Introduction to Predicate Logic

Part 1

cse352

Professor Anita Wasilewska

Predicate Logic Language

Symbols:

1. P, Q, R… **predicates symbols**, denote relations in "real life", countably infinite set
2. x,y,z…. **variables**, countably infinite set
3. c1, c2, … **constants**, countably infinite set
4. f, g, h … **functional symbols**, may be empty, denote functions in "real life"

5. **Propositional connectives:**

   $\vee$, $\wedge$, $\Rightarrow$, $\neg$, $\Leftrightarrow$

6. **Symbols for quantifiers**

   $\forall$ x – universal quantifier reads: For all x…

   $\exists$ x – existential quantifier reads: There is x…

Formulas of Predicate Logic

We use symbols **1** - **6** to build **formulas** of predicate logic as follows

1. P(x), Q(x,f(y)), R(x)… R(c1), Q(x, c3), Q(g(x,y), c), …
   are called **atomic formulas** for any variables x, y,… ,
   functions f, g …. and constants c, c1, c2, …
2. **All atomic formulas are formulas ;**
3. If **A,B** are **formulas** then (like in propositional logic):
   $(A \vee B)$, $(A \wedge B)$, $(A \Rightarrow B)$, $(A \Leftrightarrow B)$, $\neg A$
   are **formulas**
4. **$\forall x\, A$, $\exists y\, A$** are **formulas,** for any variables x, y
5. The set **F** of **all formulas** is the **smallest** set that fulfills the conditions 1 -4.

Examples

For example: let

$P(y)$, $Q(x,c)$, $R(z)$, $P_1(g(x, y), z)$ be **atomic** formulas, i.e.

$P(x)$, $Q(x,c)$, $R(z)$, $P_1(g(x, y), z) \in F$

Then we form **some** other formulas out of them as follows:

$(P(y) \lor \neg Q(x, c)) \in F$

It is a **formula A** with two **free variables** x, y

We denote it as a formula **A(x,y)**

$\exists x\, (P(y) \lor \neg Q(x, c)) \in F$ here y is a free variable

We denote it as a formula **B(y)**

$\forall y\, (P(y) \lor \neg Q(x, c)) \in F$ here x is a free variable

We denote it as a formula **C(x)**

$\forall y\, \exists x (P(y) \lor \neg Q(x, c)) \in F$ here we have no no free variables

Free and Bound Variables

Quantifiers **bound** variables within formulas

For example: A is a formula:

$\exists \mathbf{x} (P(\mathbf{x}) \Rightarrow \neg Q(\mathbf{x}, y))$

all the **x**'s in **A** are **bounded** by **∃ x**

y is a **free variable** in **A** and we write **A as A(y)**

A(y) can be bounded by a quantifier, for example

$\forall \mathbf{y} \; \exists x (P(x) \Rightarrow \neg Q(x, \mathbf{y}))$

y got **bounded** and there **are no free** variables in **A** now

A **formula without free variables** is called a **sentence**

Logic and Mathematical Formulas

We often use **logic symbols** while writing mathematical statements in a more **symbolic** way

Example of a Mathematical Statement:

$\forall x \in N \, (x > 0 \wedge \exists y \in N \, (y = 1))$

1. Quantifiers $\forall x \in N$, $\exists y \in N$ **are** called quantifiers with **restricted domain**
2. **Logic uses only quantifiers $\forall x$, $\exists y$**
3. **$x > 0$** and **$y = 1$** are mathematical statements about “real relations” **>** and **=**
4. Logic **uses symbols** P, Q, R… for relations
5. **For example we use**

   $R(y, c_1)$ for **$y = 1$** and **$P(x, c_2)$** for **$x > 0$** where **c_1** and **c_2** are **constants** representing **numbers 1 and 0**, respectively

Translation of Mathematical Statements to Logic Formulas

Consider a Mathematical Statement written with logical symbols

$$\forall x \in N\ (x > 0 \land \exists y \in N\ (y = 1))$$

$x \in N$ – we **translate it** as **one** argument predicate $Q(x)$
$x > 0$ – we **translate it** as $P(x, c_1)$, and $y = 1$ as $R(y, c_2)$ and get

$$\forall Q(x)\ (P(x, c_1) \land \exists Q(y)\ R(y, c_2))$$

↑ Logic formula with **restricted domain** quantifiers

But this is **not yet a proper logic formula** since **we cannot** have quantifiers $\forall Q(x)$, $\exists Q(y)$ **in LOGIC,** but only quantifiers $\forall x$, $\exists x$

$\forall Q(x)$, $\exists Q(y)$ are called **quantifiers with restricted domain**

Logic Formula Corresponding Mathematical Statement

We need to "**get rid**" of **quantifiers with restricted domain** i.e. to **translate them** into **logic quantifiers:** **$\forall$ x, $\exists$ y**

$\exists x \in N$, $\exists y \in N$ are restricted quantifiers

↑ certain **predicate** P(x)

General: restricted domain quantifiers are :

$\forall A(x)$, $\exists B(x)$

for A(x), B(x) any formulas that RESTRICT the domain of quantifiersin particular atomic formulas (predicates) P(x), Q(x)

Restricted Domain Existential Quantifiers

Translation for existential l quantifier

$$\exists_{A(x)} B(x) \equiv \exists x(A(x) \land B(x))$$

↑ **restricted** ↑**logic, not restricted**

Example (mathematical formulas):

$\exists\ x \neq 1\ (x>0 \Rightarrow x + y > 5)$ - restricted

$\exists\ x\ ((x \neq 1\ ) \land (x > 0 \Rightarrow x + y > 5))$ - not restricted

↑ $B(x, y)$

English statement:

Some students are good.

Logic Translation (restricted domain):

$$\exists_{S(x)} G(x)$$

Predicates are :

S(x) – x is a student

G(x) – x is good

TRANSLATION:

$$\exists x(S(X) \land G(x))$$

Restricted Quantifiers and Logic Quantifiers

Translation for universal quantifier

Restricted — Logic (non-restricted)

$\forall_{A(x)} B(x) \quad \Xi \quad \forall x\, (A(x) \Rightarrow B(x))$

Example (mathematical statement)

$\forall x \in N\ (x = 1 \lor x < 0)$ restricted domain

$\Xi\ \forall x\, (x \in N \Rightarrow (x = 1 \lor x < 0))$ – non-restricted

Translation of Mathematic statements to Logic formulas

Mathematical statement:

$$\forall x\,(x \in N \Rightarrow (\, x = 1 \lor x < 0)\,)$$

$x \in N$ – translates to $N(x)$

$x < 0$ – translates to $P(x, c_1)$

$x < y$ – $<$ is a 2 argument relation - two argument predicate $P(x, y)$, x, y are variables

0 – is a constant – denote by c_1

$x=1$ - $=$ is a two argument predicate $Q(x,y)$

$x = 1$ - 1 is constant denoted by c_2

$x=1$ translates to $Q(x, c_2)$

Corresponding logic formula:

$$\forall x\,(N(x) \Rightarrow (Q(x, c_2) \lor P(x, c_1))\,)$$

Remark

Mathematical statement: **x + y = 5**

We re-write it as

= (+ (x, y), 5)

Given x = 2, x = 1, we get **+(2,1) = 3** and the statement:

= (3,5) **is FALSE (F)**

Predicates always returns F or T

We really need also **function** symbols (like +, etc..) to translate mathematical statements to logic, even if we could use only relations as functions are special relations

This is why in **formal** definition of the predicate language we often we have **2 sets of symbols**

1. **Predicates** symbols which can be true or false in proper domains
2. **Functions** symbols (formally called **terms**)

Translations to Logic

Rules:

1. **Identify** the domain: always a set $X \neq \phi$
2. **Identify** predicates (simple: atomic)
3. **Identify** functions (if needed)
4. **Identify** the connectives $\vee, \wedge, \Rightarrow, \neg, \Leftrightarrow$
5. **Identify** the quantifiers $\forall x, \exists x$ or Restricted Quantifiers $\forall P(x), \exists Q(x)$
6. **Write a formula using only symbols for 2 ,3, 4,5**

Use restricted domain quantifier translation rules, where needed to write

7. **Write LOGIC formula** – formula without Restricted Quantifiers

Translations Examples

Translate:

For every bird there are some birds that are white

Predicates:

B(x) – x is a bird in the domain $X \neq \phi$

W(x) – x is white in the domain $X \neq \phi$

Restricted:

$\forall_{B(x)} \exists_{B(x)} W(x)$

Logic

$\forall x(B(x) \Rightarrow \exists x\,(B(x) \wedge W(x)))$

Re-name variables

$\forall x(B(x) \Rightarrow \exists y(B(y) \wedge W(y)))$

By Laws of Quantifiers - we will study the laws later, we can re-write it as

$\forall x \exists y\,(B(x) \Rightarrow (B(y) \wedge W(y)))$

BUT you do not do it NOW!

Example

For every student there is a student that is an elephant

B(x)- x is a student in the domain $X \neq \phi$

W(x) – x is an elephant in the domain $X \neq \phi$

$\forall_{B(x)} \exists_{B(x)} W(x)$ - restricted

$\forall_{B(x)} \exists x(B(x) \wedge W(x))$

$\forall x(B(x) \Rightarrow \exists x(B(x) \wedge W(x)))$ (logic formula)

Translations Example

Translate: **Some patients like all doctors**

Predicates:

P(x) – x is a patient in the domain $X \neq \phi$

D(x) – x is a doctor in the domain $X \neq \phi$

L(x,y) – x likes y in the domain $X \neq \phi$

$$\exists_{P(x)} \forall_{D(y)} L(x,y)$$

There is a **patient(x)**, such that for all **doctors(y)**, **x** likes **y**

$$\exists x(P(x) \land \forall y(D(y) \Rightarrow L(x,y)))$$

(by **law of quantifiers** to be studied later **we can** "pull out $\forall y$") and transform our formula into

$$\exists x \forall y(P(x) \land (D(y) \Rightarrow L(x,y)))$$

BUT you do not do it NOW!

Translations Example

Translate: **There are students who hate all Professors**

Predicates:

S(x) – x is a student in the domain X ≠ ϕ

P(x) – x is a Professor in the domain X ≠ ϕ

H(x,y) – x hates y in the domain X ≠ ϕ

$$\exists_{S(x)} \forall_{P(y)} H(x,y)$$

There is a student**(x)**, such that for all Proffesor**(y)**, **x** hates **y**

$$\exists x(S(x) \wedge \forall y(P(y) \Rightarrow H(x,y)))$$

(by **law of quantifiers** to be studied later **we can** "pull out $\forall y$") and transform our formula into

$$\exists x \forall y(S(x) \wedge (P(y) \Rightarrow H(x,y)))$$

BUT you do not do it NOW!

PATTERN!

Translations Exercise

- Here is a mathematical statement S:
- *For all natural numbers n the following implication holds:*

IF $n < 0$, then there is a natural number m, such that

$m+n <0$

1. Re-write S as a "formula" MF that only uses mathematical and logical symbols
2. Translate your MF to a correct logic formula LF
3. Argue whether the statement S it true of false
4. Give an interpretation of the logic formula LF (in a non-empty set X) under which LF is false