Assignment 3 - Predicate Logic and Probabilistic Inference

Artificial Intelligence
WS 2015

Due: 8th January 2016, 23:55 pm

General Information

This exercise consists of theoretical questions only. Upload your solution to MOODLE as PDF or text files (zipped). Scans of handwritten solutions are also fine, but make sure they are **readable**. Do **not** upload `.docx` files or similar.

1 Predicate Logic: Unification

Unify each of the following sets of sentences and give the most general unifier, or give an argument why this is not possible. $Student$ and $Professor$ are predicates, $AICOURSE$, $MATH$ and 150 are constants, $passed$, $enthusiastic$, $owns$, $grade$ and $genius$ are functions[1] and u, v, w, x, y and z are variables. Each unification is worth 0.5 points.

(A) $\{Student(x), Student(y)\}$

(B) $\{Student(passed(w), y), Student(z, grade(passed(u), v))\}$

(C) $\{Student(enthusiastic(x), passed(x), owns(y)), Student(x, y, z, owns(z))\}$

(D) $\{Student(y, enthusiastic(x), MATH), Student(AICOURSE, enthusiastic(x), y)\}$

(E) $\{Student(grade(x, AICOURSE), x), Student(grade(z, u), genius(150, v)),$ $Student(y, genius(v, MATH))\}$

(F) $\{Student(grade(x, y), genius(x)),$ $Student(grade(enthusiastic(z), y), genius(enthusiastic(z)))\}$

(G) $\{Student(y, passed(x), MATH, z), Professor(x, AICOURSE, enthusiastic(x), y)\}$

(H) $\{Student(passed(x), grade(passed(u), v)),$ $Student(passed(grade(w, v)), grade(z, w))\}$

[1] Due to better readability for this exercise we use lowercase letters as function symbols (in the lecture slides uppercase letters are used).

2 Probabilistic Inference using Full Joint Distributions

Consider the joint probability table shown in Table 2.1. Calculate the following probabilities. Each is worth 0.5 points.

(A) $P(hungry)$

(B) $P(\neg cold \mid hungry \wedge cold)$

(C) $P(excited \vee \neg excited)$

(D) $P(hungry \wedge cold \mid crying)$

(E) $P(\neg crying)$

(F) $P(cold \mid hungry)$

(G) $P(cold \mid excited \wedge \neg hungry)$

(H) $P(crying \vee excited)$

(I) $P(excited \wedge \neg hungry)$

(J) $P((excited \wedge cold) \vee (\neg crying \wedge hungry))$

		crying		¬ crying	
		cold	¬ cold	cold	¬ cold
excited	hungry	0.02	0.01	0.02	0.06
	¬ hungry	0.01	0.01	0.05	0.12
¬ excited	hungry	0.05	0.03	0.06	0.14
	¬ hungry	0.03	0.01	0.1	0.28

Table 2.1: *Joint probability of a toddler's behaviour*

3 Knowledge Representation, Forward/Backward Chaining

(A) Write down logical representations for the following sentences. Make sure they are usable for forward and backward chaining. Keep the order of the logical representations the same as the order of the sentences.

- Unicorns, jackalopes and hippocamps are mammals.
- An offspring of a unicorn is a unicorn.
- HappyRainbowDancer is a unicorn.
- HappyRainbowDancer is Greenyboony's parent.
- Offspring and parent are inverse relations.

(2 Points)

(B) Using **Forward Chaining**, can you prove that Greenyboony is a unicorn? Apply the rules in the order you wrote them down. Draw the entire proof tree, including substitutions. If no proof is possible, *explain in detail* why! **(3 Points)**

(C) Using **Backward Chaining**, can you prove that Greenyboony is a unicorn? Apply the rules in the order you wrote them down. Draw the entire proof tree, including substitutions. Make sure to include all failed attemps, if there are any. If no proof is possible, *explain in detail* why! **(3 Points)**

(D) If one of the proofs was not possible, can you modify the knowledge base in a way that enables the proof, without changing the encoded knowledge? Explain! **(1 Point)**

4 More Forward/Backward Chaining

Consider the following knowledge base:

1. $Horse(x) \wedge Fish(x) \implies Hippocamp(x)$
2. $Horse(x) \wedge Horned(x) \implies Unicorn(x)$
3. $Rabbit(z) \wedge Horned(z) \implies Jackalope(z)$
4. $Unicorn(x) \wedge Jackalope(y) \implies Friendlier(x, y)$
5. $Jackalope(y) \wedge Hippocamp(z) \implies Friendlier(y, z)$
6. $Friendlier(x, y) \wedge Friendlier(y, z) \implies Friendlier(x, z)$
7. $Horse(Steve)$
8. $Horse(Greenyboony)$
9. $Horned(Greenyboony)$
10. $Jackalope(Bob)$
11. $Fish(Steve)$

(A) Prove using **Forward Chaining**: $Friendlier(Greenyboony, Steve)$. Apply the rules in the order given. Draw the entire proof tree, including substitutions. If no proof is possible, explain why. **(3 Points)**

(B) Prove using **Backward Chaining**: $Friendlier(Greenyboony, Steve)$. Apply the rules in the order given. Draw the entire proof tree, including substitutions. Make sure to include all failed attemps, if there are any. If no proof is possible, explain why. **(3 Points)**

(C) Imagine there were many more descriptions of different individuals in the knowledge base (e.g. $Horse(HappyRainbowDancer)$, $Horned(HappyRainbowDancer)$, $Rabbit(Bugs)$, $Horned(Bugs)$, etc.). Which algorithm (Forward or Backward Chaining) would you use in this case? *Explain in detail.* **(1 Points)**