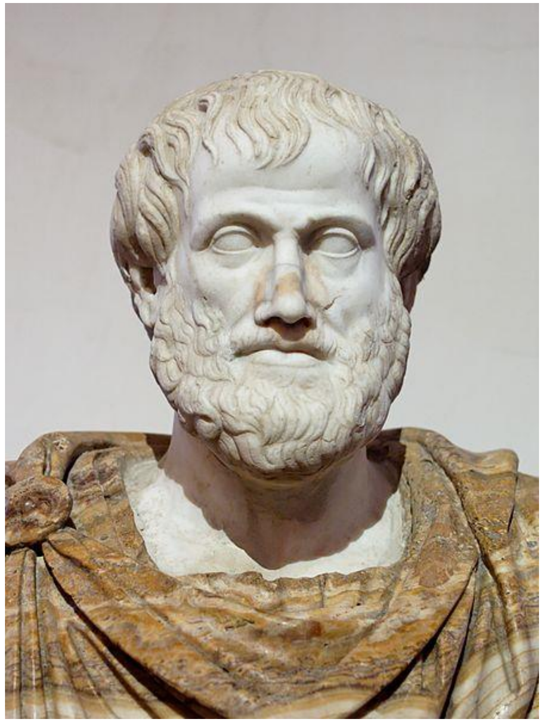
First Order Logic



Pros and cons of propositional logic

- Propositional logic is **declarative**
- Propositional logic allows partial/disjunctive/negated information
  - (unlike most data structures and databases)
- 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
  - (unlike natural language)
  - E.g., cannot say "pits cause breezes in adjacent squares“
    - except by writing one sentence for each square

First-Order Logic

- Propositional logic only deals with ‘facts’ – statements that may or may not be true of the world, e.g., *“It is raining”*.

- First-order logic, 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,
    - (relations in which there is only one value for a given input)

Grammar Predicate Logic

Sentence	$\rightarrow$ *AtomicSentence*		
	| *(Sentence Connective Sentence)*		
	| *Quantifier Variable, ... Sentence*		
	| $\neg$ *Sentence*		
AtomicSentence	$\rightarrow$ *Predicate(Term, …)*	| *Term = Term*	
Term	$\rightarrow$ *Function(Term, …)*	| *Constant*	| *Variable*
Connective	$\rightarrow \wedge \mid \vee \mid \Rightarrow \mid \Leftrightarrow$		
Quantifier	$\rightarrow \forall \mid \exists$		
Constant	$\rightarrow$ *A, B, C,* X_1, X_2, *Jim, Jack*		
Variable	$\rightarrow$ *a, b, c,* x_1, x_2, *counter, position*		
Predicate	$\rightarrow$ *Adjacent-To, Younger-Than,*		
Function	$\rightarrow$ *Father-Of, Square-Position, Sqrt, Cosine*		

Syntax of FOL: Basic elements

- Constant Symbols:
  - Stand for objects
  - e.g., KingJohn, 2, UCI,...
- Predicate Symbols
  - Stand for relations
  - e.g., Brother(Richard, John), greater_than(3,2)...
- Function Symbols
  - Stand for functions
  - E.g., Sqrt(3), LeftLegOf(John),...

Syntax of FOL: Basic Elements

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

Complex Sentences

- Complex sentences are made from atomic sentences using connectives

$$\neg S, \qquad S_1 \land S_2, \qquad S_1 \lor S_2, \qquad S_1 \Rightarrow S_2, \qquad S_1 \Leftrightarrow S_2$$

- For example,
  - $Sibling(KingJohn, Richard) \Rightarrow Sibling(Richard, KingJohn)$
  - $King(Richard) \lor King(John)$
  - $King(John) \Rightarrow \neg King(Richard)$
  - $LessThan(Plus(1,2)\,, 4)\ \land GreaterThan(1,2)$
  - $> (1,2) \lor \leq (1,2)$
  - $> (1,2) \land \neg > (1,2)$

Variables

- *Person(John)* is true or false because we give it a single argument "*John*"
- We can be much more flexible if we allow variables which can take on values in a domain. e.g., all persons x, all integers i, etc.
- E.g., can state rules like
  - $Person(x) \Rightarrow HasHead(x)$
  - $Integer(i) \Rightarrow Integer(plus(i, 1))$

- Stephen was intelligent
- Lucy is a professor.
- Lucy criticized John
- Bill is a student
- Bill takes Analysis and Geometry

Universal Quantification

- $\forall$ means *"for all" ; "for every"*
  - $\forall\langle varaible\rangle\ \langle sentence\rangle$
- Allows us to make statements about all objects that have certain properties
- Roughly speaking, equivalent to the **conjunction** of **instantiations** of P

- Example;
  - Everybody at UCP is smart: $\forall x.\quad At(x, UCP) \Longrightarrow Smart(x)$
  - $\forall x\ \ King(x) \Longrightarrow Person(x)$

- **Common Mistake:** using $\wedge$ as the main connective with $\forall$:
  - $\forall x\ \ King(x) \wedge Person(x)$ This is **NOT CORRECT**
  - This would imply that all x are Kings and a People.

Existential Quantification

- ∃x means "there exists an x such that…." (at least one object x)
- Allows us to make statements about some object without naming it

- Examples:
  - $\exists x\ King(x)$
  - $\exists x\ Lives\_in(John, Castle(x))$
  - $\exists i\ \ Integer(i) \wedge\ GreaterThan(i, 0)$

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

Example

- For all real x, $x > 2$, imples $x > 3$.
  - $\forall x \; [(x > 2) \Longrightarrow (x > 3)] \quad x \in R$
- There exists some real x whose square is minus 1.
  - $\exists x \; [(x^2 = -1)] \quad x \in R$

- Brothers are siblings
- One’s mother is one’s female parent
- A first cousin is a child of parent’s sibling

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
  - $\forall x\ Likes(x,IceCream)$ $\neg\exists x\ \neg Likes(x,IceCream)$
  - $\exists x\ Likes(x,Broccoli)$ $\neg\forall x\ \neg Likes(x,Broccoli)$

Example

- Brothers are siblings
  - $\forall x, y \;\; Brother(x, y) \Longrightarrow Sibing(x, y)$
- One's mother is one's female parent
  - $\forall x, y \;\; Mother(x, y) \Leftrightarrow \big(Female(x) \;\; \wedge \; Parent(x, y)\big)$
- A first cousin is a child of a parent sibling
  - $\forall x, y \; FirstCousin(x, y) \Leftrightarrow$
    $\exists p, ps \; Parent(p, x) \wedge Sibling(p, ps) \wedge Parent(ps, y)$

Writing FOL

- Cats are mammals
- Jane is a tall surveyor
- A nephew is a sibling’s son
- A maternal grandmother is a mother’s mother
- Everybody likes food
- Nobody loves Jane
- Everybody has a father
- Everybody has a father and a mother
- Whoever has a father, has a mother

Writing FOL

- Cats are mammals $[Cat, Mammal]$
  - $\forall\, x.\ Cat(x)\ \rightarrow\ Mammal(x)$
- Jane is a tall surveyor $[Tall, Surveyor, Jane]$
  - $Tall(Jane)\ \wedge\ Surveyor(Jane)$
- A nephew is a sibling’s son $[Nephew, Sibling, Son]$
  - $\forall\, xy.\left[Nephew(x,y) \leftrightarrow \exists z.\ \ [Sibling(y,z)\ \ \wedge\ \ Son(x,z)]\right]$
- A maternal grandmother is a mother’s mother $[functions{:}\, mgm, mother\_of]$
  - $\forall xy.\ \ x = mgm(y) \leftrightarrow \exists z.\ x = mother\_of(z)\ \ \wedge\ \ z = mother\_of(y)$
- Everybody likes food $[loves]$
  - $\forall x. \exists y.\ \ \ Loves(x,y)$
  - $\exists y. \forall x.\ \ \ Loves(x,y)$

Writing FOL

- Nobody loves Jane
  - $\forall\, x.\ \ \neg\, Loves(x, Jane)$
  - $\neg\exists x.\ \ Loves(x, Jane)$
- Everybody has a father
  - $\forall x.\ \ \exists y.\ \ \ Father(y, x)$
- Everybody has a father and a mother
  - $\forall x.\ \ \exists y.\ \ \ Father(y, x)\ \ \ \wedge\ \ \ Mother(z, x)$
- Whoever has a father, has a mother
  - $\forall x.\ \big[[\exists\, y.\ Father(y, x)] \rightarrow [\exists\, y.\ Mother(y, x)]\big]$

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*:
-

$\forall x,y\ Sibling(x,y) \Leftrightarrow [\neg(x = y) \land \exists m,f\ \neg(m = f) \land Parent(m,x) \land Parent(f,x) \land Parent(m,y) \land Parent(f,y)]$

Representing facts with First-Order Logic

- Irfan is a professor
- All professors are persons
- Salman is the dean
- Deans are professors
- All professors consider dean a friend or don’t know him
- Everyone is a friend of someone
- People only criticize people that are not their friends
- Irfan criticized Salman

Same example, more formally

- isProf (Irfan)
  - Irfan is a professor
- $\forall x(\text{isProf}(\text{x}) \rightarrow \text{isPerson}(\text{x}))$
  - All professors are persons
- isDean(Salman)
  - Salman is the dean
- $\forall x(\text{isDean}(\text{x}) \rightarrow \text{isProf}(\text{x}))$
  - Deans are professors
- $\forall x \forall y\left(isProf(x) \wedge isDean(y) \rightarrow friendOf(x, y) \vee \neg knows\,(x, y)\right)$
  - All professors consider dean a friend or don’t know him

Same example, more formally

- $\forall x \exists y\ (friendOf(x, y))$
  - Everyone is a friend of someone
- $\forall x \forall y (isPerson(x) \land isPerson(y) \land criticise(x, y) \rightarrow \neg friendOf(x, y))$
  - People only criticize people that are not their friends
- $criticize(Irfan, Salman)$
  - Irfan criticized Salman
- Question: is Salman not friend of Irfan?
  - $\neg freindOf(Salman, Irfan)$

How Machine “sees” it

Knowledge Base:

- P1(A)
- $\forall x(\mathrm{P1}(\mathrm{x}) \rightarrow \mathrm{P3}(\mathrm{x}))$
- P4(B)
- $\forall x(\mathrm{P4}(\mathrm{x}) \rightarrow P1(\mathrm{x}))$
- $\forall x \forall y\ (\big(P1(x) \wedge P4(y) \rightarrow P2(x,y)\big) \vee \neg P5(x,y))$
- $\forall x \exists y\ (P2(x,y)$
- $\forall x \forall y(P3(x) \wedge P3(y) \wedge P6(x,y) \rightarrow \neg P2(x,y))$
- $P6(A,B)$
- **Question:** $\boldsymbol{\neg P2(B,A)}$**?**

$Irfan = A$
$Salman = B$
$isProf(x) = P1(x)$
$friendOf(x) = P2(x)$
$isPerson(x) = P3(x)$
$isDean(x) = P4(x)$
$knows(x,y) = P5(x,y)$
$criticize(x,y) = P6(x,y)$

Knowledge Engineering

1. Identify the task.
2. Assemble the relevant knowledge.
3. Decide on vocabulary of predicates, functions, and constants.
4. Encode general knowledge about the domain.
5. Encode a description of the specific problem instance.
6. Pose queries to the inference procedure and get answers.
7. Debug the knowledge base.

Knowledge Engineering

1. All professors are persons **General Knowledge**
2. Deans are professors
3. All professors consider dean a friend or don't know him
4. Everyone is a friend of someone

5. Irfan is a professor **Specific Problem**
6. People only criticize people that are not their friends
7. Salman is the dean
8. Irfan criticized Salman

9. Is Salman not friend of Irfan? **Query**

Using FOL for KB

- We want to **<u>TELL</u>** things to the KB, e.g.
  - $TELL(KB, \forall x \;\; King(x) \Longrightarrow Person(x))$
  - $TELL(KB, King(John))$
- These sentences are assertions

- We also want to **<u>ASK</u>** things to the KB,
  - $ASK(KB, \;\; \exists x \;\; Person(x))$
- these are queries or goals
- The KB should $Person(x)$ is true: $\{x/John, x/Richard, \ldots\}$

FOL Version of Wumpus World

- **Percept** sentence:
  - $Percept([Stench, Breeze, Glitter, None, None], 5)$
- **Actions**:
  - Turn(Right), Turn(Left), Forward, Shoot, Grab, Release, Climb
- To determine **best action**, construct query:
  - $\forall a, BestAction(\, a, 5)$
- ASK solves this and returns {a/Grab}
  - And TELL about the action.

Simplifying the percept and deciding actions

$$\forall b, g, t \;\; Percept([Stench, b, g], t) \Rightarrow Stench(t)$$

$$\forall s, g, t \;\; Percept([s, Breeze, g], t) \Rightarrow Breeze(t)$$

$$\forall s, b, t \;\; Percept([s, b, Glitter], t) \Rightarrow AtGold(t)$$

- Simple Reflex Agent

$$\forall t \;\; AtGold(t) \Rightarrow Action(Grab, t)$$

- Agent Keeping Track of the World

$$\forall t \;\; AtGold(t) \wedge \neg Holding(Gold, t) \Rightarrow Action(Grab, t)$$

Question

- Using the predicate symbols shown and appropriate quantifiers, write each English language statement as a predicate well formed formula. (The domain is the whole world.)

B(x) is “x is a ball."

R(x) is “x is round."

S(x) is “x is a soccer ball."

- (a) All balls are round.
- (b) Not all balls are soccer balls.
- (c) All soccer balls are round.
- (d) Some balls are not round.
- (e) Some balls are round, but soccer balls are not.