Lecture 19

First Order Logic

Lusi Li

Department of Computer Science

ODU

Reading for This Class:

Chapter 8, Russell and Norvig

Review

- **Last Class**
  - **Inference**
    - **Resolution Algorithm**

- **This Class**
  - **First Order Logic**
  - **Start Homework 3**

- **Next Class**
  - **Learning**

Outline

- **Why first order logic?**
- **Syntax and semantics of first order logic**
- **Fun with sentences**

Outline

- **Why first order logic?**
- **Syntax and semantics of first order logic**
- **Fun with sentences**

Pros and Cons in Propositional Logic

- **Pro:** **Propositional logic is declarative:**
  - **pieces of syntax correspond to facts**
- **Pro:** **Propositional logic allows partial/disjunctive/negated information**
  - **(unlike most data structures and databases)**
- **Pro:** **Propositional logic is compositional:**
  - **meaning of $B_{1,1} \wedge P_{1,2}$ is derived from meaning of $B_{1,1}$ and that of $P_{1,2}$**
- **Pro:** **Meaning in propositional logic is context-independent**
  - **(unlike natural language, where meaning depends on context)**

- **Con:** **limited expressive power (unlike natural language)**
  - **Relationships among individuals: “Pits cause breezes in adjacent squares”, “Alice is a friend of Bob”**
  - **Generalizing patterns: “Every bear likes honey”, "All animals are living beings"**

First-Order Logic

- **Propositional logic: assume that world contains facts**
  - **A logical system for reasoning about facts**
- **First-order logic (FOL): assume that the world contains objects, relations, and functions**
  - **A logical system for reasoning about relations among objects**
  - **Objects:** people, houses, numbers, theories, Ronald McDonald, colors, baseball games, wars**, *. . .*** ***(nouns and noun phrases)***
  - **Relations: unary relations or properties** such as red, round, bogus, prime, multistoried *. . .*,

    or **n-ary relations** such as brother of, inside, part of, has color, occurred after, owns, comes between, *. . .* ***(verbs, verb phrases, adjective, and adverb)***
  - **Functions:** father of, best friend, third inning of, one more than, end of *. . .* ***(a mapping from objects to objects)***

  - ***E.g., "Squares neighboring the wumpus are smelly."***

  Objects: squares, wumpus; Relations: smelly (unary), neighboring (binary).

More 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, times	true/false/unknown
Probability theory	facts	degree of belief
Fuzzy logic	facts + degree of truth	known interval value

Higher-order logic:

relations and functions operate not only on objects, but also on relations and functions

Outline

- Why first order logic?
- Syntax and semantics of first order logic
- Fun with sentences

Syntax of First Order Logic

- **Symbols**

Constants	*KingJohn, 2, C,…*	Stand for objects
Predicates	*Brother, >, =, …*	Stand for relations
Functions	*Sqrt, LeftLegOf, …*	Stand for functions
Variables	*x, y, a, b, …*	
Connectives	$\wedge \quad \vee \quad \neg \quad \Rightarrow \quad \Leftrightarrow$	
Quantifiers	$\forall \quad \exists$	

Syntax of First Order Logic

- **Atomic sentences**

  predicate **(** $\boldsymbol{term_1, \ldots, term_n}$ **)**

  or

  $\boldsymbol{term_1 = term_2}$

- **Term (object)**

  constant ***//refers to a specified object***

  or

  variable ***//refers to an object without specifying a particular object; must have a quantifier in front of predicate(x)***

  or

  function **(** $\boldsymbol{term_1, \ldots, term_n}$ **)** **//take objects as input and produce objects as output.**

Syntax of First Order Logic

- **Atomic sentence examples**
- **KingJohn is a brother of RichardTheLionheart:**

$$\underbrace{\underbrace{Brother}_{\text{predicate}}\ (\ \underbrace{\underbrace{KingJohn}_{\text{constant}}}_{\text{term}},\ \underbrace{\underbrace{RichardTheLionheart}_{\text{constant}}}_{\text{term}}\ )}_{\text{atomic sentence}}$$

- **The length of the left leg of Richard is longer than the length of the left leg of KingJohn**

$$\underbrace{\underbrace{>}_{\text{predicate}}\quad (\ \underbrace{\underbrace{Length}_{\text{function}}(\underbrace{LeftLegOf}_{\text{function}}(\underbrace{Richard}_{\text{constant}}))}_{\text{term}},\ \underbrace{\underbrace{Length}_{\text{function}}(\underbrace{LeftLegOf}_{\text{function}}(\underbrace{KingJohn}_{\text{constant}}))}_{\text{term}}\ )}_{\text{atomic sentence}}$$

Syntax of First Order Logic

- **Complex sentences**
  - Built from atomic sentences using connectives as in propositional logic

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

Example:

If KingJohn is the sibling of Richard, then Richard is the sibling of KingJohn

$$\underbrace{\underbrace{\underbrace{Sibling}_{\text{predicate}}(\underbrace{KingJohn}_{\text{term}},\underbrace{Richard}_{\text{term}})}_{\text{atomic sentence}} \Rightarrow \underbrace{\underbrace{Sibling}_{\text{predicate}}(\underbrace{Richard}_{\text{term}},\underbrace{KingJohn}_{\text{term}})}_{\text{atomic sentence}}}_{\text{complex sentence}}$$

The language of First Order Logic

- **Terms**
  - A variable is a term
  - A constant symbol is a term
  - If f is an n-ary function symbol and t1,…, tn are n terms, then f(t1, …, tn) is a term
  - Nothing else is a term

- **Sentences**
  - True (False) is a sentence
  - If t1, t2 are terms, t1=t2 is a sentence
  - If p is an n-ary relation symbol and t1, …, tn are n terms, p(t1, …, tn) is a sentence
  - If φ is a sentence, $\neg \varphi$, $\forall x\ \varphi$, $\exists x\ \varphi$ are sentences
  - If $\varphi 1$, $\varphi 2$ are sentences, $\varphi 1 \Leftrightarrow \varphi 2$, $\varphi 1 \Rightarrow \varphi 2$, $\varphi 1 \wedge \varphi 2$, $\varphi 1 \vee \varphi 2$ are sentences
  - Nothing else is a sentence

Models of First-Order Logic

- **Models of first-order logic**

  Sentences are true or false with respect to models, which consist of

  – **a domain (also called universe) and an interpretation**

- **Domain: is a non-empty set of objects (domain elements); a world that our statement is situated within**
  - **Examples: natural numbers, people, animals, etc**
  - **Why is it important to specify a domain?**
    - **A statement can have different truth values in different domains.**

- **Interpretation**

  maps

  – **constant symbols to objects in the domain**

  – **predicate symbols to relations on those objects**

  – **function symbols to functions on those objects**

Semantics of First-Order Logic

- **Interprets**
  - constant symbols and variables as *objects*;

    Variables are placeholders for objects without specifying a particular object
  - functional symbols as *functions* from objects to objects;
  - relational symbols as *relations* over objects;
  - $=$ as equality (ie the *identity* relation);
  - the universal quantifier (essentially) as an *infinite conjunction*;

    $(\forall x\ Red(x) \equiv Red(Obj_1) \wedge Red(Obj_2) \wedge Red(Obj_3) \wedge Red(Obj_4) \wedge \cdots)$
  - the existential quantifier (essentially) as an *infinite disjunction*;

    $(\exists x\ Red(x) \equiv Red(Obj_1) \vee Red(Obj_2) \vee Red(Obj_3) \vee Red(Obj_4) \vee \cdots)$
  - **True**, **False**, $\wedge$, $\vee$, $\neg$, $\Rightarrow$, $\Leftrightarrow$ as in propositional logic.

Models of First-Order Logic: Example

Models for PL are just set of truth values for the propositional symbols.
Models for FOL contain a set of objects (domain elements) and relations among them.

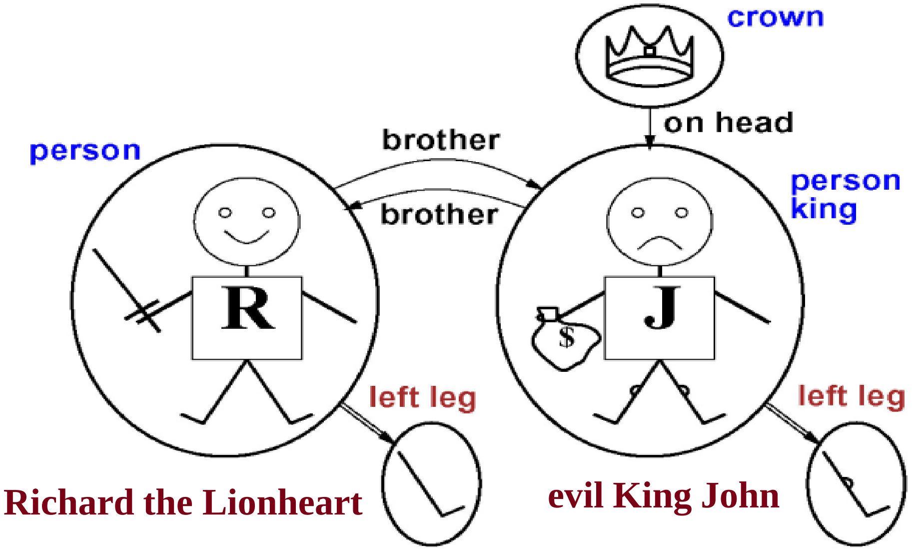
A model contains 5 objects, 2 binary relations, 3 unary relations, and 1 unary function.

5 objects: Richard the Lionheart; his younger brother, the evil King John; the left leg of Richard; the left leg of John; and a crown.

2 binary relations: Richard and John are brothers; the crown is on King John's head.

3 unary relations: the "person" property is true of both Richard and John; the "king" property is true only of John; the "crown" property is true only of the crown.

1 unary function: each person has one left leg.



Models of First-Order Logic: Example

Domain = {Richard the Lionheart, evil King John, the left leg of Richard, the left leg of John, crown}

One possible interpretation is defined:

Constants: *Richard* **refers to Richard the Lionheart**

John **refers to evil King John**

Predicates: *Person(x)* **refers to that x is a person**

King(x) **refers to that x is a king**

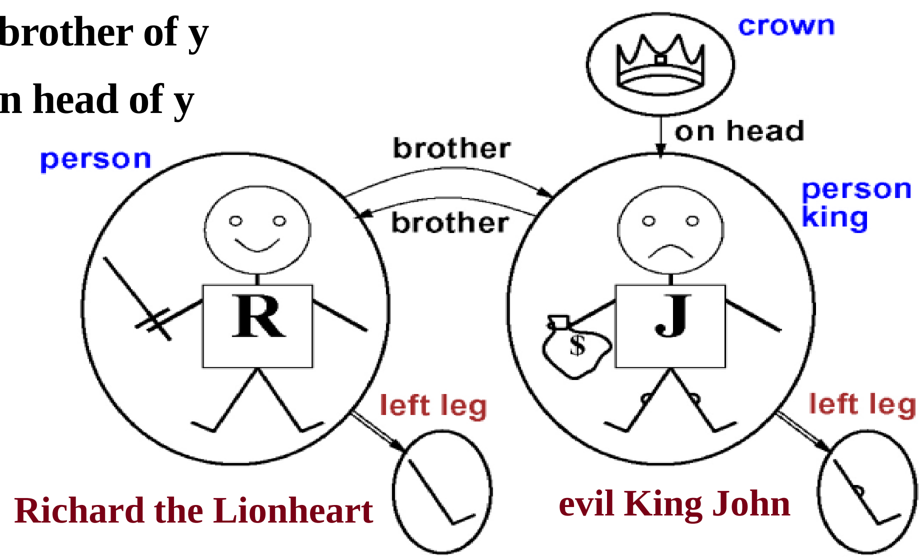
Crown(x) **refers to that x is a crown**

Brother(x,y) **refers to that x is a brother of y**

OnHead(x,y) **refers to that x is on head of y**

Functions:

LeftLegOf(x) **refers to the left leg of x**



Semantics of First-Order Logic

An atomic sentence

$$predicate\,(term_1, \ldots, term_n)$$

is true in a certain model (that consists of a domain and an interpretation) if and only if the objects referred to by $term_1, \ldots, term_n$ are in the relation referred to by *predicate*

An atomic sentence $term_1 = term_2$ is true under a given interpretation if and only if $term_1$ and $term_2$ have the same interpretation

The truth value of a complex sentence in a model is computed from the truth-values of its atomic sub-sentences in the same way as in propositional logic

Universal Quantification

Universal quantification makes statements about every object without naming them.

- **Syntax:** ∀ *(variables) (sentence)*

  If x is a variable, then ∀x is read as

  "For all x" or " For each x" or "For every x"

- **Example: "All kings are persons"**

- **Model:**
  Let the domain be the set of 5 objects.
  King(x) means that x is a king
  Person(x) means that x is a person

Domain = {Richard the Lionheart, evil King John, the left leg of Richard, the left leg of John, crown}

$$\underbrace{\forall \; x}_{\text{variables}} \quad \underbrace{King(x) \Rightarrow Person(x)}_{\text{sentence}}$$

Universal Quantification

- **Semantics**

$\forall x\ P(x)$ says that $P(x)$ is true for every object x in a model m, *or*
$\forall x\ P(x)$ is true in a model m iff $P(x)$ is true with x being each possible object in m.

Let $D = \{d_1, d_2, \ldots, d_n\}$ be the domain
$\forall x\ P(x)$ is equivalent to $(P(d_1) \wedge P(d_2) \wedge \ldots \wedge P(d_n))$

- **Example:** "All kings are persons"

$$\forall x\ King(x) \Rightarrow Person(x)$$

Domain = {Richard the Lionheart, evil King John, the left leg of Richard, the left leg of John, crown}

is equivalent to:

$$
\begin{aligned}
& (King(\text{Richard the Lionheart}) \Rightarrow Person(\text{Richard the Lionheart})) \\
\wedge\ & (King(\text{evil King John}) \Rightarrow Person(\text{evil King John})) \\
\wedge\ & (King(\text{the left leg of Richard}) \Rightarrow Person(\text{the left leg of Richard})) \\
\wedge\ & (King(\text{the left leg of John}) \Rightarrow Person(\text{the left leg of John})) \\
\wedge\ & (King(\text{crown}) \Rightarrow Person(\text{crown}))
\end{aligned}
$$

Universal Quantification

Note: $\Rightarrow$ **is the main connective with $\forall$ not $\wedge$**

Common mistake: **Using $\wedge$ as the main connective with $\forall$ will lead to an overly strong statement**

Correct: $\forall x\; King(x) \Rightarrow Person(x)$

"Everyone who is a king is a person"

Wrong: $\forall x\; (King(x) \wedge Person(x))$ **: "Everyone is a king and everyone is a person"**

$(King(\text{Richard the Lionheart}) \wedge Person(\text{Richard the Lionheart}))$

$\wedge \quad (King(\text{evil King John}) \wedge Person(\text{evil King John}))$

$\wedge \quad (King(\text{the left leg of Richard}) \wedge Person(\text{the left leg of Richard}))$

$\wedge \quad (King(\text{the left leg of John}) \wedge Person(\text{the left leg of John}))$

$\wedge \quad (King(\text{crown}) \wedge Person(\text{crown}))$

Existential Quantification

Existential quantification makes statements about *some* object without naming it.

- **Syntax:** $\exists$ *(variables) (sentence)*

  If x is a variable, then $\exists x$ is read as

  "There exists an x such that . . ." or "For some x . . ."

- **Example:** "The evil King John has a crown on his head"

- **Model:**
  Let the domain be the set of 5 objects.
  KingJohn means evil King John
  Crown(x) means that x is a crown
  OnHead(x,y) means that x is on head of y

Domain = {Richard the Lionheart, evil King John, the left leg of Richard, the left leg of John, crown}

$$\underbrace{\exists\ x}_{\text{variables}} \quad \underbrace{Crown(x) \land OnHead(x, KingJohn)}_{\text{sentence}}$$

Existential Quantification

- **Semantics**

$\exists x\ P(x)$ says that $P(x)$ is true for at least one object x, *or*

$\exists x\ P(x)$ is true in a model m iff $P(x)$ is true with x being some possible object in m

Let $D = \{d_1, d_2, \ldots, d_n\}$ be the domain

$\exists\ x\ P(x)$ is equivalent to $(P(d_1) \lor P(d_2) \lor \ldots \lor P(d_n))$

- **Example:** $\exists x\ \mathit{Crown}(x) \land \mathit{OnHead}(x, \mathit{KingJohn})$

Domain = {Richard the Lionheart, evil King John, the left leg of Richard, the left leg of John, crown}

is equivalent to:

(*Crown*(Richard the Lionheart) ∧ *OnHead*(Richard the Lionheart, *KingJohn*))

∨ (*Crown*(evil King John) ∧ *OnHead*(evil King John, *KingJohn*))

∨ (*Crown*(the left leg of Richard) ∧ *OnHead*(the left leg of Richard , *KingJohn*))

∨ (*Crown*(the left leg of John) ∧ *OnHead*(the left leg of John, *KingJohn*))

∨ (*Crown*(crown) ∧ *OnHead*(crown, *KingJohn*))

Existential Quantification

Note: ∧ is the main connective with ∃ not ⇒

Common mistake: Using ⇒ as the main connective with ∃ will lead to a very week statement !

Correct: $\exists x \; Crown(x) \land OnHead(x, KingJohn)$

"There is something which is a crown and it is on head of KingJohn"

Wrong: $\exists x \; (Crown(x) \Rightarrow OnHead(x, KingJohn))$

"There is something which, if it is a crown, is on the head of KingJohn"

This is true if there is anything that is not a crown

This is true if there is anything that is on head of KingJohn

Properties of Quantifiers

Quantifiers of same type commute

$\forall x \forall y$ **is the same as** $\forall y \forall x$

$\exists x \exists y$ **is the same as** $\exists y \exists x$

Quantifiers of different type do NOT commute

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

A quantifier holds over everything to the right of it. They have different scopes.

Example:

Let the domain be the set of persons in the world and $Loves(x,y)$ means x loves y

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

"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"

Properties of Quantifiers

Quantifier duality:

$\forall x \; Likes(x, IceCream)$ **is the same as** $\neg\exists x \; \neg Likes(x, IceCream)$

$\exists x \; Likes(x, Broccoli)$ **is the same as** $\neg\forall x \; \neg Likes(x, Broccoli)$

$\forall$ is really a conjunction over the universe of objects and
$\exists$ is a disjunction

Obey De Morgan rules

$$
\begin{array}{lll}
\forall x \;\; \neg P & \equiv & \neg\exists x \;\; P \\
\neg\forall x \;\; P & \equiv & \exists x \;\; \neg P \\
\forall x \;\; P & \equiv & \neg\exists x \;\; \neg P \\
\exists x \;\; P & \equiv & \neg\forall x \;\; \neg P
\end{array}
\qquad\qquad
\begin{array}{lll}
\neg(P \vee Q) & \equiv & \neg P \wedge \neg Q \\
\neg(P \wedge Q) & \equiv & \neg P \vee \neg Q \\
P \wedge Q & \equiv & \neg(\neg P \vee \neg Q) \\
P \vee Q & \equiv & \neg(\neg P \wedge \neg Q)
\end{array}
$$

Equality

Semantics

$term_1 = term_2$ **is true under a given interpretation if and only if**

$term_1$ **and** $term_2$ **have the same interpretation**

Equality can only be applied to objects;

to state that two propositions are equal, use ⇔

Example: **Father (John) = Henry**

∃ x, y Brother (x,Richard) ∧ Brother (y,Richard) ∧ ¬(x=y)

// Richard has at least two brothers

A Summary of Translation Idioms

- **Every/All/Each/Any**
  - $\forall x$
- **Some/At least one/There exists a/There is a**
  - $\exists x$
- **None/No x**
  - $\neg(\exists x \ldots)$
- **Not every/Not all**
  - $\neg(\forall x \ldots)$

Outline

- Why first order logic?
- Syntax and semantics of first order logic
- Fun with sentences

Fun with Sentences

Let the domain be the set of animals.

Honey(x) means that x likes honey.

Translate the following sentences into predicate logic.

1. All animals like honey. $\forall x\ Honey(x)$
2. At least one animal likes honey. $\exists x\ Honey(x)$
3. Not every animal likes honey. $\neg(\forall x\ Honey(x))$
4. No animal likes honey. $\neg(\exists x\ Honey(x))$

5. No animal dislikes honey. $\neg(\exists x\ (\neg Honey(x)))$
6. Not every animal dislikes honey. $\neg(\forall x\ (\neg Honey(x)))$
7. Some animal dislikes honey. $\exists x\ (\neg Honey(x))$
8. Every animal dislikes honey. $\forall x\ (\neg Honey(x))$

Note: Each pair of sentences (1 & 5, 2 & 6, 3 & 7, 4 & 8) is logically equivalent. However, when doing translations, always give the direct translation to avoid losing marks

Summary

- **First Order Logic**
  - **Syntax**
  - **Semantics**
- **Expressions in First Order Logic**

What I want you to do

- Review Chapter 8
- Work on your Homework 3