JUNAID · GIRKAR
60004190057
TE COMPS A4

18/12/21

ARTIFICIAL INTELLIGENCE

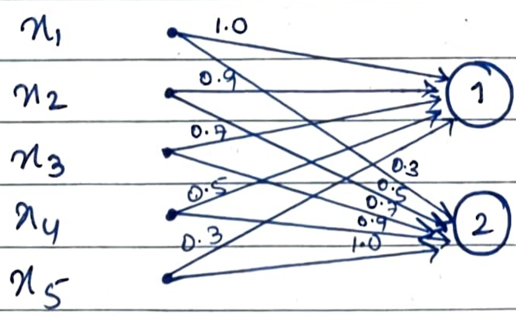
ASSIGNMENT - 2

Q1 Construct a KSOFM net with two cluster units and five input units. The weight vectors for the cluster units are given by $w1 = [1.0\ 0.9\ 0.7\ 0.5\ 0.3]$
$w2 = [0.3\ 0.5\ 0.7\ 0.9\ 1.0]$.
Use the square of the euclidean distance to find the winning cluster unit for the input pattern $x = [0.0\ 0.5\ 1.0\ 0.5\ 0.0]$. Using the learning rate of 0.25, find the new weights for the winning unit.

ANS $w1 = [1.0\ 0.9\ 0.7\ 0.5\ 0.3]$
$w2 = [0.3\ 0.5\ 0.7\ 0.9\ 1.0]$

Let the five input units $x1, x2, x3, x4, x5$ be

$x_1 = [1\ 0\ 0\ 1\ 0]$
$x_2 = [0\ 1\ 0\ 1\ 0]$
$x_3 = [0\ 0\ 0\ 1\ 1]$
$x_4 = [1\ 1\ 1\ 0\ 1]$
$x_5 = [1\ 0\ 0\ 0\ 1]$



STEP 1: $x_1 = [1 \quad 0 \quad 0 \quad 1 \quad 0]$

$D(j) = \| x_j - w_j \|^2$

$D(1) = (1-1.0)^2 + (0-0.9)^2 + (0-0.7)^2 + (1-0.5)^2 + (0-0.3)^2$

$= 1.64$

$D(2) = (1-0.3)^2 + (0-0.5)^2 + (0-0.7)^2 + (1-0.9)^2 + (0-1.0)^2$

$= 2.24$

$\because D(1) < D(2)$, the winning neuron is 1.

updating the weight for the 1st neuron.

$w_1(\text{new}) = w_1(\text{old}) + \alpha(x_1 - w_1(\text{old}))$

$w_{11}(\text{new}) = w_{11}(\text{old}) + 0.25(x - w_{11}(\text{old}))$

$= 1.0 - 0.25(1-1)$

$= 1.$

$w_{21}(\text{new}) = 0.9 + 0.25(0-0.9) = 0.675$

$w_{31}(\text{new}) = 0.7 + 0.25(0-0.7) = 0.525$

$w_{41}(\text{new}) = 0.5 + 0.25(1-0.5) = 0.625$

$w_{51}(\text{new}) = 0.3 + 0.25(0-0.3) = 0.225$

$\therefore w_1(\text{new}) = [1 \quad 0.675 \quad 0.525 \quad 0.625 \quad 0.225]$

$w_2 = [0.3 \quad 0.5 \quad 0.7 \quad 0.9 \quad 1.0]$

STEP 2: $x_2 = [0 \quad 1 \quad 0 \quad 1 \quad 0]$

$D(1) = (0-1)^2 + (1-0.675)^2 + (0-0.525)^2 + (1-0.625)^2 + (0-0.225)^2$

$= 1.5725$

$D(2) = (0-0.3)^2 + (1-0.5)^2 + (0-0.7)^2 + (1-0.9)^2 + (0-1)^2$

$= 1.84$

$\therefore D(1) < D(2)$, the winning neuron is 1

$w_{11}(new) = 1 + 0.25\,(0-1) = 0.75$

$w_{21}(new) = 0.675 + 0.25\,(1-0.675) = 0.70625$

$w_{31}(new) = 0.525 + 0.25\,(0-0.525) = 0.39375$

$w_{41}(new) = 0.625 + 0.25\,(1-0.625) = 0.71875$

$w_{51}(new) = 0.225 + 0.25\,(0-0.225) = 0.16875$

$w_1(new) = [0.75 \quad 0.70625 \quad 0.39375 \quad 0.71875 \quad 0.16875]$

$w_2 = [0.3 \quad 0.5 \quad 0.7 \quad 0.9 \quad 1.0]$

STEP 3: $x_3 = [0 \quad 0 \quad 0 \quad 1 \quad 1]$

$D(1) = (0-0.75)^2 + (0-.70625)^2 + (0-0.39375)^2 + (1-.71875)^2 + (1-0.16875$

$= 1.98$

$D(2) = (0.0-0.3)^2 + (0-0.5)^2 + (0-0.7)^2 + (1-0.9)^2 + (1-1)$

$= 0.84$

$\therefore D(2) < D(1)$, the winning neuron is 2.

$w_{12}(new) = 0.3 + 0.25\,(0-0.3) = 0.225$

$w_{22}(new) = 0.5 + 0.25\,(0-0.5) = 0.375$

$w_{32}(new) = 0.7 + 0.25\,(0-0.7) = 0.525$

$w_{42}(new) = 0.9 + 0.25\,(1-0.9) = 0.925$

$w_{52}(new) = 1.0 + 0.25\,(1-1.0) = 1$

$w_2(new) = [0.225 \quad 0.375 \quad 0.525 \quad 0.925 \quad 1]$

$w_1 = [0.75 \quad 0.70625 \quad 0.39375 \quad 0.71875 \quad 0.16875]$

STEP 4: $x_4 = [1 \quad 1 \quad 1 \quad 0 \quad 1]$

$D(1) = (1-0.75)^2 + (1-0.70625)^2 + (1-0.39375)^2 + (0-0.71875)^2 + (1-0.1$

$= 1.724$

$D(2) = (1-0.225)^2 + (1-0.375)^2 + (1-0.525)^2 + (0-0.925)^2 + (1-1)^2$
$= 2.0725$

$\because$ $D(1) < D(2)$, winning neuron is 1

w_{11} (new) $= 0.75 + 0.25(1-0.75) = 0.8125$
w_{21} (new) $= 0.7062 + 0.25(1-0.7062) = 0.7796$
w_{31} (new) $= 0.3937 + 0.25(1-0.3937) = 0.5452$
w_{41} (new) $= 0.7187 + 0.25(1-0.7187) = 0.5390$
w_{51} (new) $= 0.1687 + 0.25(1-0.1687) = 0.3765$

w_1 (new) $= [0.8125 \quad 0.7796 \quad 0.5452 \quad 0.5390 \quad 0.3765]$
$w_2 \quad = [0.225 \quad 0.375 \quad 0.525 \quad 0.925 \quad 1]$

STEP 5: $x_5 = [1 \quad 0 \quad 0 \quad 0 \quad 1]$

$D(1) = (1-0.8125)^2 + (0-0.7796)^2 + (0-0.5452)^2 + (0-0.539)^2 + (1-0.3765)^2$
$= 1.6194$

$D(2) = (1-0.225)^2 + (0-0.375)^2 + (0-0.525)^2 + (0-0.925)^2 + (1-1)^2$
$= 1.8725$

$\because$ $D(1) < D(2)$, the winning neuron is 1

w_{11} (new) $= 0.8125 + 0.25(1-0.8125) = 0.8593$
w_{21} (new) $= 0.7796 + 0.25(0-0.7796) = 0.5847$
w_{31} (new) $= 0.5452 + 0.25(0-0.5452) = 0.4089$
w_{41} (new) $= 0.5390 + 0.25(0-0.5390) = 0.4042$
w_{51} (new) $= 0.3765 + 0.25(1-0.3765) = 0.5323$

w_1 (new) $= [0.8593 \quad 0.5847 \quad 0.4089 \quad 0.4042 \quad 0.5323]$
$w_2 \quad = [0.225 \quad 0.375 \quad 0.525 \quad 0.925 \quad 1]$

After 1st iteration, the weight vectors are

$$w_1 = [0.8593 \quad 0.5847 \quad 0.4089 \quad 0.4042 \quad 0.5323]$$
$$w_2 = [0.225 \quad 0.375 \quad 0.525 \quad 0.925 \quad 1]$$

After 2nd iteration, the ~~weight vectors are~~ learning rate is

$$\alpha(t+1) = 0.5\,\alpha(t) = 0.5 \times 0.25 = 0.125$$

Q2 Discuss Atari Application using AI

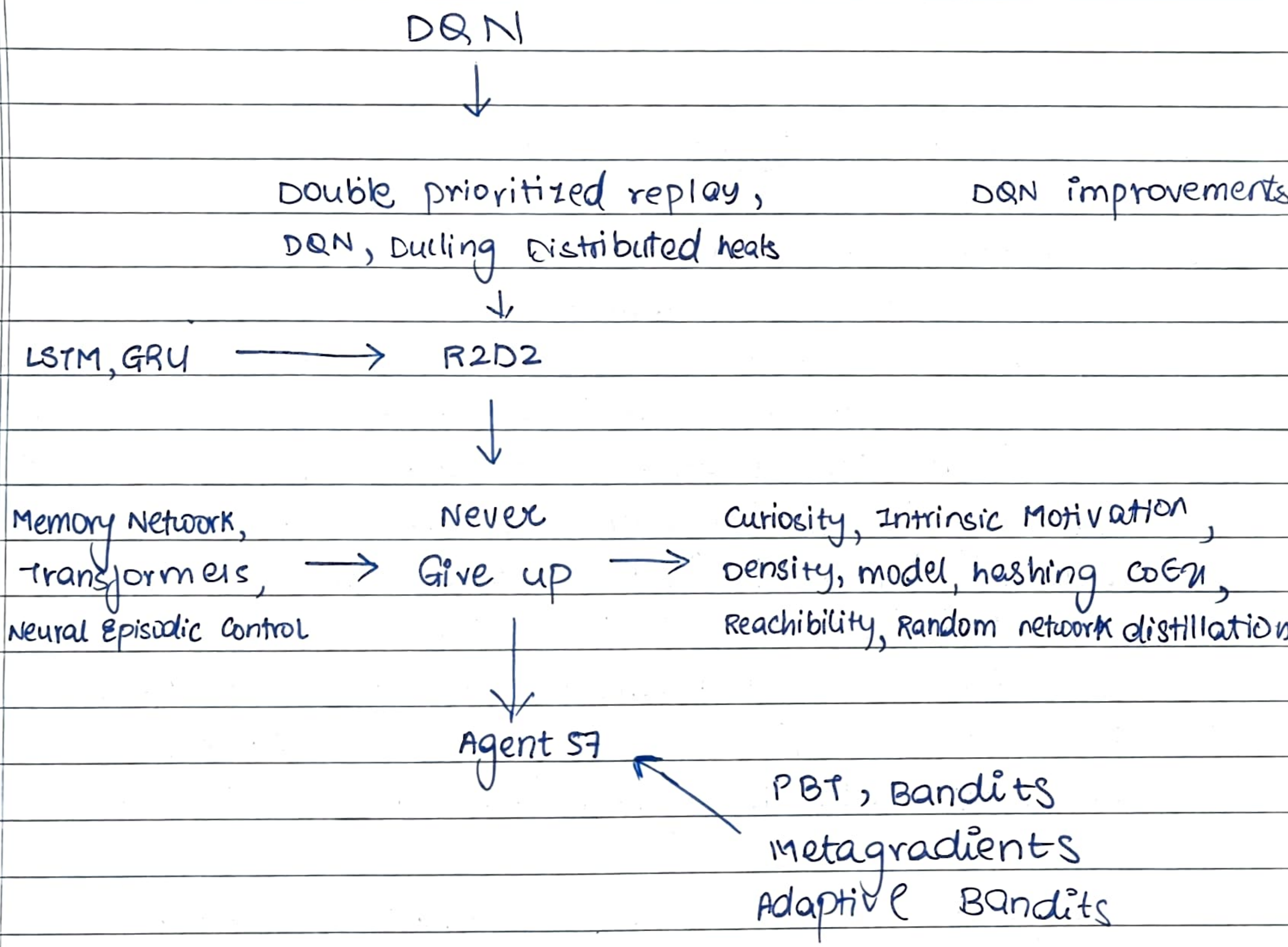
ANS The truth of Atari video game are a great way to test AI. They provide variety of challenges that force an AI to dense number of strategies yet they have a clear measure of success, a score to train against.

The artificial intelligence agent called Agent 57 has learned to play all 57 Atari video game in arcade learning environment - a collection of classic games that researchers use to test the limits of their deep learning models. Developed by Deep Mind Agent 57 uses the same deep reinforcement learning algorithm to achieve superhuman gameplay levels even in the game that previous IA's have struggled with.

Being able to learn 57 different task makes Agent 57 more versatile than previous game's AI.

The first attempt to tackle Atari 57 challenge was the Deep Q-network agent and subsequently variations of it. Despite the notable advacement most DRL agents fails to generalize knowledge to diverse task. Games such as Montezuma's Revenge and Pitfall require extensive exploration to obtain good performance.

To address some challenges the DQN model evolved incorporated several advancements in deep learning space. specifically the development of techniques such as short-term memory and episodic memory played an important role in development of more advanced DRL agents



Training an AI to excel at more than one taste of biggest open challenge in deep learning. The ability to learn 57 different task make Agent 57 more versatile than previous AI game but it still can't learn to play more than one game at a time. Agent 57 can learn to play 57 games, but it cannot learn to play 57 games at once.

Q3 Write a short note on

(i) SENSORLESS PLANNING :

Sensorless planning is a kind of planning that is based on any perception. The algorithm ensure that plan should reach its goals at any cost.

Sensorless planning is also known as conformant planning is the problem of finding a sequence of actions for achieving a goal in presence of uncertainity in the initial state or action effects.

Sensorless planning handles domains where the state of the world is not fully known. It comes up with the plan that works in all possible case

example :
- You have a wall made of bricks
- You have a can of white paint
- Action : Pain (brick), effect : color (brick, white)
- Goal ; Every brick should be painted white.
- Suppose the world isn't fully observable - we actually cannot observe the brick color.

(ii) MULTI AGENT PLANNING :

In computer science, multi-agent planning involves co-ordinating the resources and activities of multiple agents.

It can involve agents for planning for a common goal, an agent co-ordinating the plans or planning of others, or an agents refining their own plans while negotiating over tasks or resources.

If ~~a~~ new agent are introduced to a single agent environment, but the single agent does not change its basic algorithm then it may perform poorly.

Agents are not indifferent to another agents intentions (like nature is). So agents can co-operate, complete ~~or~~ or co-ordinate.

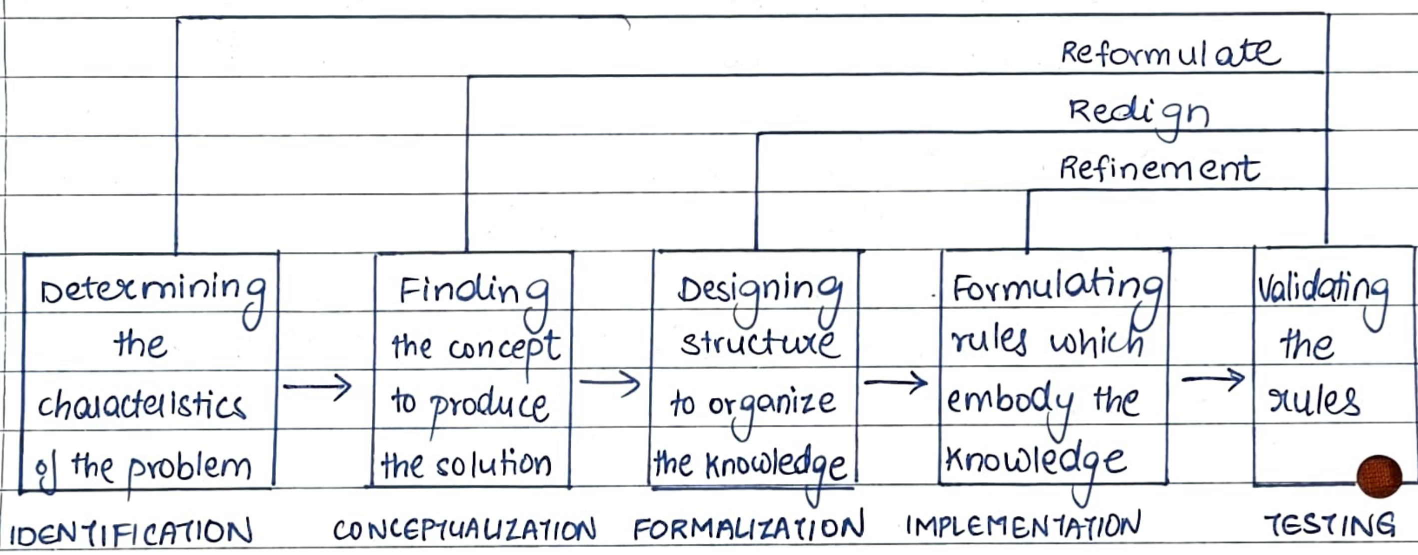
Sometimes distributed computation are easier to understand and develop.

Example : In double tennis problem if each agent uses different plan then neither will return the ball.

Having a correct joint plan doesn't guarantee success, the agent needs to arrive at some joint plan. To arrive at that particular joint plan it must be common knowledge and agents must have a mechanisms for co-ordination.

Q4 explain phases in building Expert Systems with ES architecture in detail.

ANS. There are 5 phases while developing an expert system :
(i) Identification
(ii) Conceptualization
(iii) Formalization
(iv) Implementation
(v) Testing (validation, verification and maintainence)



1) IDENTIFICATION

a) Before developing an expert system, it is important to describe, with as much precision as possible, the problem which the system is intended to solve.

b) The exact nature of the problem and start the precise goals which indicate exactly how the expert system is expected to contribute to the solution, these all must be determined.

6

~~~~

2) CONCEPTUALIZATION :

a) Once it has been identified for the problem an expert system has to solve, the next stage involves analyzing the problem further to ensure that its specifics as well as generalities are understood

b) The conceptualization stage involves a circular procedure of iteration and reiteration between the knowledge engineer and the domain expert.

c) When both agree that the key concepts - and the relationships among them - have been adequately conceptualised, this stage is complete.

3) FORMALISATION :

a) During the identification and formalisation stages, the focus is entirely on understanding the problem.

b) During the formalisation stage, the problem is connected to its proposed solution, an expert system is supplied by analyzing the relationships depicted in the conceptualization stage.

c) The knowledge engineer begins to select the techniques which are appropriate for developing the particular expert system.

d) During formalisation, it is important that the knowledge engineer be familiar with the following :

(i) The various techniques of knowledge representation and intelligent search techniques used in expert systems.
~~~~

(ii) Expert system tools which can greatly expidite the development process.

(iii) other - expert systems which may solve similar problems and thus may be adaptive to problem at hand.

e) The formalisation process is often the most interactive stage of expert system development, as well as the most time consuming.

f) The knowledge engineer must develop a set of rules and ask the domain expert if those rules adequately represent the expert's knowledge.

g) This process is also interactive : the rule review is repeated and the rules are refined continually until the results are satisfactory.

4) IMPLEMENTATION :

a) During the implementation stage the formalised concepts are programmed into the computer which has been chosen for system development, using the predetermined techniques and tools to implement a prototype of the expert system.

b) If the prototype works at all, the knowledge engineer may be able to determine if the techniques chosen to implement the expert system were the appropriate ones.

c) Once the prototype system has been refined sufficiently to allow it to be executed the expert system is ready to be tested thoroughly to ensure that it expertise's correctly.

5 TESTING :

a) Testing provides an opportunity to identify the weaknesses in the structure and implementation of the system and to make the appropriate corrections.

b) Depending on the types of problems encountered the testing procedure may indicate that the system was implemented incorrectly, or perhaps that the rules were implemented correctly but were poorly or incompletely formulated.

c) Once the system has proven to be capable of correctly solving straight-forward problems, the domain expert suggests complex problems which typically would require a great deal of human expertise.

d) The testing process is not complete unless it indicates that the solutions suggested by the expert systems are consistently valid as those provided by a human domain expert.