Knowledge Based Agents

Knowledge base = set of sentences in a formal language

Declarative approach to building an agent (or other system):
TELL it what it needs to know

Then it can ASK itself what to do—answers should follow from the KB

Agents can be viewed at the knowledge level
i.e., what they know, regardless of how implemented

Or at the implementation level
i.e., data structures in KB and algorithms that manipulate them
Knowledge Base Agent Wrapper

function KB-AGENT (percept) returns an action
    static: KB, a knowledge base
            t, a counter, initially 0, indicating time
    TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
    action <- ASK(KB, MAKE-ACTION-QUERY(t))
    TELL(KB, MAKE-ACTION-SENTENCE(action, t))
    t <- t + 1
    return action

The agent must be able to:
Represent states, actions, etc.
Incorporate new percepts
Update internal representations of the world
Deduce hidden properties of the world
Deduce appropriate actions
Logics at High Level

Logics are formal languages for representing information such that conclusions can be drawn.

Syntax defines the sentences in the language.

Semantics define the “meaning” of sentences; i.e., define truth of a sentence in a world.

E.g., the language of arithmetic.

$x + 2 \geq y$ is a sentence; $x2 + y >$ is not a sentence.

$x + 2 \geq y$ is true iff the number $x + 2$ is no less than the number $y$.

$x + 2 \geq y$ is true in a world where $x = 7, y = 1$.

$x + 2 \geq y$ is false in a world where $x = 0, y = 6$. 

Logic Types

Logics are characterized by what they commit to as “primitives”


Epistemological commitment: what states of knowledge?

<table>
<thead>
<tr>
<th>Language</th>
<th>Ontological Commitment</th>
<th>Epistemological Commitment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Propositional logic</td>
<td>facts</td>
<td>true/false/unknown</td>
</tr>
<tr>
<td>First-order logic</td>
<td>facts, objects, relations</td>
<td>true/false/unknown</td>
</tr>
<tr>
<td>Temporal logic</td>
<td>facts, objects, relations, times</td>
<td>true/false/unknown</td>
</tr>
<tr>
<td>Probability theory</td>
<td>facts</td>
<td>degree of belief 0...1</td>
</tr>
<tr>
<td>Fuzzy logic</td>
<td>degree of truth</td>
<td>degree of belief 0...1</td>
</tr>
</tbody>
</table>
Notion of a Model

Logicians typically think in terms of models, which are formally structured worlds with respect to which truth can be evaluated.

We say \( m \) is a model of a sentence \( \alpha \) if \( \alpha \) is true in \( m \).

\( M(\alpha) \) is the set of all models of \( \alpha \).

Then \( KB \models \alpha \) if and only if \( M(KB) \subseteq M(\alpha) \).

E.g. \( KB = \) Giants won and Reds won
\[ \alpha = \text{Giants won} \]
Inference

$KB \vdash_i \alpha = \text{sentence } \alpha \text{ can be derived from } KB \text{ by procedure } i$

**Soundness:** $i$ is sound if
whenever $KB \vdash_i \alpha$, it is also true that $KB \models \alpha$

**Completeness:** $i$ is complete if
whenever $KB \models \alpha$, it is also true that $KB \vdash_i \alpha$
Propositional Logic: Syntax

Propositional logic is the simplest logic—illustrates basic ideas

The proposition symbols $P_1$, $P_2$ etc are sentences

If $S$ is a sentence, $\neg S$ is a sentence

If $S_1$ and $S_2$ is a sentence, $S_1 \land S_2$ is a sentence

If $S_1$ and $S_2$ is a sentence, $S_1 \lor S_2$ is a sentence

If $S_1$ and $S_2$ is a sentence, $S_1 \Rightarrow S_2$ is a sentence

If $S_1$ and $S_2$ is a sentence, $S_1 \Leftrightarrow S_2$ is a sentence
Propositional Logic: Semantics

Each model specifies true/false for each proposition symbol

E.g. \( A \quad B \quad C \)
\[ \text{True} \quad \text{True} \quad \text{False} \]

Rules for evaluating truth with respect to a model \( m \):

\[ \neg S \quad \text{is true iff} \quad S \quad \text{is false} \]
\[ S_1 \land S_2 \quad \text{is true iff} \quad S_1 \quad \text{is true and} \quad S_2 \quad \text{is true} \]
\[ S_1 \lor S_2 \quad \text{is true iff} \quad S_1 \quad \text{is true or} \quad S_2 \quad \text{is true} \]
\[ S_1 \Rightarrow S_2 \quad \text{is true iff} \quad S_1 \quad \text{is false or} \quad S_2 \quad \text{is true} \]
\[ \text{i.e., is false iff} \quad S_1 \quad \text{is true and} \quad S_2 \quad \text{is false} \]
\[ S_1 \Leftrightarrow S_2 \quad \text{is true iff} \quad S_1 \Rightarrow S_2 \quad \text{is true and} \quad S_2 \Rightarrow S_1 \quad \text{is true} \]
Inference via Enumeration

Let \( \alpha = A \lor B \) and \( KB = (A \lor C) \land (B \lor \neg C) \)

Is it the case that \( KB \models \alpha \)?
Check all possible models—\( \alpha \) must be true wherever \( KB \) is true

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>A \lor C</td>
<td>B \lor \neg C</td>
<td>KB</td>
<td>\alpha</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>----------</td>
<td>----------------</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>True</td>
<td>False</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>False</td>
<td>False</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CSI 535 - Introduction to A.I. Lectures 17