The bottle neck in building a practical natural language processing system is not those problems which have been often discussed in research papers, but in handling much more dirty, exceptional (for theoreticians, but we frequently encounter) expressions. This panel will focus on the problem which has been rarely written but has been argued informally among researchers who have tried to build a practical natural language processing system at least once.

Theory is important and valuable for the explanation and understanding, but is essentially the first order approximation of a target object. As for language, current theories are just for the basic part of the language structure. Real language usage is quite different from the basic language structure and a supposed mechanism of interpretation. Natural language processing system must cover real language usage as much as possible. The system model must be designed in such a way that it is clearly understandable by the support of a powerful linguistic theory, and still can accept varieties of exceptional linguistic phenomena which the theory is difficult to treat. How we can design such a system is a major problem in natural language processing, especially for machine translation between the languages of different linguistic families. We have to be concerned with both linguistic and non-linguistic world. While we have to study these difficult problems, we must not forget about the realizability of a useful system from the standpoint of engineering. I received valuable comments from Dr. Karen Jensen who cannot participate in our panel, and kindly offered me to use her comments freely in our panel. I want to cite her comments in the followings.

Why Computational Grammarians Can Be Skeptical About Existing Linguistic Theories

Karen Jensen
IBM TJ Watson Research Center
Yorktown Heights, NY10598, U.S.A

1. We need to deal with huge amounts of data (number of sentences, paragraphs, etc.). Existing linguistic theories (LTs) play with small amounts of data.
2. The data involve many (and messy) details. LTs are prematurely fond of simplicity. For example, punctuation is very important for processing real text, but LTs have nothing to say about it. (This is actually strange, since punctuation represents -- to some extent -- intonational contours, and these are certainly linguistically significant.)
3. There is no accepted criterion for when to abandon an LT; one can always modify theory to fit counterexamples. We have fairly clear criteria if a computational system cannot do its job in real time, then it fails.
4. We need to use complex attribute-value structures, which cannot be manipulated on paper or on a blackboard. "Trees" are only superficially involved. This means we are absolutely committed to computation. LTs have various degrees of commitment.
5. We are not interested in using the most constrained/restricted formalism. LTs generally are, because of supposed claims about language processing mechanisms.
6. We are interested in uniqueness as much as in generality. LTs usually are not.
7. We are more interested in coverage of the grammar than in completeness of the grammar. LTs generally pursue completeness.
8. We aim for "all," but not "only" the grammatical constructions of a natural language. Defining ungrammatical structures is, by and large, a futile task (Alexin Munaster-Ramer, Wlodzimierz Zadrozny).
9. Existing LTs give at best a high-level specification of the structure of natural language. Writing a computational grammar is like writing a real program given very abstract specs (Nelson Correia).
10. We are not skeptical of theory, just of existing theories.

At first blush, it seems unnecessary to conjure up any justification for this claim. Almost by definition, the proper business of a grammar should be grammaticality. However, it has been notoriously difficult to draw a line between "grammatical" sequences and "ungrammatical" sequences, for any natural human language. It may even be provably impossible to define precisely the notion of grammaticality for any language. Natural language deals with vague predicates, and might itself be called a vague predicat. This being true, it still seems worthwhile to aim at parsing ALL of the grammatical strings of a language, but parsing ONLY the grammatical strings becomes a dubious enterprise at best. Arguments for doing so reduce either to dogma, or to some general notion of propriety. Arguments against, however, are easy to come by. Leaving theoretical considerations aside for the moment, consider these pragmatic ones:

(a) The diachronic argument. The creativity of human use of language is great, and language systems are always changing. A construction that was once unacceptable becomes acceptable over time, and vice versa. Even if a grammar could
describe all and only the grammatical sequences today, the
same may not be true tomorrow. So there is, at best, only an
academic interest in only-grammatical structures.

(6) The practical argument. In the area of applied com-
putational linguistics, ill-formed input is a part of daily life,
and a working grammar has to handle it. By "handle it" we
mean not to grind to a halt, but figure out some kind of appro-
priate analysis and then comment, if possible, on whatever is
difficult or unusual. If real-life natural language processing is
going to exist, there must be some way to extract meaning
even from utterances that violate customary syntactic rules, that
are excessively long and complex, and that are not sentences
at all.

At IBM Research, we are developing a broad-coverage
parsing grammar for English, called the PLNLPEnglish Gram-
amar, or PEG. Its initial syntactic component works only with
ill-formed input -- lexical features for parts of speech, for
morphological structure, and for some valency classes. This
component tries to assign some reasonable structure to any
input string of English.

Even in its current beginning state, PEG has proved to
be of considerable usefulness for a rather wide variety of real-
world NLP tasks. Its main use so far has been as the parsing
component of CRITIQUE, a large-scale natural language text
processing system that identifies grammar and style errors in
English text (Hfilchra et al. 1982, Richardson and Braden-
Forden 1985). A prototype CRITIQUE system is now func-
tioning in three major application areas: business offices, a
polishing center, and universities.

Real-world natural language processing must deal with
huge amounts of data, which involves many, and messy, details.
For example, punctuation is very important in processing real
text, but current linguistic theories have nothing substantial
to say about punctuation. Nor have they anything substantial
to say about analysis structures for ellipsis, or for strings
that deviate in various degrees from the canonical order of the
language as it occurs. Here is the kind of natural lan-
guage input that CRITIQUE has to deal with. (All of the text
excerpts below are written EXACTLY as they were produced.)

First, a memo that was sent via electronic mail to multiple
users in the office environment:

(1) Over the course of the next couple of days the
accounting department will conducting inventory of
lids and offices here at XIX. Two are currently
working on the first floor, and working there way up.
If you are not in your office and do not plan to be
there within the next few days, please secure all con-
fidential mail and items you may have of confidential
nature. Because if you are not on these accounting
is going to go in and inventory your equipment.

The author of text (1) is a native speaker of American
English, who has a college education and is employed in a
position of some responsibility in a large business firm. Note
the following problems:

(a) "will conducting" should be "will conduct";
(b) "conducting inventory" should be "conducted
inventory";
(c) "I they" should be just "They";
(d) "working there way up" should be "working
their way up";
(e) "days please" lacks a space between the comma
and "please";
(f) "of confidential nature" would be better written
as "of a confidential nature";
(g) The last sentence is a fragment, not a com-
plete clause, although it is presented as if it were
a sentence.

Theoretically pure grammar would ever be able to an-
alyze text like this. It may be objected that "grammar" defines
the competence that makes it possible for us to identify mis-
takes (a - g), and that any working system is an embodiment of
a kind of performance, not competence. Very well; note then
that the role of "grammar" becomes that of a "COMMENTS" in
the analysis structure, NOT the definition of the structure
itself. This is exactly the point. It may be that we need a new
definition of the term "grammar."

Within the educational environment, the challenge for a
computational grammar is even stronger. Following are two
excerpts from essays by non-native English speakers. Text
(2) is an extreme example of the run-on style of writing; the
interesting "grammatical" question is what cue might be used
to divide this text into separate sentences:

(2) After the analysis of three graphs we can
make conclusion. From 1940 to 1980 the farm pop-
ulation and farms decrease but the average farm size
increase, this tendency shows America don't have
strong interest to work on the farms, as a result it is
impossible to increase the farms but when The people
who would like to work on farms expand their farm size by themselves or the aid of government;
some other agents want to invest capital in the "farming industry".

Text (3) shows interesting problems with the definite ar-
ticle (mass vs. count NPs) and with auxiliaries in VPs:

(3) So we know, now we can use the fewer peo-
ples to get the more food. Is the decreasing farmer
deduce on the graph? Is the farms going to de-
creasing in future? Does the average of farm size
will develop? No. No. No.

The problem of non-"grammaticality" is pervasive in real
language use. The question

(4) Who did you tell me that won?

supposedly poses an extraction problem -- in terms of Gov-
ernment Binding Theory, it violates the Empty Case Principle.
Yet it can be heard from the mouths of people who would oth-
erwise qualify as speakers of Standard English. The sentence

(5) He bought for ten shillings a ring.

supposedly violates an ordering constraint in English because
the prepositional phrase "for ten shillings" precedes the direct
object "a ring." However, as the direct object NP becomes
heavier and heavier, the sentence sounds better and better:

(5') He bought for ten shillings a ring that de-
lighted the woman who had previously been pro-
duced to by millionaires.

To move "for ten shillings" to a position following the direct
object in (5') would be extremely awkward. In this case, it is
to better interpret the "grammatical" ordering rule as a stylistic
comment, "The construction

(6) Himself father came.

violates theoretical restrictions on anaphora, or Binding; but
it is fine if read with an Irish flavor. And the alternative of
having a completely separate grammar for Irish English is not
appealing. The sentence

(7) She be happy.

is censured because the main verb is not tensed; but (7) is valid
Non-standard Black English. And so on. Many theoretically
proscribed sequences exist and flourish as stylistic or social
variants. To ignore them, and to pursue the Holy Grail of a
grammar that describes "all and only" the grammatical strings
of a language, would be to defeat the enterprise of broad-
coverage computational parsing.

Furthermore, it is not necessary to enforce all of the
supposedly "grammatical" restrictions within a computational
analysis grammar that actually deals with quantities of real
text, in real time. Our experience with PEG, in the CRITIQUE
application, proves this. PEG produces appropriate parses for
(4) - (7). Then a Style component can comment on the parses,
calling attention to whatever problems or variations exist. We
do not currently handle all of the difficulties posed by (1) -
(3), but we do handle some of them. For those grammatical
restrictions that have to be enforced within the syntactic gram-
mar (such as number agreement), we have a two-pass error
detection and correction strategy. For massive problems like
the run-ons in (2), we use the technique of the "fitted pane,"
which tries to identify sensible chunks of text and present them
in some reasonable framework.

Since it is neither desirable nor necessary for a compu-
tational grammar to define "all and only" the "grammatical"
sequences of a language, and since working computational
grammars are the most comprehensive description that we
can come up with, right now, for natural languages, we suggest
that the goal of real-world grammatical analysis be re-defined:
a grammar should try to describe "all," but not "only," the
grammatical strings of a language.