CS 440: Introduction to AI

Homework 1 Solution

Due: Thursday, September 9th

Your answers must be concise and clear. Explain sufficiently that we can easily determine what you understand. We will give more points for a brief interesting discussion with no answer than for a bluffing answer.

AI Models

1. (16 points) Suppose that we expect to filter spam messages from a mailbox. We want to build a model that can classify a message into a spam or a ham (non-spam). Consider using the features below:

   (1) The number of words in the message.

   (2) If the sender has sent a spam message before, the message is likely to be a spam by 90% accuracy.

   (3) If a text contains a frequently used spam phrase, the message is a spam by 70% accuracy.

   (4) Whether the message contains a picture or not.

   (5) Whether the sender is in the receiver's contacts.

   (6) In the previous spams, the number of sexual words is 5.7 on average.

   Recall that a model is a stand-in, or an approximate, mathematically precise representation, for the real thing.

   (a) Consider models that include each feature, and say whether the resulting model is analytic or empirical and briefly explain why (for some, both answers might be acceptable if it is properly justified).

i. *(Example)* Feature (1)
Analytic. The number of words can be counted and given to the model without experiments.

ii. (2 points) Feature (2)
Empirical. Observations are needed to obtain the probability of a spam message.

iii. (2 points) Feature (3)
Empirical. Frequently used spam phrases should be obtained through observations. Also, the probability of a spam message needs to be observed.

iv. (2 points) Feature (4)
Analytic. Wheter or not the message contains a picture is a predetermined fact that do not need observations to figure out.

v. (2 points) Feature (5)
Analytic. Whether the sender is in the contacts is a predetermined fact.

vi. (2 points) Feature (6)
Empirical. Observation is needed to compute the average number of sexual words in the previous spams.

(b) For the spam classification task above, give another feature that is:

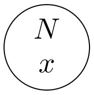
i. (2 points) Purely analytic. (i.e., requires no experiments/observation)
The length of the message.

ii. (2 points) Purely empirical. Say what must be observed.
It is not possible to have a purely empirical element.

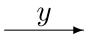
iii. (2 points) Combination of analytic and empirical. Say what must be observed.
Whether the number of hyperlinks in this message is closer to the average number of hyperlinks in spam messages or closer to the average in non-spam messages.
The empirical information to be observed from previous spam and non-spam samples are the two averages. The information to be observed from this message is its number of hyperlinks.

Search

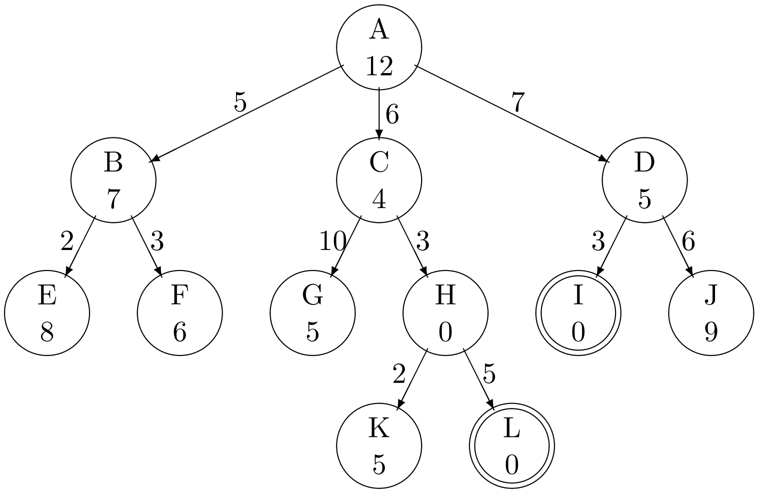
2. (28 points) Consider the search tree below. The initial state is at the top, and the goal states are represented by the double circles. Note that the edges are directed.



N is the name of the node.
x is the heuristic function's estimate of the cost to the nearst goal, $h(N)$.



y is the actual cost of traversing an edge.



(a) For each of the search strategies listed below, (1) *form an ordered list of the states explored* and (2) *indicate which goal state is reached (if any).* Also (3) *specify the final contents of the queue at the time the search terminates.* When the states are expanded, new nodes are generated in ascending alphabetical order. Assume the sort is stable (i.e., ties are left as initially ordered).

i. (4 points) Depth First Search
States explored: A, B, E, F, C, G, H, K, L
Goal state reached: L
Final queue contents: D

ii. (4 points) Breadth First Search

States explored: A, B, C, D, E, F, G, H, I
Goal state reached: I
Final queue contents: J, K, L

iii. (4 points) Uniform Cost
States explored: A, B, C, D, E, F, H, I
Goal state reached: I
Final queue contents: K, J, L, G

iv. (4 points) Greedy / Best First
States explored: A, C, H, L
Goal State reached: L
Final queue contents: D, G, K, B

v. (4 points) A^*
States explored: A, C, H, B, D, I
Goal State reached: I
Final queue contents: L, F, E, K, G, J

(b) (8 points) In this example, is $h(N)$ admissible? Is it consistent? Briefly explain your answer.

$h(N)$ is admissible if it never overestimates the cost to reach the goal, i.e., $h(N) \leq h^*(N)$. In this example, however, $h(A) = 12 \nleq 10 = h^*(A)$, which does not satisfy the condition. (There is another case in this example, which is $h(D) = 5 \nleq 3 = h^*(D)$.)

Consistency is a slightly stronger condition. $h(N)$ is consistent if, for every node n and every successor N' of N generated by any action a, the estimated cost of reaching the goal from N is no greater than the step cost of getting to N' plus the estimated cost of reaching the goal from N':

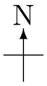
$$h(N) \leq c(N, a, N') + h(N').$$

For instance, $c(N, a, N')$ is a weight of the edge between N and N'. In this example, however, there are several places the condition is not satisfied. For example, $h(C) = 4, c(C, a_{right}, H) = 3$, and $h(H) = 0$, and thus, $4 \nleq 3 + 0$.

You will get 4 points if only one of your answers is correct.

3. (32 points) Consider the following “grid world.” Imagine a robot located at the start state trying to devise a route to reach the goal using a search algorithm. In each state, 4 operations are allowed: moving north, east, south, or west. Moving into a “Swamp” state costs 4 units, while moving into any other states costs 1 unit. Moving into the surrounding walls (e.g., moving west in the start state) is allowed and will result in the same state before the move, and the cost for the move depends on the state in which the move is attempted (4 for swamp, 1 otherwise).

Start	Swamp	Swamp
	Swamp	Goal



(a) i. (4 points) Define a data structure that is adequate to model the state representation at the nodes of this problem.
The state is represented as a pair of numbers (x,y) denoting the coordinates of the current location in the grid, where (0,0) is the lower left corner.

ii. (4 points) Define the operator “Move South” by specifying its preconditions and effects.

```
Move_South((x,y)):
PC: y does not equal to 0
Effects:  Collect((x, y - 1))
```

If we are in the southmost cells, we need a different operator.

```
Move_South((x,y)):
PC: y equals to 0
Effects:  Collect((x, y))
```

(b) Suppose that the robot is allowed to choose a random expansion order and a tie-breaker before starting exploration. Suppose also that repeating state (e.g., moving into the wall continuously) cannot be detected. For each of the search algorithms below, (1) What is the fewest possible number of nodes that must be visited? (2) Is the algorithm guaranteed to reach the goal state? (3) Would an optimal solution be found in this case?

i. (6 points) Depth First Search

(1) It will visit infinite nodes since the robot will not change its direction.

(2) No. It will finally stay at (0,2), (2,2), or (0,0).

(3) No. No solution can be found.

ii. (6 points) Breadth First Search

(1) With the expansion order (East-South-North-West), it will visit 23 nodes, which are (0,2), (1,2), (0,1), (0,2), (0,2), (2,2), (1,1), (1,2), (0,2), (1,1), (0,0), (0,2), (1,2), (1,2), (0,1), (0,2), (0,2), (1,2), (0,1), (0,2), (0,2), (2,2), (2,1). We cannot detect repeated states, so we must include the repeated states in the queue when we bump into the wall.

(2) BFS is guaranteed to reach the goal state if there is one.

(3) BFS is not guaranteed to reach the optimal solution if the path costs are not the same. In this example, the solution found ((0,2)-(1,2)-(2,2)-(2,1)) is not optimal since it costs 9 units. The optimal solution is ((0,2)-(0,1)-(0,0)-(1,0)-(2,0)-(2,1)), which costs 5 units.

iii. (6 points) Uniform Cost

(1) With an expansion order (South-East-North-West), we visit nodes as follows:

Node	Queue
-	$(0,2)_0$
$(0,2)_0$	$(0,1)_1, (0,2)_1, (0,2)_1, (1,2)_4$
$(0,1)_1$	$(0,2)_1, (0,2)_1, (0,0)_2, (0,0)_2, (0,1)_2, (1,2)_4, (1,0)_5$
$(0,2)_1$	$(0,2)_1, (0,0)_2, (0,0)_2, (0,1)_2, (0,1)_2, (0,2)_2, (0,2)_2, (1,2)_4, (1,0)_5, (1,2)_5$
$(0,2)_1$	$(0,0)_2, (0,0)_2, (0,1)_2, (0,1)_2, (0,2)_2, (0,2)_2, (0,1)_2, (0,2)_2, (0,2)_2, (1,2)_4, (1,0)_5, (1,2)_5, (1,2)_5$
...	...

This problem was not designed well. If you showed first five steps, you will get full credit.

(2) Uniform Cost is guaranted to reach the goal state if there is one.

(3) We will always find the optimal solution using uniform cost search.

iv. (6 points) A^* ($h(N)$ is Manhattan distance to Goal, e.g., $h(N)$ from Start is 3.)

(1)With an expansion order (South-East-North-West), we visit nodes as follows:

Node	Queue
-	$(0,2)_3$
$(0,2)_3$	$(0,1)_3, (0,2)_4, (0,2)_4, (1,2)_6$
$(0,1)_3$	$(0,2)_4, (0,2)_4, (0,1)_4, (0,0)_5, (0,2)_5, (1,2)_6, (1,2)_6$
$(0,2)_4$	$(0,2)_4, (0,1)_4, (0,1)_4, (0,0)_5, (0,2)_5, (0,2)_5, (0,2)_5, (1,2)_6, (1,2)_6, (1,2)_7$
$(0,2)_4$	$(0,1)_4, (0,1)_4, (0,1)_4, (0,0)_5, (0,2)_5, (0,2)_5, (0,2)_5, (0,2)_5, (0,2)_5, (1,2)_6, (1,2)_6, (1,2)_7, (1,2)_7$
...	...

This problem was not designed well. If you showed first five steps, you will get full credit.

(2) A^* is guaranted to reach the goal state if $h(N)$ is admissible. In this problem, $h(N)$ never overestimates the cost to reach the goal, so it's guaranteed to reach the goal state.

(3) A^* is guaranteed to find the optimal solution if $h(N)$ is admissible. Since $h(N)$ is admissible, it's guaranteed to find the optimal solution.