First Order Logic

Tushar B. Kute,
http://tusharkute.com

Introduction

- In the topic of Propositional logic, we have seen that how to represent statements using propositional logic.
- But unfortunately, in propositional logic, we can only represent the facts, which are either true or false.
- PL is not sufficient to represent the complex sentences or natural language statements.
- The propositional logic has very limited expressive power.

First Order Logic

- Consider the following sentence, which we cannot represent using PL logic.
  - "Some humans are intelligent", or
  - "Sachin likes cricket."
- To represent the above statements, PL logic is not sufficient, so we required some more powerful logic, such as first-order logic.

First Order Logic

- First-order logic is another way of knowledge representation in artificial intelligence. It is an extension to propositional logic.
- FOL is sufficiently expressive to represent the natural language statements in a concise way.
- First-order logic is also known as Predicate logic or First-order predicate logic.
- First-order logic is a powerful language that develops information about the objects in a more easy way and can also express the relationship between those objects.

First Order Logic

- First-order logic (like natural language) does not only assume that the world contains facts like propositional logic but also assumes the following things in the world:
  - Objects: A, B, people, numbers, colors, wars, theories, squares, pits, wumpus, ...
  - Relations: It can be unary relation such as: red, round, is adjacent, or n-any relation such as: the sister of, brother of, has color, comes between
  - Function: Father of, best friend, third inning of, end of, ...
- As a natural language, first-order logic also has two main parts:
  - Syntax
  - Semantics

First Order Logic

- Following are the basic elements of FOL syntax:
  - Constant 1, 2, A, John, Mumbai, cat,....
  - Variables x, y, z, a, b,....
  - Predicates Brother, Father, >,....
  - Function sqrt, LeftLegOf, ...
  - Connectives ∧, ∨, ¬, ⇒, ⇔
  - Equality ==
  - Quantifier ∀, ∃

First Order Logic

- Atomic sentences:
  - Atomic sentences are the most basic sentences of first-order logic. These sentences are formed from a predicate symbol followed by a parenthesis with a sequence of terms.
  - We can represent atomic sentences as Predicate (term1, term2, ..., term n).
  - Example: Ravi and Ajay are brothers: => Brothers(Ravi, Ajay).

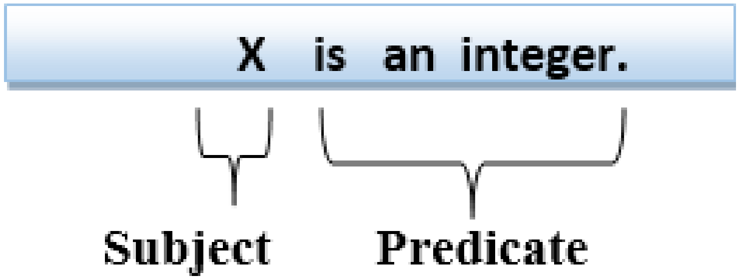
    Chinky is a cat: => cat (Chinky).

First Order Logic

- Complex Sentences:
  - Complex sentences are made by combining atomic sentences using connectives.
- First-order logic statements can be divided into two parts:
  - Subject: Subject is the main part of the statement.
  - Predicate: A predicate can be defined as a relation, which binds two atoms together in a statement.

First Order Logic

- Consider the statement: "x is an integer.", it consists of two parts, the first part x is the subject of the statement and second part "is an integer," is known as a predicate.



Quantifiers in First Order Logic

- A quantifier is a language element which generates quantification, and quantification specifies the quantity of specimen in the universe of discourse.
- These are the symbols that permit to determine or identify the range and scope of the variable in the logical expression. There are two types of quantifier:
  - Universal Quantifier, (for all, everyone, everything)
  - Existential quantifier, (for some, at least one).

Universal Quantifier

- Universal quantifier is a symbol of logical representation, which specifies that the statement within its range is true for everything or every instance of a particular thing.
- The Universal quantifier is represented by a symbol ∀, which resembles an inverted A.
- Note: In universal quantifier we use implication "→".
- If x is a variable, then ∀x is read as:

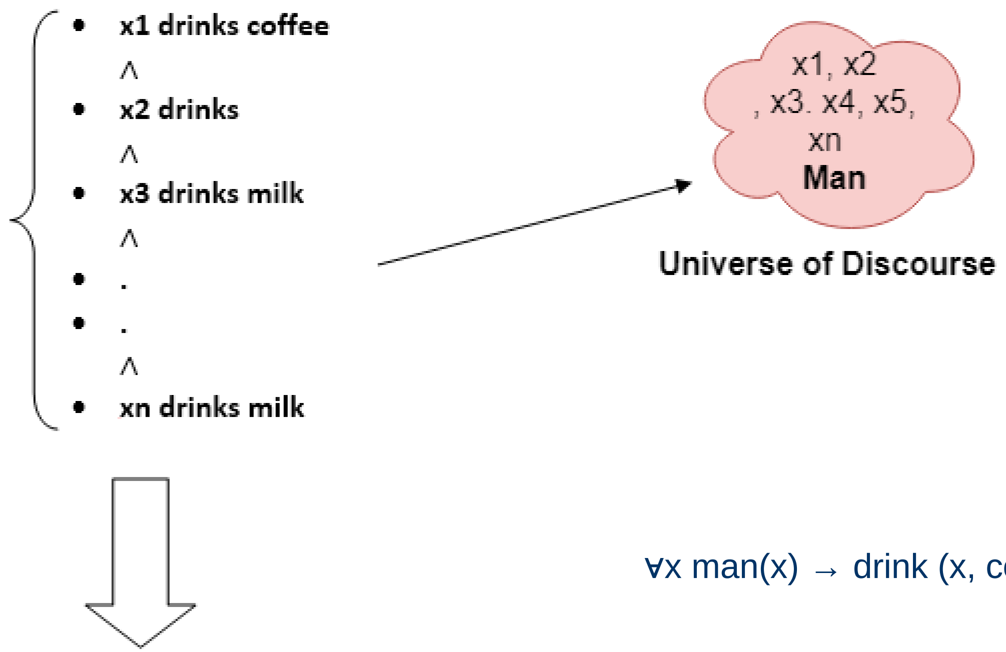
  For all x

  For each x

  For every x.

Universal Quantifier

- Example: All man drink coffee.
- Let a variable x which refers to a cat so all x can be represented in UOD as below:



So in shorthand notation, we can write it as :

$\forall x$ man(x) $\rightarrow$ drink (x, coffee).

It will be read as: There are all x where x is a man who drink coffee.

Existential Quantifier

- Existential quantifiers are the type of quantifiers, which express that the statement within its scope is true for at least one instance of something.
- It is denoted by the logical operator ∃, which resembles as inverted E. When it is used with a predicate variable then it is called as an existential quantifier.
- Note: In Existential quantifier we always use AND or Conjunction symbol (∧).
- If x is a variable, then existential quantifier will be ∃x or ∃(x). And it will be read as:

  There exists a 'x.'

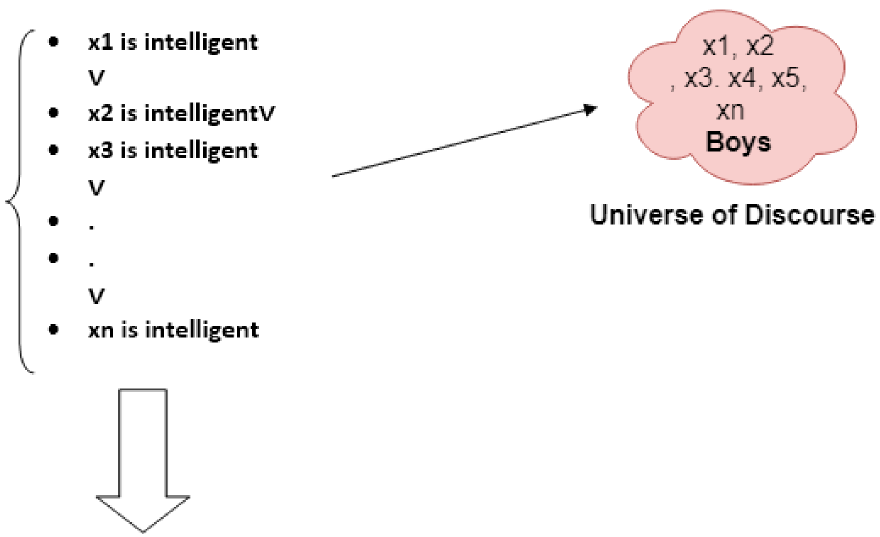
  For some 'x.'

  For at least one 'x.'

Existential Quantifier

- Example:

  Some boys are intelligent.



So in short-hand notation, we can write it as:

$\exists x$: boys(x) $\wedge$ intelligent(x)

- It will be read as: There are some x where x is a boy who is intelligent.

Points

- Points to remember:
  - The main connective for universal quantifier ∀ is implication →.
  - The main connective for existential quantifier ∃ is and ∧.
- Properties of Quantifiers:
  - In universal quantifier, ∀x∀y is similar to ∀y∀x.
  - In Existential quantifier, ∃x∃y is similar to ∃y∃x.
  - ∃x∀y is not similar to ∀y∃x.

Examples

- 1. All birds fly.

  In this question the predicate is "fly(bird)."

  And since there are all birds who fly so it will be represented as follows.

  $$\forall x \ bird(x) \rightarrow fly(x).$$

- 2. Every man respects his parent.

  In this question, the predicate is "respect(x, y)," where x=man, and y= parent.

  Since there is every man so will use ∀, and it will be represented as follows:

  $$\forall x \ man(x) \rightarrow respects\ (x, parent).$$

Examples

- 3. Some boys play cricket.

  In this question, the predicate is "play(x, y)," where x= boys, and y= game. Since there are some boys so we will use ∃, and it will be represented as:

  $$\exists x \; boys(x) \rightarrow play(x, cricket).$$

- 4. Not all students like both Mathematics and Science.

  In this question, the predicate is "like(x, y)," where x= student, and y= subject.

  Since there are not all students, so we will use ∀ with negation, so following representation for this:

  $$\neg\forall (x) \; [\, student(x) \rightarrow like(x, Mathematics) \wedge like(x, Science)].$$

Examples

- 5. Only one student failed in Mathematics.

  In this question, the predicate is "failed(x, y)," where x= student, and y= subject.

  Since there is only one student who failed in Mathematics, so we will use following representation for this:

  $$\exists(x)\ [\ \text{student}(x) \rightarrow \text{failed}\ (x, \text{Mathematics})\ \wedge \forall\ (y)\ [\neg(x==y)\ \wedge\ \text{student}(y) \rightarrow \neg\text{failed}\ (x, \text{Mathematics})].$$

Free and Bound Variables

- The quantifiers interact with variables which appear in a suitable way. There are two types of variables in First-order logic which are given below:
- Free Variable: A variable is said to be a free variable in a formula if it occurs outside the scope of the quantifier.

  Example: $\forall x\ \exists(y)[P\ (x, y, z)]$, where z is a free variable.
- Bound Variable: A variable is said to be a bound variable in a formula if it occurs within the scope of the quantifier.

  Example: $\forall x\ [A\ (x)\ B(\ y)]$, here x and y are the bound variables.

Thank you

@mitu_skillologies

/mITuSkillologies

@mitu_group

/company/mitu-skillologies

MITUSkillologies

Web Resources
https://mitu.co.in
http://tusharkute.com

contact@mitu.co.in

tushar@tusharkute.com