ICT-K61: Artificial Intelligence

April 2019

Assignment #2
Submission: May 1st, 2019 - Before midnight

Exercise 7. Constraint Satisfaction. Sudoku

Consider Sudoku, played on a 9×9 grid probably containing some digits in some grid squares in its initial state. The rules of the game are: *Fill in the grid so that every row, every column, and every* 3×3 *box contains the digits 1 through 9.*

(i) Formalize Sudoku as a Constraint Satisfaction Problem.

(ii) I dont think anyone knows how many ways there are to fill the grid if it is initially totally empty. But what can you say about an upper bound on the number of such solutions (i.e. what is your best answer for X in the statement "there are no more than X solutions.", and why?).

(iii) What would be your favored method of solving the Sudoku CSP, and why? (Hint: some ideas are clearly worse than others but for me, here, there's no single *right* answer, so don't be afraid to pick a likely method and convince me it's good.)

Exercise 8. Constraint Satisfaction. Tiling a surface

Consider the problem of tiling a surface (completely and exactly covering it) with n dominos (2×1 rectangles). The surface is an arbitrary edge-connected (i.e., adjacent along an edge, not just a corner) collection of $2n$ 1×1 squares (e.g., a checkerboard with some squares missing, a 10×1 row of squares, etc.).

(i) Formulate this problem precisely as a CSP where the dominos are the variables (i.e., define the variable domain and the constraints)

(ii) Formulate this problem as a CSP where the squares are the variables, keeping the state space as small as possible [*Hint.does it matter which particular domino goes on a given pair of squares?*]

(iii) Construct a surface consisting of 6 squares such that your CSP formulation from part *(ii)* has a tree-structured constraint graph.

(iv) Describe exactly the set of solvable instances that have a tee-structured constraint graph.

Exercise 9. Constraint Satisfaction. Crossword puzzle grid.

CryptoBoys got a crossword puzzle grid (that is, an array of blank squares and blacked out squares) and a fairly small vocabulary of specialized words; he would like to maximize the number of words from the special vocabulary that are placed in the grid. A complete solution would be good but very unlikely: he plans to try to fill remaining blanks himself. He decides to use backtracking (depth-first tree search) to place words.

(i) Describe the situation formally as a Constraint Satisfaction Problem (variables, values, constraints, success criteria).

(ii) How can he implement the fact that the best solution could have spaces with no words from the vocabulary assigned? What are the consequences for his search? Can he speed things up with constraint propagation or forward checking?

(iii) Actually, CryptoBoy always forces certain spaces to contain a particular word (or small subset of vocabulary words). Does this allow constraint propagation of any sort and if so exactly what?

Exercise 10. Propositional Logic

Suppose an agent inhabits a world with two states, S and $\neg S$, can do exactly one of two actions, a and b. Action a does nothing and action b flips from one state to the other. Let S^t be the proposition that the agent is in state S at time t, and let a^t be the proposition that the agent does action a at time t (similarly for b^t).

(i) Write a successor-state axiom for S t+1

(ii) Convert the sentence in (a) into CNF.

(iii) Show a resolution refutation proof that if the agent is in $\neg S$ at time t and does a it will still be in $\neg S$ at time $t+1$.

Exercise 11 - Propositional Logic

A propositional 2-CNF expression is a conjunction of clauses, each containing exactly 2 literals, e.g.,

$$(A \vee B) \wedge (\neg A \vee C) \wedge (\neg B \vee D) \wedge (\neg C \vee G) \wedge (\neg D \vee G)$$

(i) Prove using resolution that the above sentence entails G.

(ii) Two clauses are semantically distinct if they are not logically equivalent. How many semantically distinct 2-CNF clauses can be constructed from n proposition symbols?

(iii) Using your answer to *(ii)*, prove that propositional resolution always terminates in time polynomial in n given a 2-CNF sentence as input containing no more than n distinct symbols.

(iv) Explain why your argument in *(iii)* does not apply to 3-CNF.

Exercise 12. Proof

(i) Use a truth table to show the following sentence in propositional logic is valid.

$$[(P \Rightarrow Q) \wedge (P \Rightarrow R)] \iff [P \Rightarrow (Q \wedge R)]$$

(ii) Let the following propositional symbols have the following meaning:

A John was in a car accident.
S John is sick.
I John is injured.
D John needs to see a doctor.

Express each of the following English sentences in propositional logic.

(ii-1) John was in a car accident, but he isn't injured
(ii-2) John needs to see a doctor if he is sick or injured
(ii-3) If John wasnt in an accident and isnt sick, then he doesnt need a doctor

Exercise 13. Search Algorithms

Apply four different tree-searching algorithms to the "tree" below. It is not a tree but we are pretending we do not know that: you have an OPEN list but no CLOSED list. The figure shows the names of the nodes: G1, G2, and G3 are search-ending goal states, and S is the start state. The arcs are labeled with the cost of following them, and the heuristic h value (estimated cost to goal) is given for the relevant interior nodes. For each type of search, please list in order the nodes that are explored (taken off the OPEN list, checked for goal, successors put on OPEN list).
Hints: Each search starts by putting S on the OPEN list so each answer (A-D) starts with S. Also, some searches may explore a state more than once.

(i) Greedy (best first) Search.

(ii) Iterated Depth First Search.

(iii) Uniform Cost Search.

(iv) A* Search.

(v) Just checking. We pay A* search to find the optimal (cheapest) path to the goal. Did it work? If so, how do you know and if not why not?

Exercise 14. FOPC

Given these premises:

1. Anyone who prepares is smart

2. Everybody has a friend who prepares

Express them in FOPC, put them into clausal form and use resolution to answer the question: "Who is smart?" (That is, is anyone smart and if so, who?)

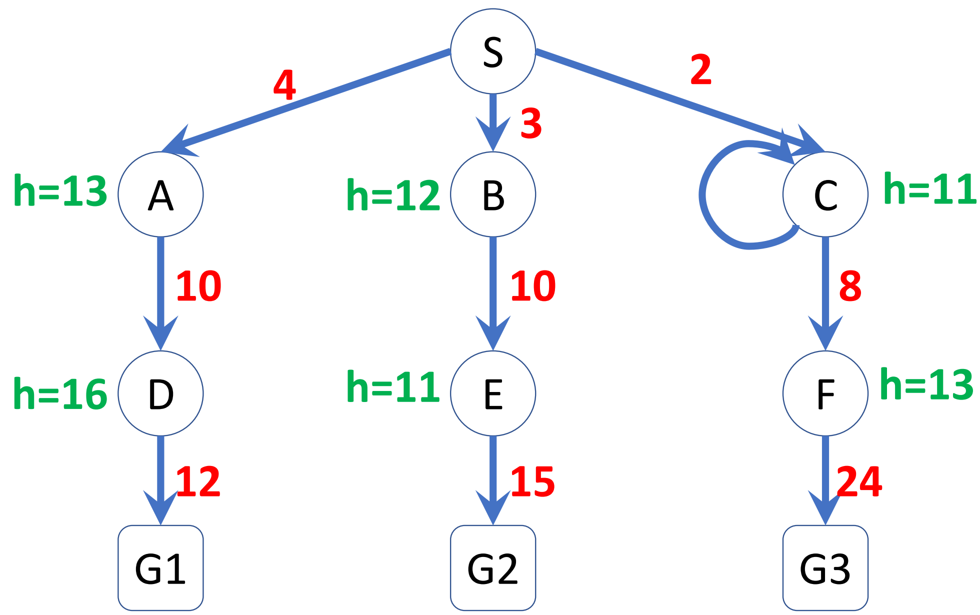

Figure 1: Search "tree".

Exercise 15. Mere Formality

We have a domain of geographical regions and map colors, a function $\textsc{MapColor}(x)$ and predicates $x = y, \textsc{In}(x, y), \textsc{Borders}(x, y), and\textsc{Country}(x)$, all with obvious meanings. We have constants for some regions. Given the English sentences below, say whether each associated logical expression is

1 $\rightsquigarrow$ correctly expresses the English sentence,

2 $\rightsquigarrow$ is syntactically invalid and thus meaningless,

3 $\rightsquigarrow$ is syntactically valid but does not express the meaning correctly.

A. No region in South America borders any region in Europe.

- **(i)** $\neg[\exists c, d \quad \textsc{In}(c, \text{SouthAm}) \wedge \textsc{In}(d, \text{Europe}) \wedge \textsc{Borders}(c, d)]$
- **(ii)** $\forall c, d \quad [\textsc{In}(c, \text{SouthAm}) \wedge \text{In}(d, \text{Europe}) \Rightarrow \neg\textsc{Borders}(c, d)]$
- **(iii)** $\neg\forall c \quad \textsc{In}(c, \text{SouthAm}) \Rightarrow \exists d \textsc{In}(d, \text{Europe}) \wedge \neg\textsc{Borders}(c, d)$
- **(iv)** $\forall c \quad \textsc{In}(c, \text{SouthAm}) \Rightarrow \forall d \quad \textsc{In}(d, \text{Europe}) \Rightarrow \neg\text{Borders}(c, d)$

B. No two adjacent countries have the same map color.

- **(i)** $\forall x, y \quad \neg\textsc{Country}(x) \vee \neg\textsc{Country}(y) \vee \neg\textsc{Borders}(x, y) \vee \neg(\textsc{MapColor}(x) = \textsc{MapColor}(y))$
- **(ii)** $\forall x, y \quad \textsc{Country}(x) \wedge \textsc{Country}(y) \wedge \textsc{Borders}(x, y) \wedge \neg(x = y) \Rightarrow \neg(\textsc{MapColor}(x) = \textsc{MapColor}(y))$
- **(iii)** $\forall x, y \quad \textsc{Country}(x) \wedge \textsc{Country}(y) \wedge \textsc{Borders}(x, y) \wedge \neg(M\textsc{apColor}(x) = \textsc{MapColor}(y))$
- **(iv)** $\forall x, y \quad (\textsc{Country}(x) \wedge \textsc{Country}(y) \wedge \textsc{Borders}(x, y)) \Rightarrow \textsc{MapColor}(x \neq y)$

Exercise 16. FOL

Put assertions A and B into FOPC formulae, convert to clause form, and use resolution to answer the question "What courses would Bob like?".

Make sure you show what converts to what, what resolves to what, etc. so the process is clear, not just the answer.

Keep this simple by assuming "typed" variables: in particular assume that x varies over the domain of courses and y varies over the domain of departments. Thus you do not have to assert and carry around facts like $\textsc{Course}(x)$ or $\textsc{Department}(y)$. Use the predicate notation $L(x), E(x), F(x)$ for "Bob likes x, x is Easy, and x is Fun".

A. Bob likes any course that is easy or fun.

B. Every department has at least one easy course.

C. No courses are fun.

Exercise 17. Clausal Form

Put the following expression into clause form.

$$(\forall x \quad [(\exists y \quad [P(x,y) \wedge Q(x,y)]) \quad \Rightarrow \quad (\exists y \quad R(x,y))]).$$

Exercise 18. Reinforcement learning for chess

Imagine you were to design a reinforcement learning agent for playing chess. The state that the agent sees on its turn is the layout of the chess board. We can set the reward structure for the agent to be +1 for winning, -1 for losing, 0 for drawing, and 0 again for every move that does not lead to a win or loss. Such an agent will essentially learn to win. It will do so eventually after much exploration and a number of episodes, since it is always trying to maximize its expected return (cumulative rewards in the long run). What might happen if, in addition, we give a positive reward to the agent for taking its opponent's pieces as well?

Exercise 19.

Consider the following update rule:

$$Q_{n+1} \leftarrow Q_n + \alpha(R_n Q_n).$$

(i) Show that

$$Q_{n+1} = (1-\alpha)^n Q_1 + \sum_{n'=1}^{b} \alpha(1-\alpha)^{n'-n} R_{n'}. \tag{1}$$

(ii) Derive a similar equation as (1) for an arbitrary sequence $\{\alpha_n\}$ of step sizes.

Exercise 20.

With depletion of the ozone layer, over-exposure to UV radition has become a serious concern. Induce a compact decision tree for the following data that indicates, based on relevant attributes, the risk of an individual to become sunburned after basking in Montreal sunshine.

Name	Hair	Height	Weight	Lotion	Sunburned?
Sarah	blonde	average	light	no	yes
Dana	blonde	tall	average	yes	no
Alex	brown	short	average	yes	no
Annie	blonde	short	average	no	yes
Emily	red	average	heavy	no	yes
Pete	brown	tall	heavy	no	no
John	brown	average	heavy	no	no
Katie	blonde	short	light	yes	no

Exercise 21.

A certain agent selects actions from (North, South, East, West) with equal probability, without regard to its percepts. If the selected move leads to an unreachable state, the agent does not move. Figure below illustrates the agents environment, the two squares marked by +1 and -1 are terminal states. Assume that the utility of a sequence of actions is equal to the reward obtained in the terminal state minus 1/20 the sequence length (i.e., the number of times the agent actually moved). Given the following environment, use value iteration to calculate utility estimates for the different states. Show the results for two iterations over the entire state space.

3				+1
2		OBSTACLE		-1
1	START			
	1	2	3	4

Exercise 22.

Suppose you are working on a spam detection system. You formulated the problem as a classification task where Spam is the positive class and not Spam is the negative class. Your training set contains m = 1000 emails, 99% of these are non-Spam and 1% are spam.

(i) What accuracy has a classifier that always predicts not Spam?

(ii) The fraction of spam emails, that are correctly classified is measured by the recall value. What is the recall of the always non-Spam classifier?

$$recall = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

(iii) Suppose you trained a classifier using a MLP with one output neuron. If the activation of the output neuron is larger than a threshold θ, the instance will be classified as positive.

The result on the training set using $\theta = 0.5$ is summarized in Table 1. In total, there are 24 instances classified as Spam. The percentage of those that really are spam is called precision:

$$precision = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

Calculate precision and recall for the MLP classifier. What happens as you vary θ?

		Predicted class	
		Spam	not Spam
Actual class	Spam	8	2
	not Spam	16	974

Table 1: Performance of the MLP spam detection system on the training set

(iv) Finally, you evaluate the performance of your classifier on independent test data and observe that the results are substantially worse than on the training set. Which of the following measures are likely to improve the performance of your spam detection system?

☐ using additional features

☐ increasing the weight decay parameter

☐ increasing the number of gradient descent iterations

☐ reducing the number of gradient descent iterations

☐ reducing the number of hidden neurons in the MLP

☐ obtaining more training data

Exercise 23.

Below is a dataset that determines the survival rate of passengers on the Titanic based on three attributes Class (C), Gender (G) and Age (A). Train a depth 1 decision tree (i.e., a decision tree with only 1 decision node) using information gain as the criterion for splitting, using $IG(S, x) = H(S) - H(S|x)$, where $x \in \{C, G, A\}$. Which node is at the root of the tree? Show all your work (i.e., include all intermediate steps) in calculating information gain for each attribute. Round to 4 decimal points for all calculations, and use log base 2.

Class	Gender	Age	No	Yes	Total
1st	Male	Child	0	5	5
1st	Male	Adult	118	57	175
1st	Female	Child	0	1	1
1st	Female	Adult	4	140	144
Lower	Male	Child	35	24	59
Lower	Male	Adult	1211	281	1492
Lower	Female	Child	17	27	44
Lower	Female	Adult	105	176	281
		Total:	1490	711	2201

Class	No	Yes	Total	Gender	No	Yes	Total	Age	No	Yes	Total
1st	122	203	325	Male	1364	367	1731	Child	52	57	109
Lower	1368	508	1876	Female	126	344	470	Adult	1438	654	2092

Exercise 24.

Using http://www.math.mcgill.ca/yyang/resources/doc/randomforest.pdf, write a set of 5 to at most 10 slides in order to present the concept of random forests, and how they work: selection of the attributes o which to branch, how to compute the information gain, etc.

NO cut and paste for the text: you need to write in your own words, however, you can cut and paste some figures.

Exercise 25.

Using https://www.mdpi.com/1999-4893/10/4/114, explain in your own words how to use random forests for prediction with time series. Write no more than 10 slides.

NO cut and paste for the text: you need to write in your own words, however, you can cut and paste some figures.