Introduction to NLP

Semantics
Semantics

• What is the meaning of: \((5+2)*(4+3)\)?
• Parse tree

```
           E
          /|
         E F E
        /|
       E F E
      /|
     E F E
    /|
   N + N
  /|
 N + N
 /|
5 2 4 3
```
Semantics

• What if we had \((5+2)*(4+z)\)?
What about (English) sentences?

- Every human is mortal.
- ??
Representing Meaning

• **Goal**
  – Capturing the meaning of linguistic utterances using formal notation

• **Linguistic meaning**
  – “It is 8 pm”

• **Pragmatic meaning**
  – “It is time to leave”

• **Semantic analysis:**
  – Assign each word a meaning
  – Combine the meanings of words into sentences

• *I bought a book:*

  \[
  \exists \ x,y: \text{Buying}(x) \land \text{Buyer}(\text{speaker},x) \land \text{BoughtItem}(y,x) \land \text{Book}(y) \\
  \text{Buying (Buyer=\text{speaker}, BoughtItem=\text{book})}
  \]
Introduction to NLP

Representing and Understanding Meaning
Understanding Meaning

• If an agent hears a sentence and can act accordingly, the agent is said to understand it.

• Example
  – Leave the book on the table.

• Understanding may involve inference
  – Maybe the book is wrapped in paper?

• And pragmatics
  – Which book? Which table?

• So, understanding may involve a procedure.
Properties

• Verifiability
  – Can a statement be verified against a knowledge base (KB)
  – Example: does my cat Martin have whiskers?

• Unambiguousness
  – Give me the book
  – Which book?

• Canonical form

• Expressiveness
  – Can the formalism express temporal relations, beliefs, ...?
  – Is it domain-independent?

• Inference
Representing Meaning

• One traditional approach
  – use logic representations, e.g., FOL (first order logic)

• Inference
  – One can then use theorem proving (inference) to determine whether one statement entails another
Syntax of Propositional Logic

- The simplest type of logic
- The proposition symbols $P_1, P_2, \ldots$ are sentences
  - If $S$ is a sentence, $\neg S$ is a sentence (negation)
  - If $S_1$ and $S_2$ are sentences, $S_1 \land S_2$ is a sentence (conjunction)
  - If $S_1$ and $S_2$ are sentences, $S_1 \lor S_2$ is a sentence (disjunction)
  - If $S_1$ and $S_2$ are sentences, $S_1 \Rightarrow S_2$ is a sentence (implication)
  - If $S_1$ and $S_2$ are sentences, $S_1 \Leftrightarrow S_2$ is a sentence (biconditional)
Propositional Logic in Backus Naur Form

- Sentence $\rightarrow$ AtomicSentence | ComplexSentence
- AtomicSentence $\rightarrow$ True | False | S | T | U ...
- ComplexSentence $\rightarrow$ (Sentence)
  | $\neg$Sentence
  | Sentence $\land$ Sentence
  | Sentence $\lor$ Sentence
  | Sentence $\Rightarrow$ Sentence
  | Sentence $\Leftrightarrow$ Sentence
Operator Precedence

¬ (highest)
∧
∨
⇒
⇔ (lowest)
Translating Propositions to English

• A = Today is a holiday.
• B = We are going to the park.

• A \Rightarrow B
• A \land \neg B
• \neg A \Rightarrow \neg B
• \neg B \Rightarrow \neg A
• B \Rightarrow A
Translating Propositions to English

- \( A = \) Today is a holiday.
- \( B = \) We are going to the park.

- \( A \Rightarrow B \)
  - If today is a holiday, we are going to the park.
- \( A \land \neg B \)
  - Today is a holiday, and we are not going to the park.
- \( \neg A \Rightarrow \neg B \)
  - If today is not a holiday, then we are not going to the park.
- \( \neg B \Rightarrow \neg A \)
  - If we are not going to the park, then today is not a holiday.
- \( B \Rightarrow A \)
  - If we are going to the park, then today is a holiday.
Semantics of Propositional Logic

• \( \neg S \) is true iff \( S \) is false
• \( S_1 \land S_2 \) is true iff \( S_1 \) is true and \( S_2 \) is true
• \( S_1 \lor S_2 \) is true iff \( S_1 \) is true or \( S_2 \) is true
• \( S_1 \Rightarrow S_2 \) is true iff \( S_1 \) is false or \( S_2 \) is true
  i.e. \( S_1 \Rightarrow S_2 \) is false iff \( S_1 \) is true and \( S_2 \) is false
• \( S_1 \Leftrightarrow S_2 \) is true iff \( S_1 \Leftrightarrow S_2 \) is true and \( S_2 \Leftrightarrow S_1 \) is true
• Recursively, one can compute the truth value of longer formulas
## Connectives

<table>
<thead>
<tr>
<th>P</th>
<th>Q</th>
<th>¬P</th>
<th>P∧Q</th>
<th>P∨Q</th>
<th>P⇒Q</th>
<th>P⇔Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
Logical Equivalence

\[
\begin{align*}
(\alpha \land \beta) & \equiv (\beta \land \alpha) \quad \text{commutativity of } \land \\
(\alpha \lor \beta) & \equiv (\beta \lor \alpha) \quad \text{commutativity of } \lor \\
((\alpha \land \beta) \land \gamma) & \equiv (\alpha \land (\beta \land \gamma)) \quad \text{associativity of } \land \\
((\alpha \lor \beta) \lor \gamma) & \equiv (\alpha \lor (\beta \lor \gamma)) \quad \text{associativity of } \lor \\
\neg(\neg \alpha) & \equiv \alpha \quad \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) & \equiv (\neg \beta \Rightarrow \neg \alpha) \quad \text{contraposition} \\
(\alpha \Rightarrow \beta) & \equiv (\neg \alpha \lor \beta) \quad \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) & \equiv ((\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)) \quad \text{biconditional elimination} \\
\neg(\alpha \land \beta) & \equiv (\neg \alpha \lor \neg \beta) \quad \text{de Morgan} \\
\neg(\alpha \lor \beta) & \equiv (\neg \alpha \land \neg \beta) \quad \text{de Morgan} \\
(\alpha \land (\beta \lor \gamma)) & \equiv ((\alpha \land \beta) \lor (\alpha \land \gamma)) \quad \text{distributivity of } \land \text{ over } \lor \\
(\alpha \lor (\beta \land \gamma)) & \equiv ((\alpha \lor \beta) \land (\alpha \lor \gamma)) \quad \text{distributivity of } \lor \text{ over } \land
\end{align*}
\]
NLP