

## Chapter 3

---

# Language Analysis and Understanding

(Following section is taken from Chapter 3 “Language Analysis and Understanding”) of the book: “Survey of the state of the art in human language technology”)

### 3.6 Sentence Modeling and Parsing

**Fernando Pereira**

AT&T Bell Labs, Murray Hill, New Jersey, USA

The complex hidden structure of natural-language sentences is manifested in two different ways: *predictively*, in that not every constituent (for example, word) is equally likely in every context, and *evidentially*, in that the information carried by a sentence depends on the relationships among the constituents of the sentence. Depending on the application, one or the other of those two facets may play a dominant role. For instance, in *language modeling* for large-vocabulary connected speech recognition, it is crucial to distinguish the relative likelihoods of possible continuations of a sentence prefix, since the acoustic component of the recognizer may be unable to distinguish reliably between those possibilities just from acoustic evidence. On the other hand, in applications such as machine translation or text summarization, relationships between sentence constituents, such as that a certain noun phrase is the direct object of a certain verb occurrence, are crucial evidence in determining the correct translation or summary. *Parsing* is the process of discovering *analyses* of sentences, that is, consistent sets of relationships between constituents that are judged to hold in a given sentence, and, concurrently, what the constituents are, since constituents are typically defined inductively in terms of the relationships that hold between their parts.

It would not be possible to model or parse sentences without mechanisms to compute the properties of larger constituents from the properties of their parts, appropriately defined, since the properties of new sentences, which are unlikely to have been seen before, can only be inferred from knowledge of how their parts participate in the sentences we have observed previously. While this point may seem obvious, it has deep consequences both in language modeling and parsing. Any language model or parser must include a generative mechanism or *grammar* that specifies how sentences are built from their parts, and how the information associated to the sentence derives

from the information associated to its parts. Furthermore, to be able to cope with previously unseen sentences, any such system must involve *generalization* with respect to the data from which the language model or parser was developed.

### 3.6.1 Grammars and Derivations

It is useful to think of the grammar in a language model or parser as the specification of a configuration space in which the configurations represent stages of constituent combination, and transitions between configurations describe how constituents are combined in deriving larger constituents. For instance, the configurations may be the states of a finite-state machine and the transitions represent how words may be appended to the end of a sentence prefix. In the more complex case of phrase-structure grammars, configurations represent sequences of phrases (*sentential forms*), and transitions the possible combinations of adjacent phrases into larger phrases. A *derivation* of a sentence according to the grammar is a path in the configuration space of the grammar, from an initial configuration to a final configuration, in which all the elementary constituents are consumed. (We will call such elementary constituents *words* in what follows, although the informal notion of word may not correspond to the appropriate technical definition, especially in languages with complex morphology.)

Even with respect to procedural parsers and language models which are not normally described as containing a grammar, such as certain deterministic (Marcus, 1980; Hindle, 1993) and probabilistic parsers (Black, Jelinek, et al., 1993; Magerman & Marcus, 1991), it is useful to identify the implicit grammar defined by the possible derivation moves which the parser can use under the control of its control automaton. For instance, in a parser based on a pushdown automaton such as a shift-reduce parser (Shieber, 1983; Pereira, 1985), the grammar corresponds to the possible transitions between stack and input configurations, while the automaton's finite-state control determines which transitions are actually used in a derivation.

### 3.6.2 Precision versus Coverage

The choice of a grammar for a particular parsing or language modeling application involves two conflicting requirements: *precision* and *coverage*. By precision, we mean how well the grammar encodes constraints on possible sentences and possible meaningful relationships carried by those sentences. By coverage, we mean what proportion of actual sentences have a reasonable derivation in the grammar. We are interested in precision because a more precise grammar is better able to rule out bad sentence hypotheses in predictive tasks and bad meaningful relationship hypotheses in evidential tasks. We are interested in coverage so that our systems will handle appropriately a wide range of actual spoken or written language. However, as we increase precision by encoding more constraints in the grammar, we tend to lose coverage of those actual sentences that violate some of the constraints while still being acceptable to language users. The reason for the problem is that the most powerful constraints are idealizations of the actual performance of language users. The tension between precision and coverage is central to the design trade-offs we will now survey.

### 3.6.3 Search Space and Search Procedure

We therefore see a sentence parser or language model as consisting of a grammar and a *search procedure* which, given an input sentence, will apply transitions specified by the grammar to construct derivations of the sentence and associated analyses. In cases where the input sentence is uncertain, such as speech recognition, we may further generalize the above picture to a simultaneous search of the configuration space for the grammar and of a space of *sentence hypotheses*, represented for instance as a *word lattice* (Murveit, Butzberger, et al., 1993).

The computational properties of parsers and language models depend on two main factors: the *structure* of the search space induced by the grammar, and the *exhaustiveness* of the search procedure.

### Search Space Structure

Under the above definition of grammar, transitions from a configuration may have to take into account the whole configuration. However, most useful grammar classes have some degree of *locality* in that transitions involve only a bounded portion of a configuration. In that case, derivations can be factored into sub-derivations concerning independent parts of configurations, allowing independent sub-derivations to be shared among derivations, for potentially exponential reductions in the size of the search space. The search algorithm can then tabulate each sub-derivation and reuse it in building any derivation that shares that sub-derivation.<sup>1</sup> Such *tabular* algorithms are widely used in parsing and language modeling with appropriate kinds of grammars (Younger, 1967; Kay, 1986; Earley, 1970; Lang, 1974; Graham, Harrison, et al., 1980; Tomita, 1987), because they support exhaustive search algorithms with polynomial space and time with respect to sentence length. Furthermore, tabular algorithms can be readily extended to *dynamic programming* algorithms to search for optimal derivations with respect to appropriate evaluation functions on derivations, as we will see below.

Finite-state grammars have a straightforward tabular algorithm in which table entries consist of a state and an input position (such a table is called a *trellis* in the speech recognition literature). Context-free grammars are the standard example of a phrase structure grammar class whose derivations can be tabulated. In a bottom-up (from words to sentences) derivation for a context-free grammar, the portion of the derivation that corresponds to the recognition of a constituent labeled by a given nonterminal can be simply represented by a table entry, giving the nonterminal as a possible label of a substring of the input (Younger, 1967). Although this information leaves out the sequence of steps of the actual derivation, all derivations of that phrase can be easily reconstructed from the set of all table entries derivable for a given input string (Pointers can also be used to keep track of what table entries are used in deriving other entries.) (Younger, 1967; Earley, 1968).

Other grammar classes, such as the mildly-context-sensitive grammars (Joshi, Vijay-Shanker, et al., 1991) and some constraint-based grammars, are also suitable for tabular procedures (Shieber, 1992).

### Search Exhaustiveness

Even if the grammar allows tabular search, it may not be computationally feasible to explore the entire search space because of the effect of grammar size on search space size. For example, the simple finite-state language models used in large-vocabulary speech recognition may have millions of states and transitions. Since each state is potentially considered at each input position, the computation per word recognized is too large for real-time performance. Many techniques have been explored in speech recognition to deal with this problem (Bahl, Jelinek, et al., 1983; Kenny, Hollan, et al., 1993; Paul, 1992; Murveit, Butzberger, et al., 1993; Nguyen, Schwartz, et al., 1993).

In general, the techniques to avoid exploring the entire grammar search space fall into two main classes, *pruning* and *admissible search*. In pruning, an *evaluation function* applied to configurations determines whether they will be expanded further. Since the evaluation function cannot predict the future (to do so accurately it would have to explore the entire search space), pruning may in fact block the correct derivation. The choice of evaluation function is thus a trade-off between reduced search space (and thus reduced runtime and memory requirements) and the risk of missing the correct analysis (or even every analysis). Currently there is no theory of pruning trade-offs relating

---

<sup>1</sup>The required properties are analogous to the cut-elimination property that underlies the connection between sequent and natural-deduction presentations of logics (Prawitz, 1965).

bounds on risk of error to the form of the evaluation function, so the design of evaluation functions is an empirical art.

Although pruning away the correct derivation is a problem in practical applications, in psycholinguistic modeling it may in fact correspond to failures of human sentence processing, for instance garden paths. *Deterministic parsers* (Marcus, 1980; Hindle, 1993) take pruning to an extreme in using elaborate evaluation functions to select exactly one course of derivation. Dead ends are then supposed to model the situations in which human subjects are forced to recover from parsing failures. Other models, particularly those based on neuronal notions of activation and lateral inhibition, may allow a local *race* between alternative expansions of a configuration but inhibit all but one of the alternatives within a bounded number of steps (Stevenson, 1993; McRoy & Hirst, 1990; Pritchett, 1988).

Admissible search procedures do not block potential derivations. Instead, they order sub-derivations, so that the ones that are more likely to be expanded to the best complete derivations will be considered before less promising ones. The difficulty is of course to define ordering criteria with high probability of reaching the best derivations before exploring a large set of useless configurations. Of particular interest here are A\*-type algorithms (Nilsson, 1980) which expand the configuration with the lowest *cost estimate*, where the estimate is required to be a lower bound of the true cost (under a cost model appropriate for the task, see below) and identical to the true cost for complete derivations. The first complete derivation reached by an A\* algorithm is then guaranteed to have the lowest cost. However, since it is difficult to choose cost estimates that narrow the search sufficiently, more aggressive estimates that may overshoot the true cost are often used, with the result that the first complete derivation may not be the best one.

Clearly, the selection of evaluation functions for pruning or admissible search is closely tied to the precision-coverage trade-off.

### 3.6.4 Grammar Classes

A wide range of grammar classes have been investigated in parsing and language modeling, depending on the nature of the application and on particular insights on language structure and sentence distribution. Grammar classes have been characterized along many different theoretical dimensions. What is known in those areas about certain important grammar classes is described elsewhere in this document 3.3.

Here, we consider a more informal and empirical dimension of variation that has great impact in the development of parsers and language models: how much of the required predictive and evidential power belongs to the grammar itself and how much resides in the search procedure controlling the use of the grammar. Choices along this dimension often involve philosophical disagreements on whether language is fundamentally governed by an innate system of rules (the rationalist position most closely identified with Chomsky) or rather a system of statistical regularities, associations and constructions derived by learning (the empiricist position informing much work in statistical language modeling). However, they also relate to different choices with respect to the coverage/precision trade-off.

At one end of the spectrum, which is often associated with empiricist work, extremely unconstraining grammars are controlled by search evaluation functions automatically learned from language data. An extreme example is finite-state *n-gram* grammars, in which states encode information on the last  $n - 1$  observed words, have been used with practical success in speech recognition (Jelinek, Mercer, et al., 1992). In these grammars every sequence of words is considered a possible sentence, but probabilities are assigned to state transitions to model the relative likelihoods of different strings. As we will see, the association of probabilities to transitions is a useful technique in a wide range of grammatical settings.

While n-gram grammars have proven very useful for language modeling, derivation steps do not correspond

in any direct way to possible meaningful relations in the sentence, for instance, those between a main verb and its arguments. Parsing requires more complex grammars, in which derivation steps are associated to possible relations of interest. Even in language modeling, distributional regularities associated with meaningful relationships may be an important source of additional predictive power (Hindle, 1990; Hindle & Rooth, 1991; Dagan, Markus, et al., 1993; Lafferty, Sleator, et al., 1992).

Grammatical representations of meaningful relationships may be usefully classified into three main classes: *linguistic grammars*, *task-oriented grammars* and *data-oriented grammars*. Linguistic grammars and task-oriented grammars have been in use since the beginning of computational linguistics. Data-oriented grammars, in their finite-state form, as discussed above, go back to the beginning of statistical studies of language by Markov, but investigations into data-oriented grammars capable of representing meaningful relationships have only recently begun.

### Linguistic Grammars

Most formal linguistic theories have been used at some time or other as the basis for computational grammars. The main issues in applying linguistic theory to the development of computational grammars are: *coverage*, *predictive power* and *computational requirements*.

**Coverage:** Linguistic theories are typically developed to explain puzzling aspects of linguistic competence, such as the relationships between active and passive sentences, the constraints on use of anaphoric elements, or the possible scopes of quantifying elements such as determiners and adverbs. However, actual language involves a wide range of other phenomena and constructions, such as idioms, coordination, ellipsis, apposition and extraposition, which may not be germane to the issues addressed by a particular linguistic theory or which may offer unresolved challenges to the theory. Therefore, a practical grammar will have to go far beyond the proposals of any given theory to cover a substantial proportion of observed language. Even then, coverage gaps are relatively frequent and difficult to fill, as they involve laborious design of new grammar rules and representations.

**Predictive Power:** Linguistic grammars, being oriented towards the description of linguistic competence, are not intended to model *distributional* regularities arising from pragmatics, discourse and conventional use that manifest themselves in word and construction choice. Yet those are the regularities that appear to contribute most to the estimation of relative likelihoods of sentences or analyses. The encoding of distributional predictions must therefore be left to the search procedure, in the form of an appropriate evaluation function. However, the configurations generated by a grammar may not carry the most useful information in evaluating them. For example, whether a particular prepositional phrase modifies a direct object noun phrase or the main verb depends heavily on the actual verb, noun, preposition and prepositional object (Hindle & Rooth, 1991), but a traditional phrase-structure grammar does not make that information available in the syntactic categories of noun phrase, verb phrase and prepositional phrase. Therefore, in a phrase-structure setting whole derivations rather than individual configurations would have to be evaluated. But this would preclude the factorization of derivations that leads to tractable search as noted above. These considerations explain in part the recent growing interest in *lexicalized* grammatical frameworks such as dependency grammar (Mel'čuk, 1988; Hudson, 1990; Sleator & Temperley, 1991), slot grammar (McCord, 1980; McCord, 1989), categorial grammar (Lambek, 1958; Ades & Steedman, 1982; Moortgat, 1988), Head-Driven Phrase-Structure Grammar (HPSG) (Pollard & Sag, 1987) and lexicalized tree-adjoining grammar (Schabes, 1990), all of which lead to configurations made up of lexical items and direct relationships between them.

**Computational Requirements:** The best formal explanation of a particular aspect of linguistic competence has no necessary correlation with computational efficiency. For instance, modern versions of transformational grammar based on the theory of government and binding, or its more recent developments, involve either very complex search procedures or very complex compilation procedures into formalisms with better search properties (Stabler, 1992; Fong, 1992; Johnson, 1992). Similar problems have been noted with respect to HPSG and certain varieties of categorial grammar.

While direct use of formalized linguistic theories for parsing and language modeling seems computationally problematic, much progress has been made in the development of tractable grammatical formalisms capable of encoding important aspects of linguistic theory. The class of mildly context-sensitive formalisms (Joshi, Vijay-Shanker, et al., 1991), of which tree-adjoining grammars (Joshi, Levy, et al., 1975; Joshi, 1985) and combinatory categorial grammar (Ades & Steedman, 1982) are two notable instances, has polynomial-time and space parsing algorithms, and can encode important aspects of transformational and categorial linguistic analysis. Constraint-based grammar formalisms can be intractable or even undecidable in general (Carpenter, 1992), but special cases of interest are often efficiently parsable (Alshawi, 1992). For instance, lexical-functional grammar combines a context-free skeleton with constraints describing non-constituent syntactic properties. Although the combination is intractable in general, a carefully designed constraint-application schedule can make it possible to parse with linguistically-plausible grammars in such a way that the intractability does not arise (Maxwell & Kaplan, 1989).

However, even polynomial-time algorithms may not be sufficiently fast for practical applications, given the effect of grammar size on parsing time. Search reduction techniques like those described in section 3.6.3 would then be needed to keep performance within reasonable bounds, at the risk of worse coverage.

### Task-Oriented Grammars

For most current applications in text summarization, information retrieval and speech understanding, the predictive and evidential power of a general-purpose grammar and a general control mechanism are insufficient for reasonable performance in the task. Furthermore, even when parameters of the grammar and control mechanism can be learned automatically from training corpora, the required corpora do not exist or are too small for proper training. The alternative is then to devise grammars that specify directly how relationships relevant to the task may be expressed in natural language. For instance, one may use a phrase-structure grammar in which nonterminals stand for task concepts and relationships (for example, *flight* or *leave* in an airline reservation task) and rules specify possible expressions of those concepts and relationships (Seneff, 1992; Ward, 1991b). Such *semantic grammars* have often been used for database access tasks. More generally, a knowledge-representation language (for instance, a frame language) can be used to specify the possible relationships between concepts, and relatively low-power grammatical descriptions (often finite-state) describe natural-language expressions that give strong evidence for concepts and relationships (Jacobs & Rau, 1993; Hobbs, Appelt, et al., 1993).

Task-oriented grammars provide very strong guidance to a parser, but that guidance is bought at the expense of generality and coverage, since the detailed specifications they rely on may often fail to fit naturally-occurring language. Therefore, parsing algorithms for task-oriented grammars are usually allowed to *relax* the grammar by ignoring portions of the input that do not fit the given grammar (Ward, 1991a; Jackson, Appelt, et al., 1991). This can increase coverage usefully in applications such as limited-domain speech understanding and text-summarization, where there are very strong expectations of what are the relevant inputs, but the increase of coverage is in general at the expense of precision.

### Data-Oriented Grammars

In so far as linguistic grammars and task-oriented grammars provide strong constraints for modeling and parsing, they risk low coverage because the constraints limit the transitions between configurations, and thus the availability of derivations for strings. As we have seen, this problem can be alleviated in a task-oriented setting, but as far as we know relaxation is a sensible policy only for highly-constrained tasks. An alternative way of increasing coverage is to start with less constraining grammars, and rely on an evaluation function to select the most likely derivations in the more densely connected search space that results from the less constraining grammar. However, this requires finding an appropriate evaluation function. In a data-oriented framework, a *learning* or *training* procedure tries to determine the evaluation function that produces best results on an appropriate training corpus. For example, an n-gram grammar allows any word sequence, but transitions are given probabilities derived from how often states were reached and transitions crossed running the grammar over a training corpus (Jelinek, Mercer, et al., 1992). As another example, frequencies of rule and nonterminal use can be used to estimate rule probabilities for an underconstrained context-free grammar (Baker, 1979; Lari & Young, 1990; Pereira & Schabes, 1992).

Although there have been some successes in training evaluation functions for previously-designed grammars (Fujisaki, Jelinek, et al., 1989; Black, Lafferty, et al., 1992), training with respect to a fixed grammar has the problem that either the grammar allows many transitions that are never observed in reality, forcing the evaluation function to be more complex to rule them out effectively, or it is too restrictive and does not allow transitions that actually occur. That difficulty has motivated the investigation of grammatical frameworks and learning algorithms that will concurrently learn a grammar and an appropriate evaluation function (Sampson, Haigh, et al., 1989; Bod, 1993; Hindle, 1992; Stolcke & Omohundro, 1993). One particular class of such procedures constructs a dictionary of commonly observed substrings or sub-analyses that can be combined by a small set of rules to yield the observed sentences or analyses, with the evaluation function discriminating between alternatives ways of reconstructing a sentence or analysis from the fragments in the dictionary (Bod, 1993; Hindle, 1992). A variety of predictive power and grammar size criteria (for example, Bayesian, minimum-description length) may then be used to find good trade-offs between grammar (dictionary) size, prediction of the training set, and generalization to new material.

#### 3.6.5 Evaluating Derivations

In the overall view of parsing and language modeling given above, a parser or language model searches the configuration space defined by a grammar for possible derivations of the sentence(s) under analysis. Since the grammar by itself is unlikely to encode all the semantic, pragmatic and discourse information relevant to distinguishing plausible analyses from implausible ones, the search needs to be guided by an evaluation function that assigns plausibility scores to derivations. An especially important case is that of *probabilistic grammars*, which associate with each transition the conditional probability of taking that transition from a configuration given that the configuration was reached. Such grammars are based on a *Markovian* or *conditional independence* assumption that the probability of a (partial) derivation depends just on its penultimate configuration and the transition taken from it. Then the probability of a derivation is simply the product of the probability of its initial configuration by the product of the probabilities of the transitions in the derivation.

When transitions are directly associated with observable events (for example, extension of a partial sentence by one word in a finite-state model), transition probabilities can be estimated by simply counting the number of times the transition is taken for all possible derivations of all sentences in a training corpus. In general, however, the transition probabilities are not associated to directly observable events. In that case, iterative procedures may be used to find the transition probabilities that maximize the probability that the training corpus was observed (Dempster, Laird, et al., 1977; Baum & Petrie, 1966; Baker, 1979). For language modeling, the training

corpus may just be a set of sentences, while for parsing a set of sentences tagged with constraints on possible grammatical relationships (for example, phrase boundaries) is often preferable (Black, Lafferty, et al., 1992; Pereira & Schabes, 1992).

While probabilistic evaluation functions dominate in language modeling, where they are used to estimate the likelihood that a certain word sequence was uttered, other types of evaluation function are often used in parsing, especially those based on the degree of agreement of the best scoring analyses and analyses in a training corpus (Alshawi & Carter, 1994).

Computationally, the critical property of an evaluation function is whether it is compatible with tabular algorithms for searching the derivation space, in the sense that the score of a derivation is determined by the scores of the subderivations into which the derivation is factored by tabulation. For probabilistic functions, this amounts to a strengthened Markovian condition for derivations, which, for instance, is satisfied by stochastic context-free grammars (Booth & Thompson, 1973; Baker, 1979), certain kinds of parsers for constraint-based grammars (Briscoe & Carroll, 1993) and stochastic tree-adjointing grammars (Schabes, 1992). In such cases, the tabular search algorithms can be converted into dynamic programming algorithms (Teitelbaum, 1973; Baker, 1979; Lang, 1989; Jelinek, Lafferty, et al., 1990; Lafferty, Sleator, et al., 1992; Schabes, 1992) to search efficiently for best-scoring derivations.

### 3.6.6 Future Directions

The issue that dominates current work in parsing and language modeling is to design parsers and evaluation functions with high coverage and precision with respect to naturally occurring linguistic material (for example, news stories, spontaneous speech interactions). Simple high-coverage methods such as n-gram models miss the higher-order regularities required for better prediction and for reliable identification of meaningful relationships, while complex hand-built grammars often lack coverage of the *tail* of individually rare but collectively frequent sentence structures (cf. Zipf's law). Automated methods for grammar and evaluation function acquisition appear to be the only practical way to create accurate parsers with much better coverage. The challenge is to discover how to use linguistic knowledge to constrain that acquisition process.

## 3.7 Chapter References

- ACL (1983). *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, Cambridge, Massachusetts. Association for Computational Linguistics.
- ACL (1990). *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, Pittsburgh, Pennsylvania. Association for Computational Linguistics.
- ACL (1992). *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, University of Delaware. Association for Computational Linguistics.
- ACL (1993). *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, Ohio State University. Association for Computational Linguistics.
- Ades, A. E. and Steedman, M. J. (1982). On the order of words. *Linguistics and Philosophy*, 4(4):517–558.
- Allen, J., Hunnicutt, M. S., and Klatt, D. (1987). *From text to speech—the MITalk system*. MIT Press, Cambridge, Massachusetts.

- Allen, J. F., Schubert, L. K., Ferguson, G., Heeman, P., Hwang, C. H., Kato, T., Light, M., Martin, N., Miller, B., Poesio, M., and Traum, D. R. (1995). The TRAINS project: a case study in building a conversational planning agent. *Journal of Experimental and Theoretical AI*.
- Alshawi, H. (1990). Resolving quasi logical form. *Computational Linguistics*, 16:133–144.
- Alshawi, H., editor (1992). *The Core Language Engine*. MIT Press, Cambridge, Massachusetts.
- Alshawi, H., Arnold, D. J., Backofen, R., Carter, D. M., Lindop, J., Netter, K., Pulman, S. G., and Tsujii, J.-I. (1991). Rule formalism and virtual machine design study. Technical Report ET6/1, CEC.
- Alshawi, H. and Carter, D. (1994). Training and scaling preference functions for disambiguation. *Computational Linguistics*, 20:635–648.
- ANLP (1994). *Proceedings of the Fourth Conference on Applied Natural Language Processing*, Stuttgart, Germany. ACL, Morgan Kaufmann.
- Antworth, E. L. (1990). PC-KIMMO: a two-level processor for morphological analysis. Technical Report Occasional Publications in Academic Computing No. 16, Summer Institute of Linguistics, Dallas, Texas.
- Appel, A. W. and Jacobson, G. J. (1988). The world's fastest scrabble program. *Communications of the ACM*, 31(5):572–578.
- ARPA (1993). *Proceedings of the 1993 ARPA Human Language Technology Workshop*, Princeton, New Jersey. Advanced Research Projects Agency, Morgan Kaufmann.
- Atkins, B. T. S. and Levin, B. (1992). Admitting impediments. In *Lexical Acquisition: Using On-Line Resources to Build a Lexicon*. Lawrence Erlbaum, Hillsdale, New Jersey.
- Bahl, L. R., Jelinek, F., and Mercer, R. L. (1983). A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):179–190.
- Baker, J. K. (1979). Trainable grammars for speech recognition. In Wolf, J. J. and Klatt, D. H., editors, *Speech communication papers presented at the 97th Meeting of the Acoustical Society of America*, pages 547–550. Acoustical Society of America, MIT Press.
- Barwise, J. and Cooper, R. (1981). Generalized quantifiers and natural language. *Linguistics and Philosophy*, 4:159–219.
- Barwise, J. and Etchemendy, J. (1987). *The Liar*. Chicago University Press, Chicago.
- Barwise, J. and Perry, J. (1983). *Situations and Attitudes*. MIT Press, Cambridge, Massachusetts.
- Bates, M., Bobrow, R., Ingria, R., and Stallard, D. (1994). The delphi natural language understanding system. In *Proceedings of the Fourth Conference on Applied Natural Language Processing*, pages 132–137, Stuttgart, Germany. ACL, Morgan Kaufmann.
- Baum, L. E. and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, 37:1554–1563.
- Berwick, R. C., Abney, S. P., and Tenny, C., editors (1992). *Principle-Based Parsing: Computation and Psycholinguistics*. Kluwer, Dordrecht, The Netherlands.
- Black, E., Garside, R., and Leech, G., editors (1993). *Statistically-Driven Computer Grammars of English: The IBM/Lancaster Approach*. Rodopi, Amsterdam, Atlanta.

- Black, E., Jelinek, F., Lafferty, J., Magerman, D. M., Mercer, D., and Roukos, S. (1993). Towards history-based grammars: Using richer models for probabilistic parsing. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 31–37, Ohio State University. Association for Computational Linguistics.
- Black, E., Lafferty, J., and Roukos, S. (1992). Development and evaluation of a broad-coverage probabilistic grammar of English-language computer manuals. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, pages 185–192, University of Delaware. Association for Computational Linguistics.
- Bod, R. (1993). Using an annotated corpus as a stochastic parser. In *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics*, pages 37–44, Utrecht University, The Netherlands. European Chapter of the Association for Computational Linguistics.
- Booth, T. L. and Thompson, R. A. (1973). Applying probability measures to abstract languages. *IEEE Transactions on Computers*, C-22(5):442–450.
- Bouma, G., Koenig, E., and Uszkoreit, H. (1988). A flexible graph-unification formalism and its application to natural-language processing. *IBM Journal of Research and Development*.
- Bresnan, J., editor (1982). *The Mental Representation of Grammatical Relations*. MIT Press, Cambridge, Massachusetts.
- Briscoe, E. J. (1992). Lexical issues in natural language processing. In Klein, E. and Veltman, F., editors, *Natural Language and Speech*, pages 39–68. Springer-Verlag.
- Briscoe, E. J. (1994). Prospects for practical parsing: robust statistical techniques. In de Haan, P. and Oostdijk, N., editors, *Corpus-based Research into Language: A Festschrift for Jan Aarts*, pages 67–95. Rodopi, Amsterdam.
- Briscoe, E. J. and Carroll, J. (1993). Generalized probabilistic LR parsing of natural language (corpora) with unification-based grammars. *Computational Linguistics*, 19(1):25–59.
- Briscoe, E. J. and Waegner, N. (1993). Undergeneration and robust parsing. In Meijs, W., editor, *Proceedings of the ICAME Conference*, Amsterdam. Rodopi.
- Carpenter, B. (1992). ALE—the attribute logic engine user’s guide. Technical report, Carnegie Mellon University, Carnegie Mellon University, Pittsburgh, Pennsylvania.
- Carpenter, B. (1992). *The Logic of Typed Feature Structures*, volume 32 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press.
- Church, K. (1988). A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of the Second Conference on Applied Natural Language Processing*, pages 136–143, Austin, Texas. ACL.
- COLING (1994). *Proceedings of the 15th International Conference on Computational Linguistics*, Kyoto, Japan.
- Cooper, R. (1983). *Quantification and Syntactic Theory*. Reidel, Dordrecht.
- Copestake, A. and Briscoe, E. J. (1992). Lexical operations in a unification based framework. In Pustejovsky, J. and Bergler, S., editors, *Lexical Semantics and Knowledge Representation*. Springer-Verlag, Berlin.
- Cutting, D., Kupiec, J., Pedersen, J., and Sibun, P. (1992). A practical part of speech tagger. In *Proceedings of the 3rd Conference on Applied Language Processing*, pages 133–140, Trento, Italy.

- Dagan, I., Markus, S., and Markovitch, S. (1993). Contextual word similarity and estimation from sparse data. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 164–171, Ohio State University. Association for Computational Linguistics.
- DARPA (1989). *Proceedings of the Second DARPA Speech and Natural Language Workshop*, Cape Cod, Massachusetts. Defense Advanced Research Projects Agency.
- DARPA (1991). *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, Pacific Grove, California. Defense Advanced Research Projects Agency, Morgan Kaufmann.
- Davidson, D. (1969). Truth and meaning. In Davis, J. W. et al., editors, *Philosophical*, pages 1–20. Hingham.
- Davidson, D. (1973). In defense of Convention T. In Leblanc, H., editor, *Truth, Syntax and Modality*, pages 76–85. North Holland.
- de Marcken, C. (1990). Parsing the LOB corpus. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 243–251, Pittsburgh, Pennsylvania. Association for Computational Linguistics.
- De Rose, S. J. (1988). Grammatical category disambiguation by statistical optimization. *Computational Linguistics*, 14(1):31–39.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38.
- EACL (1993). *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics*, Utrecht University, The Netherlands. European Chapter of the Association for Computational Linguistics.
- Earley, J. C. (1968). *An Efficient Context-Free Parsing Algorithm*. PhD thesis, Computer Science Department, Carnegie-Mellon University.
- Earley, J. C. (1970). An efficient context-free parsing algorithm. *Communications of the ACM*, 13(2):94–102.
- Elworthy, D. (1993). Part-of-speech tagging and phrasal tagging. Technical report, University of Cambridge Computer Laboratory, Cambridge, England.
- Emele, M. and Zajac, R. (1990). Typed unification grammars. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, Pittsburgh, Pennsylvania. Association for Computational Linguistics.
- Flickinger, D. (1987). *Lexical Rules in the Hierarchical Lexicon*. PhD thesis, Stanford University.
- Fong, S. (1992). The computational implementation of principle-based parsers. In Berwick, R. C., Abney, S. P., and Tenny, C., editors, *Principle-Based Parsing: Computation and Psycholinguistics*, pages 65–82. Kluwer, Dordrecht, The Netherlands.
- Frege, G. (1892). Über sinn und bedeutung (translated as ‘on sense and reference’). In Geach and Black, editors, *Translations from the Philosophical Writings of Gottlob Frege*. Blackwell, Oxford. translation 1960.
- Fujisaki, T., Jelinek, F., Cocke, J., Black, E., and Nishino, T. (1989). A probabilistic parsing method for sentence disambiguation. In *Proceedings of the International Workshop on Parsing Technologies*, Pittsburgh.
- Gärdenfors, P. (1990). The dynamics of belief systems: Foundations vs. coherence theories. *Revue Internationale de Philosophie*, 172:24–46.

- Garside, R., Leech, G., and Sampson, G. (1987). *Computational Analysis of English: A Corpus-based Approach*. Longman, London.
- Gazdar, G. (1987). Linguistic applications of default inheritance mechanisms. In Whitelock, P. H., Somers, H., Bennet, P., Johnson, R., and Wood, M. M., editors, *Linguistic Theory and Computer Applications*, pages 37–68. Academic Press, London.
- Graham, S. L., Harrison, M. A., and Ruzzo, W. L. (1980). An improved context-free recognizer. *ACM Transactions on Programming Languages and Systems*, 2(3):415–462.
- Greene, B. B. and Rubin, G. M. (1971). Automatic grammatical tagging of English. Technical report, Brown University.
- Grice, H. P. (1975). Logic and conversation. In Cole, P., editor, *Speech Acts, Syntax and Semantics, Vol III: Speech Acts*. Academic Press, New York.
- Groenendijk, J. and Stokhof, M. (1984). On the semantics of questions and the pragmatics of answers. In Landman, F. and Veltman, F., editors, *Varieties of Formal Semantics*, pages 143–170. Foris, Dordrecht.
- Groenendijk, J. and Stokhof, M. (1990). Dynamic montage grammar. In Kalman, L. and Polos, L., editors, *Papers from the Second Symposium on Logic and Language*, pages 3–48. Akademiai Kiadoo, Budapest.
- Groenendijk, J. and Stokhof, M. (1991a). Dynamic predicate logic. *Linguistics and Philosophy*, 14:39–100.
- Groenendijk, J. and Stokhof, M. (1991b). Two theories of dynamic semantics. In van Eijck, J., editor, *Logics in AI—European Workshop JELIA '90, Springer Lecture Notes in Artificial Intelligence*, pages 55–64. Springer-Verlag, Berlin.
- Haddock, J. N., Klein, E., and Morrill, G. (1987). *Unification Categorical Grammar, Unification Grammar and Parsing*. University of Edinburgh.
- Hindle, D. (1983). Deterministic parsing of syntactic nonfluencies. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pages 123–128, Cambridge, Massachusetts. Association for Computational Linguistics.
- Hindle, D. (1983). User manual for Fidditch, a deterministic parser. Technical Report Technical Memorandum 7590-142, Naval Research Laboratory.
- Hindle, D. (1989). Acquiring disambiguation rules from text. In *Proceeds of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 118–125, Vancouver, Canada.
- Hindle, D. (1990). Noun classification from predicate-argument structures. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 268–275, Pittsburgh, Pennsylvania. Association for Computational Linguistics.
- Hindle, D. (1992). An analogical parser for restricted domains. In *Proceedings of the Fifth DARPA Speech and Natural Language Workshop*, pages 150–154. Defense Advanced Research Projects Agency, Morgan Kaufmann.
- Hindle, D. (1993). A parser for text corpora. In Atkins, B. T. S. and Zampolli, A., editors, *Computational Approaches to the Lexicon*. Oxford University Press.
- Hindle, D. and Rooth, M. (1991). Structural ambiguity and lexical relations. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 229–236, Berkeley, California. Association for Computational Linguistics.

- Hobbs, J. R., Appelt, D., Bear, J., Israel, D., Kameyama, M., and Tyson, M. (1993). FASTUS: a system for extracting information from text. In *Proceedings of the 1993 ARPA Human Language Technology Workshop*, pages 133–137, Princeton, New Jersey. Advanced Research Projects Agency, Morgan Kaufmann.
- Hobbs, J. R., Stickel, M., Appelt, D., and Martin, P. (1993). Interpretation as abduction. *Artificial Intelligence*, 63(1-2):69–142.
- Hudson, R. (1990). *English Word Grammar*. Blackwell, Oxford, England.
- Jackson, E., Appelt, D., Bear, J., Moore, R., and Podlozny, A. (1991). A template matcher for robust natural-language interpretation. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, pages 190–194, Pacific Grove, California. Defense Advanced Research Projects Agency, Morgan Kaufmann.
- Jacobs, P. S. and Rau, L. F. (1993). Innovations in text interpretation. *Artificial Intelligence*, 63(1-2):143–191.
- Järvinen, T. (1994). Annotating 200 million words. In *Proceedings of the 15th International Conference on Computational Linguistics*, Kyoto, Japan.
- Jelinek, F., Lafferty, J. D., and Mercer, R. L. (1990). Basic methods of probabilistic context free grammars. Technical Report RC 16374 (72684), IBM, Yorktown Heights, NY 10598.
- Jelinek, F., Mercer, R. L., and Roukos, S. (1992). Principles of lexical language modeling for speech recognition. In Furui, S. and Sondhi, M. M., editors, *Advances in Speech Signal Processing*, pages 651–699. Marcel Dekker.
- Jensen, K. (1991). A broad-coverage natural language analysis system. In Tomita, M., editor, *Current Issues in Parsing Technology*. Kluwer Academic Press, Dordrecht.
- Jensen, K. and Heidorn, G. (1993). *Natural Language Processing: The PLNLP Approach*. Kluwer Academic, Boston, Dordrecht, London.
- Jensen, K. and Heidorn, G. E. (1983). The fitted parse: 100% parsing capability in a syntactic grammar of English. In *Proceedings of the First Conference on Applied Natural Language Processing*, pages 3–98.
- Johnson, C. D. (1972). *Formal Aspects of Phonological Description*. Mouton, The Hague.
- Johnson, M. (1992). Deductive parsing: The use of knowledge of language. In Berwick, R. C., Abney, S. P., and Tenny, C., editors, *Principle-Based Parsing: Computation and Psycholinguistics*, pages 39–64. Kluwer, Dordrecht, The Netherlands.
- Jones, B. (1994). Can punctuation help parsing? In *Proceedings of the 15th International Conference on Computational Linguistics*, Kyoto, Japan.
- Joshi, A. K. (1985). How much context-sensitivity is necessary for characterizing structural descriptions—Tree adjoining grammars. In Dowty, D., Karttunen, L., and Zwicky, A., editors, *Natural Language Processing—Theoretical, Computational and Psychological Perspectives*. Cambridge University Press, New York.
- Joshi, A. K., Levy, L. S., and Takahashi, M. (1975). Tree adjunct grammars. *Journal of Computer and System Sciences*, 10(1).
- Joshi, A. K. and Schabes, Y. (1992). Tree-adjoining grammars and lexicalized grammars. In *Tree Automata and LGS*. Elsevier Science, Amsterdam.
- Joshi, A. K., Vijay-Shanker, K., and Weir, D. J. (1991). The convergence of mildly context-sensitive grammatical formalisms. In Sells, P., Shieber, S., and Wasow, T., editors, *Foundational Issues in Natural Language Processing*. MIT Press.

- Kamp, H. (1981). A theory of truth and semantic representation. In Groenendijk, J., Janssen, T., and Stokhof, M., editors, *Formal Methods in the Study of Language*. Mathematisch Centrum, Amsterdam.
- Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*. Kluwer, Dordrecht.
- Kaplan, R. M. and Bresnan, J. (1982). Lexical-functional grammar: a formal system for grammatical representation. In Bresnan, J., editor, *The Mental Representation of Grammatical Relations*. MIT Press, Cambridge, Massachusetts.
- Kaplan, R. M. and Kay, M. (1994). Regular models of phonological rule systems. *Computational Linguistics*, 20(3):331–378. written in 1980.
- Karlgren, H., editor (1990). *Proceedings of the 13th International Conference on Computational Linguistics*, Helsinki. ACL.
- Karlsson, F., Voutilainen, A., Heikkilä, J., and Anttila, A., editors (1994). *Constraint Grammar: A Language-Independent Formalism for Parsing Unrestricted Text*. Mouton de Gruyter, Berlin, New York.
- Karttunen, L. (1989). Radical lexicalism. In Baltin, M. and Kroch, A., editors, *Alternative Conceptions of Phrase Structure*. The University of Chicago Press, Chicago.
- Karttunen, L. (1993). Finite-state lexicon compiler. Technical Report ISTL-NLTT-1993-04-02, Xerox PARC, Palo Alto, California.
- Karttunen, L. and Beesley, K. R. (1992). Two-level rule compiler. Technical Report ISTL-92-2, Xerox PARC, Palo Alto, California.
- Kay, M. (1979). Functional grammar. In *Proceedings of the Fifth Annual Meeting of the Berkeley Linguistic Society*, pages 142–158.
- Kay, M. (1984). Functional unification grammar: a formalism for machine translation. In *Proceedings of the 10th International Conference on Computational Linguistics*, Stanford University, California. ACL.
- Kay, M. (1986). Algorithm schemata and data structures in syntactic processing. In Grosz, B. J., Sparck Jones, K., and Webber, B. L., editors, *Readings in Natural Language Processing*, chapter I. 4, pages 35–70. Morgan Kaufmann Publishers, Inc., Los Altos, California. Originally published as a Xerox PARC technical report, 1980.
- Kenny, P., Hollan, R., Gupta, V. N., Lenning, M., Mermelstein, P., and O’Shaughnessy, D. (1993). A\*-admissible heuristics for rapid lexical access. *IEEE Transactions on Speech and Audio Processing*, 1(1):49–57.
- Koskenniemi, K. (1983). *Two-Level Morphology: a General Computational Model for Word-Form Recognition and Production*. PhD thesis, University of Helsinki. Publications of the Department of General Linguistics, University of Helsinki, No. 11. Helsinki.
- Koskenniemi, K. (1990). Finite-state parsing and disambiguation. In Karlgren, H., editor, *Proceedings of the 13th International Conference on Computational Linguistics*, volume 2, pages 229–232, Helsinki. ACL.
- Krieger, H.-U. and Schaefer, U. (1994). TDL—a type description language of HPSG. Technical report, Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Saarbrücken, Germany.
- Krifka, M. (1989). Nominal reference, temporal constitution and quantification in event semantics. In Bartsch, R., van Benthem, J., and van Emde-Boas, P., editors, *Semantics and Contextual Expressions*, pages 75–115. Foris, Dordrecht.

- Kupiec, J. (1992). Robust part-of-speech tagging using a hidden Markov model. *Computer Speech and Language*, 6.
- Kwasny, S. and Sonheimer, N. (1981). Relaxation techniques for parsing ill-formed input. *American journal of Computational Linguistics*, 7(2):99–108.
- Lafferty, J., Sleator, D., and Temperley, D. (1992). Grammatical trigrams: a probabilistic model of link grammar. In Goldman, R., editor, *AAAI Fall Symposium on Probabilistic Approaches to Natural Language Processing*, Cambridge, Massachusetts. AAAI Press.
- Lambek, J. (1958). The mathematics of sentence structure. *American Mathematical Monthly*, 65:154–170.
- Lang, B. (1974). Deterministic techniques for efficient non-deterministic parsers. In Loeckx, J., editor, *Proceedings of the 2nd Colloquium on Automata, Languages and Programming*, pages 255–269, Saarbrücken, Germany. Springer-Verlag.
- Lang, B. (1989). A generative view of ill-formed input processing. In *ATR Symposium on Basic Research for Telephone Interpretation*, Kyoto, Japan.
- Lari, K. and Young, S. J. (1990). The estimation of stochastic context-free grammars using the Inside-Outside algorithm. *Computer Speech and Language Processing*, 4:35–56.
- Leech, G. and Garside, R. (1991). Running a grammar factory: the production of syntactically analysed corpora or ‘treebanks’. In Johansson, S. and Stenstrom, A., editors, *English Computer Corpora: Selected Papers and Bibliography*. Mouton de Gruyter, Berlin.
- Leech, G., Garside, R., and Bryant, M. (1994). The large-scale grammatical tagging of text. In Oostdijk, N. and de Haan, P., editors, *Corpus-Based Research into Language*, pages 47–63. Rodopi, Atlanta.
- Levinson, S. C. (1983). *Pragmatics*. Cambridge University Press.
- Lucchesi, C. L. and Kowaltowski, T. (1993). Applications of finite automata representing large vocabularies. *Software-Practice and Experience*, 23(1):15–30.
- Magerman, D. M. and Marcus, M. P. (1991). Pearl: A probabilistic chart parser. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, Pacific Grove, California. Defense Advanced Research Projects Agency, Morgan Kaufmann.
- Magerman, D. M. and Weir, C. (1992). Efficiency, robustness and accuracy in Picky chart parsing. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, University of Delaware. Association for Computational Linguistics.
- Marcus, M., Hindle, D., and Fleck, M. (1983). D-theory: talking about talking about trees. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pages 129–136, Cambridge, Massachusetts. Association for Computational Linguistics.
- Marcus, M. P. (1980). *A Theory of Syntactic Recognition for Natural Language*. MIT Press, Cambridge, Massachusetts.
- Marshall, I. (1983). Choice of grammatical word-class without global syntactic analysis: tagging words in the LOB corpus. *Computers in the Humanities*, 17:139–150.
- Maxwell, John T., I. and Kaplan, R. M. (1989). An overview of disjunctive constraint satisfaction. In Tomita, M., editor, *Proceedings of the First International Workshop on Parsing Technology*, Pittsburgh, Pennsylvania. Carnegie-Mellon University.

- McCord, M. C. (1980). Slot grammars. *American journal of Computational Linguistics*, 6(1):255–286.
- McCord, M. C. (1989). Design of LMT: A Prolog-based machine translation system. *Computational Linguistics*, 15(1):33–52.
- McKeown, K., Kukich, K., and Shaw, J. (1994). Practical issues in automatic documentation generation. In *Proceedings of the Fourth Conference on Applied Natural Language Processing*, pages 7–14, Stuttgart, Germany. ACL, Morgan Kaufmann.
- McRoy, S. and Hirst, G. (1990). Race-based parsing and syntactic disambiguation. *Cognitive Science*, 14:313–353.
- Mel'čuk, I. A. (1988). *Dependency Syntax: Theory and Practice*. State University of New York Press, Albany, New York.
- Montague, R. (1973). The proper treatment of quantification in ordinary English. In Hintikka, J., editor, *Approaches to Natural Language*, pages 221–242. Reidel.
- Moortgat, M. (1988). *Categorial Investigations: Logical and Linguistic Aspects of the Lambek Calculus*. PhD thesis, University of Amsterdam, The Netherlands.
- Murveit, H., Butzberger, J., Digilakis, V., and Weintraub, M. (1993). Large-vocabulary dictation using SRI's DE-CIPHER speech recognition system: Progressive search techniques. In *Proceedings of the 1993 International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 319–322, Minneapolis, Minnesota. Institute of Electrical and Electronic Engineers.
- Nguyen, L., Schwartz, R., Kubala, F., and Placeway, P. (1993). Search algorithms for software-only real-time recognition with very large vocabularies. In *Proceedings of the 1993 ARPA Human Language Technology Workshop*, pages 91–95, Princeton, New Jersey. Advanced Research Projects Agency, Morgan Kaufmann.
- Nilsson, N. J. (1980). *Principles of Artificial Intelligence*. Tioga Publishing Company, Palo Alto, California.
- Nunberg, G. (1990). The linguistics of punctuation. Technical Report Lecture Notes 18, CSLI, Stanford, California.
- Nunberg, G. and Zaenen, A. (1992). Systematic polisemy in lexicology and lexicography. In *Proceedings of Eurolex 92*, Tampere, Finland.
- Oostdijk, N. (1991). *Corpus Linguistics and the Automatic Analysis of English*. Rodopi, Amsterdam, Atlanta.
- Ostler, N. and Atkins, B. T. S. (1992). Predictable meaning shifts: Some linguistic properties of lexical implication rules.
- Partee, B. (1986). Noun phrase interpretation and type shifting principles. In Groenendijk, J. et al., editors, *Studies in Discourse Representation Theory and the Theory of Generalised Quantifiers*, pages 115–144. Foris, Dordrecht.
- Paul, D. B. (1992). An efficient A\* stack decoder algorithm for continuous speech recognition with a stochastic language model. In *Proceedings of the 1992 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 25–28, San Francisco. Institute of Electrical and Electronic Engineers.
- Pereira, F. C. N. (1985). A new characterization of attachment preferences. In Dowty, D. R., Karttunen, L., and Zwicky, A. M., editors, *Natural Language Parsing—Psychological, Computational and Theoretical perspectives*, pages 307–319. Cambridge University Press.
- Pereira, F. C. N. and Schabes, Y. (1992). Inside-outside reestimation from partially bracketed corpora. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, pages 128–135, University of Delaware. Association for Computational Linguistics.

- Petheroudakis, J. (1991). MORPHOGEN automatic generator of morphological information for base form reduction. Technical report, Executive Communication Systems ECS, Provo, Utah.
- Pollard, C. and Sag, I. (1994). *Head-driven Phrase Structure Grammar*. Center for the Study of Language and Information (CSLI) Lecture Notes. Stanford University Press and University of Chicago Press.
- Pollard, C. and Sag, I. A. (1987). *An Information-Based Approach to Syntax and Semantics: Fundamentals*. Number 13 in Center for the Study of Language and Information (CSLI) Lecture Notes. Stanford University Press and Chicago University Press.
- Prawitz, D. (1965). *Natural Deduction: A Proof-Theoretical Study*. Almqvist and Wiksell, Uppsala, Sweden.
- Pritchett, B. (1988). Garden path phenomena and the grammatical basis of language processing. *Language*, 64(3):539–576.
- Pustejovsky, J. (1991). The generative lexicon. *Computational Linguistics*, 17(4).
- Pustejovsky, J. (1994). Linguistic constraints on type coercion. In St. Dizier, P. and Viegas, E., editors, *Computational Lexical Semantics*. Cambridge University Press.
- Pustejovsky, J. and Boguraev, B. (1993). Lexical knowledge representation and natural language processing. *Artificial Intelligence*, 63:193–223.
- Revuz, D. (1991). *Dictionnaires et lexiques, méthodes et algorithmes*. PhD thesis, Université Paris, Paris.
- Ritchie, G. D., Russell, G. J., Black, A. W., and Pulman, S. G. (1992). *Computational Morphology*. MIT Press, Cambridge, Massachusetts.
- Roche, E. (1993). Dictionary compression experiments. Technical Report IGM 93-5, Université de Marne la Vallée, Noisy le Grand, France.
- Sampson, G. (1994). Susanne: a doomsday book of English grammar. In Oostdijk, N. and de Haan, P., editors, *Corpus-based Linguistics: A Festschrift for Jan Aarts*, pages 169–188. Rodopi, Amsterdam.
- Sampson, G., Haigh, R., and Atwell, E. (1989). Natural language analysis by stochastic optimization: a progress report on project APRIL. *Journal of Experimental and Theoretical Artificial Intelligence*, 1:271–287.
- Sanfilippo, A. (1993). LKB encoding of lexical knowledge. In Briscoe, T., Copestake, A., and de Paiva, V., editors, *Default Inheritance within Unification-Based Approaches to the Lexicon*. Cambridge University Press.
- Sanfilippo, A. (1995). Lexical polymorphism and word disambiguation. In *Working Notes of the AAAI Spring Symposium on Representation and Acquisition of Lexical Knowledge: Polysemy, Ambiguity and Generativity*. Stanford University.
- Sanfilippo, A., Benkerimi, K., and Dwehus, D. (1994). Virtual polysemy. In *Proceedings of the 15th International Conference on Computational Linguistics*, Kyoto, Japan.
- Schabes, Y. (1990). *Mathematical and Computational Aspects of Lexicalized Grammars*. PhD thesis, University of Pennsylvania, Philadelphia. Also technical report (MS-CIS-90-48, LINC LAB179) from the Department of Computer Science.
- Schabes, Y. (1992). Stochastic lexicalized tree-adjoining grammars. In *Proceedings of the 14th International Conference on Computational Linguistics*, Nantes, France. ACL.

- Schabes, Y., Roth, M., and Osborne, R. (1993). Parsing the Wall Street Journal with the inside-outside algorithm. In *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics*, Utrecht University, The Netherlands. European Chapter of the Association for Computational Linguistics.
- Schüller, G., Zierl, M., and Hausser, R. (1993). MAGIC. A tutorial in computational morphology. Technical report, Friedrich-Alexander Universität, Erlangen, Germany.
- Seneff, S. (1992). TINA: A natural language system for spoken language applications. *Computational Linguistics*, 18(1):61–86.
- Sharman, R., Jelinek, F., and Mercer, R. L. (1990). Generating a grammar for statistical training. In *Proceedings of the Third DARPA Speech and Natural Language Workshop*, pages 267–274, Hidden Valley, Pennsylvania. Defense Advanced Research Projects Agency, Morgan Kaufmann.
- Shieber, S. M. (1983). Sentence disambiguation by a shift-reduce parsing technique. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pages 113–118, Cambridge, Massachusetts. Association for Computational Linguistics.
- Shieber, S. M. (1992). *Constraint-Based Grammar Formalisms*. MIT Press, Cambridge, Massachusetts.
- Shieber, S. M., Uszkoreit, H., Robinson, J., and Tyson, M. (1983). *The formalism and Implementation of PATR-II*. SRI International, Menlo Park, California.
- Sleator, D. and Temperley, D. (1991). Parsing English with a link grammar. Technical report CMU-CS-91-196, Department of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania.
- Sproat, R. (1992). *Morphology and Computation*. MIT Press, Cambridge, Massachusetts.
- Stabler, Edward P., J. (1992). *The Logical Approach to Syntax: Foundations, Specifications and Implementations of Theories of Government and Binding*. MIT Press, Cambridge, Massachusetts.
- Stevenson, S. (1993). A competition-based explanation of syntactic attachment preferences and garden path phenomena. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 266–273, Ohio State University. Association for Computational Linguistics.
- Stolcke, A. and Omohundro, S. (1993). Hidden Markov model induction by Bayesian model merging. In Hanson, S. J., Cowan, J. D., and Giles, C. L., editors, *Advances in Neural Information Processing Systems 5*, pages 11–18. Morgan Kaufmann.
- Teitelbaum, R. (1973). Context-free error analysis by evaluation of algebraic power series. In *Proceedings of the Fifth Annual ACM Symposium on Theory of Computing*, pages 196–199, Austin, Texas.
- Tomita, M. (1987). An efficient augmented context-free parsing algorithm. *Computational Linguistics*, 13(1):31–46.
- Touretzky, D. S., Horty, J. F., and Thomason, R. M. (1987). A clash of intuitions: the current state of nonmonotonic multiple inheritance systems. In *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, pages 476–482, Milan, Italy. Morgan Kaufmann.
- Turner, R. (1988). A theory of properties. *The journal of Symbolic Logic*, 54.
- Turner, R. (1992). Properties, propositions and semantic theory. In Rosner, M. and Johnson, R., editors, *Computational Linguistics and Formal Semantics*. Cambridge University Press, Cambridge.

- Tzoukermann, E. and Liberman, M. Y. (1990). A finite-state morphological processor for Spanish. In Karlgren, H., editor, *Proceedings of the 13th International Conference on Computational Linguistics*, volume 3, pages 277–286, Helsinki. ACL.
- Uszkoreit, H. (1986). Categorical unification grammars. In *Proceedings of the 11th International Conference on Computational Linguistics*, Bonn. ACL.
- van Eijck, J. and de Vries, F. J. (1992). A sound and complete calculus for update logic. In Dekker, P. and Stokhof, M., editors, *Proceedings of the Eighth Amsterdam Colloquium*, pages 133–152, Amsterdam. ILLC.
- Veltman, F. (1985). *Logics for Conditionals*. PhD thesis, University of Amsterdam, Amsterdam.
- Voutilainen, A. (1994). *Three Studies of Grammar-Based Surface Parsing of Unrestricted English Text*. PhD thesis, University of Helsinki, Department of General Linguistics, University of Helsinki.
- Ward, W. (1991a). Evaluation of the CMU ATIS system. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop*, pages 101–105, Pacific Grove, California. Defense Advanced Research Projects Agency, Morgan Kaufmann.
- Ward, W. (1991b). Understanding spontaneous speech: the Phoenix system. In *Proceedings of the 1991 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 365–367, Toronto. Institute of Electrical and Electronic Engineers.
- Younger, D. H. (1967). Recognition and parsing of context-free languages in time  $n^3$ . *Information and Control*, 10(2):189–208.
- Zeevat, H., Klein, E., and Calder, J. (1987). An introduction to unification categorial grammar. In Haddock, J. N., Klein, E., and Morrill, G., editors, *Edinburgh Working Papers in Cognitive Science, volume 1: Categorical Grammar, Unification Grammar, and Parsing*, volume 1 of *Working Papers in Cognitive Science*. Centre for Cognitive Science, University of Edinburgh.

