Artificial Intelligence - Practice Part

Contact Data

Marek Kubis

e-mail: mkubis@amu.edu.pl

www: http://www.staff.amu.edu.pl/~mkubis/

room: B3-19

office hours: Monday 18:45 - 19:45, Tuesday 12:00 - 13:00

General Guidelines

1. Set up a code repository (e.g. Git, GitHub, Bitbucket) and send the link to mkubis@amu.edu.pl.
2. Work iteratively and commit your solutions to the repository.
3. Preserve deadlines.
4. Grades are issued according to the following table

Table 1: Grades

Points	Grade
[0, 3.3)	2.0
[3.3, 3.8)	3.0
[3.8, 4.3)	3.5
[4.3, 4.8)	4.0
[4.8, 5.3)	4.5
[5.3, 6]	5.0

Task 1: Knowledge representation (1 point, deadline: March, 31)

Task 1.1

Design a semantic network that consists of at least 15 concepts. Introduce inheritance (is_a relation) and at least three other semantic relations.

Task 1.2

Design a conceptual graph that consists of at least 15 concepts.

Task 1.3

Design a frame system that consists of at least 15 frames. Introduce inheritance and other relations between frames.

Literature

Luger G. Stubblefield W., Artificial Intelligence: Structures and Strategies for Complex Problem Solving (3rd ed.), p. 300, 309-313, 320-324

Task 2: Search strategies (2 points, deadline: April, 30)

Task 1.1: Uninformed search strategies

Find a route from Szczecin to Kraków using the simplified map of directed roads defined by Table 2. Formulate the problem in terms of initial state, successor function, goal test and path cost. Solve the problem using the following search strategies:

- breadth first search
- uniform-cost search
- deep-first search
- deep-limited search
- iterative deeping deep-first search
- bi-directional search

The solution should consist of the road names (i.e. actions) that link Szczecin to Kraków. Implement the solution in your favorite programming language.

Table 2: Road Map

Source	Target	Route name	Route length
Szczecin	Gdańsk	T1	320
Szczecin	Poznań	T2	300
Szczecin	Wrocław	T3	330
Wrocław	Opole	T4	100
Opole	Katowice	T5	80
Katowice	Kraków	T6	100
Poznań	Opole	T7	240
Poznań	Łódź	T8	250
Poznań	Warszawa	T9	300
Warszawa	Kraków	T10	300
Łódź	Kraków	T11	220
Gdańsk	Olsztyn	T12	170
Olsztyn	Warszawa	T13	210
Poznań	Wrocław	T14	180
Wrocław	Opole	T15	100
Poznań	Łódź	T16	240
Katowice	Kraków	T17	95
Katowice	Poznań	T18	30
Warszawa	Szczecin	T19	560

Task 1.2: Informed search strategies

Find a route from Szczecin to Kraków using the simplified map of directed roads defined by Table 2 and the straight-line distance heuristic defined by Table 3. Formulate the problem in terms of initial state, successor function, goal test and path cost. Solve the problem using the following search strategies:

- Greedy best-first search
- A* search
- IDA* search

The solution should consist of the road names (i.e actions) that link Szczecin to Kraków. Implement the solution in your favorite programming language.

Table 3: Straight-line Distance between Cities

Source	Target	Straight-line distance
Katowice	Kraków	70
Łódź	Kraków	190
Warszawa	Kraków	250
Opole	Kraków	160
Wrocław	Kraków	240
Poznań	Kraków	335
Olsztyn	Kraków	415
Gdańsk	Kraków	485
Szczecin	Kraków	530
Kraków	Kraków	0

Literature

Russel S., Norvig P., Artificial Intelligence: A Modern Approach (2nd ed.), Chapter 3. "Solving Problems by Searching", Chapter 4.1. "Informed (Heuristic) Search Strategies"

Task 3: Machine Learning (3 points, deadline: May, 31)

Task 3.1: Genetic algorithms

Solve CNF-Satisfaction problem using a genetic algorithm.

Task 3.2: Neural networks

Train a Multi Layer Perceptron network (MLP) using BACK-PROP-LEARNING algorithm to recognize handwritten digits. Use the MNIST database http://yann.lecun.com/exdb/mnist/ for training and testing the network.

Task 3.3 Symbolic learning

Assume that you have to create a quality control machine, which removes from the assembly line objects having unacceptable properties (too big, too heavy, wrong color, wrong material, etc.). Design the concept space for this problem. Create a set of training instances (at least 20 examples). Implement the Candidate-Elimination Algorithm. Use the algorithm to learn the concept of an invalid object.

Literature

Luger G. Stubblefield W., Artificial Intelligence: Structures and Strategies for Complex Problem Solving (3rd ed.), p. 606-620, 715-719

Russel S., Norvig P., Artificial Intelligence: A Modern Approach (2nd ed.), p. 736-748