CS 1571 Introduction to AI
Lecture 14

First-order logic

Milos Hauskrecht

milos@cs.pitt.edu

5329 Sennott Square

Midterm

- **The midterm** for the course will be held on
    - October 28, 2014
    - In class exam
    - Closed book
    - Material covered by October 23, 2014

Simulated annealing competition

Tournament winner:

- Adam Schwab   (66.87)

Second

- Michael Minella   (68.09)

Third

- Chris Iwaszko   (68.90)

Congratulations!

Limitations of propositional logic

World we want to represent and reason about consists of a number of **objects** with variety of **properties** and **relations** among them

Propositional logic:

- Represents statements about the world without reflecting this structure and without modeling these entities explicitly
- **Example:** 'Peter likes ice-cream' treated as an atomic entity

Consequence:

- some knowledge is hard or impossible to encode in the propositional logic.
- Two cases that are hard to represent:
  - **Statements about similar objects, or relations**
  - **Statements referring to groups of objects**.

Limitations of propositional logic

- **Statements about similar objects and relations needs to be enumerated**

- **Example:** Seniority of people domain

  knowledge that would support inferences we need:

  > *John is older than Mary* $\wedge$ *Mary is older than Paul*
  > $\Rightarrow$ *John is older than Paul*

  > *Jane is older than Mary* $\wedge$ *Mary is older than Paul*
  > $\Rightarrow$ *Jane is older than Paul*

- **Problem:** if we have many people and their age relations we need to represent many rules to support the inferences

- **Solution: ??**

Limitations of propositional logic

- **Statements about similar objects and relations needs to be enumerated**

- **Example:** Seniority of people domain

  For inferences we need:

  > *John is older than Mary* $\wedge$ *Mary is older than Paul*
  > $\Rightarrow$ *John is older than Paul*

  > *Jane is older than Mary* $\wedge$ *Mary is older than Paul*
  > $\Rightarrow$ *Jane is older than Paul*

- **Problem:** if we have many people and their age relations we need to represent many rules to support the inferences

- **Solution: introduce variables**

  > ***PersA*** *is older than* ***PersB*** $\wedge$ ***PersB*** *is older than* ***PersC***
  > $\Rightarrow$ ***PersA*** *is older than* ***PersC***

Limitations of propositional logic

- **Statements referring to groups of objects require exhaustive enumeration of objects**
- **Example:**

  Assume we want to express *Every student likes vacation*

  Doing this in propositional logic would require to include statements about every student

  > *John likes vacation* ∧
  >
  > *Mary likes vacation* ∧
  >
  > *Ann likes vacation* ∧
  >
  > …

- **Solution:** Allow quantified statements

First-order logic (FOL)

- More expressive than **propositional logic**

- **Eliminates deficiencies of PL by:**
  - Representing objects, their properties, relations and statements about them;
  - Introducing variables that refer to an arbitrary objects and can be substituted by a specific object
  - Introducing quantifiers allowing us to make statements over groups objects without the need to represent each of them separately

Logic

Logic is defined by**:**

- **A set of sentences**
  - A sentence is constructed from a set of primitives according to syntax rules.
- **A set of interpretations**
  - An interpretation gives a semantic to primitives. It associates primitives with objects, values in the real world.
- **The valuation (meaning) function** *V*
  - Assigns a truth value to a given sentence under some interpretation

$$V : \text{sentence} \times \text{interpretation} \rightarrow \{True, False\}$$

First-order logic: Syntax

Term – **a syntactic entity for representing objects**

Terms in FOL:
- **Constant symbols:** represent specific objects
  - E.g. *John, France, car89*
- **Variables:** represent objects of a certain type (type = domain of discourse)
  - E.g. *x,y,z*
- **Functions** applied to one or more terms
  - E.g. *father-of (John)*
    father-of(father-of(John))

First order logic: Syntax

Sentences in FOL:

- **Atomic sentences:**
    - **A predicate symbol** applied to 0 or more terms
      Examples:
      Red(car12),
      Sister(Amy, Jane);
      Manager(father-of(John));

    - t1 = t2 **equivalence** of terms
      Example:
      John = father-of(Peter)

First order logic: Syntax

Sentences in FOL:

- **Complex sentences:**
- Assume ϕ, ψ are sentences in FOL. Then:
    - $(\phi \wedge \psi)$ $(\phi \vee \psi)$ $(\phi \Rightarrow \psi)$ $(\phi \Leftrightarrow \psi)$ $\neg \psi$
      and
    - $\forall x\ \phi$ $\exists y\ \phi$
    are sentences

Symbols $\exists, \forall$
- stand for the **existential** and the **universal** quantifier

Semantics: Interpretation

An interpretation **I** is defined by **mapping** constants, functions and predicates to the **domain of discourse D or relations on D**

- **domain of discourse:** a set of objects in the world we represent and refer to;

An interpretation *I* maps:

- Constant symbols to objects in D

   I(*John*) =

- Predicate symbols to relations, properties on D

   I(*brother*) = { ⟨ ⟩; ⟨ ⟩ ; …. }

- Function symbols to functional relations on D

   I(*father-of*) = { ⟨ ⟩ → ;   ⟨ ⟩ → ; …. }

Semantics of sentences

Meaning (evaluation) function:

   V : sentence   × interpretation   → {*True* , *False* }

A **predicate** *predicate(term-1, term-2, …, term-n)* is true for the interpretation *I* , iff the objects referred to by *term-1, term-2, …, term-n* are in the relation referred to by *predicate*

   I(*John*) =   I(*Paul*) =
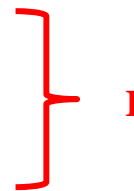
   I(*brother*) = { ⟨ ⟩; ⟨ ⟩ ; …. }   **I**

   brother(John, Paul) = ⟨ ⟩   in I(*brother*)

   V(*brother(John, Paul), I*) = **True**

Semantics of sentences

- **Equality** $V(term\text{-}1 = term\text{-}2, I) = \textbf{\textit{True}}$

                       Iff $I(term\text{-}1) = I(term\text{-}2)$

- **Boolean expressions: standard**

    E.g. $V(sentence\text{-}1 \lor sentence\text{-}2, I) = \textbf{\textit{True}}$

         Iff $V(sentence\text{-}1, I) = \textbf{\textit{True}}$ or $V(sentence\text{-}2, I) = \textbf{\textit{True}}$

- **Quantifications**

    $V(\forall x\ \phi,\ I) = \textbf{\textit{True}}$ substitution of x with d

         Iff   for all $d \in D$ $V(\phi, I[x/d]) = \textbf{\textit{True}}$

    $V(\exists x\ \phi,\ I) = \textbf{\textit{True}}$

         Iff   there is a $d \in D$, s.t. $V(\phi, I[x/d]) = \textbf{\textit{True}}$

Sentences with quantifiers

- **Universal quantification**

      All Upitt students are smart

- **Assume the universe of discourse of x are Upitt students**
              $\forall x\ smart(x)$
- **Assume the universe of discourse of x are students**
            $\forall x\ at(x, Upitt\ ) \Rightarrow smart(x)$
- **Assume the universe of discourse of x are people**

      $\forall x\ student(x)\ \land at(x, Upitt\ ) \Rightarrow smart(x)$

Sentences with quantifiers

- **Universal quantification**

    All Upitt students are smart

- **Assume the universe of discourse of x are Upitt students**

$$\forall x \; smart(x)$$

- **Assume the universe of discourse of x are students**

$$\forall x \; at(x,Upitt\;) \Rightarrow smart(x)$$

- **Assume the universe of discourse of x are people**

$$\forall x \; student(x) \;\wedge at(x,Upitt\;) \Rightarrow smart(x)$$

Typically the universal quantifier connects with an implication

---

Sentences with quantifiers

- **Existential quantification**

    Someone at CMU is smart

- **Assume the universe of discourse of x are CMU affiliates**

$$\exists x \; smart(x)$$

- **Assume the universe of discourse of x are people**

$$\exists x \; at(x,CMU) \quad \wedge smart(x)$$

Sentences with quantifiers

- **Existential quantification**

    Someone at CMU is smart

- **Assume the universe of discourse of x are CMU affiliates**

    $\exists x \ smart(x)$

- **Assume the universe of discourse of x are people**

    $\exists x \ at(x,CMU) \quad \wedge \ smart(x)$

Typically the existential quantifier connects with a conjunction

---

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

Nested quantifiers

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

- 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:

  ?

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:

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

Translation exercise

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

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

Order of quantifiers

- **Order of quantifiers of the same type does not matter**

  For all x and y, if x is a parent of y then y is a child of x

  $$\forall x, y\ parent\ (x, y) \Rightarrow child\ (y, x)$$
  $$\forall y, x\ parent\ (x, y) \Rightarrow child\ (y, x)$$

- **Order of different quantifiers changes the meaning**

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

  Everybody loves somebody

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

  There is someone who is loved by everyone

Connections between quantifiers

Everyone likes ice cream

?

Connections between quantifiers

Everyone likes ice cream

$$\forall\, x\; likes\,(x, IceCream\;)$$

Is it possible to convey the same meaning using an existential quantifier ?

There is no one who does not like ice cream

$$\neg \exists\, x\; \neg\, likes\,(x, IceCream\;)$$

A universal quantifier in the sentence can be expressed using an existential quantifier !!!

Connections between quantifiers

Someone likes ice cream

$$\exists x \; likes \; (x, IceCream)$$

Is it possible to convey the same meaning using a universal quantifier ?

Not everyone does not like ice cream

$$\neg \forall \; x \; \neg \; likes \; (x, IceCream)$$

An existential quantifier in the sentence can be expressed using a universal quantifier !!!

Representing knowledge in FOL

Example:
  Kinship domain

- **Objects:** people
  $$John, Mary, Jane, \ldots$$
- **Properties:** gender
  $$Male \; (x), Female \; (x)$$
- **Relations:** parenthood, brotherhood, marriage
  $$Parent \; (x, y), Brother \; (x, y), Spouse \; (x, y)$$
- **Functions:** mother-of (one for each person x)
  $$MotherOf \; (x)$$

Kinship domain in FOL

Relations between predicates and functions: write down what we know about them; how relate to each other.

- Male and female are disjoint categories

$$\forall x \ Male \ (x) \Leftrightarrow \neg Female \ (x)$$

- Parent and child relations are inverse

$$\forall x, y \ Parent \ (x, y) \Leftrightarrow Child \ (y, x)$$

- A grandparent is a parent of parent

$$\forall g, c \ Grandparent(g, c) \Leftrightarrow \exists p \ Parent(g, p) \wedge Parent(p, c)$$

- A sibling is another child of one's parents

$$\forall x, y \ Sibling \ (x, y) \Leftrightarrow (x \neq y) \wedge \exists p \ Parent \ (p, x) \wedge Parent \ (p, y)$$

- And so on ….