Outline

◊ Communication
◊ Grammar
◊ Syntactic analysis
◊ Problems
Communication

“Classical” view (pre-1953):
language consists of sentences that are true/false (cf. logic)

“Modern” view (post-1953):
language is a form of action

Wittgenstein (1953) Philosophical Investigations
Austin (1962) How to Do Things with Words
Searle (1969) Speech Acts

Why?

To change the actions of other agents
Speech acts achieve the speaker’s goals:

- **Inform**
  
  “There’s a pit in front of you”

- **Query**
  
  “Can you see the gold?”

- **Command**
  
  “Pick it up”

- **Promise**
  
  “I’ll share the gold with you”

- **Acknowledge**
  
  “OK”

Speech act planning requires knowledge of:

- Situation
- Semantic and syntactic conventions
- Hearer’s goals, knowledge base, and rationality
Stages in communication (informing)

**Intention**  
S wants to inform H that \( P \)

**Generation**  
S selects words \( W \) to express \( P \) in context \( C \)

**Synthesis**  
S utters words \( W \)

**Perception**  
H perceives \( W' \) in context \( C' \)

**Analysis**  
H infers possible meanings \( P_1, \ldots P_n \)

**Disambiguation**  
H infers intended meaning \( P_i \)

**Incorporation**  
H incorporates \( P_i \) into KB

How could this go wrong?

- Insincerity (S doesn’t believe \( P \))
- Ambiguous utterance
- Differing understanding of current context (\( C \neq C' \))
Vervet monkeys, antelopes etc. use isolated symbols for sentences
   ⇒ restricted set of communicable propositions, no generative capacity
(Chomsky (1957): *Syntactic Structures*)

Grammar specifies the compositional structure of complex messages
   e.g., speech (linear), text (linear), music (two-dimensional)

A formal language is a set of strings of terminal symbols

Each string in the language can be analyzed/generated by the grammar

The grammar is a set of rewrite rules, e.g.,

\[ S \rightarrow NP \ VP \]
\[ Article \rightarrow \text{the} \mid a \mid an \mid \ldots \]

Here \( S \) is the sentence symbol, \( NP \) and \( VP \) are nonterminals
Grammar types

Regular: nonterminal $\rightarrow$ terminal[nonterminal]

\[ S \rightarrow aS \]
\[ S \rightarrow \Lambda \]

Context-free: nonterminal $\rightarrow$ anything

\[ S \rightarrow aSb \]

Context-sensitive: more nonterminals on right-hand side

\[ ASB \rightarrow AAaBB \]

Recursively enumerable: no constraints

Natural languages probably context-free, parsable in real time!
Wumpus lexicon

Noun $\rightarrow$ stench | breeze | glitter | nothing
   | wumpus | pit | pits | gold | east | ...
Verb $\rightarrow$ is | see | smell | shoot | feel | stinks
   | go | grab | carry | kill | turn | ...
Adjective $\rightarrow$ right | left | east | south | back | smelly | ...
Adverb $\rightarrow$ here | there | nearby | ahead
   | right | left | east | south | back | ...
Pronoun $\rightarrow$ me | you | I | it | S/HE | Y’ALL...
Name $\rightarrow$ John | Mary | Boston | UCB | PAJC | ...
Article $\rightarrow$ the | a | an | ...
Preposition $\rightarrow$ to | in | on | near | ...
Conjunction $\rightarrow$ and | or | but | ...
Digit $\rightarrow$ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Divided into closed and open classes
Wumpus grammar

\[
S \rightarrow NP \ VP \quad \text{l + feel a breeze}
\]

\[
S \rightarrow S \ Conjunction \ S \quad \text{l feel a breeze + and + l smell a wumpus}
\]

\[
NP \rightarrow \text{Pronoun} \quad \text{l}
\]

\[
NP \rightarrow \text{Noun} \quad \text{pits}
\]

\[
NP \rightarrow \text{Article Noun} \quad \text{the + wumpus}
\]

\[
NP \rightarrow \text{Digit Digit} \quad \text{3 4}
\]

\[
NP \rightarrow NP \ PP \quad \text{the wumpus + to the east}
\]

\[
NP \rightarrow NP \ RelClause \quad \text{the wumpus + that is smelly}
\]

\[
VP \rightarrow \text{Verb} \quad \text{stinks}
\]

\[
VP \rightarrow VP \ NP \quad \text{feel + a breeze}
\]

\[
VP \rightarrow VP \ Adjective \quad \text{is + smelly}
\]

\[
VP \rightarrow VP \ PP \quad \text{turn + to the east}
\]

\[
VP \rightarrow VP \ Adverb \quad \text{go + ahead}
\]

\[
PP \rightarrow \text{Preposition} \ NP \quad \text{to + the east}
\]

\[
RelClause \rightarrow \text{that} \ VP \quad \text{that + is smelly}
\]
Grammaticality judgements

Formal language $L_1$ may differ from natural language $L_2$

Adjusting $L_1$ to agree with $L_2$ is a learning problem!

* the gold grab the wumpus
* I smell the wumpus the gold
  I give the wumpus the gold
* I donate the wumpus the gold

Intersubjective agreement somewhat reliable, independent of semantics!
Real grammars 10–500 pages, insufficient even for “proper” English
Parse trees

Exhibit the grammatical structure of a sentence

I shoot the wumpus

Diagram:
- S
  - NP: Pronoun - I
  - VP: Verb - shoot
  - NP: Article - the
  - NP: Noun - wumpus
Most view syntactic structure as an essential step towards meaning; 
“Mary hit John” $\neq$ “John hit Mary”

“And since I was not informed—as a matter of fact, since I did not know that there were excess funds until we, ourselves, in that checkup after the whole thing blew up, and that was, if you’ll remember, that was the incident in which the attorney general came to me and told me that he had seen a memo that indicated that there were no more funds.”

“Wouldn’t the sentence ’I want to put a hyphen between the words Fish and And and And and And and Chips in my Fish-And-Chips sign’ have been clearer if quotation marks had been placed before Fish, and between Fish and and, and and and And, and And and and, and and and And, and And and and, and and and and Chips, as well as after Chips?”
Context-free parsing

Bottom-up parsing works by replacing any substring that matches RHS of a rule with the rule’s LHS.

Efficient algorithms (e.g., chart parsing, Section 22.3) \( O(n^3) \) for context-free, run at several thousand words/sec for real grammars.

Chart Parsing: Store parses of substrings in a chart so we do not have to reanalyze it later.

- ”Have the students in section 2 of Computer Science 101 take the exam.”
- ”Have the students in section 2 of Computer Science 101 taken the exam?”

Context-free parsing \( \equiv \) Boolean matrix multiplication (Lee, 2002)

\( \Rightarrow \) unlikely to find faster practical algorithms.
Logical grammars

BNF notation for grammars too restrictive:
- difficult to add “side conditions” (number agreement, etc.)
- difficult to connect syntax to semantics

Idea: express grammar rules as logic

\[ X \rightarrow YZ \quad \text{becomes} \quad Y(s_1) \land Z(s_2) \Rightarrow X(Append(s_1, s_2)) \]
\[ X \rightarrow \text{word} \quad \text{becomes} \quad X([“\text{word}”]) \]
\[ X \rightarrow Y \mid Z \quad \text{becomes} \quad Y(s) \Rightarrow X(s) \quad Z(s) \Rightarrow X(s) \]

Here, \( X(s) \) means that string \( s \) can be interpreted as an \( X \)
Logical grammars contd.

Now it’s easy to augment the rules

\[
NP(s_1) \land EatsBreakfast(\text{Ref}(s_1)) \land VP(s_2)
\]
\[
\Rightarrow NP(\text{Append}(s_1, ["who"], s_2))
\]

\[
NP(s_1) \land \text{Number}(s_1, n) \land VP(s_2) \land \text{Number}(s_2, n)
\]
\[
\Rightarrow S(\text{Append}(s_1, s_2))
\]

Parsing is reduced to logical inference:
\[
\text{Ask}(KB, S(["I" "am" "a" "wumpus"]))
\]
(Can add extra arguments to return the parse structure, semantics)

Generation simply requires a query with uninstantiated variables:
\[
\text{Ask}(KB, S(x))
\]

If we add arguments to nonterminals to construct sentence semantics, NLP generation can be done from a given logical sentence:
\[
\text{Ask}(KB, S(x, \text{At}(\text{Robot}, [1, 1])))
\]