Grammar Engineering

May 2, 2005

Clausal semantics, precision grammars and corpora

For Wednesday

- Begin exploring the syntactic reflexes of illocutionary force. How are propositions expressed? Questions? Commands?
- Find examples of verbs that embed clausal (propositional) complements, e.g., *know*, *believe*, *say*

Overview

- Clausal semantics

  Clausal semantics in Ginzburg & Sag 2000

  Messages in MRS

  Messages in the Matrix

- Beauty and the Beast

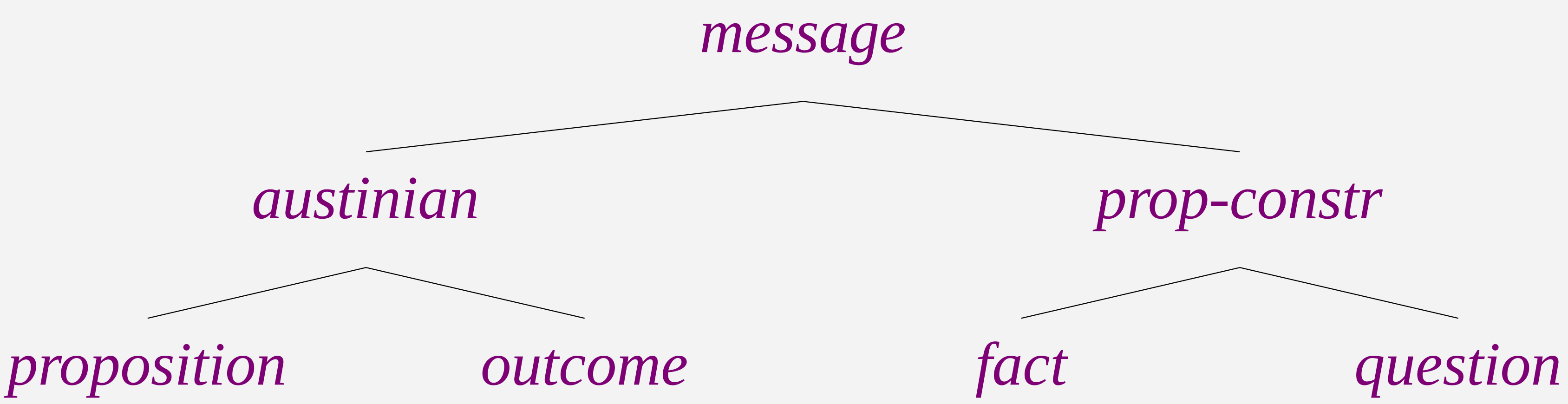
  Theoretical motivation

  Methodology

  Results

Clausal semantics: Messages

- “*Message* is the semantic type that is the most basic to communication—its (maximal) subtypes constitute the descriptive contents of basic illocutionary acts such as assertion, querying, commanding, exclaiming and the like.” (Ginzburg & Sag 2000:121)
- Partial hierarchy under *message*:



message

austinian *prop-constr*

proposition *outcome* *fact* *question*

Clausal semantics in recursive representation (1/2)

$$
\left[\begin{array}{ll}
\text{SYNSEM} & \left[\begin{array}{ll}
\text{LOC.CONT} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{proposition}}\\
\text{SIT} & s_0\\
\text{SOA} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{soa}}\\
\text{QUANTS} & \langle\,\rangle\\
\text{NUCL} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{leave-rel}}\\ \text{LEAVER} & i\end{array}\right]
\end{array}\right]
\end{array}\right]\\
\text{BCKGRD} & \left\{\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{fact}}\\
\text{PROP} & \left[\begin{array}{ll}
\text{SIT} & s_1\\
\text{SOA} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{r-soa}}\\
\text{QUANTS} & \langle\,\rangle\\
\text{NUCL} & \left[\begin{array}{ll}\multicolumn{2}{l}{\textit{name-rel}}\\ \text{NAMED} & i\end{array}\right]
\end{array}\right]
\end{array}\right]
\end{array}\right]\right\}
\end{array}\right]
\end{array}\right]
$$

Clausal semantics in recursive representation (2/2)

$$\begin{bmatrix} \text{SYNSEM} & \begin{bmatrix} \text{LOC.CONT} & \begin{bmatrix} \textit{question} \\ \text{PARAMS} & \{\} \\ \text{PROP} & \begin{bmatrix} \textit{proposition} \\ \text{SIT} & s_0 \\ \text{SOA} & \begin{bmatrix} \textit{soa} \\ \text{QUANTS} & \langle\ \rangle \\ \text{NUCL} & \begin{bmatrix} \textit{leave-rel} \\ \text{LEAVER} & i \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\ \text{BCKGRD} & \left\{ \begin{bmatrix} \textit{fact} \\ \text{PROP} & \begin{bmatrix} \text{SIT} & s_1 \\ & \begin{bmatrix} \textit{r-soa} \\ \text{QUANTS} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Messages in MRS

$$\begin{bmatrix}
\text{LTOP} & h1 \\
\text{INDEX} & e2 \\
\text{RELS} & \left\langle \begin{array}{l}
\begin{bmatrix} \textbf{int\_m\_rel} & \\ \text{LBL} & h1 \\ \text{MARG} & h5 \end{bmatrix},
\begin{bmatrix} \textbf{prpstn\_m\_rel} & \\ \text{LBL} & h5 \\ \text{MARG} & h6 \end{bmatrix},
\begin{bmatrix} \textbf{proper\_q\_rel} & \\ \text{LBL} & h9 \\ \text{ARG0} & x11 \\ \text{RSTR} & h10 \\ \text{BODY} & h12 \end{bmatrix}, \\
\begin{bmatrix} \textbf{named\_rel} & \\ \text{LBL} & h13 \\ \text{ARG0} & x11 \\ \text{CARG} & \text{“}kim\text{”} \end{bmatrix},
\begin{bmatrix} \textbf{\_leave\_v\_1\_rel} & \\ \text{LBL} & h14 \\ \text{ARG0} & e2 \\ \text{ARG1} & x11 \\ \text{ARG2} & i15 \end{bmatrix}
\end{array} \right\rangle \\
\text{HCONS} & \langle\, h6\ qeq\ h14,\ h10\ qeq\ h13 \,\rangle
\end{bmatrix}$$

Messages in the Matrix (1/3)

```
mrs := mrs-min &
  [ HOOK hook,
    RELS diff-list,
    HCONS diff-list,
    MSG basic_message ].
```

- Messages appear on the RELS list, but also have a dedicated pointer in CONT.MSG.
- We can use CONT.MSG to ensure that only clauses are accepted as stand alone utterances.

Messages in the Matrix (2/3)

```
basic_message := relation.
message := basic_message &
  [ PRED message_m_rel,
    MARG handle ].

no-msg := basic_message.
```

Messages in the Matrix (3/3)

```
message_m_rel := predsort.
command_m_rel := message_m_rel.
prop-or-ques_m_rel := message_m_rel.
    ;for COMPS of e.g. ’know’
proposition_m_rel := prop-or-ques_m_rel.
abstr-ques_m_rel := prop-or-ques_m_rel.
question_m_rel := abstr-ques_m_rel.
ne_m_rel := abstr-ques_m_rel.
```

Introducing Messages

- Ginzburg & Sag propose cross-classifying the phrase types with clause types, so that, e.g., *decl-hd-su-cl* pairs a (VP) head with its subject and introduces the *proposition*.
- Working at a cross-linguistic level, we find it more convenient to introduce messages via non-branching constructions (S $\longrightarrow$ S, etc).

Clausal semantics: Summary

- Clauses are distinguished from other constituents in that they carry illocutionary force (even when embedded).
- The illocutionary force is modeled by *message*s, which embed descriptions of states of affairs.
- Non-clausal fragments need to be analyzed in terms of additional non-branching constructions which introduce appropriate *message*s.

Beauty and the Beast: Overview

- Theoretical motivation
- Methodology
- Results

Theoretical motivation (1/2)

- Corpora as a sole source of data are inadequate because:

  They are limited in size and may not reflect the full range of grammatical constructions.

  They contain errors due to processing and reflect other extragrammatical factors.

  They can only provide positive (attested) examples, and not contrasting negative ones.

Theoretical motivation (2/2)

- Intuitions as data are inadequate because:

  Grammaticality is neither homogeneous nor categorical.

  Grammaticality judgments are frequently formed in unnatural context vacuums.

  Social/cultural biases color judgments.

  Relying solely on intuitions limits linguists to only the data they have the imagination to think up.

Combine the two types of data for better results!

- Grammar engineering provides a sophisticated way of doing so.
- Precision grammars encode a sharp notion of grammaticality.
- Use grammar as a representation of intuitions.
- Use the corpus as a source of further data to explore.
- Process the corpus with the grammar...

Methodology

- Randomly select 20,000 strings (‘sentence tokens’) from the BNC written component.
- Strip punctuation, tag for part-of-speech, tokenize proper names and number expressions, normalize to American spelling.
- Select those strings with full lexical span (32%)
- Process these strings with the ERG to isolate those that can’t presently be parsed
- Propose paraphrases of the unparseable strings until the ERG is able to parse one

Results: Grammar coverage

- 57% of strings parsed
- 83% of parsed strings assigned a correct (preferred) parse, perhaps among others
- Average ambiguity for 10-20 word strings: 64 parses

Results: Causes of parse failure

Cause of parse failure	Frequency	Category
Missing lexical entry	41%	grammar
Missing construction	39%	grammar
Fragment	4%	grammar
Preprocessor error	4%	neither
Parser resource limitations	4%	neither
Ungrammatical string	6%	corpus
Extragrammatical string	2%	corpus

Missing lexical entries (1/2)

- Incomplete categorization of existing lexical items

  table as a verb

  ‘universal grinder’

- Syntactically-marked MWEs

  take off, verb + *up*

  off screen, at arm’s length

  High frequency: verb-particles constitute 1.6% of BNC word tokens

Missing lexical entries (2/2)

- Drawbacks to introspection alone: subtle gaps like transitive *suffer*

- Drawbacks to corpus data alone: *tell* in the ‘discover’ sense:

  @Not sure how you can tell.

  Can/could you tell?

  Are you able to tell?

  *They might/ought to tell.

  How might you tell?

  *How ought they to tell?

Missing constructions

- @*However pissed off* we might get from time to time...
- @He’s a good player and a *hell of a* nice guy, too.
- @The price of train tickets can vary from *the reasonable* to *the ridiculous*.
- @This sort of response was also noted in the sample task for *criterion 2*.

Extragrammatical strings

- Prime example: Structural markup:

  @There are five of these general arrest conditions: (a) the name of. . .

  @(I) The Mrs Simpson could never be Queen.

  @(I) rarely took notes during the thousands of informal conversational interviews.

Grammar and corpus summary

- Methodology goes beyond merely using the corpus for inspiration.

  encoding intuitions in the grammar

  use the grammar to process the corpus, twice: filter out ‘easy’ cases, investigate where in a string the problems are

- Provides detailed feedback to grammar developers
- Turns up previously unnoted constructions, which might be too low frequency to be found otherwise

Overview

- Clausal semantics
  - Clausal semantics in Ginzburg & Sag 2000
  - Messages in MRS
  - Messages in the Matrix
- Beauty and the Beast
  - Theoretical motivation
  - Methodology
  - Results