Languages for AI, Module 1
July 6, 2021

Time: 2 hours.

In the following $A, B, \ldots$ are propositional variables, $a, b, \ldots$ constant symbols, $f, g, \ldots$ function symbols, $X, Y \ldots$ variables, $p, q, \ldots$ predicate symbols and $F, G, \phi, \psi, \ldots$ formulas (unless differently specified).

1. (4 points) Consider the language of propositional logic. Use natural deduction to prove that the following holds, or find a counter-example to show that it does not hold (remember that $\neg F$ is only a shorthand for $F \to \bot$).

   - $\vdash (F \to G) \to (\neg(F \land \neg G))$
   - $\vdash (F \to G) \to (\neg F \to \neg G)$

   - For the first deduction:

   $$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[F \land \neg G]_1}{F}\land E \qquad [F \to G]_2}{G}\to E \qquad \dfrac{[F \land \neg G]_1}{\neg G}\land E}{\bot}\to E}{\neg(F \land \neg G)}\to I_1}{(F \to G) \to (\neg(F \land \neg G))}\to I_2$$

   $\vdash (F \to G) \to (\neg F \to \neg G)$ does not hold, as one can easily prove by using truth tables.

2. (3 points) Transform the following propositional logic formula into an equivalent formula in Disjunctive Normal Form

   - $\neg(A \to (B \lor C)) \lor ((B \lor A) \to (C \land D))$

   Solution:

   $$\begin{aligned}
   &\neg(A \to (B \lor C)) \lor ((B \lor A) \to (C \land D)) \equiv \\
   &\neg(\neg A \lor (B \lor C)) \lor (\neg(B \lor A) \lor (C \land D) \equiv \\
   &(A \land \neg B \land \neg C) \lor (\neg B \land \neg A) \lor (C \land D) \equiv
   \end{aligned}$$

3. (4 points) Formalize the following sentences in the language of propositional logic. Your formalizations should be as detailed as possible. If there are several equally natural formalizations – e.g., if the sentence is ambiguous – list all of

   1 Alice married and got pregnant.

   2 Anne and Barbara carried the piano and sweated.

   Solution

   1) In some circumstances, it is natural to formalize this as $P \land Q$ where

   P: Alice married.

   Q: Alice got pregnant.

   $P \land Q$ is logically equivalent to the sentence $Q \land P$ which is most naturally read as formalizing 'Alice got pregnant and married'. But in some circumstances, we intend to convey different things with these two sentences; in particular, they might be used to convey information about the temporal order of the relevant events. In this case, the sentence must be formalized simply as P, where

   P: Alice married and got pregnant.

   2) The sentence is ambiguous. It is most naturally read as saying that Anne and Barbara carried the piano together. On the less natural reading, it says that Anne and Barbara each carried the piano individually – presumably at diffeerent moments in time. The first reading can be formalized as $P \land Q \land R$, where

   P: Anne and Barbara carried the piano.

   Q: Anne sweated.

R: Barbara sweated.

The second reading can be formalized as $P1 \wedge P2 \wedge Q1 \ \wedge Q2$, where

P1: Anne carried the piano.

P2: Barbara carried the piano.

Q1: Anne sweated.

Q2: Barbara sweated.

4. (5 points) Discuss whether the following argument is valid.

   Consider the following sentences:

   1 If Alice is happy she forgot Bob's birthday and she is at a party.

   2 Alice forgot Bob's birthday and is at a party or is not happy. Both of them are ambiguous.

   (i) Determine all possible readings of both sentences and formalize all readings of both sentences in the language of propositional logic, using the same dictionary.

   (ii) Discuss whether for some of the provided formalization the argument whose single premise is (a) and whose conclusion is (b) is valid.

   Solution

   There are two readings of (1). According to the first, the consequent of the conditional is 'she forgot Bob's birthday'; according to the second, it is 'she forgot Bob's birthday and she is at a party'. There are also two readings of (2). According to the first, the first disjunct is 'Alice forgot Bob's birthday and is at a party'; according to the second, the second conjunct is '[Bob] is at a party or is not happy'. We will use the following dictionary:

   P: Alice is happy.

   Q: Alice forgot Bob's birthday.

   R: Alice is at a party.

   With this, we can formalize the two readings of (1) using (A1) and (A2), and the two readings of (2) using (B1) and (B2):

   (A1) $(P \rightarrow Q) \wedge R$

   (A2) $P \rightarrow (Q \wedge R)$

   (B1) $(Q \wedge R) \vee \neg P$

   (B2) $Q \wedge (R \vee \neg P)$

   B1 follows logically from A1, and B1 follows logically from A2. This can be shown using partial truth tables.

5. (5 points) Define an appropriate language and formalize the following sentences using FOL formulas.

   1 In order to be able attend the Analysis course its is necessary to have passed the Algebra exam and it is sufficient to have passed the Geometry exam.

   2 In order to attend the Algebra course a student must attend the Geometry course, but if she is attending the Analysis course then it is sufficient that she attends the geometry course

   3 No student passed the Analysis and Geometry exam, unless she had passed the algebra exam.

   Solution:

   Consider a FOL which contain the constants *geometry*, *analysis* and *algebra*, which represent the courses in the obvious way, and the following predicates: $failed(X, Y)$, meaning that the student X has failed the exam of the course Y; $attend(X, Y)$, meaning that the student X is atttending the course Y; $passed(X, Y)$ meaning that the student X has passed the exam of the course Y; $=$ with the obvious meaning. Then we have the following three axioms which formalize the above sentences:

   1 $\forall x.((attend(X, analysis) \rightarrow passed(X, algebra)) \wedge (passed(X, geometry) \rightarrow attend(X, geometry)))$

   2 $\forall X.((attend(X, algebra) \rightarrow attend(X, algebra)) \vee$
   $(attend(X, analysis) \rightarrow (attend(X, geometry) \rightarrow attend(X, algebra))))$

3 $\forall x.(\neg passed(X, algebra) \rightarrow \neg(passed(X, analysis) \wedge passed(X, geometry)))$

6. (5 points) A thief is stealing sculptures in an art gallery. Each sculpture has a specific value and a weight. The thief wants to steal sculptures for more than 600$ of total value, but can only have 11 kilos of sculptures in the sack. The thief must therefore decide what to take and what to leave. Write a CLP or minizinc program to compute which choices thief has, and for each choice, what is the final value of the stolen sculptures. The sculptures are 4 (A, B, C, D): A weights 10kg and it is worth 500$, B weights 4kg and it is worth 600$, C weights 2kg and it is worth 100$, D weights 2kg and it is worth 200$.

   Solution

```
bag(VARIABLES) :-
                length(VARIABLES, 6),
                VARIABLES = [A,B,C,D,W,V],
                [A,B,C,D] ins 0..1,
                % all_distinct(VARIABLES),
        Aw #= A * 10,
                Bw #= B * 4,
        Cw #= C * 2,
        Dw #= D * 2,
        Av #= A * 500,
        Bv #= B * 600,
        Cv #= C * 100,
        Dv #= D * 200,
        W #= Aw + Bw + Cw + Dw,
        V #= Av + Bv + Cv + Dv,
                W #< 12,
        V #> 600.

% query example
?- bag(V), labeling([],V), V=[A,B,C,D].
```

7. (5 points) Write a Prolog program that defines a predicate `selectdiscard(L1, L2, R, S)` that given 2 lists L1 and L2 compares their elements pairwise and returns the list R of the greater elements, along with the sum S of all the elements that have not been included in R. If one of the two lists has more elements than the other, all the remaining elements will not be included in R, (but will be included in the sum!) If two elements have the same value, such a value is included in R, but is not considered in S.

   Examples:

```
?- selectdiscard([1, 4, 5], [2, 5, 3], R, S). outputs R=[2, 5, 5], S=8.
?- selectdiscard([5, 4, 5], [2, 5, 3], R, S). outputs R=[5, 5, 5], S=9.
?- selectdiscard([5, 5, 5], [2, 5, 3], R, S). outputs R=[5, 5, 5], S=5.
?- selectdiscard([1, 4, 5], [2, 5], R, S). outputs R=[2, 5], S=10.
*/
```

   Solution (recursive):

```
selectdiscard([], [], [], 0):- !.
selectdiscard([], [E|L], R, S):- selectdiscard([], L, R, S2), S is S2+E, !.
selectdiscard([E|L], [], R, S):- selectdiscard(L, [], R, S2), S is S2+E, !.
selectdiscard([E1|L1], [E2|L2], [GE|R], S):- E1>E2, selectdiscard(L1, L2, R, S2),
        GE is E1, S is S2+E2, !.
selectdiscard([E1|L1], [E2|L2], [GE|R], S):- E1==E2, selectdiscard(L1, L2, R, S),
        GE is E1, !.
selectdiscard([E1|L1], [E2|L2], [GE|R], S):- E1<E2, selectdiscard(L1, L2, R, S2),
        GE is E2, S is S2+E1.
```

8. (3 points) Consider the Prolog program P consisting of the clauses below:

```
p(X,Y):-q(X,Y), q(Y,X).
q(f(V),g(V)):-r(V).
q(f(V),f(V)):-s(V).
r(a).
s(b).
```

   Show the derivation tree for the evaluation of the goal `p(W,Z)` in P and provide the compute answer substitution for such an evaluation. (As usual `K, X, Y, V, W,Z` are variables, `a, b` constants and `f, g` are function symbols). The derivatrion tree is

```
 p(W,Z)
  |
  |
q(X,Y), q(Y,X)
  |                \
  |                   \
  |                      \
r(V), q(g(V),f(V))   s(V), q(f(V),f(V))
  |                              |
  |                              |
 q(g(a),f(a))                    q(f(b),f(b))
  |                              |
  |                              |
fail                              s(b)
                                 |
                                 success
```

The computer answer substitution is {W/f(b), Z/f(b)}.