First-Order Logic and Generalised Modus Ponens

Kobby Nuamah, Michael Rovatsos

Week 5

1 First-Order Logic

Part 1: Represent the following sentences in first-order logic. You will have to define a vocabulary (which should be consistent between sentences).

2. Every student who takes French passes it.
3. Only one student took Greek in spring 2001.
4. The best score in Greek is always higher than the best score in French.
5. There is a male barber who shaves all the men who do not shave themselves.

Part 2: Write down a first-order logic sentence such that every world in which it is true contains exactly one object.

2 Most General Unifier (MGU)

The most general unifier (MGU) is the least constrained substitution that makes two clauses unify with each other. What is the MGU for each pair of clauses below? If there is no MGU, explain why.

The Unify algorithm in figure 1 (also in R&N Section 9.2, p.328.)

1. \( p(A, B, B) \) and \( p(x, y, z) \)
2. \( q(y, g(A, B)) \) and \( q(g(x, x), y) \)
3. \( \text{older}(\text{father}(y), y) \) and \( \text{older}(\text{father}(x), \text{John}) \)
4. knows(father(y), y) and knows(x, x)

Note that, constants are upper case (e.g. A, B) and variables are lower case (e.g. x, y, z).

3 Generalised Modus Ponens

Part 1: Convert the following sentences to first-order logic formulae suitable for use with Generalised Modus Ponens.

1. Horses, cows and pigs are mammals.
2. An offspring of a horse is a horse.
3. Bluebeard is a horse.
4. Bluebeard is Charlie’s parent.
5. Offspring and parent are inverse relations.

Part 2: Use the sentences to answer a query using a backward-chaining algorithm.

• Draw the proof tree generated by an exhaustive backward-chaining algorithm for the query \( \text{Horse}(h) \), where clauses are matched in the order given.
• How many solutions are a logical consequence of your knowledge base?
• How could we solve this problem?
Figure 1: Unification Algorithm.