Course: CSC 520, Introduction to Artificial Intelligence
Homework 4
Student: Xusheng Xiao
Unity ID: xxiao2
Email: xxiao2@ncsu.edu

---

1. (20 points) Two grad students in the Knowledge Discovery Lab (KDL) prefer two different coffees (Brazilian and Indian). However, there is only one coffee machine in the lab! There are three operators: load coffee into the machine, brew the coffee, and clean the machine. Suppose the machine contains old Brazilian coffee, and a student wishes to brew a fresh pot of Indian coffee. Help the grad students with a plan!

   The PDDL description of the actions are as follows (X is a variable; nil and waste are constants):

   Action: load(X)
   PRECOND: coffee(X), loaded(nil)
   EFFECT: loaded(X), ¬loaded(nil)
   Action: brew(X)
   PRECOND: loaded(X), ¬loaded(nil), ¬loaded(waste)
   EFFECT: ¬loaded(X), loaded(waste), pot(X)
   Action: clean(X)
   PRECOND: loaded(X), ¬loaded(nil)
   EFFECT: ¬loaded(X), loaded(nil)

   (a) (5 pts.) Specify the initial state and the goal conditions, and draw the "relaxed planning graph" for this problem. Use b and i as constants to represent the coffee. (A relaxed planning graph ignores negative interactions–ie, no mutexes.)

   Init(¬loaded(b) ∧ ¬loaded(nil) ∧ loaded(waste) ∧ coffee(b) ∧ coffee(i))

   goal(pot(i))

   Figure 1 shows the relaxed planning graph.

   (b) (10 pts.) Now make the graph into a standard planning graph by including the mutexes. Indicate the appropriate mutexes at each level. Indicate which type of mutex each is.

   Since it is hard to see all the types of conflicts in the graph, I list all the conflicts in Figure 1 as below:

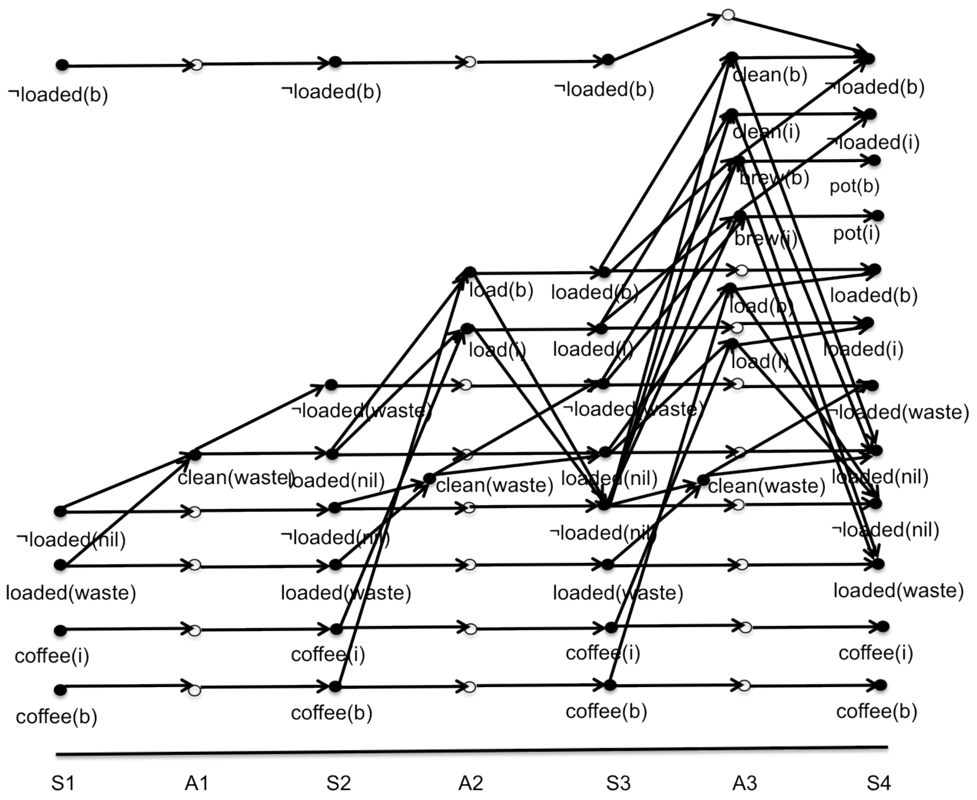

Figure 1: Relaxed Planning Graph

- level S2:
  - **Negated Literals:** [¬loaded(waste), loaded(waste)], [¬loaded(nil), loaded(nil)]
  - **Inconsistent Support:** [loaded(waste), loaded(nil)], [¬loaded(waste), ¬loaded(nil)]
- level A2:
  - **Inconsistent Effect:** [clean(waste), load(b)], [clean(waste), load(i)],
  - **Interference:** [load(b), load(i)]
  - **Competing Needs:**
- level S3:
  - **Negated Literals:** [¬loaded(waste), loaded(waste)], [¬loaded(nil), loaded(nil)]
  - **Inconsistent Support:** [loaded(waste), loaded(nil)], [loaded(waste), loaded(b)], [loaded(waste), loaded(i)], [loaded(b), loaded(i)],

- level A3:
  - **Inconsistent Effect:** [clean(waste), load(b)], [clean(waste), load(i)], [clean(b), load(b)],[clean(i), load(i)],[brew(b),load(b)], [brew(i),load(i)],[brew(b),clean(waste)],[brew(i),clean(waste)]
  - **Interference:** [load(b), load(i)], [brew(b),brew(i)],[brew(b),clean(b)],[brew(i),clean(i)]
  - **Competing Needs:** [brew(b), brew(i)], [clean(b),load(b)], [clean(b),load(i)], [clean(i),load(b)], [clean(i),load(i)],[brew(b),clean(i)],[brew(i),clean(b)],
- level S4:
  - **Negated Literals:** [¬loaded(waste), loaded(waste)], [¬loaded(nil), loaded(nil)], [¬loaded(b), loaded(b)],[¬loaded(i), loaded(i)]
  - **Inconsistent Support:** [loaded(waste), loaded(nil)], [loaded(waste), loaded(b)], [loaded(waste), loaded(i)], [loaded(b), loaded(i)], [pot(b), pot(i)], [pot(b), loaded(b)],[pot(i), loaded(i)],[pot(b),¬loaded(waste)], [pot(i),¬loaded(waste)],[pot(b),loaded(nil)],[pot(i),loaded(nil)],[pot(b),loaded(i)],[pot(i),loaded(b)]

(c) (5 pts.) Avoiding the mutexes, extract a plan from the planning graph in (b).

The plan is [clean(waste), load(i), brew(i)]

2. (30 pts.) Consider the following description: There are four animals in the world, by the names of Mufasa, Velocitus Incalculii, Willie, and Woody. Mufasa is a lion, Velocitus Incalculii is a roadrunner, Willie is a whale, and Woody is a woodpecker. Roadrunners and woodpeckers are kinds of birds, while lions and whales are kinds of mammals. All mammals are covered with hair while birds are covered with feathers. As a general rule, mammals can run and birds can fly, but there are exceptions. Whales can swim and not run, and roadrunners can run and not fly, and therefore both are exceptions to the general rules. Lions and woodpeckers are not exceptions.

Answer the following questions:

(a) (5 pts.) Represent the above world description using a semantic network.

Figure 2 shows the semantic network.

(b) (5 pts.) Implement the semantic net in Prolog. Use a technique patterned on examples in class and webnotes.

Rules:
value(Node, Slot, Value):- edge(Node, Slot, Value).

(c) (5 pts.) Mufasa is a lion, which is a kind of mammal. Therefore, mufasa is a mammal. Add a rule to your prolog database that generalizes this to all animals. Run a sample query to demonstrate this

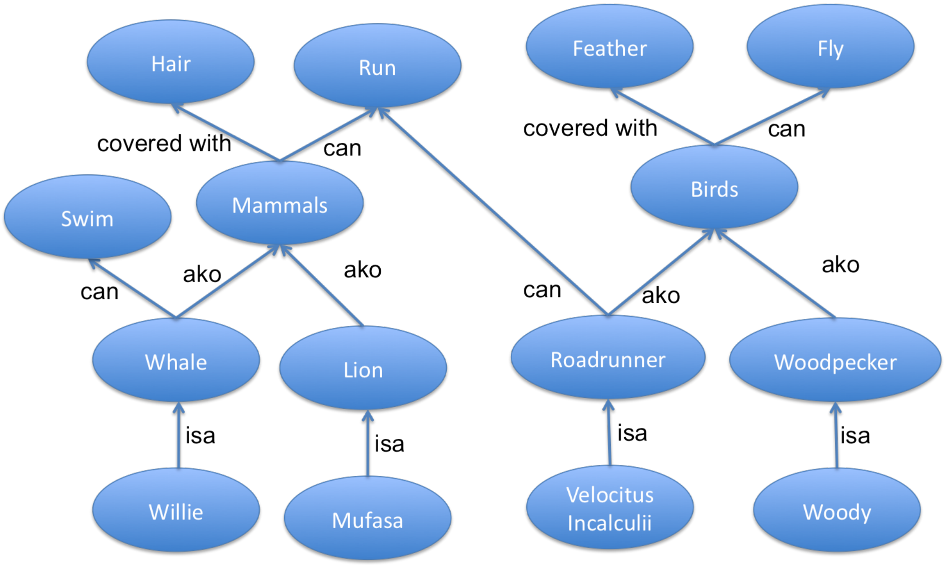

Figure 2: Semantic Network

rule and show the output.

Rules:
value(Node, isa, Value):-
        edge(Node, isa, Node1),
        edge(Node1, ako, Value).

Query: value(mufasa,isa,X).
Result: X = lion; X = mammals ;

Query: value(velocitusIncalculii, isa, X).
Result: X = roadrunner; X = birds ;

(d) (10 pts.) Now add rules to the prolog database so that all animal instances inherit the properties of their kind, namely being covered with hair or feathers, running, swimming, or flying. Run a sample query to demonstrate the rules and show the output.

Rules:
value(Node, Slot, Value):-
        edge(Node, isa, Node1),
        value(Node1, Slot, Value).

```
value(Node, Slot, Value):-
        edge(Node, ako, Node1),
        value(Node1, Slot, Value).
```

Query: value(willie, coveredWith, X).
Result: X = hair ;

Query: value(velocitusIncalculii, can, X).
Result: X = run ; X = fly.

Query: value(mufasa, can, X).
Result: X = run ;

Query: value(woody, coveredWith, X).
Result: X = feather ;

(e) (5 pts.) Because your implementation did not account for exceptions, it should conclude that Willie can run and Velocitus Incalculii can fly!! Fix this problem. (Note: While it may be true that road runners can indeed fly, they cannot be airborne for more than a few seconds, and hence for this question consider them non-flying birds.) Show the output to show that you have fixed this problem.

Modified the original rules to include exceptions of whale and roadrunner (Please see submitted prolog files).

Query: value(willie, can, X).
Result: X = swim ; false.

Query: value(willie, can, run).
Result: false.

Query: value(whale, can, run).
Result: false.

Query: value(velocitusIncalculii, can, X).
Result: X = run ; false.

Query: value(velocitusIncalculii, can, fly).
Result: false.

Query: value(roadrunner, can, fly).
Result: false.

To earn full credit, submit the prolog file, and sample queries and outputs for (c), (d) and (e).

3. (15 pts.) In a document classification problem, we wish to classify each document in the collection into a predefined set of categories based on the text (or, the words) contained in them. A fairly straightforward way is to use a Naive Bayes classifier where the document is the hypothesis variable and the evidence is the presence (or absence) of the words. We assume that the words occur independently.

   (a) (5 pts.) If you were asked to build such a Naive Bayes model, how would you do it, given a training set? (You don't have to actually build anything.)

   Suppose the words are $w_1, w_2, \ldots, w_n$ and the categories of documents are $C_1, C_2, \ldots, C_n$.
   Using a naive bayes classifier to classify documents, we can compute the probability for the categories using the following equations:

   $$P(w_1, w_2, \ldots, w_n, C_i) \tag{1}$$

   Since the words occur independently, the above equation is equivalent to:

   $$P(C_i)(\prod_{k=1}^{n} P(w_k|C_i)) \tag{2}$$

   (b) (5 pts.) If a new document is presented, explain how it would be classified. If a new document is presented, we first compute the occurences of words in the new document. Assume that the values are $w_1, w_2, \ldots, w_n$. Then, we can compute the probabilities for all the categories using the equation (2) and choose the categories that has the highest probability.

   (c) (5 pts.) Is the conditional independence assumption valid for this problem? Explain.

   The condition indenpence assumption may not be valid for this problem. Since for the documents that describes various topics, some words coexists for each topic. For example, for a document describing cardiac diseases, the word "heart" and "cardiac" may apprear together many times.

4. (10 pts.) As a part of a regular health checkup mandated by your insurance company, you undergo a medical check for presence of a particular new virus in your blood. The test is 98% accurate. Recent statistics show that only a small proportion of the population has the virus (about 1 in 10000). Should you be happy that this is a rare virus? Given a positive test result, what are the chances that you actually have the virus?

Since the proportion of the population that has the virus is $\frac{1}{10000}$, it is a rare virus. Given the test is 98% accurate, once the test tells you that you got the virus, the probability that you actually got the virus is 98%. Given a positive test result, the chances that you can have the virus can be computed using the following equation if we assume the test result is independent with the probability that people has the virus:

$$P(gotVirus|test) = P(gotVirus, test)/P(test) = P(gotVirus) = \frac{1}{10000}$$

If the assumption is not supported, then we need to compute the probability of $P(gotVirus, test)$ in order to compute $P(gotVirus|test)$.

5. (25 pts) Babies have few and simple needs, but they have even fewer ways to communicate those needs to their parents. Parents need some way to figure out what is the matter with their child. Assume we know the following:

   P(sick) = .3
   P(ache | sick) = .75
   P(ache | ¬sick) = .1
   P(tired) = .6
   P(cry | ache, tired) = .9
   P(cry | ache, ¬tired) = .65
   P(cry | ¬ache, tired) = .55
   P(cry | ¬ache, ¬tired) = .2

   (a) (5 pts) Assuming that the above conditional probabilities are sufficient to specify the values for a Bayes net for this problem, draw a picture of such a Bayesian network.

   Figure 3 shows the bayes network for the babies problem.

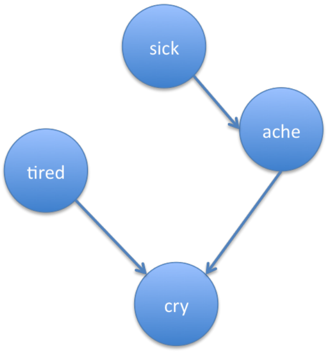

Figure 3: Bayes Network

(b) (20 pts) Compute the full joint probability table for this Bayesian network. Please model your table on the following template:

		sick		¬sick	
		tired	¬tired	tired	¬tired
ache	cry	0.1215	0.0585	0.0378	0.0182
	¬cry	0.0135	0.0315	0.0042	0.0098
¬ache	cry	0.02475	0.006	0.2079	0.0504
	¬cry	0.02025	0.024	0.1701	0.2016

Below are the euqations for the four cells at the first row.

$$\begin{aligned} P(sick, tired, ache, cry) &= P(sick) * P(tired|sick) * P(ache|sick, tired) * P(cry|sick, tired, ache) \\ &= P(sick) * P(tired) * P(ache|sick) * P(cry|ache, tired) \\ &= 0.3 * 0.6 * 0.75 * 0.90 \\ &= 0.1215 \end{aligned}$$

$$\begin{aligned} P(sick, \neg tired, ache, cry) &= P(sick) * P(\neg tired|sick) * P(ache|sick, \neg tired) * P(cry|sick, \neg tired, ach \\ &= P(sick) * P(\neg tired) * P(ache|sick) * P(cry|ache, \neg tired) \\ &= 0.3 * 0.4 * 0.75 * 0.65 \\ &= 0.0585 \end{aligned}$$

$$\begin{aligned} P(\neg sick, tired, ache, cry) &= P(\neg sick) * P(tired|\neg sick) * P(ache|\neg sick, tired) * P(cry|\neg sick, tired, a \\ &= P(\neg sick) * P(tired) * P(ache|\neg sick) * P(cry|ache, tired) \\ &= 0.7 * 0.6 * 0.1 * 0.9 \\ &= 0.0378 \end{aligned}$$

$$\begin{aligned} P(\neg sick, \neg tired, ache, cry) &= P(\neg sick) * P(\neg tired|\neg sick) * P(ache|\neg sick, \neg tired) * P(cry|\neg sick, \neg t \\ &= P(\neg sick) * P(\neg tired) * P(ache|\neg sick) * P(cry|ache, \neg tired) \\ &= 0.7 * 0.4 * 0.1 * 0.65 \\ &= 0.0182 \end{aligned}$$