● Predicate Logic: FOPL, Syntax, Semantics, Quantification, Inference rules in FOPL

In the topic of Propositional logic, we have seen 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 complex sentences or natural language statements. The propositional logic has very limited expressive power. 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 require 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 (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

Syntax of First-Order logic:

The syntax of FOL determines which collection of symbols is a logical expression in first-order logic. The basic syntactic elements of first-order logic are symbols. We write statements in short-hand notation in FOL.

Basic Elements of 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	∀, ∃

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).

Complex Sentences:

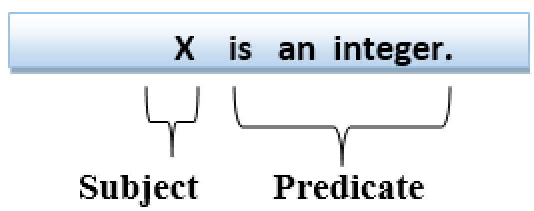
Complex sentences are made by combining atomic sentences using connectives.

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.

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 $\forall$, which resembles an inverted A.

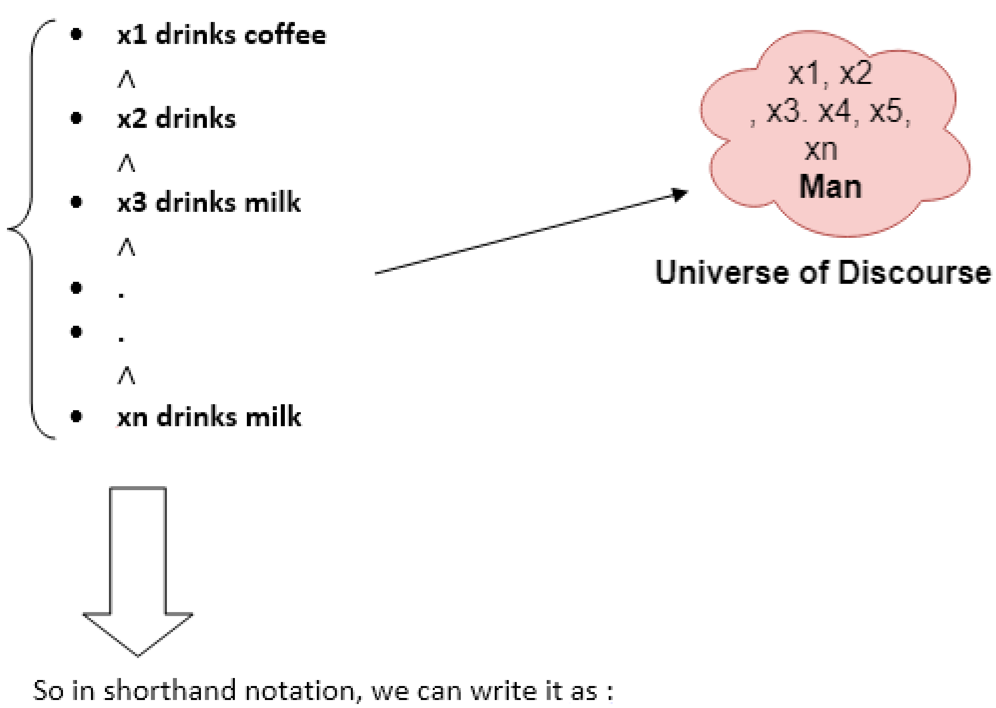
If x is a variable, then ∀x is read as:

- For all x
- For each x
- For every x.

Example:

All men drink coffee.

Let a variable x which refers to a cat so all x can be represented in UOD as below:



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

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

If x is a variable, then existential quantifier will be ∃x or ∃(x). And it will be read as:

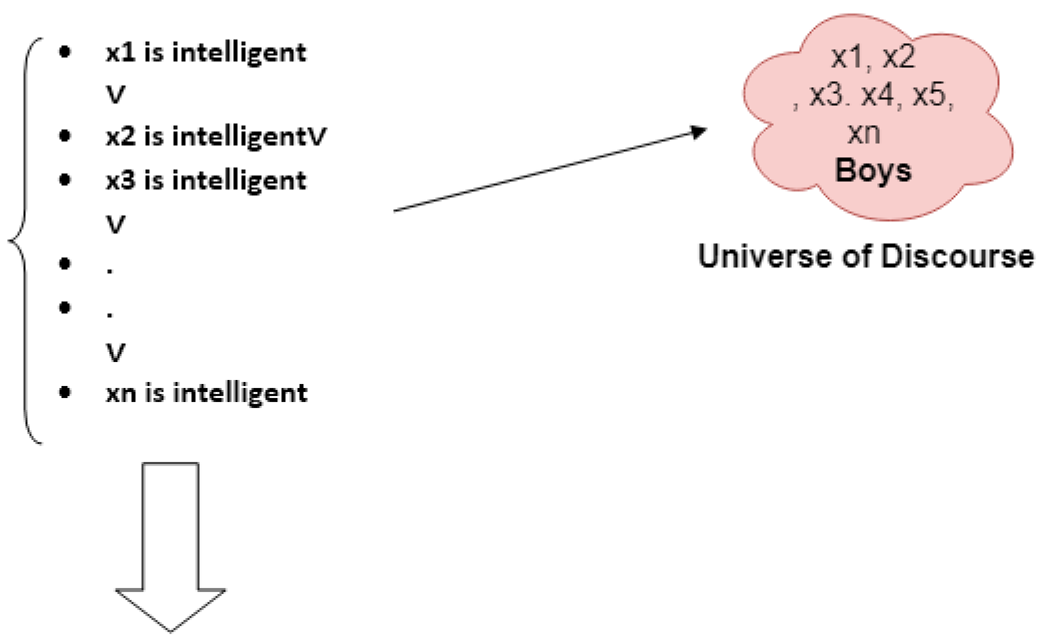
There exists a 'x.'

For some 'x.'

For at least one 'x.'

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 to remember:

- The main connective for universal quantifier $\forall$ is implication $\rightarrow$.
- The main connective for existential quantifier $\exists$ is and $\wedge$.

Properties of Quantifiers:

- In universal quantifier, $\forall x \forall y$ is similar to $\forall y \forall x$.
- In Existential quantifier, $\exists x \exists y$ is similar to $\exists y \exists x$.
- $\exists x \forall y$ is not similar to $\forall y \exists x$.

Some Examples of FOL using quantifier:

1. All birds fly.

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

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

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

2. Every man respects his parents.

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

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

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

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 $\exists$, 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 $\forall$ with negation, so following representation for this:

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

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

Inference in First-Order Logic

Inference in First-Order Logic is used to deduce new facts or sentences from existing sentences. Before understanding the FOL inference rule, let's understand some basic terminologies used in FOL.

Substitution:

Substitution is a fundamental operation performed on terms and formulas. It occurs in all inference systems in first-order logic. The substitution is complex in the presence of quantifiers in FOL. If we write F[a/x], it refers to substituting a constant "a" in place of variable "x".

Equality:

First-Order logic does not only use predicates and terms for making atomic sentences but also uses another way, which is equality in FOL. For this, we can use equality symbols which specify that the two terms refer to the same object.

Example: Brother (John) = Smith.

As in the above example, the object referred by the Brother (John) is similar to the object referred by Smith. The equality symbol can also be used with negation to represent that two terms are not the same objects.

Example: $\neg$(x=y) which is equivalent to x $\neq$y.

FOL inference rules for quantifier:

As propositional logic we also have inference rules in first-order logic, so following are some basic inference rules in FOL:

- Universal Generalization
- Universal Instantiation
- Existential Instantiation
- Existential introduction

1. Universal Generalization:

Universal generalization is a valid inference rule which states that if premise P(c) is true for any arbitrary element c in the universe of discourse, then we can have a conclusion as ∀ x P(x).

It can be represented as:

$$\frac{P(c)}{\forall x\, P(x)}$$

This rule can be used if we want to show that every element has a similar property.

In this rule, x must not appear as a free variable.

Example: Let's represent, P(c): "A byte contains 8 bits", so for ∀ x P(x) "All bytes contain 8 bits.", it will also be true.

2. Universal Instantiation:

Universal instantiation is also called as universal elimination or UI is a valid inference rule. It can be applied multiple times to add new sentences.

The new KB is logically equivalent to the previous KB.

As per UI, we can infer any sentence obtained by substituting a ground term for the variable.

The UI rule state that we can infer any sentence P(c) by substituting a ground term c (a constant within domain x) from ∀ x P(x) for any object in the universe of discourse.

It can be represented as:

$$\frac{\forall x\, P(x)}{P(c)}$$

Example:1.

IF "Every person like ice-cream"=> $\forall x\, P(x)$ so we can infer that

"John likes ice-cream" => $P(c)$

Example: 2.

Let's take a famous example,

"All kings who are greedy are Evil." So let our knowledge base contains this detail as in the form of FOL:

$\forall x\; king(x) \wedge greedy(x) \rightarrow Evil(x)$,

So from this information, we can infer any of the following statements using Universal Instantiation:

King(John) $\wedge$ Greedy (John) $\rightarrow$ Evil (John),

King(Richard) $\wedge$ Greedy (Richard) $\rightarrow$ Evil (Richard),

King(Father(John)) $\wedge$ Greedy (Father(John)) $\rightarrow$ Evil (Father(John)),

3. Existential Instantiation:

Existential instantiation is also called as Existential Elimination, which is a valid inference rule in first-order logic.

It can be applied only once to replace the existential sentence.

The new KB is not logically equivalent to old KB, but it will be satisfiable if old KB was satisfiable.

This rule states that one can infer $P(c)$ from the formula given in the form of $\exists x\, P(x)$ for a new constant symbol c.

The restriction with this rule is that c used in the rule must be a new term for which $P(c)$ is true.

It can be represented as:

$$\frac{\exists x\, P(x)}{P(c)}$$

Example:

From the given sentence: ∃x Crown(x) ∧ OnHead(x, John),

So we can infer: Crown(K) ∧ OnHead(K, John), as long as K does not appear in the knowledge base.

The above used K is a constant symbol, which is called Skolem constant.

The Existential instantiation is a special case of Skolemization process.

4. Existential introduction

An existential introduction is also known as an existential generalization, which is a valid inference rule in first-order logic.

This rule states that if there is some element c in the universe of discourse which has a property P, then we can infer that there exists something in the universe which has the property P.

It can be represented as:

$$\frac{P(c)}{\exists x P(x)}$$

Example: Let's say that,

"Priyanka got good marks in English."

"Therefore, someone got good marks in English."

● Resolution in FOL

Resolution is a theorem proving technique that proceeds by building refutation proofs, i.e., proofs by contradictions. It was invented by Mathematician John Alan Robinson in 1965.

Resolution is used, if various statements are given, and we need to prove a conclusion of those statements. Unification is a key concept in proofs by resolutions. Resolution is a single inference rule which can efficiently operate on the conjunctive normal form or clausal form.

Clause: Disjunction of literals (an atomic sentence) is called a clause. It is also known as a unit clause.