FIRST-ORDER LOGIC

1

- Does not have any variables
- Cannot reason about a group of objects
- Cannot handle a domain with infinite objects
- We need
  - Predicates
  - Quantifiers

2

FOL

- Combining the best of formal and natural languagesCombining the best of formal and natural languages
  - Objects
  - Relations
  - Finctions
  - "One plus two equals three."
    - Objects: one, two, three, one plus two;
    - Relation: equals;
    - Function: plus.
    - ("One plus two" is a name for the object that is obtained by applying the function "plus" to the objects "one" and "two." "Three" is another name for this object.)
  - "Squares neighboring the wumpus are smelly."
    - Objects: wumpus, squares; Property: smelly;
    - Relation: neighboring
  - "Evil King John ruled England in 1200."
    - Objects: John, England, 1200; Properties: evil, king
    - Relation: ruled;

3

Sentence → AtomicSentence | ComplexSentence

AtomicSentence → Predicate | Predicate(Term,...) | Term = Term

ComplexSentence → (Sentence) | [Sentence] | ¬Sentence
| Sentence ∧ Sentence
| Sentence ∨ Sentence
| Sentence ⇒ Sentence
| Sentence ⇔ Sentence
| Quantifier Variable,... Sentence

Term → Function(Term,...) | Constant | Variable

Quantifier → ∀| ∃

Constant → A| X1 | John | ···

Variable → a| x| s| ···

Predicate → True| False | After | Loves | Raining | ···

Function → Mother | LeftLeg | ···

OPERATOR PRECEDENCE : ¬,=,∧,∨,⇒,⇔

The syntax of first-order logic with equality, specified in Backus–Naur form (see page 1060 if you are not familiar with this notation). Operator precedences are specified, from highest to lowest. The precedence of quantifiers is such that a quantifier holds over everything to the right of it.

4

P(x)

- P is the name of predicate
- x is the variable and the argument of P
- With n variables the predicate p has arity n

- It has no truth values by itself
  - P1(x) : 3+x = 8
  - You can not associate any truth value with the above predicate without defining the domain of x

5

UNIVERSAL QUANTIFIER

6

- All men are mortal
  - ∀x man(x) ⇒ mortal(x)
- All cats are mammals
  - ∀x cat(x) ⇒ mammal(x)
- Universal Quantifier is good for writing general rules with universal quantifiers.
- ∀x (x+3=8) : This is not true by the rules of number theory if x is a member of the set of whole numbers
- ∀x (x+1>x): This is true for all x where x is an integer

6

Terms

•**Definition:** A term is a logical expression referring to an object. It can be a constant symbol or a function symbol like LeftLeg(John).

•**Examples:**

- •John (constant symbol)
- •LeftLeg(John) (function symbol referring to King John's left leg)

Atomic Sentences

•**Definition:** An atomic sentence is formed by a predicate symbol followed by a list of terms, stating facts about objects.

•**Examples:**

- •Brother(Richard, John) states that Richard the Lionheart is King John's brother.
- •Married(Father(Richard), Mother(John)) states Richard's father is married to John's mother.

7

7

Complex Sentences

•**Definition:** Complex sentences are constructed using logical connectives, similar to propositional calculus.

•**Examples:**

- •¬Brother(LeftLeg(Richard), John)
  - •states that Richard's left leg is not John's brother.
- •Brother(Richard, John) ∧ Brother(John, Richard)
  - •implies the brotherhood is mutual.

8

8

Universal and Existantial Quantifiers

•**Definition:** **Universal quantifiers** express that a property holds for all objects in a domain.

•**Examples:**

- •∀x King(x) ⇒ Person(x)
  - •states that if x is a king, then x is a person.
  - •Applied to: Richard the Lionheart, King John, and others.

•**Definition:** **Existential quantifiers** assert that there is at least one object in the domain for which a property holds.

•**Examples:**

- •∃x Crown(x) ∧ OnHead(x, John)
  - •states that there exists a crown on John's head.
  - •The crown could be any object satisfying the conditions.

9

9

EXISTENTIAL QUNATIFIER

Someone from Pakistan had won the Nobel prize

- ∃x pakistani(x) ∧ wonNobelPrize(x)

Some of the students in the AI class got an A.

- ∃x studentOfAI(x) ∧ gotAnA(x)

Existential quantifier is true if at least one element x from the specified domain satisfies the statement.

∃x (x+3=5) is true

∃x (x+1 < x) is false

10

10

SEMANTICS

∀x p(x) is equivalent to p(a1) ^ p(a2) … p(an)

- true for all objects in the domain of discourse D

∀x p(x) is true iff p(x) is true for ALL x in D

∃x p(x) is equivalent to p(a1) v p(a2) … p(an)

- Only a particular p(x) has to be true if the entire propositional statement is true

∃x p(x) is true iff p(x) is true for SOME x in D

11

11

Relation BETWEEN ∃ & ∀

Everyone dislikes Ahmad:

- ∀x ¬Likes (x, Ahmad)
- ¬ ∃x Likes (x, Ahmad)

Everyone likes Ahmad:

- ∀x Likes (x, Ahmad)
- ¬ ∃x ¬ Likes (x, Ahmad)

Rules

∀x ¬P(x) ≡ ¬ ∃xP(x)	**¬ P ∧¬ Q ≡ ¬ (P∨Q)**
¬∀x P(x) ≡ ∃x ¬ P(x)	**¬ (P ∧ Q) ≡ ¬ P∨¬ Q**
∀x P(x) ≡ ¬ ∃x ¬ P(x)	**(P ∧ Q) ≡ ¬(¬ P∨¬ Q)**
∃x P(x) ≡ ¬∀x ¬ P(x)	**(P ∨ Q) ≡ ¬(¬ P∧¬ Q)**

12

12

KR IN FIRST ORDER LOGIC

You have to perform the following tasks for representing the knowledge by first order logic:

- Identify the objects
- Establish relationships among those identified objects
- Encode the knowledge about these objects using relations in the form of rules
  - Grandfather(X,Y) <- father (X,Z) ^ father (Z,Y)

Inference engine then can use that knowledge to infer new facts and knowledge.

13

13

KR IN FIRST ORDER LOGIC EXAMPLES

- Mother is a female parent:
  - ∀x ∃y female(y) ∧ parent(y,x) ⇔ mother(y,x)
- Husband is male spouse
  - ∀w,h husband(h,w) ⇔ Male(h) ∧ spouse(h,w)
- Disjoint categories e.g. male, female
  - ∀x male(x) ⇔ ¬ Female(x)
- Grandparent relation
  - ∀g,c grandparent(g,c) ⇔ ∃p Parent(g,p) ∧ Parent(p,c)
- Sibling relation
  - ∀x,y Sibling(x,y) ⇔ x≠y ∧ ∃p Parent(p,x) ∧ Parent(p,y)

14

14

SKOLEMIZATION

- The process of removing existential quantifiers
- Example: someone likes Ahmad
  - ∃x Likes (x, Ahmad)
- Get rid of existential quantifier by replacing x with a constant(called skolem constant) that doesn't exist in the KB
  - ∃x Likes (Const1, Ahmad) {x/Const1}
- For complex examples you may have to use Skolem function. Consider this

$$\forall x[\exists y[mother(y,x)]]$$

  - If just replace the existential quantification with a constant we get

$$\forall x[mother(M302,x)]]$$

- Which says M302 is everyone's mother.In this case we have to use a Skolem function rather than Skolem constant

$$\forall x[mother(f1(x),x)]]$$

15

15

Substitution

Substitutions are also known as bindings

A substitution is a list of variable-constant bindings of the form

- θ = {$var_1/val_1, v_2/g_2$,...} where **var$_i$** are variables and **val$_i$** is the value substituted for **var$_i$**
- If there is a clause C then Cθ is the atom we get when all occurrences of variables in θ are replaced by their ground terms
  - Example: C = parent(X,Y), θ= {X/Ali,Y/Akram},Cθ=father(Ali,Akram)

16

16

UNIFICATION

It is the process or algorithm for determining substitutions so that we can make two predicates match

- Example: unify(father(X,Akram),father(Ali,Akram))={X/Ali}

FORMAL DEFINITION

- Unify(C,D)= θ so that Cθ = Dθ
- If no such θ exists then fail

RULES: θ must satisfy the following:

- No variable is bound to two different values θ={X/Ann,X/Bill} is invalid
- No variable in θ is bound to a term that contains the variable itself θ = {X/f(X)} is invalid

In unification you can do the following:

- replace variable by constant
- replace variable by variable
- replace a variable by a function expression

17

17

UNIFICATION EXAMPLES

Unify(f(a,X), f(a,b))
= {X/b}

Unify(f(a,X),f(Y,b))
= {Y/a, X/b}

Unify(f(a,b),f(a,b))
= {}

Unify(X,Y)
= {X/Y}

Unify(f(a,X),f(a,g(Y)))
= {X/g(Y)}

Unify(f(a,X),f(X,b))
= fail (X is bound to a & b)

Unify(f(a,b),f(a,c)) a,b,c are constants
= fail (no match was found)

Unify(f(a,b),g(a,X))
= fail (different predicates)

Unify(f(a,X),f(a,g(X)))
= fail(X is bound to a term that contains itself)

18

18

Example

- hasWings(X) ⇒ flies(X)
- hasWings(Tweety)

- θ = Unify (has_wings(X),has_wings(Tweety)) = {X/Tweety}

- So we can conclude that:
  - flies(Tweety)