TDT4136 Introduction to Artificial Intelligence

Summary of Lecture 9 - Inference in First Order Logic

Chapter 9 in the textbook

Liyuan Xing

Adjunct Associate Professor @ NAIL/IDI/NTNU
Machine Learning Engineer @ TrønderEnergi

2022

Outline

- Reduction of FOL to propositional logic (and use propositional inference methods)
- Unification and First-Order Inference
- Forward/Backward Chaining by generalized Modus Ponens, on Definite Clauses
- Inference using Resolution by generalized resolution rule, on CNF representation

Inference by Reduction to Propositionalized Sentences

- One inference approach is to propositionalize the sentences in KB and then apply inference methods (e.g., resolution refutation) used in propositional logic
  - Every FOL KB can be propositionalized so as to preserve entailment
  - A sentence is entailed by new KB iff it is entailed by the original KB

Propositionalization process:

Every KB in FOL can be "propositionalized"

- instantiate universal quantification
- instantiate existential quantification

Propositionalization

- Suppose the KB contains just the following:

$$\forall x King(x) \land Greedy(x) \implies Evil(x)$$
$$King(John)$$
$$Greedy(John)$$
$$Brother(Richard, John)$$

- To "propositionalize", we need to get rid of **quantifiers** and **variables**.

Propositionalization of Universal Quantification

- We instantiate the universal sentence in *all possible* ways.
  - The "Substitution" rule for instantiation of variables (for "propositionalization):

$$\frac{\forall x \;\; \alpha}{\textsc{Subst}(\{x/g\}, \alpha)}$$

variable x in the sentence α is substituted with a ground term g [1]

- Variable x is *substituted* with the *ground terms* referring to the objects *John* and *Richard* in the model one by one.

[1] A term is used to denote an object in the world. It can be a constant, variable or a Function(term1, ..., termn). A ground term is a term with no variables.

Propositionalization of Universal Quantif.- cont.

- Suppose the KB contains just the following:

  $\forall x \ \ King(x) \wedge Greedy(x) \implies Evil(x)$
  King(John)
  Greedy(John)
  Brother(Richard,John)

- How to propositionalize the first sentence?

Propositionalization of Existential Quantification

- Through **Skolemization**: Each existentially quantified variable is replaced by a *Skolem constant* or a *Skolem function*.
- *Skolem Constant*: if the existential variable is not within the scope of any universaly quantified variable. Every instance of the existentially quantified variable is replaced with the same unique constant, a brand new one that does not appear elsewhere in the knowledge base.

$$\frac{\exists x \;\; \alpha}{\textsc{Subst}(\{x/k\}, \alpha)}$$

Example: $\exists y \;\; (P(y) \land Q(y))$ is converted to: $P(CC) \land Q(CC)$

Skolemization Function

- *Skolem Function*: If the existentiail quantifier is in the **scope** (i.e., "inside") of a (or more) universally quantified variable(s), then replace it with a unique n-ary function over these universally quantified variables. Remove then the existential quantifier.

E.g., $\forall x \ \exists y \ (P(x) \vee Q(y))$ converted to: $\forall x \ P(x) \vee Q(F(x))$

Problems with propositionalization

- Propositionalization seems to generate lots of irrelevant sentences. E.g., from

$$\forall x \;\; King(x) \land Greedy(x) \implies Evil(x)$$
$$King(John)$$
$$Greedy(John)$$
$$Brother(Richard, John)$$

- It seems obvious that $Evil(John)$ will be inferred at the end, but propositionalization produces lots of facts such as $Greedy(Richard)$ that are irrelevant
- With p k-ary predicates and n constants, there are $p \cdot n^k$ instantiations

Inference without propositionalization

- Problem : Universal elimination gives us (too) many opportunities for substituting variables with ground terms
- Solution: avoid making blind substitution of ground terms
  - Make substitutions that help to advance inferences
  - i.e., use substitutions matching "similar" sentences in KB
- How?
- UNIFICATION: takes two similar sentences and computes the **unifier** for them (a **substitution**) that makes them look the same, if it exists
  UNIFY(P,Q) $= \theta$; where SUBST (θ, P)= SUBST (θ, Q)

Adapting Modus Ponens to FOL

Modus Ponens in propositional logic:

$$\frac{\alpha \rightarrow \beta \qquad \alpha}{\beta}$$

Generalized Modus Ponens - FOL

$$\frac{P_1{}',\ \ P_2{}',\ \ldots,\ P_n{}',\ \ (P_1 \wedge P_2 \wedge \ldots \wedge P_n \Rightarrow Q)}{Q\theta}$$

where $P_i{}'\theta = P_i\theta$ for all i , θ is Substitution

Adapting Modus Ponens to FOL

- Suppose this KB:

$$S1 : \forall x \;\; King(x) \land Greedy(x) \implies Evil(x)$$
$$S2 : King(John)$$
$$S3 : \forall y \;\; Greedy(y)$$

- We want to find if John is Evil.
- Make a **substitution** for values of variables that make the premise of the implication identical to the sentences in the knowledge base.
  - ${P_1}'$ is King(John) and P_1 is King(x)
  - ${P_2}'$ is Greedy(y) and P_2 is Greedy(x)
  - θ is $\{x/John, y/John\}$,
  - Q is Evil(x)
  - SUBS(θ,Q) is Evil(John)

Lifted Inference Rules and Unification

- Generalized Modus ponens is the *lifted* version of Modus Ponens(i.e., variable free version)
- Lifted Inferences need finding substitutions that make 2 sentences look identical.
- This is called **Unification**

Unification process

- Unify procedure: Unify(P,Q) takes two atomic (i.e. single predicates) sentences P and Q and returns a substitution that makes P and Q identical.
- The aim is be able to match literals even when they have variables.
- Rules for substitutions: Can replace a variable
  - by a constant.
  - by a variable.
  - by a function expression, as long as the function expression does not contain the variable.

Unifier: a substitution that makes two clauses resolvable. e.g., θ:$\{x_1/M; x_2/x_3; x_4/F(..)\}$

Most General Unifier

- Our aim is to be able to match conflicting literals (for the use of resolution), even when they have variables. Unification process determines whether there is a"specialization" that matches.
- However, we don't want to over specialize.

Inference through Forward Chaining in FOL

- Forward Chaining is an important inference in FOL - without propositionalisation
- FC uses definite clauses. In FOL
  - A definite clause: either atomic, or implication
  - Existential quantifiers are not allowed
  - Universal quantifications are implicit.
  - Example sentences in DC:

    King (John) - literal
    King(x) $\rightarrow$ Evil(x)
    Evil(x). (i.e., everyone is evil). - literal with variable

Example on Forward Chaining in FOL

Suppose we have the following knowledge in the KB:

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.

We want to Prove that "**Colonel West is a criminal".**

Colonel West example cont.

- KB: 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.
- We need
  - to translate these natural language sentences to FOL
  - convert these sentences in the KB to Definite Clauses if they are not in DC form by eliminating quantifiers:
    - remove $\forall$
    - eliminate $\exists$ by skolemization

Example knowledge base contd.

S1: It is a crime for an American to sell weapons to hostile nations:

$\forall x, y, z \ American(x) \wedge Weapon(y) \wedge Sells(x, y, z) \wedge Hostile(z) \implies Criminal(x)$

S1: $American(x) \wedge Weapon(y) \wedge Sells(x, y, z) \wedge Hostile(z) \implies Criminal(x)$

S2: Nono . . . has some missiles

Example knowledge base contd.

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

S1: $American(x) \wedge Weapon(y) \wedge Sells(x, y, z) \wedge Hostile(z) \implies Criminal(x)$

S2: Nono ... has some missiles

$\exists\, x \; Owns(Nono, x) \wedge Missile(x)$:

S2: $Owns(Nono, M_1)$ and $Missile(M_1)$

S3: ... all of its missiles were sold to it by Colonel West

Forward chaining algorithm - abstract

- Similar to Propositional Logic, restriction on sentences - definite clause.
- Find and use rules that have as premises the know facts in the KB.
  - In Propositional Logic: $A \wedge B \implies C$
  - In FOL: King(x) $\wedge$ Greedy(x) $\implies$ Evil(x)
  - FOL Needs Unification to match sentences.
- Continue matching and deriving consequences of sentences until deriving the goal.

Forward chaining proof of Colonel West

Our KB:

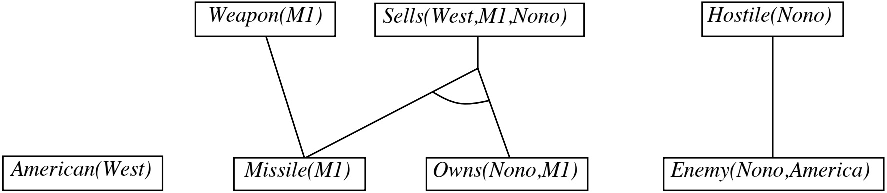
1. $American(x) \wedge Weapon(y) \wedge Sells(x, y, z) \wedge Hostile(z) \implies Criminal(x)$
2. $Owns(Nono, M_1)$
3. $Missile(M_1)$
4. $Missile(x) \Rightarrow Weapon(x)$
5. $Missile(x) \wedge Owns(Nono, x) \implies Sells(West, x, Nono)$
6. $Enemy(x, America) \implies Hostile(x)$
7. $American(West)$
8. $Enemy(Nono, America)$

Forward chaining proof of Colonel West

$\boxed{American(West)}$ $\boxed{Missile(M1)}$ $\boxed{Owns(Nono,M1)}$ $\boxed{Enemy(Nono,America)}$

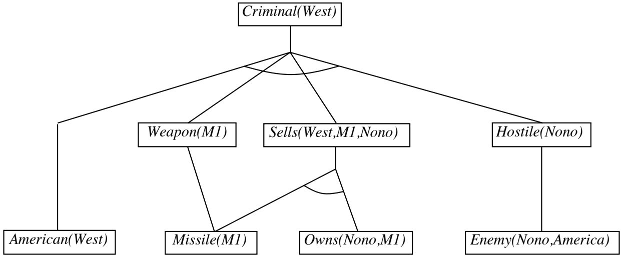
1. $American(x) \wedge Weapon(y) \wedge Sells(x, y, z) \wedge Hostile(z) \implies Criminal(x)$
2. $Owns(Nono, M_1)$
3. $Missile(M_1)$
4. $Missile(x) \Rightarrow Weapon(x)$
5. $Missile(x) \wedge Owns(Nono, x) \implies Sells(West, x, Nono)$
6. $Enemy(x, America) \implies Hostile(x)$
7. $American(West)$
8. $Enemy(Nono, America)$

Forward chaining proof



1. $American(x) \wedge Weapon(y) \wedge Sells(x, y, z) \wedge Hostile(z) \implies Criminal(x)$
2. $Owns(Nono, M_1)$
3. $Missile(M_1)$
4. $Missile(x) \Rightarrow Weapon(x)$
5. $Missile(x) \wedge Owns(Nono, x) \implies Sells(West, x, Nono)$
6. $Enemy(x, America) \implies Hostile(x)$
7. $American(West)$
8. $Enemy(Nono, America)$

Forward chaining proof



1. $American(x) \land Weapon(y) \land Sells(x, y, z) \land Hostile(z) \implies Criminal(x)$
2. $Owns(Nono, M_1)$
3. $Missile(M_1)$
4. $Missile(x) \Rightarrow Weapon(x)$
5. $Missile(x) \land Owns(Nono, x) \implies Sells(West, x, Nono)$
6. $Enemy(x, America) \implies Hostile(x)$
7. $American(West)$
8. $Enemy(Nono, America)$

Example Resolution Refutation - same example

Now we solve the same Colonel West example using resolution, more correctly resolution refutation.

First, we look at Resolution rule in FOL.

Resolution Rule

Full first-order version:

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

where $\textsc{Unify}(\ell_i, \neg m_j) = \theta$.

Two standardized clauses can be resolved if they contain complementary literals (one is the negation of the other).

FOL literals are complementary if one *unifies* with the negation of the other.

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

Conversion to CNF - example

We'll soon start solving Colonel West example using Resolution Refutation.

- The idea is the same as in Propositional Logic:
  Our goal is to determine if KB $\models \alpha$:
  1. Add $\neg\alpha$ to the KB
  2. Convert KB and α to Conjunctive Normal Form
  3. Use the resolution rule and search to determine whether the system is satisfiable (SAT)
- Let us look at the procedure for conversion of FOL sentences to CNF through an example.

Conversion to CNF - example

Everyone who loves all animals is loved by someone:

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

1. Eliminate biconditionals and implications

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

2. Move $\neg$ inwards: $\neg\forall x, p \ \equiv \exists x\ \neg p$, $\ \neg\exists x, p \ \equiv \forall x\ \neg p$:

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

Conversion to CNF contd.

3. Standardize variables: each quantifier should use a different variable name

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

4. Skolemize: Each existential variable is replaced by a Skolem function of the enclosing universally quantified variables:

$$\forall x \;\; [Animal(F(x)) \land \neg Loves(x, F(x))] \lor Loves(G(x), x)$$

5. Drop universal quantifiers:

$$[Animal(F(x)) \land \neg Loves(x, F(x))] \lor Loves(G(x), x)$$

6. Distribute $\land$ over $\lor$:

$$[Animal(f(x)) \lor Loves(G(x), x)] \land [\neg Loves(x, F(x)) \lor Loves(G(x), x)]$$

Colonel West example using Resolution Refutation

- KB:
  $American(x) \wedge Weapon(y) \wedge Sells(x, y, z) \wedge Hostile(z) \implies Criminal(x)$
  $Owns(Nono, M_1)$ and $Missile(M_1)$
  $Missile(x) \Rightarrow Weapon(x)$
  $Missile(x) \wedge Owns(Nono, x) \implies Sells(West, x, Nono)$
  $Enemy(x, America) \implies Hostile(x)$
  $American(West)$
  $Enemy(Nono, America)$

- Question: Is colonel West criminal?

Colonel West example in CNF form

1. $\neg American(x) \lor \neg Weapon(y) \lor \neg Sells(x, y, z) \lor \neg Hostile(z) \lor Criminal(x)$
2. $\neg Missile(x) \lor \neg Owns(Nono, x) \lor Sells(West, x, Nono)$
3. $\neg Enemy(x, America) \lor Hostile(x)$
4. $\neg Missile(x) \lor Weapon(x)$

- Facts: $American(West), Owns(Nono, M1), Missile(M1), Enemy(Nono, America$
- Add $\neg Criminal(West)$

Colonel West example using Resolution Refutation - cont