CSE 604

Artificial Intelligence

Chapter 8: First Order Logic

Adapted from slides available in Russell & Norvig's textbook webpage

Dr. Ahmedul Kabir

IIT
University of Dhaka

Outline

- Why 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
  – Pieces of syntax correspond to fact

☺ Propositional logic allows partial/disjunctive/negated information
  – (unlike most data structures and databases)

☺ Propositional logic is compositional:
  – meaning of $B_{1,1} \wedge P_{1,2}$ is derived from meaning of $B_{1,1}$ and of $P_{1,2}$

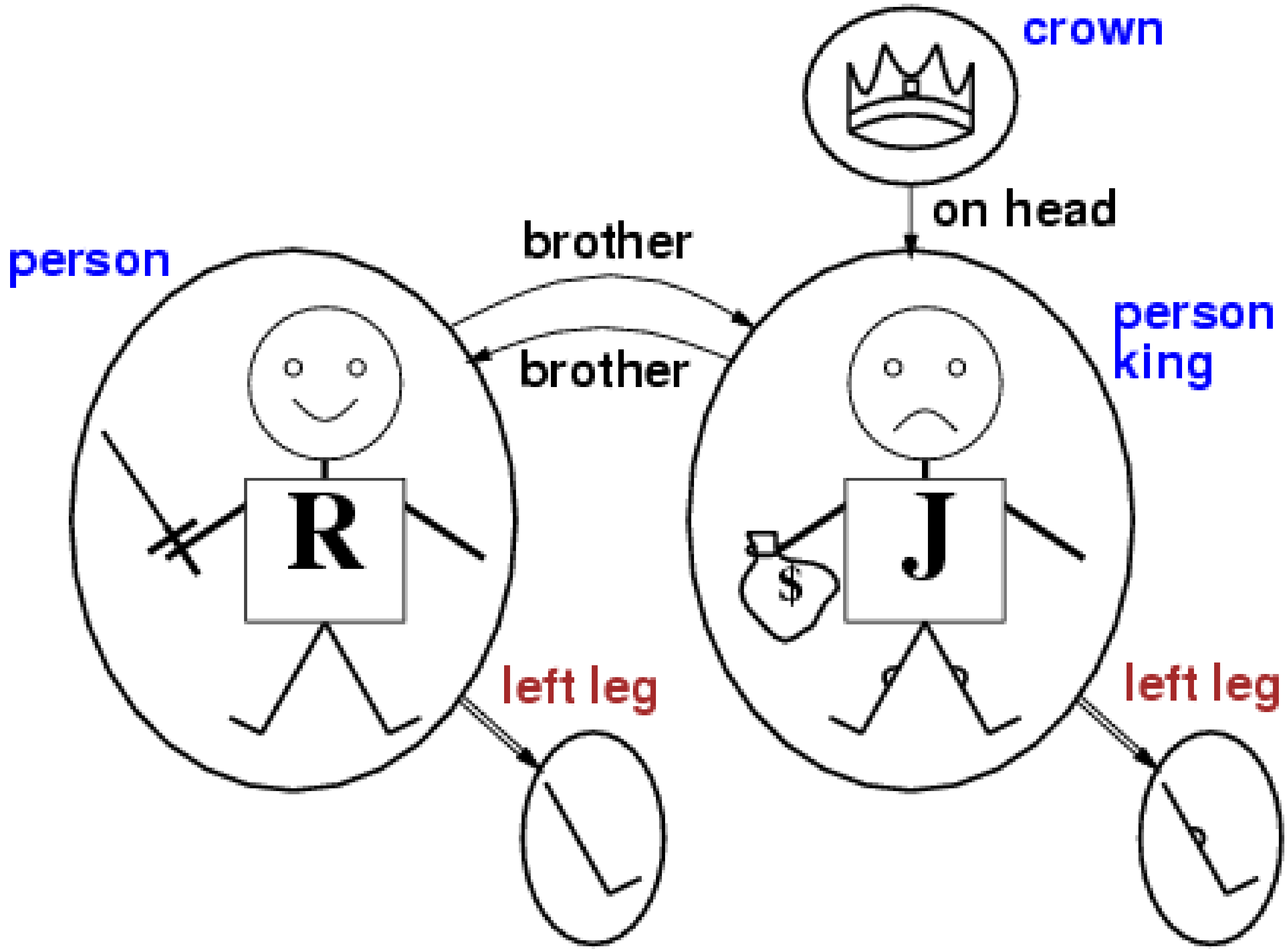
☺ Meaning in propositional logic is context-independent
  – (unlike natural language, where meaning depends on context)

☹ Propositional logic has very limited expressive power
  – (unlike natural language)
  – E.g., cannot say "pits cause breezes in adjacent squares“
    • except by writing one sentence for each square

First-order logic

- Whereas propositional logic assumes the world contains facts,
- first-order logic (like natural language) assumes the world contains
  - Objects: people, houses, numbers, colors, baseball games, wars, …
  - Relations: red, round, prime, brother of, bigger than, part of, comes between, …
  - Functions: father of, best friend, one more than, plus, …

Models for FOL: Example



Syntax of FOL: Basic elements

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

Atomic sentences

Atomic sentence = *predicate* $(term_1, \ldots, term_n)$
or $term_1 = term_2$

Term = *function* $(term_1, \ldots, term_n)$
or *constant* or *variable*

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

Complex sentences

- Complex sentences are made from atomic sentences using connectives

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

E.g. $Sibling(KingJohn,Richard) \Rightarrow Sibling(Richard,KingJohn)$

$>(1,2) \vee \leq (1,2)$

$>(1,2) \wedge \neg >(1,2)$

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 $predicate(term_1,\ldots,term_n)$ is true
  iff the objects referred to by $term_1,\ldots,term_n$
  are in the relation referred to by *predicate*

Universal quantification

- $\forall$ <*variables*> <*sentence*>

  Everyone at IIT is smart:

  $\forall x\ At(x, IIT) \Rightarrow Smart(x)$

- $\forall x\ P$ is true in a model *m* iff *P* is true with *x* being each possible object in the model

- Roughly speaking, equivalent to the conjunction of instantiations of *P*

$$
\begin{aligned}
& At(KingJohn,IIT) \Rightarrow Smart(KingJohn) \\
\wedge\ & At(Richard,IIT) \Rightarrow Smart(Richard) \\
\wedge\ & At(Pikachu,IIT) \Rightarrow Smart(Pikachu) \\
\wedge\ & \ldots
\end{aligned}
$$

A common mistake to avoid

- Typically, $\Rightarrow$ is the main connective with $\forall$

- **Common mistake**: using $\wedge$ as the main connective with $\forall$:

  $\forall x\ At(x, IIT) \wedge Smart(x)$

  means "Everyone is at IIT and everyone is smart"!

Existential quantification

- $\exists$<*variables*> <*sentence*>
- Someone at CSE is smart:
  $\exists x$ At(x,CSE) $\wedge$ Smart(x)

- $\exists x\ P$ is true in a model *m* iff *P* is true with *x* being some possible object in the model
- Roughly speaking, equivalent to the disjunction of instantiations of *P*

  At(KingJohn,CSE) $\wedge$ Smart(KingJohn)
  $\vee$ At(Richard, CSE) $\wedge$ Smart(Richard)
  $\vee$ At(Pikachu, CSE) $\wedge$ Smart(Pikachu)
  $\vee$ ...

Another common mistake to avoid

- Typically, $\wedge$ is the main connective with $\exists$
- **Common mistake**: using $\Rightarrow$ as the main connective with $\exists$:

$$\exists x \; \text{At}(x, \text{IIT}) \Rightarrow \text{Smart}(x)$$

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

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)

Fun with sentences

Brothers are siblings

$\forall x,y\ Brother(x,y) \Rightarrow Sibling(x,y)$

"Sibling" is symmetric

$\forall x,y\ Sibling(x,y) \Leftrightarrow Sibling(y,x)$

One's mother is one's female parent

$\forall m,c\ Mother(c) = m \Leftrightarrow (Female(m) \wedge Parent(m,c))$

First cousin is a child of a parent's sibling

$\forall$ x, y FirstCousin(x, y) $\Leftrightarrow$ $\exists$ p, ps Parent(p, x) $\wedge$ Sibling(ps, p) $\wedge$ Parent(ps, y)