CS 441 Discrete Mathematics for CS
Lecture 4

Predicate logic

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Announcements

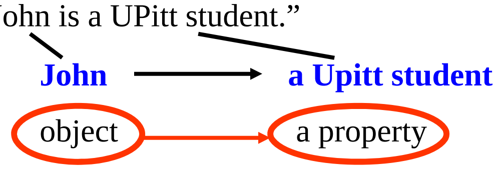
- **Homework assignment 1 due today**
- **Homework assignment 2:**
  - **posted on the course web page**
  - **Due on Thursday January 23, 2013**
- **Recitations today and tomorrow:**
  - **Practice problems related to assignment 2**

Propositional logic: limitations

Propositional logic: the world is described in terms of elementary propositions and their logical combinations

Elementary statements/propositions:

- **Typically refer to objects, their properties and relations. But these are not explicitly represented** in the propositional logic
  - **Example:**
    - "John is a UPitt student."



    - Objects and properties are hidden in the statement, it is not possible to reason about them

Propositional logic: limitations

(1) Statements that hold for many objects must be enumerated

- **Example:**
  - John is a CS UPitt graduate → John has passed cs441
  - Ann is a CS Upitt graduate → Ann has passed cs441
  - Ken is a CS Upitt graduate → Ken has passed cs441
  - …
- **Solution:** make statements with **variables**
  - x is a CS UPitt graduate → x has passed cs441

Propositional logic: limitations

(2) Statements that define the property of the group of objects

- **Example:**
  - All new cars must be registered.
  - Some of the CS graduates graduate with honor.
- **Solution:** make statements with **quantifiers**
  - **Universal quantifier** –the property is satisfied by all members of the group
  - **Existential quantifier** – at least one member of the group satisfy the property

Predicate logic

Remedies the limitations of the propositional logic

- Explicitly models objects and their properties
- Allows to make statements with variables and quantify them

Predicate logic:

- **Constant** –models a specific object

  Examples: "John", "France", "7"
- **Variable** – represents object of specific type **(defined by the *universe of discourse*)**

  Examples: x, y

  (universe of discourse can be people, students, numbers)
- **Predicate -** over one, two or many variables or constants.
  - Represents properties or relations among objects

  Examples: Red(car23), student(x), married(John,Ann)

Predicates

Predicates represent properties or relations among objects

- A predicate P(x) assigns a value **true or false** to each x depending on whether the property holds or not for x.
- The assignment is best viewed as a big table with the variable x substituted for objects from ***the universe of discourse***

Example:

- Assume **Student(x)** where the universe of discourse are people
- Student(John) …. T (if John is a student)
- Student(Ann) …. T (if Ann is a student)
- Student(Jane) ….. F (if Jane is not a student)
- …

Predicates

Assume a predicate P(x) that represents the statement:

- **x is a prime number**

Truth values for different x:

- P(2) T
- P(3) T
- P(4) F
- P(5) T
- P(6) F

All statements P(2), P(3), P(4), P(5), P(6) are propositions

…

But P(x) with variable x is not a proposition

Quantified statements

Predicate logic lets us to make statements about groups of objects

- To do this we use **special quantified expressions**

Two types of quantified statements:

- **universal**

  Example: ‘ all CS Upitt graduates have to pass cs441”

  – the statement is true for all graduates
- **existential**

  Example: ‘Some CS Upitt students graduate with honor.’

  – the statement is true for some people

Universal quantifier

Quantification converts a propositional function into **a proposition** by binding a variable to a set of values from the universe of discourse.

Example:

- Let P(x) denote x > x - 1. Assume x are real numbers.
- Is P(x) a proposition? **No.** Many possible substitutions.
- Is ∀x P(x) a proposition? **Yes.**
- What is the truth value for ∀x P(x) ?
  – **True**, since P(x) holds for all x.

Existential quantifier

Quantification converts a propositional function into **a proposition** by binding a variable to a set of values from the universe of discourse.

Example:

- Let T(x) denote x > 5 and x is from Real numbers.
- Is T(x) a proposition? **No.**
- Is ∃x T(x) a proposition? **Yes.**
- What is the truth value for ∃x T(x) ?
  - Since 10 > 5 is true. Therefore, ∃x T(x) is **true.**

Summary of quantified statements

- When **∀x P(x)** and **∃x P(x)** are true and false?

Statement	When true?	When false?
∀x P(x)	P(x) true for all x	There is an x where P(x) is false.
∃x P(x)	There is some x for which P(x) is true.	P(x) is false for all x.

Suppose the elements in the universe of discourse can be enumerated as x1, x2, ..., xN then:

- **∀x P(x) is true whenever P(x1) ∧ P(x2) ∧ ... ∧ P(xN) is true**
- **∃x P(x) is true whenever P(x1) ∨ P(x2) ∨ ... ∨ P(xN) is true**.

Translation with quantifiers

Sentence:

- All Upitt students are smart.

- **Assume:** the domain of discourse of x are Upitt students
- **Translation:**
- $\forall x$ Smart(x)

- **Assume:** the universe of discourse are students (all students):
- $\forall x$ at(x,Upitt) → Smart(x)

- **Assume:** the universe of discourse are people:
- $\forall x$ student(x) $\wedge$ at(x,Upitt) → Smart(x)

M. Hauskrecht

Translation with quantifiers

Sentence:

- Someone at CMU is smart.

- **Assume:** the domain of discourse are all CMU affiliates
- **Translation:**
- $\exists$ x Smart(x)

- **Assume:** the universe of discourse are people:
- $\exists$ x at(x,CMU) $\wedge$ Smart(x)

M. Hauskrecht

Translation with quantifiers

- **Assume two predicates S(x) and P(x)**

Universal statements typically tie with implications

- **All S(x) is P(x)**
  - $\forall x\ (S(x) \rightarrow P(x))$
- **No S(x) is P(x)**
  - $\forall x(S(x) \rightarrow \neg P(x))$

Existential statements typically tie with conjunctions

- **Some S(x) is P(x)**
  - $\exists x\ (S(x) \wedge P(x))$
- **Some S(x) is not P(x)**
  - $\exists x\ (S(x) \wedge \neg P(x))$

M. Hauskrecht

Nested quantifiers

- More than one quantifier may be necessary to capture the meaning of a statement in the predicate logic.

Example:

- Every real number has its corresponding negative.
- **Translation:**
  - Assume:
    - a real number is denoted as x and its negative as y
    - A predicate P(x,y) denotes: "$x + y = 0$"
- Then we can write:

  $\forall x\ \exists\ y\ P(x,y)$

M. Hauskrecht

Nested quantifiers

- More than one quantifier may be necessary to capture the meaning of a statement in the predicate logic.

Example:

- There is a person who loves everybody.
- **Translation:**
  - Assume:
    - Variables x and y denote people
    - A predicate L(x,y) denotes: “x loves y”
- Then we can write in the predicate logic:

  $\exists x \forall y\ L(x,y)$

M. Hauskrecht

Order of quantifiers

The order of nested quantifiers matters if quantifiers are of different type

- $\forall x \exists y\ L(x,y)$ is not the same as $\exists y \forall x\ L(x,y)$

Example:

- Assume L(x,y) denotes “x loves y”

- Then: $\forall x \exists y\ L(x,y)$
- Translates to: Everybody loves somebody.
- And: $\exists y \forall x\ L(x,y)$
- Translates to: There is someone who is loved by everyone.

The meaning of the two is different.

M. Hauskrecht

Order of quantifiers

The order of nested quantifiers does not matter if quantifiers are of the same type

Example:

- For all x and y, if x is a parent of y then y is a child of x
- **Assume:**
  - Parent(x,y) denotes "x is a parent of y"
  - Child(x,y) denotes "x is a child of y"

- Two equivalent ways to represent the statement:
  - $\forall x\ \forall y\ \text{Parent}(x,y) \rightarrow \text{Child}(y,x)$
  - $\forall y\ \forall x\ \text{Parent}(x,y) \rightarrow \text{Child}(y,x)$

Translation exercise

Suppose:

- Variables x,y denote people
- L(x,y) denotes "x loves y".

Translate:

- Everybody loves Raymond. $\forall x\ L(x,\text{Raymond})$
- Everybody loves somebody. $\forall x \exists y\ L(x,y)$
- There is somebody whom everybody loves. $\exists y \forall x\ L(x,y)$
- There is somebody who Raymond doesn't love.
  $\exists y \neg L(\text{Raymond},y)$
- There is somebody whom no one loves.
  $\exists y\ \forall x\ \neg L(x,y)$