Introduction to AI: Tutorial

Foundation of Logic and Resolution-based Proof Procedures

Alessandra Russo

The aim of this tutorial is to enable you to practise more with fundamental logic-based concepts, introduced in Unit 2, and to gain more practice with SLD and SLDNF derivations from KB expressed using definite clauses and normal clauses.

Question 1

Convert the following first-order sentences into clausal representation:

i) $\forall X \exists Y\, friend(X,Y) \rightarrow \exists V\, happy(V)$

ii) $\forall Y (\exists X\, P(X,Y) \rightarrow \neg S(Y))$

iii) $\forall X (philosopher(X) \rightarrow \exists Y (book(Y) \wedge write(X,Y)))$

Solution

1. $\forall \mathbf{X} \exists \mathbf{Y} \mathbf{friend}(\mathbf{X},\mathbf{Y}) \rightarrow \exists \mathbf{V} \mathbf{happy}(\mathbf{V})$

$\neg(\forall X \exists Y\, friend(X,Y)) \vee \exists V\, happy(V)$	remove implications
$\exists X (\forall Y \neg friend(X,Y)) \vee \exists V\, happy(V)$	push negation next to atoms
$\forall Y \neg friend(sk1,Y) \vee happy(sk2)$	eliminate existential quantifiers
$\neg friend(sk1,Y) \vee happy(sk2)$	remove universal quantifiers

2. $\forall \mathbf{Y} (\exists \mathbf{X} \mathbf{P}(\mathbf{X},\mathbf{Y}) \rightarrow \neg \mathbf{S}(\mathbf{Y}))$

$\forall Y (\neg \exists X\, P(X,Y) \vee \neg S(Y))$	remove implications
$\forall Y (\forall X \neg P(X,Y) \vee \neg S(Y))$	push negation next to atoms
$\forall Y \forall X (\neg P(X,Y) \vee \neg S(Y))$	move universal quantifiers to the front
$\neg P(X,Y) \vee \neg S(Y)$	remove universal quantifiers

3. $\forall \mathbf{X} (\mathbf{philosopher}(\mathbf{X}) \rightarrow \exists \mathbf{Y} (\mathbf{book}(\mathbf{Y}) \wedge \mathbf{write}(\mathbf{X},\mathbf{Y})))$

$\forall X (\neg philosopher(X) \vee \exists Y (book(Y) \wedge write(X,Y)))$	remove implications
$\forall X (\neg philosopher(X) \vee (book(g(X)) \wedge write(X,g(X))))$	eliminate existentials
$\neg philosopher(X) \vee (book(g(X)) \wedge write(X,g(X)))$	remove universals
$\{\neg philosopher(X) \vee book(g(X)); \neg philosopher(X) \vee write(X,g(X))\}$	

Question 2

If possible unify the following pairs and give the unification ϕ, otherwise explain why they do not unify:

1) $p(f(X), g(Y))$ and $p(Z, g(f(a)))$
2) $p(Y, a, b, Y)$ and $p(c, F, G, F)$
3) $p(X, X)$ and $p(E, E)$
4) $p(f(X))$ and $q(f(X))$
5) $p(V, g(X))$ and $p(f(X), V)$

Solution

1. $p(f(X), g(Y))$ and $p(Z, g(f(a)))$
   unify with $\phi = [Z = f(X), Y = f(a)]$.

2. $p(Y, a, b, Y)$ and $p(c, F, G, F)$
   does not unify because $Y = c$ and $F = a$ but if $F = Y$ then it must be that $a = c$ which is not possible.

3. $p(X, Y)$ and $p(E, E)$
   they unify and the most general unifer is $\phi = [X = Y = E]$.

4. $p(f(X))$ and $q(f(X))$
   do not unify as they use different predicate symbols.

5. $p(V, g(X))$ and $p(f(X), V)$
   do not unify as it cannot be that $V = f(X)$ and $V = g(X)$ since $f(X)$ and $g(X)$ have different functors.

Question 3

Consider a knowledge base (KB) about the following sentences:

```
(a) Lucy is a professor.
(b) All professor are people.
(c) John is a dean.
(d) Deans are professors.
(e) All professor consider the dean a friend or they don't know him.
```

1. Formalise each of the above sentence into first-order logic

2. Convert them into clausal form.

3. Let KB be the set of clauses that you have given in your answer to part (2) above. Write KB in rule form.

4. Assume the only constants to be Lucy and John. Write in full the $ground(KB)$ (i.e. the grounding of KB).

5. Give the Herbrand base of KB.

6. Give the Least Herbrand model of KB, and an example of an Herbrand interpretation that is not a model of KB.

7. Using resolution show that $KB \not\models friendOf(lucy, john)$. Explain also semantically why this is the case.

Solution

1. Each of the above sentences can be fomalised in first-order logic in the following way:

   a. $prof(lucy)$.

   b. $\forall X(prof(X) \rightarrow person(X))$

   c. $dean(john)$.

   d. $\forall X(dean(X) \rightarrow prof(X))$

   e. $\forall X \forall Y(prof(X) \wedge dean(Y) \rightarrow friendOf(X, Y) \vee \neg know(X, Y))$

2. The above formulae are converted in clausal form as follows:

   a. $prof(lucy)$.

   b. $\neg prof(X) \vee person(X)$

   c. $dean(john)$.

   d. $\neg dean(X) \vee prof(X))$

   e.
   $\forall X(\forall Y(prof(X) \wedge dean(y) \rightarrow friendOf(X, Y) \vee \neg know(X, Y)))$
   remove implications
   $\forall X(\forall Y(\neg(prof(X) \wedge dean(y)) \vee friendOf(X, Y) \vee \neg know(X, Y)))$
   push negation next to atoms
   $\forall X(\forall Y(\neg prof(X) \vee \neg dean(y) \vee friendOf(X, Y) \vee \neg know(X, Y)))$
   removing the universal quantifiers
   $\neg prof(X) \vee \neg dean(y) \vee friendOf(X, Y) \vee \neg know(X, Y)$

3. The KB is therefore given by the following clauses written in rule form:

$$KB = \left\{ \begin{array}{l} prof(lucy). \\ person(X) \leftarrow prof(X) \\ dean(john). \\ prof(X) \leftarrow dean(X). \\ friend(X, Y) \leftarrow prof(X), dean(Y), know(X, Y) \end{array} \right\}$$

4.

$$ground(KB) = \left\{ \begin{array}{l} prof(lucy). \\ person(lucy) \leftarrow prof(lucy) \\ person(john) \leftarrow prof(john) \\ dean(john). \\ prof(lucy) \leftarrow dean(lucy). \\ prof(john) \leftarrow dean(john). \\ friend(john, john) \leftarrow prof(john), dean(john), know(john, john) \\ friend(john, lucy) \leftarrow prof(john), dean(lucy), know(john, lucy) \\ friend(lucy, john) \leftarrow prof(lucy), dean(john), know(lucy, john) \\ friend(lucy, lucy) \leftarrow prof(lucy), dean(lucy), know(lucy, lucy) \end{array} \right\}$$

5. The Herbrand base of KB, denoted as HB(KB), is given by all possible ground atoms that can be constructed, given the signature of KB.

   HB(KB)={prof(lucy), prof(john), person(lucy), person(john), dean(lucy), dean(john), friend(lucy, lucy), friend(john, lucy), friend(john, john), friend(lucy, john), know(john, john), kwo(john, lucy), know(lucy, lucy), know(john, john)}

6. The Least Herbrand model of KB, denoted as LHM(KB), is the smallest subset of the Herbrand Base of KB that satisfies KB.

   LHM(KB) = {prof(lucy), prof(john), dean(john), person(lucy), person(john)}

   Any subset of LHM(KB) is an Herbrand interpretation of KB that does not satisfies KB.

7. To show that $KB \not\models friendOf(lucy, john)$, we need to show that $KB, \neg friendOf(lucy, john) \not\models []$. This means proving that that a resolution proof does not terminate with an empty clause $KB, \neg friendOf(lucy, john) \not\vdash []$. This is shown in Figure 1. Semantically, $friendOf(lucy, john)$ is false in the LHM(KB), hence there exists a model of KB that does not satisfy $friendOf(lucy, john)$.

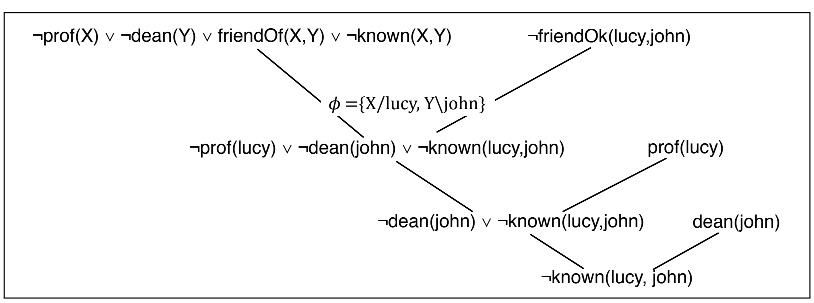

Figure 1: ResolutionTREE for Question 3, part (7)

Question 4

Consider the following KB. Give the SLDNF tree of all derivations of the goal $p(X)$ from KB.

$$KB = \left\{ \begin{array}{l} p(X) \leftarrow not\ q(X), s(X,Y) \\ q(X) \leftarrow not\ r(X). \\ r(a). \\ r(b). \\ s(a,b). \\ s(c,b). \end{array} \right\}$$

Solution The SLDNF tree of all derivations of $KB \vdash P(X)$ is shown in Figure 2. Only one of the branches succeeds, so the only successful unification is $\{X/a, Y/b\}$ and the anser is $P(a)$.

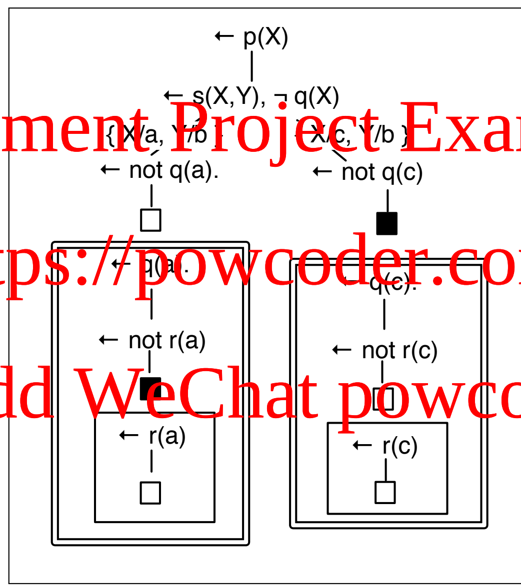

Figure 2: SLDNF tree of all derivation of $KB \vdash P(X)$.

Question 5

Consider the following KB, which formalises the notion that

```
A student passes the year in which he/she is enrolled if
he/she has not failures in any course. John in a student enrolled
in his first year, and Logic is a course.
```

$$KB = \left\{ \begin{array}{l} passedYear(X,Y) \leftarrow year(Y), enrolled(X,Y), not\ failures(X,C) \\ year(firstYear). \\ enrolled(john, firstYear). \\ course(Logic). \end{array} \right\}$$

1. Explain why there does not exist an SLDNF derivation of $passedYear(X, Y)$ from KB.

2. Modify the KB so that it does accept a derivation of $passedYear(X, Y)$ for some unification of X and Y, and give an example of such an SLDNF derivation.

Solution

1. The first clause in KB has a negated condition with a variable C that does not appear in the head of the rule or in any positive condition before. An SLDNF derivation of $passedYear(X, Y)$ from KB will therefore flounder.

2. To resolve the floundering problem, we could redefined the KB, as follows:

$$KB = \left\{ \begin{array}{l} passedYear(X, Y) \leftarrow year(Y), enrolled(X, Y), not\ failures(X) \\ failures(X) \leftarrow course(C), fail(X, C). \\ year(firstYear). \\ enrolled(john, firstYear). \\ course(Logic). \end{array} \right\}$$

The SLDNF derivation of $KB \vdash passYear(X, Y)$ is shown in Figure 3, which shows that $passedYear(john, firstYear)$ is derivable from KB.

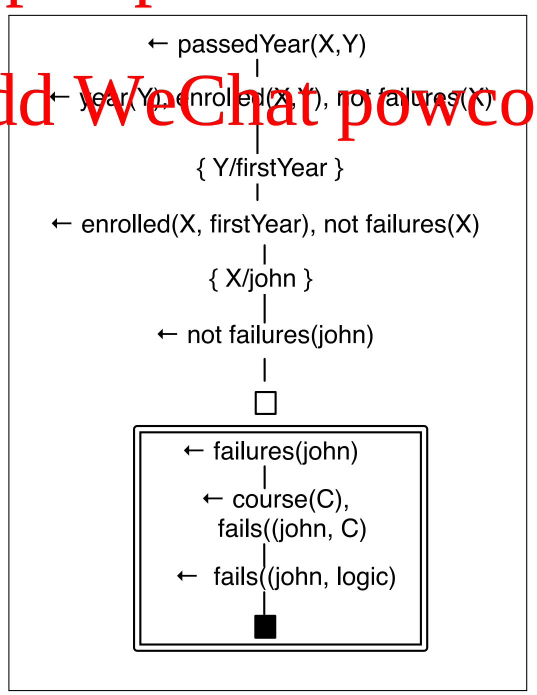

Figure 3: SLDNF derivation of $KB \vdash passedYear(X, Y)$.