AI 2002
Artificial Intelligence

Dr. Hashim Yasin

Knowledge, Reasoning and Logic

Knowledge

Humans know things …

$\Rightarrow$ **the knowledge helps them to do various tasks.**

$\Rightarrow$ **The knowledge has been achieved**

- not by purely reflex mechanisms
- but by the processes of **reasoning**

- In AI, the example is **knowladge-based agent** which contains **set of sentences** referred as **knowledge-base.**

Knowledge-based Agent

For a generic knowledge-based agent:

- A **percept is given** to the agent.
- The agent **adds the percept** to its knowledge base.
- **Perform best action** according to the knowledge base.
- Tells the knowledge base that it has in fact **taken that action**.

Knowledge-based Agent

function KB-AGENT(*percept*) **returns** an *action*
  persistent: KB, a knowledge base
    t, a counter, initially 0, indicating time

  TELL(KB, MAKE-PERCEPT-SENTENCE(*percept*, t))
  action $\leftarrow$ ASK(KB, MAKE-ACTION-QUERY(t))
  TELL(KB, MAKE-ACTION-SENTENCE(*action*, t))
  $t \leftarrow t + 1$
  return *action*

constructs a **sentence** asserting that the agent ***perceived the given percept*** at time ***t***

constructs a sentence that asks ***what action should be done*** at time ***t***

constructs a sentence that ***the chosen action was executed*** at time ***t***

Logic

- **We'll look at two kinds of logic:**

Propositional Logic

which is relatively simple.

First-order Logic

which is more complicated.

Propositional Logics

Propositional Logic: Syntax

- The **syntax** of propositional logic defines the *allowable sentences*.

What are the sentences?

- Sentance are well formed formulas
- **True** and **False** are sentances
- Propositional variables are sentences. P, Q, R, S etc.

Propositional Logic: Syntax

- The **atomic sentences** consist of a **single proposition symbol**.
- Each such symbol stands for a proposition that can be **True or False**.
- The **complex sentences** are constructed from simpler sentences, using parentheses and **logical connectives**.
- There are five connectives in common use:

  $\neg$ (not), $\wedge$ (and), $\vee$ (or),

  $\Rightarrow$ (implies), $\Leftrightarrow$ (if and only if)

Propositional Logic: Syntax

- **$\neg$ (not)** A sentence such as $\neg W_{1,3}$ is called the negation of $W_{1,3}$.
  - A **literal** is either an atomic sentence (a positive literal) or a negated atomic sentence (a negative literal).
- **$\wedge$ (and)** A sentence whose main connective is $\wedge$, such as $W_{1,3} \wedge P_{3,1}$, is called a conjunction; its parts are the *conjuncts*.
- **$\vee$ (or)** A sentence using $\vee$, such as $(W_{1,3} \wedge P_{3,1}) \vee W_{2,2}$, is a disjunction of the *disjuncts* $(W_{1,3} \wedge P_{3,1})$ and $W_{2,2}$.

Propositional Logic: Syntax

- **⇒ (implies)** A sentence such as $(W_{1,3} \wedge P_{3,1}) \Rightarrow \neg W_{2,2}$ is called an implication (or conditional). The premise or antecedent is $(W_{1,3} \wedge P_{3,1})$.
- Implications are also known as rules or **if–then** statements.
- The implication symbol is sometimes written as ⊃ or → or ⇒.
- **⇔ (if and only if)** The sentence $W_{1,3} \Leftrightarrow \neg W_{2,2}$ is a bi-conditional. Sometime it is written as ≡.

Propositional Logic: Syntax

If S is a sentence, $\neg S$ is a sentence (negation)

If S_1 and S_2 are sentences, $S_1 \wedge S_2$ is a sentence (conjunction)

If S_1 and S_2 are sentences, $S_1 \vee S_2$ 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)

BNF (Backus–Naur Form) Grammar

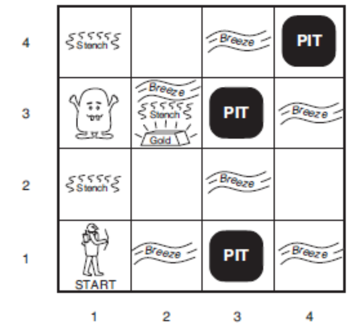
$$
\begin{array}{rcl}
Sentence & \rightarrow & AtomicSentence \mid ComplexSentence \\
AtomicSentence & \rightarrow & True \mid False \mid P \mid Q \mid R \mid \ldots \\
ComplexSentence & \rightarrow & (\ Sentence\ ) \mid [\ Sentence\ ] \\
 & \mid & \neg\ Sentence \\
 & \mid & Sentence \wedge Sentence \\
 & \mid & Sentence \vee Sentence \\
 & \mid & Sentence \ \Rightarrow\ Sentence \\
 & \mid & Sentence \ \Leftrightarrow\ Sentence \\
 & & \\
\textsc{Operator Precedence} & : & \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow
\end{array}
$$

BNF (Backus–Naur Form) Grammar

Precedence Example:

$A \vee B \wedge C$	$A \vee (B \wedge C)$
$A \wedge B \rightarrow C \vee D$	$(A \wedge B) \rightarrow (C \vee D)$
$A \rightarrow B \vee C \leftrightarrow D$	$(A \rightarrow (B \vee C)) \leftrightarrow D$

Propositional Logic: Semantics



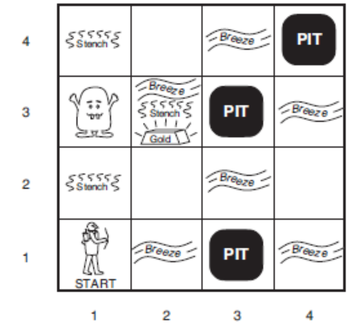
- **The semantics** defines **the rules** for determining the truth value of a sentence with respect to a particular model.
- In propositional logic, a model simply fixes the truth value—**true** or **false**—for every proposition symbol

For example:

- If the sentences in the knowledge base make use of the proposition symbols $\boldsymbol{P}_{1,2}$, $\boldsymbol{P}_{2,2}$, and $\boldsymbol{P}_{3,1}$, then one possible model is:

$$m_1 = \{P_{1,2} = false,\ P_{2,2} = false,\ P_{3,1} = true\}$$

Propositional Logic: Semantics



- The semantics for propositional logic must specify **how to compute the truth value** of *any sentence, given a model.*

For Atomic sentences:

- **True** is true in every model and **False** is false in every model.
- *The truth value of every other proposition symbol must be specified directly in the model.*
  - For example, in the model $\boldsymbol{m_1}$ given earlier, $\boldsymbol{P}_{1,2}$ is false.

$$m_1 = \{P_{1,2} = false,\ P_{2,2} = false,\ P_{3,1} = true\}$$

Propositional Logic: Semantics

For complex sentences

- We have five rules, which hold for any sub-sentences $\boldsymbol{P}$ and $\boldsymbol{Q}$ in any model $\boldsymbol{m}$

- $\neg P$ is true iff P is false in m.
- $P \wedge Q$ is true iff both P and Q are true in m.
- $P \vee Q$ is true iff either P or Q is true in m.
- $P \Rightarrow Q$ is false unless P is true and Q is false in m.
- $P \Leftrightarrow Q$ is true iff P and Q are both true or both false in m.

Propositional Logic: Semantics

- The propositional logic **does not require any relation of *causation* or *relevance*** between P and Q.
  - For example, the sentence **"5 is odd implies Tokyo is the capital of Japan"** is a true sentence of propositional logic, even though it is not a well-formed English sentence.

- In case of implication, **any implication is true whenever its antecedent is false**.
  - For example, **"5 is even implies Sam is smart"** is true, regardless of whether Sam is smart or not.

Propositional Logic: Truth Table

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \rightarrow Q$	$Q \rightarrow P$	$P \leftrightarrow Q$
f	f	t	f	f	t	t	t
f	t	t	f	t	t	f	f
t	f	f	f	t	f	t	f
t	t	f	t	t	t	t	t

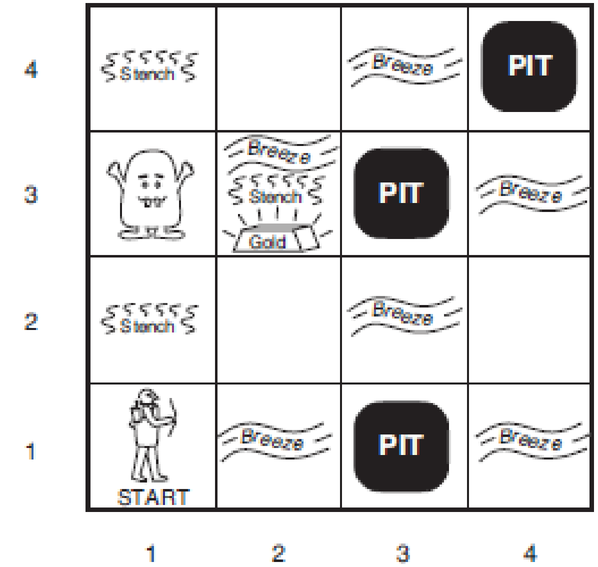
A simple knowledge base

- With propositional logic, we can construct a knowledge base for the Wumpus world.

For Example:

$P_{x,y}$ is true if there is a pit in $[x, y]$.
$W_{x,y}$ is true if there is a wumpus in $[x, y]$, dead or alive.
$B_{x,y}$ is true if the agent perceives a breeze in $[x, y]$.
$S_{x,y}$ is true if the agent perceives a stench in $[x, y]$.

A simple knowledge base



- There is **no** pit in [1,1]:

$$R_1 : \quad \neg P_{1,1} \, .$$

- A square is breezy **if and only if** there is a pit in a neighbouring square. This has to be stated for each square; for now, we include just the relevant squares:

$$R_2 : \quad B_{1,1} \quad \Leftrightarrow \quad (P_{1,2} \vee P_{2,1}) \, .$$
$$R_3 : \quad B_{2,1} \quad \Leftrightarrow \quad (P_{1,1} \vee P_{2,2} \vee P_{3,1}) \, .$$

Standard Logical Equivalences

$$
\begin{aligned}
(\alpha \wedge \beta) &\equiv (\beta \wedge \alpha) \quad \text{commutativity of } \wedge \\
(\alpha \vee \beta) &\equiv (\beta \vee \alpha) \quad \text{commutativity of } \vee \\
((\alpha \wedge \beta) \wedge \gamma) &\equiv (\alpha \wedge (\beta \wedge \gamma)) \quad \text{associativity of } \wedge \\
((\alpha \vee \beta) \vee \gamma) &\equiv (\alpha \vee (\beta \vee \gamma)) \quad \text{associativity of } \vee \\
\neg(\neg \alpha) &\equiv \alpha \quad \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) &\equiv (\neg \beta \Rightarrow \neg \alpha) \quad \text{contraposition} \\
(\alpha \Rightarrow \beta) &\equiv (\neg \alpha \vee \beta) \quad \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) \quad \text{biconditional elimination} \\
\neg(\alpha \wedge \beta) &\equiv (\neg \alpha \vee \neg \beta) \quad \text{De Morgan} \\
\neg(\alpha \vee \beta) &\equiv (\neg \alpha \wedge \neg \beta) \quad \text{De Morgan} \\
(\alpha \wedge (\beta \vee \gamma)) &\equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) \quad \text{distributivity of } \wedge \text{ over } \vee \\
(\alpha \vee (\beta \wedge \gamma)) &\equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) \quad \text{distributivity of } \vee \text{ over } \wedge
\end{aligned}
$$

Reading Material

- **Artificial Intelligence,** A Modern Approach

  Stuart J. Russell and Peter Norvig
  - **Chapter 7.**