Introduction to Artificial Intelligence

Chapter 3: Knowledge Representation and Reasoning (3) First-order Logic

Nguyễn Hải Minh, Ph.D
nhminh@fit.hcmus.edu.vn

Outline

- ❑Why First Order Logic (FOL)?
- ❑Syntax and semantics of FOL
- ❑Using FOL
- ❑Wumpus world in FOL
- ❑Knowledge engineering in FOL

Pros and cons of propositional logic

- Propositional logic is declarative
- Propositional logic allows partial/disjunctive/negated information
- Propositional logic is compositional:
  - meaning of $B_{1,1} \land P_{1,2}$ is derived from meaning of $B_{1,1}$ and of $P_{1,2}$
- Meaning in propositional logic is context-independent
  - unlike natural language, where meaning depends on context
- Propositional logic has very limited expressive power
  - E.g., cannot say "pits cause breezes in adjacent squares“
    - except by writing one sentence for each square
    - $B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$, $B_{2,2} \Leftrightarrow (P_{1,2} \lor P_{2,1} \lor P_{3,1} \lor P_{1,3})$

Pros and cons of propositional logic

- Sentences that can not be represented using Propositional logic
  - Because Socrates is a human, Socrates dies.
  - When a box is painted blue, it becomes a blue box
  - A student can log in to Moodles if he is given an account and the teacher adds him to the class.

Facts about some or all of the objects in the universe

General rules

First-order logic

- ❑Whereas propositional logic assumes the world contains facts,
- ❑**First-order logic** (like natural language) assumes the world contains
  - o Objects: people, houses, numbers, colors, Bill Gates, games, wars, ...
  - o Relations:
    - Properties: red, round, prime,
    - *n-ary* relations: brother of, bigger than, part of, comes between, ...plus,
  - o Functions: father of, best friend, one more than, ...

First-order logic – Example

1. *"One plus two equals three."*
   - Object: one, two, three, one plus two
   - Relation: equal
   - Function: plus
2. *"Squares neighboring the wumpus are smelly."*
   - Object: squares, Wumpus
   - Property: smelly
   - Relation: neighboring
3. *"Intelligent AlphaGo beat the world champion in 2016."*
   - Object: AlphaGo, world champion, 2016
   - Relation: beat
   - Property: intelligent

5 Types of Logics

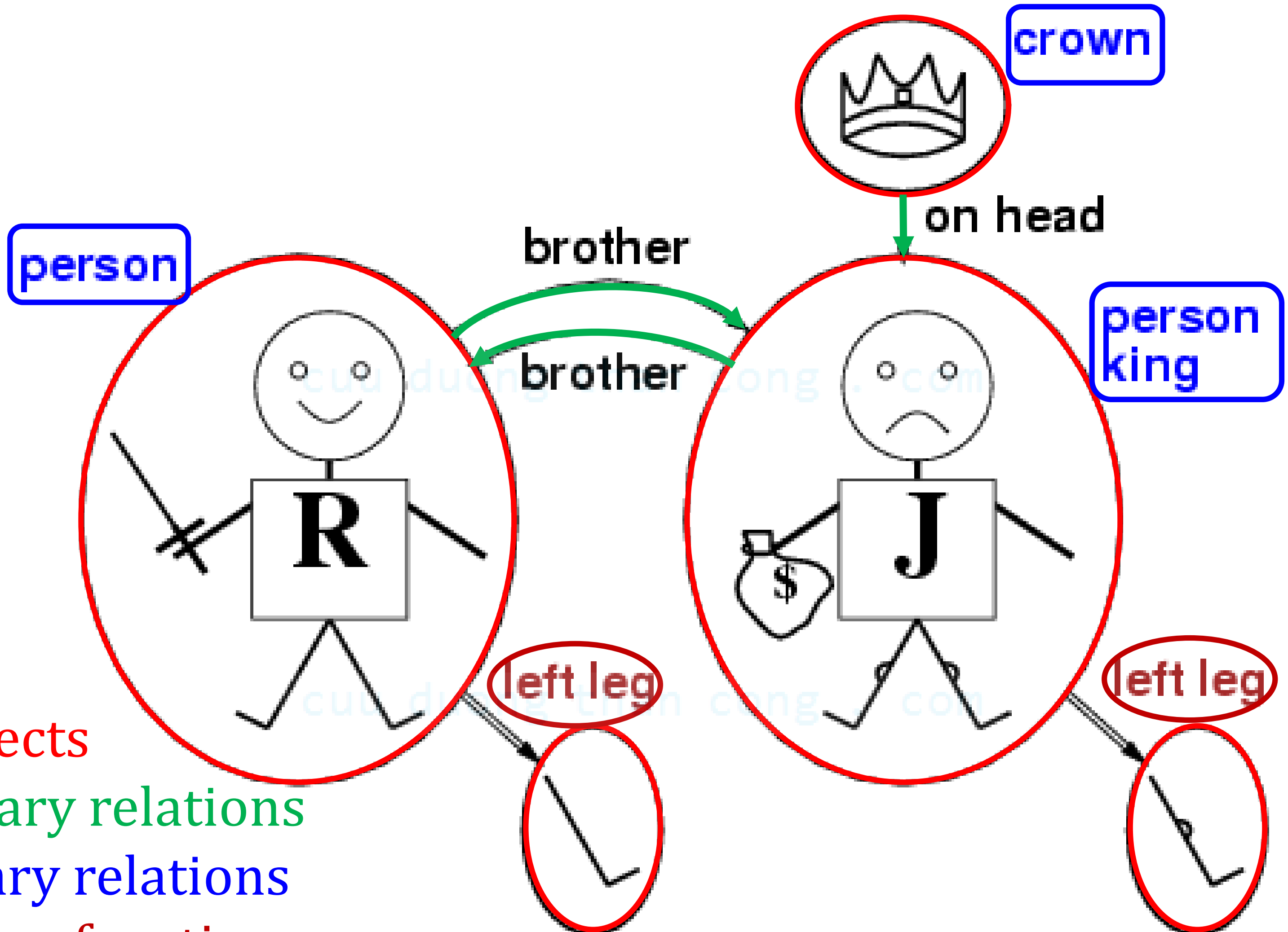
Language	Ontological Commitment (What exists in the world)	Epistemological Commitment (What an agent believes about facts)
Propositional logic	Facts	True/false/unknown
First-order logic	Facts, objects, relations	True/false/unknown
Temporal logic	Facts, objects, relations, time	True/false/unknown
Probability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

Formal languages and their ontological and epistemological commitments of 5 types of logics

Models for FOL

- FOL models have **objects** in them
  - Domain of a model is the set of objects it contains
  - Domain must not be empty
  - It doesn't matter what these objects are, but how many there are in each particular model

Models for FOL: Example



- 5 objects
- 2 binary relations
- 3 unary relations
- 1 unary function

Models for FOL: Example

- 5 objects:
  - Richard (King of England 1189-1199)
  - John (King of England 1199-1215)
  - The left leg of Richard
  - The left leg of John
  - A crown

- Relations:
  - Binary relations:
    - The brotherhood relation: {<Richard, John>. <John, Richard>}
    - The "on head" relation: {<The crown, John>}
  - Unary relations: "person", "king", "crown"
  - Functions: "left leg"
    - <Richard> → Richard's left leg
    - <John> → John's left leg

Syntax of FOL: Basic elements

- Constants AlphaGo, John, US, ...
- Predicates Brother, >,...
- Functions Sqrt, LeftLegOf,...
- Variables x, y, a, b,...
- Connectives $\neg, \Rightarrow, \wedge, \vee, \Leftrightarrow$
- Equality =
- Quantifiers $\forall, \exists$

Syntax of FOL: Terms

- A term: a logical expression that refers to an object.
  - Constant symbols: John
  - Function symbols: LeftLeg(John)

Term = $\mathit{function}(\mathit{term}_1,\ldots,\mathit{term}_n)$ or *constant* or *variable*

Syntax of FOL: Atomic Sentences

- An atomic sentence (Atom) is formed from a predicate symbol followed by a parenthesized list of terms
  - Brother(Richard, John)
  - Married(Father(Richard), Mother(John))

Atomic sentence = $\mathit{predicate}(\mathit{term}_1,\ldots,\mathit{term}_n)$

Syntax of FOL: Complex Sentences

- Complex sentences are made from atomic sentences using connectives
  - ¬Brother (LeftLeg(Richard), John)
  - Brother (Richard , John) ∧ Brother (John, Richard)
  - King(Richard) ∨ King(John)
  - ¬King(Richard) ⇒ King(John)
  - …

Truth in first-order logic

- Sentences are true with respect to a model and an interpretation
- Model contains objects (domain elements) and relations among them
- Interpretation specifies referents for
  - constant symbols → objects
  - predicate symbols → relations
  - function symbols → functional relations
- An atomic sentence *predicate*($term_1$,...,$term_n$) is true

  iff the objects referred to by $term_1$,...,$term_n$

  are in the relation referred to by *predicate*

Syntax of FOL: Universal Quantification

∀<variables> <sentence>

- ❑∀: For all…
- ❑E.g., "All kings are persons": ∀x King(x) ⇒ Person(x)

"Students of FIT are intelligent: ∀x Student(x, FIT) ⇒ Smart(x)

∀x *P* is true in a model *m* iff *P* is true with *x* being each possible object in the model

→Equivalent to the conjunction of instantiations of *P*

Student(Lan, FIT) ⇒ Smart(Lan)
∧ Student(Tuan, FIT) ⇒ Smart(Tuan)
∧ Student(Long, FIT) ⇒ Smart(Long)
∧ …

A common mistake to avoid

- Typically, $\Rightarrow$ is the main connective with $\forall$
- Common mistake: using $\wedge$ as the main connective with $\forall$:
  - $\forall x\ Student(x, FIT) \wedge Smart(x)$
  - means "Everyone is a student of FIT and everyone is smart"

Syntax of FOL: Existential Quantification

$\exists$ *<variables> <sentence>*

- ∃: Some of the collection
- E.g., "Some students of FIT are intelligent:

  ∃x Student(x, FIT) ⇒ Smart(x)

∃ x *P* is true in a model *m* iff *P* is true with *x* being some possible object in the model

→Equivalent to the disjunction of instantiations of *P*

Student(Lan, FIT) ∧ Smart(Lan)
∨ Student(Tuan, FIT) ∧ Smart(Tuan)
∨ Student(Long, Fit) ∧ Smart(Long)
∨ ...

Another common mistake to avoid

- Typically, $\wedge$ is the main connective with $\exists$

- Common mistake: using $\Rightarrow$ as the main connective with $\exists$:

$$\exists x \text{ Student}(x, \text{FIT}) \Rightarrow \text{Smart}(x)$$

is true if there is anyone who is not at FIT!

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 $\forall y\ \exists x$
  - ○ $\exists x\ \forall y$ Loves(x,y)

    → "There is a person who loves everyone in the world"
  - ○ $\forall y\ \exists x$ Loves(x,y)

    → "Everyone in the world is loved by at least one person"
- ❑ **Quantifier duality:** each can be expressed using the other
  - ○ $\forall x$ Likes(x,IceCream) $\neg\exists x \neg$Likes(x,IceCream)
  - ○ $\exists x$ Likes(x,Broccoli) $\neg\forall x \neg$Likes(x,Broccoli)

Equality

- $term_1 = term_2$ is true under a given interpretation if and only if $term_1$ and $term_2$ refer to the same object
- E.g., definition of *Sibling* in terms of *Parent*:
  - $\forall x,y\ Sibling(x,y) \Leftrightarrow [\neg(x = y) \land \exists m,f \neg (m = f) \land Parent(m,x) \land Parent(f,x) \land Parent(m,y) \land Parent(f,y)]$

Using FOL: The kinship domain

- Brothers are siblings
  - $\forall x,y\ Brother(x,y) \Leftrightarrow Sibling(x,y)$
- One's mother is one's female parent
  - $\forall m,c\ Mother(c) = m \Leftrightarrow (Female(m) \wedge Parent(m,c))$
- "Sibling" is symmetric
  - $\forall x,y\ Sibling(x,y) \Leftrightarrow Sibling(y,x)$
- DIY:
  - Parent and child are inverse relations
  - A grandparent is a parent of one's parent
  - A sibling is another child of one's parent
  - One's husband is one's male spouse

Using FOL: The set domain

- Sets are the empty set and those made by adjoining something to a set:
  - $\forall s\ Set(s) \Leftrightarrow (s = \{\}\ ) \lor (\exists x,s_2\ Set(s_2) \land s = \{x|s_2\})$
- The empty set has no elements adjoined into it.
  - $\neg\exists x,s\ \{x|s\} = \{\}$

- Adjoining an element already in the set has no effect:
  - $\forall x,s\ x \in s \Leftrightarrow s = \{x|s\}$
- The only members of a set are the elements that were adjoined into it.
  - $\forall x,s\ x \in s \Leftrightarrow [\ \exists y,s_2\}\ (s = \{y|s_2\} \land (x = y \lor x \in s_2))]$

- Can you interpret the following sentences?
  - $\forall s_1,s_2\ s_1 \subseteq s_2 \Leftrightarrow (\forall x\ x \in s_1 \Rightarrow x \in s_2)$
  - $\forall s_1,s_2\ (s_1 = s_2) \Leftrightarrow (s_1 \subseteq s_2 \land s_2 \subseteq s_1)$
  - $\forall x,s_1,s_2\ x \in (s_1 \cap s_2) \Leftrightarrow (x \in s_1 \land x \in s_2)$
  - $\forall x,s_1,s_2\ x \in (s_1 \cup s_2) \Leftrightarrow (x \in s_1 \lor x \in s_2)$

Using FOL: The Wumpus World

- Typical percept sentence:
  - Percept([Stench, Breeze, Glitter, None, None]. 5)
- Actions:
  - Turn(Right), Turn(Left), Forward, Shoot, Grab, Release, Climb
- To determine the best action, construct query:
  - ASKVARS(∃a BestAction(a, 5))
  - Returns a binding list such as {a/Grab}

QUIZ

Write this sentence using FOL:

"Students can miss some classes of all courses, and they can miss all classes of some courses, but they cannot miss all classes of all courses."

Giving the following predicates:

- Student(x) = x is a student
- Class(z, y) = z is a class of course y
- Miss(x, z) = x miss class z

Deadline: 20h today on Moodles

Knowledge base for the Wumpus World

❑Perception

- $\forall$ t, s, g, m, c Percept ([s, Breeze, g, m, c], t) $\Rightarrow$ Breeze(t)
- $\forall$ t, s, b, m, c Percept ([s, b, Glitter, m, c], t) $\Rightarrow$ Glitter (t)

  …

❑Reflex

- $\forall$t Glitter(t) $\Rightarrow$ BestAction(Grab,t)

Deducing hidden properties

- Environment definition:

  $\forall x,y,a,b\ Adjacent([x,y],[a,b]) \Leftrightarrow$

  $(x = a \wedge (y = b - 1 \vee y = b + 1)) \vee (y = b \wedge (x = a - 1 \vee x = a + 1))$

  - Properties of squares:

    $\forall s,t\ At(\text{Agent},s,t) \wedge \text{Breeze}(t) \Rightarrow \text{Breezy}(s)$

- Squares are breezy near a pit:
  - Diagnostic rule---infer cause from effect

    $\forall s\ \text{Breezy}(s) \Leftrightarrow \exists\ r\ \text{Adjacent}(r, s) \wedge \text{Pit}(r)$

  - Causal rule---infer effect from cause

    $\forall r\ \text{Pit}(r) \Leftrightarrow [\forall s\ \text{Adjacent}(r,s) \Rightarrow \text{Breezy}(s)]$

Summary

- First-order logic:
  - objects and relations are semantic primitives
  - syntax: constants, functions, predicates, equality, quantifiers
- Increased expressive power: sufficient to define wumpus world