Artificial Intelligence

Topic 11

Logical Agents

Reading: Russell and Norvig, Chapter 7, Sections 1–4

Outline

- Knowledge-based agents
- Wumpus world
- Logics
- Propositional (Boolean) logic
  - syntax
  - semantics
- Models
- Entailment
- Equivalence, validity, satisfiability

Knowledge bases

<table>
<thead>
<tr>
<th>Inference engine</th>
<th>domain-independent algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge base</td>
<td>domain-specific content</td>
</tr>
</tbody>
</table>

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

A simple knowledge-based agent

```
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())

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
Wumpus World PEAS description

Performance measure
- gold +1000, death -1000
- -1 per step, -10 for using the arrow

Environment
- Squares adjacent to wumpus are smelly
- Squares adjacent to pit are breezy
- Glitter iff gold is in the same square
- Shooting kills wumpus if you are facing it
- Shooting uses up the only arrow
- Grabbing picks up gold if in same square
- Releasing drops the gold in same square

Actuators
- Left turn, Right turn
- Forward, Grab, Release, Shoot

Sensors
- Breeze, Glitter, Smell

Wumpus world characterization

Observable??
- No—only local perception

Deterministic??
- Yes—outcomes exactly specified
- Static??
Wumpus world characterization

Observable?? No—only local perception
Deterministic?? Yes—outcomes exactly specified
Static?? Yes—Wumpus and Pits do not move
Discrete?? Yes

Wumpus world characterization

Observable?? No—only local perception
Deterministic?? Yes—outcomes exactly specified
Static?? Yes—Wumpus and Pits do not move
Discrete?? Yes

Single-agent?? Yes—Wumpus is essentially a natural feature

Exploring a wumpus world (different world to previous picture)
Exploring a wumpus world

- **P?**
- **W**
- A
- S
- OK
- **OK**

Exploring a wumpus world

- **P?**
- **OK**
- **A**
- S
- **OK**

Other tight spots

- **B**
- **OK**
- **A**
- **OK**

Breeze in (1,2) and (2,1) ⇒ no safe actions

Smell in (1,1) ⇒ cannot move
Can use a strategy of coercion:
- shoot straight ahead
  - wumpus was there ⇒ dead ⇒ safe
  - wumpus wasn’t there ⇒ safe
Logics

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; \[ x^2 + y \geq 0 \] is not a sentence.

\[ x + 2 \geq y \] is true if the number \( x + 2 \) is no less than the number \( y \).

\[ x + 2 \geq y \] is false in a world where \( x = 7 \), \( y = 1 \).

Propositional logic: Syntax

Propositional logic is the simplest logic—illustrates basic ideas.

The proposition symbols \( P_1, P_2 \) etc are sentences.

- \( \neg S \) is a sentence U negation ("not")
- \( S_1 \land S_2 \) is a sentence U conjunction ("and")
- \( S_1 \lor S_2 \) is a sentence U disjunction ("or")
- \( S_1 \Rightarrow S_2 \) is a sentence U implication ("if-then")
- \( S_1 \iff S_2 \) is a sentence U biconditional ("if-and-only-if")

Wumpus world sentences

Let \( P_{i,j} \) be true if there is a pit in \([i, j]\).
Let \( B_{i,j} \) be true if there is a breeze in \([i, j]\).

\[
\neg P_{1,1}, \neg P_{2,1}, \neg P_{1,1} \\
\neg B_{1,1}, B_{1,2}, \neg B_{2,1}
\]

"Pits cause breezes in adjacent squares"

\[
P_{1,1} \Rightarrow (B_{1,2} \land B_{2,3} \land B_{1,4}) \\
P_{2,2} \Rightarrow (B_{1,2} \land B_{3,3} \land B_{1,2} \land B_{2,1})
\]

Can we conclude \( \neg P_{2,2} \), \( P_{1,1} \)?
Wumpus world sentences

Let $P_{i,j}$ be true if there is a pit in $[i,j]$. Let $B_{i,j}$ be true if there is a breeze in $[i,j]$. 

$$\neg P_{1,1}, \neg P_{1,2}, \neg P_{3,1}$$

$$\neg B_{1,1}, B_{1,2}, \neg B_{2,1}$$

“A square is breezy if and only if there is an adjacent pit”

$$B_{1,2} \iff (P_{1,1} \lor P_{2,2} \lor P_{3,1})$$

$$B_{2,1} \iff (P_{1,1} \lor P_{2,2} \lor P_{3,1})$$

Can we conclude $\neg P_{2,1}, P_{3,2}$?

What do we mean by “conclude”? 

Propositional logic: Semantics

Specifies true/false for each proposition symbol

E.g. $P_{1,2}$ $P_{2,2}$ $P_{3,1}$

true true false

With $n$ symbols, $2^n$ possible truth assignments can be enumerated automatically.

Rules for evaluating truth of compound sentences:

$$\neg S$$ 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., $S_1 \iff S_2$ is true iff $S_1 \Rightarrow S_2$ is true and $S_2 \Rightarrow S_1$ is true

Models

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 $\alpha$, or $m$ satisfies $\alpha$, if $\alpha$ is true in $m$.

For example, the model illustrated here satisfies the sentences:

$\alpha_1 = $ there is a breeze in square $[4,1]$ and:

$\alpha_2 = $ every square adjacent to a pit has a breeze but not the sentence:

$\alpha_3 = $ every square adjacent to a breeze has a pit.

We say a model $m$ satisfies a set of sentences $KB$ iff $m$ satisfies all sentences $\alpha \in KB$. 

Truth tables for connectives

<table>
<thead>
<tr>
<th>$P$</th>
<th>$Q$</th>
<th>$\neg P$</th>
<th>$P \land Q$</th>
<th>$P \lor Q$</th>
<th>$P \Rightarrow Q$</th>
<th>$P \equiv Q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>

Simple recursive process evaluates an arbitrary sentence, e.g.,

$\neg P_{1,2} \land (P_{2,2} \lor P_{3,1}) = true \land (false \lor true) = true \land true = true$
Entailment

Entailment means that one thing follows from another:

\[ KB \models \alpha \]

Knowledge base \( KB \) entails sentence \( \alpha \) if and only if \( \alpha \) is true whenever \( KB \) is true.

E.g., the KB containing “the Dockers won” and “the Eagles lost” entails “Either the Dockers won or the Eagles won” (irrespective of the other teams).

E.g., \( x + y = 4 \) entails \( 4 = x + y \)

Entailment is a relationship between sentences (i.e., syntax) that is based on semantics.

Models and Entailment

Let \( M(\alpha) \) be the set of all models satisfying \( \alpha \), and \( M(KB) \) be the set of all models satisfying \( KB \).

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

E.g. \( KB = \) Dockers won and Eagles lost
\( \alpha = \) Dockers won

Assume there are 4 teams in the league.

How many possible outcomes (assuming no draws)?

Which results satisfy \( KB \)? Which results satisfy \( \alpha \)?

Does \( KB \models \alpha \)?

Entailment in the wumpus world

Situation after detecting nothing in [1,1], moving right, breeze in [2,1]

Consider possible models for ⬜️ assuming only ⬜️s

3 Boolean choices ⇒ 8 possible models
KB = wumpus-world rules + observations

α₁ = “[1,2] is safe”, KB |= α₁, proved by model checking

α₂ = “[2,2] is safe”, KB ⊭ α₂
Two sentences are logically equivalent if

\( \alpha \equiv \beta \) if and only if \( \models \alpha \) and \( \models \beta \)

\[\begin{array}{cccccc}
\neg B_{21}, \ P_{22} & \Rightarrow B_{12} \land B_{21} & \models \neg P_{22} \\
B_{12} & B_{21} & P_{22} & \neg B_{21} & P_{22} & B_{12} \land B_{21} & \neg P_{22} \\
\hline
F & F & F & T & T & T & T \\
F & F & T & T & F & F & F \\
F & T & F & T & T & F & F \\
F & T & F & T & F & F & F \\
T & F & T & T & F & F & F \\
T & T & F & T & T & T & F \\
T & T & T & F & T & F & F \\
T & T & T & T & T & T & T
\end{array}\]

Enumerate rows (different assignments to symbols), if \( KB \) is true in row, check that \( \alpha \) is too

Validity and satisfiability

A sentence is valid if it is true in all models.

\[\text{e.g., True, } A \lor \neg A, \ A \Rightarrow (A \land (A \Rightarrow B)) \Rightarrow B\]

A sentence is satisfiable if it is true in some model

\[\text{e.g., } A \lor B, \ C\]

A sentence is unsatisfiable if it is true in no models

\[\text{e.g., } A \land \neg A\]