From: AAAI-92 Proceedings. Copyright ©1992, AAAI (www.aaai.org). All rights reserved.

An On-Line Computational Model of Human Sentence Interpretation

Daniel Jurafsky
Computer Science Division, 573 Evans Hall
University of California at Berkeley
Berkeley, California 94720
jurafsky@cs.berkeley.edu

Introduction

Models of parsing or of sentence interpretation generally fall into one of three paradigms. The *linguistic paradigm* is concerned with computational models of linguistic theories, the *computational* paradigm with the computationally best process for computing the meaning or structure of a sentence, and the *psychological paradigm* with psychological modeling of human interpretation of language. Rarely have models attempted to cross disciplinary boundaries and meet the criteria imposed by each of these paradigms.

This paper describes a model of human sentence interpretation which addresses the fundamental goals of each of these paradigms. These goals include the need to produce a high-level and semantically rich representation of the meaning of the sentence, to include a motivated and declarative theory of linguistic knowledge which captures linguistic generalizations, and to account in a principled manner for psycholinguistic results.

The model consists of a grammar and a semantic interpreter named *Sal* which include:

- a theory of grammar called *Construction-Based Interpretive Grammar* (CIG) which represents knowledge of language as a collection of uniform structures called *grammatical constructions* representing lexical, syntactic, semantic, and pragmatic information.
- an evidential *access* function, which uses different knowledge sources to interactively guide its search for the correct constructions to access.
- an information-combining operation called *integration*, which is a unification-like operation augmented with knowledge about the semantic representation language and with a mechanism for functional application.
- a *selection* algorithm which prefers more *coherent* interpretations and which prunes low-ranked interpretations.

Thanks to Robert Wilensky, Nigel Ward, Michael Braverman, Marti Hearst, Jane Edwards, Graeme Hirst, and Peter Norvig for many helpful comments on these ideas and on the dissertation on which this paper is based. This research was sponsored by the Defense Advanced Research Projects Agency (DoD), monitored by the Space and Naval Warfare Systems Command under N00039-88-C-0292, and the Office of Naval Research, under contract N00014-89-J-3205.

The Grammar

The grammar that is embedded in Sal is a linguistic theory called *Construction-Based Interpretive Grammar* (CIG), part of a family of theories called *Construction Grammars* (Fillmore *et al.* 1988; Kay 1990; Lakoff 1987). CIG defines a grammar as a declarative collection of structures called *grammatical constructions* which resemble the constructions of traditional pre-generative grammar. Each of these constructions represents information from various domains of linguistic knowledge, including phonological, syntactic, semantic, pragmatic, and frequency information. Thus the grammar constitutes a database of these constructions, which might be called a *constructicon* (on the model of the word *lexicon*). Lexical entries, idioms, and syntactic structures are all represented uniformly as constructions; thus the constructicon subsumes the lexicon, the syntactic rule base, and the idiom dictionary needed by other theories.

Like many recent theories of grammar (such as Pollard & Sag 1987; Bresnan 1982; Uszkoreit 1986) CIG represents constructions as partial information structures which can be combined to build up larger structures. CIG differs from most recent grammatical theories in a two major ways. First, CIG allows constituents of constructions to be defined *semantically* as well as syntactically, such as the first constituent of the WH-NON-SUBJECT-QUESTION construction introduced below (Jurafsky 1991 & 1992 discuss constructions like the HOW-SCALE construction which require semantic information to correctly specify their constituents). Second, CIG includes *weak constructions*, which are abstract constructions (like the lexical weak constructions VERB or NOUN) which structure the grammar in an *abstraction hierarchy* that is useful for access, for defining other constructions, and for learning.

Figure 1 shows an example of the CREATE construction, the lexical construction which accounts for the verb *create*, showing that the construction is named CREATE, has a frequency of 177, and is a subtype of the weak VERB construction. The *constitute* of the construction is a semantic structure which is an instance of the **Creation-Action** concept. Semantic structures in CIG are represented in a simple frame-like representation language. This concept has two subordinate relations, **Creator** and **Created**, both currently unfilled (that is, filled by unbound variables $a

and $b). This construction has only one *constituent*, which includes phonological (or graphemic) conditions specifying the form "create". The frequency numbers which appear in constructions are taken as number of occurrences per million in the Brown Corpus, computed from Francis & Kučera (1982) and Ellegård (1978); although these works assume different models of syntax than CIG, it was usually possible to translate the numbers appropriately.

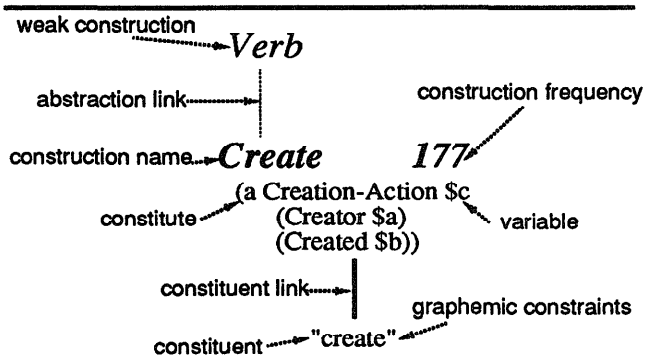

Figure 1: The "Create" Construction

A more complex construction is the WH-NON-SUBJECT-QUESTION construction (Jurafsky 1990), which accounts for sentences which begin with certain wh-clauses which do not function as the subject of the sentence, such as the sentence *"How can I create disk space?"*. The representation for the construction appears in Figure 2. The constitute of the construction instantiates the **Question** concept; its semantics is specified by integrating the semantics of the 'fronted' wh-element with the semantics of the other constituents. (Thus by not using syntactic traces, empty categories, or coindexing as place-holders for semantic integration, Sal is consistent with the evidence in Figure 4 that distant fillers are directly integrated.) Each of the construction's four constituents are specified by other predicates. The first, *"how"*, is a wh-element, as shown in Figure 2 by the **Identify** concept. The **Identify** concept characterizes all *wh-* constructions — it instantiates a frame in which the identity of some element is in question, and where some background information is provided to help identify the element. The second constituent (*"can"*) is an auxiliary verb, and participates together with the third constituent (the NOUN-PHRASE) in the SUBJECT-PREDICATE construction. The second and fourth constituents are constrained to occur in an instance of the VERB-PHRASE construction.

The Interpreter

As a cognitive model, Sal is qualitatively consistent with all of the psycholinguistic results summarized in Figure 4. These results, and the criteria discussed in the introduction above, constrain Sal's architecture toward four properties; it is *on-line*, *parallel*, *interactive*, and *uniform*. Sal is *on-line* because it maintains an interpretation for the utterance at all times, updating after each constituent of each construction.

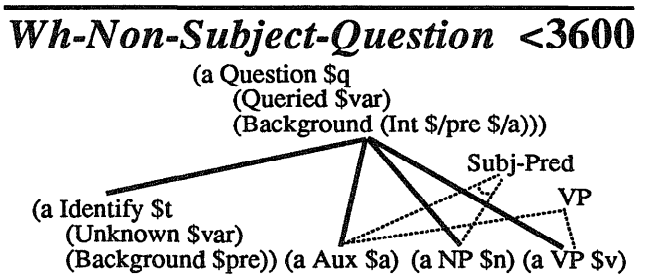

Figure 2: The Wh-Non-Subject-Question Construction

This *constituent-by-constituent* method is more fine-grained and on-line than the *rule-to-rule* method (Bach 1976) used by most other models which waits for semantic integration until the completion of an entire construction or rule. Sal is *parallel* because it can maintain more than one interpretation simultaneously, although only for a limited time. Sal is *interactive* because it allows syntactic, semantic, and higher-level expectations to help in the access of linguistic information, integrate constructions into the interpretation, and choose among candidate interpretations. Sal is *uniform* because a single interpretation mechanism accounts for the access, integration, and selection of structures at all levels of sentence processing. Sal's single mechanism thus subsumes a *lexical analyzer*, a *parser*, an *idiom processor*, and a *semantic interpreter*. Figure 3 outlines the architecture.

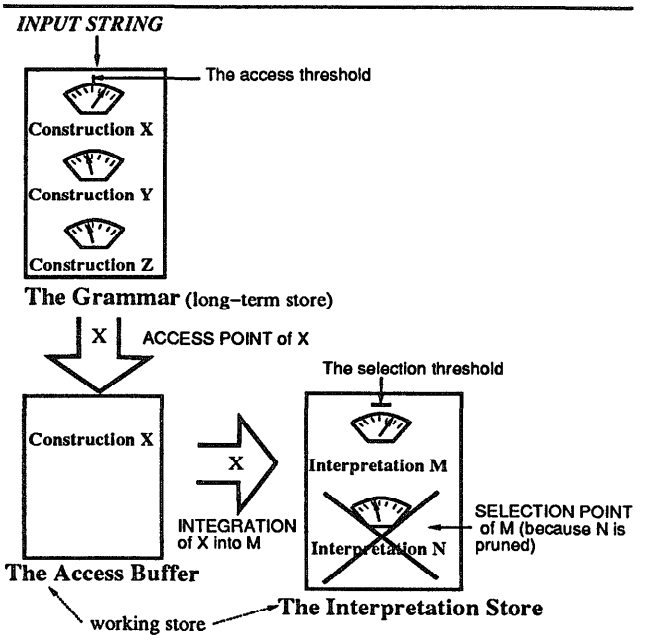

Figure 3: The Architecture of Sal

Sal's architecture consists of three components: the *working store*, the *long-term store*, and the *interpretation func-*

Lexical constructions are accessed in parallel.	Swinney (1979), Tanenhaus *et al.* (1979)
Idioms are accessed in parallel.	Cacciari & Tabossi (1988)
Syntactic constructions are accessed in parallel.	Kurtzman (1985), Gorrell (1987) and (1989)
More frequent constructions are accessed more easily.	Tyler (1984), Zwitserlood (1989), Simpson & Burgess (1985)
The *access-point* of a construction is not immediate, varying with the context and the construction.	Swinney & Cutler (1979), Cacciari & Tabossi (1988)
The interpreter builds interpretations on-line.	Marslen-Wilson *et al.* (1988), Potter & Faulconer (1979), Marslen-Wilson (1975)
The processor uses lexical valence and control knowledge in integration, including semantic and thematic knowledge.	Shapiro *et al.* (1987), Clifton *et al.* (1984), Boland *et al.* (1990), Tanenhaus *et al.* (1989)
The processor experiences difficulty when encountering "filled gaps" in non-subject position, but not in subject position.	Crain & Fodor (1985), Tanenhaus *et al.* (1985), Stowe (1986), Garnsey *et al.* (1989), Kurtzman *et al.* (1991)
The processor integrates distant fillers directly into the predicate, rather than mediated by an empty category.	Pickering & Barry (1991), Boland & Tanenhaus (1991)
The processor prefers filling arguments to adjuncts	Ford *et al.* (1982)
The processor prefer to fill expected constituents	Crain & Steedman (1985)
Lexical, syntactic, and semantic knowledge are all used in computing preferences	Taraban & McClelland (1988), Stowe (1989), Trueswell & Tanenhaus (1991), Zwitserlood (1989)

Figure 4: Sal Models Psycholinguistic Data

tion. The *working store* holds constructions in parallel as they are accessed, and partial interpretations in parallel as they are being built up. The *long-term store* holds the linguistic knowledge of the interpreter (i.e., the grammar). The *interpretation function* includes functions of *access*, *interpretation*, and *selection*. The first of Sal's sub-functions is the *access function*.

Access Function: Access a construction whenever the evidence for it passes the access threshold α.

The *access function* amasses evidence for constructions that might be used in an interpretation by assigning values to each piece of evidence, including syntactic, semantic, and frequency evidence, amassed from both bottom-up and top-down sources. When the evidence for a construction (i.e., the number of access points) has passed the *access threshold* α, the interpreter copies the construction into the **access buffer**, which is part of the *working store*; this point in time is the *access point*. The access buffer can hold more than one construction in parallel — Figure 4 summarizes evidence for parallelism, variable access-point, and sensitivity to frequency in access. This access threshold is a constant value; the access point, however, will vary across different constructions and contexts. This evidential access mechanism is a more general and knowledge-based approach to the access of linguistic knowledge than previous models, which have generally relied on a single kind of information to access rules. This might be *bottom-up* information, as in shift-reduce parsers, or *top-down* information, as in many Prolog parsers, solely *syntactic* information, as in the above as well as left-corner parsers, or solely *semantic* or *lexical* information, as in conceptual analyzers like Riesbeck & Schank (1978). The access algorithm presented here can use any of these kinds of information, as well as *frequency* information, to suggest grammatical constructions; for example top-down knowledge of a verb's semantic argument structure may suggest possible verb-phrase constructions to access, while bottom-up semantic information about a word's meaning may suggest the access of constructions which might begin with words of that semantic class.

Integration Function: An interpretation is built up for each construction, as each of its constituents are processed, by integrating the partial information provided by each constituent.

The *integration function* incrementally combines the meaning of a construction and its various constituents into an interpretation for the construction. The operation used to combine structures is also called *integration*, designed as an extension of the *unification* operation. While unification has been used very successfully in building *syntactic* structure, extending the operation to building more complex *semantic* structures requires three major augmentations (see Jurafsky (1991) for some traces and 1992 for further details):

- The integration operation is defined over the frame-like representation language which is used to describe constructions. This allows the interpreter to use the same semantic language to specify constructions as it uses to build final interpretations, without requiring translation from feature structures to semantics. The integration operation can also use information about the representation language to decide if structures should integrate; for example if the representation language specifies that construction A abstracts over construction B, the operation will integrate them successfully.
- The integration operation distinguishes *constraints* on constituents or on valence arguments from *fillers* of constituents or valence arguments
- The integration operation is augmented by a *slash* operator, which allows it to join semantic structures by embedding one inside another. This is accomplished by finding a semantic gap inside one structure (the *matrix*), and binding this gap to the other structure (the *filler*). This operation is similar to the *functional-application* operation and the *lambda-calculus* used by other models of semantic interpretation such as Moore (1989).

The *selection function* chooses an interpretation from the set of candidate interpretations in the interpretation store.

The function chooses the interpretation which is the most *coherent* with grammatical expectations:

Selection Choice Principle: Prefer the interpretation whose *most recently integrated element* was the most *coherent* with the interpretation and its lexical, syntactic, semantic, and probabilistic expectations.

Sal emphasizes the use of local coherence, that is, coherence with grammaticalized expectations such as valence, constituent, or frequency expectations. Interpretations are ranked according to the coherence ranking below:

The Coherence Ranking (in order of preference):

- *(3)* Integrations which fill a *very strong* expectation such as one for an exact construction, or for a construction which is extremely *frequent*.
- *(1)* Integrations which fill a *strong* expectation such as a *valence expectation* or a *constituent expectation*.
- *(1)* Integrations which fill a *weak expectation*, such as for an optional adjunct or include feature matching rather than feature imposing.
- *(1)* Integrations which fill no expectations, but which are still successfully integrated into the interpretation.

Psycholinguistic evidence for preferring more coherent interpretations is summarized in Figure 4; that grammatical expectations are useful in interpretation is not surprising. More surprising perhaps is that these grammatical expectations can override globally more plausible interpretations (Norvig 1988). In the following well-known garden-path examples the reader initially arrives at an interpretation which is semantically anomalous due to grammatical expectations:

- The landlord painted all the walls with cracks.
- The horse raced past the barn fell.
- The complex houses married and single students.
- The grappling hooks on to the enemy ship.

Although the current implementation of Sal's grammar is not large enough to parse these sentence, the selection algorithm does account for each of them. In each case, they occur because the more plausible interpretation is pruned too early, because some alternative interpretation was more coherent. In the first sentence, the verb 'paint' has a valence expectation for an instrument, in the second the main verb sense of 'raced' is better integrated than the reduced relative, in the third the nominal sense of 'houses' is 10 times as frequent as the verbal sense, and in the last the construction 'grappling hooks' has a constituent expectation for the word 'hooks'.

Sal's Selection Timing Principle is an extension of Gibson's (1991) pruning algorithm:

Selection Timing Principle: Prune interpretations whenever the difference between their ranking and the ranking of the most-favored interpretation is greater than the selection threshold σ.

Interpretations are ranked according to the number of points they are assigned by the coherence ranking above, and an interpretation is pruned whenever it becomes much worse than the most-favored interpretation in the interpretation store. The threshold, σ, at which interpretations are pruned, is currently set to 2 points.

The fact that Sal prunes less-favored interpretations on-line proves useful in solving some well-known complexity problems for parsing. Many researchers have noted that the problem of computing syntactic structure for a sentence can be quite complex. Church & Patil (1982) showed, for example, that the number of ambiguous phrase-structure trees for a sentence with multiple preposition-phrases was proportional to the Catalan numbers, while Barton *et al.* (1987) showed that the parsing problem for a grammar with agreement and lexical ambiguity is NP-complete. It might seem that the problems of computing *interpretations* would be even more complex, as the interpreter must produce a semantic structure as well as a syntactic one.[1]

However, it is the need to represent ambiguities for indefinite lengths of time in parsing that causes complexity. Sal builds interpretations on-line, and hence ambiguities are resolved quickly, either because of local syntactic or semantic constraints or by the **Selection Timing Principle**, which prunes less-favored interpretations on-line. Thus placing cognitive constraints on Sal actually simplifies the processing enough to avoid complexity problems.[2]

Related Research

While there have been many models of natural language interpretation, most have been limited their scope to a single paradigm. For example many processing models which are associated with linguistic theories, such as Ford *et al.* (1982) (LFG), Marcus (1980) (EST), or Fong (1991) (GB), are solely *parsing* models, including no semantic knowledge. Alternatively, some models such as Riesbeck & Schank (1978), DeJong (1982) and others of the Yale school, emphasize semantic knowledge but exclude syntactic knowledge. Some models, such as Lytinen (1986), which do include both syntactic and semantic knowledge, don't address psychological criteria. Models from the psycholinguistic community often address only small domains like lexical access.

There are some models of interpretation which attempt to address all three paradigms. Hirst's (1986) model included a Marcus-like parser, a lexical disambiguation system consistent with psychological results, a semantic interpreter, and a mechanism for resolving structural ambiguities. Hirst's model influenced the design of Sal; Sal differs in consist-

[1]The most common way of reducing complexity, the use of dynamic programming techniques (e.g., *well-formed substring table* of Kuno (1965), the *chart parsing* algorithm of Kay (1973), and the *Earley* algorithm of Earley (1970)), may not generalize from parsing to interpretation, since the internal structure of a given substructure may be needed by the interpreter, and may involve binding variables differently in the context of different interpretations.

[2]In fact, augmenting Sal by the simple assumption that the interpreter's *working store* is limited in the number of total structures it can maintain, as suggested by Gibson (1991), would ensure that the total amount of ambiguity the interpreter can maintain will *always* be limited by a small constant.

ing of a single unified mechanism rather than four separate modules, and emphasizing the use of semantic knowledge directly in the CIG grammar, thus accounting for long-distance dependencies and other phenomena in semantic rather than syntactic ways.

Cardie & Lehnert's (1991) interpreter resembles Sal in attempting to model psychological results, particularly in the use of semantic constraints to process wh-clauses. Their system differs from Sal in its focus, emphasizing robustness rather than linguistic criteria. For example, the interpreter doesn't represent larger grammatical constructions, only including local intraclausal linguistic information. Also, because their model only allows linguistic structures to be accessed bottom-up by lexical input, it is unable to account for psycholinguistic results on access in Figure 4.

Implementation Comments

Sal has been implemented in Common Lisp with a small CIG test grammar of about 50 constructions. This section presents a trace of the interpretation of the sentence *"How can I create disk space?"*.

In the first part of the trace the input word *"how"* supplies sufficient evidence for two constructions, MEANS-HOW and HOW-SCALE to be accessed. These constructions then supply evidence for the WH-NON-SUBJECT-QUESTION construction, and these are integrated together. At the end of this stage, the interpretation store contains two WH-NON-SUBJECT-QUESTION interpretations, one with the MEANS-HOW construction and one with the HOW-SCALE construction.

In this second part of the trace, the input *"can"* provides evidence for the three lexical constructions CAN-1, CAN-2, and CAN-3, as well as some larger constructions. After integration, the interpretation store contains a large number of possible interpretations, most of which are ruled out because they failed to integrate successfully, leaving only one. This successful interpretation includes the MEANS-HOW construction and the auxiliary sense of *"can"*.

Next the word *"i"* is input and integrated into the interpretation. Note that although there is some top-down evidence for the verb-phrase construction (because the WH-NON-SUBJECT-QUESTION construction expects a verb-phrase next), it is not accessed, because it is an abstract *weak construction*, and there is insufficient evidence for any of the more concrete *strong* verb-phrase constructions. Next the word *"create"* is input and integrated into the interpretation, along with an appropriate type of verb-phrase. Again, the top-down evidence for the abstract NOUN-PHRASE construction is insufficient to access it. Next the word *"disk"* is input, which provides evidence for the lexical construction DISK, as well as the noun-compound DISK-SPACE, and the DOUBLENOUN construction, which handles compound nouns.

Finally, the word *"space"* accesses the lexical construction SPACE. The selection algorithm must now choose between two interpretations, one with the DISK-SPACE construction, and one with the DOUBLENOUN construction in which the nouns are respectively *"disk"* and *"space"*. Be-

```
<cl> (parse '(how can i create disk space))

*** ACCESS ***
Input word: how
Bottom-up Evidence for constructions (means-how
 howscale)
Constructions (means-how howscale) are accessed
Bottom-up Access of (whnonsubjectquestion),
integrated directly into Access Buffer
Top-down Evidence for constructions (aux)
Access of constructions nil

*** INTEGRATION ***
After integration, Store contains:
((whnonsubjectquestion whnonsubjectquestion
 whnonsubjectquestion))

*** SELECTION ***
After removing failed integrations Store contains
(whnonsubjectquestion whnonsubjectquestion)

*** ACCESS ***
Input word: can
Bottom-up Evidence for constructions (can-1
 can-2 can-3 doublenoun bare-mono-trans-vp)
Top-down Evidence for constructions nil
Access of constructions nil

*** INTEGRATION ***
After integration, Store contains:
((whnonsubjectquestion whnonsubjectquestion
whnonsubjectquestion whnonsubjectquestion
whnonsubjectquestion whnonsubjectquestion
whnonsubjectquestion whnonsubjectquestion
whnonsubjectquestion whnonsubjectquestion))

*** SELECTION ***
After removing failed integrations Store contains
(whnonsubjectquestion)

*** ACCESS ***
Input word: i
Bottom-up Evidence for constructions (i)
Top-down Evidence for constructions (verb-phrase)
Access of constructions nil

*** INTEGRATION ***
After integration, Store contains:
((whnonsubjectquestion whnonsubjectquestion))

*** SELECTION ***
After removing failed integrations Store contains
(whnonsubjectquestion)

*** ACCESS ***
Input word: create
Bottom-up Evidence for constructions (create
 bare-mono-trans-vp)
Top-down Evidence for constructions (noun-phrase)
Access of constructions nil

*** INTEGRATION ***
After integration, Store contains:
((whnonsubjectquestion whnonsubjectquestion
 whnonsubjectquestion))

*** SELECTION ***
After removing failed integrations Store contains
 (whnonsubjectquestion)
```

cause the DISK-SPACE construction has a strong expectation for the word *"space"*, this first interpretation is selected.

```
*** ACCESS ***
Input word: disk
Bottom-up Evidence for constructions (disk
 diskspace doublenoun)
Top-down Evidence for constructions (noun)
Access of constructions nil

*** INTEGRATION ***
After integration, Store contains:
((whnonsubjectquestion whnonsubjectquestion
 whnonsubjectquestion whnonsubjectquestion))

*** SELECTION ***
After removing failed integrations Store contains
(whnonsubjectquestion whnonsubjectquestion)

*** ACCESS ***
Input word: space
Bottom-up Evidence for constructions (space
 doublenoun)
Top-down Evidence for constructions nil
Access of constructions nil

*** INTEGRATION ***
After integration, Store contains:
((whnonsubjectquestion whnonsubjectquestion
 whnonsubjectquestion whnonsubjectquestion
 whnonsubjectquestion whnonsubjectquestion))

*** SELECTION ***
After removing failed integrations Store contains
(whnonsubjectquestion whnonsubjectquestion)
Pruning construction 'whnonsubjectquestion'
(1 points), because difference from construction
'whnonsubjectquestion' (3 points) exceeds
selection threshold

Input Exhausted. Result is:
(a question $q
   (queried $p*)
   (background
    (a means-for $newvar285
       (means $p*)
       (goal
        (a ability-state $as
           (actor (a speechsituation-speaker))
           (action
            (a forcedynamicaction $newvar291
               (action
               (a creation-action
                  (created (a disk-freespace))
                  (creator (a speechsitn-speaker]
```

Summary

Sal and CIG arose from an attempt to build a model which jointly incorporated the insights of artificial intelligence and natural language processing systems, of psycholinguistic models of processing, and of linguistic models of grammatical representation. The interdisciplinary nature of the work results in a number of advantages. First, combining syntactic, semantic, lexical and idiomatic knowledge in a single linguistic knowledge structure, the grammatical construction, obviates the need for a separate lexical analyzer, parser, and semantic interpreter, and allows access, integration, and selection to be sensitive to a rich framework of linguistic evidence. Second, by filling the roles of a parser, a semantic interpreter, a linguistic model, and a cognitive model, Sal can use linguistic functionality such as capturing generalizations to aid construction access, and cognitive requirements such as modeling human memory limitations to reduce processing complexity.

References

BACH, EMMON. 1976. An extension of classical transformational grammar. In *Problems of Linguistic Metatheory (Proceedings of the 1976 Conference)*. Michigan State University.

BARTON, JR, G. EDWARD, ROBERT C. BERWICK, & ERIC SVEN RISTAD. 1987. *Computational complexity and natural language*. Cambridge, MA: MIT Press.

BOLAND, JULIE E. & MICHAEL K. TANENHAUS. 1991. The role of lexical representations in sentence processing. In *Understanding word and sentence*, ed. by G. B. Simpson, 331–366. Elsevier.

BOLAND, JULIE E., MICHAEL K. TANENHAUS, & SUSAN M. GARNSEY. 1990. Evidence for the immediate use of verb control information in sentence processing. *Journal of Memory and Language*, 29:413–432.

BRESNAN, JOAN (ed.). 1982. *The mental representation of grammatical relations*. Cambridge, MA: MIT Press.

CACCIARI, CRISTINA & PATRIZIA TABOSSI. 1988. The comprehension of idioms. *Journal of Memory and Language*, 27:668–683.

CARDIE, CLAIRE & WENDY LEHNERT. 1991. A cognitively plausible approach to understanding complex syntax. In *Proceedings of the Ninth AAAI*, 117–124. Morgan Kaufmann.

CHURCH, KENNETH & RAMESH PATIL. 1982. Coping with syntactic ambiguity. *American Journal of Computational Linguistics*, 8(3-4):139–149.

CLIFTON, JR, CHARLES, LYN FRAZIER, & CYNTHIA CONNINE. 1984. Lexical expectations in sentence comprehension. *Journal of Verbal Learning and Verbal Behavior*, 23:696–708.

CRAIN, STEPHEN & JANET DEAN FODOR. 1985. How can grammars help parsers? In *Natural language parsing*, ed. by David R. Dowty, Lauri Kartunnen, & Arnold Zwicky, 94–128. Cambridge: Cambridge University Press.

CRAIN, STEPHEN & MARK STEEDMAN. 1985. On not being led up the garden path: the use of context by the psychological syntax processor. In *Natural language parsing*, ed. by David R. Dowty, Lauri Kartunnen, & Arnold Zwicky, 320–358. Cambridge: Cambridge University Press.

DEJONG, GERALD. 1982. An overview of the FRUMP system. In *Strategies for natural language processing*, ed. by Wendy G. Lehnert & Martin H. Ringle, 149–176. NJ: Lawrence Erlbaum.

EARLEY, JAY. 1970. An efficient context-free parsing algorithm. *Communications of the ACM*, 6(8):451–455.

ELLEGÅRD, ALVAR. 1978. *The syntactic structure of English texts: A computer-based study of four kinds of text in the Brown University Corpus*. Number 43 in Gothenburg Studies in English. Göteborg: Acta Universitatis Gothoburgensis.

FILLMORE, CHARLES J., PAUL KAY, & M. C. O'CONNOR. 1988. Regularity and idiomaticity in grammatical constructions: The case of let alone. *Language*, 64(3):501–538.

FONG, SANDIWAY. 1991. The computational implementation of principle-based parsers. In *Principle-based parsing*, ed. by

Robert C. Berwick, Steven P. Abney, & Carol Tenny, 65–82. Dordrecht: Kluwer.

FORD, MARILYN, JOAN BRESNAN, & RONALD M. KAPLAN. 1982. A competence-based theory of syntactic closure. In *The mental representation of grammatical relations*, ed. by Joan Bresnan. Cambridge, MA: MIT Press.

FRANCIS, W. NELSON & HENRY KUČERA. 1982. *Frequency analysis of English usage*. Boston: Houghton Mifflin.

GARNSEY, SUSAN M., MICHAEL K. TANENHAUS, & ROBERT M. CHAPMAN. 1989. Evoked potentials and the study of sentence comprehension. *Journal of Psycholinguistic Research*, 18(1):51–60.

GIBSON, EDWARD, 1991. *A computational theory of human linguistic processing: Memory limitations and processing breakdown*. Pittsburgh, PA: Carnegie Mellon University dissertation.

GORRELL, PAUL G., 1987. *Studies of human syntactic processing: Ranked-parallel versus serial models*. Connecticut: University of Connecticut dissertation.

GORRELL, PAUL G. 1989. Establishing the loci of serial and parallel effects in syntactic processing. *Journal of Psycholinguistic Research*, 18(1):61–73.

HIRST, GRAEME. 1986. *Semantic interpretation and the resolution of ambiguity*. Cambridge: Cambridge University Press.

JURAFSKY, DANIEL. 1990. Representing and integrating linguistic knowledge. In *Proceedings of 13th Coling*, 199–204, Helsinki.

JURAFSKY, DANIEL. 1991. An on-line model of human sentence interpretation. In *Proceedings of the 13th Annual Conference of the Cognitive Science Society*, 449–454, Chicago.

JURAFSKY, DANIEL, 1992. *An on-line computational model of human sentence interpretation: A theory of the representation and use of linguistic knowledge*. Berkeley, CA: University of California dissertation. Computer Science Division Report #92/676.

KAY, MARTIN. 1973. The MIND system. In *Natural language processing*, ed. by Randall Rustin, 155–188. New York: Algorithmics Press.

KAY, PAUL. 1990. Even. *Linguistics and Philosophy*, 13:59–216.

KUNO, SUSUMU. 1965. The predictive analyzer and a path elimination technique. *Communications of the ACM*, 8(7):453–462.

KURTZMAN, HOWARD S., 1985. *Studies in syntactic ambiguity resolution*. Cambridge, MA: MIT dissertation.

KURTZMAN, HOWARD S., LOREN F. CRAWFORD, & CAYLEE NYCHIS-FLORENCE. 1991. Locating wh- traces. In *Principle-based parsing*, ed. by Robert C. Berwick, Steven P. Abney, & Carol Tenny, 347–382. Dordrecht: Kluwer.

LAKOFF, GEORGE. 1987. *Women, fire, and dangerous things*. Chicago: University of Chicago Press.

LYTINEN, STEVEN L. 1986. Dynamically combining syntax and semantics in natural language processing. In *Proceedings of the Fifth AAAI*, volume 1, 574–578. Morgan Kaufmann.

MARCUS, MITCHELL P. 1980. *A theory of syntactic recognition for natural language*. Cambridge, MA: MIT Press.

MARSLEN-WILSON, WILLIAM. 1975. Sentence perception as an interactive parallel process. *Science*, 189:226–228.

MARSLEN-WILSON, WILLIAM, COLIN M. BROWN, & LORRAINE KOMISARJEVSKY TYLER. 1988. Lexical representations in spoken language comprehension. *Language and Cognitive Processes*, 3(1):1–16.

MOORE, ROBERT C. 1989. Unification-based semantic interpretation. In *Proceedings of the 27th Annual Conference of the Association for Computational Linguistics*, 33–41, Vancouver, Canada.

NORVIG, PETER. 1988. Interpretation under ambiguity. In *Proceedings of the 14th Annual Meeting of the Berkeley Linguistics Society*, 188–201, Berkeley, CA.

PICKERING, MARTIN & GUY BARRY. 1991. Sentence processing without empty categories. *Language and Cognitive Processes*, 6(3):229–259.

POLLARD, CARL & IVAN A. SAG. 1987. *Information-based syntax and semantics: Volume 1: Fundamentals*. Chicago: University of Chicago Press.

POTTER, MARY C. & BARBARA A. FAULCONER. 1979. Understanding noun phrases. *Journal of Verbal Learning and Verbal Behavior*, 18:509–521.

RIESBECK, CHRISTOPHER K. & ROGER C. SCHANK. 1978. Comprehension by computer: Expectation-based analysis of sentences in context. In *Studies in the perception of language*, ed. by W. J. M. Levelt & G. B. Flores d'Arcais, 247–293. London: Wiley.

SHAPIRO, L. P., E. ZURIF, & J. GRIMSHAW. 1987. Sentence processing and the mental representation of verbs. *Cognition*, 27:219–246.

SIMPSON, GREG B. & CURT BURGESS. 1985. Activation and selection processes in the recognition of ambiguous words. *Journal of Experimental Psychology: Human Perception and Performance*, 11(1):28–39.

STOWE, LAURIE A. 1986. Evidence for on-line gap location. *Language and Cognitive Processes*, 1(3):227–245.

STOWE, LAURIE A. 1989. Thematic structures and sentence comprehension. In *Linguistic structure in language processing*, ed. by G. N. Carlson & M. K. Tanenhaus, 319–357. Dordrecht: Kluwer.

SWINNEY, DAVID A. 1979. Lexical access during sentence comprehension: (re)consideration of context effects. *Journal of Verbal Learning and Verbal Behavior*, 18:645–659.

SWINNEY, DAVID A. & ANNE CUTLER. 1979. The access and processing of idiomatic expressions. *Journal of Verbal Learning and Verbal Behavior*, 18:523–534.

TANENHAUS, MICHAEL K., JULIE E. BOLAND, SUSAN M. GARNSEY, & GREG CARLSON. 1989. Lexical structures in parsing long-distance dependencies. *Journal of Psycholinguistic Research*, 18(1):37–50.

TANENHAUS, MICHAEL K., JAMES M. LEIMAN, & MARK S. SEIDENBERG. 1979. Evidence for multiple stages in the processing of ambiguous words in syntactic contexts. *Journal of Verbal Learning and Verbal Behavior*, 18:427–440.

TANENHAUS, MICHAEL K., LAURIE A. STOWE, & GREG CARLSON. 1985. The interaction of lexical expectation and pragmatics in parsing filler-gap constructions. In *Program of the 7th Annual Conference of the Cognitive Science Society*, 361–365, Irvine.

TARABAN, ROMAN & JAMES L. MCCLELLAND. 1988. Constituent attachment and thematic role assignment in sentence processing: Influences of content-based expectations. *Journal of Memory and Language*, 27:597–632.

TRUESWELL, JOHN C. & MICHAEL K. TANENHAUS. 1991. Tense, temporal context and syntactic ambiguity resolution. *Language and Cognitive Processes*, 6(4):303–338.

TYLER, LORRAINE K. 1984. The structure of the initial cohort: Evidence from gating. *Perception & Psychophysics*, 36(5):417–427.

USZKOREIT, HANS. 1986. Categorial unification grammars. In *Proceedings of 11th Coling*, Bonn. University of Bonn.

ZWITSERLOOD, PIENIE. 1989. The locus of the effects of sentential-semantic context in spoken-word processing. *Cognition*, 32:25–64.