Efficient Parsing for Korean and English: A Parameterized Message-Passing Approach

Bonnie J. Dorr*
University of Maryland

Dekang Lin†
University of Manitoba–Winnipeg

Jye-hoon Lee‡
University of Maryland

Sungki Suh§
Seoul National University

1. Introduction

This article presents an efficient, implemented approach to cross-linguistic parsing based on Government Binding (GB) Theory (Chomsky 1986) and followers. One of the drawbacks to alternative GB-based parsing approaches is that they generally adopt a filter-based paradigm. These approaches typically generate all possible candidate structures of the sentence that satisfy $\overline{\mathrm{X}}$ theory, and then subsequently apply filters in order to eliminate those structures that violate GB principles. (See, for example, Abney 1989; Correa 1991; Dorr 1993; Fong 1991.) The current approach provides an alternative to filter-based designs that avoids these difficulties by applying principles to *descriptions* of structures without actually building the structures themselves. Our approach is similar to that of Lin (1993) in that structure-building is deferred until the descriptions satisfy all principles; however, the current approach differs in that it provides a parameterization mechanism along the lines of Dorr (1994) that allows the system to be ported to languages other than English. We focus particularly on the problem of processing head-final languages such as Korean.

We are currently incorporating the parser into a machine translation (MT) system called PRINCITRAN.[1] In general, parsers of existing principle-based interlingual MT systems are exceedingly inefficient, since they tend to adopt the filter-based paradigm. We combine the benefits of the message-passing paradigm with the benefits of the parameterized approach to build a more efficient, but easily extensible system, that will ultimately be used for MT. The algorithm has been implemented in C++ and successfully tested on well-known, translationally divergent sentences.

We present a general framework for parsing by message passing and describe our implementation of GB principles as attribute-value constraints. We then present the parameterization framework, demonstrating the feasibility of handling cross-linguistic variation within the message-passing framework. A technique for automatic precompilation of parameter settings is described. Finally, we compare the efficiency of the

* Department of Computer Science, University of Maryland, College Park, Maryland 20742. E-mail: bonnie@cs.umd.edu
† Department of Computer Science, University of Manitoba, Winnipeg, Manitoba, Canada, R3T 2N2. E-mail: lindek@cs.umanitoba.ca
‡ Department of Computer Science, University of Maryland, College Park, MD 20742. E-mail: jlee@cs.umd.edu
§ Language Research Institute, Seoul National University, Seoul, 151-742, Korea. E-mail: sksuh@alliant.snu.ac.kr

[1] The name PRINCITRAN is derived from the names of two systems, UNITRAN (Door 1993) and PRINCIPAR (Lin 1993).

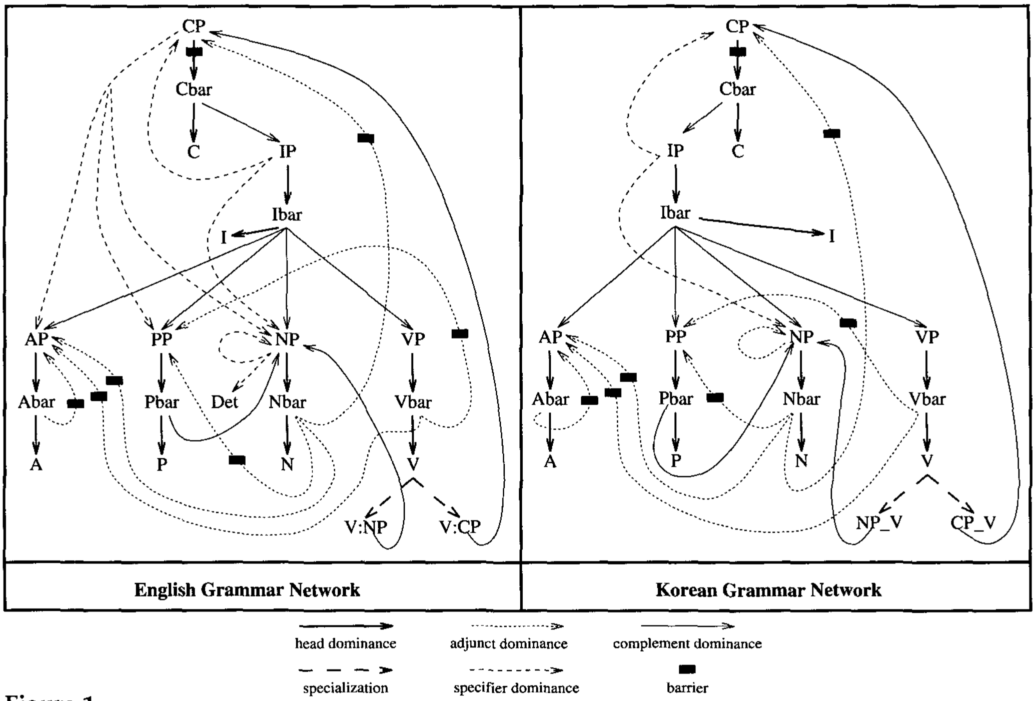

Figure 1
Network representation of English and Korean grammar.

parser to that of the original CFG algorithm as well as Tomita's algorithm (Tomita 1986) on a test suite of representative sentences. We argue that the efficiency of the system is not simply a side effect of using an efficient programming language (i.e., C++), but that the algorithm is inherently efficient, independent of the programming language used for the implementation.

2. Message Passing Paradigm

There has been a great deal of interest in exploring new paradigms of parsing, especially nontraditional parallel architectures for natural language processing (Abney 1989; Cottrell 1989; Selman and Hirst 1985, among many others). Recent work (Stevenson 1994) provides a survey of symbolic, nonsymbolic, and hybrid approaches. Stevenson's model comes the closest in design to the current principle-based message-passing model in that it uses distributed message passing as the basic underlying mechanism and it encodes GB principles directly (i.e., there are precise correspondences between functional components and linguistic principles). However, the fundamental goals of the two approaches are different: Stevenson's objective concerns the modeling of human processing behavior and producing a single parse at the end. Her system incorporates,a number of psycholinguistic-based processing mechanisms for handling ambiguity and making attachment decisions. Our model, on the other hand, is more concerned with efficiency issues, broad-scale coverage, and cross-linguistic applicability; we produce all possible parse alternatives wherever disambiguation requires extra-sentential information.

We provide a language-independent processing mechanism that accommodates structurally different languages (e.g., head-initial vs. head-final) with equally efficient run times. The grammar for each language is encoded as a network of nodes that represent grammatical categories (e.g., NP, Nbar, N) or subcategories, such as V:NP (i.e., a transitive verb that takes an NP as complement). Figure 1 depicts portions of the grammar networks used for English and Korean.

There are two types of links in the network: **subsumption links** (e.g., V to V:NP) and **dominance links** (e.g., Nbar to N). A dominance link from α to β is associated with an integer id that determines the linear order between β and other categories immediately dominated by α, and a binary attribute to specify whether β is optional or obligatory.[2]

Input sentences are parsed by passing messages in the grammar network. The nodes in the network are computing agents that communicate with each other by sending messages in the reverse direction of the links. Each node locally stores a set of items. An item is a triplet that represents an $\bar{X}$ structure α: <surface-string, attribute-values, source-messages>, where *surface-string* is an integer interval [i,j] denoting the i'th to j'th word in the input sentence; *attribute-values* specifies syntactic features of the root node (β); and *source-messages* is a set of messages that represent immediate constituents of β and from which this item is combined. Each node has a completion predicate that determines whether an item at the node is "complete," in which case the item is sent as a message to other nodes.

When a node receives an item, it attempts to form new items by combining it with items from other nodes. Two items, $<[i_1,j_1], A_1, S_1>$ and $<[i_2,j_2], A_2, S_2>$, can be combined if: (1) their surface strings are adjacent to each other: $i_2 = j_1+1$; (2) their attribute values A_1 and A_2 are unifiable; and (3) the source messages come via different links: links(S_1) $\cap$ links(S_2) = $\emptyset$, where links(S) is a function that, given a set of messages, returns the set of links via which the messages arrived. The result of the combination is a new item, $<[i_1,j_2]$, unify(A_1, A_2), $S_1 \cup S_2>$. Once a sentence has been parsed, the corresponding parse trees are retrieved from a parse forest one by one. Details are given in Lin (1993).

3. Implementation of Principles

GB principles are implemented as **local constraints** attached to nodes and **percolation constraints** attached to links. All items at a node must satisfy the node's local constraint. A message can be sent across a link only if it satisfies the link's percolation constraint.[3] We will discuss three examples to illustrate the general idea of how GB principles are interpreted as local and percolation constraints. See Lin (1993) for more details.

3.1 $\bar{X}$ Theory

The central idea behind $\bar{X}$ theory is that a phrasal constituent has a layered structure. Every phrasal constituent is considered to have a head ($X^0 \equiv X$), which determines the

[2] For the purpose of readability, we have omitted integer id's in the graphical representation of the grammar network. Linear ordering is indicated by the starting points of links. For example, C precedes IP in the English network of Figure 1.

[3] The idea of constraint application through feature passing among nodes is analogous to techniques applied in the TINA spoken language system (Seneff 1992) except that, in our design, the grammar network is a *static* data structure; it is not *dynamically* modified during the parsing process. Thus, we achieve a reduction space requirements. Moreover, our design achieves a reduction in time requirements because we do not retrieve a structure until the resulting parse descriptions satisfy all the network constraints.

properties of the phrase containing it. A phrase potentially contains a complement, resulting in a one-bar level ($\bar{X} \equiv$ Xbar) projection; it may also contain a specifier (or modifier), resulting in a double-bar level ($\bar{\bar{X}} \equiv$ XP) projection. The phrasal representation assumed in the current framework is the following:

1. $[_{XP}$ Specifier $[_{Xbar}$ Complement X]]

We implement the relative positioning of Specifier, Complement, and Head constituents by means of dominance links as shown in each of the networks of Figure 1. In addition, adjuncts are associated with the Xbar level by means of an adjunct-dominance link in the grammar network. The structure in 1 represents the relative order observed in Korean.

3.2 Trace Theory

A trace represents a position from which some element has been extracted.[4] The main constraint of Trace Theory is the Subjacency Condition, which prohibits movement across "too many" barriers. (The notion of "too many" is specified on a per-language basis, as we will see shortly.)

An attribute named `barrier` is used to implement this principle. A message containing the attribute value `-barrier` is used to represent an $\bar{X}$ structure containing a position out of which a wh-constituent has moved, but without yet crossing a barrier. The value `+barrier` means that the movement has already crossed one barrier. Certain dominance links in the network are designated as barrier links (indicated in Figure 1 by solid rectangles). The Subjacency condition is implemented by the percolation constraints attached to the barrier links, which block any message with `+barrier` and changes `-barrier` to `+barrier` (i.e., it allows the message to pass through).

3.3 Case Theory

Case theory requires that every NP be assigned abstract case. The Case Filter rules out sentences containing an NP with no case. Case is assigned structurally to a syntactic position governed by a case assigner. Roughly, a preposition assigns Oblique Case to a prepositional object NP; a transitive verb assigns Accusative Case to a direct object NP; and tensed Infl(ection) assigns Nominative Case to a subject NP.

The implementation of case theory in our system is based on the following attribute values: `ca`, `govern`, `cm`. The attribute values `+ca` and `+govern` are assigned by local constraints to items representing phrases whose heads are case assigners (e.g., tensed I) and governors (e.g., V), respectively. A Case Filter violation is detected if an item containing `-cm` is combined with another item containing `-ca +govern`.

4. Implementation of Parameters

While the principles described in the previous section are intended to be language-independent, the structure of each grammar network in Figure 1 is too language-specific to be applicable to languages other than the one for which it is designated. The most obvious language-specific feature is the ordering of head links with respect to complement links; in the graphical representation, link ordering of this type is indicated by the starting points of links, e.g., C precedes IP under Cbar since the link leading to C is to the left of the link leading to IP. In the English network, all

[4] A trace is represented as t_i, where i is a unique index referring to an antecedent.

phrasal heads precede their complements. In head-final languages such as Korean, the reverse order is required. In order to capture this distinction, we incorporate the parameterization approach of Dorr (1994) into the message-passing framework so that grammar networks can be automatically generated on a per-language basis.

The reason the message-passing paradigm is so well-suited to a parameterized model of language parsing is that, unlike head-driven models of parsing, the main message-passing operation is capable of combining two nodes (in any order) in the grammar network. The result is that a head-final language such as Korean is as efficiently parsed as a head-initial language such as English. What is most interesting about this approach is that the parameterized model is consistent with experimental results (see, for example, Suh [1993]) that suggest that constituent structure is computed prior to the appearance of the head in Korean.

We will first present our approach to parameterization of each subtheory of grammar and then describe the automatic construction of grammar networks for English and Korean using the parameter settings.

4.1 $\overline{X}$ Theory

$\overline{X}$ theory assumes that a constituent order parameter is used for specifying phrasal ordering on a per-language basis:

2. **Constituent Order:** The relative order between the head and its complement can vary, depending on whether the language in question is (i) head-initial or (ii) head-final.

The structure above represents the relative order observed in Korean, i.e., the head-final parameter setting (ii). In English, the setting of this parameter is (i). This ordering information is encoded in the grammar network by virtue of the relative ordering of integer id's associated with network links.

4.2 Trace Theory

In general, adjunct nodes are considered to be barriers to movement. However, Korean allows the head noun of a relative clause to be construed with the empty category across more than one intervening adjunct node (CP), as shown in the following:

3. [$_{CP}$ [$_{CP}$ t_1 t_2 kyengyengha-ten] hoysa$_2$-ka manghayperi-n] Bill$_1$-un
   yocum uykisochimhay issta
   [$_{CP}$ [$_{CP}$ managed-Rel] company-Nom is bankrupt-Rel] Bill-Top
   these days depressed is
   'Bill, who is such a person that the company he was managing has been bankrupt, is depressed these days'

The subject NP 'Bill' is coindexed with the trace in the more deeply embedded relative clause. If we assume, following Chomsky (1986), that relative clause formation involves movement from an inner clause into an outer subject position, then the grammaticality of the above example suggests that the Trace theory must be parameterized so that crossing more than one barrier is allowed in Korean. Our formulation of this parametric distinction is as follows:

4. **Barriers:** (i) only one crossing permitted; (ii) more than one crossing permitted.

In English the setting would be (i); in Korean the setting would be (ii).

4.3 Case Theory

In general, it is assumed that the relation between a case assigner and a case assignee is biunique. However, this assumption rules out so-called multiple subject constructions, which are commonly used in Korean:

5. John-i phal-i pwureciessta
   -Nom arm-Nom was broken
   'John is in the situation that his arm has been broken'

The grammaticality of the above example suggests that nominative case in Korean must be assigned by something other than tensed Infl(ection). Thus, we parameterize case assignment as follows:

6. **Case Assignment:** Accusative case is assigned by transitive V;
   Nominative case is assigned by (i) tensed Infl(ection); (ii) IP predication.

In a biunique case-assignment language such as English, the setting for Nominative case assignment would be (i); in Korean, the settings would be (i) and (ii).

4.4 Construction of Grammar Network from Parameter Settings

We have just seen that certain types of syntactic parameterization may be captured in the grammar network. In addition to these, there are syntactic parameters that must be programmed into the message-passing mechanism itself, not just into the grammar network. Our focus is on the automatic construction of the Korean and English grammar networks from $\overline{\text{X}}$ parameter settings. The grammar network construction algorithm consists of two steps: the first defines the basic structural description (i.e., bar-level nodes); and the second defines the satellites (i.e., adjunct and specifier nodes). The English and Korean grammar networks in Figure 1 are the result of executing this algorithm on the Korean $\overline{\text{X}}$ parameter settings.

5. Results of Time Test Comparisons

As a broad-coverage system, PRINCITRAN is very efficient. The parsing component (PRINCIPAR) processes real-world sentences 20–30 words long from sources such as the *Wall Street Journal* within a couple of seconds. The complexity of the current version of the system has not yet been formally determined. However, we claim that the efficiency of the system is not purely a result of using an efficient programming language (C++); this has been achieved by running experiments that compare the performance of the parser with two alternative CFG parsers. Since PRINCIPAR has a much broader coverage than these alternative approaches, the absolute measurements do not provide a complete picture of how these three systems compare. However, the most interesting point is that the trends of the three performance levels relative to sentence length are essentially the same. If PRINCIPAR had an **average** case complexity that was exponential relative to sentence length, but had only managed to be efficient because of the implementation language, the sentence length vs. performance curve would clearly be different from the curves for CFG parsers, which are known to have a **worst** case complexity that is polynomial relative to sentence length.

The two CFG parsers used for comparison are: a C implementation of Tomita's parser by Mark Hopkins (University of Wisconsin-Milwaukee, 1993) and the CFG

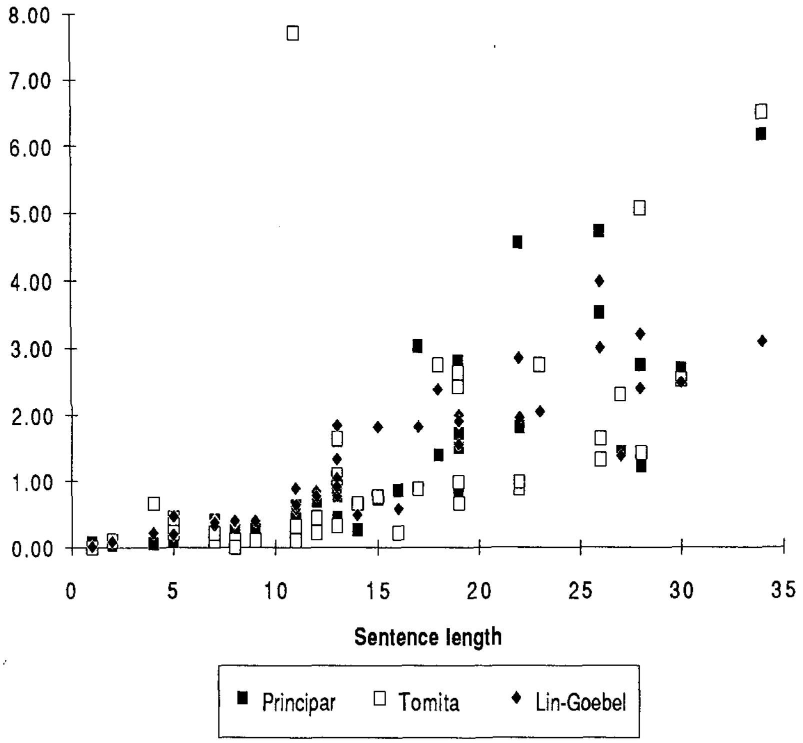

Figure 2
Adjusted timings of three parsers.

parser in Lin and Goebel (1993). The test sentences are from Tomita (1986). There are 40 of them. The sentence lengths vary from 1 to 34 words with an average of 15.18. Both CFG parsers use the Grammar III in Tomita (1986, pp. 172–6), which contains 220 rules, and a small lexicon containing only the words that appear in the test sentences. The lexicon in PRINCIPAR, on the other hand, contains about 90,000 entries extracted from machine-readable dictionaries.

Tomita's parser runs about 10 times faster than PRINCIPAR; Lin and Goebel's runs about twice as fast. To make the parsing time vs. sentence length distribution of these three parsers more comparable, we normalized the curves; the parsing time of each of the CFG parses was multiplied by a constant so that they would have the same average time as PRINCIPAR. The adjusted timings are plotted in Figure 2. These results show that PRINCIPAR compares quite well with both CFG parsers.

6. Implications for Machine Translation

Our ultimate objective is to incorporate the parameterized parser into an interlingual MT system. The current framework is well suited to an interlingual design, since the linking rules between the syntactic representations given above and the underlying lexical-semantic representation are well defined. We adopt Lexical Conceptual Structure (LCS) and use a parameter-setting approach to handle well-known, translationally divergent sentences.

Consider the following English and Korean sentences:[5]

7. **Structural Divergence:**
E: John married Sally .15 seconds
K: John-i Sally-wa kyelhonhayssta .12 seconds
-Nom -with married
'John married with Sally'

8. **Conflational Divergence:**
E: John helped Bill .10 seconds
K: John-i Bill-eykey towum-ul cwuessta .19 seconds
-Nom -Dative help-Acc gave
'John gave help to Bill'

9. **Categorial Divergence:**
E: John is fond of music .12 seconds
K: John-un umak-ul coahanta .07 seconds
-Nom music-Acc like
'It is John (who) likes music'

In general, the times demonstrate a speedup of two to three orders of magnitude over previous principle-based parsers on analogous examples such as those given in Dorr (1993). Even more significant is the negligible difference in processing time between the two languages, despite radical differences in structure, particularly with respect to head-complement positioning. This is an improvement over previous parameterized approaches in which cross-linguistic divergences frequently induced timing discrepancies of one to two orders of magnitude due to the head-initial bias that underlies most parsing designs.

7. Future Work and Conclusions

Three areas of future work are relevant to the current framework: (1) scaling up the Korean dictionary, which currently has only a handful of entries for testing purposes;[6] (2) the installation of a Kimmo-based processor for handling Korean morphology; and (3) the incorporation of nonstructural parameterization (i.e., parameters not pertaining to $\overline{X}$ theory such as barriers and case assignment).

A preliminary investigation has indicated that the message-passing paradigm is useful for generation as well as parsing, thus providing a suitable framework for bidirectional translation. Our algorithm for generation is similar to that of parsing in that both construct a syntactic parse tree over an unstructured or partially structured set of lexical items. The difference is characterized as follows: in parsing, the inputs are sequences of words and the output is a structure produced by combining two adjacent trees into a single tree at each processing step; in generation, the inputs are a set of unordered words with dependency relationships derived from the interlingua

[5] The results shown above were obtained from running the program on a Sparcstation ELC. These are not necessarily geared toward demonstrating the full capability of the parser, which handles many types of syntactic phenomena, including complex movement types. (See Lin [1993] for more details.) Rather, these examples are intended to illustrate that the parser is able to handle translationally contrastive sentences equally efficiently.

[6] Our English dictionary has 90,000 entries, constructed automatically by applying a set of conversion routines to OALD entries. We have begun negotiations with the LDC for the acquisition of a Korean MRD, for which we intend to construct similar routines.

(LCS). The generation algorithm must produce structures that satisfy the same set of principles and constraints as the parsing algorithm.

In summary, we have shown that the parametric message-passing design is an efficient and portable approach to parsing. We have automated the process of grammar-network construction and have demonstrated that the system handles well-known, translationally divergent sentences.

Acknowledgments

Bonnie Dorr and her students, Jye-hoon Lee and Sungki Suh, have been partially supported by the Army Research Office under contract DAAL03-91-C-0034 through Battelle Corporation, by the National Science Foundation under grant IRI-9120788 and NYI IRI-9357731, and by the Army Research Institute under contract MDA-903-92-R-0035 through Microelectronics and Design, Inc.
Dekang Lin has been supported by Natural Sciences and Engineering Research Council of Canada grant OGP121338.

References

Abney, S. (1989). "A computational model of human parsing." *Journal of Psycholinguistic Research*, 18(1), 129–144.

Chomsky, N. (1986). *Knowledge of Language: Its Nature, Origin and Use*. MIT Press.

Correa, N. (1991). "Empty categories, chains, and parsing." In *Principle-Based Parsing: Computation and Psycholinguistics*, edited by R. Berwick, S. Abney, and C. Tenny, 83–121. Kluwer Academic Publishers.

Cottrell, G. (1989). *A Connectionist Approach to Word Sense Disambiguation*. Morgan Kaufmann.

Dorr, B. (1993). *Machine Translation: A View from the Lexicon*. MIT Press.

Dorr, B. (1994). "Machine translation divergences: A formal description and proposed solution." *Computational Linguistics*, 20(4), 597–633.

Fong, S. (1991). "The computational implementation of principle-based parsers." In *Principle-Based Parsing: Computation and Psycholinguistics*, edited by R. Berwick, S. Abney, and C. Tenny, 65–82. Kluwer Academic Publishers.

Lin, D. (1993). "Principle-based parsing without overgeneration." In *Proceedings, ACL-93*. Columbus, Ohio, 112–120.

Lin, D., and Goebel, R. (1993). "Context-free grammar parsing by message passing." In *Proceedings, PACLING-93*. Vancouver, BC.

Selman, B., and Hirst, G. (1985). "A rule-based connectionist parsing system." In *Proceedings, Seventh Annual Conference of the Cognitive Science Society*. 212–219.

Seneff, S. (1992). "Tina: A natural language system for spoken language applications." *Computational Linguistics*, 18(1), 61–86.

Stevenson, S. (1994). *A competitive attachment model for resolving syntactic ambiguities in natural language parsing*. Doctoral dissertation, University of Maryland, College Park, MD.

Suh, S. (1993). "How to process constituent structure in head final languages: The case of Korean." In *Proceedings, Chicago Linguistic Society, No. 29*.

Tomita, M. (1986). *Efficient Parsing for Natural Language*. Kluwer Academic Publishers.