The Grammar as Science Project - Part 1

Richard K. Larson (SUNY - Stony Brook) 2.July 1997

I. Background

A. Recent NSF Initiatives in Science Education (1986 NSF Neal Report; 1996 NSF Report: *Shaping the Future)*

The prevailing model of an undergraduate major in the sciences: graduate-program-in-miniature designed to produce professional scientists & academics (preprofessional training)

The reality: few students who take science courses become professional scientists & academics

B. The Goal

Science education should aim to produce individuals who understand the practice of scientific inquiry, not just its results; we want people who can acquire new scientific knowledge and solve new problems.

Teaching people to understand scientific inquiry means teaching higher-order, domain-independent skills (classification and categorization of data, formulation and testing of hypotheses, counterfactual reasoning ("what if...?"), pattern detection, integration of experience into a whole).

Research from Cog Psych has shown domain-independent skills can be taught. but is most effective in the context of domain-specific inquiry (Bruer 1994); it must be grounded in the detailed investigation of a rich and specific arena of facts, with its own richly connected domain-specific concepts

II. The Place of Linguistics in All This

A. Attractions of Linguistics as a Vehicle for Science Instruction

- Richly structured domain of facts that can be organized by a small number of fundamental ideas (structure, category, constituency, dependency, permutation, etc.)
- Highly accessible subject matter
- Simple technical notions can take you a long way
- Social factors
- A natural contact point between science and the humanities

B. Some Challenges

- Almost universally, students enter with an incorrect, "anthropological" view of natural language as a cultural "artifact", or human social institution
- "Hey, this is math with words!": modeling & exploring linguistic knowledge = modeling & exploring formal systems; we must teach understanding & mastery of deductive technique, as in logic and math
- Linguistic hypotheses = rules & principles, evaluated by tracing deductions & interactions, through potentially complex derivations, across a potentially broad range of linguistic expressions

III. Rethinking Undergraduate Syntax & Semantics at STONY BROOK

A. Three Central Questions

1) What is the general educational value of studying syntax & semantics?
2) What broad intellectual issues are engaged?
3) What general intellectual skills are developed?

B. Some Answers

1) S&S offers an excellent instrument for introducing students from a wide variety of backgrounds to the principles of scientific theorizing and thought.
2) S&S engages the general intellectual issues present in all scientific theorizing, and ones that arise specifically with the modern cog.sci, e.g.:
   - How does a scientist construct, test, evaluate and refine a theory?
   - How does a scientist choose between alternative theories
   - What constitute significant generalizations, and how does one capture them?
   - When does a scientist propose or assume unseen objects or structure, and how are such objects justified?
   - How secure is scientific knowledge?
   - Can one study a human phenomenon as a natural object and gain scientific understanding of it?
3) S&S offers an excellent medium through which to teach the skill of framing exact, explicit arguments for theories - the articulation of hypotheses, principles, data, and reasoning into a coherent, convincing whole.

C. Understanding Scientific Practice - "Teaching for Understanding"

Five Principles for Fostering Understanding (from Nickerson 1995)

- Start from where the students <u>are</u>:
  what do they know?
  what don't they know?
  what misconceptions do they have?
- Promote active processing & discovery
- Use appropriate representations & models
- Use simulations
- Provide supportive environment

D. The GAS Plan

Create/revise sophomore-level courses that would introduce students from a wide variety of backgrounds to scientific reasoning & procedure using syntax & semantics as the medium.

Create software tools to assist active processing & discovery.

Use appropriate representations & models - not necessarily the ones embraced by current linguistic theory.

SUNY - Stony Brook Courses	Software tool
LIN 211 Syntax	*Syntactica*
LIN 346 Language & Meaning	*Semantica*

IV. Designing the Software Tools

A. Models of Successful Tools

- *Mathematica/Maple*
- graphing calculators in the Reform Calculus Program

B. What Makes a Useful Tool

- Good tools assist the user with mechanical computations, with visualizing results, with storing outcomes, etc.
- Good tools lift no serious <u>intellectual</u> burdens from the user
- A good tool has "depth" and can be extended to different uses during the term; this allows with a collection of shallower tools, the learning curve becomes more formidable for students
- A good tool works makes heavy use of information and results that the user provides

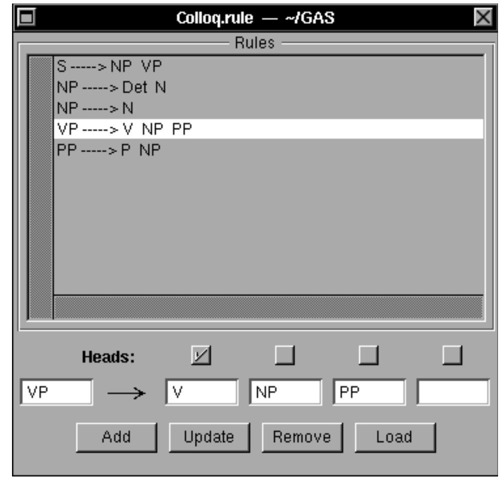
C. Basic Functions We Wanted to be Instantiated

- Calculation (of derivations)
- Visualization (of results)
- Modeling (of the human grammar mechanism)

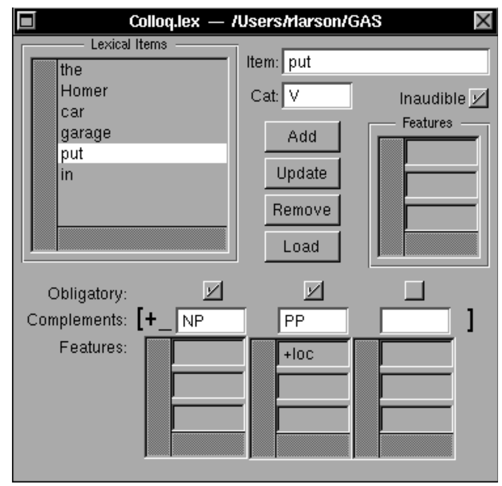
V. *SYNTACTICA :* How It Works

- User enters a syntactic theory (PS rules or PS rules + lexicon)
- User enters an input string
- *Syntactica* attempts to derive a tree for input using the rules
- *Syntactica* displays the results of successful derivations (trees and PF).
- Results can be further modified by transforms

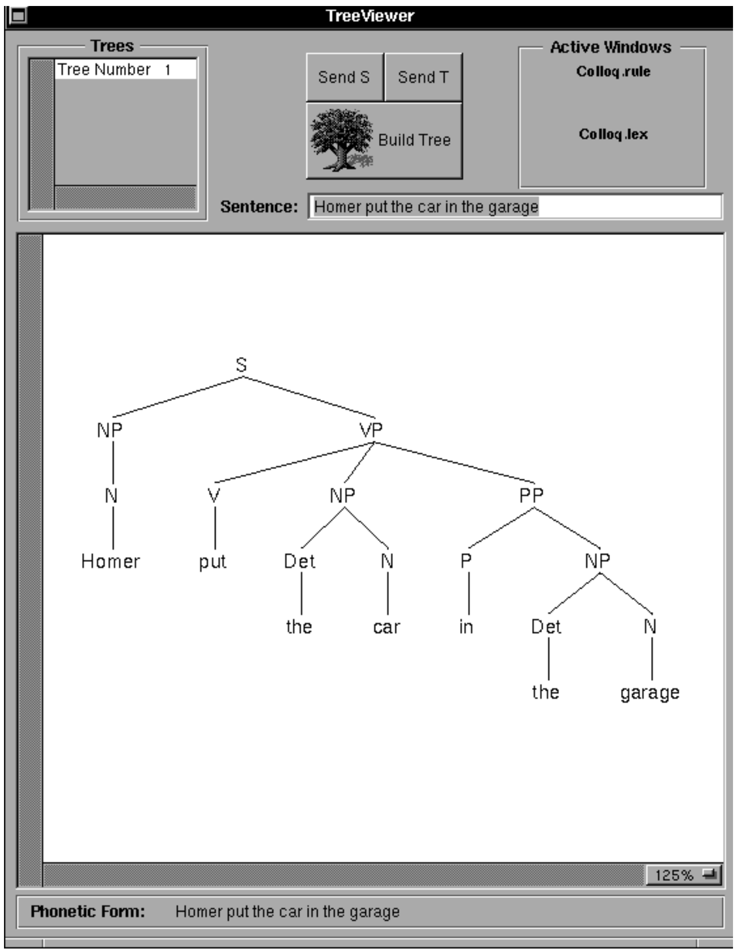
PS rules are entered in a window containing a rule template:



Lexicons are entered in a window containing a lexical item template:



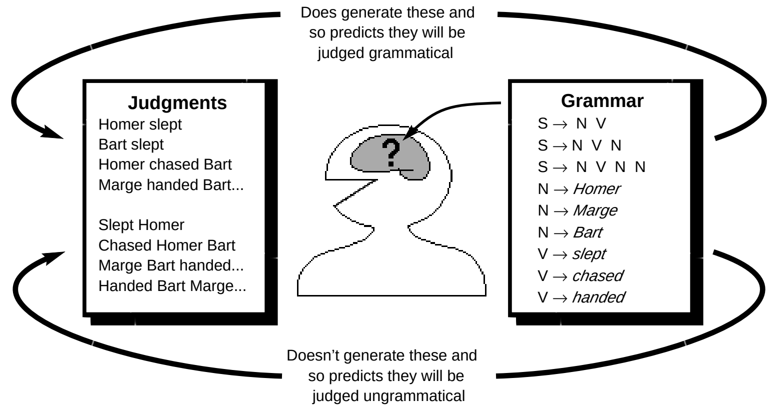
- Rules & lexicons are "loaded" into *Syntactica* (e.g., by clicking the Load button)
- An input sentence is entered, e.g.: *Homer put the car in the garage*.
- Clicking the Build Tree button results in a tree displayed in TreeViewer window



VI. *SYNTACTICA :* How Its Used in Our Course

A. Modeling Grammatical Competence

We want students to understand the idea of attributing knowledge of a formal system as a way for explaining competence and abilities. *Syntactica* provides a very concrete example of a formal system, and a potential model of speakers' grammatical competence:



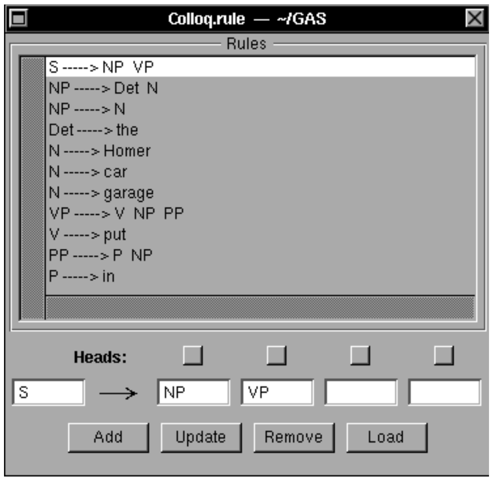
B. Constructing and Comparing Grammars

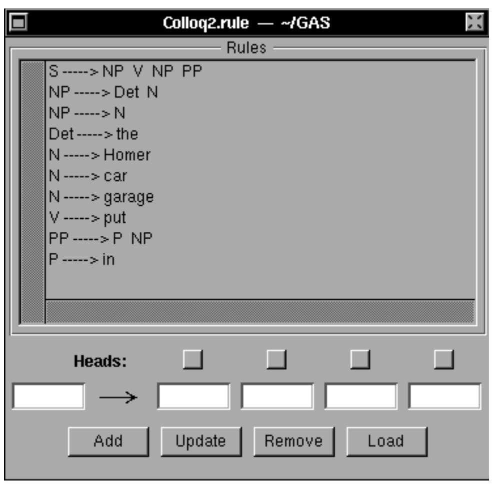
1. We want people to understand how one constructs a scientific theory, tests it, refines it, etc. We also want them to be able to separate a theory, the data it explains, and the predications that it makes. *Syntactica* provides a convenient workbench for building, testing, and refining grammars (represented in *Syntactica* as PS rule files or PS rule files + lexicon files). Its window structure also visually separates the parts of an explanation:

- the data (input sentences or target trees)
- the theory proposed to explain the data (what's in the grammar windows)
- the predictions of the theory (what appears in the TreeViewer window)

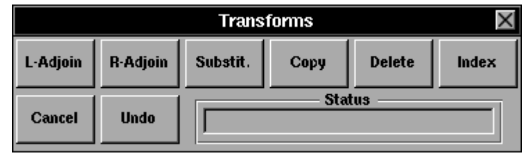
Example: A standard early exercise in *LIN 211* is for students to build rules for a preassigned set of sentences. Students work individually or in small groups. The mechanical aspects of the task quickly become clear: "get the computer to draw a tree" for each sentence. The nonmechanical aspects of the task also quickly become clear. Student results are easily compared by emailing their files around. Who is right (if anyone), and how do we tell??

2. We also want people to understand how one evaluates competing scientific theories. *Syntactica* provides a convenient workbench for comparing grammars. Competing grammar files can be quickly loaded and tested against a given input to check whether the input is accepted, and if so what structures are assigned.





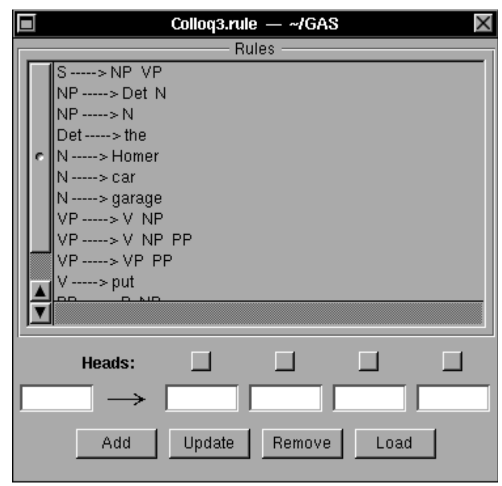
Familiar constituency tests (proform replacement (PR), ellipsis (E), movement (M) are easily applied in this context. With PR, we can show that adding a rule like VP → *did-too* works with the first rule set but not the second. With ellipsis and movement we can use the Transforms panel:



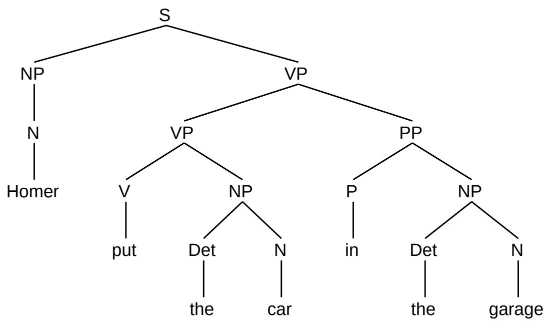
B. The Relation Between a Structure and the Properties of its Elements

The physical sciences give many examples of the constituent elements of a substance determining its structure and hence its behavior (e.g., carbon atoms in diamond; hydrogen & oxygen in water). We want people to understand the parallel rich relation between a syntactic structure and the properties of the words in it (selection & projection). *Syntactica* allows the user to specify certain lexical information and to control its propagation in the tree.

Example: In *LIN 211* students initially use only PS rules. At a certain point they meet rules like the following, which generating more than one structure for an input like *Homer put the car in the garage*.



Along with the tree displayed earlier on page 5, these rules <u>also</u> yield:



In exploring constituency, students discover reason to retain <u>both</u> rules responsible for this result:

VP $\rightarrow$ VP PP   VP $\rightarrow$ V NP PP

But they also learn that the verbs occurring in the two configurations are different. The lexicon and headedness are introduced as tools of capturing these relations. The Lexicon window (p.4) provides a template for stating properties of lexical items. The Rule window allows the head(s) in a rule to be explicitly declared.

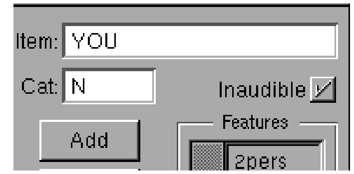
C. Covert Elements & Structure

We want students to understand when & why scientists propose "unobservables" in trying to explain phenomena, e.g., as a means for keeping the overall theory simple. *Syntactica* allows the user to postulate both unpronounced lexical items & unpronounced levels of structure.

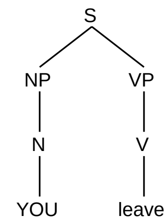
The PF display, located beneath the TreeViewer window, gives the surface string for a tree minus any elements that are declared or known by the system to be inaudible .

1. The inaudibility of individual lexical items can be declared by the user in the Lexicon window.

Example: In LIN 211 students meet imperative forms like *Leave!* A natural idea is that these are sentences with a unspoken 2nd pers. subject *YOU*. If YOU is entered in the lexicon as shown, where the box "Inaudible" is checked:



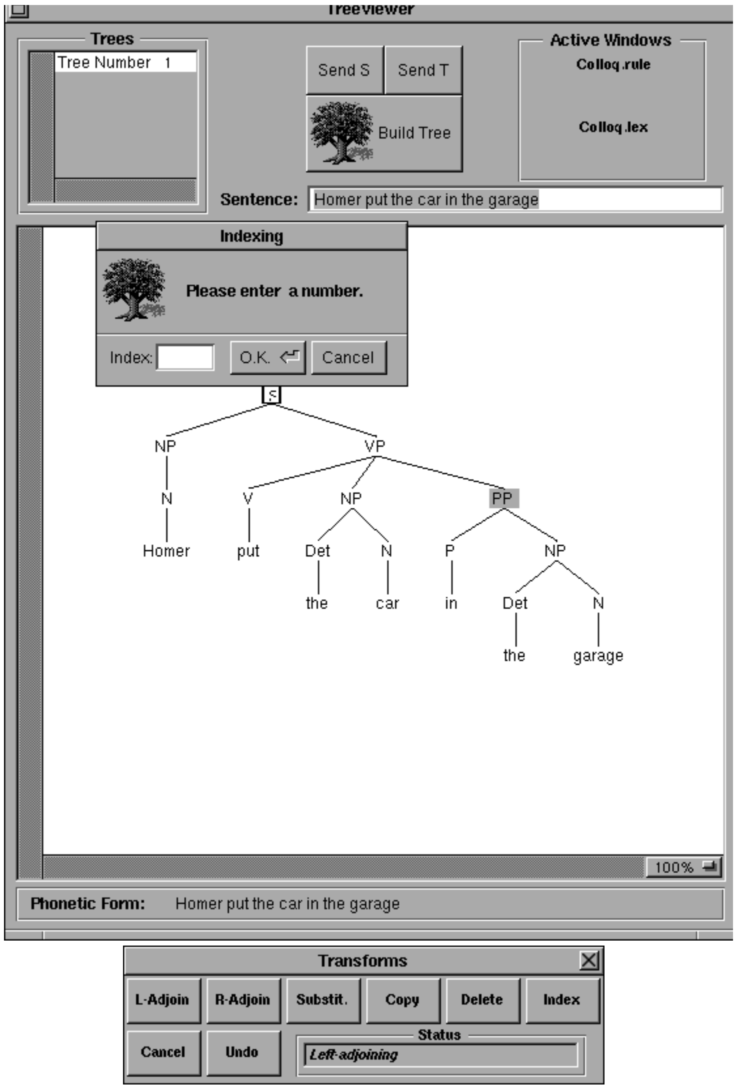
then the following tree gets the PF shown below it:

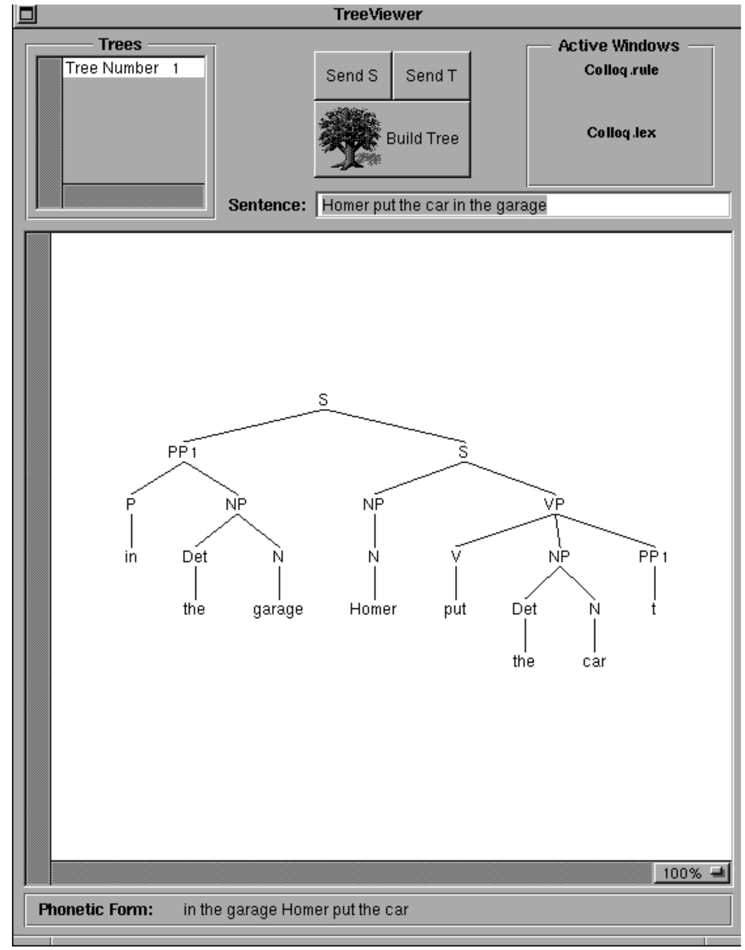


2. *Syntactica* provides a panel of elementary operations for transforming a tree by simple point and click operations. As students work with producing structures with successively more transforms, they come to see that accounting for the surface form of sentences requires appeal to a progressively more remote underlying form.

Example: The following two screen shots show *Syntactica* performing a left-adjunction of a PP to its containing S. The user (a) clicks the node to be moved, (b) clicks the relevant operation in the Transforms panel. (c) clicks on the node to be adjoined-to. An panel appears asking the user to supply an index. Once supplied, *Syntactica* performs the operation using the index given.

Notice that traces do not show up in the PF; the system knows them to be "silent".


TreeViewer
Trees
Tree Number 1
Send S
Send T
Build Tree
Active Windows
Colloq.rule
Colloq.lex
Sentence: Homer put the car in the garage
Indexing
Please enter a number.
Index:
O.K.
Cancel
S
NP
VP
N
V
NP
PP
Homer
put
Det
N
P
NP
the
car
in
Det
N
the
garage
100%
Phonetic Form: Homer put the car in the garage
Transforms
L-Adjoin
R-Adjoin
Substit.
Copy
Delete
Index
Cancel
Undo
Status
Left-adjoining


TreeViewer
Trees
Tree Number 1
Send S
Send T
Build Tree
Active Windows
Colloq.rule
Colloq.lex
Sentence: Homer put the car in the garage
S
PP 1
S
P
NP
NP
VP
in
Det
N
N
V
NP
PP 1
the
garage
Homer
put
Det
N
t
the
car
100%
Phonetic Form: in the garage Homer put the car