

First-Order Logic

Doug Downey

Northwestern EECS 348 – Intro to AI
Based on slides by Stuart Russell

Pros and Cons of Propositional Logic

- 😊 Declarative: pieces of syntax correspond to facts
- 😊 Allows partial/disjunctive/negated info
 - Unlike most data structures and DBs
- 😊 Compositional:
 - Meaning of $A \vee B$ derived from meanings of A, B
- 😊 Meaning is context-independent
- 😞 Very limited expressiveness
 - Can't say "pits cause breezes in adjacent squares" except by writing a sentence for each square

First-order Logic

- **Objects**
 - Greg, Sheridan Road, 25, blue, Moby Dick (the novel), ...
- **Relations**
 - Tall, four-lane, perfect square, has color, has read, ...
- **Functions**
 - Height, father of, square root, ...

FOL: Syntax

| | |
|-------------|--|
| Constants | <i>KingJohn, 2, UCB, ...</i> |
| Predicates | <i>Brother, >, ...</i> |
| Functions | <i>Sqrt, LeftLegOf, ...</i> |
| Variables | <i>x, y, a, b, ...</i> |
| Connectives | $\wedge \vee \neg \Rightarrow \Leftrightarrow$ |
| Equality | $=$ |
| Quantifiers | $\forall \exists$ |

Atomic Sentences

Atomic sentence = *predicate*(*term*₁, ..., *term*_{*n*})
or *term*₁ = *term*₂

Term = *function*(*term*₁, ..., *term*_{*n*})
or *constant* or *variable*

E.g., *Brother*(*KingJohn*, *RichardTheLionheart*)
> (*Length*(*LeftLegOf*(*Richard*)), *Length*(*LeftLegOf*(*KingJohn*)))

Complex Sentences

- Formed from simple sentences using connectives:

$$\neg S, \quad S_1 \wedge S_2, \quad S_1 \vee S_2, \quad S_1 \Rightarrow S_2, \quad S_1 \Leftrightarrow S_2$$

- E.g.,
 - MayorOf(Daley, Chicago) \Rightarrow LivesIn(Daley, Chicago)
 - Eat(Doug, Nachos) \vee Eat(Doug, Grapes)

Truth in FOL

- A *model* in FOL is a set of objects, and full definitions for relations and functions
- Model enumeration totally infeasible
 - For each relation (of arity k)
 - For each distinct subset of all combinations of k objects...
- Technically, objects/relations in model are mapped to particular KB symbols through an *interpretation*
 - This provides yet more complexity

Universal Quantification

- \forall <variables> <sentence>
- Everyone at Northwestern is smart:
 $\forall x \text{ At}(x, \text{Northwestern}) \Rightarrow \text{Smart}(x)$
- $\forall x P$ is true in a model m if P is true with **any** object from m substituted for x in P

Common Mistake to Avoid

- Typically \Rightarrow is the main connective with \forall
- Don't use \wedge by mistake, e.g.:
 $\forall x \text{ At}(x, \text{Northwestern}) \wedge \text{Smart}(x)$

Existential Quantification

- \exists <variables> <sentence>
- Someone at Northwestern is rich:
 $\exists x \text{ At}(x, \text{Northwestern}) \wedge \text{Rich}(x)$
- $\exists x P$ is true in a model m if P is true with **some** object from m substituted for x in P

Common Mistake to Avoid

- Typically \wedge is the main connective with \exists
- Don't use \Rightarrow by mistake, e.g.:
$$\exists x \text{ At}(x, \text{Northwestern}) \Rightarrow \text{Rich}(x)$$
- True if anyone is not at Northwestern!

Properties of Quantifiers

- $\forall x \forall y$ is the same as $\forall y \forall x$
- $\exists x \exists y$ is the same as $\exists y \exists x$
- $\exists x \forall y$ is **not** the same as $\forall y \exists x$

- Related through negation:
 - $\forall x \text{ Likes}(x, \text{IceCream}) \Leftrightarrow \neg \exists x \neg \text{ Likes}(x, \text{IceCream})$
 - $\exists x \text{ Likes}(x, \text{Broccoli}) \Leftrightarrow \neg \forall x \neg \text{ Likes}(x, \text{Broccoli})$

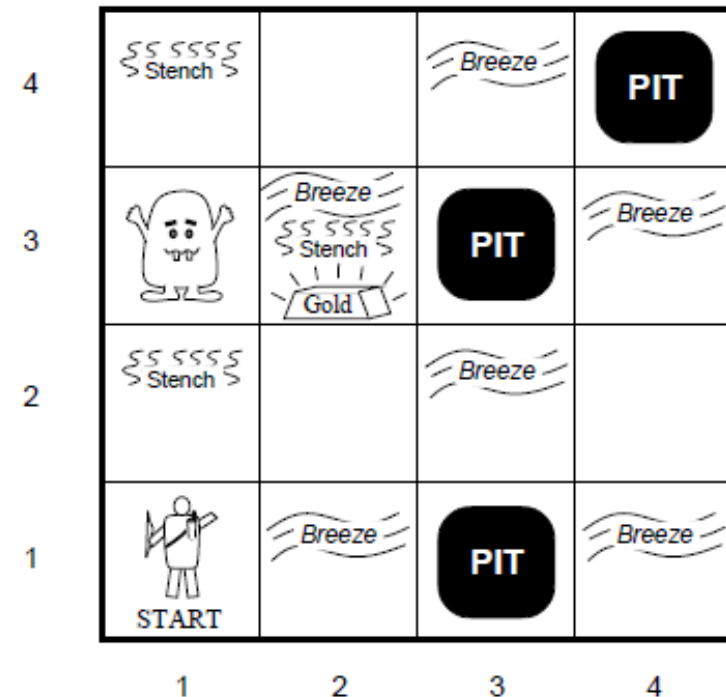
Examples!

- Brothers are siblings
 - $\forall x \forall y \text{ Brother}(x, y) \Rightarrow \text{Sibling}(x, y)$
- Sibling is symmetric:
 - $\forall x \forall y \text{ Sibling}(x, y) \Rightarrow \text{Sibling}(y, x)$
- One's mother is one's female parent
 - $\forall x \forall y \text{ Mother}(x, y) \Leftrightarrow (\text{Female}(x) \wedge \text{Parent}(x, y))$
- A first cousin is a child of a parent's sibling...

Payoff in the Wumpus World

- FOL is concise
 - Can define $\text{Adjacent}(x, y)$ for squares x, y
 - Then:

$$\forall y \text{ Breezy}(y) \Leftrightarrow [\exists x \text{ Pit}(x) \wedge \text{Adjacent}(x, y)]$$



Interlude: Recap and RoadMap

- So far – classical AI
 - Overview and Philosophy of AI (e.g., Turing Test)
 - General search methods and techniques
 - A*, Local Search, CSPs, etc.
 - Logic
 - Logical Agents, Propositional Logic, FOL (inference: today)
- To Come
 - “modern AI”
 - Challenges for classical AI
 - Leveraging data (machine learning)
 - The “big questions”

Interacting with FOL KBs

- Say a wumpus-world agent perceives a smell and a breeze, but no glitter, at $t = 5$:

Tell(KB, Percept([Smell, Breeze, None], 5))

Ask(KB, $\exists a$ Action(a, 5))

- I.e., does the KB entail any actions at $t = 5$?
 - Answer: Yes, $\{a / \text{Shoot}\}$ \leq **substitution**
- Ask(KB, S) returns the substitutions of S that KB entails

Inference in first-order knowledge bases

- We just said:
 - Ask(KB, S) returns the substitutions of S that KB entails
- How?
 - Propositionalize
 - “Instantiation”
 - Forward/Backward Chaining
 - *Resolution*



Key Idea: Unification

Resolution use in practice

- Theorem provers
 - Provided major mathematical results
 - “Otter” proved a conjecture (the Robbins algebra) which had been unsolved for 60 years
 - Verification
 - Hardware (adders, CPUs)
 - Algorithms
 - RSA encryption
 - Software
 - Spacecraft control

Logic as a Foundation for AI

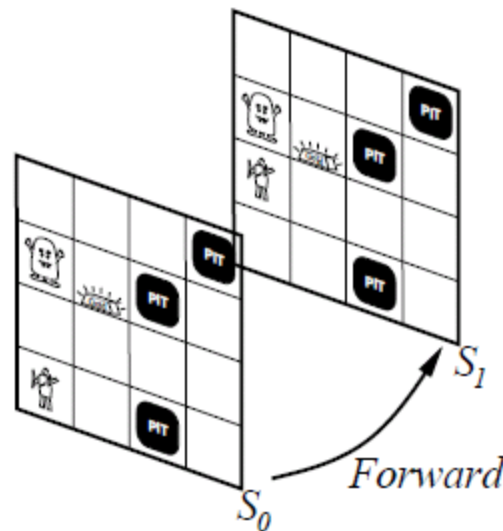
- Logic: extremely expressive, powerful
 - Theorem provers: useful in practice
- But:
 - Writing down needed knowledge is hard
 - So-called Frame, qualification, ramification problems
 - => Knowledge acquisition bottleneck
 - Logic systems are “incomplete”
 - Logic systems are brittle

The real world: Sensing and Acting

- Perception
 - three binary inputs [smell, breeze, glitter] at each time t
 - $\forall s, b, t$ $\text{Percept}([s, b, \text{Glitter}], t) \Rightarrow \text{AtGold}(t)$
- $\forall t$ $\text{AtGold}(t) \Rightarrow \text{Action}(\text{Grab}, t)$?
 - Infinite Loop!
- $\forall t$ $\text{AtGold}(t) \wedge \neg \text{Holding}(\text{Gold}, t) \Rightarrow$
 $\text{Action}(\text{Grab}, t)$

Keeping track of Change

- Facts hold in particular situations
 - E.g., $\text{Holding}(\text{Gold}, t)$ may be False, $\text{Holding}(\text{Gold}, t+8)$ true
- Agent must keep track of change



Frame Problem

- Effect axioms
 - $\forall t$ Standing((i, j), t) \wedge Facing(Up, t) \wedge Action(Forward, t)
 \Rightarrow Standing((i,j+1), $t + 1$)
- But...HaveArrow($t + 1$) ?
- “Frame” axioms keep track of what *doesn't* change
 - Action(Forward, t) \Rightarrow (HaveArrow(t) \wedge HaveArrow($t + 1$))
 - Etc. etc. etc.

Representational Frame Problem

- Historically thought to be extremely tricky
- Can be solved by writing axioms about fluents rather than actions

Holding(Gold, t)

\Leftrightarrow

\neg Holding(Gold, t-1) and action at t-1 made it true

or

Holding(Gold, t-1) and no action at t-1 made it false

Qualification Problem

- Action's preconditions can be complex
- $\text{Action}(\text{Grab}, t) \Rightarrow \text{Holding}(t)$

....unless gold is slippery or nailed down or too heavy or our hands are full or...

Ramification Problem

- Actions can have many consequences
 - $\forall t \text{ Standing}((i, j), t) \wedge \text{Facing}(\text{Up}, t) \wedge \text{Action}(\text{Forward}, t)$
 $\Rightarrow \text{Standing}((i, j+1), t + 1)$
 - But also
 $\Rightarrow \text{In}(\text{Basketball}, (i, j+1), t + 1)$
 - if I'm holding a basketball**
- Writing all these down -- difficult

Knowledge Acquisition

- Remember the Colonel West story
 - We converted text to logic
 - In practice...who does this?
- Qualification, Ramification problems tell us we need *tons* of “common-sense” knowledge
- The infamous “knowledge acquisition bottleneck”

Knowledge Acquisition: Options

- Type it all in yourself
 - Cyc
- Get Web citizens to type it all in
 - Open Mind
- Extract it from the Web
 - KnowItAll, TextRunner, Fusion Tables, Wolfram Alpha, WikiData....

Logic as a Foundation for AI

- Logic: extremely expressive, powerful
 - Theorem provers: useful in practice
- But:
 - Writing down needed knowledge is hard
 - So-called Frame, qualification, ramification problems
 - => Knowledge acquisition bottleneck
 - **Logic systems are “incomplete”**
 - Logic systems are brittle

Gödel's Incompleteness Theorem

- Completeness Theorem: All valid statements have proofs in FOL
- Incompleteness Theorem: For any FOL KB enhanced to allow mathematical induction, there are true statements that *can't* be proved.

Gödel's Theorem: Sketch (1)

- Idea:

This statement is false.

- More specifically:

This statement has no proof.

Gödel's Theorem: Sketch (2)

- Assign numbers to sentences, proofs
 - E.g. by sorting by length, then alphabetically
- Consider the sentence $\alpha(j, A)$
 - For all numbers i , statement $\#i$ is not a proof for statement $\#j$ from the axioms A
- Let σ be the sentence $\alpha(\#\sigma, A)$
 - σ false? But it has a proof!
 - σ true? It's unprovable!

Gödel's Theorem: Ramifications

- Argument: Computers are limited by Gödel's theorem, whereas humans aren't.
- Thus, AI is doomed

Three counter-arguments

- Gödel's theorem applies to math induction systems, e.g. Turing Machines
 - Computers aren't *really* Turing machines
- “Steve cannot say this sentence is true.”
 - But Steve might be able to do other cool stuff
- Are humans really immune to the theorem?

Logic as a Foundation for AI

- Logic: extremely expressive, powerful
 - Theorem provers: useful in practice
- But:
 - Writing down needed knowledge is hard
 - So-called Frame, qualification, ramification problems
 - => Knowledge acquisition bottleneck
 - Logic systems are “incomplete”
 - **Logic systems are brittle**

Brittleness of Logic Systems

- Consider a KB with just one contradiction
- That KB entails everything
- This is a problem because much of the world is uncertain
 - Perception, action, incomplete information, controversies, etc.

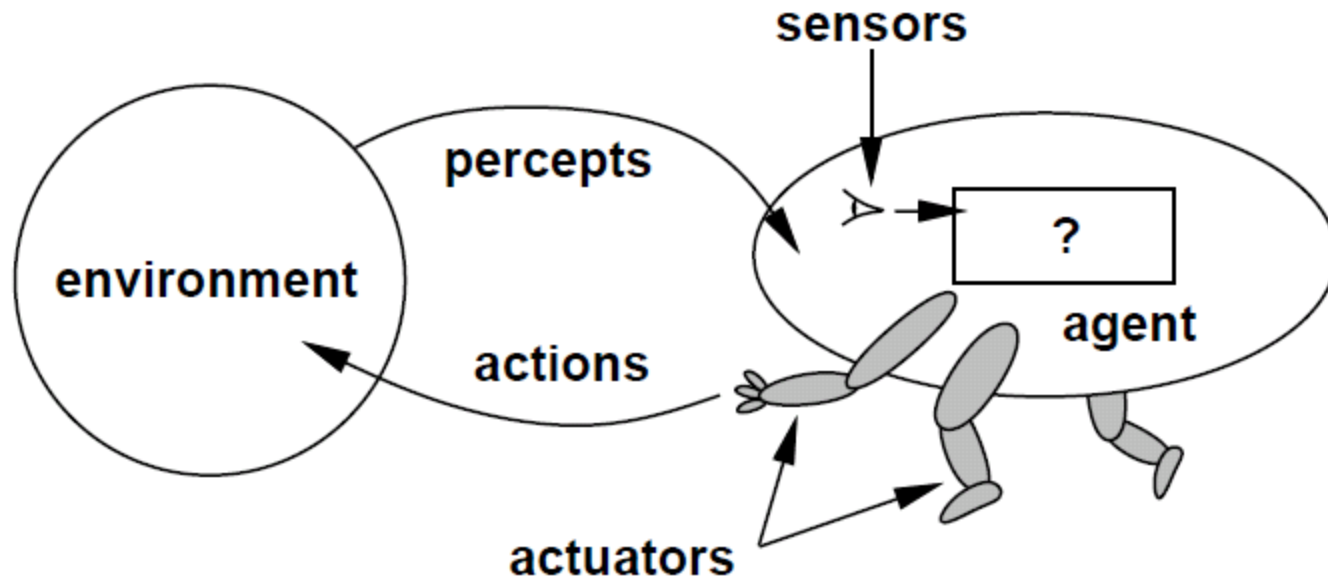
Toward “Modern” AI

- Limitations:
 - Knowledge Acquisition Bottleneck, Brittleness
- “Modern” directions:
 - Situatedness, embodiment
 - Probability
 - Learning from data
 - *Combining* the above with logic

Alternatives: Focus on Behavior

- Argument: we can't even build systems that do what ants do
- In the timeline of evolution, simple cells->ants took much longer than ants->humans
- Let's start by building ants
 - Environment, body can make tasks **easier**
 - Incrementally solve **real** problems end-to-end

Intelligent Agents



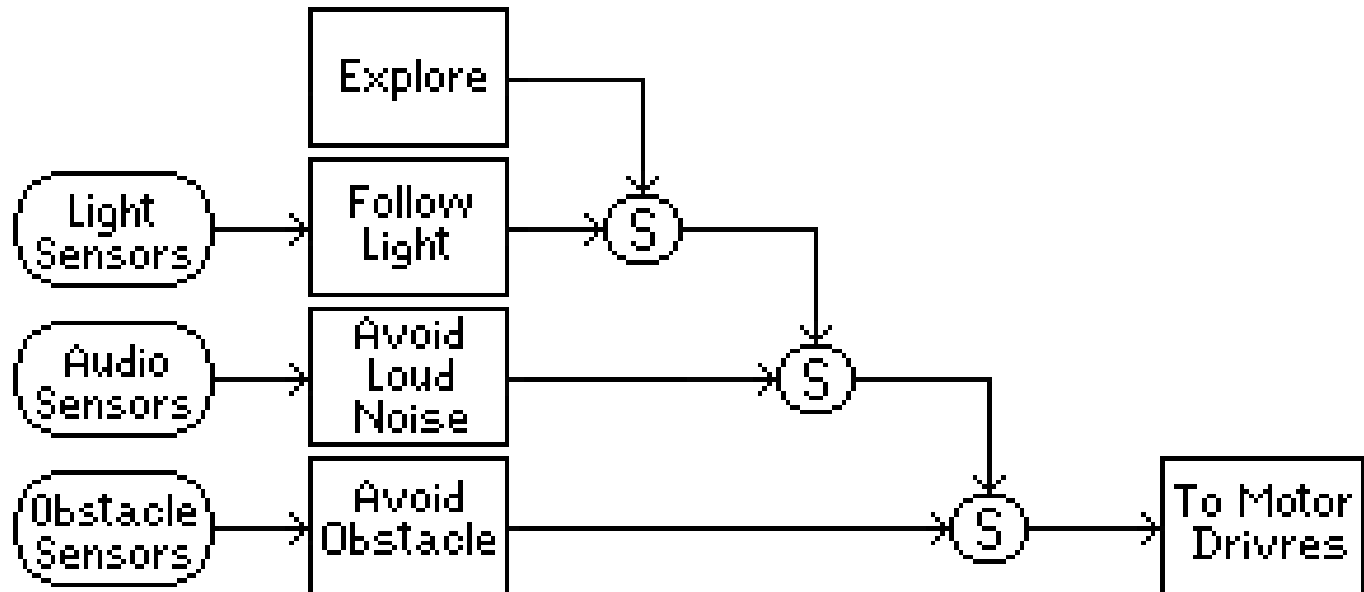
- Sensory/motor aspect
 - more **important**, more **coupled**, more **integrated** with rest of intelligence than originally thought

Behavior-based robots as a foundation for AI

- Common-sense knowledge arises from our interaction in the world
- Thus, the road to AI is paved with real-world interaction
 - We must build robots
- Another possibility: softbots

Subsumption Architecture

- Behavior-based robotics



Other “modern” trends

- Biological inspiration, e.g.:
 - Neural networks
 - Hexapod robots drawing on insect nervous systems followed subsumption architecture
- Probability theory
 - Handles uncertainty, overcomes brittleness
- Data

Learning from Data

- Quantities of data are exploding -- let's learn from it
- “Machine learning”