

CHAPTER 9

Inference with First Order Logics

Why first order?

- FOL permits quantification over variables
- Higher order logic permits quantification over functions and predicates

Inference with FOL

- Propositionalization
- Prove by resolution

Universal Instantiation (UI)

Every instantiation of a universally quantified sentence is entailed by it:

$\forall v \alpha / \text{SUBST}(\{v, g\}, \alpha)$

For any variable v and ground term g

ex. $\forall x \text{King}(x) \wedge \text{Greedy}(x) \Rightarrow \text{Evil}(x)$

- Variable (v) = x
- Ground Term (g) = John, Richard, Father(John), ... etc (all the possible inferences)
- Alpha is all the possible substitutions
 - $\forall x \text{King}(\text{John}) \wedge \text{Greedy}(\text{John}) \Rightarrow \text{Evil}(\text{John})$
 - * (x substituted with g)

Existential Instantiation (EI)

For any sentence α , variable v , and constant symbol k that does not appear elsewhere in the knowledge base:

$\exists v \alpha / \text{SUBST}(\{v/k\}, \alpha)$

Instantiation

UI can be applied several times to add new sentences

- The new KB is logically equivalent to the old

EI can be applied once to replace the existential sentence

- The new KB is not equivalent to the old, but is satisfiable iff the old KB was satisfiable:

- The new KB is not **logically equivalent** to the original KB, which, yet, is **inferentially equivalent** to the original KB

Reduction to Propositional Inference

Suppose the KB contains just the following:

- $\forall x \text{ King}(x) \wedge \text{Greedy}(x) \Rightarrow \text{Evil}(x)$
- $\text{King}(\text{John})$
- $\text{Greedy}(\text{John})$
- $\text{Brother}(\text{Richard}, \text{John})$

Instantiating the universal sentence in **all possible** ways, we have

- $\text{King}(\text{John}) \wedge \text{Greedy}(\text{John}) \Rightarrow \text{Evil}(\text{John})$
- $\text{King}(\text{Richard}) \wedge \text{Greedy}(\text{Richard}) \Rightarrow \text{Evil}(\text{Richard})$
- $\text{King}(\text{John})$
- $\text{Greedy}(\text{John})$
- $\text{Brother}(\text{Richard}, \text{John})$

The new KB is propositionalized: proposition symbols are:

- $\text{King}(\text{John}), \text{Greedy}(\text{John}), \text{Evil}(\text{John}), \text{King}(\text{Richard}), \text{etc.} \dots$

Reduction to Propositional Inference - Semi-decidable

Claim: a ground sentence is entailed by new KB iff entailed by original KB

Claim: every FOL KB can be propositionalized so as to preserve entailment

Idea: propositionalize KB and query, apply resolution, return result

Problem: with function symbols, there are infinitely many ground terms

- ex. $\text{Father}(\text{Father}(\text{Father}(\text{John})))$

Theorem: Herbrand (1930). If a sentence α is entailed by an FOL KB, it is entailed by a **finite** subset of a propositional KB

Idea: For $n = 0$ to ∞ do

- Create a propositional KB by instantiating with depth- n terms, see if α is entailed by this KB

Problem: works if α is entailed, loops if α is not entailed

Theorem: Turing (1936), Church (1936), entailment in FOL is **semidecidable**

Problems with Propositionalization

Propositionalization seems to generate lots of irrelevant sentences.

With p k -ary predicates and n constants, there are $p * n^k$ instantiations

With function symbols, it gets much worse

Generalized Modus Ponens (GMP)

GMP is used with KB of **definitive clause** (exactly one positive literal).

All variables assumed universally quantified

$$\frac{p_1', p_2', \dots, p_n', (p_1 \wedge p_2 \wedge \dots \wedge p_n \Rightarrow q)}{q\theta} \quad \text{where } p_i'\theta = p_i\theta \text{ for all } i$$

p_1' is *King(John)* p_1 is *King(x)*
 p_2' is *Greedy(y)* p_2 is *Greedy(x)*
 θ is $\{x/\text{John}, y/\text{John}\}$ q is *Evil(x)*
 $q\theta$ is *Evil(John)*

Unification

We can get the inference immediately if we can find a substitution θ such that King(x) and Greedy(x) match King(John) and Greedy(y)

$\theta = \{x/\text{John}, y/\text{John}\}$ works

$\text{UNIFY}(\alpha, \beta) = \theta$ if $\alpha\theta = \beta\theta$

p	q	θ
<i>Knows(John, x)</i>	<i>Knows(John, Jane)</i>	$\{x/\text{Jane}\}$
<i>Knows(John, x)</i>	<i>Knows(y, OJ)</i>	$\{x/\text{OJ}, y/\text{John}\}$
<i>Knows(John, x)</i>	<i>Knows(y, Mother(y))</i>	$\{y/\text{John}, x/\text{Mother(John)}\}$
<i>Knows(John, x)</i>	<i>Knows(x, OJ)</i>	<i>fail</i>

Standardizing apart eliminates overlap of variables, ex. $\text{Knows}(z_{17}, \text{OJ})$

Resolution: Brief Summary

Full First-order version:

$$\frac{\ell_1 \vee \dots \vee \ell_k, \quad m_1 \vee \dots \vee m_n}{(\ell_1 \vee \dots \vee \ell_{i-1} \vee \ell_{i+1} \vee \dots \vee \ell_k \vee m_1 \vee \dots \vee m_{j-1} \vee m_{j+1} \vee \dots \vee m_n)\theta}$$

where $\text{UNIFY}(l, \neg m_j) = \theta$

Apply resolution steps to $\text{CNF}(\text{KB} \wedge \neg \alpha)$; complete for FOL

Conversion to CNF - Example

Everyone who loves all animals is loved by someone:

- $\forall x [\forall y \text{ Animal}(y) \Rightarrow \text{Loves}(x, y)] \Rightarrow [\exists y \text{ Loves}(y, x)]$

1) Eliminate biconditionals and implications

- $\forall x [\neg \forall y \neg \text{Animal}(y) \vee \text{Loves}(x, y)] \vee [\exists y \text{ Loves}(y, x)]$

2) Move \neg inwards: $\neg \forall x, p = \exists x \neg p$, $\neg \exists x, p = \forall x \neg p$:

- $\forall x [\exists y \neg(\neg \text{Animal}(y) \vee \text{Loves}(x, y))] \vee [\exists y \text{ Loves}(y, x)]$
- $\forall x [\exists y \neg \neg \text{Animal}(y) \wedge \neg \text{Loves}(x, y)] \vee [\exists y \text{ Loves}(y, x)]$
- $\forall x [\exists y \text{ Animal}(y) \wedge \neg \text{Loves}(x, y)] \vee [\exists y \text{ Loves}(y, x)]$

3) Standardize variables: each quantifier should use a different one

- $\forall x [\exists y \text{ Animal}(y) \wedge \neg \text{Loves}(x, y)] \vee [\exists z \text{ Loves}(z, x)]$

4) Skolemize: a more general form of existential instantiation

- Each existential variables is replaced by a Skolem function of the enclosing universally quantified variables:
- $\forall x [\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x))] \vee \text{Loves}(G(x), x)$

5) Drop universal quantifiers:

- $[\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x))] \vee \text{Loves}(G(x), x)$

6) Distribute \wedge over \vee :

- $[\text{Animal}(F(x)) \vee \text{Loves}(G(x), x)] \wedge [\neg \text{Loves}(x, F(x)) \vee \text{Loves}(G(x), x)]$

Knowledge Base in First Order Logic

The law says that it is a crime for an American to sell weapons to hostile nations. The country Nono, an enemy of America, has some missiles, and all of its missiles were sold to it by Colonel West, who is American.

Prove that Col. West is a criminal

... it is a crime for an American to sell weapons to hostile nations:

- $\text{American}(x) \wedge \text{Weapon}(y) \wedge \text{Sells}(x, y, z) \wedge \text{Hostile}(z) \Rightarrow \text{Criminal}(x)$

... Nono has some missiles

- $\text{Owns}(\text{Nono}, M_1)$ and $\text{Missile}(M_1)$

... All of its missiles were sold to it by Colonel West

- $\forall x \text{ Missile}(x) \wedge \text{Owns}(\text{Nono}, x) \Rightarrow \text{Sells}(\text{West}, x, \text{Nono})$

... Missiles are weapons

- $\text{Missile}(x) \Rightarrow \text{Weapon}(x)$
- ... An enemy of America counts as “hostile”
 - $\text{Enemy}(x, \text{America}) \Rightarrow \text{Hostile}(x)$
- ... West, who is American
 - $\text{American}(\text{West})$
- ... The country Nono, an enemy of America
 - $\text{Enemy}(\text{Nono}, \text{America})$

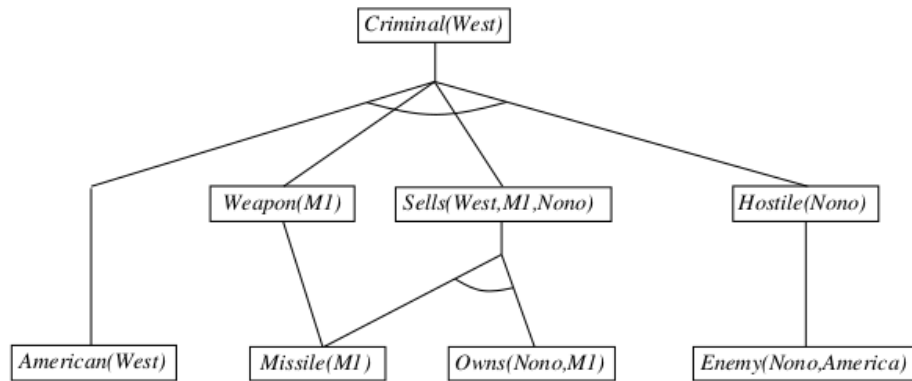
Forward Chaining Algorithm

```

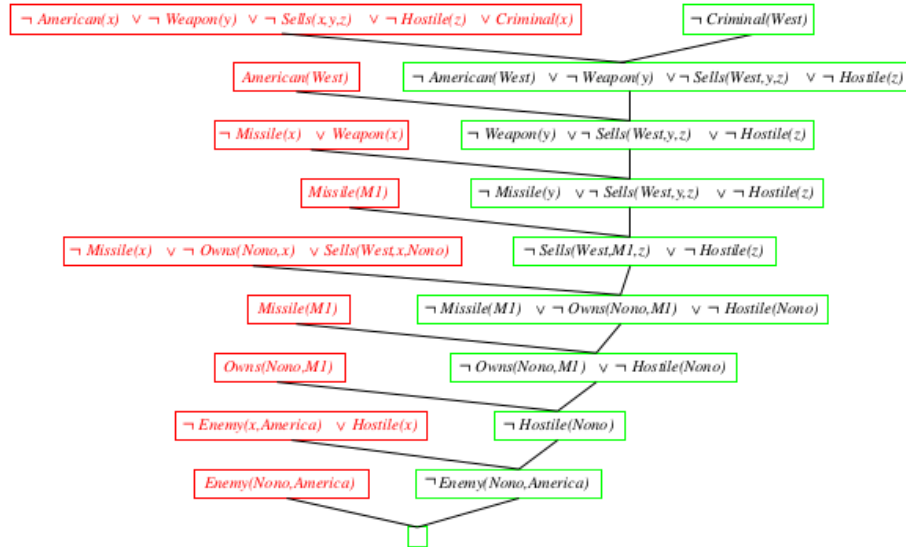
function FOL-FC-Ask( $KB, \alpha$ ) returns a substitution or false
  repeat until  $new$  is empty
     $new \leftarrow \{ \}$ 
    for each sentence  $r$  in  $KB$  do
       $(p_1 \wedge \dots \wedge p_n \Rightarrow q) \leftarrow \text{STANDARDIZE-APART}(r)$ 
      for each  $\theta$  such that  $(p_1 \wedge \dots \wedge p_n)\theta = (p'_1 \wedge \dots \wedge p'_n)\theta$ 
        for some  $p'_1, \dots, p'_n$  in  $KB$ 
           $q' \leftarrow \text{SUBST}(\theta, q)$ 
          if  $q'$  is not a renaming of a sentence already in  $KB$  or  $new$  then do
            add  $q'$  to  $new$ 
             $\phi \leftarrow \text{UNIFY}(q', \alpha)$ 
            if  $\phi$  is not fail then return  $\phi$ 
  add  $new$  to  $KB$ 
  return false

```

Forward Chaining Proof



Proof by Resolution



Summary of Logic

Propositional Logic: atomic statements are facts

- Inference via resolution is sound and complete (though likely computationally intractable)
- Linear-time inference via forward and backward chaining possible for knowledge bases consisting of definite clauses

First-order Logic: adds variables, relations, and quantification

- Inference is essentially a generalization of propositional inference
 - Propositionalization
 - “lifted” inference rules like generalized Modus Ponens
- Resolution is still sound and complete, but semidecidable: not guaranteed to terminate on non-entailed sentences