CS 1571 Introduction to AI
Lecture 18

First-order logic

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

Tic-tac-toe competition

Results:

1. Richard Matchett
2. Vinay Sarpeshkar
3. Brandon Blatnick

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 - 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

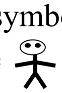
Semantics. Interpretation.

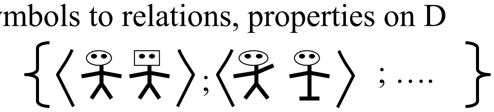
An interpretation *I* is defined by a **mapping** to the **domain of discourse D or relations on D**

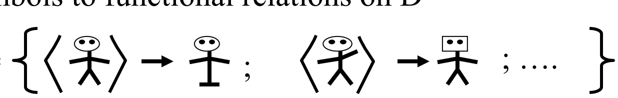
- **domain of discourse:** a set of objects in the world we represent and refer to;

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

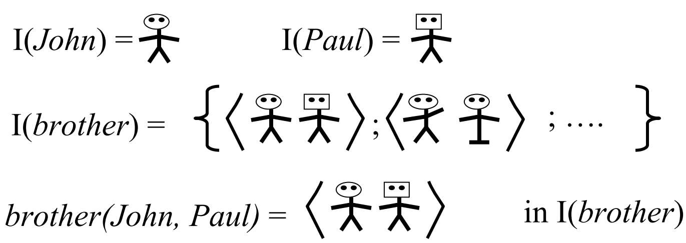
  I(*father-of*) = 

Semantics of sentences.

Meaning (evaluation) function:

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

A **predicate** *predicate(term-1, term-2, term-3, term-n)* is true for the interpretation I , iff the objects referred to by *term-1, term-2, term-3, term-n* are in the relation referred to by *predicate*



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

Semantics of sentences.

- **Equality** V(*term-1= term-2, I*) = ***True***
  Iff I(*term-1*) =I(*term-2*)
- **Boolean expressions:** **standard**

  E.g. V(*sentence-1* $\vee$ *sentence-2, I*) = ***True***
  Iff V(*sentence-1,I*)= ***True*** or V(*sentence-2,I*)= ***True***

- **Quantifications**

  V($\forall x\ \phi$, *I*) = ***True***
  Iff for all $d \in D$ V(ϕ ,$\overbrace{I[x/d]}^{\text{substitution of } x \text{ with } d}$)= ***True***

  V($\exists x\ \phi$, *I*) = ***True***
  Iff there is a $d \in D$, s.t. V(ϕ ,*I[x/d]*)= ***True***

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 ….

Inference in First order logic

Logical inference in FOL

Logical inference problem:

- Given a knowledge base KB (a set of sentences) and a sentence α , does the KB semantically entail α ?

$$KB \models \alpha \quad ?$$

In other words: In all interpretations in which sentences in the KB are true, is also α true?

Logical inference problem in the first-order logic is undecidable !!!. No procedure that can decide the entailment <u>for all possible input sentences</u> in a finite number of steps.

Logical inference problem in the Propositional logic

Computational procedures that answer:

$$KB \models \alpha \ ?$$

Three approaches:

- **Truth-table approach**
- **Inference rules**
- **Conversion to the inverse SAT problem**
  - **Resolution-refutation**

Inference in FOL: Truth table

- Is the Truth-table approach a viable approach for the FOL?

  ?

Inference in FOL: Truth table approach

- Is the Truth-table approach a viable approach for the FOL?
  ?
- **NO!**
- Why?
- It would require us to enumerate and list all possible interpretations **I**
- I = (assignments of symbols to objects, predicates to relations and functions to relational mappings)
- Simply there are too many interpretations

Inference in FOL: Inference rules

- Is the Inference rule approach a viable approach for the FOL?
  ?

Inference in FOL: Inference rules

- Is the Inference rule approach a viable approach for the FOL?

  ?
- Yes.
- The inference rules represent sound inference patterns one can apply to sentences in the KB
- What is derived follows from the KB
- Caveat: we need to add rules for handling quantifiers

Inference rules

- **Inference rules from the propositional logic**:
  - Modus ponens

$$\frac{A \Rightarrow B, \quad A}{B}$$

  - Resolution

$$\frac{A \vee B, \quad \neg B \vee C}{A \vee C}$$

  - and others: And-introduction, And-elimination, Or-introduction, Negation elimination
- **Additional inference rules** are needed for sentences with quantifiers and variables
  - Must involve variable substitutions

Sentences with variables

First-order logic sentences can include variables.

- **Variable** is:
  - **Bound** – if it is in the scope of some quantifier

    $\forall x \; P(x)$
  - **Free** – if it is not bound.

    $\exists x \; P(y) \wedge Q(x)$ y is free

Examples:

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

- Bound or free?

Sentences with variables

First-order logic sentences can include variables.

- **Variable** is:
  - **Bound** – if it is in the scope of some quantifier

    $\forall x \; P(x)$
  - **Free** – if it is not bound.

    $\exists x \; P(y) \wedge Q(x)$ y is free

Examples:

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

- Bound

$\forall x \left(Likes(x, y) \wedge \exists y \; Likes(y, Raymond) \right)$

- Bound or free?

Sentences with variables

First-order logic sentences can include variables.

- **Variable** is:
  - **Bound** – if it is in the scope of some quantifier

    $\forall x \; P(x)$
  - **Free** – if it is not bound.

    $\exists x \; P(y) \wedge Q(x)$ $\quad y$ is free

Examples:

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

- Bound

$\forall x \; \big(Likes\,(x, y) \wedge \exists y \; Likes\,(y, Raymond\,)\big)$

- Free

Sentences with variables

First-order logic sentences can include variables.

- **Sentence** (formula) is:
  - **Closed** – if it has no free variables

    $\forall y \exists x \; P(y) \Rightarrow Q(x)$
  - **Open** – if it is not closed

    $\exists x \; P(y) \wedge Q(x)$ $\quad y$ is free
  - **Ground** – if it does not have any variables

    $Likes(John, Jane)$

Variable substitutions

- Variables in the sentences can be substituted with terms.
  (terms = constants, variables, functions)
- **Substitution:**
  - Is represented by a mapping from variables to terms

    $\{x_1 / t_1, x_2 / t_2, \ldots\}$

  - Application of the substitution to sentences

$$SUBST(\{x / Sam, y / Pam\}, Likes(x, y)) = Likes(Sam, Pam)$$

$$SUBST(\{x / z, y / fatherof(John)\}, Likes(x, y)) = ?$$

Variable substitutions

- Variables in the sentences can be substituted with terms.
  (terms = constants, variables, functions)
- **Substitution:**
  - Is represented by a mapping from variables to terms

    $\{x_1 / t_1, x_2 / t_2, \ldots\}$

  - Application of the substitution to sentences

$$SUBST(\{x / Sam, y / Pam\}, Likes(x, y)) = Likes(Sam, Pam)$$

$$SUBST(\{x / z, y / fatherof(John)\}, Likes(x, y)) =$$
$$Likes(z, fatherof(John))$$

Inference rules for quantifiers

- **Universal elimination**

$$\frac{\forall x\ \phi(x)}{\phi(a)} \qquad a \text{ - is a constant symbol}$$

  - substitutes a variable with a constant symbol

$$\forall x\ Likes(x, IceCream) \qquad Likes(Ben, IceCream)$$

- **Existential elimination.**

$$\frac{\exists x\ \phi(x)}{\phi(a)}$$

  - Substitutes a variable with a constant symbol that does not appear elsewhere in the KB

$$\exists x\ Kill(x, Victim) \qquad Kill(Murderer, Victim)$$

Inference rules for quantifiers

- **Universal instantiation (introduction)**

$$\frac{\phi}{\forall x\ \phi} \qquad x \text{ – is not free in } \phi$$

  - Introduces a universal variable which does not affect ϕ or its assumptions

$$Sister(Amy, Jane) \qquad \forall x\ Sister(Amy, Jane)$$

- **Existential instantiation (introduction)**

$$\frac{\phi(a)}{\exists x \phi(x)} \qquad \begin{array}{l} a \text{ – is a ground term in } \phi \\ x \text{ – is not free in } \phi \end{array}$$

  - Substitutes a ground term in the sentence with a variable and an existential statement

$$Likes(Ben, IceCream) \qquad \exists x\ Likes(x, IceCream)$$

Unification

- **Problem in inference:** Universal elimination gives many opportunities for substituting variables with ground terms

$$\frac{\forall x\ \phi(x)}{\phi(a)} \qquad a \text{ - is a constant symbol}$$

- **Solution:** Try substitutions that may help
  - Use substitutions of "similar" sentences in KB

- **Unification** – takes two similar sentences and computes the substitution that **makes them look the same**, if it exists

$$UNIFY\ (p, q) = \sigma \text{ s.t. } SUBST(\sigma, p) = SUBST\ (\sigma, q)$$

Unification. Examples.

- **Unification:**

  $UNIFY\ (p, q) = \sigma$ s.t. $SUBST(\sigma, p) = SUBST\ (\sigma, q)$

- **Examples:**

$$UNIFY\,(Knows(John, x), Knows(John, Jane)) = \{x / Jane\}$$

$$UNIFY(Knows(John, x), Knows(y, Ann)) = ?$$

Unification. Examples.

- **Unification:**

  $UNIFY(p,q) = \sigma$ s.t. $SUBST(\sigma, p) = SUBST(\sigma, q)$

- **Examples:**

$UNIFY(Knows(John, x), Knows(John, Jane)) = \{x / Jane\}$

$UNIFY(Knows(John, x), Knows(y, Ann)) = \{x / Ann, y / John\}$

$UNIFY(Knows(John, x), Knows(y, MotherOf(y)))$
$= ?$

Unification. Examples.

- **Unification:**

  $UNIFY(p,q) = \sigma$ s.t. $SUBST(\sigma, p) = SUBST(\sigma, q)$

- **Examples:**

$UNIFY(Knows(John, x), Knows(John, Jane)) = \{x / Jane\}$

$UNIFY(Knows(John, x), Knows(y, Ann)) = \{x / Ann, y / John\}$

$UNIFY(Knows(John, x), Knows(y, MotherOf(y)))$
$= \{x / MotherOf(John), y / John\}$

$UNIFY(Knows(John, x), Knows(x, Elizabeth)) = ?$

Unification. Examples.

- **Unification:**

  $UNIFY(p,q) = \sigma$ s.t. $SUBST(\sigma, p) = SUBST(\sigma, q)$

- **Examples:**

$$UNIFY(Knows(John, x), Knows(John, Jane)) = \{x / Jane\}$$

$$UNIFY(Knows(John, x), Knows(y, Ann)) = \{x / Ann, y / John\}$$

$$UNIFY(Knows(John, x), Knows(y, MotherOf(y))) = \{x / MotherOf(John), y / John\}$$

$$UNIFY(Knows(John, x), Knows(x, Elizabeth)) = fail$$

Generalized inference rules.

- **Use substitutions that let us make inferences**

Example: Modus Ponens

- **If there exists a substitution σ such that**

$$SUBST(\sigma, A_i) = SUBST(\sigma, A_i') \quad \textbf{for all i=1,2, n}$$

$$\frac{A_1 \wedge A_2 \wedge \ldots A_n \Rightarrow B, \quad A_1', A_2', \ldots A_n'}{SUBST(\sigma, B)}$$

- Substitution that satisfies the generalized inference rule can be build via unification process
- Advantage of the generalized rules: they are focused
  - only substitutions that allow the inferences to proceed

Resolution inference rule

- **Recall:** Resolution inference rule is sound and complete (refutation-complete) for the **propositional logic** and CNF

$$\frac{A \vee B, \quad \neg A \vee C}{B \vee C}$$

- **Generalized resolution rule is sound and refutation complete** for the first-order logic and CNF w/o equalities (if unsatisfiable the resolution will find the contradiction)

$$\sigma = UNIFY\ (\phi_i, \neg \psi_j) \neq fail$$

$$\frac{\phi_1 \vee \phi_2 \ldots \vee \phi_k, \quad \psi_1 \vee \psi_2 \vee \ldots \psi_n}{SUBST(\sigma, \phi_1 \vee \ldots \vee \phi_{i-1} \vee \phi_{i+1} \ldots \vee \phi_k \vee \psi_1 \vee \ldots \vee \psi_{j-1} \vee \psi_{j+1} \ldots \psi_n)}$$

Example: $$\frac{P(x) \vee Q(x), \quad \neg Q(John) \vee S(y)}{P(John) \vee S(y)}$$

Resolution inference rule

- **Recall:** Resolution inference rule is sound and complete (refutation-complete) for the **propositional logic** and CNF

$$\frac{A \vee B, \quad \neg A \vee C}{B \vee C}$$

- **Generalized resolution rule is sound and refutation complete** for the first-order logic and CNF w/o equalities (if unsatisfiable the resolution will find the contradiction)

$$\sigma = UNIFY\ (\phi_i, \neg \psi_j) \neq fail$$

$$\frac{\phi_1 \vee \phi_2 \ldots \vee \phi_k, \quad \psi_1 \vee \psi_2 \vee \ldots \psi_n}{SUBST(\sigma, \phi_1 \vee \ldots \vee \phi_{i-1} \vee \phi_{i+1} \ldots \vee \phi_k \vee \psi_1 \vee \ldots \vee \psi_{j-1} \vee \psi_{j+1} \ldots \psi_n)}$$

Example: $$\frac{P(x) \vee Q(x), \quad \neg Q(John) \vee S(y)}{?}$$