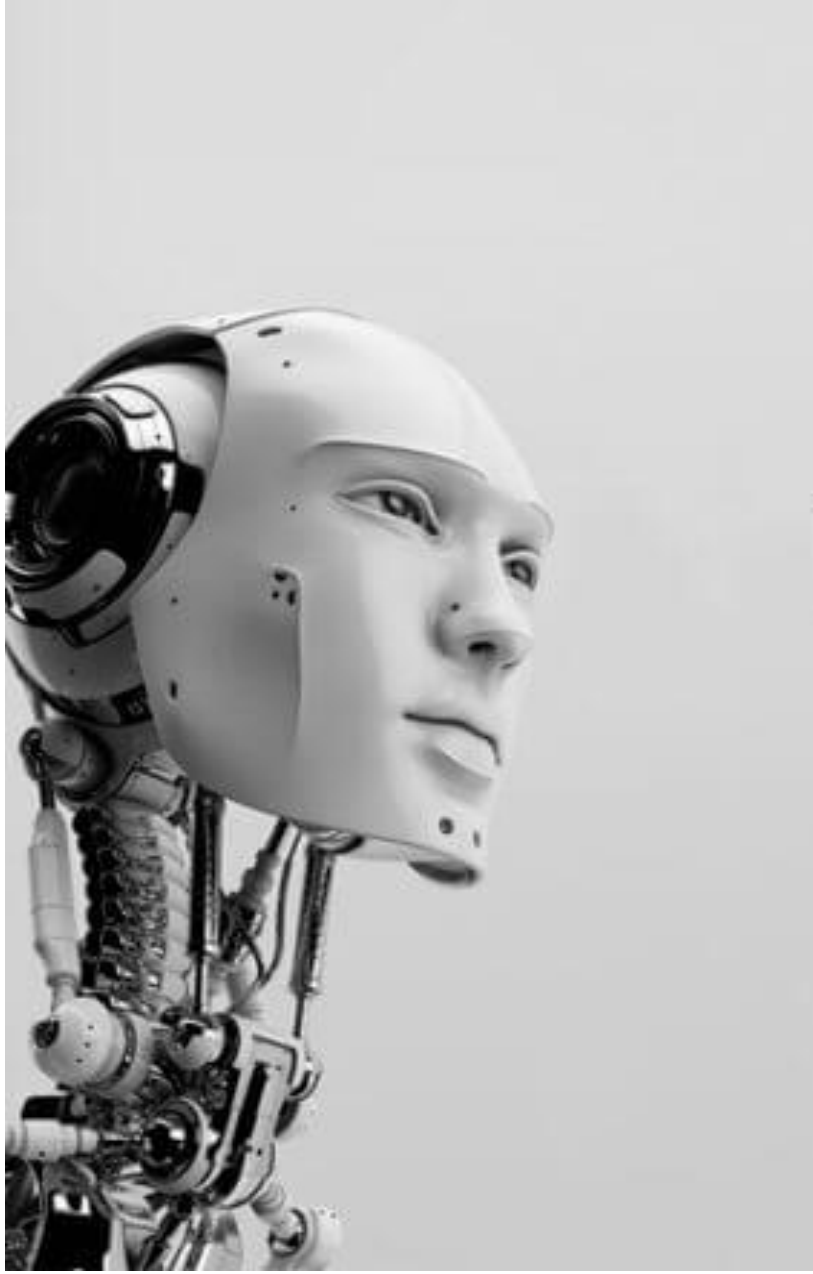


Artificial Intelligence

9 Inference in First-Order Logic

Russell & Norvig, AI: A Modern Approach, 3rd Ed

Lec Zinia Sultana
CSE Dept, MIST

Outline

- Reducing first-order inference to propositional inference
- Unification
- Generalized Modus Ponens
- Forward and backward chaining
- CNF and Resolution

Inference rules for quantifiers

Universal Instantiation

The rule of **Universal Instantiation** (**UI** for short) says that we can infer any sentence obtained by substituting a **ground term** (a term without variables) for the variable.

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

Example: All greedy kings are evil.

∀ x King(x) ∧ Greedy(x) ⇒ Evil(x)	**SUBST({v/g}, α)**
King(John) ∧ Greedy(John) ⇒ Evil(John)	{x/John}
King(Richard) ∧ Greedy(Richard) ⇒ Evil(Richard)	{x/Richard}
King(Father (John)) ∧ Greedy(Father (John)) ⇒ Evil(Father (John))	{x/Father(John)}

Existential Instantiation

In the rule for **Existential Instantiation**, the variable is replaced by a single *new constant symbol*. The formal statement is as follows: for any sentence α, variable v, and constant symbol k that does not appear elsewhere in the knowledge base,

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

Example: King John has a crown on his head.

$\exists$ **x Crown(x) ∧ OnHead(x, John)**	**SUBST({v/k}, α)**
Crown(C1) ∧ OnHead(C1, John)	{x/C1}
Where C1 is a Skolem constant and the process is called skolemization.	

Reduction to propositional inference

❖ **Propositionalization**

For example, suppose our knowledge base contains –

1. $\forall$ x King(x) ∧ Greedy(x) ⇒ Evil(x)
2. King(John)
3. Greedy(John)
4. Brother (Richard, John)

After Universal Instantiation (UI) on 1st Sentence:

i. King(John) ∧ Greedy(John) ⇒ Evil(John)
ii. King(Richard) ∧ Greedy(Richard) ⇒ Evil(Richard)

can we now use the inference rules of proposition?

Unification and Lifting

We have the inference rule

- $\forall\, x\ King(x) \land Greedy(x) \Rightarrow Evil(x)$

We have facts that (partially) match the precondition

– King(John)

– $\forall y$ Greedy(y)

We can infer that, **Evil(John)** – how?

Because we know that John is a king (given) and John is greedy (because everyone is greedy). What we need for this to work is to find a substitution both for the variables in the implication sentence and for the variables in the sentences that are in the knowledge base. In this case, applying the substitution {x/John, y/John} to the implication premises King(x) and Greedy(x) and the knowledge-base sentences King(John) and Greedy(y) will make them identical. Thus, we can infer the conclusion of the implication.

Unification and Lifting

Generalized Modus Ponens – lifted version of Modus Ponens

For atomic sentences pi, pi’, and q, where there is a substitution θ, such that SUBST(θ, pi’) = SUBST(θ, pi), for all i

$$\frac{p_1',\ \ p_2',\ \ \ldots,\ \ p_n',\ \ (p_1 \wedge p_2 \wedge \ldots \wedge p_n \Rightarrow q)}{\mathrm{SUBST}(\theta, q)}$$

KB	Atomic Sentences		θ	Conclusion: SUBST(θ, q)
∀ x King(x) ∧ Greedy(x) ⇒ Evil(x) King(John) Greedy(John)	P1	King(x)	{x/John , y/John}	Evil(John) .
	P2	Greedy(x)		
	P1’	King(John)		
	P2’	Greedy(John)		
	q	Evil(x)		

Unification

The process of lifted inference rules require finding substitutions that make different logical expressions look identical known as unification.

UNIFY(p, q)=θ where SUBST(θ, p)= SUBST(θ, q)
E.g. UNIFY(Knows(John, x), Knows(John, Jane)) = {x/Jane}

Finding all sentences in the KB that unify with Knows(John, x):

p	q	θ
Knows(John, x)	Knows(y, Bill))	{x/Bill, y/John}
Knows(John, x)	Knows(y, Mother (y))	{y/John, x/Mother (John)}
Knows(John, x)	Knows(x, Elizabeth)	fail

Forward Chaining

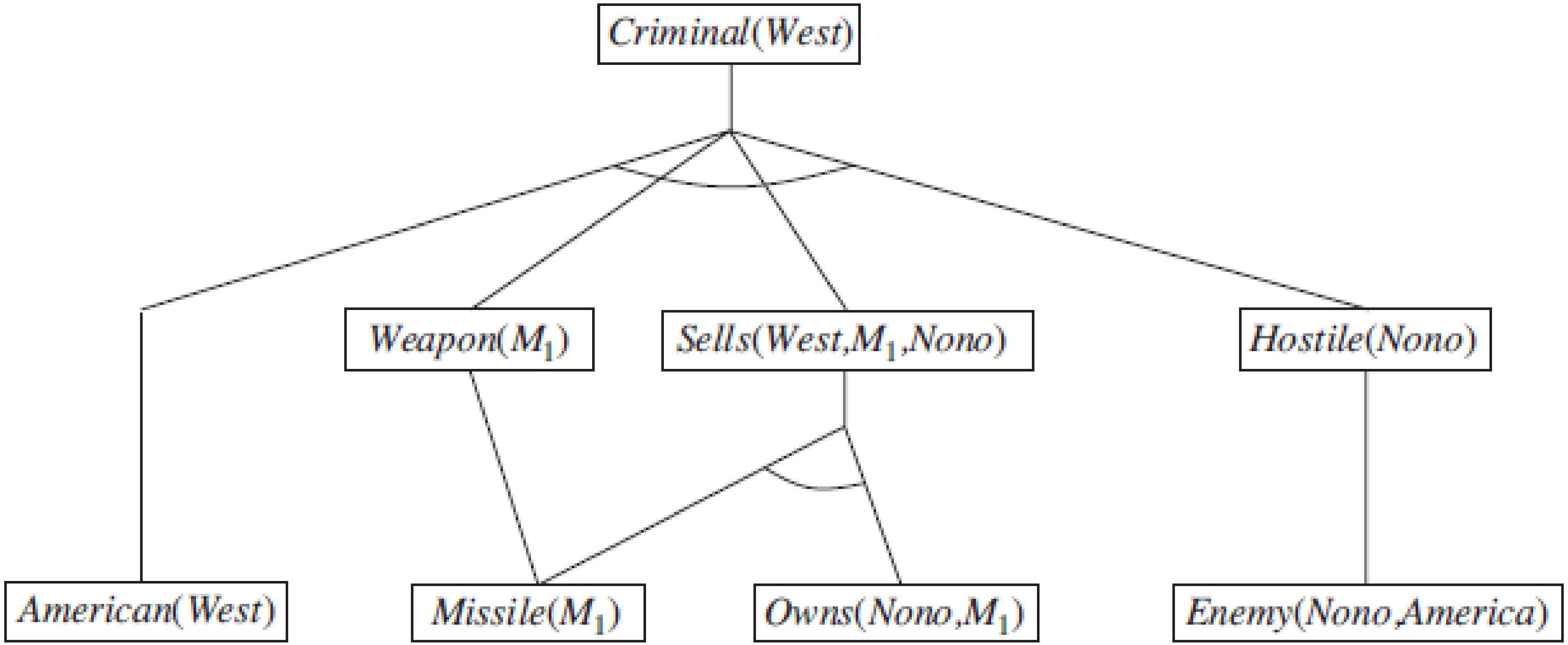
The law says that it is a crime for an American to sell weapons to hostile nations. The country Nono, an enemy of America, has some missiles, and all of its missiles were sold to it by Colonel West, who is American.

Using Forward Chaining prove that, “West is a criminal” – **Self Study**

Forward Chaining

1. It is a crime for an american to sell weapons to hostile nations
   American(x) ∧ weapon(y) ∧ sells(x, y, z) ∧ hostile(z) ⇒ criminal (x)
2. Nono has some missiles
   ∃x owns(nono, x) ∧ missile(x)
      Owns(nono,m1)
      Missile(m1)
3. All of its missiles were sold to it by colonel west
   Missile(x) ∧ owns(nono, x) ⇒ sells(west, x, nono)
4. Missiles are weapons: Missile(x) ⇒ weapon(x)
5. America counts as “hostile”: Enemy(x, america) ⇒ hostile(x)
6. “West, who is american . . .”: American(west)
7. “The country nono , an enemy of america . . .”:
   Enemy(nono, America)

Forward Chaining



CNF - First Order Logic

First-order resolution requires that sentences be in conjunctive normal form (CNF) that is, a conjunction of clauses, where each clause is a disjunction of literals.

$\forall x$ American(x) $\land$Weapon(y) $\land$ Sells(x, y, z) $\land$ Hostile(z) $\Rightarrow$ Criminal (x)

In CNF,

$\neg$American(x) $\lor$ $\neg$Weapon(y) $\lor$ $\neg$Sells(x, y, z) $\lor$ $\neg$Hostile(z) $\lor$ Criminal (x) .

CNF - Rules

1. Eliminate bi-conditionals and implications:
   i. Eliminate $\Leftrightarrow$, replacing $\alpha \Leftrightarrow \beta$ with $(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$.
   ii. Eliminate $\Rightarrow$, replacing $\alpha \Rightarrow \beta$ with $\neg\alpha \vee \beta$. 2.
2. Move $\neg$ inwards:
   i. $\neg(\forall\ x\ p) \equiv \exists\ x\ \neg p$,
   ii. $\neg(\exists\ x\ p) \equiv \forall\ x\ \neg p$,
   iii. $\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$,
   iv. $\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$,
   v. $\neg\neg\alpha \equiv \alpha$.
3. Standardize variables apart by renaming them: each quantifier should use a different variable.

CNF - Rules

4. Skolemize: each existential variable is replaced by a Skolem constant or Skolem function of the enclosing universally quantified variables.
   i. For instance, $\exists x$ Rich(x) becomes Rich(G1) where G1 is a new Skolem constant.
   ii. "Everyone has a heart"
   $\forall x\ Person(x) \Rightarrow \exists y\ Heart(y) \wedge Has(x, y)$
   becomes $\forall\ x\ Person(x) \Rightarrow Heart(H(x)) \wedge Has(x, H(x))$,
   where H is a new symbol (Skolem function).
5. Drop universal quantifiers
   i. For instance, $\forall x$ Person(x) becomes Person(x).
6. Distribute $\wedge$ over $\vee$:
   i. $(\alpha \wedge \beta) \vee \gamma \equiv (\alpha \vee \gamma) \wedge (\beta \vee \gamma)$.

CNF - Example

Everyone who loves all animals is loved by someone

$\forall$ x [$\forall$ y Animal(y) $\Rightarrow$ Loves(x, y)] $\Rightarrow$ [$\exists$ y Loves(y, x)]

- Convert in CNF.

CNF - Example

$\forall x\ [\forall y\ Animal(y) \Rightarrow Loves(x, y)] \Rightarrow [\exists y\ Loves(y, x)]$

1. **Eliminate implications**:
   $\forall x\ [\neg\forall y\ \neg Animal(y) \lor Loves(x, y)] \lor [\exists y\ Loves(y, x)]$

CNF - Example

$\forall\ x\ [\neg\forall\ y\ \neg Animal(y) \lor Loves(x, y)] \lor [\exists\ y\ Loves(y, x)]$

2. **Move $\neg$ inwards**

   $\forall\ x\ [\exists\ y\ \neg(\neg Animal(y) \lor Loves(x, y))] \lor [\exists\ y\ Loves(y, x)]$

   $\forall\ x\ [\exists\ y\ \neg\neg Animal(y) \land \neg Loves(x, y)] \lor [\exists\ y\ Loves(y, x)]$

   $\forall\ x\ [\exists\ y\ Animal\ (y) \land\neg Loves(x, y)] \lor [\exists\ y\ Loves(y, x)]$

CNF - Example

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

3. **Standardize variables**
   $\forall$ x [$\exists$ y Animal (y) $\wedge \neg$Loves(x, y)] $\vee$ [$\exists$ z Loves(z, x)]

4. **Skolemize**:
   $\forall$ x [Animal (F(x)) $\wedge \neg$Loves(x, F(x))] $\vee$ Loves(G(z), x)

5. **Drop universal quantifiers**
   [Animal (F(x)) $\wedge$ $\neg$Loves(x, F(x))] $\vee$ Loves(G(z), x)

6. **Distribute** $\vee$ **over** $\wedge$:
   [Animal (F(x)) $\vee$ Loves(G(z), x)] $\wedge$ [$\neg$Loves(x, F(x)) $\vee$ Loves(G(z), x)]

Resolution

Propositional literals are complementary if one is the negation of the other; first-order literals are complementary if one *unifies with* the negation of the other.

$$\frac{\ell_1 \vee \cdots \vee \ell_k, \qquad m_1 \vee \cdots \vee m_n}{\text{SUBST}(\theta, \ell_1 \vee \cdots \vee \ell_{i-1} \vee \ell_{i+1} \vee \cdots \vee \ell_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n)}$$

Where, UNIFY(li, ¬mj) = θ

[Animal (F(x)) ∨ Loves(G(x), x)]

[¬Loves(u, v) ∨ ¬Kills(u, v)]

UNIFY(Loves(G(x), x), ¬Loves(u, v)) = θ ={u/G(x), v/x}

[Animal (F(x)) ∨ ¬Kills(G(x), x)]

Resolution

1. Convert the facts into First Order Logic
2. Convert the FOL into CNF
3. Assume negation of the goal is true
4. Apply resolution until derive the empty clause

Resolution - Example

Everyone who loves all animals is loved by someone. Anyone who kills an animal is loved by no one. Jack loves all animals. Either Jack or Curiosity killed the cat, who is named Tuna.

Did Curiosity kill the cat?

FOLs:

A. $\forall x\ [\forall y\ Animal(y) \Rightarrow Loves(x, y)] \Rightarrow [\exists y\ Loves(y, x)]$

B. $\forall x\ [\exists z\ Animal(z) \land Kills(x, z)] \Rightarrow [\forall y\ \neg Loves(y, x)]$

C. $\forall x\ Animal(x) \Rightarrow Loves(Jack, x)$

D. $Kills(Jack, Tuna) \lor Kills(Curiosity, Tuna)$

E. $Cat(Tuna)$

F. $\forall x\ Cat(x) \Rightarrow Animal(x)$

Resolution – Example . .

After applying CNF:

A1. Animal(F(x)) ∨ Loves(G(x), x)

A2. ¬Loves(x, F(x)) ∨ Loves(G(x), x)

B. ¬Loves(y, x) ∨ ¬Animal (z) ∨ ¬Kills(x, z)

C. ¬Animal(x) ∨ Loves(Jack, x)

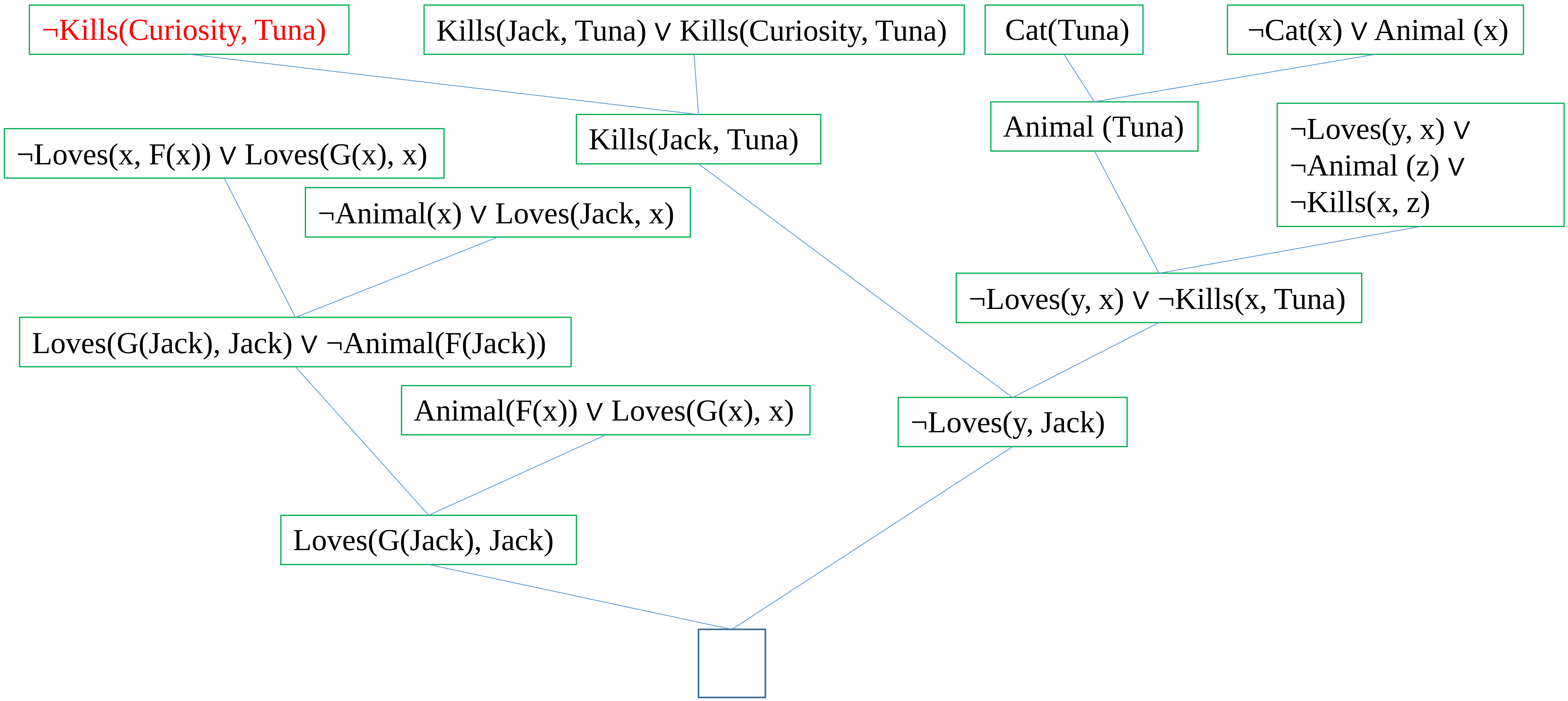
D. Kills(Jack, Tuna) ∨ Kills(Curiosity, Tuna)

E. Cat(Tuna)

F. ¬Cat(x) ∨ Animal (x)

F. ¬Kills(Curiosity, Tuna) - negated goal in CNF form

Resolution – Example . . .



Thank you ☺