Artificial Intelligence

Knowledge-Based AI

Christopher Simpkins

Knowledge and AI

In AI, **knowledge-based** agents use a process of **reasoning** over an internal **representation** of knowledge to decide what actions to take.

- **Knowledge base**: a set of sentences.
- **Sentence**: an assertion about the world expressed in a **knowledge represeantation language**, like propositional logic.
- **Axioms**: sentences taken as given – not derived from other sentences, assumptions.
- **Inference**: deriving new sentences from old sentences.
- **Background knowledge**: sentences present in an agent's knowledge base before it starts perceiving and acting.

Knowledge-Based Agents

- MAKE-PERCEPT-SENTENCE constructs sentence asserting that agent perceived the given percept at the given time.
- MAKE-ACTION-QUERY constructs sentence that asks what action to take at current time.
- MAKE-ACTION-SENTENCE constructs sentence asserting chosen action was executed.

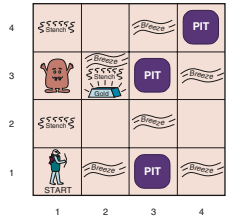
function KB-AGENT(*percept*) **returns** an *action*
  persistent: *KB*, a knowledge base
    t, a counter, initially 0, indicating time

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

- Logical agents are described at the **knowledge level** using **declarative** statements of knowledge and goals.
- At the **implemetation level** we use a **procedureal** approach, encoding behaviors directly in program code.

KENNESAW STATE UNIVERSITY

The Wumpus World



A cave that you can drop into or climb out of at square [1, 1].

- **Performance measure**: +1000 for climbing out of the cave with the gold, –1000 for falling into a pit or being eaten by the wumpus, –1 for each action taken, –10 for using the arrow. The game ends either when the agent dies or when the agent climbs out of the cave.

- **Environment**: A 4×4 grid of rooms, agent always starts at [1,1], facing east. Gold and the wumpus placed uniformly randomly from the squares other than the start square. Each non-start square can be a pit with probability 0.2.
- **Actuators**: Forward, TurnLeft by $90°$, or TurnRight by $90°$. The agent dies if it enters pit or a live wumpus square. Moves into walls have no effect. Grab picks up gold if agent in gold square. Shoot can fires arrow in direction agent is facing. The arrow continues until it either hits (and hence kills) the wumpus or hits a wall. The agent has only one arrow, so only the first Shoot action has any effect. Climb climbs out of the cave if at [1,1].

KENNESAW STATE UNIVERSITY

First Steps Wumpus World

- **Sensors**: The agent has five sensors, each of which gives a single bit of information:
  - In squares directly (not diagonally) adjacent to wumpus, agent perceives a Stench.
  - In squares directly adjacent to a pit, the agent perceives a Breeze.
  - In the square with gold, agent perceives a Glitter.
  - When an agent walks into a wall, it perceives a Bump.
  - When the wumpus is killed, it emits a Scream perceivable anywhere in the cave.

Percepts encoded as list of five symbols indicating presense or absence (by None) of: [Stench,Breeze,Glitter,Bump,Scream] (a bit vector).

- (a) after percept [None,None,None,None,None]
- (b) after moving to [2,1] and perceiving [None,Breeze,None,None,None]

Logic

Basics:

- **Syntax** specifies the form of sentences.
  - $x + y = 4$ is *well-formed*, but $x4y+ =$ is not.
- **Semantics** specifies the meaning of sentences.
  - $x + y = 4$ is **true** in a world where $x = 1$ and $y = 3$.
- **Model**: a formal specification of a *possible world*, that is, a set of assignments of values to the variables in the sentences of a knowledge base.
  - Given a model $\{x = 3, y = 2\}$, the sentence $x + y = 4$ is false.

Satisfaction:

- "m satisfies α": sentence α is true in model m, also "m is a model of α."
- $M(\alpha)$ the set of all models of α, i.e., the set of all models in which α is true.

Entailment: $\alpha \models \beta$: β *follows logically* from α

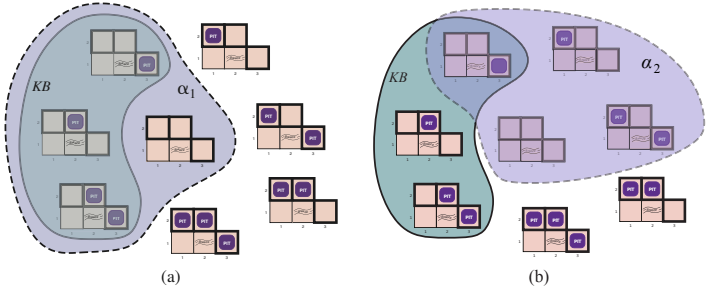
Formal definition of entailment:

$$\alpha \models \beta \text{ if and only if } M(\alpha) \subseteq M(\beta)$$

KENNESAW STATE UNIVERSITY

Possible Models of Pits in Wumpus World

The presence of pits in squares [1, 2], [2, 2] and [3, 1] gives rise to $2^3 = 8$ possible models.



Solid line delineates KB based on percept [None, None, None, None, None] in [1,1].

- (a). α_1 = "There is no pit in [1, 2]." Here, $KB \models \alpha_1$
- (b). α_2 = "There is no pit in [2, 2]." Here, $KB \not\models \alpha_2$

Logical inference via model checking: because of (b), cannot conclude α_2 (or $\neg\alpha_2$).

- $M(KB) \models \alpha_1$ but $M(KB) \not\models \alpha_2$

Inference Algorithms

If an inference algorithm i can derive α from KB, we write

$$KB \vdash_i \alpha$$

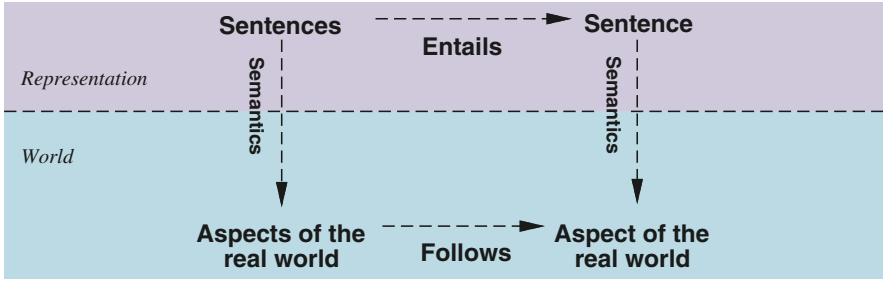
which is pronounced "α is derived from KB by i" or "i derives α from KB."

Important properties of inference algorithms:

- An inference algorithm that derives only entailed sentences is called **sound** or **truth-preserving**.
- An inference algorithm is **complete** if it can derive any sentence that is entailed.

Representation vs World

If KB is true in the real world, then any sentence α derived from KB by a sound inference procedure is also true in the real world?



Grounding: connection between logical reasoning and the real environment. How do we know that KB is true in the real world.

- Subject of volumes of philosophical investigation.
- For us: if agent perceives it, it is true.

Propositional Logic

- **Atomic** sentences consist of a single proposition symbol.
- Proposition symbol stands for a proposition that can be true or false.
  - We use symbols that start with uppercase letter and may contain other letters or subscripts, e.g., : P, Q, R, $W_{1,3}$ and FacingEast
  - True and False have fixed meanings
- **Complex sentence**: one or more atomic sentences constructed from **logical connectives**.
- $\neg$ (not) Unary connective. $\neg W_{1,3}$ is the **negation** of $W_{1,3}$
  - A **positive literal** is an atomic sentence.
  - a ** negative literal** is a negated atomic sentence.
- $\wedge$ (and). Binary connective. **Conjunction**, e.g., $W_{1,3} \wedge P_{3,1}$
- $\vee$ (or). Binary connective. **Disjunction**, e.g., $(W_{1,3} \wedge P_{3,1}) \vee W_{2,2}$
- $\implies$ (implies). Binary connective. **Implication**, e.g., $(W_{1,3} \wedge P_{3,1}) \implies \neg W_{2,2}$
  - $(W_{1,3} \wedge P_{3,1})$ is the **premise** or **antecedent**.
  - $\neg W_{2,2}$ is the **conclusion** or **consequent**.
  - Also known as rules or if-then statements.
  - Some authors use $\supset$ or $\rightarrow$
- $\iff$ (if and only if). Binary connective. $W_{1,3} \iff \neg W_{2,2}$ is a **biconditional**

KENNESAW STATE UNIVERSITY

Grammar of Propositional Logic

$$
\begin{aligned}
\textit{Sentence} &\rightarrow \textit{AtomicSentence} \mid \textit{ComplexSentence} \\
\textit{AtomicSentence} &\rightarrow \textit{True} \mid \textit{False} \mid P \mid Q \mid R \mid \ldots \\
\textit{ComplexSentence} &\rightarrow (\ \textit{Sentence}\ ) \\
&\mid \neg\ \textit{Sentence} \\
&\mid \textit{Sentence} \wedge \textit{Sentence} \\
&\mid \textit{Sentence} \vee \textit{Sentence} \\
&\mid \textit{Sentence} \Rightarrow \textit{Sentence} \\
&\mid \textit{Sentence} \Leftrightarrow \textit{Sentence}
\end{aligned}
$$

$$
\textsc{Operator Precedence} \quad : \quad \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow
$$

Semantics of Propositional Logic

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
false	*false*	*true*	*false*	*false*	*true*	*true*
false	*true*	*true*	*false*	*true*	*true*	*false*
true	*false*	*false*	*false*	*true*	*false*	*false*
true	*true*	*false*	*true*	*true*	*true*	*true*

Propositional Theorem Proving

So far we've done **model checking**: enumerating models and showing that the sentence must hold in all models.

Now we turn to **theorem proving**: applying rules of inference directly to the sentences in our knowledge base to construct a proof of the desired sentence without consulting models.

Some basic concepts:

- **Logical equivalence**: two sentences α and β are logically equivalent if they are true in the same set of models. $\alpha \equiv \beta$
  - $\alpha \equiv \beta$ if and only if $\alpha \models \beta$ and $\beta \models \alpha$
- **Validity**: A sentence is valid if it is true in all models. For example, the sentence $P \wedge \neg P$ is valid. Valid sentences are also known as **tautologies**.
- **Deduction theorem**:
  For any sentences α and β , $\alpha \models \beta$ if and only iff the sentence $(\alpha \implies \beta)$ is valid.
- A sentence is **satisfiable** if it is true in, or satisfied by, *some* model

Inference Rules

General form:

$$\frac{Givens}{Conclusions}$$

Modus Ponens:

$$\frac{\alpha \implies \beta, \alpha}{\beta}$$

And-Elimination:

$$\frac{\alpha \wedge \beta}{\alpha}$$

KENNESAW STATE UNIVERSITY

Logical Equivalences

Functionally equivalent to inference rules. Left side on top, right side on bottom.

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

Representational Power of Formal Languages

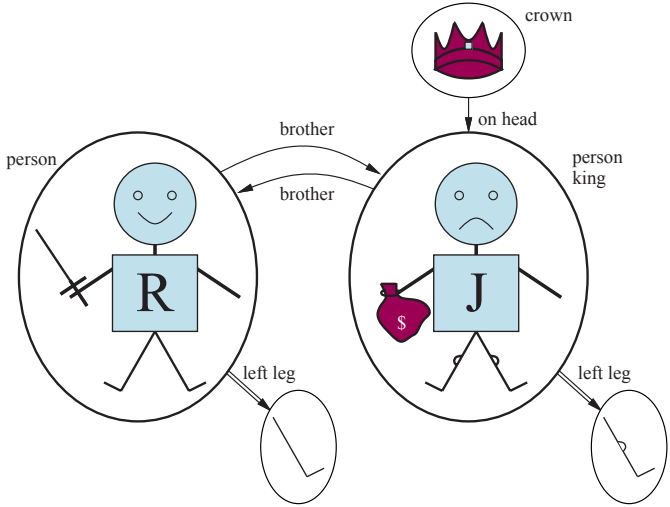
- Propositional logic assumes that there are facts that either hold or do not hold in the world. Each fact can be in one of two states—true or false—and each model assigns true or false to each proposition symbol.
- First-order logic assumes that the world consists of objects with certain relations among them that do or do not hold.

Language	Ontological Commitment (What exists in the world)	Epistemological Commitment (What an agent believes about facts)
Propositional logic	facts	true/false/unknown
First-order logic	facts, objects, relations	true/false/unknown
Temporal logic	facts, objects, relations, times	true/false/unknown
Probability theory	facts	degree of belief $\in [0, 1]$
Fuzzy logic	facts with degree of truth $\in [0, 1]$	known interval value

- **Ontological commitment**: what a language assumes about the nature of reality.
- **Epistemological commitments**: the possible states of knowledge a language allows with respect to each fact.

Sapir-Whorf Hypothesis: you can only think things you can express in a language you know.

Representational Power of First-Order Logic



First-Order Logic

Also known as first-order predicate logic.

- **Constant symbols** stand for objects, e.g., $Richard$, $John$.
- **Predicate symbols** stand for relations, e.g., $Brother(Richard, John)$.
- **Function symbols** stand for functions, e.g., $LeftLeg(John)$
  - Above is also a **term** – a logical expression that refers to an object.

Atomic sentences:

- $Brother(Richard, John)$, $Married(Father(Richard), Mother(John))$

Complex sentences:

- $Brother(Richard, John) \wedge Brother(John, Richard)$

Quantifiers:

- $\forall x, King(x) \implies Person(x)$
  - For all objects x, if x is a $King$, then x is a $Person$.
  - In English:"All kings are persons."
- $\exists x, Crown(x) \wedge Onhead(x, John)$
  - There exists an x such that x is a $Crown$ and x is on the head of $John$.
  - In English: "John has a crown on his head."

KENNESAW STATE UNIVERSITY

Knowledge Representation

Abstract concepts:

Events, Time, Physical Objects, and Beliefs

Representing abstract concepts is sometimes called **ontological engineering**.

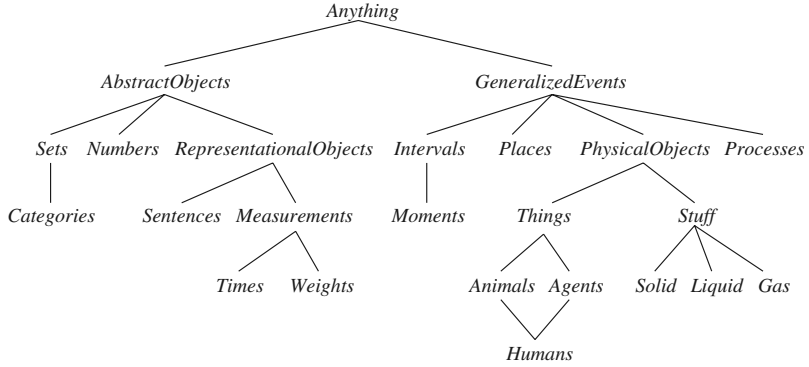
Categories and Objects

Two choices for representing a category:

- use **predicates**, like $Basketball(b)$, or
- **reify** the category as an object itself and say
  - $Member(b, Basketballs)$, or
  - $b \in Basketballs$.

Can also have subcategories, e.g., $Subset(Basketballs, Balls)$ or $Basketballs \subset Balls$.

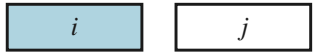
Categories organize knowledge into **inheritance hierarchies**, or **taxonomies**, like this upper ontology of the world:



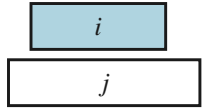
Events

Meet(*i*, *j*)

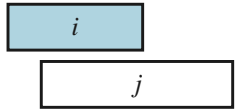
Before(*i*, *j*)
After(*j*,*i*)



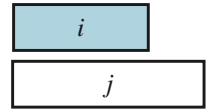
During(*i*, *j*)



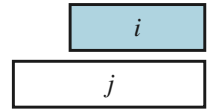
Overlap(*i*, *j*)



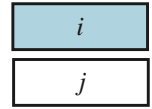
Starts(*i*, *j*)



Finishes(*i*, *j*)



Equals(*i*, *j*)



KENNESAW STATE UNIVERSITY

Fluents