A General Computational Treatment of Comparatives for Natural Language Question Answering

Bruce W. Ballard

AT&T Bell Laborotories
600 Mountain Avenue
Murray Hill, N.J. 07974

Abstract

We discuss the techniques we have developed and implemented for the cross-categorial treatment of comparatives in TELI, a natural language question-answering system that's transportable among both application domains and types of backend retrieval systems. For purposes of illustration, we shall consider the example sentences "List the cars at least 20 inches more than twice as long as the Century is wide" and "Have any US companies made at least 3 more large cars than Buick?" Issues to be considered include comparative inflections, left recursion and other forms of nesting, extraposition of comparative complements, ellipsis, the *wh* element "how", and the translation of normalized parse trees into logical form.

1. Introduction

We shall describe a general treatment of comparatives that has been implemented in the context of TELI, a question-answering system which is transportable among both domains of discourse and different types of backend retrieval systems.[1] Comparatives are *important* because of the dramatic increase in expressive power they allow; they are *interesting* at least because of the variety of issues (from morphology on up) one must deal with in order to provide for them.

1. The examples in this paper illustrate TELI as a front-end to the Kandor knowledge representation system (Patel-Schneider, 1984); we will give examples in terms of a knowledge base of information about 1987 cars. TELI has produced queries for at least four different "backend" systems and has been adapted for over a dozen domains of data.

1.1 Goals

In seeking to provide TELI with general capabilities for comparatives, our primary goals have been

to formulate *cross-categorial* techniques that treat the comparativizations of different syntactic elements (e.g. adjectives, quantifiers, and measure nouns) with the same mechanisms;

to allow comparatives to be *composed* with themselves (e.g. "at least 3 more than 3 times as many") and with other syntactic features (e.g. *wh* elements);

to be *faithful* to what is known from work in theoretical linguistics; we draw from Bresnan (1973), Cushing (1982), Dik (1980), Jackendoff (1977), Sells (1985), and Winograd (1983);

to account for as many of the specific cases of comparatives found in the *literature* of implemented NL processors as possible.

1.2 Achievements

Letting <X> denote a grammatical category to be comparativized, we begin by providing for comparativized structures C{<X>} of the form

C{<X>} → (<Qual>) CC{<X>} <Comp>
<Qual> → at most | at least | no | exactly | precisely | just | only
CC{<X>} → (CC{<X>}) (<Measure>) <c1> (<X>) <c2>
<Measure> → <Number> (<Ordinal> | percent | times) |
half (again) | twice (again)
<Ordinal> → half | third | thirds | ...
<Comp> → <NP> <Etc_X>
<c1>/<c2> → -er/than | less/than | as/as

where (...) denotes optionality; "/" indicates "agreement" between comparative particles; and <Etc_X> accounts for items parallel to those in the matrix clause in which the comparative occurs (e.g. "cars that are longer than *the Regal (is (wide))*"). In addition, a variety of extrapositions (i.e. rightward and occasional leftward movement) from C{<X>} may (and sometimes must) occur. For example, both "cars *larger than the Century*" and "*larger* cars *than the Century*" are allowed.

Since we wish to allow C{<X>} structures to occur wherever <X> could occur, arbitrarily complex interactions with quantifiers (within the complement), ordinals, superlatives, raisings, *wh* elements, and other constructs must be provided for. In addition to the structures indicated by the BNF above, we allow for some simpler expressions not conventionally classified as comparatives. Some examples are "6 cars" (cf. "as many as 6 cars") and "3 inches long" (cf. "as long as 3 inches"). We also provide for structures involving the nominal counterpart of an adjective, as in "more than 185 inches *in length*".

To date, we have fully implemented a wide variety of comparatives related to adjectives, quantifiers, and measure nouns (e.g. "cars that cost *at least $100 more than the Park Avenue*"). Due to the commonality among the comparativized syntactic structures, our grammar for these three types of comparatives is produced by meta-rules suggested by the BNF rules shown above. Although the feature agreement provided by our parser is used to eliminate spurious structures such as "cars more than 3 (inches/*dollars) long", we avoid conflicts between pure numbers and measure phrases that involve a unit (e.g. "companies that make more than 3 (*dollars) cars") by having two (very nearly identical) Quantity routines in the grammar.

1.3 Limitations

In addition to some specific limitations to be stated in the remainder of the paper, there are some general limitations of our work to date, many of which are being rectified by the work mentioned in Section 8.3. (1) By analogy with conjunctions, with which comparatives share a number of properties (cf. Sager 1981, pp. 196ff), our comparative particle pairs (-er/than etc.) provide for *co-ordinate* comparatives, in contrast to pairs such as so/that, as in "Buick makes *so* many cars *that it's the largest company*." (2) Comparative complements are expected in a limited number of places. For example, "Audi makes *more* large cars *than Pontiac* in France" is recognized but "Audi makes *more* large cars in France *than Pontiac*" is not. This is because we currently propagate the evidence of having found a comparative particle ("more") to the noun phrase headed by "cars", hence the complement ("than ...") can attach there, but not to the higher level verb phrase headed by "makes". This limitation also prevents our processing "What companies make a larger car than Buick", whose *exact* meaning(s) the reader is invited to ponder. (3) Since comparative complements are based on noun phrases, neither "Audi makes *more* large cars in France *than in Germany*" nor "Audi makes large cars *more* in France *than in Germany*" is recognized. (4) We attempt no pragmatic disambiguation of semantically ambiguous comparatives. Thus, when confronted with "more than 3 inches shorter" or "more than 3 fewer cars", we provide the compositional interpretation associated with our left recursive syntax. Even expressions such as "as many" and "as large" are ambiguous between *at least* and *exactly*. (5) We attempt no anaphora processing, and so comparatives without a complement, as in "Which cars are larger?", are not processed. (6) We provide general conversion of units of measure (e.g. "2 feet longer" is the same as "24 inches longer") but they are not fully incorporated into the system.

2. An Initial Example

The mechanisms we shall describe apply a conventional series of transformations to sentences containing one or more comparatives, ultimately resulting in an executable expression. As an example of this process,[2] we'll consider the input

"List the cars at least 20 inches more than twice as long as the Century is wide"

which contains a highly comparativized adjective. First, this input is scanned and parsed, yielding the parse tree shown in Figure 1. Note that each COMPAR node has a QUANTITY node and a MODE[3] of its own. Also, the MODE of the top COMPAR (whose value is "equal") is co-indexed (indicated by the subsrcipt *i*) with the MODE feature associate with the particle ("as") that intervenes between the ADJ and its COMPAR-ARG; this assures that -er/than, less/than, and as/as pairs collocate correctly. Next, we build a "normalized" parse tree by reconstructing elements that were discontinuous in the surface structure and

2. A formal account the associated formalisms, including a BNF syntax and a denotational semantics for our "normalized parse trees" and "algebraic-logical form" language, is given in Ballard and Stumberger (1987).

3. Dashed lines indicate features, as distinct from lexical items, and empty nodes, which result from Whiz-deletion, are denoted by "?".

by performing other simplifications. This yields the following structure, whose 2-place predicate, with P (parameter) and A (argument) as variables, corresponds to "at least 20 inches more than twice as ... as".

Normalized Parse Tree:

```
(CAR (NOUN CAR)
     (COMPAR (ADJ LONG)
             (λ (P A) (≥ P (+ 20 (* 2 A))))
             (CAR (= CENTURY))
             (ADJ WIDE)))
```

Next, user-defined meanings of words and phrases are looked up[4] and the comparativization operations described in Section 6 are performed, yielding

Algebraic-Logical Form:

```
(SET (CAR P1)
     (≥ (Length-of-Car P1)
        (+ 20 (* 2 (Width-of-Car CENTURY]
```

Finally, this representation is converted into the executable expression indicated by

Final Executable Expression:

```
(SUBSET (λ (P1)
          (≥≥ (KSV P1 @S{LENGTH})
              (+ 20
                 (* 2 (KSV @I{CENTURY}
                           @S{WIDTH}))))
          (KI @F{CAR})))
```

where KSV and KI are primitive retrieval functions of the Kandor back-end; @I{...}, @F{...} and @S{...} are Lisp objects respectively denoting instances, frames, and slots in Kandor's taxonomic knowledge base; and ≥≥ is a coercion routine supplied by TELI to accommodate backend retrieval system that produce numbers in disguise (e.g. a Lisp object or a singleton set) on which the standard Lisp functions would choke.[5] However, since compositionally created structures such as the preceding one are often intolerably inefficient, optimizations are carried out *while the executable expression is being formed.* In the case at hand, the second argument of ≥≥ is constant, so it is evaluated, producing

Optimized Executable Expression:

```
(SUBSET (λ (P1)
          (≥≥ (KSV P1 @S{LENGTH}) 158))
          (KI @F{CAR}))
```

A second example, which illustrates a comparative quantifier, is given in an appendix where, as a result of optimizations analogous to those which produced the constant 158 above, the comparative "at least 3 more large cars than Buick" is eventually processed exactly as though it had been "at least 6 cars" (since Buick made 3 large cars).

3. Lexical Provisions for Comparatives

Our current repertoire of *domain-independent* lexical items associated with comparatives includes "many", "few", and "much"; "more", with 3 readings (er, er+many, er+much), following Bresnan (1972) and similar to Robinson (1982, p. 28); "fewer (er+few); "less", with 3 readings (less, er+few[6], less+much); several formatives and adverbials ("at", "least", "most", "exactly", "precisely", "only", "just", "half", "again", "times", "percent"); and a handful of spelled-out ordinals ("thirds" etc.). Though not stored in the lexicon, both integers and floating-point numbers (cf. "3.45 inches") are also involved in comparativization.

The *domain-dependent* portion of the lexicon includes members of the open categories of adjectives, measure nouns, and comparative inflections of adjectives. The scanner output for the comparative of the adjective *A* is *er+A* (e.g. "larger" becomes *er*+large).

4. Syntax for Comparatives

The basic syntax for comparatives adheres to the meta-rules given in Section 1.2. As indicated in the parse tree of Figure 1, COMPAR is never a primary tree node but is instead a *daughter* of the node being comparativized. Furthermore, since our grammar has recently taken on somewhat of an X-bar flavor (cf. Jackendoff, 1977), the complement for a comparativized item is found as either its sister or its parent's sister. Complex comparatives derive from left-recursive structures.[7] Our present grammar for comparatives is set up partly by meta-rules[8] and partly by hand-coded rules relating to such idiosyncracies as "more than 3 inches *in length*" (however, cf. "more than 6 *in number*").

4. In TELI, meanings may be arbitrary expressions in the extended first-order language discussed in Ballard and Stumberger (1987).

5. Similar functions are also supplied for arithmetic operators.

6. To the possible horror of the prescriptive grammarian, this accounts for such attrocities as "less books".

7. Though our parser operates top-down, we've incorporated a general mechanism for left recursion that's also utilized by possessives (e.g. "the newest car's company's largest competitor's smallest car").

8. Meta-rules are also used to produce the grammar for relative clauses, yes-no questions, and a host of other structures (e.g. various slash categories) from a hand-coded grammar for basic declarative sentences.

5. Parse Tree Normalization

Letting Node{<X>} denote a node of the normalized parse tree associated with an element of type <X>, comparatives involve the replacement denoted by

Node{C{<X>}}
⟶ (COMPAR Node{<X>} <Rel> <Arg> <Etc_X>)

where <Arg> corresponds to an optional noun phrase, <Etc_X> captures non-elided material associated with the matrix clause, and the 2-place-relation denoted by <Rel> is the most interesting (and by far the most complex) element produced. The algorithm that produces it converts "more", "less", and "times" respectively into +, −, and •. This process is left recursive; the relational operator is determined from the highest MODE, and by default it is assigned to be =.[9] As indicated below, these algebraic and arithmetic symbols will be preserved in the executable expression unless the word being comparativized indicates a downward direction on the scale applicable to it (e.g. "fewer", "shorter"), in which case they will be reversed (e.g. ⩾ becomes ⩽ and + becomes −). Each 2-place-relation is the body of a 2-place lambda whose variables, P and A, are associated with values obtained from a *parameter* and an *argument* against which a comparison is being made. Some example 2-place-predicates are

more than 166 inches long	(> P 166)
more than 15 feet long	(> P 180)
at most 180 inches long	(⩽ P 180)
longer than	(> P A)
at least as long as	(⩾ P A)
1 inch longer than	(⩾ P (+ 1 A))
exactly twice as long as	(= P (• 2 A))
3 times as long as	(⩾ P (• 3 A))
half again as long as	(⩾ P (• 1.5 A))
forty percent longer than	(⩾ P (• (+ (/ 40 100) 1) A))
less than one third as long as	(< P (• (/ 1 3) A))
at least 3 inches more than twice as long as	(⩾ P (+ 3 (• 2 A)))

When the measure noun appearing in an English input differs from that by which the objects being tested are measured, as indicated by the second example above, a scalar conversion is required.

9. This addresses the inherent ambiguity of as/as structures without an adverbial element, such as "exactly" or "at least". Thus, "people with 3 children" is interpreted as people with *exactly* 3 children.

6. Semantics for Comparatives

The semantics of comparativization involves converting a one-place predicate into another one-place predicate by performing arbitrarily complex operations on it. For example, if "large car" has been defined as a car whose length exceeds 190 inches, then, letting "A" denote a noun phrase complement, some examples are

long	Length(x) ⩾ 190
longer than 180 inches	Length(x) > 180
longer than A	Length(x) > Length(A)
no longer than A	Length(x) ⩽ Length(A)
twice as long as A is wide	Length(x) ⩾ 2 • Width(A)
3 inches more than twice as long as A	Length(x) > 3 + 2 • Length(A)

where each of these right-hand-sides is the body of a one-place predicate whose single variable is x.

As a second example, comparative quantifiers such as "more than 6" are handled by an identical process[10], as indicated by[11]

x has many y's	Size {y | Has(x,y)} ⩾ Constant
x has more than 6 y's	Size {y | Has(x,y)} > 6
x has more y's than A	Size {y | Has(x,y)} > Size {y | Has(A,y)}
x has at least 2 more y's than A	Size {y | Has(x,y)} ⩾ 2 + Size {y | Has(A,y)}

where the initial *Constant* denotes some arbitrary constant.

In general, comparativizing a one-place predicate takes place as follows.

1. Find (a) an appropriate *one-place function* and (b) an associated *relational operator* that tells which direction on a linear scale indicates having "more" of the property.
2. Apply the *relational operator* located above to the *modality* of the comparison to determine the relational operator that will appear in the IR+. If the relational operator of the definition being comparativized is either > or ⩾, use the mode occurring in the IR; otherwise, "reverse" the mode by doing what would be a negation but leaving untouched the = portion of the operator. Thus, the reversal of < is >, the

10. That is, we have no special purpose processing for "more than", "how many" etc.

11. We use "has" in these examples for clarity; naturally, the scope of a comparative quantifier may contain an arbitrarily complex predicate.

reversal of $\leqslant$ is $\geqslant$, and so forth. Similarly, +, and − are switched.

3. Determine the *argument* being compared against (possibly a constant).
4. *Link* these pieces together. If the argument was not constant (e.g. "... longer than at least 3 foreign cars"), *wrap* its scope around the resulting expression.

For example, if "short car" has been defined as

"x is short": Length(x) $\leqslant$ 160

then the 1-place function and relational operator are determined in step 1 to be *Length* and $\leqslant$, and thus we have

"shorter than A" ⟶ Length(x) < Length(A)
"exactly 3 inches shorter than A"
⟶ Length(x) = Length(A) − 3

7. Comparatives Containing a *Wh* Element

In addition to recognizing *wh* elements associated with a relative or interrogative clause,[12] TELI recognizes the word *how* when it appears in place of a quantity, e.g. "*how* long" (cf. "6 inches long") and "*how* many more" (cf. "6 more"[13]). Wherever *wh* appears, however, we treat its semantics as roughly "solve for *wh* such that". In the case of interrogative pronouns (e.g. "what"), this leads rather obviously to an internal representation asking for a SET. In the case of "how", this treatment is also in order since it represents a (quantity) NP. For simplicity, we produce an expression containing an unbound *wh* and later give it wide scope.[14] In particular, subsequent processing involves moving the *wh* element *upward* in the logical form tree[15] by performing appropriate transformations.

For illustration, consider the absurdly complicated example

"Buick makes 3 more than how many percent more cars than Audi?"

the comparative portion of whose internal representation[16] is

```
(λ (P A) (= P (+ (* A (+ 1 (/ WH 100))) 3]
```

At this point, we proceed with semantic processing, ignoring for the moment the presence of the unbound WH element. In the case at hand, this leads to

```
(= (COUNT (SET (CAR P1) (Make BUICK P1)))
   (+ (* (COUNT (SET (CAR P1)
                     (Make AUDI P1)))
         (+ 1
            (/ WH 100)))
      3))
```

after which we "solve for" WH to yield

```
(* (- (/ (- (COUNT (SET (CAR P1)
                        (Make BUICK P1)))
            3)
         (COUNT (SET (CAR P1)
                     (Make AUDI P1))))
      1)
   100)
```

This process is not dependent on the position in which the *wh* occurred, and thus takes the place of special-purposes interpretation routines for "how many", "How <Adjective>", and so forth.[17]

8. Discussion

Thus far, we have presented an overview of our treatment of comparatives, with as much detail as we're able to supply in a conference-length paper. Although we can offer no substantive *empirical* evidence with TELI (e.g. results of use by non-authors), we believe some of the techniques we've presented can be put to use by the reader. Further information, especially with regard to the interaction of comparatives with a variety of other types of constructs, can be found in Ballard and Stumberger (1987).

---

12. To see that *wh* is less than a "word", consider pairs such as *wh*at/*th*at, *wh*ere/*th*ere and *wh*en/*th*en. The advantage of recognizing sub-word units as the primitives on which syntax and/or semantic analysis is based should come as no surprise to anyone acquainted with the structure of languages other than English, which is unusual in coming so close to being treatable solely at the word level.
13. As stated earlier, we have adopted derivations suggested by Bresnan (1973) such as -er+many⟶more. In the case at hand, we must assume something like Q+many⟶Q, where *Q* denotes a quantity.
14. The scope is wide but not global because of inputs such as "*How* many cars does *each* US company make?"
15. Of course, our algebraic-logical forms, based on operators and their associated arguments, amount to being trees.
16. The sentence is ambiguous, with readings indicated by "3 more than [how many percent]" and "[3 more than how many] percent". As indicated earlier, we presently take the reading that favors the use of left recursion.
17. Problematic situations can arise in which simple algebraic operations aren't sufficienct. For example, in examples such as "Cars were sold to people with how many children?", we must move *wh* past a logical quantifier, rather than the arithmetic operators as shown above.

8.1 Related Work

Although the literature describing implemented NL processors contains many *examples* of comparative constructions (cf. Kirsch (1964) for a wealth of early examples), at least two qualifications may be given concerning the current "state of the art" of comparative treatment. First, the majority of the examples appearing in the literature are quite simple[18] (e.g. "more than $250") and can be prepared for by specifying a 2-place predicate in advance that's effectively equivalent to the 2-place predicate we construct from an underlying 1-place predicate by way of coercion into a 1-place function. This allows one to avoid some slippery problems of movement (which we have adressed but have certainly not disposed of), to ignore morphological subtleties (e.g. recognizing the "er" of "larger" or "more" as *-er*, a "word" to be input to the parser), and to take other shortcuts.[19] Second, although *examples* of various types of comparatives are not hard to come by, accounts of the actual *mechanisms* that treat comparatives are harder to find, as are specific statements of the *generality* which authors believe themselves to have provided for.

8.2 Levels of Representation

The architecture of TELI resembles that of similarly motivated question answering systems (cf. Grosz et al, 1987; Hafner and Godden, 1985; Bates and Bobrow, 1983 and Bates et al 1985) by comprising a linear sequence of processing stages which produce successively "lower" level representations of the input.[20] Although our parse tree format is rather conventional,[21] what we have called "normalized parse tree" and "algebraic-logical form" correspond rather loosely to what in the literature are often called "logical form" and "meaning representation", respectively. Furthermore, in the most recent work with TELI, meaningful distinctions between modules have become blurred, although the relative order in which operations are carried out is largely the same.

In seeking to compare our formalisms and processing strategies with others that have been proposed, we have found terms such as "logical form" being used in the literature in quite vague and often incompatible ways. Furthermore, we know of no compelling arguments that suggest that a psychologically plausible model of human information processing will require intermediate levels such as parse trees, logical forms, and the like. Is it even clear that there ought be be a finite number of successive "levels", whatever they might be? We are increasingly doubtful that the trappings spawned by linguists and philosophers can be put in a bag, sprinkled with Common Lisp, shaken, and expected to yield robust natural language processors. More of an interdisciplinary effort may be required than has yet been seen.

8.3 Current Work

The representation given in Section 5 fundamentally restricts us from handling comparatives whose complement is more than one level above the word being comparativized (e.g. "John persuaded his students to contribute to *more* museums *than Bill did*"). Our current work involves producing normalized parse tree structures of roughly the form

(COMPAR-2 C_i <Comp>
...
(COMPAR-1 C_i ...) ...)

where the COMPAR-1 and <Comp> structures correspond to the COMPAR structure given in Section 5; C_i provides for co-indexing when multiple comparativizations are present; and the first "..." allows for arbitrarily many levels. This calls upon us to modify the semantic processing presented in Section 6, making it resemble the treatment given to *wh* elements as described in Section 7.

9. Conclusions

We have presented algorithms aimed at the morphological, syntactic, and semantic problems associated with a large variety of comparative structures that arise in the context of question answering. We believe the extent of our coverage equals in several ways and exceeds in some ways the capabilities known to us via the literature. However,

18. Evidence of the gap between what's been studied and what may actually be important is expressed, in the context of pronoun resolution, in Hobbs (1978, p. 343) as follows: "There are classes of examples from the literature which are not ... handled by the algorithm, but they occur rarely in actual texts, and in view of the fact that the algorithm fails on much more natural and common examples, there seems to be little point in greatly complicating the algorithm to handle them."

19. The extent to which "shortcuts" are justified, from either a psychological or system designer's standpoint, is not clear. As a possibly bizarre example, consider the word "after", which could be treated as "-er •aft than", where •aft is the Anglo-Saxon root (extant only on board ship) from which current English word derives. A perhaps even more bizarre opportunity may exist for treating "rather" as "-er •rathe", where "•rathe" is a Middle English adverb meaning "quickly".

20. We're using "low" to refer to level of abstraction. Perhaps ironically, successively *higher* levels of *cognitive* information are involved in producing these "lower" level representation.

21. The *methods* whereby TELI produces parse trees are less conventional than the trees it produces, due to our provision for having the parser enforce agreements automatically while it is running, rather than doing subsequent filtering.

comparatives operate as a "meta" phenomenon and thus cut across many issues; we have ignored certain problems and knowingly treated others inadequately. Further work is certainly required, and we hope to have presented a framework in which (1) some interesting and important capabilities can be provided for now and (2) further computational studies can be carried out.

10. Acknowledgements

The author wishes to acknowledge the many insights displayed by Mark Jones and Guy Story during a number of intense discussions concerning the issues discussed in this paper.

11. References

Ballard, B. The Syntax and Semantics of User-Defined Modifiers in a Transportable Natural Language Processor. *10th International Conference on Computational Linguistics*, Stanford University, July 1984, 52-56.

Ballard, B. User Specification of Syntactic Case Frames in TELI, A Transportable, User-Customized Natural Language Processor. *11th International Conference on Computational Linguistics*, University of Bonn, August 1986, 454-460.

Ballard, B., Lusth, J., and Tinkham, N. LDC-1: A Transportable Natural Language Processor for Office Environments. *ACM Transactions on Office Information Systems* 2, 1 (1984), 1-23.

Ballard, B. and Stumberger, D. Semantic Acquisition in TELI: A Transportable, User-Customized Natural Language Processor. *24th Annual Meeting of the Association for Computational Linguistics*, Columbia University, June 1986, pp. 20-29.

Ballard, B. and Stumberger, D. The Design and Use of a Logic-Based Internal Representation Language for Backend-Independent Natural Language Processing. AT&T Bell Laboratories Technical Memorandum, October 1987.

Ballard, B. and Tinkham, N. A Phrase-Structured Grammatical Framework for Transportable Natural Language Processing. *Computational Linguistics* 10, 2 (1984), 81-96.

Bates, M., Moser, M. and Stallard, D. The IRUS Transportable Natural Language Interface. *Proc. First Int. Workshop on Expert Database Systems*, Kiawah Island, October 1984.

Bresnan, J. Syntax of the Comparative Clause Construction in English. *Linguistic Inquiry* 4, 3 (1973), 275-344.

Cushing, S. *Quantifier Meanings: A Study in the Dimensions of Semantic Competence.* North-Holland, Amsterdam, The Netherlands, 1982.

Damerau, F. Problems and Some Solutions in Customization of Natural Language Database Front Ends. *ACM Transactions on Office Information Systems* 3, 2 (1985), 165-184.

Dik, S. *Studies in Functional Grammar.* Academic Press, London, England, 1980.

Ginsparg, J. "Natural Language Products", unpublished document, 1987.

Grosz, B., Appelt, D., Martin, P., and Pereira, F. TEAM: An Experiment in the Design of Transportable Natural-Language Interfaces. *Artificial Intelligence*, 32, 2 (1987), pp. 173-243.

Hafner, C. and Godden, C. Portability of Syntax and Semantics in Datalog. *ACM Transactions on Office Information Systems* 3, 2 (1985), 141-164.

Jackendoff, R. *X-Bar Syntax: A Study of Phrase Structure.* MIT Press, Cambridge, Mass., 1977.

Kirsch, R. Computer interpretation of English text and picture patterns. *IEEE Trans. on Electronic Computers*, 1964.

Moore, R. Problems in Logical Form. *19th Meeting of the Association for Computational Linguistics*, Stanford, California, 1981, pp. 117-124.

Patel-Schneider, P. Small Can Be Beautiful in Knowledge Representation. *Proceedings of the IEEE Workshop on Principles of Knowledge-Based Systems*, Denver, Colorado, December 1984.

Robinson, J. DIAGRAM: a grammar for dialogues. *Communications of the ACM*, 25, 1 (1982), 27-47.

Sager, N. *Natural Language Information Processing: A Computer Grammar of English and Its Applications.* Addison-Wesley, 1981.

Sells, P. *Lectures on Contemporary Syntactic Theories.* Center for the Study of Language and Information, Stanford University, 1985.

Thompson, B. and Thompson, F. ASK Is Transportable in Half a Dozen Ways. *ACM Trans. on Office Information Systems* 3, 2 (1985), 185-203.

Woods, W. Semantics and Quantification in Natural Language Question Answering. *Advances in Computers*, Vol. 17, New York, Academic Press, 1978

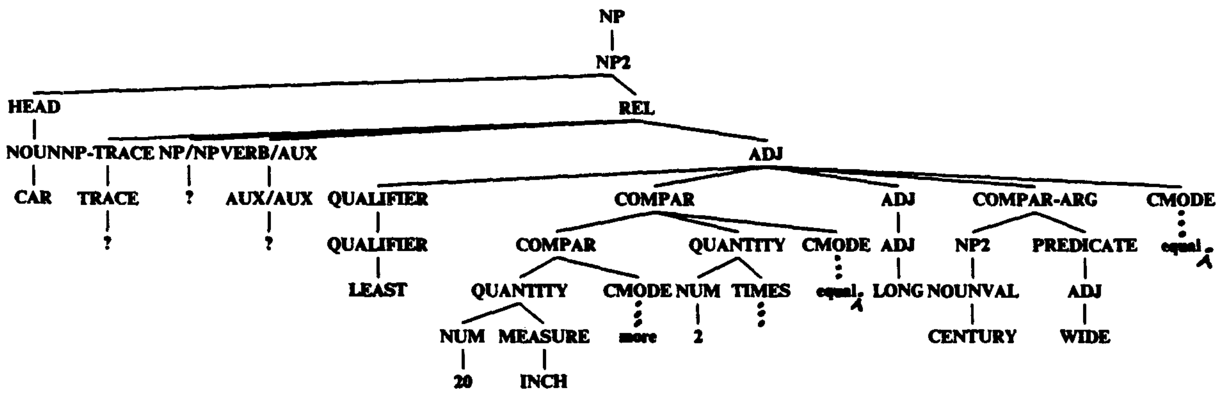

Figure 1: Parse Tree for The Example of Section 2

Appendix: Processing a Comparative Quantifier

English Input:

"Have any US companies made at least 3 more large cars than Buick?"

Normalized Parse Tree:

```
(VP (COMPANY MAKE CAR NIL NIL NIL)
    (SUBJ (COMPANY (QUANT ANY)
                   (COMPANY (ADJ US)
                            (NOUN COMPANY))))
    (OBJ (CAR (COMPAR (QUANT MANY) (≥ Q (+ CQ 3))
                      (COMPANY (= BUICK)))
              (CAR (ADJ LARGE)
                   (NOUN CAR)))))
```

Algebraic-Logical Form:

```
(QUANT (COMPANY P1) (≥ Q 1)
       (US-Company P1)
       (≥ (COUNT (SET (CAR P2)
                      (AND (> (Length-of-Car P2) 190)
                           (= (Company-of-Car P1) P2))))
          (+ 3 (COUNT (SET (CAR P2)
                           (AND (> (Length-of-Car P2) 190)
                                (= (Company-of-Car P2) BUICK)))))))
```

Final Executable Expression:

```
(GPC-SOME '(1 CQ)
          (λ (P1)
             (AND (KI? P1 @F(US-COMPANY))
                  (≥≥ (GPC-COUNT (SUBSET (λ (P2)
                                              (AND (>> (KSV P2 @S(LENGTH)) 190)
                                                   (== (KSV P2 @S(COMPANY)) P1)))
                                          (KI @F(CAR))))
                      (GPC-+ 3
                             (GPC-COUNT (SUBSET (λ (P2)
                                                     (AND (>> (KSV P2 @S(LENGTH)) 190)
                                                          (== (KSV P2 @S(COMPANY)) @I(BUICK))))
                                                 (KI @F(CAR))))))))
          (KI @F(COMPANY))))
```

Optimized Executable Expression:

```
(GPC-SOME '(1 CQ)
          (λ (P1)
             (GPC-SOME '(6 CQ)
                       (λ (P2)
                          (AND (>> (KSV P2 @S(LENGTH)) 190)
                               (== (KSV P2 @S(COMPANY)) P1)))
                          '(@I(INTEGRA) @I(NOVA) ...))))
                       (KI @F(US-COMPANY)))
```