. Cargèse.

Jacquemin, Christian & Didier Bourigault. 2002. "Term extraction and automatic indexing". *The Oxford Handbook of Computational Linguistics* ed. by Ruslan Mitkov, Oxford: Oxford University Press.

Jacquemin, Christian & Évelyne Tzoukermann. 1999. "NLP for term variant extraction: A synergy of morphology, lexicon, and syntax". *Natural language information retrieval* ed. by Tomek Strzalkowski, chap. 2, 25–74. Dordrecht & Boston: Kluwer Academic Publishers.

Joshi, Aravind K., K. Vijay-Shanker, & David J.Weir. 1991. "The convergence of mildly context-sensitive grammar formalisms". In *Foundational issues in Natural Language Processing* ed. by Peter Sells, Stuart Shieber & Thomas Wasow, 31–81. Cambridge, MA: MIT Press.

Joshi, Aravind K. 1987. "An introduction to Tree Adjoining Grammars". In *Mathematics of Language* ed. by Alexis Manaster-Ramer, 87–115. Amsterdam: John Benjamins.

Jurafsky, Daniel & James H. Martin. 2000. *Speech and language Processing: An introduction to natural language processing, computational linguistics and speech recognition.* Upper Saddle River, NJ: Prentice Hall.

Kilgarriff, Adam & Martha Palmer. 2000. "Special issue on SENSEVAL: Evaluating word sense disambiguation programs". *Computers and the Humanities* 34: 1–2.

Kittredge, Richard. 1982. "Variation and homogeneity of sublanguage". In *Sublanguage: Studies of language in restricted domains* ed. by Richard Kittredge & John Lehrberger, chap. 4, 107–137. New York: Walter de Gruyter.

Lindberg, Don A B, Betsy L Humphreys & Alexa T. McCray. 1993. "The Unified Medical Language System". *Methods of Information in Medicine* 32.2: 81–91.

London, Joyce. 1987. "The healthcare lexicon". Sager *et al.* 1987, chap. 6, 137–144.

Losee, Robert M. 1998. *Text Retrieval and Filtering: Analytic models of performance*. Information Retrieval 3. Dordrecht & Boston: Kluwer Academic Publishers.

Lyman, Margaret S. 1987. "Medical applications of computer-processed narrative". In Sager *et al.* 1987, chap. 2, 23–57.

Maingueneau, Dominique. 1991. *L'analyse du Discours: Introduction aux lectures de l'archive*. Paris: Hachette.

Manning, Christopher D. & Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.

Mitchell, Tom M. 1997. *Machine Learning*. New York: McGraw-Hill.

Morin, Emmanuel. 1998. "Prométhée: un outil d'aide à l'acquisition de relations sémantiques entre termes". *Proceedings of TALN 1998 (Traitement automatique des langues naturelles)* ed. by Pierre Zweigenbaum, 172–181. ATALA. Paris. MUC6. 1996.

MUC-6: Proceedings of the Sixth Message Understanding Conference, Columbia, MD, Morgan Kaufmann. November 1995.

Nazarenko, Adeline, Pierre Zweigenbaum, Benoît Habert, & Jacques Bouaud. 2001. "Corpus-based extension of a terminological semantic lexicon". *Recent Advances in Computational Terminology* ed. by Didier Bourigault, Christian Jacquemin & Marie-Claude L'Homme, 327–351. Amsterdam: John Benjamins.

Partee, Barbara H., Alice Ter Meulen, & Robert E. Wall. 1990. *Mathematical Models in Linguistics*. Dordrecht & Boston: Kluwer Academic Publishers.

Pêcheux, Michel. 1969. *Analyse Automatique du Discours*. Paris: Dunod.

Poibeau, Thierry. 2000. "De l'acquisition de classes lexicales à l'induction semi-automatique de grammaires locales". *Proceedings of TALN 2000 (Traitement automatique des langues naturelles)* ed. by Eric Wehrli & Martin Rajman, 307–316. ATALA. Lausanne.

Pollard, Carl & Ivan A. Sag. 1987. *Information-Based Syntax and Semantics*. Lecture Notes 13. Stanford, CA: CSLI.

Riloff, Ellen & Rosie Jones. 1999. "Learning dictionaries for information extraction using multi-level bootstrapping". *Sixteenth National Conference on Artificial Intelligence*, 1044–1049. AAAI Press/MIT Press.

Riloff, Ellen & Jessica Shepherd. 1997. "A corpus-based approach for building semantic lexicons". *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, 117–124.

Roark, Brian & Eugene Charniak. 1998. "Noun-phrase cooccurrence statistics for semi-automatic semantic lexicon construction". *Proceedings of the 17th COLING* ed. by Christian Boitet, 1110–1116.Montréal, Canada.

Roche, Emmanuel & Yves Schabes, eds. 1997. *Finite State Devices for Natural Language Processing*. Cambridge, MA: MIT Press.

Royauté, Jean. 1999. *Les groupes nominaux complexes et leurs propriétés: Application à l'analyse de l'information*, PhD dissertation, Université Henri Poincaré, Nancy 1.

Ryckman, Thomas. 1990. "De la structure d'une langue aux structures de l'information dans le discours et dans les sous-langages scientifiques". *Langages* 99. 21–28. Ed. by A. Daladier.

Sager, Naomi, Carol Friedman, & Margaret S. Lyman, eds. 1987. *Medical Language Processing: Computer management of narrative data*. Reading, Mass: Addison Wesley.

Sager, Naomi. 1981. *Natural Language Information Processing: A computer grammar of English and its applications*. Reading, MA: Addison Wesley.

Sager, Naomi. 1986. "Sublanguage: Linguistic phenomenon, computational tool". In *Analyzing Language in Restricted Domains*, ed. by Ralph Grishman & Richard Kittredge, chap. 1, 1–18. Hillsdale, NJ: Lawrence Erlbaum Associates.

Schütze, Hinrich. 1992. "Dimensions of meaning". *Proceedings of Supercomputing' 92*, 787–796. Los Alamitos, CA.

Schütze, Hinrich. 1993. "Word space". In *Advances in Neural Information Processing Systems 5* ed. by Stephen J. Hanson, Jack D. Cowan, & Lee Giles, pp. 895–900. San Mateo, CA: Morgan Kauffmann.

Sekine, Satoshi. 1997. "The domain dependence of parsing". Grishman 1997: 96–102.

Soderland, Stephen. 1999. "Learning information extraction rules for semistructured and free text". *Machine Learning* 34: 1. 233–272.

Spärck Jones, Karen & Julia R. Galliers. 1996. *Evaluating Natural Language Processing Systems*. Berlin: Springer-Verlag.

Tapanainen, Pasi & Timo Järvinen. 1997. "A non-projective dependency parser". Grishman 1997, 202–208.

Vergne, Jacques. 2000. *Trends in Robust Parsing*, GREYC, Université de Caen, Nancy. COLING 2000 Tutorial.

Vossen, Piek, ed. 1998. *EuroWordNet: A multilingual database with lexical semantic networks*. Dordrecht & Boston: Kluwer Academic Publishers. Repr. from *Computers and the Humanities* 32: 2–3, 1998.

Voutilainen, Atro & Juha Heikkilä. 1993. "An English constraint grammar (ENGCG): a surface-syntactic parser of English". In *Creating and Using English Language Corpora*, ed. by Udo Fries, Gunnel Tottle, & Peter Schneider, 189–199. Amsterdam: Rodopi.

Zweigenbaum, Pierre & Consortium MENELAS. 1994. "MENELAS: an access system for medical records using natural language". *Computer Methods and Programs in Biomedicine* 45: 117–120.

Zwicky, Arnold M. & Ann D. Zwicky. 1982. "Register as a dimension of linguistic variation". In *Sublanguage: Studies of Language in Restricted Domains* ed. by Richard Kittredge & John Lehrberger, chap. 9, 213–218. New York: Walter de Gruyter.

Chapter 9

Text generation within sublanguages

Richard I. Kittredge
University of Montreal

1. Introduction

This chapter aims to give an overview of recent research and applications development in the automatic synthesis of texts that conform to the patterns of a given sublanguage. The subfield of computational linguistics known as 'text generation' (cf. McKeown 1985) has been particularly active since the early 1980s, when it became clear that certain types of informative texts can be derived from databases or 'knowledge bases', provided that one has a sufficiently detailed description of the grammar, style, and potential information content of the intended output texts.

Very few existing sublanguages have proven amenable to text generation in any useful way. However, the attempt to model real sublanguage texts by generating them has improved our understanding of text structure, and of the relationship between language and information. A few practical applications that have entered service are modeled on existing sublanguages, albeit with some simplifications.

The primary focus here is on the kind of language used in stereotypical reports. For the purpose of this discussion, a report is a text (1) produced in a recurrent situation of communication, (2) concerning a set of related events which have quantifiable parameters of fixed types, (3) including variation over time, and (4) involving some degree of conceptual summarization (cf. Kittredge & Polguère 2000). Report sublanguages are widely used for weather forecasting, for overviews of commercial market activity, and for summarizing highlights of sporting events. Texts in each of these sublanguages show a high degree of recurrence in their word cooccurrence patterns and sentence structures. Each report sublanguage exhibits stereotypical ways that paragraphs are built from sentences and then ordered to form whole texts (text macrostructure). As we will see below, the automatic generation of reports from databases became a

reality around 1983, following the application of Zellig Harris's work on discourse analysis to simple reporting sublanguages of English and French.

Text generation has for the most part been studied, and applied practically, in 'technologies' where the information to be conveyed can be derived from what is already available in electronic form for some non-linguistic purpose. The source of information may be numerical data (e.g., hourly quotations of stock market prices and volume of shares traded), or alternatively some kind of knowledge representation (e.g., the object model of a stereotypical situation or process, or of the components of a complex device). The sublanguages in question are therefore simpler, more closed, and quite different from the sublanguages of scientific writing identified by Zellig Harris in his pioneering works (cf. Harris 1952, 1968, Harris et al. 1989). The open-ended (i.e., unpredictable) nature of science writing makes it difficult to even imagine a pre-linguistic source for the propositional content of typical sentences, not to mention finding the source information in ready electronic form. Moreover, the principles for organizing sentences into paragraphs, and paragraphs into whole texts, are far less clear in most sciences than they are in technical manuals or reports. Therefore, scientific texts have not been considered amenable to text generation.[1]

Text generation has captured the attention of computational linguists because it presents the set of questions to be answered about the language-information relationship in quite a different light from what we see in the text analysis tradition. Section 2 gives an overview of the text-generation task, comparing it with the task of text analysis as traditionally seen. Section 3 traces the development of practical text-generation systems in sublanguages having straightforward information mappings and used for reporting on fixed types of data. Section 4 discusses how multilingual texts can be derived from a single information source in certain cases, and illustrates the difficulties which may arise when parallel sublanguage texts in two or more languages have different content or have different semantics. Section 5 takes up the problem of sublanguage engineering, which becomes possible as soon as one envisages a practical text-generation system. One can introduce aspects of a 'controlled language' into the resultant text output in order to reduce ambiguity, improve readability, or otherwise make the text more accessible to non-native or non-

1. Science writing seems also to involve complex predications, and to require a much larger grammatical description than is needed to account for texts of the kind discussed here (see Section 6).

expert readers. Finally, in Section 6, we look at some issues for long-term research in applying text-generation techniques to a wider variety of sublanguages, to improving multilingual generation results, and to the problem of summarization, especially in scientific writing.

2. The task of text generation from a sublanguage perspective

The task of generating informative text within a sublanguage forces us to address (or perhaps view differently) at least three important questions that have not always been central to sublanguage research. First, what kind and form of information is appropriate and adequate as a starting point for synthesizing a complete, well-formed text? (The answer to this question seems to be quite dependent on the text type.) The second question concerns how to model the global structure of well-formed texts in terms of groupings of sentences of various types. A third question regards the 'communicative structure' of sentences in a cohesive text: how to choose the most appropriate paraphrastic form of each intended sentence as a function of the thematic development of the text.

2.1 Text generation vs. text understanding

Much of the research in computational linguistics during the past half-century has been oriented towards the problem of language understanding, driven by the need to accomplish practical tasks such as machine translation, content extraction from documents, and question answering. Natural language understanding involves producing a meaning representation for each successive sentence of text (i.e., in a monologue) or for each utterance in a dialogue, a task which typically involves the subtasks of word sense disambiguation, representation of domain knowledge, common-sense reasoning, and pragmatic interpretation. Most research in understanding has been limited to interpreting single sentences, or to tracking one aspect of understanding through a monolog or dialog.

Only a few research groups[2] have focused on the representation of meaning or content for whole texts (as opposed to representations for individual

2. The work by N. Sager and colleagues at New York University's Linguistic String Project on information formatting of medical texts since the 1970s has been one of the most significant achievements in representing whole texts.

sentences, or dynamic models of belief states during dialogue), even though this might be one legitimate end-goal of text understanding. (Another might be some representation of how the understanding of a whole text changes or updates the set of beliefs, attitudes, etc. of the reader.) And yet it is this unordered global representation of content (or intention to change the reader's belief state in a particular way) that might best be taken as the starting point for text generation, which can be seen as the inverse process of text understanding. In the words of one researcher (McDonald 1993: 192):

> A truly reversible linguistic mapping between intentional situation and utterance will have the comprehension process end where the generation process begins. Thus just as the psycholinguistically correct source for generation is still very much a matter of research (. . .), so too is the end-point of comprehension, and by implication the division of that process into components and representational levels.

2.2 Different emphasis in language generation

The task of language generation puts a different emphasis on language modeling than does that of language comprehension. In particular, the generation task includes the problem of planning and ordering the content of individual sentences to make a coherent and rhetorically motivated whole. Not only is this a difficult if not intractable problem for general texts, it is also one that precedes, and therefore takes priority over sentence synthesis in the sequence of generation steps.[3] Most researchers consider the problem of global organization (text planning) to be far more difficult than the problem of sentence generation.

There is a second difference in emphasis that comes with generation research. A language-generation model must never allow the production of a sentence or text that is not acceptable. However, it may 'undergenerate' the set of acceptable output sentences (or texts) somewhat, provided that each input meaning is mapped to an acceptable output among the many paraphrastic alternatives. A perfect analysis model, on the other hand, must account for all observed (and otherwise possible) input sentences, even at the cost of allowing some unobserved sentences to be considered acceptable. That is, it may

3. Harris's comment that generation is less difficult than analysis, in the chapter that introduces Volume 1 of the present publication, should be taken as referring to sentence generation, not text generation.

'overgenerate', but cannot 'undergenerate' the set of acceptable input sentences. In actual practice, language generators tend to seriously undergenerate the target language and sublanguage at the sentence level, while language analysis models tend to seriously overgenerate the acceptable inputs at the text level, i.e., they don't enforce textual constraints that should be a part of the model.

2.3 The 'standard model' of generation

Although there is no consensus about the exact form of speaker intentions which serve as the starting point for generation, most attempts to model language generation in a specific context have broken the process down into three main processing stages (cf. Reiter 1994):

1. Content determination
2. Text structuring
3. Grammatical realization

The first two of these stages may occur in sequence, or in some concurrent or interleaved fashion. The third stage, in most models of generation, can apply only after the first two stages have been completed.

The main task of content determination is to use the intentions of the speaker (or writer) together with the data available in the context or situation to build a set of abstract statements (i.e., propositions) to be communicated by the text. These propositions may be represented in some logical form, and they may each be marked with some degree of importance or salience for the reader.

Text planning determines the global structure of the text as well as the order of sentences within each section and paragraph. Text planning also determines how many propositions to pack into each sentence, and builds a linguistic specification for each sentence. This specification may be in the form of a semantic representation, an articulated syntactic representation, or in some cases even a specification for concatenating ready-made phrases. The form of the specification depends on the requirements of the application, the complexity of the sublanguage and the amount of paraphrase afforded by the targeted sublanguage grammar.

In the final stage, grammatical realization, the word order for each output sentence is determined, and words are grammatically inflected and assembled under the control of the grammatical specification. (Many current systems also include punctuation, formatting, and hyperlink creation during realization, although factors guiding these choices may originate during text planning.) In

principle, grammatical realization can produce phonetic spellings of sentences, including prosodic markings, so that the output can be synthesized speech rather than written text. (Such 'concept-to-speech' generation is less developed at present than concept-to-text generation; thus, the focus here is on written sublanguages.)

2.4 Sublanguage features that must be captured by a generator

Let us consider, at this point, some of the dimensions of a sublanguage description that must be captured in a generation program if it is to be applied to create acceptable texts. First of all, at the level of sentence description, the lexicon and syntactic structures of each sublanguage are specialized, and the usage of words within sentences must reflect the cooccurrence restrictions that apply between verbs and their argument nouns, between nouns and their modifiers, etc. These special properties may be specified simply as restrictions in the lexicon and syntax of the standard language. Nevertheless, unique lexical items and sentence patterns (including sentence fragments) which may exist in certain sublanguages preclude simple application of a subset of standard language resources.[4] Also at the sentence level, there may be a need to state frequency preferences among paraphrastic choices, when they exist. Note, however, that the thematic development of text may strongly favor one choice (e.g., passive over active) within a set of sentences which out of that context might be considered paraphrastic.

In addition to restrictions within sentences, sublanguages may differ widely in the use of cohesion devices to link successive sentences. Cooking recipes in English and French, for example, require specific types of intersentential anaphora (both zero anaphora and pronominalization), while many kinds of instruction manuals use partial repetition anaphora (i.e., coreferential noun phrases reoccurring in reduced form), but virtually no intersentential pronominalization or zero anaphora (Kittredge 1982: 126–128).

Sublanguage texts differ widely in the organization of their macrostructure (the major section and paragraph divisions). Many reporting sublanguages use a fixed and seemingly arbitrary sequence for presenting different kinds of information, as seen in weather forecasts, which invariably discuss wind con-

4. Such restrictions have been studied in detail for medical sublanguages by NYU's linguistic string project (Sager 1981; Marsh 1983).

ditions before temperature variation. Other text types, including instructional texts, show an organization that is dictated more by the steps and motivation for the task to be solved, with special rhetorical links between text segments. Some guiding principles are required for ordering the major constituents of the text, down to the sentence or clause level, and these must be stated within the knowledge available to the text planner.

It is important to emphasize here that the vast majority of text types cannot be generated now or in the near future, for any of several reasons: (1) there is no plausible machine-readable information source, (2) the information does not consist mostly of recurrent (thus predictable) types, (3) the texts do not a show a sufficient degree of regularity with respect to macrostructure, (4) the sentences do not show strong patterns of word-class cooccurrence, or (5) the representations required are simply too complex and numerous, and therefore beyond the current state of research. Research papers in sciences may show sublanguage tendencies at the sentence level, but lack the predictability of text structure and type of information currently required for generation, while also presenting problems of semantic complexity.[5] Newspaper editorials, without domain restriction, would resist generation for all five reasons. Note again, however, that any sublanguage text could be realized (in the third stage of generation as identified in 2.3 above) from a ready-made string of sentence linguistic representations. The prior and larger problem is to determine the content of each sentence from some information source, and then to plan and build the sentence representations in a principled manner.

3. Practical implementations of NLG in sublanguages

This section surveys applications that generate reports of stock market trends, of weather forecasts, of economic statistics, and of sporting events. Some of these applications grew out of research on automatic English-French translation undertaken by the author and colleagues at the University of Montreal's TAUM project during the 1970s.[6] The approach to translation at TAUM had

5. Some of this complexity is due to argumentation carried out within the metascience portion of the discourse. In contrast, many reporting sublanguages lack metastatements on domain material.

6. In particular, the TAUM research group at the University of Montreal built two prototype English-French translation systems, METEO for the sublanguage of weather

been strongly inspired by Harris's work on transformational decomposition and transfer grammar (cf. Kittredge 1976). The first general-purpose TAUM system of 1973 was initially tested on *Scientific American* articles and Canadian administrative texts.

Around 1974 TAUM's focus changed to sublanguage texts, and the system was adapted to telegraphic weather forecasts, which required an entirely different (if simpler) grammar. Contacts were established between the TAUM group and the Linguistic String Project at New York University, where Naomi Sager and her colleagues were investigating methods for the analysis of medical reporting sublanguages. The LSP's fruitful application of information formatting to medical reports (such as those written by physicians upon discharging a patient) suggested to us in Montreal that information formats might serve as a way of comparing and transferring information between comparable sublanguages of two different languages.[7] The close parallels observed in Montreal between English and French forecasts and technical manuals prompted a wider contrastive study of English and French texts in eleven sublanguages (cf. Kittredge 1979, 1982, 1983b). A detailed study of stock market reports led to the observation that the 'core' component of certain stock market reports could be mapped to an information format in the sense used by Harris (1968) and Sager (1972, 1981), where each row (proposition) in the information format could be related to one or more entries in the tabular data published with the report in a newspaper (cf. Kittredge 1981, 1983a). By May 1980 it had become clear that one could in principle automatically derive the crucial core sentences of stock market reports from the tabular data.[8] A paper to this effect was presented to the ASIS conference in 1981

forecasts, and AVIATION, for aircraft hydraulics manuals (cf. Lehrberger 1982). METEO has been used operationally by the Canadian Translations Bureau since 1977, and currently translates more than 30 million words of text annually.

7. Informal discussions with Jim Munz in Philadelphia during 1970–1973 played an important role in bringing the LSP results to my attention.

8. This is when Kittredge & Lehrberger 1982 went to press, see pp.135–136:

> One of the most obvious applications of the observations given above on sentence and text structure within sublanguages is to the generation of text from a semantic base. It seems quite probable that most applications of text-generation systems will be confined to a single domain, requiring that the text be well-formed in a particular sublanguage. Particularly in the cases where texts are produced daily on the basis of numerical or other easily quantifiable information (weather reports, stock market reports), it would be feasible to generate texts in one or more languages.

(Kittredge 1981), leading to discussions with Karen Kukich at the University of Pittsburgh. Over the following two years, Kukich implemented the first stock market report generator as her Ph.D. project.

3.1 Stock market reports — ANA

Kukich's ANA system for stock market summaries (Kukich 1983) became the first actual implementation of a text generator for an existing sublanguage. ANA uses four modules of rules:

1. Fact generator
2. Message generator
3. Discourse organizer
4. Linguistic 'text generator'

These four modules are applied in sequence to synthesize texts from numerical data — the twice-hourly quotations for stock indexes on North American exchanges.

The fact generator (1) applies simple arithmetic operations to derive elementary facts in abstract form. An example might be that the Dow Jones index of industrial stocks decreased by a certain value over a particular half-hour interval. In message generation (2) the rules interpret configurations of elementary facts to instantiate abstract concepts and build 'messages' of interest to the audience. For example, a net decrease in the industrial index followed during the same day by a significant and sustained increase might trigger a rule which describes an 'afternoon rally' in the index value. These first two steps, fact selection and message generation, together constitute content selection for ANA's reports.

The discourse organizer (3) groups and orders the messages into paragraphs according to topic, so that the content and overall form of each output sentence is specified. Thus the text is fully planned, except for the choice of specific lexical phrase to convey each abstract element of sentence meaning. The linguistic text generator (4) in ANA uses a 'phrasal lexicon' (Becker 1975) containing a large number of paraphrastic variant phrases for each concept to be expressed. For example, to signal a large increase in price following a downward trend earlier the same day, the phrasal lexicon allows choice between phrases such as "rebounded sharply", "made a strong recovery from its earlier losses", "moved up smartly following a previous decline", etc.

ANA was not designed to be a practical application, but rather to prove that

the initial (and most stereotypical) paragraphs of human-written market reports could be modeled convincingly in a rule-based system using only stock data as input. The use of a phrasal lexicon allowed very colorful language to be produced with relative ease, since there is a great deal of paraphrastic variation in stock reports, but the phrases fall into a very small number of semantic categories. Thus the random selection of a phrase for each category in a sentence could give rise to a unique and idiomatic output sentence (see Figure 1).

The stock market was catapulted sharply and broadly higher yesterday, as stock prices posted gains for most of the day. Trading was active.

Figure 1. Fragment of stock market report generated by ANA

3.2 Weather forecasts — the FoG system

The first operational text-generation system based on an existing sublanguage was FoG (an acronym for 'Forecast Generator'), designed to synthesize marine and public weather forecasts from the forecast data produced by numerical weather models. An initial sublanguage study for FoG was undertaken in 1985 on a corpus of English marine forecasts issued by Environment Canada for Arctic regions. The resulting sublanguage grammar, covering most of the corpus, served as the basis for a prototype forecast generator, which used essentially the same four processing steps as ANA to produce acceptable forecasts by means of a phrasal lexicon (Kittredge et al. 1986). FoG was then expanded to cover marine and public forecasts in both English and French as written by forecasters in Eastern Canada (Bourbeau *et al.* 1990, Goldberg *et al.* 1994). Moreover, the bilingual FoG system used a dependency grammar to represent the output of the text planning stage, and abstract away from the superficial differences between English and French syntax. Linguistic work on FoG benefited from the earlier development of the TAUM-METEO translation system at the University of Montreal for the same sublanguage during 1974–1976. (But now, the goal was to eliminate the need for machine translation by generating both English and French texts simultaneously from the same input data — see Section 4 below for details on multilingual generation.)

Figure 2 gives a sample forecast generated by FoG for the Halifax region during a severe storm in 1990.As can be seen from this text (and on marine

NORTHERN HALF OF GULF MAGDALEN
CHALEUR MISCOU
ANTICOSTI.
GALE WARNING UPGRADED TO STORM WARNING.
FREEZING SPRAY WARNING CONTINUED.
WINDS WESTERLY 10 TO 20 KNOTS SHIFTING TO EASTERLY 15 TO 25 THIS AFTERNOON THEN INCREASING TO EASTERLY GALES 35 TO 45 THIS EVENING.
NORTHEAST GALES 35 TO 50 LATE TONIGHT DIMINISHING TO NORTHEAST 30 TO GALES 45 SATURDAY AFTERNOON.
A FEW FLURRIES TODAY AND CONTINUING SATURDAY.
VISIBILITY NEAR 3 IN FLURRIES AND ONE HALF TO 2 IN SNOW.
FREEZING SPRAY TONIGHT AND SATURDAY.
TEMPERATURES MINUS 6 TO MINUS 3.
OUTLOOK FOR SUNDAY STRONG TO GALE FORCE NORTHERLIES.

Figure 2. Marine forecast for one Atlantic maritime area (output from FoG system)

weather web sites around the world), the sublanguage of marine weather forecasting makes use almost exclusively of sentence fragments containing no tensed verb. The most complex sentences usually deal with wind speed and direction, which may change several times over the 2–3 day forecast period. The global text structure is quite fixed, with information usually presented in the order in Figure 3 (parentheses indicate information types which may be absent in a particular forecast, depending on actual conditions).

The standard information ordering appears to be a somewhat arbitrary fact about this sublanguage, which needs to be learned by new 'speakers', and which does not seem to be derived entirely from 'first principles'. It has thus

<list of forecast regions sharing the same forecast>
(<warnings of strong winds>)
(< freezing spray warnings>)
<wind direction and speed>
(<precipitation>)
(<fog / mist>)
<visibility>
(<freezing spray details>)
<temperatures>
(<outlook for next day>)

Figure 3. Information type ordering for Canadian Atlantic marine forecast

been called an example of *domain communication knowledge* (Kittredge, Korelsky & Rambow 1991), a kind of knowledge about sublanguages that falls under the general label of text linguistics. However, the ordering alone is not the entire story.

The freezing spray details are dependent in certain ways not only upon the occurrence of the freezing spray warnings earlier in the text but also their precise content. Likewise, wind warnings are linked with details of wind event descriptions. Other dependencies link the details of statements about precipitation, fog and mist, and visibility. These links are generally cross-serial, that is, they create texts containing sequences of the form A B C A′ B′ C′ , where A, B and C are linked with A′, B′ and C′ respectively. Thus, the macrostructure of the texts is essentially context-sensitive. It is easy to show that such texts cannot be planned dynamically by a recursive top-down process of expanding rhetorical operators, which is the method of choice in planning instructional texts. Report texts can thus be seen as less rhetorically 'logical' in their global organization, and more 'conventional'.

3.3 Economic statistics — LFS

'Well-behaved' sublanguages (such as weather forecasts) are not always easy to find, but the statistical offices of national governments provide a goldmine in the form of periodic economic summaries of economic indicators. Figure 4 shows a paragraph from a long monthly report on the national employment picture, issued by Statistics Canada.

In addition to these reports on employment, similar summaries cover retail trade, the Consumer Price Index, and other measures of national economic health. Reports are published together with tabular data that support the propositional content of each sentence. Indeed, the data in the tables that accompany a monthly report are sufficient to determine the main content of the report, provided that one has heuristic rules to select facts of interest from the

> For the week ending September 15, 1990, the seasonally adjusted estimate of employment increased by 30,000 to 12,602,000. This month's rise in employment was evenly distributed between men and women. The overall employment / population ratio edged up to 1.5 (+0.1).

Figure 4. Paragraph from *The Labour Force*, page A-2, Statistics Canada, September, 1990

data (e.g., a definition of what constitutes a 'slight' change in the percent of unemployment from one month to the next). To organize the content into a typical three-page report, one must also follow a stereotypical pattern of information ordering and sentence structure. Each set of input data (organized into four tables in the case of employment summaries) thus can be mapped to a report. To be more exact, each dataset can be seen to determine a set of equivalent reports containing the same information, since semantic paraphrases exist in this sublanguage.[9]

3.3.1 *Text planning in the LFS generation system*

The LFS system was developed during 1991–93 in Montreal to test the feasibility of generating extended texts with a linguistic model that accounts for semantic paraphrases, thematic progression and other linguistic features needed for full control over the production of professional-sounding text. The Meaning-Text stratificational model of language (cf. Mel'čuk & Pertsov 1987) was used for sentence realization, acting on sentence specifications that were output from an integrated module that interleaved content determination and text planning.[10]

One of the significant issues in generating statistical summaries of this type is the planning of their global text structure. These summary texts have a relatively fixed overall structure, carrying certain obligatory information, but may contain some optional sentence-forms which give extra detail when the month-to-month changes in the data values are relatively large, or when the size and direction of change casts light on a longer trend. The outline structure of the texts can be represented as a tree whose nodes correspond to text segments that might appear in a maximally complex well-formed report. The leaves of the tree correspond to potential propositions that might be

9. Such government reports are carefully edited before publication, and we might be tempted to speak here of a controlled language (see Section 5). Nevertheless, there seem to be no guidelines for writing them. Employment reports from different areas and sources seem to represent a single natural sublanguage, showing very similar word distribution.

10. The Meaning-Text framework was chosen because of its orientation to language generation, its detailed representational mechanisms for semantic paraphrase, and the ready availability of English and French descriptions within this framework in Montreal. There are important similarities between the Meaning-Text and Harrisian frameworks, especially regarding paraphrase, but a discussion of these is beyond the scope of this summary article.

instantiated from the data. In any given report, however, some optional branches of the tree may not be developed, because the corresponding data do not exceed the quantitative threshold of interest that would justify instantiating propositions for the report The text planning tree structure guides the process of verifying if the data for each information type are significant enough to be conveyed in one or more propositions, perhaps through further branching. A fact representation is built for each such piece of information (e.g., that the nationwide employment level for women rose by 1.2%, or that employment in the construction industry was down by 7% in Alberta).

Activating the text plan in LFS consists of traversing this plan tree while building a separate instantiated tree, and at each node requesting a specific kind of computation on the data. Nodes associated with an obligatory type of information require traversal of daughter nodes. Optional branches in the tree are traversed depending on the significance (e.g., size of change, support for trend pattern, etc.) of the associated messages. Thus the instantiated tree built during plan tree traversal will correspond to a trimmed-down version of the plan tree, modulo a few local changes.

Local changes occur when some sentences or clauses of the text must be ordered according to salience. For example, the tests on the data dictated by the plan tree may result in reordering some messages in the instantiated tree. Thus, if the employment situation for women has changed more dramatically than that for men, the text section dealing with breakdown by age group and sex will discuss changes among women before those for men. Another example of local change is a process called *aggregation*, in which two or more messages with very similar types of information may be fused together to build a single sentence specification (e.g., "Employment in the construction industry rose in Alberta, British Columbia, and Saskatchewan.").

The development of a text plan tree for a report sublanguage requires distributional analysis at the sentence level. Rather than word classes based on cooccurrence similarity, the text grammar incorporated into a text plan tree uses sentence classes based on distribution with respect to neighboring sentence classes. The sentence classes of interest for a text grammar use a notion of information equivalence. For example, the initial sentence of an LFS report, no matter what the particular words or syntax chosen, concerns the overall monthly change in the Canadian employment situation.

Sentences in the same class, i.e., that are informationally equivalent, can be seen as answers to the same question, such as "Which Canadian provinces showed employment gains in the construction industry over the past month?". If two sentences are both answers to this question, and can both occur at the

same point in a sublanguage text, they are considered informationally equivalent.

Note that not all general-language paraphrases of the same sentence will be in the same equivalence class, since equivalent sentences must be sensitive to the thematic progression of the text. The structure of the question that defines each equivalence class can be used to set up the theme/rheme structure of any sentence which can be generated in response to that question (with respect to a given data set) to satisfy the need to communicate information at that point in the report.

3.4 Game summaries

One text genre in which sublanguage texts have recently been generated is that of short synopses of sporting events. The press summary of a football game shown in Figure 5 illustrates a sports sublanguage for which report generation has been shown to be feasible.

Outback bowl

South Carolina 24,
No.19 Ohio St. 7
TAMPA, Fla. – The Gamecocks, *winless in 1999 and losers of 21 straight entering this season, finished the most dramatic one-year turnaround in Southeastern Conference history* by winning the Outback Bowl.

Ryan Brewer, *an Ohio native*, rushed for 109 yards and scored on runs of 7 and 2 yards, as well as a 28-yard screen pass play and gained 219 total yards for South Carolina (*8–4*).

The Buckeyes (*8–4*) avoided a shutout when offensive lineman Mike Gurr recovered Jonathan Wells' fumble in the end zone for a third-quarter touchdown.

Figure 5. Football game summary–italics added for background segments (The Associated Press, Tuesday, January 2, 2001)

Certain parts of the text above refer directly to actions and their results during the game (e.g., *Ryan Brewer rushed for 109 yards*), while other segments provide background information on the teams, the players, or the league (e.g., (*Ryan Brewer is) an Ohio native*). This is an example of a two-component reporting sublanguage with a grammatically dominant 'core' component referring to the main subject matter, a bounded event.[11] The background information is generally conveyed in phrases which are adjoined in grammatically subordinate

11. This phenomenon is also exemplified by stock market reports (cf. Kittredge 1982: 130–133; 1983a)

positions such as reduced relative clauses (e.g., ", winless in 1999 and losers of 21 straight entering this season, ") and parentheticals ("(8–4)").

The STREAK system (Robin 1994) demonstrated that a basic description of a basketball game could be generated from a game score box containing information about the scoring activity of each player. In the lexically rich sublanguage of sports reporting, as in stock market reports, nuances of word choice can convey several facts simultaneously. For example, the sentence "Utah outlasted Portland 102–98" conveys the facts that (a) the Utah team beat Portland 102–98, (b) it was a close game during most of the contest, and (c) the lead changed hands many times until the very end. Like some earlier systems such as FoG and LFS, STREAK used a fine-grained linguistic approach to lexicalization (i.e., instantiation of lexical entries based on simultaneous satisfaction of multiple facts or concepts derived from the input data).

An important innovation of STREAK, however, was to incrementally revise the basic game text by attaching additional background information about teams and players at appropriate points. This two-component aspect of a reporting sublanguage had been described in earlier linguistic work on stock market reports, but STREAK was the first system to use a revision strategy that explicitly represents the historical (i.e., background) information and weaves it into the output report.

4. Multi-lingual report generation

The bilingual nature of Canadian society has offered several good opportunities to study parallel sublanguages of English and French. Many reports produced by government agencies, including weather forecasts and economic summaries, are regularly composed by human authors from the same reference data. Official reports are usually edited and translated carefully[12] to ensure that they convey the same information in both languages, yet conform to their respective sublanguage standards. This has presented an interesting challenge for text generation, because the equivalent information conveyed in two different languages may appear in quite different (non-parallel) sentence structures, or even be distributed across a different number of sentences. A

12. Source texts may be composed in either language, then translated, or in some cases co-written by two editors.

bilingual text generator needs to start from a common set of input data, and presumably make a single selection of facts and concepts to be used in both languages. At some point, however, the linguistic differences of the output languages become important, so that separate lexicons and grammars must apply. Both FoG and LFS were implemented as bilingual generators, but they used quite different solutions to the language divergence problem.

4.1 Bilingual FoG

Telegraphic weather forecasts, as traditionally written by Canadian forecasters, exhibit strong similarities in English and French. Not only is the information conveyed strictly equivalent, but the sentence boundaries correspond closely. Furthermore, the conventional choices for sentence syntax are typically the same, except for a few non-parallel local syntactic constructions:

(1) a. Winds northwesterly 25 knots veering to northerly
b. Vents du nord-ouest 25 noeuds devenant du nord

An unusual feature of the English telegraphic forecasting sublanguage is the post-position of direction adjectives to a head noun such as *wind*, especially in sentence-initial noun phrases such as (1a). (Note that one could analyze this otherwise 'bizarre' construction as being derived from a zeroed *will be* occurring between head noun and adjective in an elementary sentence (*The) winds will be northwesterly.*) French uses a prepositional phrase *du nord-ouest*, as witnessed by (1b), a small and very local structural difference. Leaving aside such local and superficial differences, the two sublanguages are highly parallel. Neither sublanguage permits any significant paraphrasing or stylistic variation. For the most part, there is only one 'best' way to express any given content.

Under these conditions, the text planning for FoG's automatically generated forecasts can be carried out simultaneously for English and French. Moreover, the output of text planning can be written in the form of 'intralingual' syntactic specifications that abstract away from the most important English-French differences, but otherwise map straightforwardly to surface syntactic patterns in the two languages. (For details of FoG's dependency grammar formalism, which use Meaning-Text deep syntactic structures as interlingual representation, see Kittredge & Polguère 1991.) From the interlingual specifications, separate English and French grammars operate independently to derive the bilingual output.

4.2 Bilingual LFS

The bilingual statistical summaries issued by Statistics Canada represent a somewhat more complex text genre than telegraphic weather forecasts. The texts are not only relatively lengthy, with several sections and subsections covering a variety of phenomena, they also show significant paraphrastic variation within each language's sublanguage. For example, sentences (2a), (2b), and (2c), all found in naturally occurring LFS reports, are strictly paraphrastic:

(2) a. Employment remained virtually unchanged.
b. There was virtually no change in employment.
c. There was little change in employment.

Given that natural texts show frequent variation among such forms, the challenge for automatic generation of these texts is therefore to use a language model that allows any of the paraphrase forms to be generated from the same input. Some method of random variation (or principled variation, which would include avoiding repetition of nearby words and phrases) should then be implemented to provide the stylistic effect of human-written texts. Expressing these paraphrases, however, requires a rather deep semantic representation to be used during the choice of lexical and structural means for an output sentence.

A similar amount of paraphrase variation is found in statistical summaries written in French. Moreover, the 'parallel' sentences chosen by the translator or editor in published bilingual texts often show deep semantic differences in the two languages. For example, an English report may contain (2b), while the French equivalent may contain (3).

(3) L'emploi a peu varié. (roughly, 'Employment changed little.')

The practical solution used in LFS was to create 'intralingual' semantic specifications for sentences as the output of text planning (Iordanskaja et al. 1992, Lavoie 1995). In the parallel sublanguages of quantitative economic statistics, most sentences in both languages can be paraphrased in terms of a set of simple words that correspond one-to-one between English and French. These serve as the basis for semantic network representations according to Meaning-Text models for the two languages (cf. Melčuk & Pertsov 1987, Ch. 1). Most of the semantic variation in LFS texts could be represented by lexicalizing in different ways from the initial semantic primitives found in the networks. Some of the

cross-linguistic differences could also be represented by this means, although not in a principled way for the example pair (2b) and (3).

5. Sublanguage and controlled language generation

Many sublanguages used in reporting of the kind described above have evolved gradually over many years and exist in many variant forms, depending on space constraints of the print medium and the specific language community of the author. We have used the term sublanguage rather loosely to include one variant, or a set of related variants, which presumably exhibit enough regularities of word-class distribution to distinguish them as a grammatical subsystem.

In most cases, reporting sublanguages can be called 'natural sublanguages', in that they have arisen spontaneously, continue to evolve over time, and reinforce their regularities through some recurrent conditions of usage that apply to a more or less loosely defined community of specialized language users.

In some cases, however, reporting sublanguages, like other sublanguages, have been standardized by governmental or professional organizations with a view to removing any potential ambiguity for language users who may lack either native command of the standard language or expert knowledge of the subject matter. The term *controlled language* has been used to refer to a sublanguage or a family of related sublanguages which have been 'engineered' into a simpler and less ambiguous standardized form. A controlled language is often much broader than a sublanguage, and it is subject to explicit rules that have been promulgated and are consciously enforced. The case of Canadian weather reports borders on a controlled language, in the sense that forecasters-in-training use a kind of style manual (MANPUB) illustrating well-formed forecasts, giving definitions of terms, and containing a number of writing caveats. However, MANPUB guidelines do not include explicit limits on sentence length and the kind of grammatical constraints that appear in Simplified English, as used by aerospace manufacturers to write their maintenance manuals (cf. AECMA 1995).

The AECMA controlled language standard for writing aerospace manuals includes several dozen rules ranging from precise limits on nominal compounding (no more than three nouns in a row), limits on zeroings (no deletion of articles), to vague admonitions to express no more than one thought per sentence. More crucially, AECMA Simplified English limits the verbs that can be used to a specific list, and suggests paraphrastic replacements

for commonly used alternatives (e.g., replace *test the landing gear* by *do a test of the landing gear*). These rules and vocabulary limitations are intended to cover the family of related sublanguages used in mechanical, hydraulic, electronic, and other aerospace systems. Engineers who typically write manuals have sometimes reacted negatively to the deformations which controlled language standards apparently inflict on their natural sublanguages. A niche industry has sprung up involving controlled language style checkers, which automatically scan text as it is being written, detecting violations of standards and if possible suggesting alternatives.

The issue of controlled language is particularly relevant to sublanguage when there is an effort to generate texts for a domain that is characterized by a sublanguage. By nature, most reporting sublanguages have a relatively fixed lexicon of frequently-occurring words, as well as an inventory of typical syntactic constructions. But a corpus study normally reveals infrequent variant words and alternative sentence patterns that are accepted as professional, but used only occasionally. A language generator, although built with a goal of matching the variety and complexity of a natural sublanguage, inevitably falls short. It is necessary, in most cases, to 'truncate' the lexicon and grammar at some level of frequency, in order to simplify and perhaps clarify usage.

Applications of natural language generation to sublanguages are still in a very preliminary stage, so there is little evidence for a set of principles to guide the simplification and standardization of sublanguages in the course of constructing generators. Where professional report writers have had an opportunity to react to generated output over a period of months or years, as in the case of FoG and LFS, the elimination of infrequent variants from a sublanguage does not seem to be a major problem, so long as infrequent messages (i.e., types of content) are 'sayable' in some form by the generator. Thus, the emphasis has been put on developing front-end content determination rules that can detect most if not all configurations of data that constitute significant concepts for a report.

Future years will no doubt witness an expansion of attempts to generate sublanguage texts, both monolingually and in multiple languages. The opportunity to introduce standardization and regularization of output during construction of a generator presents both problems (if not done with a sensitivity to existing sublanguage practice) and opportunities (especially for sublanguages where texts have often been hastily written, as with weather forecasts). Standards introduced may apply only to the 'in-house' practice of a particular organization, or they may try to capture the best practice of a

specific group of professional text writers, translators and editors. Since so little is known about sublanguage grammars and their relationship to emerging controlled language standards, the linguistic study of these issues should become increasingly active in the foreseeable future.

6. Research issues in generation of sublanguage texts

The discussion above has dealt with generation in natural sublanguages that fall into the report genre, and has identified in passing some open issues that require further study. We now look more broadly at the set of research issues facing generation within sublanguages, which has raised important new challenges for sublanguage analysis methodology.

6.1 Generating instructional texts

An important body of current language-generation research has been concerned with generation of instructional sections within technical manuals, which differ considerably from reports in two major ways. First, instructional texts are characterized by the frequent occurrence of characteristic syntactic phenomena such as imperative sentence forms, zeroing of anaphoric object noun phrases, and ellipsis of definite articles. Second, the global organization of an instructional text is not nearly so conventional as that of a report, but instead reflects the logical links between steps in achieving the goals of operation, maintenance, repair, and so forth, in the particular complex system. The planning of instructional sections requires a representation of the rhetorical relations between text segments, and in particular of how the juxtaposition of segments, with or without an explicit rhetorical connective, accomplishes specific communication goals. Two of the most significant recent projects in text generation have involved instructional texts from existing technical sublanguages.

One of the main analytical tools used in instructional text planning is the Rhetorical Structure Theory (RST) of Mann and Thompson (1987), which identifies explicit or implicit rhetorical relations between clauses and (recursively) larger text segments. The recognition by the reader of a rhetorical link between text segments is an important part of successful goal-oriented communication and must be planned (by the writer or by the text generator) for global coherence and effectiveness. The rhetorical structures of two

important varieties of instructional texts have been studied in some detail in two European projects of the past decade. In the TECHDOC project in Ulm during the early 1990s (Roesner & Stede 1994) rhetorical structure planning was used to create automotive maintenance instructions for an experimental prototype system. One of the goals was to develop techniques for multilingual generation, covering English, French and German. Although the TECHDOC project did not result in a practical implementation, it discovered some important differences among the three languages in the typical rhetorical development of instructions at the sentence level. This area of contrastive rhetorical structure analysis needs to be pursued further, and extended to other kinds of sublanguages.

Software user manuals are a second type of instructional text, investigated in the DRAFTER project (Paris et al. 1995). In this project, the specific goal was to produce drafts of English and French software instructions, allowing an editor to reformulate the draft by modifying the prelinguistic specifications, rather than, or in addition to, the text itself. Much of the work in DRAFTER went into flexible text planning, including building a library of discourse plans that achieve different communication goals. The project showed that discourse plans must be instantiated somewhat differently both for different languages and for specific sublanguages about software.

6.2 Generating equivalent texts in multiple languages

A major problem that has surfaced in multilingual generation concerns the notion of 'equivalent text'. The parallel texts of Canadian English and French that were modeled in LFS showed relatively frequent differences in exact semantic content, and in the distribution of that content over successive sentences. Since the texts are produced in the same cultural context we assume that the observed variation must be accounted for in the descriptive model. Likewise, in the case of cross-language variation in rhetorical organization of instructions observed in TECHDOC and DRAFTER, researchers have assumed that the global goal of the instructions has been preserved in the (high-quality) translations used as a corpus, so the variation represents a different organizational preference within a common set of linguistic means (different sequence of rhetorical links between text segments) for achieving the same communicative goal. In cases where texts are not used in culturally similar situations across languages, however, one could argue that quite different conceptualizations are being made within the same domain, and that no single instantiation

of propositions from the initial data will be meaningful to text users in two or more languages. A clear case of different conceptualizations occurs in weather forecasts as given by speakers of Inuktitut in Northern Canada, where wind direction must be given relative to the shoreline, and not with respect to points of the compass.[13] An extension of the FoG system to such a language would require separating the content determination components of the output languages, and thus also require independent text planning for each language.

6.3 Generating summaries of text

One of the most active current research areas for generation in sublanguages is the automatic production of summaries of a source text. Much of the work to date has aimed to provide domain-independent techniques for selecting the most 'salient' sentences (or other segments) in a text, and then perhaps trimming and transforming these sentences before concatenating them to form a summary text. Limited success in summarizing news articles has been achieved by the use of statistical methods, sometimes in conjunction with basic linguistic information such as part-of-speech tags. It already seems clear, however, that different text types will require some differences in the approach to summarization, if high-quality summaries are to be achieved. In particular, the text summarization model needs to know about the text-planning principles for each text type, and include fine-grained linguistic knowledge of how paraphrase and text cohesion works for the particular text type and sublanguage.

In the special case of science articles, where abstracting has long been a practical goal, there may be a new opportunity to bring together three distinct streams of research: (1) traditional (Harrisian) grammatical descriptions of sublanguage which specify well-formedness at the sentence level, (2) text-generation achievements in sentence planning, rhetorical structure analysis and semantic paraphrasing, and (3) research into the argumentation of scientific texts, including Harrisian analyses in terms of meta-science predicates. One of the problems then posed to Harrisian sublanguage analysis is how to characterize the differences between full scientific texts and their abstracts. Can one characterize, if not in general then for each science sublanguage, the distinctions between source texts and (author/expert-written)

13. Personal communication circa 1990 from Sam Metcalf, an Inuit elder working at Northern Affairs Canada, Ottawa.

abstracts within the same sublanguage? How do full texts differ from abstracts in their use of metascience predicates? Composition of abstracts could be seen as a transformational problem of semantic paraphrasing under constraints of space and argumentation structure. More practically, it might be seen as the problem of planning a coherent sublanguage text using selected elementary sentences of the source article, while preserving certain aspects of the source article's argumentation.

References

AECMA. 1995. *AECMA Simplified English: A guide for the preparation of aircraft maintenance documentation in the international aerospace maintenance language.* Document PC-85-16598. Brussels: The European Association of Aerospace Industries.

Becker, Joseph. 1975. "The phrasal lexicon". In *Theoretical Issues in Natural Language Processing: Proc. of the (First) Interdisciplinary Workshop* (Cambridge, MA, 10–13 June 1975). Ed. by R. Schank & B. Nash-Webber, Association for Computational Linguistics, 70–73. Cambridge: MIT Press. [O.p., repr. by ACL forthcoming.]

Bourbeau, Laurent, D. Carcagno, E. Goldberg, R. Kittredge, & A. Polguère. 1990. "Bilingual generation of weather forecasts in an operations environment", in *Proceedings of the 13th International Conference on Computational Linguistics* (COLING'90), Helsinki, August 1990, vol. 3: 318–320. San Fransisco: Morgan Kaufman.

Goldberg, Eli, Norbert Driedger, & Richard Kittredge 1994. "FoG: a new approach to the synthesis of weather forecast text." *IEEE Expert* 9. 2: 45–53.

Grishman, Ralph. 1979. "Response generation in question answering systems." *Proceedings, 17th Annual Meeting of the Association for Computational Linguistics*, La Jolla, 99–101. Cambridge: MIT Press. [O.p., repr. by ACL forthcoming.]

Grishman, Ralph & Richard Kittredge, eds. 1986. *Analyzing Language in Restricted Domains: Sublanguage description and processing.* Hillsdale, NJ: Erlbaum.

Harris, Zellig S. 1952. "Discourse analysis." *Language* 23: 1–30.

Harris, Zellig S. 1968. *Mathematical Structures of Language.* New York: Wiley-Interscience.

Harris, Zellig S., Michael Gottfried, Thomas Ryckman, Paul Mattick, Jr., Anne Daladier, T. N. Harris, & S. Harris 1989. *The Form of Information in Science: Analysis of an immunology sublanguage.* Dordrecht: Kluwer.

Hirschman, Lynette. 1986. "Discovering sublanguage structures." In Grishman & Kittredge (1986: 211–234).

Iordanskaja, Lidija, Myunghee Kim, Richard Kittredge, Benoit Lavoie, & Alain Polguère. 1992. "Generation of extended bilingual statistical reports." *Proceedings of the 14th International Conference on Computational Linguistics* (COLING'92), Nantes, 1019–1023. San Fransisco: Morgan Kaufman.

Iordanskaja, Lidija, Richard Kittredge, & Alain Polguère. 1991. "Lexical selection and paraphrase in a Meaning-Text generation model", In C. Paris, W. Swartout, & W. Mann,

eds., *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Dordrecht: Kluwer, 293–312.

Kittredge, Richard. 1976, "Transformational decomposition and transfer grammar". *T.A. Informations* 2: 50–54.

Kittredge, Richard. 1979. "Textual cohesion within sublanguages: implications for automatic analysis and synthesis". In *Représentation des connaissances et raisonnement dans les sciences de l'homme* (Proceedings of Colloquium held at Saint Maximin), ed. by M. Borillo, INRIA, Rocquencourt, France.

Kittredge, Richard. 1981. "A contrastive view of sublanguages". In *Proceedings of the Annual Meeting of the American Society of Information Science* (ASIS–81). Washington, DC, October 1981.

Kittredge, Richard. 1982. "Variation and homogeneity of sublanguages". In Kittredge & Lehrberger (1982: 107–137).

Kittredge, Richard. 1983a. "Semantic processing of texts in restricted sublanguages". *Comp. & Maths. with Applications* 9.1: 45–58.

Kittredge, Richard. 1983b. "Sublanguage-specific computer aids to translation: a survey of the most promising application areas". Unpublished report to Translation Bureau, Secretary of State Dept., Ottawa, Canada.

Kittredge, Richard. In press. "Sublanguages and controlled languages". In *A Handbook of Computational Linguistics*, ed. by Ruslan Mitkov. Oxford: Oxford University Press.

Kittredge, Richard, Lidija Iordanskaja, & Alain Polguère. 1988. "Multi-lingual text generation and the Meaning-Text theory". *Proc. of the 2nd Intl. Conf. on Theoretical and Methodological Issues in Machine Translation*, Carnegie Mellon University, 1–12.

Kittredge, Richard, Tanya Korelsky, & Owen Rambow. 1991. "On the need for domain communication knowledge." *Computational Intelligence*, 7. 4: 305–314.

Kittredge, Richard & John Lehrberger, eds. 1982. *Sublanguage: Studies of language in restricted semantic domains*. Berlin: DeGruyter.

Kittredge, Richard & Alain Polguère. 1991. "Dependency grammars for bilingual text generation: inside FoG's stratificational model", *Proceedings of the International Conference on Current Issues in Computational Linguistics*, 318–330. Penang: Universiti Sains Malaysia.

Kittredge, Richard & Alain Polguère. 2000. "The generation of reports from databases". In Robert Dale, Hermann Moisl & Harold Somers, eds., *A Handbook of Natural Language Processing*, pp. 261–304. New York: Marcel Dekker.

Kukich, Karen. 1983. *Knowledge-Based Report Generation: A knowledge-engineering approach to natural language report generation*. Ph.D. dissertation, University of Pittsburgh.

Lavoie, Benoît. 1995. "Interlingua for bilingual statistical reports". In Richard Kittredge, ed., *Working Notes of the IJCAI–95 Workshop on Multilingual Text Generation* (Montreal, 20–21 August 1995), pp. 84–94.

Lehrberger, John. 1982. "Automatic translation and the concept of sublanguage". In Kittredge & Lehrberger (1982. 81–106).

Mann, William & Sandra A. Thompson. 1987. "Rhetorical structure theory: description and construction of text structures". In G. Kempen, ed., *Natural Language Generation: New*

results in artificial intelligence, psychology and linguistics, pp. 85–95. Dordrecht: Martinus Nijhoff Publishers.

Marsh, Elaine. 1983. "Utilizing domain-specific information for processing compact text." In *Proc. of the (First) Conference on Applied Natural Language Processing* (Santa Monica, CA, 1–3 February 1983). Assoc. for Computational Linguistics, pp. 99–103. San Fransisco: Morgan Kaufman.

McDonald, David. 1993. "Issues in the choice of a source for natural language generation." *Computational Linguistics* 19:191–197.

McKeown, Kathleen. 1985. *Text Generation.* Cambridge: Cambridge University Press.

McKeown, Kathleen, Jacques Robin, & Karen Kukich. 1995. "Generating concise natural language summaries." *Information Processing & Management* 31:703–733.

Mel'čuk, Igor & Nikolai Pertsov. 1987. *Surface Syntax of English: A formal model within the Meaning-Text framewor*k. Amsterdam & Philadelphia: John Benjamins.

Paris, Cecile, Keith vander Linden, Markus Fischer, Anthony Hartley, Lyn Pemberton, Richard Power, & Donia Scott. 1995. "A Support tool for writing multilingual instructions." *Proc. of the 14th International Joint Conference on Artificial Intelligence* (IJCAI'95), Montreal, 20–25 August 1995, 1398–1404. San Fransisco: Morgan Kaufman.

Reiter, Ehud. 1994. "Has a consensus NL generation architecture appeared, and is it psycholinguistically plausible?" *Proc. of the Seventh International Workshop on Natural Language Generation, Assoc. for Computational Linguistics*, Kennebunkport, 163–170.

Robin, Jacques. 1994. *Revision-based generation of natural language summaries: Corpus-based analysis, design, implementation and evaluation.* Ph.D. dissertation, Computer Science Department, Columbia University.

Roesner, Dietmar & Manfred Stede 1994. "Generating multilingual documents from a knowledge base: the TECHDOC project." In *Proc. of the Fifteenth International Conference on Computational Linguistics* (COLING'94), Kyoto, pp. 339–346. Cambridge: MIT Press. [O.p., repr. by ACL forthcoming.]

Sager, Naomi. 1972. "Syntactic formatting of science information". In *AFIPS Conference Proceedings* 41:791–800. [Repr. in Kittredge & Lehrberger 1982:9–26.]

Sager, Naomi. 1981. *Natural Language Information Processing.* Reading, MA: Addison-Wesley.

Chapter 10

A distributional semantics applied to computer user interfaces

Richard Smaby
Clarion University of Pennsylvania

Methods of distributional analysis are useful for understanding multi-modal computer user interfaces and this understanding can guide the design of readily learnable natural-language interfaces. Distributional methods reveal structures of operator structure and co-reference in user interfaces. The distributional methods utilized here are based on and in some cases extend those found in Zellig Harris's works. Applying distributional methods to user interface design also sheds light on ideas at the foundations of semantics: paraphrase and reference.[1]

Introduction

A hot topic in computer user interfaces today is multi-modal user interfaces. The idea is to mix various modalities of user interaction with a computer, e.g., pointing at graphical icons and clicking with a mouse, gesturing with one's hand, typing keys, and speech input and output. I argue that classic methods of linguistic analysis are useful for understanding multi-modal interfaces and this understanding can in turn lead to improved user interfaces, especially those having a natural-language component. I will also describe how this research area sheds light on ideas at the foundations of semantics, in particular, the notions of reference, operator structure, and paraphrase found in the works of Zellig Harris (Harris 1951, 1954, 1970, 1988, 1991). Harris succinctly

1. I wish to thank Bruce Nevin and an anonymous reviewer for criticisms of an earlier draft that have led to a much improved chapter.

presented most of the distributional methods used in the current chapter already in "Distributional structure" (Harris 1954). He presents a succinct overview of his final synthesis of the kinds of constraints found in natural languages in *A Theory of Language and Information* (Harris 1991:336ff.).

Describing a semantics of a computer user interface begins with recording the events that occur during the interaction of a user and a software application. A user issues commands by means of keystrokes, mouse clicks, gestures, and sounds. For example, the user moves the mouse to the print icon and clicks the left mouse button. The application responds by relocating the mouse pointer and changing the appearance of the icon. Then a printed version of the document being edited appears at the printer. There is extensive research on natural-language interface design (Schneiderman 1998) and on applying linguistics in the tradition of Saussure and Hjelmslev to the study of user interfaces in working with tools (Andersen 1997).

Events that occur during a session of a user interacting with a computer are observable, including the events that produce the effects of commands. We can record these events as easily as linguists record speech events. We can then study the distribution of various features in these events. A distributional semantics describes the distribution of a bundle of features in the context of sequences of events. Zellig Harris focused on linguistic context.[2] He was an empiricist; he wanted to be able base his claims on what he could observe; he could observe linguistic events; he could not observe, for example, mental events very reliably. However, he did feel it was possible to go beyond what are usually considered linguistic events, as long as we stick to events we can observe and describe reliably.[3] I argue in the current chapter that distribution-

---

2. I encountered the view that meaning was best studied in terms of linguistic environment early and often as a graduate student at the University of Pennsylvania in the mid 1960s in seminars with Zellig Harris and Henry Hiż. This view is central to Harris's theory of information in *A Theory of Language and Information* (Harris 1991:321ff.). He takes the general position that ". . . for each word, what the learner or analyzer of a language does is not to think deeply about what the word 'really means' but to see how it is used, keeping in mind the fact that the various life-situations in which it is used are largely reflected in the various word combinations in which it occurs." (Harris 1991:332).

3. Zellig Harris held this view early (Harris 1951: 375):

> There are possible correlations between the descriptive system of a language and investigations in other disciplines. The whole system or features of it may correlate with features of . . . the relation of speech to other human actions. . . .The present operations of descriptive linguistics as most narrowly understood make a methodological whole,

al methods can guide research on and applications to user interface design.

In the following I will try to provide background for answering the following questions. What kinds of structures does a graphical user interface exhibit? How do they compare to the structures we find in natural languages? How can we define reference, paraphrase, and operator? How can a distributional semantics be used to guide the design of natural-language interfaces between humans and computers?[4]

Key ideas behind graphical user interfaces

In the 1970s researchers at Xerox PARC initiated the development of the principles behind today's graphical user interfaces. The following principles are especially relevant to the present chapter.

> and cannot by themselves yield these added results, although they can serve the further investigations which will obtain them.

Later, in *Language and Information* Harris warns us, but discusses this kind of correspondence as a referring relation which holds between structures of language and structures of research activity (Harris 1988: 84):

> There is no basis here for any general claim that language mirrors the world it talks about, or that the structure of language and information corresponds to the structure of the world. However, we can see in the languages of science that their classes of entities and relations are distinguished vis-à-vis each other comparably to the entities and relations of the science itself. . . . And if in immunological events objects that are antigens have a different standing than objects that are antibodies, with only particular classes of events holding between them, so in the immunology language there is a class of antigen words and a class of antibody words, with particular classes of operators or constructions occurring between them.

In *A Theory of Language and Information* he says the following (Harris 1991: 348): "In the real world, properties or events 'co-occur' with the objects which participate in them. This is reflected in the basic co-occurrence types of language: the dependence of an operator on its arguments." He goes on to say (Harris 1991: 354):

> What we have in language is, first, a correspondence between the proper property-object co-occurrence in perceived nature and the operator-argument non-zero probability in language; and, second, a correspondence between the relative likelihoods of property-object associations and operator-argument co-occurrences.

4. For a nice overview of relevant ideas from Zellig Harris, see the review of Harris (1988) in (Nevin 1988). Of course, there is no substitute for reading Harris himself. *Language and Information* (Harris 1988) is a very readable introduction to these ideas.

The elements of a good user interface should be visible. Command buttons, such as the bold button in Microsoft Word®, and elements of the document, such as segments of text being edited, should be laid out on the two dimensional screen of a computer monitor for the user to choose by pointing. This contrasts with command language user interfaces, which consist of a set of words or abbreviations that the user at best remembers and at worst must guess.

Interactions between user and user interface should be concrete. For example, the user should be able to point to elements on the screen in a familiar way. A touch sensitive screen would be the most familiar, but the typical interface device for pointing is a mouse which the user moves on a mouse pad, while the operating environment moves a pointer on the screen in a parallel manner. Many interactions are analogous to physical activities, for example, dragging an element from one location to another.

A good user interface should provide a lot of feedback. Some of this feedback is necessary for the user to carry out the physical actions accurately, such as tracking the movement of the mouse on the mouse pad with a pointer on the screen. Other feedback confirms the accomplishment of a higher level task, such as thickening the display of the letters of a word that the user has made bold.

There are additional important principles, but the preceding are the ones that figure prominently in the present chapter. For a detailed discussion of the issues in designing a graphical user interface see the chapter on direct manipulation and virtual environments in (Schneiderman 1998: 185 ff.).

Data and notation

We need to collect samples of humans interacting with a software application. I use Microsoft Word® as the application from which I draw my samples. I must warn the reader that choosing a word processor becomes slightly problematic when I discuss natural-language interfaces, because words will figure both in the interface language and in the objects being manipulated. My arguments apply equally well to an image editor, but I have chosen a word processor, because it is the most familiar of computer applications.

It is relatively natural to describe the sequence of events in texts of natural languages with a linear notation, since these events are largely linear. It may not be as obvious that a linear notation is also useful for describing

events in a graphical user interface, since the objects of the interface are displayed in two dimensions and persist over the time of the events. For example, the bold button in Microsoft Word® is visible during the whole of an event in which the user bolds a segment of text, as is the segment of text being bolded. And both the button and the text segment undergo visible alteration. We might be inclined to describe the events in parallel threads. There are two-dimensional notations for describing events in this way. However, I will use a linear notation. The interactions are largely linear, as the user directs attention and action to one object after another. The persistence in the display of elements can be accounted for by assuming the display stays the same, unless otherwise indicated by the occurrence of an event that adds or removes the element or some of its features. For example, the reverse-video feature assigned to a segment of text, when it is selected, is always removed when the user clicks somewhere else in the document. There is also a systematic relationship between the event changing the appearance of an object and the subsequent persistent display of the object, for example, when the application bolds a text segment, it will continue to be displayed as bold, until another event changes that. If there is any question, the display event can be repeated in the description. Consider the following example of a user bolding a text segment.

(1) *u:dragOver:Zellig*$_1$ *u:click:btnBold a:bold:Zellig*$_1$
a:display:bold:Zellig$_1$
user-dragover-Zellig-1 user-click-bold-button application-display-bold-Zellig-1
"The user drags over an occurrence of the text segment *Zellig* and then clicks the bold button. Then the application bolds that occurrence of *Zellig*."

The *Zellig*$_1$ in the description refers to a collection of visible features of a segment of text. I will use English notation that resembles the text segment, for example, *Zellig*$_1$. I use subscripts to emphasize that these are specific occurrences, that is, segments with specific locations in the text being edited. I will analyze this example in more detail in what follows.

I have indicated the feedback events by prefixing them with *a*. User controlled sub-events are prefixed with *u*. Including both the user events and application events interleaved in a hierarchy of events is useful for the following reasons. (1) Not all events in graphical user interfaces are summarized uniquely by application events. For example, when a user types a password,

the application typically displays a sequence of asterisks. (2) When we consider allowing the user to speak a natural language to command an application, the elements of the user's speech events may not be uniquely summarized by intermediate feedback events. We ultimately desire a smooth melding of a graphical and a natural-language interface and would like to describe to what extent natural-language utterances utilize the same feedback events as mouse or keyboard activities. (3) It is often important to know the relation between a user event and its feedback for reasons outside of purely syntactic patterns. For example, efficiency affects a user's choice of action, because of the physical locations of the keyboard and mouse. If his hand is already on the mouse, then a subsequent mouse action is efficient. However, if his hands are on the keyboard, he might prefer a keyboard command. (4) Feedback events frequently indicate structural boundaries.

Using a hierarchy of definitions, one can hide lower-level relationships between user events and application events, as appropriate for making generalizations. Events may contain sub-events. The notation allows for definitions. For example, we can define *u:click:btnBold* as follows.[5]

(2) *u:movePointer:arrow:btnBold a:raise:btnBold u:mouseDown:keyLM a:recess:btnBold u:mouseUp:keyLM a:display:recessed:btnBold*
user-move-mouse-pointer-bold-button user-press-down-left-mouse-key app-recess-bold-button user-let-up-left-mouse-key app-display-recessed-bold-button
"The user moves the arrow pointer over the bold button, raising the button, and presses the left mouse key down. At that point the application recesses the bold button. Then the user lets the left mouse key up. The application keeps the bold button recessed."

While employing a notation that interleaves user events and application events, it is important to get clear about how to use distributional methods, because the application feedback events are predictable, at least in a larger domain. I will describe the distribution of feedback events carefully in the section below on feedback and domain of substitution.

5. One could also click on the bold button with the right mouse button, but that has an entirely different effect and I will not consider right clicking at all in this chapter, except to presume that the *keyLM* is distinguished from *keyRM*: the right mouse key.

Establishing the basic elements

In a distributional analysis we must segment the flow of events into basic elements, which are distributed at various locations in that flow. This not a trivial task and the first step of segmenting may be arbitrary, if necessary (Harris 1954:787). Many of the events that take place during interaction between user and application in a graphical user interface are physical or analogous to physical event sequences. When a user applies a certain force to the mouse on the mouse pad, the mouse moves in a certain direction with a certain speed. At the same time a mouse pointer moves across the computer monitor in a certain direction with a certain speed. The finest level of elements I can imagine using is illustrated in the following representation of moving the mouse.

(3) *u:moveMouse:padVec1 a:movePointer:IBar:winVec1 . . . u:moveMouse:padVecN a:movePointer:arrow:btnBold*
user-move-mouse-pad-vector-1 app-move-ibar-pointer-to-window-vector-1 . . . user-move-mouse-pad-vector-N app-move-arrow-pointer-place-bold-button
"The user moves the mouse in a certain direction, speed, and distance on the mouse pad and the application moves the display of the mouse pointer on the monitor to a new location. At first the pointer has the shape of an I bar. This continues. Finally the user again moves the mouse in a certain direction, speed, and distance on the mouse pad and the application moves the display of the mouse pointer, now in the shape of an arrow, to the location of the bold button."

We need to join any elements that always occur together into a single element. Harris refers to this step as removing complete dependence (Harris 1954:788). For example, we should join movements of the mouse on the mouse pad and movements of the pointer on the screen into a single element.

Another of the basic distributional methods is grouping elements which occur as alternatives in the same environments into a set of elements which are so similar to each other that their choice is entirely free. This situation is termed free variation (Harris 1954:788). In example (3) above there is a large number of paths that that a mouse and mouse pointer could traverse and still end at the bold button. The user does not distinguish these paths from each other and they have no difference in effect on other events in the sequence of

events. I will define the event class *u:movePointer:shape:vec* to cover both the user events and application feedback in sequences like those in (3). Free variation is not a necessity of using a mouse. Some interfaces, e.g., the Opera® Web browser, use mouse gestures, in which the shape and direction of the paths effect such commands as reverting to a previous page: left moving path, or reloading the current page: up and then down path. Harris restricts this use of free variation to sound and excludes it from applying to meaning.[6] The relation between the level of representation of sound in natural languages and the levels of representation of events in a graphical user interface is not very close. So, we cannot argue by analogy, even if inclined to do so.

I will not take the space in this chapter to justify every basic element of representations used. The reader who is familiar with Microsoft Word® will likely be able to generate the necessary examples.

Grouping and classification of sequences of events

Once the elements of the representation are established, a method of substitution can be used to group event (element) sequences and assign them to the same syntactic class (Harris 1954:789, Harris 1946). For example, *young boy* can substitute for *boy* or *child* in many sentences and produce an acceptable sentence. We can find many examples of such substitutions in the event sequences in Microsoft Word®. A detailed description of an event of a user bolding a segment of text provides a starting point for our substitutions. I will ignore the details of shift or control clicking with the mouse. I have added brackets to the following representation, in order to assist in relating it to the discussion that follows.

(4) *[a:display:normal:Zellig$_1$]*
[u:movePointer:Ibar:vec$_{beginZellig1}$ u:mouseDown:keyLM
a:display:steady:cursor:loc$_{beginZellig1}$ u:movePointer:Ibar:vec$_{endZellig1}$
a:reverse:Zellig$_1$ u:mouseUp:keyLM a:display:reversed:Zellig$_1$]
[a:display:flat:btnBold u:movePointer:arrow:btnBold a:raise:btnBold
u:mouseDown:keyLM a:recess:btnBold u:mouseUp:keyLM

6. Harris also uses the concept of free variation when discussing paraphrastic transformations in *Report and Paraphrase* (Harris 1969:619). However, this use of the concept applies only after a great deal of the structure of a language has been discovered.

a:display:recessed:btnBold]
[a:bold:Zellig$_1$]
app-display-normal-style-Zellig-1 user-move-Ibar-pointer-vec-begin-Zellig-1 u:press-down-left-mouse-key app-display-non-blinking-cursor-loc-begin-Zellig-1 user-move-Ibar-pointer-vec-end-Zellig-1 app-reverse-Zellig-1 user-release-up-left-mouse-key app-display-reversed-Zellig-1 app-display-flat-bold-button user-move-arrow-pointer-bold-button app-raise-bold-button user-press-down-left-mouse-key app-recess-bold-button user-release-up-left-mouse-button app-display-recessed-bold-button app-bold-Zellig-1
"The application shows *Zellig*$_1$ in normal style. The user moves the Ibar pointer to the beginning of *Zellig*$_1$ and presses down the left mouse key. Keeping the left mouse key down, the user moves the Ibar pointer so that *Zellig*$_1$ is in reverse-video. The user then releases the left mouse key and moves the arrow pointer to the bold button, which changes from flat to raised. The user then clicks on the left mouse button and the application recesses the bold button and bolds *Zellig*$_1$."

The following sequence of events, used in example (4) above, is one way of selecting a segment of text.

(5) *u:movePointer:Ibar:vec*$_{beginZellig_1}$ *u:mouseDown:keyLM a:display:steady:cursor:loc*$_{beginZellig_1}$ *u:movePointer:Ibar:vec*$_{endZellig_1}$ *a:reverse:Zellig*$_1$ *u:mouseUp:keyLM a:display:reversed:Zellig*$_1$
user-move-Ibar-pointer-vec-begin-Zellig-1 u:press-down-left-mouse-key app-display-non-blinking-cursor-loc-begin-Zellig-1 user-move-Ibar-pointer-vec-end-Zellig-1 app-display-reversed-Zellig-1 user-release-up-left-mouse-key app-display-reversed-Zellig-1
"The user moves the Ibar pointer to the beginning of *Zellig*$_1$ and presses down the left mouse key. Keeping the left mouse key down, the user moves the Ibar pointer so that *Zellig*$_1$ is in reverse-video. The user then releases the left mouse key."

There are other ways of selecting text. They justify us grouping (5) as an event sequence, which I will define as *u:dragOver:text*. Some ways use keyboard events exclusively and others a combination of mouse and keyboard events. One alternative is the following.

(6) *a:display:blinking:cursor:loc*$_{beginZellig_1}$ *u:keyBdDown:control-shift:keyRtArrow a:reverse:Zellig*$_1$ *u:keyBdUp:control:shift:keyRtArrow a:display:reversed:Zellig*$_1$
app-display-blinking-cursor-loc-Zellig-1 user-press-down-control-shift-right-arrow-key app-reverse-Zellig-1 user-let-up-control-shift-right-arrow-key key app-display-reversed-Zellig-1
"The cursor is displayed at the beginning of an occurrence of *Zellig* in the text. The user holds down the control and shift keys and presses the right arrow key, extending reversed text to cover the occurrence of *Zellig* and then releases the keys."

Yet another way to bold text is illustrated in the following.

(7) *u:movePointer:Ibar:vec*$_{inZellig_1}$ *u:mouseDown:keyLM a:display:steady:cursor:loc*$_{inZellig_1}$ *u:mouseUp:keyLM a:display:blinking:cursor:loc*$_{inZellig_1}$ *u:keyBdDown:control:keyB a:bold:Zellig*$_1$

Table 1. Groupings and classifications of event sequences

Selection	Command	Feedback
u:movePointer:iBar:vec$_{inZ1}$ *u:mouseDown:keyLM a:display:steady:cursor:loc*$_{inZ1}$ *u:mouseUp:keyLM a:display:blinking:cursor:loc*$_{inZ1}$	*u:keyBdDown: control:keyB*	*a:bold:Z*$_1$
ditto	*u:click:btnBold*	*a:bold:Z*$_1$
ditto	*u:click:btnItalic*	*a:italicize:Z*$_1$
u:movePointer:iBar:vec$_{beginZ1}$ *u:mouseDown:keyLM a:display:steady:cursor:loc*$_{beginZ_1}$ *u:movePointer:iBar:vec*$_{endZ_1}$*a:reverse:Z*$_1$ *u:mouseUp:keyLM a:display:reversed:Z*$_1$	*u:click:btnBold*	*a:bold:Z*$_1$
ditto	*u:click:btnCut*	*a:cut:Z*$_1$
a:display:blinking:cursor:loc$_{beginZ_1}$ *u:keyBdDown: control:shift:keyRtArrow a:reverse:Z*$_1$ *u:keyBdUp:shift:keyRtArrow a:display:reversed:Z*$_1$	*u:click:btnBold*	*a:bold:Z*$_1$
ditto	*u:click:btnCut*	*a:cut:Z*$_1$

> user-move-ibar-pointer-vec-in-Zellig-1 user-press-down-left-mouse-key app-display-steady-cursor-loc-in-Zellig-1 user-let-up-left-mouse-key app-display-blinking-cursor-loc-in-Zellig-1 u:keyBdDown:control:keyB a:bold:Zellig$_1$.
> "The user moves the Ibar pointer to a location internal to an occurrence of the word *Zellig* and then types the control-B key. The application bolds the word with the cursor."

Table 1 collects a number of samples of event sequences that show the distribution of substitutions that group and classify sub-sequences. The columns indicate substitutions and the rows indicate sequence. The *u:click:btnBold* event was defined in (2). Other button clicks are defined similarly. *Zellig* is replaced by *Z* throughout to save space. The event sequences in the first column can be classified together as *u:select:text* events.

Dependence constraints and operator structure

Harris distinguishes a specific kind of constraint among elements of a sentence which he calls a dependence constraint and he uses dependence constraints to determine the operator structure of a language.[7] A word is dependent on a word class or sequence of word classes if, whenever that word occurs, it requires some sequence of words belonging to those classes to occur together with it. For example, the verb *eat* requires two nouns to surround it. Apparent exceptions,

---

7. Harris relied on the concept of dependence throughout most of his research beginning early (Harris 1954, 1970). In later work (Harris 1988, 1991), the term *constraint* replaced *dependence* in many contexts. He provides an overview of the various kinds of constraints in *A Theory of Language and Information* (Harris 1991:336ff.). He defines the specific constraint of operator dependence as follows (Harris 1991:55)

> If A is a simple word, and b, . . . , e is an ordered set of classes of simple words, then A is said to depend on (or, require) b, . . ., e if and only if for every sentence in the base, if A is in the sentence, then there occurs in the sentence a sequence of simple words B . . . E which are respectively members of b, . . . , e. Within the given sentence, A may then be said to depend on the word sequence B . . . E. If in the given sentence there is no other word G such that A depends on G and G depends on the given occurrence of B . . . E, then A depends immediately on that occurrence of B . . . E. A is then called the operator on that B . . . E, which in turn is called the argument of A in the sentence; B may be called the first argument, and so on.'

such as *sheep eat slowly* and *the grass was eaten* are transformational reductions of the base sentences and may not always exhibit the restriction overtly. Harris goes on to define the notion of operator by factoring the dependence relation. Some verbs depend on other verbs which depend on nouns (Harris 1991:55). For example, *know* may appear to be dependent on three nouns and a verb, as in *I know sheep eat grass.* In fact, *know* is only dependent on one noun and a verb class of which, e.g., *eat* is a member. So, *eat* is an argument of *know* and it in turn has its own arguments. Some operators require zero arguments. In English these are nouns. These notions of dependence and operator are abstract enough to permit applying them to the analysis of a graphical user interface. The following analysis is not based on the possible analogy to natural language, but rather the application of a particular distributional method.

Some events are easily seen to be arguments not dependent on other events. The display of a blinking cursor is not dependent on other events related to the user's editing process, neither is the default display of the bold button, the segments of text, nor locations in the visible text. These all qualify as operators with zero arguments.

We can find examples of operator dependence in the events from the preceding discussion. The *u:click:btnBold* event is preceded by an event of the *u:select:text* class. Thus, we establish a dependence of *u:click:btnBold* events on the *u:select:text* class of events. *u:click:btnBold* is the operator and an event of the *u:select:text* class or some part of it is its argument. The events of the selection process can be analyzed into operator and argument by taking the zero argument operator: *a:display:normal:Zellig*$_1$ as the argument and the remaining events that work to transform its appearance as the operator. If we analyze the selection process in this way, then we would say that *u:click:btnBold* operates on the selection operator, which in turn operates on *a:display:normal:Zellig*$_1$.

Using these definitions, a whole event of bolding a word is represented as follows.

(8) *u:dragOver:Zellig*$_1$ *u:click:btnBold a:bold:Zellig*$_1$
user-select-Zellig-segment-1 user-click-bold-button application-bold-Zellig-1
"The user selects an occurrence of the text segment *Zellig* and clicks the bold button. Then the application bolds that occurrence of *Zellig*."

We can define *u:bold:Zellig*$_1$ as (8). It can occur almost unconstrained among events of the same level, as in the following sequence.

(9) *u:bold:Zellig*$_1$ *u:delete:Harris*$_1$ *u:save:document*
user-bold-Zellig-1 user-delete-Harris-1 user-save-document
"The user bolds an occurrence of *Zellig*, then deletes an occurrence of *Harris*, and finally save the document."

The only constraints are those that make it impossible to carry out the event, such as bolding *Zellig*$_1$ after it has been deleted or constraints imposed by extending the domain to include events during the printing of the document. Table 2 combines the preceding examples, in order to provide an overview of the structure of an event sequence. The rows follow temporal sequence reading down. The columns from left to right show increasing hierarchy of definition: An event is defined by the events to its immediate left and up, excepting those events which are already part of an earlier definition. Persistent events, such as recess or reverse events, may appear multiple places. *Z* is short for *Zellig*, *Ptr* for *Pointer*.

When defining a higher-level event, some features of the sub-events may be promoted to the higher-level event representation, e.g., *btnBold* in *u:raise:btnBold* or *Zellig*$_1$ in *u:dragOver:Zellig*$_1$. This is an abstraction from the purely contiguous and sequential layout. It is useful in relating the defined events to events outside the definition. A requirement for making such an abstraction is that the abstracted element be everywhere simultaneously replaceable by other elements. In these cases, *btnBold* by *btnCut*, *btnItalic*, etc. and *Zellig*$_1$ by *Harris*$_2$, etc.

Table 2. A sample segmented and grouped event sequence

	a:display:normal:Z_1		
	u:movePtr:iBar:vec$_{beginZ_1}$		
	u:mouseDown:keyLM		
	a:display:steady:cursor:loc$_{beginZ_1}$		
	u:movePtr:iBar:vec$_{endZ_1}$		
	a:reverse:Z_1		
	u:MouseUp:keyLM		
	a:display:reversed:Z_1	*u:dragOver:*Z_1	
a:display:flat:btnBold			
u:movePtr:arrow:btnBold			
a:raise:btnBold	*u:raise:btnBold*		
	u:mouseDown:keyLM		
	a:recess:btnBold		
	u:mouseUp:keyLM		
	a:display:recessed:btnBold	*u:click:btnBold*	
		a:bold:Z_1	*u:bold:*Z_1

Feedback and domain of substitution

Harris is careful to point out that patterns discovered with the method of substitution are relative to the domain of the substitution (Harris 1954: 789). In the analysis of natural languages the sentence is frequently the domain. We substitute one word or phrase for another word or phrase and consider whether the result is still an acceptable sentence, that is, can ever occur as a sentence of the language in a larger discourse. If we take as domain the entire discourse, then substitutions are much more limited. Substituting *cow* for *sheep* in an entire discourse will likely produce an unacceptable, i.e., non-occurring, discourse, since the meanings of the words *cow* and *sheep* are involved. We must pay close attention to the domain of substitution in the analysis of the events of a graphical user interface, particularly when analyzing feedback events.

In the preceding analysis of operator structure in user interface event sequences we limited the domain of substitution to user commands and simply alternated application feedback as needed. That is, we did not really treat the application feedback as part of the domain of substitution. If we extend the domain of substitution to include a whole editing session, it becomes clear that there are many constraints imposed by the larger domain. For example, the persistent display of a text segment in bold style precludes substituting *u:click:btnItalic* for *u:click:btnBold*, without a corresponding alternation in the environment of display of italic style with display of bold style. The feedback events, such as *a:bold:text* are accompanied by display events, such as *a:display:bold:text*, *a:display:bold+italic:text*, etc. These display events persist and become zero level operators for further editing, like *a:display:normal:text*. Thus, the display events represent compactly the constraints of the larger environment.

Like an operator, a feedback event exhibits dependence constraints on the events which lead up to it, but its dependence reaches down to other operators and their arguments alike. For example, *a:bold:text* depends on both *u:select:text* and *u:click:btnBold*. This does not fit the definition of an operator. It has a different kind of structure.[8] I will discuss this structure further in the section below, especially in Operator Structure and Denotational Semantics.

---

8. I note that the pattern for feedback is not an artifact of my choice to represent both user events and application events. We could, for example, describe the event of bolding entirely in terms of application events as follows. *a:reverse:Zellig*$_1$ *a:recess:btnBold a:bold:Zellig*$_1$. We

Selectional constraints

In addition to the operator dependence constraints, natural languages have constraints on how likely specific words are to occur in syntactic structures. These selectional constraints complement the operator dependence constraints, because the dependence constraints are stated on argument classes and the selectional constraints further constrain the arguments of an operator in terms of specific words (Harris 1991:63). Let us consider whether constraints on the kinds of objects in a document that can be made bold might qualify as selectional constraints. For example, a Microsoft Word® document can contain a rectangle figure in addition to text. If the user selects this figure, the bold button is grayed out. This feedback is indicating that clicking on the bold button would not bold the rectangle. Is this an example of a selectional constraint, similar to the low selectional likelihood of *ideas*, *eat*, and *slowly* in the construction *ideas eat slowly*? Or does it reflect an operator dependence constraint? The likelihood of an operator occurring with any particular word in its argument classes never reaches 'permanent zero' (Harris 1991:63). There must be some environment in which the words can take on associations that allow *ideas eat slowly* to occur. In contrast, the likelihood of bolding a rectangle seems to be permanent zero, certainly within the version of Microsoft Word® I am currently using. Having selectional constraints may be a sharp difference between natural languages and graphical user interfaces. This would not be surprising. Natural languages evolve constantly and must allow words to be used flexibly and even creatively to support communication in varied and overlapping environments. Graphical user interfaces have been constructed by programmers with tools, both of which prefer discrete choices. Of course, Microsoft Word® might evolve to allow rectangles to be made bold by pressing the bold button. If it is still only marginally likely to occur, because users do not think of bolding a rectangle that often, are we now dealing with a selectional constraint? The key requirement here is that selectional constraints are by definition open-ended. When we add natural language to an interface, the user will naturally bring selectional constraints along with the words of his language. At one extreme the user interface could

---

have the same constraints here. *a:bold:Zellig*$_1$ requires *a:recess:btnBold* and *a:reverse:Zellig*$_1$. The user-application distinction is not a source of difficulty. In fact, having user events present makes the feedback events more intelligible. This adds to the reasons for including both given above in the section on Data and Notation.

force categorical choices on the user and abolish the selectional constraints. At the other extreme the user interface (and application) could adapt to selectional constraints found during its use. It is a bit scary, but imagine Microsoft Word® doing its best to figure out what action most resembles bolding text, when a user selects a rectangle and clicks the bold button.

Meaning features

Harris describes a number of aspects of meaning. Words and sentences have extensional meaning, which is their relation to the external world. He speculates that this is the first form of meaning to appear in the evolution of language (Harris 1991:323). Words also have a meaning that arises by virtue of their co-occurrence with other words in discourse (Harris 1991:325). This is the view that the meaning of a word is the set of environments in which it occurs. Harris feels describing co-occurrence is where linguistics will be most successful in the task of formulating constraints (Harris 1991:42). Operator-argument dependence accounts for a structural part of meaning (Harris 1991: 22–23, 62ff.). He views what remains of meaning as best described through selectional constraints (Harris 1991:62).

Selectional constraints are likelihood constraints, i.e., the likelihood of specific words occurring in a sentence. Selectional constraints are accounted for in part in terms of operator-argument co-occurrences, but not entirely, as illustrated by the fact that *the house melted* has a low likelihood, while *the house made of snow melted* has a high likelihood.[9] Harris claims that similarity of selection is closely related to similarity of meaning. This is due in part to the fact that the combinations of words and structures a word occurs in is part of the meaning of a word (Harris 1991:65).

---

9. While focusing on likelihood or selectional constraints among operator and arguments, Harris notes the impact of the larger environment (Harris 1991:64):

> The likelihoods are stated in respect to the argument word [of an operator], as a further restriction on what the dependence admits. While virtually everything in language structure will be found to be encased in the relation between an operator and its argument, the likelihood of an operator can be affected not only by its immediate argument but also by some distinguished word further down in the dependence chain.

It can also be affected by words elsewhere in discourse, because of co-reference (Harris 1991:337).

Let us take a careful look at meaning in terms of distribution. How can we identify the meaning of an element or combination of elements? I will be less ambitious and simply try to identify features of the environment which reflect the meaning of an element or combination of elements.[10] I will proceed by identifying a meaning difference between two elements of the same class (in a given environment). Let C[ ] be a discourse environment in which A can occur acceptably. That is, C[A] is an acceptable discourse, i.e., extended utterance, text, or sequence of events. Many of the features of the environment C[ ] are meaning features of A — the question is which constrain A or are constrained by A. Let B be another element of the same class as A. Two possibilities arise. C[B] is an acceptable discourse or it is not.[11] If C[B] is an acceptable discourse, then A and B share all meaning features in the environment C [ ]. If C[B] is not an acceptable discourse, then there is some meaning feature of A in C[ ] which is not a meaning feature of B. If we can delete a feature F of C[ ] to form an environment C^{-F}[ ] in which both A and B occur acceptably, then we have located a feature F of C which is a meaning feature of A and not of B. Equivalently we could start with an environment C[ ] as the common environment and C^{+F}[ ] as the environment in which A is acceptable and B unacceptable. Another way to locate a meaning feature is to replace a feature F with a feature G. If C^{+F}[A] is acceptable and C^{+F}[B] is not, but C^{+G}[B] is, then F is a meaning feature of A. Consider the following examples.

(10) John was wounded in the leg. So, he was evacuated from the front to a hospital in the Philippines. He returned to fight two months later.

(11) John was wounded in the arm. So, he was evacuated from the front to a hospital in the Philippines. He returned to fight two months later.

---

10. See (Harris 1991:325): "Once we see that the meaning of a word in a particular occurrence depends on its environment, the categorization of meaning-ranges can be replaced by a categorization of the environing words. Note that when we judge the meaning of a word-occurrence by what can replace it in the text, we are really judging the meaning of the textual environment."

11. I do not grade acceptability of discourse by degrees. A discourse is either acceptable or not. I suggest that the grading of the acceptability of co-occurrence, i.e., the likelihood of co-occurrence, is directly related to how difficult it is to come up with a discourse environment that is acceptable. That is, the binary choice of acceptable versus unacceptable for (grammatical) discourse together with a measure of the difficulty of finding an acceptable discourse provides a foundation for the concept of likelihood.

(12) John was wounded in the leg; so, **walk**ing caused him a great deal of pain. So, he was evacuated from the front to a hospital in the Philippines. He returned to fight two months later.

(13) John was wounded in the arm; so, **walk**ing caused him a great deal of pain. So, he was evacuated from the front to a hospital in the Philippines. He returned to fight two months later.

(14) John was wounded in the arm; so, **crawl**ing caused him a great deal of pain. So, he was evacuated from the front to a hospital in the Philippines. He returned to fight two months later.

There are many environments, such as (10) and (11), where *leg* and *arm* can be interchanged with no alternation in the environment. *Arm* and *leg* share all meaning features in this environment, since nothing in the environment serves to distinguish them. However, there are also environments, such as (12) and (13), which contain key items that work against the acceptability of the replacement. Dropping the first *so*-clause from (12) and (13) gives us (10) and (11). Replacing *walk* in the unacceptable (13) by *crawl* yields (14), which is acceptable. We conclude that the occurrence of *walk* in the first *so*-clause is a meaning feature of *leg* in the cited environment. Comparing more environments involving the *so*-clause and *leg* and *wounded in the leg* can refine the attribution of the meaning feature.

A meaning feature is a distinguished element of an environment. It is distinguished by comparisons such as (10)–(14). We can generalize the meaning features by noting that there is a correspondence between the slot of the environment and the distinguished element. In the examples the slot is indicated by underline and the distinguished element by bold.

(15) John_i was wounded in the N; so, **V**ing caused him a great deal of pain.

The following relation lays out the correspondence.

(16)

N	*V*
leg	*walk*
leg	*run*
leg	*jump*
leg	*crawl*
arm	*carry*
arm	*crawl*
. . .	. . .

The form (15) and relation (16) capture a more general meaning feature, i.e., one describing a whole set of meaning features. I say more about such forms and relations in graphical user interfaces in the sections below.

An ideal situation would be to find a set of discourses that can be used as a minimal basis to distinguish the set of all the meaning features for a language or sub-language.[12] Identifying meaning features in natural languages is a lot of work, perhaps only feasible for limited domains, such as science sub-languages (Harris 1991: 272 ff.).

Identifying meaning features in a graphical user interface is a lot easier and it should also be easier in a well-designed natural-language interface. Consider the example (1) above. Replacing *btnBold* by *btnItalic* produces a non-occurring sequence of events. Thus, there are some meaning features of *btnBold* which are not shared by *btnItalic* in the environment in (1). Concomitant replacement of *bold* in *a:bold:Zellig1* with *italicize* and in *a:display:bold:Zellig*$_1$ by *italic* yields an event sequence which does occur. So, *bold* is a meaning feature of *btnBold* in that environment and *italicize* and *italic* in the environments in question are meaning features of *btnItalic*. Providing a basis set is important in the process of designing a user interface, a point explained in a bit more detail in the section below: Requirements for Natural-language Interfaces.

Paraphrase

Linguists have struggled to define a notion of paraphrase or synonymy. A strong definition is that one item is a paraphrase of another if each item can be substituted for the other in every environment. This definition identifies very few paraphrases, since there always seem to be some features of the environment that reject or reduce the likelihood of the substitution. However, in particular environments and especially in sublanguages it is possible to find synonymy (Harris 1991: 282).

Some of these features involve systematic co-reference patterns in natural languages (Smaby 1971, 1981). Consider the following pair of examples from English.

---

12. Harris discusses characterizing words by their selection sets, in the context of the operator-argument relation. "Characterizing words by their selection allows for considering the kind and degree of overlap, inclusion, and difference between words in respect to their selection sets '' (Harris 1991: 330).

(17) John was in the room. Bill was also in the room. He had arrived late.

(18) John and Bill were in the room. He had arrived late.

The first example in the pair seems to convey a clear idea. The second is difficult to understand. These examples also serve to point out an important function of paraphrase. Speakers find themselves in a linguistic environment at each point of their speech, which requires them to choose variants that will convey their ideas. The speaker of the second example would need to use *Bill* instead of *he*. Another function of paraphrase is to avoid repeating features, as seen, for example, in pro-words. These two functions clearly interact. These systematic interactions suggest a weaker definition of paraphrase that allows certain elements of the environment to vary, while requiring others to remain fixed. Graphical user interfaces show some patterns of paraphrase, but do not have the intricate co-reference patterns of English, likely due to the briefness and independence of event sequence episodes and to the ready availability of zero-level operators through persistent display of objects in the interface. The patterns of English do become relevant, when we discuss natural-language interfaces.

Once we have ferreted out the meaning features of an element, we can then use them to characterize paraphrase. A and B are paraphrases in a basis set, if and only if they have the same meaning features based on the discourses in that set. We can also formulate a definition of similarity of meaning relative to a subset of meaning features: A and B are similar relative to a set of meaning features S if and only if A and B both share all the meaning features in S.[13]

In a graphical user interface, feedback provides easily identifiable meaning features, such as the display of the bold feature on a text segment. This is not surprising, since the user is using the interface to command the application to perform some task. Even in natural-language discourse, a command or request given by one person typically results in observable events. Being observable, these events provide feedback. The feedback in such cases is not by design, but by inclination on the part of the hearer to carry out the command. In a graphical user interface the feedback is by design, inspired by the natural feedback of the physical operation, e.g., bolding of text on the printed page.

---

13. Harris refers to the idea of synonymy relative to an environment in (Harris 1991: 325): "Also, words which are very different in meaning may be almost synonymous in a particular environment, although each keeps its own meaning throughout: e.g., *the cells divide* and *the cells multiply*." Sets that characterize meaning features fit with Harris's idea of characterizing words by their selection sets. See footnote 12.

A simple example of paraphrase in a graphical user interface is found in the alternation of using the bold button or the control-b key, or of clicking on the format menu button, then selecting the font item and further selecting and confirming in the dialog box that appears. Any of those scenarios will end with the application displaying the text segment in bold style. The ways of bolding text are similar in meaning to each other relative to the meaning feature of the display of the text segment in bold. That is, they are all interchangeable in all environments in which they are followed by the display of the text segment in bold. They are in fact paraphrases. Any environment in which one occurs, the others can occur in. Constraints by these events on other meaning features, such as printing the text segment in bold, can be entirely accounted for through constraints from the display of the bolding of the text segment. That is, this meaning feature is sufficient to completely characterize the meaning of the other events of the bolding process.

Let us consider an example of paraphrase reminiscent of studies of natural language. A user can select an extent of text and then change its type face, size, style, etc., while it remains selected. In the following sequence of events a user changes a segment of text to bold italic in a long-winded way.

(19) *a:display:normal:Zellig*$_1$ *u:dragOver:Zellig*$_1$ *a:reverse:Zellig*$_1$
u:click:btnBold a:bold:Zellig$_1$ *u:click:wndDoc a:unreverse:Zellig*$_1$
a:display:bold:Zellig$_1$
u:dragOver:Zellig$_1$ *a:display:reversed:Zellig*$_1$ *u:click:btnItalic*
a:italicize:Zellig$_1$ *u:click:wndDoc a:display:italic+bold:Zellig*$_1$
"The user selects an occurrence of *Zellig* and pushes the bold button, which bolds the occurrence of *Zellig*. The user then deselects the occurrence of *Zellig* by left clicking in the document window. The user then selects the same text again and italicizes it in the same way."

The next sequence of events is somewhat shorter.

(20) *a:display:normal:Zellig*$_1$ *u:dragOver:Zellig*$_1$ *a:reverse:Zellig*$_1$
u:click:btnBold a:bold:Zellig$_1$ *u:click:btnItalic a:italicize:Zellig*$_1$
u:click:wndDoc a:unreverse:Zellig$_1$ *a:display:italic+bold:Zellig*$_1$
"The user selects an occurrence of *Zellig* and pushes the bold button, which bolds the occurrence of *Zellig*. The user then pushes the italicize button, which italicizes the occurrence of *Zellig*. Finally the user deselects the occurrrence of *Zellig* by left clicking in the document window."

Both sequences of events (19) and (20) end in *a:display:italic+bold:Zellig*$_1$. They have similar meaning relative to that meaning feature. Since that meaning feature accounts for all further constraints, they are in fact paraphrases. These paraphrases are reminiscent of topic-comment constructs or conjunction in natural languages.

Co-reference constraints

Reference and co-reference among features of event sequences are central to a semantics of a language and of a user interface. Harris argues that all referentials can be derived from cross-reference: when one occurrence of a word in a discourse refers to another occurrence of a word in the same discourse. Repetition of a word or collection of features is one of the key properties of cross-reference.[14] We have seen many examples of repeated elements above. I will use this repetition to describe the structure of a graphical user interface.

An editing session in a word processor contains many similar event sequences. For example, there are many different text segments which can be selected and made bold. If we compare these event sequences to each other, we find the same pattern of repetition. We can state this pattern as a generalization that helps us understand reference and co-reference in the interface.

(21) *u:dragOver:txt*$_i$ *u:click:btnBold a:bold:txt*$_i$
user-drags-over-text-segment-i user-clicks-bold-button app-bolds-text-segment-i
"The user drags the mouse pointer over text segment txt i. The application highlights that occurrence of txt i. The user then left clicks the bold button. Then the application bolds txt i."

The individual descriptions of the same text segment are replaced by a general description and a variable *i* to indicate that a description is constrained to be

14. Harris describes the capacity for cross-reference in a language as arising from three properties (Harris 1991: 128 ff.). The first of these is repetition. The second is having words that can state the relative location of occurrences of words in a discourse. The third is having words that state that two or more occurrences of words have the same referent. Continuity in the display of features of elements may fill a function similar to that in the last two properties in natural language.

repeated.[15] A variable indicates a constraint of co-reference. The constraint is very strong, due in part to our including the application events in the description of the event sequence, but not entirely due to that, since disturbing a co-reference configuration in a natural-language discourse can also make the discourse unacceptable. In a graphical user interface it is very simple, consisting in the identity of the description. The user is able to graphically delimit the text segment that the application modifies. Later, when we look at natural-language interfaces, the constraint will be less graphic. We can find other constraints in the example, if we compare scenarios in which the user bolds, italicizes, underlines, and colors the text segment.

(22) *u:dragOver:txt*$_i$ *u:click:btnItalic a:italicize:txt*$_i$

I will leave glosses off examples, when it is clear from the context or previous examples what they should be. Generalizing (21) and (22) we have the following description.

(23) *u:dragOver:txt*$_i$ *u:click:btnStyle*$_k$ *a:modifyStyle*$_k$*:txt*$_i$

The new notation in (23) is that the subscript k is used on two different generalized descriptions: *btnStyle* and *modifyStyle*. We are no longer dealing with the identity function and need to provide a constraint relation for style buttons.

(24)

btnStyle	modifyStyle
btnBold	*bold*
btnItalic	*italicize*
btnUnderline	*underline*

15. Harris's description of the relation of cross-reference to discourse analysis is similar to the one proposed here (Harris 1952: 329):

> One major example [of dependent occurrence] is that of pronouns. If the advertisement had read *You . . . will prefer it* instead of *You . . . will prefer X*—, we would at first regard *it* as a new element, to be placed in a new equivalence class. However, the occurrence of *it* is dependent on the occurrence of *X*—: if the preceding *X*— had contained the plural morpheme (*X-s*), the pronoun in this sentence would have been *them*. Other words of the *it* group, say *he* or *you*, will not occur here as long as *X*— occurs in the preceding sentence; but they could occur if certain other words were used in place of *X*—.

He also discusses co-reference constraints later in *A Theory of Language and Information* (Harris 1991: 336ff.). Henry Hiż emphasized co-reference as a key to structure in seminars in the mid nineteen sixties. See also (Hiż 1968).

Descriptions (23) and (24) combine to form the description of the scenarios in question.

How can we move from a concept of co-reference to a concept of reference? We typically feel that there is more to reference than just co-reference of linguistic items. There is some connection to the non-linguistic world. In usual English discourse we feel that there must be more than just the cross-reference patterns of the words *Zellig*, *he*, *him*, *his*. There is some connection to the person Zellig. Unfortunately, it is very hard to vary the world in all the ways needed to establish that connection using distributional methods. Just as with natural languages, we feel there is more to changing the style of a segment of text than the co-reference patterns of that segment in the process, but with computer user interfaces we are not so limited. If we extend the domain in the above scenarios to include the printing of hard copy of the document, we can observe co-reference with a set of printing features. We can choose to regard the effect on the printing as the real world and take the occurrence of the features in the printing of the text segment as an anchor and say that all the other occurrences of the features refer to that occurrence. Choosing a co-referent as the referent is valuable, if it improves our ability to describe the semantics of the user interface.

Operator structure and denotational semantics

Describing how the application of an operation to some object corresponds to the concatenation of user event sequences is reminiscent of denotational semantics. However, there are important differences between a distributional semantics and a denotational semantics.

Operator structure and co-reference

Users of graphical user interfaces quickly learn that there is a general structure to events that consists in selecting an object and then commanding the application to modify it. Distributional methods for determining classification and operator structure can capture such generalizations. First we can divide scenario (23) into three parts

(25) a. $u{:}dragOver{:}txt_i$
b. $u{:}click{:}btnStyle_k$
c. $a{:}modifyStyle_k{:}txt_i$

Part a is selection, part b command, part c modification. As summarized in Table 1 above, there are various ways to select and various ways to command the alteration of the selected text. These are instances of a class of event sequences called *u:select:text* and *u:commandText*, respectively. Assuming that *u:select:text* and *u:commandText* have been defined to include the co-referential indices, we can now generalize (25a–c) to (26).

(26) a. $u{:}select{:}txt_{i} \wedge u{:}commandText_{k}$
b. $a{:}modifyText_{k}{:}txt_{i}$

The caret in (26a) is required to show that $u{:}select{:}text_{i}$ is contiguous with $commandText_{k}$, since elements in the descriptions are not always contiguous. (26b) shows a relation that matches a text segment *i* and a command feature *k* with a modification of the text segment. If (26b) were to describe the process of printing hardcopy, it would not have to be contiguous with (26a), but some conditions would have to be added to the effect that certain other events did not occur to further modify i. (26a) is suggestive of N^V structure in English. This is not surprising since both exhibit what Harris calls operator structure. The nice thing about computer user interfaces is that we can connect structure of selection and command with structure of modification of text or at least the display of the text.This correspondence between one structuring operation and another structuring operation is at the center of the semantics of operator structure (Harris 1991:354).

Comparison with denotational semantics

A denotational semantics consists of a mapping of elements of a language to elements of a world. In simple formal languages a predicate symbol is mapped to a set of values and an argument symbol is mapped to a value, which may or may not be in the set. Concatenation of the predicate symbol and the argument symbol maps to a test for membership of the image of the argument symbol in the image of the predicate symbol. Language and world belong to disjoint universes. They are related only via this mapping and this mapping is a fixed formal definition.

Examples (23), (24), and (26) are suggestive of a denotational semantics, insofar as they involve relations between user events and syntactic constructions on user events to application events and application constructions. One important difference between a denotational semantics and the distributional semantics proposed here is that the distributional semantics does not limit meaning of an item to a single object or construction, but instead regards

various meaning features as parts of the item's meaning. A second difference is that it does not insist on a complete definition of an item's meaning. A third difference is it does not separate the linguistic domain from the ontological domain. While it may be tempting and useful to take the user events as the linguistic domain and the application events as the ontological domain, it is also useful to consider how user events are constrained by feedback events from the application. For example, the dialogue box provided as the result of the user choosing a menu item provides the user access to choices not available otherwise. We can also consider how user events are constrained by other user events.

Requirements for natural-language interfaces

I believe there is a close relation between the distributional methods I described above and the ways in which human users of computer software learn the structure of a user interface. As we design natural-language interfaces for computers, it is important to design them to be easily learnable. Harris argues that humans rely on distributional methods much like those linguists use, in order to learn their natural languages.[16] They are not provided with a precise preset definition in some scientific meta-language (Harris 1991:402). Human users will not learn a structure that is not there. Therefore, when we design natural-language interfaces, we need to create a large set of scenarios: a basis set that could function as the data that supports the structures we imagine the user using. We also need to be clear whether the selectional constraints on the words of a natural language are to be maintained and exploited or replaced by dependence constraints. In the software development process these scenarios establish requirements for the application (Rumbaugh 1991:170 ff.). We need to generate enough scenarios to justify operator structure and reference structure using distributional methods, when developing natural-language interfaces. In principle, requirements analysis precedes the design, which

16. ". . . [T]he position of the speakers is after all similar to that of the linguist. They have heard (and used) a great many utterances among which they perceive partial similarities: parts which occur in various combinations with each other. They produce new combinations of these along the lines of the ones they have heard. The formation of new utterances in the language is therefore based on the distributional relations [. . .] among the parts of the previously heard utterances." (Harris 1954:779).

precedes implementation and deployment of the application. In practice, software applications evolve by iterating this process. Our initial scenarios will likely be incomplete. However, we can apply distributional methods to analyze actual event sequences on prototypes and existing versions of the application, in preparation for succeeding versions.

I use the term natural-language interface to mean a computer user interface that incorporates structures found in natural languages, the more structures the more natural. I do not mean English or Russian language interface. Natural-language interfaces could be classified as pidgins or sub-languages, since the human users of an interface will learn to pare their English or Russian down to the minimum needed to interact effectively with the applications using the interface.

What follows is a brief look at some kinds of natural-language interfaces that are currently found in applications. It also offers some suggestions on how to use distributional methods to provide structure in the interface language. For background on natural-language interfaces, see (Schneiderman 1998: 293 ff.). Harris's focus on method and his characterizations of linguistic structure in such abstract concepts as string and operator structure provide the flexibility needed in the task of stating requirements for natural-language interfaces.

Discrete word interfaces

One of the easiest interfaces to implement is one in which the words of a natural language simply replace the events of the mouse, for such simple events as button clicks and menu choices. For example, assuming txt_i has been selected, the following scenario might follow.

(27) *u:utter:format a:dropDown:mnuFormat u:utter:font a:display:dlgFont u:utter:bold a:highlight:dlgFont.lblFontStyle.itmBold u:utter:okay a:bold:*txt_i

"The user utters *format* and the application drops down the format menu. Then the user utters *font* and the application displays the font dialog box. The user then utters *bold* and the application highlights the bold item in the font style list. The user utters *okay* and the application bolds the selected text."

It is clear that this kind of user interface is structurally similar to the graphical user interface, substituting utterances for mouse movements and clicks. In particular, the operator structure is the same. It has the advantage of freeing

the user's hands. A natural-language interface can provide more, however. It can also free the user's attention. A discrete word interface keeps the semantics clear, but perhaps too lockstep. One of the great benefits of natural language is that an utterance can proceed with a higher degree of independence from the events of the environment. One problem in designing a natural language for an interface is to balance these two somewhat contradictory requirements: affording the user the independence and flexibility of natural language while keeping the semantics of the events in the application clear.

Operator-less interfaces

There are user interfaces which appear at first glance to allow users to use their natural language, but in fact lack operator structure. A readily available example of an operator-less language is the query language of Microsoft Office Assistant®. This assistant asks the user to type a question. The user types an English sentence, for example, asking how to save a macro. The assistant responds with a list of possible answers, which the user can choose from. The responses are dependent on what the user types. However, the response is the same in both of the following cases.

(28) How do I save a macro?

(29) How do I macro a save? (using *macro* as a verb)

Microsoft Office Assistant® is not sensitive to the operator structure of English. The user eventually learns that it is not worth the effort to craft a carefully worded English question and instead economizes with

(30) save macro

or

(31) macro save

It becomes clear that this embedded version of English has no operator structure. That does not mean it has no semantics. It has an operator-less semantics (Harris 1991:369). When creating a natural-language user interface, it is important to be clear with the user about whether the language has operator structure or not. Microsoft Office Assistant® usually makes it possible for users of any level to eventually find the help they seek. However, novice users especially are initially puzzled and frustrated by the wide range of choices offered to them, many of which seem to have nothing to do with their question. The interface does have the advantage that users do not have to spend

the thought to analyze their intended query and extract key words, since Microsoft Office Assistant® does that for them. An improvement on the prompt provided in the dialogue box *Type your question here* would be *Type your question or key words here*. We need to be especially careful to avoid confusing users, as we add more natural-language structures to user interfaces.

Natural-language structures to consider

When designing a natural-language interface, it is natural for us to look to a familiar language to suggest features that might be useful. We then need to construct numerous scenarios that could provide a basis set from which distributional methods would be able to establish whether those features were present. These scenarios can function as requirements for the user interface, guiding the design and implementation of the interface, as noted above.

For a first simple example, assume a scenario in which there is only one occurrence of the word *Zellig* in the document.

(32) *u:utter:bold a:display:boldFlag u:utter:zellig a:highlight:Zellig a:emphasize:Zellig u:pause a:bold:Zellig a:unemphasize:Zellig a:unhighlight:Zellig*
user-utters-word-bold app-displays-bold-flag user-utters-word-zellig app-highlights-text-*Zellig* app-emphasizes-text-*Zellig* user-pauses app-bolds-text-*Zellig* app-unemphasizes-text-*Zellig* app-unhighlights-text-*Zellig*
"The user utters the word *bold* and the application displays the bold-flag. Then the user utters the word *zellig* and the application highlights the text segment *Zellig* and emphasizes it. The user pauses significantly and the application bolds the text segment *Zellig*, unemphasizes it, and unhighlights it."

I remind the reader that my choice of a word processor is problematical in this example, because it can lead us to unwarranted assumptions. We need to be clear that the uttered sounds *zellig* should not be thought of as predisposed to have any connection to the sequence of letters *Zellig*. That is, until we can show with distributional methods that the sounds *zellig* and the letters *Zellig* are co-referential, there is no justifiable connection between the sound sequences and the sequences of letters. We need further examples in which *bold*, *zellig*, and *Zellig* are replaced with other features. We need to decide to have our user interface support a whole set of scenarios of the following form.

(33) *u:utter:cmdWord*$_k$ *a:display:cmdFlag*$_k$ *u:utter:word*$_i$ *a:highlight:txt*$_i$ *a:emphasize:txt*$_i$ *u:pause a:textCmd*$_k$*:txt*$_i$ *a:unemphasize:txt*$_i$ *a:unhighlight:txt*$_i$

with relations

(34)

cmdWord	cmdFlag
bold	*boldFlag*
italicize	*italicFlag*
underline	*underlineFlag*

(35)

cmdWord	textCmd
bold	*bold*
italicize	*italicize*
underline	*underline*

(36)

word	textSegment
zellig	*Zellig*
harris	*Harris*
linguist	*linguist*

The above scenario is similar to examples (23), (24), and (26) in the preceding discussion of graphical user interfaces. The same distributional methods should be applied to the design of natural-language interfaces to enhance the learnability of co-reference structure and operator structure.

How much detail from a familiar natural language do we want to try to support in our natural-language user interface? Word classes and word class sequences are likely candidates for a natural language. Do we want to support string structure, i.e., modifier structure? For example, should the following two user event sequences have the same associated application events?

(37) *u:utter:bold u:utter:zellig u:utter:after u:utter:met*
"User utters *bold zellig after met*"

(38) *u:utter:bold u:utter:met u:utter:after u:utter:zellig*
"User utters *bold met after zellig*"

If not, we should generate scenarios that support assigning string structure, i.e., modifier structure, using distributional methods.

Does the English pronoun *it* actually function as a pronoun in the following?

(39) Select zellig and bold it

And how does a pronoun function exactly? What methods can be used to support learning the concept of a pronoun? Co-reference patterns form a central part of the concept. Recent linguistics research abounds with examples and theories about pronouns.

What about determiners and quantifiers? Should our user interface support the differences among the following English sentences?

(40) Bold the word zellig
Bold all words zellig
Bold one word zellig
Bold any word zellig
Bold some words zellig

Ambiguity and correctness

A successful natural-language interface should imitate human natural languages when reacting to ambiguities and mistakes. A distributional semantics is able to describe interactions that allow for clarification of ambiguities. Feedback is part of the sequence of events that is being described. For example, an application might supply possible co-referents by highlighting them and emphasizing the current choice. It could then reduce that set of co-referents as qualifiers are provided, or even change it if the user supplies a correction.

(41) *u:utter:bold a:display:boldFlag u:utter:zellig a:highlight:Zellig*$_1$ *a:highlight:Zellig*$_2$ *a:emphasize:Zellig*$_1$ *u:utter:not u:utter:that u:utter:zellig a:unemphasize:Zellig*$_1$ *a:emphasize:Zellig*$_2$ *u:utter:yes u:utter:that u:utter:one u:pause a:bold:Zellig*$_2$ *a:unemphasize:Zellig*$_2$ *a:unhighlight:Zellig*$_2$ *a:unhighlight:Zellig*$_1$
"The user utters the words *bold zellig*. The application displays the bold-flag, highlights two *Zellig* text segments, and emphasizes the first. The user utters the words *not that one* and the application switches the emphasis to the second *Zellig*. The user utters the words *yes that one* and pauses significantly. The application bolds the second *Zellig*, unemphasizes it, and unhighlights both *Zellig*s."

The highlighted alternatives should reflect the operator structure of the language, not simply all possible environments in which a word could be used. Structure is important for avoiding the largely irrelevant feedback characteristic of operator-less interfaces. An application might also allow a deictic event

of clicking the mouse on an item to resolve ambiguity, with or without uttering phrases, such as *this one*.

When the user makes a mistake, the application should resolve it in a manner similar to the way it resolves ambiguities. The application should not balk at ungrammatical sentences or incorrect pronunciation. While completely grammatical and correctly pronounced utterances should produce results quickly, most utterances will not be completely correct. The application should provide feedback using its best guess. It should not just indicate there was an error and suggest a correction. The application should allow the user to continue to apply the distributional method in reasonably productive activities, even when he or she makes mistakes. This is especially important in the initial learning of an application. The designers of the interface need to provide an extensive set of error scenarios. The application should not revert to a less structured interface, such as an operator-less interface, for the incorrect stretches. It should avoid providing a large number of mostly irrelevant options.

Summary

We have seen that distributional methods can reveal semantic structures in graphical user interfaces. We can also use such methods to design natural-language interfaces by providing feedback that allows a user to learn a particular natural-language interface. A natural-language interface based on scenarios supported by distributional methods will suggest to the user the kinds of structures that are already familiar in human natural languages. Zellig Harris's use of distributional methods to determine linguistic structure provides both flexibility and constraints in designing a natural language for humans to interface with a computer application.

References

Andersen, Peter Bøgh 1997. *A Theory of Computer Semiotics*. Cambridge University Press.

Harris, Zellig S. 1951. *Methods in Structural Linguistics*. Chicago: University of Chicago Press, xvi, 384 pp. (Repr. as "Phoenix Books" P 52 with the title *Structural Linguistics*, 1960.) [Preface signed "Philadelphia, January 1947".]

Harris, Zellig S. 1952. "Discourse analysis." *Language* 28.1:1–30. (Page references from reprint in Harris 1970:313–348.)

Harris, Zellig S. 1954. "Distributional structure." *Word* 10.2/3:146–162. (Page references from reprint in Harris 1970:775–794.)
Harris, Zellig S. 1965. "Transformational Theory." *Language* 41.3:363–401. (Page references from reprint in Harris 1970:533–577.)
Harris, Zellig S. 1969. "The two systems of grammar: Report and paraphrase". (=Transformations and Discourse Analysis Papers, 79.) Philadelphia: University of Pennsylvania. (Page references from reprint in Harris 1970:612–692.)
Harris, Zellig S. 1970. *Papers in Structural and Transformational Linguistics*. Dordrecht: D. Reidel.
Harris, Zellig S. 1988. *Language and Information*. New York. Columbia University Press.
Harris, Zellig S. 1991. *A Theory of Language and Information: A mathematical approach*. Oxford: Clarendon Press.
Hiż, Henry 1968. "Referentials." *Semiotica* 1:136–166.
Nevin, Bruce 1988. "Review of *Language and information*". *Computational Linguistics* 14.4:87–90.
Rumbaugh, James, et al. 1991. *Object-Oriented Modeling and Design*. Englewood Cliffs: Prentice Hall.
Schneiderman, Ben. 1998. *Designing the User Interface: Strategies for effective human-computer interaction*. Reading, Mass.: Addison Wesley Longman, Inc.
Smaby, Richard M. 1971. *Paraphrase Grammars*. Dordrecht: D. Reidel.
Smaby, Richard M. 1981. "Ambiguity of pronouns." *Aspects of Philosophical Logic*, ed. by Uwe Moennich. Dordrecht: D. Reidel, 129–156.

Zellig Sabbettai Harris

A comprehensive bibliography of his writings, 1932–2002*

Compiled by

E. F. K. Koerner
University of Ottawa

1932. *Origin of the Alphabet.* Unpublished M.A. thesis, University of Pennsylvania, Philadelphia, Pa., 111 typed pp.

1933. "Acrophony and Vowellessness in the Creation of the Alphabet". *Journal of American Oriental Society* 53:387. [Summary of 1932 thesis.]

1934a. "The Structure of Ras Shamra C". *Journal of the American Oriental Society* 54:80–83.

1934b. Review of Raymond P[hilip] Dougherty [(1877–1933)], *The Sealand of Ancient Arabia* (New Haven, Conn.: Yale University Press; London: Oxford University Press, 1932). *Journal of the American Oriental Society* 54:93–95.

1935a. Review of Edward Chiera, *Joint Expedition [of the American School of Oriental Research in Bagdad] with the Iraq Museum of Nuzi*, vols. 4–5 (Paris: P. Geuthner, 1933–1934). *Language* 11:262–263.

1935b. "A Hurrian Affricate or Sibilant in Ras Shamra". *Journal of the American Oriental Society* 55:95–100.

1936a. (Together with James A[lan] Montgomery [(1866–1949)].) *The Ras Shamra Mythological Texts.* (=*Memoirs of the American Philosophical Society*, 4.) Philadelphia: American Philosophical Society, 134 pp.
Reviewed by
Edward Sapir (1884–1939) in *Language* 13:326–331 (1937).

1936b. *A Grammar of the Phoenician Language.* (=*American Oriental Series*, 8.) New Haven, Conn.: American Oriental Society, xi, 172 pp.

---

* A first version of the present list appeared in *Historiographia Linguistica* 20:2/3.509–522 (1993), pp. 510–520. (The "Introductory remarks" [p. 509] and Appendix "Appraisals of Zellig S. Harris, 1962–1993" [520–522] are omitted here). I'd like to thank Bruce E. Nevin for his corrections and additions.

[Ph.D. dissertation, University of Pennsylvania, Philadelphia, 1934.]
Reviewed by
Edward Sapir in *Language* 15: 60–65 (1939);
Vojtěch Šanda (1873–*post* 1939) in *Archiv Orientální* 11: 177–178 (1939);
Charles François Jean (1874–1965) in *Revue des Études Sémitiques* 1940: 94–96;
Maria Höfner (1901–1995?) in *Wissenschaftliche Zeitschrift für die Kunde des Morgenlandes* 48: 153 (1941).

1936c. "Back Formation of *itn* in Phoenician and Ras Shamra". *Journal of the American Oriental Society* 56: 410. [Abstract.]

1937. "A Conditioned Sound Change in Ras Shamra". *Journal of the American Oriental Society* 57: 151–157.

1938a. "Expression of the Causative in Ugaritic". *Journal of the American Oriental Society* 58: 103–111.

1938b. "Ras Shamra: Canaanite civilization and language". *Annual Report of the Smithsonian Institution 1937*: 479–502; illus., 4 pl., 1 map on 2 leaves. Washington, D.C.

1939a. *Development of the Canaanite Dialects: An investigation in linguistic history*. (= *American Oriental Series*, 16.) New Haven, Conn.: American Oriental Society, x, 108 pp.; illus., map. (Repr., Millwood, N.Y.: Kraus, 1978.)
Reviewed by
William Foxwell Albright (1891–1971) in *Journal of the American Oriental Society* 60: 414–422 (1940);
René Dussaud (1868–1958) in *Syria: Revue d'art oriental et d'archéologie* 1940: 228–230 (Paris);
Gonzague Ryckmans (1887–1969) in *Le Muséon* No. 53: 135 (1940);
Harold Louis Ginsberg (b.1903) in *Journal of Biblical Literature* 59: 546–551 (1940);
Max Meir Bravmann (1909–1977?) in *Kirjath Sepher* 17: 370–381 (1940);
Marcel Cohen (1884–1974) in *Bulletin de la Société de Linguistique de Paris* No. 123: 62 (1940–1941);
Raphaël Savignac in *Vivre et Penser* 1941: 157–159;
Albrecht Goetze (1897–1971) in *Language* 17: 167–170 (1941);
Alexander Mackie Honeyman (1907–1988) in *Journal of the Royal Asiatic Soci-ety of Great Britain and Ireland* 17: 167–170 (1941);
Bernard Baron Carra de Vaux (1867–*c.*1950) in *Journal of the Palestine Oriental Society* 19: 329–330 (Jerusalem, 1941);
Franz Rosenthal (b.1914) in *Orientalia* 11: 179–185 (1942);
Ronald J[ames] Williams (b.1917) in *Journal of Near Eastern Studies* 1: 378–380 (Chicago, 1942).

1939b. "Development of the the West Semitic Aspect System". *Journal of the American Oriental Society* 59: 409–410. [Abstract.]

1939c. (Together with Charles F. Voegelin [(1906–1986)].) *Hidatsa Texts Collected by Robert H. Lowie*, with grammatical notes and phonograph transcriptions by Z. S. Harris & C. F. Voegelin. (= *Prehistory Research Materials*, 1:6), 173–239. Indianapolis: Indiana Historical Society. (Repr., New York: AMS Press, 1975.)
[According to Harris (1990, 2002), in the late 1930s and early 1940s he used this

Hidatsa material, together with the Kota texts that are reviewed in (1945f), for his development of substitution grammar, discourse analysis, and transformational analysis, research which was demonstrated at the Linguistic Institute and subsequently first published in (1946a) and (1952a).]

1940. Review of Louis H[erbert] Gray (1875–1955), *Foundations of Language* (New York: Macmillan, 1939). *Language* 16.3: 216–231. (Repr., with the title "Gray's *Foundations of Language*", in Harris 1970a: 695–705.)

1941a. "Linguistic Structure of Hebrew". *Journal of the American Oriental Society* 61: 143–167. [Also published as *Publications of the American Oriental Society*; Offprint series, No. 14.]

1941b. Review of N[ikolaj] S[ergeevič] Trubetzkoy (1890–1938), *Grundzüge der Phonologie* (Prague: Cercle Linguistique de Prague, 1939). *Language* 17: 345–349. (Repr. in Harris 1970a: 706–711, and in *Phonological Theory: Evolution and current practice* ed. by Valerie Becker Makkai, 301–304. New York: Holt, Rinehart & Winston, 1972; repr., Lake Bluff, Ill.: Jupiter Press, 1978.)

1941–1946. "Cherokee Materials". Manuscript 30(I2.4). [Typed D. and A.D. 620L., 575 slips, 10 discs.] Philadelphia: American Philosophical Society Library.

1942a. "Morpheme Alternants in Linguistic Analysis". *Language* 18.3: 169–180. (Repr. in *Readings in Linguistics* [I]: *The development of descriptive linguistics in America since 1925* [*in later editions: 1925–56*] ed. by Martin Joos [Washington, D.C.: American Council of Learned Societies, 1957; 4th ed., Chicago & London: University of Chicago Press, 1966], pp. 109–115 [with a postscript by Joos, p. 115] and subsequently in Harris 1970a: 78–90, and 1981: 23–35.)

1942b. "Phonologies of African Languages: The phonemes of Moroccan Arabic". *Journal of the American Oriental Society* 62.4: 309–318. (Repr., under the title of "The Phonemes of Moroccan Arabic", in Harris 1970a. 161–176.)

[Read at the Centennial Meeting of the Society, Boston 1942. — Cf. the critique by Jean Cantineau, "Réflexions sur la phonologie de l'arabe marocain", *Hespéris* 37. 193–207 (1951 for 1950). —Harris (2002[1990]) states that "it was possible to describe the entire program from the outset, e.g. in" this paper.]

1942c. Review of *Language, Culture, and Personality: Essays in memory of Edward Sapir* ed. by Leslie Spier, A[lfred] Irving Hallowell & Stanley S[tewart] Newman (Menasha, Wisconsin: Edward Sapir Memorial Fund, 1941). *Language* 18: 238–245.

1942d. (Together with William E[verett] Welmers [1916–1988].) "The Phonemes of Fanti". *Journal of the American Oriental Society* 62: 318–333.[1]

1942e. (Together with Fred Lukoff [1920–2000].) "The Phonemes of Kingwana-Swahili". *Journal of the American Oriental Society* 62: 333–338.

1944a. "Yokuts Structure and [Stanley] Newman's Grammar". *IJAL* 10.4: 196–211. (Repr. in Harris 1970a: 188–208.)

1944b. "Simultaneous Components in Phonology". *Language* 20: 181–205. (Repr. in *Readings in Linguistics* [I]: *The development of descriptive linguistics in America since 1925* [*in later editions: 1925–56*] ed. by Martin Joos [Washington, D.C.: American

1. Harris also served as the editor of *JAOS* from 1941 to 1947.

Council of Learned Societies, 1957; 4th ed., Chicago & London: University of Chicago Press, 1966], pp. 124–138 [with a postscript by Joos, p. 138] and subsequently also in Harris 1970a: 3–31 as well as in *Phonological Theory: Evolution and current practice* ed. by Valerie Becker Makkai, 115–133. New York: Holt, Rinehart & Winston, 1972; repr., Lake Bluff, Ill.: Jupiter Press, 1978.)

1945a. "Navaho Phonology and [Harry] Hoijer's Analysis". *IJAL* 11.4: 239–246. (Repr. in Harris 1970a: 177–187.)

1945b. "Discontinuous Morphemes". *Language* 21.2: 121–127. (Repr. in Harris 1970a: 91–99, and in Harris 1981: 36–44.)

1945c. "American Indian Linguistic Work and the Boas Collection". *Library Bulletin of the American Philosophical Society* 1945: 57–61. Philadelphia.
Review note by Thomas A[lbert] Sebeok (1920–2002) in *IJAL* 13: 126 (1947).

1945d. (Together with Charles F. Voegelin.) *Index to the Franz Boas Collection of Materials for American Linguistics.* (= *Language Monographs*, 22.) Baltimore, Md.: Linguistic Society of America, 43 pp. (Repr., New York: Kraus, 1974.)

1945e. (Together with Charles F. Voegelin.) "Linguistics in Ethnology". *Southwestern Journal of Anthropology* 1: 455–465.

1945f. Review of Murray B[arnson] Emeneau, *Kota Texts*, vol. I (Berkeley: University of California Press, 1944). *Language* 21: 283–289. (Repr., under the title "Emeneau's Kota Texts", in Harris 1970a: 209–216.)
[According to Harris (1990, 2002), in the late 1930s and early 1940s he used the Kota texts that are reviewed here, together with the Hidatsa material of (1939c), for his development of substitution grammar, discourse analysis, and transformational analysis, research which was demonstrated at the Linguistic Institute and subsequently first published in (1946a) and (1952a).]

1946a. "From Morpheme to Utterance". *Language* 22.3: 161–183. (Repr. in Harris 1970a: 100–125, and in Harris 1981: 45–70.)

1946b. (Together with Ernest Bender [b.1919].) "The Phonemes of North Carolina Cherokee". *IJAL* 12: 14–21. (Repr. in *Readings in Linguistics* [I]*: The development of descriptive linguistics in America since 1925* [*in later editions: 1925–56*] ed. by Martin Joos [Washington, D.C.: American Council of Learned Societies, 1957; 4th ed., Chicago & London: University of Chicago Press, 1966], pp. 142–153 [with a postscript by Joos, p. 153].)

1947a. "Developments in American Indian Linguistics". *Library Bulletin of the American Philosophical Society* 1946: 84–97. Philadelphia.
Review note by Thomas A[lbert] Sebeok in *IJAL* 14: 209 (1948).

1947b. "Structural Restatements I: Swadesh's Eskimo; Newman's Yawelmani". *IJAL* 13.1: 47–58. (Repr. in Harris 1970a: 217–234, and in Harris 1981: 71–88.)
["Attempt to restate in summary fashion the grammatical structures of a number of American Indian languages. The languages to be treated are those presented in H[arry] Hoijer and others, *Linguistic Structures of Native America* [New York, 1946]." – On Morris Swadesh's account of Eskimo and Stanley S. Newman's of Yawelmani Yokuts.]

1947c. "Structural Restatements II: Voegelin's Delaware". *IJAL* 13.3: 175–186. (Repr. in Harris 1970a: 235–250, and 1981: 89–104.)
[On Voegelin's grammatical sketch of Delaware.]

1947d. (Together with Charles F. Voegelin.) "The Scope of Linguistics". *American Anthropologist* 49:588–600.
[1, The place of linguistics in cultural anthropology; 2, Trends in linguistics.]

1947e. (Associate ed., with Helen Boas-Yampolsky as main ed.) Franz Boas, *Kwakiutl Grammar, with glossary of the suffixes.* (= *Transactions of the American Philosophical Society*, n.s. 37:199–377.) Philadelphia: American Philosophical Society.
Reviewed by
Morris Swadesh in *Word* 4:58–63 (1948);
C[harles] F[rederick] Voegelin in *Journal of American Folklore* 61:414–415(1948).

1948. "Componential Analysis of a [Modern] Hebrew Paradigm". *Language* 24.1:87–91. (Repr. in *Readings in Linguistics* [I]*: The development of descriptive linguistics in America since 1925* [*in later editions: 1925–56*] ed. by Martin Joos [Washington, D.C.: American Council of Learned Societies, 1957; 4th ed., Chicago & London: University of Chicago Press, 1966], pp. 272–274 [with a postscript by Joos, p. 274] and — with 'Hebrew' in the title dropped — in Harris 1970a:126–130.)

1951a. *Methods in Structural Linguistics.* Chicago: University of Chicago Press, xvi, 384 pp. (Repr., under the title of *Structural Linguistics*, as "Phoenix Books" P 52, 1960; 7th impression, 1966; repr. again in Harris 1984.) [Preface signed "Philadelphia, January 1947".]
Reviewed by
Norman A[nthony] McQuown in *Language* 28.4:495–504 (1952);
Murray Fowler in *Language* 28:504–509 (1952);
C[harles] F[rederick] Voegelin in *Journal of the American Oriental Society* 72:113–114 (1952);
Charles F[rancis] Hockett in *American Speech* 27:117–121 (1952);
Stanley S[tewart] Newman in *American Anthroplogist* 54:404–405 (1952);
Margaret Mead in *IJAL* 18:257–260 (1952);
Fred W[alter] Householder in *IJAL* 18:260–268 (1952);
Fernand Mossé in *Études Germaniques* 7:274 (1952);
Walburga von Raffler[-Engel] in *Paideia* 8:229–230 (1953);
Knud Togeby in *Modern Language Notes* 68:191–194 (1954);
K[enneth] R. Brooks in *Modern Language Review* 48:496 (1953);
Milka Iviç in *JuĬnoslovenski Filolog* 20:474–478 (Belgrade, 1953/54);
Jean Cantineau in *Bulletin de la Société de Linguistique de Paris* 50.2:4–9(1954);
Eugene Dorfman in *Modern Language Journal* 38:159–160 (1954);
Robert Léon Wagner in *Journal de Psychologie* 47:537–539 (1954);
Harry Hoijer in *Romance Philology* 9:32–38 (1955–56);
Paul L[ucian] Garvin in *Romance Philology* 9:38–41 (1955/56).

1951b. (With Charles F. Voegelin.) "Methods for Determining Intelligibility among Dialects of Natural Languages". *Proceedings of the American Philosophical Society* 95: 322–329; 1 fig. Philadelphia.

1951c. Review of David G. Mandelbaum (ed.), *Selected Writings of Edward Sapir in Language, Culture, and Personality* (Berkeley & Los Angeles: University of California Press, 1949). *Language* 27.3:288–333. (Repr. in Harris 1970a:712–764, and in *Edward Sapir:*

Appraisals of his life and work ed. by Konrad Koerner, 69–114. Amsterdam & Philadelphia: John Benjamins, 1984.)

[This insightful review article is also to be reprinted in *Edward Sapir: Critical Assessments* ed. by E. F. K. Koerner, vol. I (London & New York: Routledge, 2003).]

1951d. "Ha-Safah ha-Ivrit l'or ha-balshanut ha-chadashah [The Hebrew language in the light of modern linguistics]". *Lěšonénu: A journal for the study of the Hebrew language and cognate studies* 17: 128–132 (1950/1951). Jerusalem.

1952a. "Culture and Style in Extended Discourse". *Selected Papers from the 29th International Congress of Americanists (New York, 1949)*, vol. III: *Indian Tribes of Aboriginal America* ed. by Sol Tax & Melville J[oyce] Herskovits, 210–215. New York: Cooper Square Publishers. (Entire volume repr., New York: Cooper Press, 1967; paper repr. in Harris 1970: 373–389.)

[Proposes a method for analyzing extended discourse, with sample analyses from Hidatsa, a Siouan language spoken in North Dakota.]

1952b. "Discourse Analysis". *Language* 28.1: 1–30. (Repr. in *The Structure of Language: Readings in the philosophy of language* ed. by Jerry A[lan] Fodor & Jerrold J[acob] Katz, 355–383. Englewood Cliffs, N.J.: Prentice-Hall, 1964; and also in Harris 1970a: 313–348 and Harris 1981: 107–142.)

[Presents a method for the analysis of connected speech or writing.]

1952c. "Discourse Analysis: A sample text". *Language* 28.4: 474–494. (Repr. in Harris 1970a: 349–379.)

1952d. (Together with Charles F. Voegelin.) "Training in Anthropological Linguistics". *American Anthropologist* 54: 322–327.

1953. (Together with C. F. Voegelin.) "Eliciting in Linguistics". *Southwestern Journal of Anthropology* 9.1: 59–75. (Repr. in Harris 1970a: 769–774.)

[1, Practices with respect to eliciting; 2, Imitation and repetition; 3, Eliciting with pictures; 4, Translation eliciting; 5, Text eliciting, and 6, The validity of eliciting.]

1954a. "Transfer Grammar". *IJAL* 20.4: 259–270. (Repr. in Harris 1970a: 139–157.)

[1, "Defining difference between languages"; 2, "Structural transfer"; 3, "Phonetic and phonemic similarity"; 4, "Morphemes and morphophonemes"; 5, "Morphological translatability".]

1954b. "Distributional Structure". *Word* 10:2/3: 146–162. (Also in *Linguistics Today: Published on the occasion of the Columbia University Bicentennial* ed. by André Martinet & Uriel Weinreich, 26–42. New York: Linguistic Circle of New York, 1954. Repr. in *The Structure of Language: Readings in the philosophy of language* ed. by Jerry A[lan] Fodor & Jerrold J[acob] Katz, 33–49. Englewood Cliffs, N.J.: Prentice-Hall, 1964, and also in Harris 1970a: 775–794 and 1981: 3–22.)

1955a. "From Phoneme to Morpheme". *Language* 31.2: 190–222; 7 tables. (Repr. in Harris 1970a: 32–67.)

[Presents a constructional procedure segmenting an utterance in a way which correlates well with word and morpheme boundaries.]

1955b. "American Indian Work and the Boas Collection". *Library Bulletin of the American Philosophical Society* 1955: 57–61. Philadelphia.

1956a. (Editor), *A Bushman Dictionary* by Dorothea F[rances] Bleek [d. 1948]. (= *American Oriental Series*, 41.) New Haven, Conn.: American Oriental Society, xii, 773 pp.
Reviewed by
C[lement] M[artyn] Doke in *African Studies* 16: 124–125 (1957);
E.O.J. Westphal in *Africa* 27: 203–204 (1957);
A. J. C[oetzee] in *Tydskrif vir Volkskunde en Volkstaal* 14.1: 29–30(Johannesburg, 1957);
Joseph H[arold] Greenberg in *Language* 33: 495–497 (1957);
Henri Peter Blok in *Neophilologus* 41: 232–234 (1957);
Louis Deroy in *Revue des Langues Vivantes* 23: 174–175 (1957);
Otto Köhler in *Afrika und Übersee* 43: 133–138 (1959).

1956b. "Introduction to Transformations". (= *Transformations and Discourse Analysis Papers*, No. 2.) Philadelphia: University of Pennsylvania. (Repr. in Harris 1970a: 383–389.)

1957a. "Co-Occurrence and Transformation in Linguistic Structure". *Language* 33.3: 283–340. (Repr. in *The Structure of Language: Readings in the philosophy of language* ed. by Jerry A[lan] Fodor & Jerrold J[acob] Katz, 155–210. Englewood Cliffs, N.J.: Prentice-Hall, 1964, and also in Harris 1970: 390–457, 1972: 78–104 [in parts]. Anthologized in *Syntactic Theory 1: Structuralist. Selected readings* ed. by Fred W. Householder, 151–185. Harmondsworth, Middlesex & Baltimore, Md.: Penguin Books, 1972, and also repr. in Harris 1981: 143–210.)
[Revised and enlarged version of Presidential Address, Linguistic Society of America, December 1955. — Defines a formal relation among sentences, by virtue of which one sentence structure may be called a transform of another sentence structure.]

1957b. "Canonical Form of a Text". (= *Transformations and Discourse Analysis Papers*, No. 3b.) Philadelphia: University of Pennsylvania.
[This and two other previously unpublished papers — items 4a and 3c in the same series — were combined to form entry 1963a (below).]

1959a. "The Transformational Model of Language Structure". *Anthropological Linguistics* 1.1: 27–29.

1959b. "Computable Syntactic Analysis". (= *Transformations and Discourse Analysis Papers*, No. 15.) Philadelphia: University of Pennsylvania. (Revised version published as item 1962a; excerpted, with the added subtitle "The 1959 computer sentence-analyzer", in Harris 1970a: 253–277.)

1959c. *Linguistic Transformations for Information Retrieval.* (= *Interscience Tracts in Pure and Applied Mathematics*, 1958:2.) Washington, D.C.: National Academy of Sciences — National Research Council. (Repr. in Harris 1970a: 458–471.)
[From the 1958 Proceedings of the International Conference on Scientific Information.]

1960a. *Structural Linguistics.* (= *Phoenix Books*, P 52.) Chicago: University of Chicago Press, xvi, 384 pp. (7th impression, 1966; repr., 1984.) [Reprint of item 1951, with a supplementary preface (vi-vii).]
Reviewed by
Simeon Potter in *Modern Language Review* 57: 139 (1962).

1960b. "English Transformation List". (= *Transformations and Discourse Analysis Papers*, No. 30.) Philadelphia: University of Pennsylvania.

1961. "Strings and Transformations in Language Description". Published as No. 1 of *Papers in Formal Linguistics* ed. by Henry Hiż. Department of Linguistics, University of Pennsylvania. (Published, under the title "Introduction to String Analysis", in Harris 1970a: 278–285.)

1962a. *String Analysis of Sentence Structure*. (= *Papers on Formal Linguistics*, 1.) The Hague: Mouton, 70 pp. (2nd ed., 1964; repr., 1965.)
[Revised version of item 1959b.]
Reviewed by
Robert E[dmondson] Longacre in *Language* 39: 473–478 (1963);
László Antal in *Linguistics* No. 1: 97–104 (1963);
Murray Fowler in *Word* 19: 245–247 (1963);
Klaus Baumgärtner in *Germanistik* 4: 194 (1963);
Robert B[enjamin] Lees in *IJAL* 30: 415–420 (1964);
Karel Pala in *Sborník Prací Filosofické Fakulty Brnûnské Univerzity* 13 (A 12).238–241 (Brno, 1964);
G. G. Pocepkov in *Voprosy Jazykoznania* 13.1: 123–128 (1965);
Karel Pala in *Slovo a Slovesnost* 26: 78–80 (1965);
Kazimierz Polaěski in *Biuletyn Fonegraficzne* 8: 139–143 (Pozna , 1967).

1962b. "Sovmestnaja vstrecaemost' i transformacija v jazykovoj strukture". *Novoe v lingvistike* ed. by V[ladimir] A[ndreevič] Zvegincev, vol. II: *Transformacionnaja grammatika*, 528–636. Moscow: Izd. Innostr. Literatury. [Transl. by T(atjana) N. Molosaja of item 1957a, with an introd. by S(ebastian) K(onstantinovič) Šaumjan.]

1962c. "A Language for International Cooperation". *Preventing World War III: Some proposals* ed. by Quincy Wright, William M. Evan & Morton Deutsch, 299–309. New York: Simon & Schuster. (Repr. in Harris 1970a. 795–805.)

1963a. *Discourse Analysis Reprints*. (= *Papers on Formal Linguistics*, 2.) The Hague: Mouton, 73 pp. [See comment on entry 1957b (above).]
Reviewed by
Klaus Baumgärtner in *Germanistik* 5: 412 (1964);
Manfred Bierwisch in *Linguistics* No. 13: 61–73 (1965);
Fred[erick] C[hen] C[hung] Peng in *Lingua* 1 6: 325–330 (1966);
György Hell in *Acta Linguistica Academiae Scientarum Hungaricae* 18: 233–235(1968);
Tae-Yong Pak in *Language* 46: 754–764 (1970).

1963b. "Immediate-Constituent Formulation of English Syntax". (= *Transformations and Discourse Analysis Papers*, No. 45.) Philadelphia: University of Pennsylvania. (Repr. in Harris 1970a: 131–138.)

1964a. "Transformations in Linguistic Structure". *Proceedings of the American Philosophical Society* 108.5: 418–422. (Repr. in Harris 1970a: 472–481.) [Read on 25 April 1964.]

1964b. "The Elementary Transformations". (= *Transformations and Discourse Analysis Papers*, No. 54.) Philadelphia: University of Pennsylvania. (Excerpted in Harris 1970a: 482–532, 1972: 57–75, and, in abbreviated form, in Harris 1981: 211–235.)

1965. "Transformational Theory". *Language* 41.3: 363–401. (Repr. in Harris 1970a: 533–577, 1972: 108–154, and 1981: 236–280.)

1966a. "Algebraic Operations in Linguistic Structure". Paper read at the International Congress of Mathematicians, Moscow 1966. (Published in Harris 1970a: 603–611.)

1966b. "A Cyclic-Cancellation Automation for Sentence Well-Formedness". *International Computation Centre Bulletin* 5: 69–94. (Also distributed as *Transformations and Discourse Analysis Papers*, No. 51. Repr. in Harris 1970a: 286–309.)

1967a. "Decomposition Lattices". (= *Transformations and Discourse Analysis Papers* No. 70.) Philadelphia: University of Pennsylvania. (Repr. in Harris 1970a. 578–602, and excerpted in Harris 1981: 281–290.)

1967b. "Morpheme Boundaries within Words: Report on a computer test". (= *Transformations and Discourse Analysis Papers*, No. 73.) Philadelphia: University of Pennsylvania. (Repr. in Harris 1970a: 68–77.)

1968a. *Mathematical Structures of Language*. (= *Interscience Tracts in Pure and Applied Mathematics*, 21.) New York: Interscience Publishers John Wiley & Sons, ix, 230 pp. [Index of terms compiled by Maurice Gross.]

Reviewed by

Wojciech Skalmowski in *ITL: Tijdschrift van het Instituut voor Toegepaste Linguïstiek* 4: 56–61 (Leuven, 1969);

Maurice Gross in *Semiotica* 2: 380–390 (1970), repr. in item 1972: 314–324 (withan introd. in German by Senta Plötz [p. 313] and an English abstract by theauthor [p. 314]);

Maurice Gross & Marcel-Paul Schützenberger in *The American Scientist* 58(1970); repr. in Harris 1972: 308–312 (with summaries in German andEnglish by Senta Plötz [p. 307]);

Petr Pitha in *Slovo a Slovesnost* 32: 59–65 (1971);

Lucia Vaina-Puşcă in *Revue Roumaine de Linguistique* 16: 369–371 (1971).

1968b. "Edward Sapir: Contributions to linguistics". *International Encyclopedia of the Social Sciences* ed. by David L. Sills, vol. XIV, pp. 13–14. New York: Macmillan. (Repr., in a somewhat longer, probably the original, form in Harris 1970a: 765–768.)

1968c. "Du morphème à l'expression". *Langages* No. 9: 23–50. [Transl. of item 1946b.]

1969a. *The Two Systems of Grammar: Report and paraphrase*. (= *Transformations and Discourse Analysis Papers*, No. 79.) Philadelphia: University of Pennsylvania. (Repr. in Harris 1970a: 612–692, in Harris 1972: 158–240 (revised), and in Harris 1981: 293–351 (shortened).)

1969b. "Analyse du discours". *Langages* No. 13: 8–45. [French transl. of item 1952b.]

1969c. "Mathematical Linguistics". *The Mathematical Sciences* ed. by the Committee on Support of Research in the Mathematical Sciences (COSRIMS), with the collaboration of George A. W. Boehm, 190–196. Cambridge, Mass.: MIT Press.

1970a. *Papers in Structural and Transformational Linguistics*. [Ed. by Henry Hiż.] Dordrecht/Holland: D. Reidel., x, 850 pp.

[Collection of 37 papers originally published between 1940–1969. These are organized under the following headings: 1, "Structural Linguistics, 1: Methods"; 2, "Structural Linguistics, 2: Linguistic structures"; 3, "String Analysis and Computa-

tion"; 4, "Discourse Analysis"; 5, "Transformations", and 6, "About Linguistics". "Preface" (v–vii).]
Reviewed by
Ferenc Kiefer in *Statistical Methods in Linguistics* 7: 60–62 (Stockholm, 1971);
Michael B[enedict] Kac in *Language* 49: 466–473 (1973).

1970b. "La structure distributionnelle". *Analyse distributionnelle et structurale* ed. by Jean Dubois & Françoise Dubois-Charlier (= *Langages*, No. 20), 14–34. Paris: Didier / Larousse. [Transl. of item 1954b.]

1970c. "New Views of Language". Manuscript. (Published in Harris 1972: 242–248, with an introd. in German by the ed. [241–242].)

1971. *Structures mathématiques du langage*. Transl. into French by Catherine Fuchs. (= *Monographies de Linguistique mathématique*, 3.) Paris: Dunod, 248 pp. [Transl. of item 1968a.]
Reviewed by
Yves Gentilhomme in *Bulletin de la Société de Linguistique de Paris* 69.2: 37–53 (1974).

1972. *Transformationelle Analyse: Die Transformationstheorie von Zellig Harris und ihre Entwicklung / Transformational Analysis: The transformational theory of Zellig Harris and its development*. Ed. by Senta Plötz. (= *Linguistische Forschungen*, 8.) Frankfurt/Main: Athenäum-Verlag, viii, 511 pp.
[Reprint of items 1964b (57–75), 1957 (78–104), 1965 (108–154), 1969a (158–240) —revised by the author in 1972, and 1970c (242–248), each introduced, in German, by the ed. (55–57, 76–78, 105–108, 155–157, and 241–242, respectively.]

1973a. "Les deux systèmes de grammaire: Prédicat et paraphrase". *Langages* No. 29: 55–81. [Partial transl., by Danielle Leeman, of item 1969a.]

1973b. Review of *A Leonard Bloomfield Anthology* ed. by Charles F. Hockett (Bloomington & London: Indiana University Press, 1970). *IJAL* 39.4: 252–255.

1976a. "A Theory of Language Structure". *American Philosophical Quarterly* 13: 237–255. (Repr. in Harris 1981: 352–376.)
[Theory of the structure and information of sentences.]

1976b. "On a Theory of Language". *Journal of Philosophy* 73: 253–276. (Excerpted in Harris 1981: 377–391.)

1976c. *Notes du cours de syntaxe*. Transl. and presented by Maurice Gross. Paris: Éditions du Seuil, 236 p. [Transl. of lectures on English syntax given at the Département de Linguistique, University de Paris–Vincennes, 1973–1974.]
Reviewed by
G. L[urquin] in *Le Langage et l'Homme* 31: 114–115 (1976);
Claude Hagège in *Bulletin de la Société de Linguistique de Paris* 72.2: 35–37 (1974);
Riccardo Ambrosini in *Studi e Saggi Linguistici* 17: 309–340 (1977).

1976d. "Morphemalternanten in der linguistischen Analyse". *Beschreibungsmethoden des amerikanischen Strukturalismus* ed. by Elisabeth Bense, Peter Eisenberg & Hartmut Haberland, 129–143. München: Max Hueber. [Transl. by Elisabeth Bense of item 1942a.]

1976e. "Vom Morphem zur Äußerung". *Ibid.*, 181–210. [Transl., by Dietmar Rösler, of item 1946b.]

1976f. "Textanalyse". *Ibid.*, 261–298. [Transl., by Peter Eisenberg, of item 1952b.]

1978a. "Grammar on Mathematical Principles". *Journal of Linguistics* 14:1–20. (Repr. in Harris 1981:392–411.)
["Given as a lecture in Somerville College, Oxford, 16 March 1977".]

1978b. "Operator-Grammar of English". *Lingvisticae Investigationes* 2:55–92. (Excerpted in Harris 1981:412–435.)

1978c. "The Interrogative in a Syntactic Framework". *Questions* ed. by Henry Hiż (= *Synthese Language Library*, 1), 1–35. Dordrecht/Holland: D. Reidel.

1979a. "Założenia metodologiczne językoznawstwa strukturalnego [The methodological basis of structural linguistics]". *Językoznawstwo strukturalne: Wybór tekstów* ed. by Halina Kurkowska & Adam Weinsberg, 158–174. Warsaw: Państwowe Wydawnictwo Naukowe, 274 pp. [Polish transl., by the first editor, of Harris (1951a:4–24), "Methodological Preliminaries".]

1979b. "Mathematical Analysis of Language". Paper delivered to the 6th International Congress on Logic, Methodology, and the Philosophy of Science, held in Hanover, Germany, August 1979. Unpublished.

1981. *Papers on Syntax*. Ed. by Henry Hiż. (= *Synthese Language Library*, 14.) Dordrecht/Holland: D. Reidel, vii, 479 pp.
[Collection of 16 previously published papers, organized under 3 sections: I, "Structural Analysis"; II, "Transformational Analysis", and III, "Operator Grammar". Index (437–479).]

1982a. *A Grammar of English on Mathematical Principles*. New York: John Wiley & Sons, xvi, 429 pp.
Reviewed by
William Frawley in *Language* 60.1:150–152 (1984);
Frank Heny in *Journal of Linguistics* 20.1:181–188 (1984);
Bruce E. Nevin in *Computational Linguistics* 10:3/4:203–211 (1984);
Eric S. Wheeler in *Computers in the Humanities* 17.3:88–92 (1984).

1982b. "Discourse and Sublanguage". *Sublanguage: Studies of language in restricted semantic domains* ed. by Richard Kittredge & John Lehrberger, 231–236. Berlin: Walter de Gruyter.

1985. "On Grammars of Science". *Linguistics and Philosophy: Essays in honor of Rulon S. Wells* ed. by Adam Makkai & Alan K. Melby (= *Current Issues in Linguistic Theory*, 42), 139–148. Amsterdam & Philadelphia: John Benjamins.

1987. "The Structure of Science Information". Paper submitted to the journal *Science*, but rejected by the editor, allegedly because it contained no reference to Chomsky. Unpublished.

1988a. *Language and Information*. (= *Bampton Lectures in America*, 28.) New York: Columbia University Press, ix, 120 pp.
[Revised version of lectures given at Columbia University, New York City, in Oct. 1986. — 1, "A Formal Theory of Syntax"; 2, "Scientific Sub-Languages"; 3, "Information", and 4, "The Nature of Language".]
Reviewed by
P[eter] H[ugoe] Matthews in *Times Literary Supplement* (London, 23–29 Dec. 1988), with the title "Saying Something Simple".

1988b. (Together with Paul Mattick, Jr.) "Scientific Sublanguages and the Prospects for a Global Language of Science". *Annals of the American Association of Philosophy and Social Sciences* No. 495: 73–83.

1989. (Together with Michael Gottfried, Thomas Ryckman, Paul Mattick, Jr., Anne Daladier, Tzvee N. Harris & Suzanna Harris.) *The Form of Information in Science: Analysis of an immunology sublanguage*. Preface by Hilary Putnam. (= *Boston Studies in the Philosophy of Science*, 104.) Dordrecht/Holland & Boston: Kluwer Academic Publishers, xvii, 590 pp.

1990. "La genèse de l'analyse des transformations et de la métalangue". *Langages* No. 99 (Sept. 1990), 9–19. [Transl. of item 2002 by Anne Daladier.]

1991. *A Theory of Language and Information: A mathematical approach*. Oxford & New York: Clarendon Press, xii, 428 pp.; illustr.

Reviewed by

Jorge Baptista in *Revista da Faculdade de Letras* 15 (5ª, Série): 203–205. Lisboa: Faculdade de Letras da Universidade de Lisboa (FLUL);

D. Terence Langendoen in *Language* 70.3: 585–588 (1994).

1997. *The Transformation of Capitalist Society*. Foreword by Wolf V. Heydebrand [xi-xiii]. Baltimore, Md.: Rowman & Littlefield, xvi, 244 pp. (and an unnumbered page "About the Author").

["On Behalf of the Author" (signed by Murray Eden, William M. Evan, Seymour Melman) concludes with the sentence "Several of his old friends collaborated in preparing the manuscript for publication." "Preface" by Zellig S. Harris (xv–xvi). Contents: Chap. 1, "Overview: The possibilities of change" (1–7), with Appendix "Criticizing capitalist society" (9–10); Chap. 2, "Basic terms in describing society" (11–22); Chap. 3, "Capitalist decisions on production" (23–42); Chap. 4, "Considerations in analyzing social change" (43–55); Chap. 5, "Potentially post-capitalist developments" (57–86); Chap. 6, "How capitalism began" (87–112); Chap. 7, "Self-governed production" (113–182); Chap. 8, "In the aftermath of Soviet communism" (183–208), and Chap. 9, "Intervening in the historical process" (209–233). Index (235–244).]

Reviewed by

Peter Franz in *The European Legacy* 3: 112–113 (1998).

2002. "The Background of Transformational and Metalanguage Analysis". *The Legacy of Zellig Harris: Language and information into the 21st century*, Volume I: *Philosophy of science, syntax, and semantics* ed. by Bruce E. Nevin (= *Current Issues in Linguistic Theory*, 228), 1–14. Amsterdam & Philadelphia: John Benjamins.

[Publication of original English text with portions not included in item 1990. — Proposes a method for analyzing extended discourse, with sample analyses from Hidatsa, a Siouan language spoken in North Dakota.]

Name index

Subject index

CURRENT ISSUES IN LINGUISTIC THEORY

E. F. K. Koerner, Editor
Institut für Sprachwissenschaft, Universität zu Köln
D-50923 KÖLN, Germany
efk.koerner@uni-koeln.de

The *Current Issues in Linguistic Theory* (CILT) series is a theory-oriented series which welcomes contributions from scholars who have significant proposals to make towards the advancement of our understanding of language, its structure, functioning and development. CILT has been established in order to provide a forum for the presentation and discussion of linguistic opinions of scholars who do not necessarily accept the prevailing mode of thought in linguistic science. It offers an alternative outlet for meaningful contributions to the current linguistic debate, and furnishes the diversity of opinion which a healthy discipline must have. In this series the following volumes have been published thus far or are scheduled for publication:

1. KOERNER, Konrad (ed.): *The Transformational-Generative Paradigm and Modern Linguistic Theory*. 1975.
2. WEIDERT, Alfons: *Componential Analysis of Lushai Phonology*. 1975.
3. MAHER, J. Peter: *Papers on Language Theory and History I: Creation and Tradition in Language. Foreword by Raimo Anttila.* 1979.
4. HOPPER, Paul J. (ed.): *Studies in Descriptive and Historical Linguistics. Festschrift for Winfred P. Lehmann.* 1977.
5. ITKONEN, Esa: *Grammatical Theory and Metascience: A critical investigation into the methodological and philosophical foundations of 'autonomous' linguistics.* 1978.
6. ANTTILA, Raimo: *Historical and Comparative Linguistics.* 1989.
7. MEISEL, Jürgen M. & Martin D. PAM (eds): *Linear Order and Generative Theory*. 1979.
8. WILBUR, Terence H.: *Prolegomena to a Grammar of Basque.* 1979.
9. HOLLIEN, Harry & Patricia (eds): *Current Issues in the Phonetic Sciences. Proceedings of the IPS-77 Congress, Miami Beach, Florida, 17-19 December 1977.* 1979.
10. PRIDEAUX, Gary D. (ed.): *Perspectives in Experimental Linguistics. Papers from the University of Alberta Conference on Experimental Linguistics, Edmonton, 13-14 Oct. 1978.* 1979.
11. BROGYANYI, Bela (ed.): *Studies in Diachronic, Synchronic, and Typological Linguistics: Festschrift for Oswald Szemérenyi on the Occasion of his 65th Birthday.* 1979.
12. FISIAK, Jacek (ed.): *Theoretical Issues in Contrastive Linguistics.* 1981. Out of print
13. MAHER, J. Peter, Allan R. BOMHARD & Konrad KOERNER (eds): *Papers from the Third International Conference on Historical Linguistics, Hamburg, August 22-26 1977.* 1982.
14. TRAUGOTT, Elizabeth C., Rebecca LaBRUM & Susan SHEPHERD (eds): *Papers from the Fourth International Conference on Historical Linguistics, Stanford, March 26-30 1979.* 1980.
15. ANDERSON, John (ed.): *Language Form and Linguistic Variation. Papers dedicated to Angus McIntosh.* 1982.
16. ARBEITMAN, Yoël L. & Allan R. BOMHARD (eds): *Bono Homini Donum: Essays in Historical Linguistics, in Memory of J.Alexander Kerns.* 1981.
17. LIEB, Hans-Heinrich: *Integrational Linguistics. 6 volumes. Vol. II-VI n.y.p.* 1984/93.
18. IZZO, Herbert J. (ed.): *Italic and Romance. Linguistic Studies in Honor of Ernst Pulgram.* 1980.
19. RAMAT, Paolo et al. (eds): *Linguistic Reconstruction and Indo-European Syntax. Proceedings of the Colloquium of the 'Indogermanischhe Gesellschaft'. University of Pavia, 6-7 September 1979.* 1980.
20. NORRICK, Neal R.: *Semiotic Principles in Semantic Theory*. 1981.
21. AHLQVIST, Anders (ed.): *Papers from the Fifth International Conference on Historical Linguistics, Galway, April 6-10 1981.* 1982.

22. UNTERMANN, Jürgen & Bela BROGYANYI (eds): *Das Germanische und die Rekonstruktion der Indogermanischen Grundsprache. Akten des Freiburger Kolloquiums der Indogermanischen Gesellschaft, Freiburg, 26-27 Februar 1981.* 1984.
23. DANIELSEN, Niels: *Papers in Theoretical Linguistics. Edited by Per Baerentzen.* 1992.
24. LEHMANN, Winfred P. & Yakov MALKIEL (eds): *Perspectives on Historical Linguistics. Papers from a conference held at the meeting of the Language Theory Division, Modern Language Assn., San Francisco, 27-30 December 1979.* 1982.
25. ANDERSEN, Paul Kent: *Word Order Typology and Comparative Constructions.* 1983.
26. BALDI, Philip (ed.): *Papers from the XIIth Linguistic Symposium on Romance Languages, Univ. Park, April 1-3, 1982.* 1984.
27. BOMHARD, Alan R.: *Toward Proto-Nostratic. A New Approach to the Comparison of Proto-Indo-European and Proto-Afroasiatic. Foreword by Paul J. Hopper.* 1984.
28. BYNON, James (ed.): *Current Progress in Afro-Asiatic Linguistics: Papers of the Third International Hamito-Semitic Congress, London, 1978.* 1984.
29. PAPROTTÉ, Wolf & René DIRVEN (eds): *The Ubiquity of Metaphor: Metaphor in language and thought.* 1985 (publ. 1986).
30. HALL, Robert A. Jr.: *Proto-Romance Morphology. = Comparative Romance Grammar, vol. III.* 1984.
31. GUILLAUME, Gustave: *Foundations for a Science of Language.*
32. COPELAND, James E. (ed.): *New Directions in Linguistics and Semiotics.* Co-edition with Rice University Press who hold exclusive rights for US and Canada. 1984.
33. VERSTEEGH, Kees: *Pidginization and Creolization. The Case of Arabic.* 1984.
34. FISIAK, Jacek (ed.): *Papers from the VIth International Conference on Historical Linguistics, Poznan, 22-26 August. 1983.* 1985.
35. COLLINGE, N.E.: *The Laws of Indo-European.* 1985.
36. KING, Larry D. & Catherine A. MALEY (eds): *Selected papers from the XIIIth Linguistic Symposium on Romance Languages, Chapel Hill, N.C., 24-26 March 1983.* 1985.
37. GRIFFEN, T.D.: *Aspects of Dynamic Phonology.* 1985.
38. BROGYANYI, Bela & Thomas KRÖMMELBEIN (eds): *Germanic Dialects:Linguistic and Philological Investigations.* 1986.
39. BENSON, James D., Michael J. CUMMINGS, & William S. GREAVES (eds): *Linguistics in a Systemic Perspective.* 1988.
40. FRIES, Peter Howard (ed.) in collaboration with Nancy M. Fries: *Toward an Understanding of Language: Charles C. Fries in Perspective.* 1985.
41. EATON, Roger, et al. (eds): *Papers from the 4th International Conference on English Historical Linguistics, April 10-13, 1985.* 1985.
42. MAKKAI, Adam & Alan K. MELBY (eds): *Linguistics and Philosophy. Festschrift for Rulon S. Wells.* 1985 (publ. 1986).
43. AKAMATSU, Tsutomu: *The Theory of Neutralization and the Archiphoneme in Functional Phonology.* 1988.
44. JUNGRAITHMAYR, Herrmann & Walter W. MUELLER (eds): *Proceedings of the Fourth International Hamito-Semitic Congress.* 1987.
45. KOOPMAN, W.F., F.C. Van der LEEK , O. FISCHER & R. EATON (eds): *Explanation and Linguistic Change.* 1986
46. PRIDEAUX, Gary D. & William J. BAKER: *Strategies and Structures: The processing of relative clauses.* 1987.
47. LEHMANN, Winfred P. (ed.): *Language Typology 1985. Papers from the Linguistic Typology Symposium, Moscow, 9-13 Dec. 1985.* 1986.
48. RAMAT, Anna G., Onofrio CARRUBA and Giuliano BERNINI (eds): *Papers from the 7th International Conference on Historical Linguistics.* 1987.
49. WAUGH, Linda R. and Stephen RUDY (eds): *New Vistas in Grammar: Invariance and*

Variation. Proceedings of the Second International Roman Jakobson Conference, New York University, Nov.5-8, 1985. 1991.

50. RUDZKA-OSTYN, Brygida (ed.): *Topics in Cognitive Linguistics.* 1988.
51. CHATTERJEE, Ranjit: *Aspect and Meaning in Slavic and Indic. With a foreword by Paul Friedrich.* 1989.
52. FASOLD, Ralph W. & Deborah SCHIFFRIN (eds): *Language Change and Variation.* 1989.
53. SANKOFF, David: *Diversity and Diachrony.* 1986.
54. WEIDERT, Alfons: *Tibeto-Burman Tonology. A comparative analysis.* 1987
55. HALL, Robert A. Jr.: *Linguistics and Pseudo-Linguistics.* 1987.
56. HOCKETT, Charles F.: *Refurbishing our Foundations. Elementary linguistics from an advanced point of view.* 1987.
57. BUBENIK, Vít: *Hellenistic and Roman Greece as a Sociolinguistic Area.* 1989.
58. ARBEITMAN, Yoël. L. (ed.): *Fucus: A Semitic/Afrasian Gathering in Remembrance of Albert Ehrman.* 1988.
59. VAN VOORST, Jan: *Event Structure.* 1988.
60. KIRSCHNER, Carl & Janet DECESARIS (eds): *Studies in Romance Linguistics. Selected Proceedings from the XVII Linguistic Symposium on Romance Languages.* 1989.
61. CORRIGAN, Roberta L., Fred ECKMAN & Michael NOONAN (eds): *Linguistic Categorization. Proceedings of an International Symposium in Milwaukee, Wisconsin, April 10-11, 1987.* 1989.
62. FRAJZYNGIER, Zygmunt (ed.): *Current Progress in Chadic Linguistics. Proceedings of the International Symposium on Chadic Linguistics, Boulder, Colorado, 1-2 May 1987.* 1989.
63. EID, Mushira (ed.): *Perspectives on Arabic Linguistics I. Papers from the First Annual Symposium on Arabic Linguistics.* 1990.
64. BROGYANYI, Bela (ed.): *Prehistory, History and Historiography of Language, Speech, and Linguistic Theory. Papers in honor of Oswald Szemérenyi I.* 1992.
65. ADAMSON, Sylvia, Vivien A. LAW, Nigel VINCENT and Susan WRIGHT (eds): *Papers from the 5th International Conference on English Historical Linguistics.* 1990.
66. ANDERSEN, Henning and Konrad KOERNER (eds): *Historical Linguistics 1987.Papers from the 8th International Conference on Historical Linguistics,Lille, August 30-Sept., 1987.* 1990.
67. LEHMANN, Winfred P. (ed.): *Language Typology 1987. Systematic Balance in Language. Papers from the Linguistic Typology Symposium, Berkeley, 1-3 Dec 1987.* 1990.
68. BALL, Martin, James FIFE, Erich POPPE &Jenny ROWLAND (eds): *Celtic Linguistics/ Ieithyddiaeth Geltaidd. Readings in the Brythonic Languages. Festschrift for T. Arwyn Watkins.* 1990.
69. WANNER, Dieter and Douglas A. KIBBEE (eds): *New Analyses in Romance Linguistics. Selected papers from the Linguistic Symposium on Romance Languages XVIIII, Urbana-Champaign, April 7-9, 1988.* 1991.
70. JENSEN, John T.: *Morphology. Word structure in generative grammar.* 1990.
71. O'GRADY, William: *Categories and Case. The sentence structure of Korean.* 1991.
72. EID, Mushira and John MCCARTHY (eds): *Perspectives on Arabic Linguistics II. Papers from the Second Annual Symposium on Arabic Linguistics.* 1990.
73. STAMENOV, Maxim (ed.): *Current Advances in Semantic Theory.* 1991.
74. LAEUFER, Christiane and Terrell A. MORGAN (eds): *Theoretical Analyses in Romance Linguistics.* 1991.
75. DROSTE, Flip G. and John E. JOSEPH (eds): *Linguistic Theory and Grammatical Description. Nine Current Approaches.* 1991.
76. WICKENS, Mark A.: *Grammatical Number in English Nouns. An empirical and theoretical account.* 1992.
77. BOLTZ, William G. and Michael C. SHAPIRO (eds): *Studies in the Historical Phonology of Asian Languages.* 1991.

78. KAC, Michael: *Grammars and Grammaticality.* 1992.
79. ANTONSEN, Elmer H. and Hans Henrich HOCK (eds): *STAEF-CRAEFT: Studies in Germanic Linguistics. Select papers from the First and Second Symposium on Germanic Linguistics, University of Chicago, 24 April 1985, and Univ. of Illinois at Urbana-Champaign, 3-4 Oct. 1986.* 1991.
80. COMRIE, Bernard and Mushira EID (eds): *Perspectives on Arabic Linguistics III. Papers from the Third Annual Symposium on Arabic Linguistics.* 1991.
81. LEHMANN, Winfred P. and H.J. HEWITT (eds): *Language Typology 1988. Typological Models in the Service of Reconstruction.* 1991.
82. VAN VALIN, Robert D. (ed.): *Advances in Role and Reference Grammar.* 1992.
83. FIFE, James and Erich POPPE (eds): *Studies in Brythonic Word Order.* 1991.
84. DAVIS, Garry W. and Gregory K. IVERSON (eds): *Explanation in Historical Linguistics.* 1992.
85. BROSELOW, Ellen, Mushira EID and John McCARTHY (eds): *Perspectives on Arabic Linguistics IV. Papers from the Annual Symposium on Arabic Linguistics.* 1992.
86. KESS, Joseph F.: *Psycholinguistics. Psychology, linguistics, and the study of natural language.* 1992.
87. BROGYANYI, Bela and Reiner LIPP (eds): *Historical Philology: Greek, Latin, and Romance. Papers in honor of Oswald Szemerényi II.* 1992.
88. SHIELDS, Kenneth: *A History of Indo-European Verb Morphology.* 1992.
89. BURRIDGE, Kate: *Syntactic Change in Germanic. A study of some aspects of language change in Germanic with particular reference to Middle Dutch.* 1992.
90. KING, Larry D.: *The Semantic Structure of Spanish. Meaning and grammatical form.* 1992.
91. HIRSCHBÜHLER, Paul and Konrad KOERNER (eds): *Romance Languages and Modern Linguistic Theory. Selected papers from the XX Linguistic Symposium on Romance Languages,University of Ottawa, April 10-14, 1990.* 1992.
92. POYATOS, Fernando: *Paralanguage: A linguistic and interdisciplinary approach to interactive speech and sounds.* 1992.
93. LIPPI-GREEN, Rosina (ed.): *Recent Developments in Germanic Linguistics.* 1992.
94. HAGÈGE, Claude: *The Language Builder. An essay on the human signature in linguistic morphogenesis.* 1992.
95. MILLER, D. Gary: *Complex Verb Formation.* 1992.
96. LIEB, Hans-Heinrich (ed.): *Prospects for a New Structuralism.* 1992.
97. BROGYANYI, Bela & Reiner LIPP (eds): *Comparative-Historical Linguistics: Indo-European and Finno-Ugric. Papers in honor of Oswald Szemerényi III.* 1992.
98. EID, Mushira & Gregory K. IVERSON: *Principles and Prediction: The analysis of natural language.* 1993.
99. JENSEN, John T.: *English Phonology.* 1993.
100. MUFWENE, Salikoko S. and Lioba MOSHI (eds): *Topics in African Linguistics. Papers from the XXI Annual Conference on African Linguistics, University of Georgia, April 1990.* 1993.
101. EID, Mushira & Clive HOLES (eds): *Perspectives on Arabic Linguistics V. Papers from the Fifth Annual Symposium on Arabic Linguistics.* 1993.
102. DAVIS, Philip W. (ed.): *Alternative Linguistics. Descriptive and theoretical Modes.* 1995.
103. ASHBY, William J., Marianne MITHUN, Giorgio PERISSINOTTO and Eduardo RAPOSO: *Linguistic Perspectives on Romance Languages. Selected papers from the XXI Linguistic Symposium on Romance Languages, Santa Barbara, February 21-24, 1991.* 1993.
104. KURZOVÁ, Helena: *From Indo-European to Latin. The evolution of a morphosyntactic type.* 1993.
105. HUALDE, José Ignacio and Jon ORTIZ DE URBANA (eds): *Generative Studies in Basque Linguistics.* 1993.
106. AERTSEN, Henk and Robert J. JEFFERS (eds): *Historical Linguistics 1989. Papers from the 9th International Conference on Historical Linguistics, New Brunswick, 14-18 August 1989.* 1993.

107. MARLE, Jaap van (ed.): *Historical Linguistics 1991. Papers from the 10th International Conference on Historical Linguistics, Amsterdam, August 12-16, 1991.* 1993.
108. LIEB, Hans-Heinrich: *Linguistic Variables. Towards a unified theory of linguistic variation.* 1993.
109. PAGLIUCA, William (ed.): *Perspectives on Grammaticalization.* 1994.
110. SIMONE, Raffaele (ed.): *Iconicity in Language.* 1995.
111. TOBIN, Yishai: *Invariance, Markedness and Distinctive Feature Analysis. A contrastive study of sign systems in English and Hebrew.* 1994.
112. CULIOLI, Antoine: *Cognition and Representation in Linguistic Theory. Translated, edited and introduced by Michel Liddle.* 1995.
113. FERNÁNDEZ, Francisco, Miguel FUSTER and Juan Jose CALVO (eds): *English Historical Linguistics 1992. Papers from the 7th International Conference on English Historical Linguistics, Valencia, 22-26 September 1992.*1994.
114. EGLI, U., P. PAUSE, Chr. SCHWARZE, A. von STECHOW, G. WIENOLD (eds): *Lexical Knowledge in the Organisation of Language.* 1995.
115. EID, Mushira, Vincente CANTARINO and Keith WALTERS (eds): *Perspectives on Arabic Linguistics. Vol. VI. Papers from the Sixth Annual Symposium on Arabic Linguistics.* 1994.
116. MILLER, D. Gary: *Ancient Scripts and Phonological Knowledge.* 1994.
117. PHILIPPAKI-WARBURTON, I., K. NICOLAIDIS and M. SIFIANOU (eds): *Themes in Greek Linguistics. Papers from the first International Conference on Greek Linguistics, Reading, September 1993.* 1994.
118. HASAN, Ruqaiya and Peter H. FRIES (eds): *On Subject and Theme. A discourse functional perspective.* 1995.
119. LIPPI-GREEN, Rosina: *Language Ideology and Language Change in Early Modern German. A sociolinguistic study of the consonantal system of Nuremberg.* 1994.
120. STONHAM, John T. : *Combinatorial Morphology.* 1994.
121. HASAN, Ruqaiya, Carmel CLORAN and David BUTT (eds): *Functional Descriptions. Theorie in practice.* 1996.
122. SMITH, John Charles and Martin MAIDEN (eds): *Linguistic Theory and the Romance Languages.* 1995.
123. AMASTAE, Jon, Grant GOODALL, Mario MONTALBETTI and Marianne PHINNEY: *Contemporary Research in Romance Linguistics. Papers from the XXII Linguistic Symposium on Romance Languages, El Paso//Juárez, February 22-24, 1994.* 1995.
124. ANDERSEN, Henning: *Historical Linguistics 1993. Selected papers from the 11th International Conference on Historical Linguistics, Los Angeles, 16-20 August 1993.* 1995.
125. SINGH, Rajendra (ed.): *Towards a Critical Sociolinguistics.* 1996.
126. MATRAS, Yaron (ed.): *Romani in Contact. The history, structure and sociology of a language.* 1995.
127. GUY, Gregory R., Crawford FEAGIN, Deborah SCHIFFRIN and John BAUGH (eds): *Towards a Social Science of Language. Papers in honor of William Labov. Volume 1: Variation and change in language and society.* 1996.
128. GUY, Gregory R., Crawford FEAGIN, Deborah SCHIFFRIN and John BAUGH (eds): *Towards a Social Science of Language. Papers in honor of William Labov. Volume 2: Social interaction and discourse structures.* 1997.
129. LEVIN, Saul: *Semitic and Indo-European: The Principal Etymologies. With observations on Afro-Asiatic.* 1995.
130. EID, Mushira (ed.) *Perspectives on Arabic Linguistics. Vol. VII. Papers from the Seventh Annual Symposium on Arabic Linguistics.* 1995.
131. HUALDE, Jose Ignacio, Joseba A. LAKARRA and R.L. Trask (eds): *Towards a History of the Basque Language.* 1995.
132. HERSCHENSOHN, Julia: *Case Suspension and Binary Complement Structure in French.* 1996.

133. ZAGONA, Karen (ed.): *Grammatical Theory and Romance Languages. Selected papers from the 25th Linguistic Symposium on Romance Languages (LSRL XXV) Seattle, 2-4 March 1995.* 1996.
134. EID, Mushira (ed.): *Perspectives on Arabic Linguistics Vol. VIII. Papers from the Eighth Annual Symposium on Arabic Linguistics.* 1996.
135. BRITTON Derek (ed.): *Papers from the 8th International Conference on English Historical Linguistics.* 1996.
136. MITKOV, Ruslan and Nicolas NICOLOV (eds): *Recent Advances in Natural Language Processing.* 1997.
137. LIPPI-GREEN, Rosina and Joseph C. SALMONS (eds): *Germanic Linguistics. Syntactic and diachronic.* 1996.
138. SACKMANN, Robin (ed.): *Theoretical Linguistics and Grammatical Description.* 1996.
139. BLACK, James R. and Virginia MOTAPANYANE (eds): *Microparametric Syntax and Dialect Variation.* 1996.
140. BLACK, James R. and Virginia MOTAPANYANE (eds): *Clitics, Pronouns and Movement.* 1997.
141. EID, Mushira and Dilworth PARKINSON (eds): *Perspectives on Arabic Linguistics Vol. IX. Papers from the Ninth Annual Symposium on Arabic Linguistics, Georgetown University, Washington D.C., 1995.* 1996.
142. JOSEPH, Brian D. and Joseph C. SALMONS (eds): *Nostratic. Sifting the evidence.* 1998.
143. ATHANASIADOU, Angeliki and René DIRVEN (eds): *On Conditionals Again.* 1997.
144. SINGH, Rajendra (ed): *Trubetzkoy's Orphan. Proceedings of the Montréal Roundtable "Morphophonology: contemporary responses (Montréal, October 1994).* 1996.
145. HEWSON, John and Vit BUBENIK: *Tense and Aspect in Indo-European Languages. Theory, typology, diachrony.* 1997.
146. HINSKENS, Frans, Roeland VAN HOUT and W. Leo WETZELS (eds): *Variation, Change, and Phonological Theory.* 1997.
147. HEWSON, John: *The Cognitive System of the French Verb.* 1997.
148. WOLF, George and Nigel LOVE (eds): *Linguistics Inside Out. Roy Harris and his critics.* 1997.
149. HALL, T. Alan: *The Phonology of Coronals.* 1997.
150. VERSPOOR, Marjolijn, Kee Dong LEE and Eve SWEETSER (eds): *Lexical and Syntactical Constructions and the Construction of Meaning. Proceedings of the Bi-annual ICLA meeting in Albuquerque, July 1995.* 1997.
151. LIEBERT, Wolf-Andreas, Gisela REDEKER and Linda WAUGH (eds): *Discourse and Perspectives in Cognitive Linguistics.* 1997.
152. HIRAGA, Masako, Chris SINHA and Sherman WILCOX (eds): *Cultural, Psychological and Typological Issues in Cognitive Linguistics.* 1999.
153. EID, Mushira and Robert R. RATCLIFFE (eds): *Perspectives on Arabic Linguistics Vol. X. Papers from the Tenth Annual Symposium on Arabic Linguistics, Salt Lake City, 1996.* 1997.
154. SIMON-VANDENBERGEN, Anne-Marie, Kristin DAVIDSE and Dirk NOËL (eds): *Reconnecting Language. Morphology and Syntax in Functional Perspectives.* 1997.
155. FORGET, Danielle, Paul HIRSCHBÜHLER, France MARTINEAU and María-Luisa RIVERO (eds): *Negation and Polarity. Syntax and semantics. Selected papers from the Colloquium Negation: Syntax and Semantics. Ottawa, 11-13 May 1995.* 1997.
156. MATRAS, Yaron, Peter BAKKER and Hristo KYUCHUKOV (eds): *The Typology and Dialectology of Romani.* 1997.
157. LEMA, José and Esthela TREVIÑO (eds): *Theoretical Analyses on Romance Languages. Selected papers from the 26th Linguistic Symposium on Romance Languages (LSRL XXVI), Mexico City, 28-30 March, 1996.* 1998.
158. SÁNCHEZ MACARRO, Antonia and Ronald CARTER (eds): *Linguistic Choice across Genres. Variation in spoken and written English.* 1998.

159. JOSEPH, Brian D., Geoffrey C. HORROCKS and Irene PHILIPPAKI-WARBURTON (eds): *Themes in Greek Linguistics II.* 1998.
160. SCHWEGLER, Armin, Bernard TRANEL and Myriam URIBE-ETXEBARRIA (eds): *Romance Linguistics: Theoretical Perspectives. Selected papers from the 27th Linguistic Symposium on Romance Languages (LSRL XXVII), Irvine, 20-22 February, 1997.* 1998.
161. SMITH, John Charles and Delia BENTLEY (eds): *Historical Linguistics 1995. Volume 1: Romance and general linguistics.* 2000.
162. HOGG, Richard M. and Linda van BERGEN (eds): *Historical Linguistics 1995. Volume 2: Germanic linguistics.Selected papers from the 12th International Conference on Historical Linguistics, Manchester, August 1995.* 1998.
163. LOCKWOOD, David G., Peter H. FRIES and James E. COPELAND (eds): *Functional Approaches to Language, Culture and Cognition.* 2000.
164. SCHMID, Monika, Jennifer R. AUSTIN and Dieter STEIN (eds): *Historical Linguistics 1997. Selected papers from the 13th International Conference on Historical Linguistics, Düsseldorf, 10-17 August 1997.* 1998.
165. BUBENÍK, Vit: *A Historical Syntax of Late Middle Indo-Aryan (Apabhramśa).* 1998.
166. LEMMENS, Maarten: *Lexical Perspectives on Transitivity and Ergativity. Causative constructions in English.* 1998.
167. BENMAMOUN, Elabbas, Mushira EID and Niloofar HAERI (eds): *Perspectives on Arabic Linguistics Vol. XI. Papers from the Eleventh Annual Symposium on Arabic Linguistics, Atlanta, 1997.* 1998.
168. RATCLIFFE, Robert R.: *The "Broken" Plural Problem in Arabic and Comparative Semitic. Allomorphy and analogy in non-concatenative morphology.* 1998.
169. GHADESSY, Mohsen (ed.): *Text and Context in Functional Linguistics.* 1999.
170. LAMB, Sydney M.: *Pathways of the Brain. The neurocognitive basis of language.* 1999.
171. WEIGAND, Edda (ed.): *Contrastive Lexical Semantics.* 1998.
172. DIMITROVA-VULCHANOVA, Mila and Lars HELLAN (eds): *Topics in South Slavic Syntax and Semantics.* 1999.
173. TREVIÑO, Esthela and José LEMA (eds): Semantic Issues in Romance Syntax. 1999.
174. HALL, T. Alan and Ursula KLEINHENZ (eds): *Studies on the Phonological Word.* 1999.
175. GIBBS, Ray W. and Gerard J. STEEN (eds): *Metaphor in Cognitive Linguistics. Selected papers from the 5th International Cognitive Linguistics Conference, Amsterdam, 1997.* 2001.
176. VAN HOEK, Karen, Andrej KIBRIK and Leo NOORDMAN (eds): *Discourse in Cognitive Linguistics. Selected papers from the International Cognitive Linguistics Conference, Amsterdam, July 1997.* 1999.
177. CUYCKENS, Hubert and Britta ZAWADA (eds): *Polysemy in Cognitive Linguistics. Selected papers from the International Cognitive Linguistics Conference, Amsterdam, 1997.* 2001.
178. FOOLEN, Ad and Frederike van der LEEK (eds): *Constructions in Cognitive Linguistics. Selected papers from the Fifth International Cognitive Linguistic Conference, Amsterdam, 1997.* 2000.
179. RINI, Joel: *Exploring the Role of Morphology in the Evolution of Spanish.* 1999.
180. MEREU, Lunella (ed.): *Boundaries of Morphology and Syntax.* 1999.
181. MOHAMMAD, Mohammad A.: *Word Order, Agreement and Pronominalization in Standard and Palestinian Arabic.* 2000.
182. KENESEI, István (ed.): *Theoretical Issues in Eastern European Languages. Selected papers from the Conference on Linguistic Theory in Eastern European Languages (CLITE), Szeged, April 1998.* 1999.
183. CONTINI-MORAVA, Ellen and Yishai TOBIN (eds): *Between Grammar and Lexicon.* 2000.
184. SAGART, Laurent: *The Roots of Old Chinese.* 1999.
185. AUTHIER, J.-Marc, Barbara E. BULLOCK, Lisa A. REED (eds): *Formal Perspectives on Romance Linguistics. Selected papers from the 28th Linguistic Symposium on Romance Languages (LSRL XXVIII), University Park, 16-19 April 1998.* 1999.

186. MIŠESKA TOMIĆ, Olga and Milorad RADOVANOVIĆ (eds): *History and Perspectives of Language Study*. 2000.
187. FRANCO, Jon, Alazne LANDA and Juan MARTÍN (eds): *Grammatical Analyses in Basque and Romance Linguistics*. 1999.
188. VanNESS SIMMONS, Richard: *Chinese Dialect Classification. A comparative approach to Harngjou, Old Jintarn, and Common Northern Wu*. 1999.
189. NICHOLOV, Nicolas and Ruslan MITKOV (eds): *Recent Advances in Natural Language Processing II. Selected papers from RANLP '97*. 2000.
190. BENMAMOUN, Elabbas (ed.): *Perspectives on Arabic Linguistics Vol. XII. Papers from the Twelfth Annual Symposium on Arabic Linguistics*. 1999.
191. SIHLER, Andrew L.: *Language Change. An introduction*. 2000.
192. ALEXANDROVA, Galina M. and Olga ARNAUDOVA (eds.): *The Minimalist Parameter. Selected papers from the Open Linguistics Forum, Ottawa, 21-23 March 1997*. 2001.
193. KLAUSENBURGER, Jurgen: *Grammaticalization. Studies in Latin and Romance morphosyntax*. 2000.
194. COLEMAN, Julie and Christian J. KAY (eds): *Lexicology, Semantics and Lexicography. Selected papers from the Fourth G. L. Brook Symposium, Manchester, August 1998*. 2000.
195. HERRING, Susan C., Pieter van REENEN and Lene SCHØSLER (eds): *Textual Parameters in Older Languages*. 2000.
196. HANNAHS, S. J. and Mike DAVENPORT (eds): *Issues in Phonological Structure. Papers from an International Workshop*. 1999.
197. COOPMANS, Peter, Martin EVERAERT and Jane GRIMSHAW (eds): *Lexical Specification and Insertion*. 2000.
198. NIEMEIER, Susanne and René DIRVEN (eds): *Evidence for Linguistic Relativity*. 2000.
199. PÜTZ, Martin and Marjolijn H. VERSPOOR (eds): *Explorations in Linguistic Relativity*. 2000.
200. ANTTILA, Raimo: *Greek and Indo-European Etymology in Action. Proto-Indo-European *ag-*.2000.
201. DRESSLER, Wolfgang U., Oskar E. PFEIFFER, Markus PÖCHTRAGER and John R. RENNISON (eds.): *Morphological Analysis in Comparison*. 2000.
202. LECARME, Jacqueline, Jean LOWENSTAMM and Ur SHLONSKY (eds.): *Research in Afroasiatic Grammar. Papers from the Third conference on Afroasiatic Languages, Sophia Antipolis, 1996*. 2000.
203. NORRICK, Neal R.: *Conversational Narrative. Storytelling in everyday talk*. 2000.
204. DIRVEN, René, Bruce HAWKINS and Esra SANDIKCIOGLU (eds.): *Language and Ideology. Volume 1: cognitive theoretical approaches*. 2001.
205. DIRVEN, René, Roslyn FRANK and Cornelia ILIE (eds.): *Language and Ideology. Volume 2: cognitive descriptive approaches*. 2001.
206. FAWCETT, Robin: *A Theory of Syntax for Systemic-Functional Linguistics*. 2000.
207. SANZ, Montserrat: *Events and Predication. A new approach to syntactic processing in English and Spanish*. 2000.
208. ROBINSON, Orrin W.: *Whose German? The ach/ich alternation and related phenomena in 'standard' and 'colloquial'*. 2001.
209. KING, Ruth: *The Lexical Basis of Grammatical Borrowing. A Prince Edward Island French case study*. 2000.
210. DWORKIN, Steven N. and Dieter WANNER (eds.): *New Approaches to Old Problems. Issues in Romance historical linguistics*. 2000.
211. ELŠÍK, Viktor and Yaron MATRAS (eds.): *Grammatical Relations in Romani. The Noun Phrase*. 2000.
212. REPETTI, Lori (ed.): *Phonological Theory and the Dialects of Italy*. 2000.
213. SORNICOLA, Rosanna, Erich POPPE and Ariel SHISHA-HALEVY (eds.): *Stability, Variation and Change of Word-Order Patterns over Time*. 2000.

214. WEIGAND, Edda and Marcelo DASCAL (eds.): *Negotiation and Power in Dialogic Interaction.* 2001.
215. BRINTON, Laurel J.: *Historical Linguistics 1999. Selected papers from the 14th International Conference on Historical Linguistics, Vancouver, 9-13 August 1999.* 2001.
216. CAMPS, Joaquim and Caroline R. WILTSHIRE (eds.): *Romance Syntax, Semantics and L2 Acquisition. Selected papers from the 30th Linguistic Symposium on Romance Languages, Gainesville, Florida, February 2000.* 2001.
217. WILTSHIRE, Caroline R. and Joaquim CAMPS (eds.): *Romance Phonology and Variation. Selected papers from the 30th Linguistic Symposium on Romance Languages, Gainesville, Florida, February 2000.* 2002.
218. BENDJABALLAH, S., W.U. DRESSLER, O. PFEIFFER and M. VOEIKOVA (eds.): *Morphology 2000. Selected papers from the 9th Morphology Meeting, Vienna, 24–28 February 2000.* 2002.
219. ANDERSEN, Henning (ed.): *Actualization. Linguistic Change in Progress.* 2001.
220. SATTERFIELD, Teresa, Christina TORTORA and Diana CRESTI (eds.): *Current Issues in Romance Languages. Selected papers from the 29th Linguistic Symposium on Romance Languages (LSRL), Ann Arbor, 8-11 April 1999.* 2002.
221. D'HULST, Yves, Johan ROORYCK and Jan SCHROTEN (eds.): *Romance Languages and Linguistic Theory 1999. Selected papers from 'Going Romance' 1999, Leiden, 9-11 December.* 2001.
222. HERSCHENSOHN, Julia, Enrique MALLÉN and Karen ZAGONA (eds.): *Features and Interfaces in Romance. Essays in honor of Heles Contreras.* 2001.
223. FANEGO, Teresa, María José LÓPEZ-COUSO and Javier PÉREZ-GUERRA (eds.): *English Historical Syntax and Morphology. Selected papers from 11 ICEHL, Santiago de Compostela, 7-11 September 2000.* 2002.
224. FANEGO, Teresa, Belén MÉNDEZ-NAYA and Elena SEOANE (eds.): *Sounds, Words, Texts and Change. Selected papers from 11 ICEHL, Santiago de Compostela, 7-11 September 2000.* 2002
225. SHAHIN, Kimary N.: *Postvelar Harmony.* n.y.p.
226. LEVIN, Saul: *Semitic and Indo-European. Volume II: comparative morphology, syntax and phonetics; with observations on Afro-Asiatic.* 2002.
227. FAVA, Elisabetta (ed.): *Clinical Linguistics. Theory and applications in speech pathology and therapy.* 2002.
228. NEVIN, Bruce E. (ed.): *The Legacy of Zellig Harris. Language and information into the 21st century. Volume 1: philosophy of science, syntax and semantics.* 2002.
229. NEVIN, Bruce E. and Stephen JOHNSON (eds.): *The Legacy of Zellig Harris. Language and information into the 21st century. Volume 2: computability of language and computer applications.* 2002.
230. PARKINSON, Dilworth B. and Elabbas BENMAMOUN (eds.): *Perspectives on Arabic Linguistics XIII-XIV. Papers from the Thirteenth and Fourteenth Annual Symposia on Arabice Linguistics.* 2002.
231. CRAVENS, Thomas D.: *Comparative Historical Dialectology. Italo-Romance clues to Ibero-Romance sound change.* 2002.
232. BEYSSADE, Claire, Reineke BOK-BENNEMA, Frank DRIJKONINGEN and Paola MONACHESI (eds.): *Romance Languages and Linguistic Theory 2000. Selected papers from 'Going Romance' 2000, Utrecht, 30 November - 2 December.* 2002.
233. WEIJER, Jeroen van de, Vincent J. van HEUVEN and Harry van der HULST (eds.): *The Phonological Spectrum. Part I: Segmental structure.* n.y.p.
234. WEIJER, Jeroen van de, Vincent J. van HEUVEN and Harry van der HULST (eds.): *The Phonological Spectrum. Part II: Suprasegmental structure.* n.y.p.