Roshan B
D15C
4

# Assignment - 1

04/05

1 What is AI? Considering the COVID-19 pandemic situation, how AI helped to survive and renovated our way of life with different applications?

→ Artificial Intelligence (AI) is the simulation of human intelligence in machines that can learn, reason, and make decision It includes machine learning, neural networks, and natural language Processing.
During COVID 19, AI played a crucial role in virus detection, vaccine research, and patient care. AI Powered Chatbots provided health advice, while medical imaging AI helped diagnose infections. AI optimized logistics, ensuring smooth supply Chain management during lockdowns.

2. What are AI Agents terminology, explain with examples.

An AI agent is an autonomous entity that perceives its environment and takes action to achive goals

Simple Reflex Agent:- Act only on current data (eg- thermostat, adjusting temperature)

Model Based Agents :- Maintain and internal model of the world (eg self driving cars)

Goal Based agents :- make decisions to achive gaals (eg Chess playing AI)

3. How AI technique is used to sdve 8 puzzle problem?

→ The 8 puzzle problem involves arranging numbered tiles in order by moving them into an empty space
AI techniques like BFS, DFS and A* algorithm are used to solve it BFS explores all possible moves level by level ensuring the shortest. DFS searches deep but may not be optimal. the A* algorithm is efficient using a heuristic like Manthan Distance (h) to estimate remaining moves.

4. What is PEAS descriptor? Give PEAS descriptor for following :-

| | |
|---|---|
| Medical diagnosis system | A robotic Sentry gun for the Keck lab. |
| A music composer | |
| An aircraft autolander | |
| An essay evaluator | |

PEAS (Performance Measure, Environment, Actuators, Sensors) is a framework used to describe the structure of intelligent agent.

PEAS Descriptors for Given Systems:

1. Taxi Drivers.

P: Passengers safety, fuel efficently, shortest route, Customers satisfaction

E: Roads, traffic, Pedestrians, weathers condⁿ

A:- steering, accelerator, brakes, GPS, free meterse

S: Cameras, GPS, Speed sensors, fuel gauge

2 Medical Diagnosis System:-

P: Accurate diagnosis, treatment effectiveness

E: Patient symptoms, medical records

A: Display results, Prescribing

S: Patient history, test inputs

3 Music Composers:-

P: Melody harmony, originality, audence satisfaction

E: Musical trends, user preferences, genre constraints

A: notes generation, instrument selection

S: User feedback, existing music

4 Aircraft Auto landers:
P: Safe landing, fuel efficency, smooth touchdown
E: Weather, runway conditions, altitude
A: Engine throttle, landing gear.
S: Altimeters, GPS.

5 Essay Evaluator:-
P: Grammar accuracy, coherence, relevance to topic
E: Essays submited, language rules, grading
A: Highlighting mistakes, Provides scores
S: Text input, grammar rules.

6 Robatic Sentery for keck Lab:-
P: Accuracy in targeting threats, avoiding false alarm
E: Lab envirsoments, intruders, authorized personnel
A: Rotating gun turret, fixing mechanism
S: motion sensors, cameras, thermal detectors.

5 Categorize a Shopping bot for an offline bookstore according to each of the six dimensions

Observability :- the bot may not have full access to inventory changes or Customer Preferences.

Stochastic :- Book availability and customer choices are uncertain.

Sequential :- Each shopping decision affects future ones, like recommedations.
Dynamic :- Book availability may change in real time.
Discrete :- Limited choices
Muti Agent :- Interacts with customers, suppliers

6 Differentaite Model based and utility agent.

| Model Based Agent | Utility Based agent. |
|---|---|
| ① Maintains an internal model of world. | ① Uses a utility func to measure performance |
| ② ~~Uses~~ Understanding and predicting the environment | ② Choosing the most optimal action based on utility. |
| ③ Adapts by updating its model of the world | ③ Considers multiple possible outcomes for the best decision |
| ④ Requires maintaining and updating a model. | ④ Requires defining and calculating utility values. |

⑦ Explain architecture of a knowledge based agent and learning based agent.

4. Knowledge - Based Agent Architecture Consists of a Knowledge Base (KB) storing facts and rules, and an Interference Engine that drives conclusion.
It uses perception to gather information and Reasoning to make decision
The agent updates its KB based on new inputs and Reasoning to make decision.

Learning Agent Architecture:- Includes a Learning element that improves performance by adapting to experience.
It has a Platform Element that selects actions, a critic that evaluates them, and a Problem generator that suggest improvements

5. What is AI? Considering the COVID-19 Pandemic situation, how AI helped to survive and renovated our way of life with diff applications?

Artifical Intelligence (AI) is the branch of Computer science that enables machines to simulate human - Intelligence.
AI Role during the COVID 19
Early Detection and Diagnosis. -
AI powered models analyze large datasets to detect COVID 19 outbreaks early, aiding in rapid response.

(1) Early detection and Analysis
(2) Medical Imagning and Diagnosis
(3) Drug Discovery and Diagnosis.
(4) Contact Tracing and surveillance
(5) Fake News detection.
(6) Supply Chain and automations

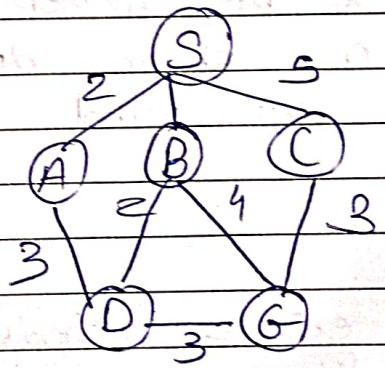
9 Convert following to Predicates.
Anita travels by car if available otherwise travels by bus.

$\forall x$ (Available (car)) → Travels (Anita, (car)) ∧
(¬ Available ((ar)) → (Travels (Anita, Bus))

Bus goes via Andheri and Goregaon.
Goes via (Bus, Andheri) ∧ Goes Via (Bus, Goregaon)

Car has a puncture, so it is not available.
Has puncture (car) → Available (car)

10 BFS



Vertices {S, A, B, C, D, G}

S → A (2)

S → B (5)

S → C (5)

A → D (3)

B → D (2)

B → G (4)

C → G (3)

D → G (3)

Q = [S]

Queue = [A, B, C]

Queue = [B, C, D]

Queue = [C, D, G]

Goal G is found

S → B → G

is the shortest path.

11. What do you mean by depth limited search? Explain Iterative Deepening search with example.

Depth limited search (DLS):

is a variation of DFS where the search is restricted to a specific depth limited LL If a goal is not found within limit, the search terminates.

Eg:-

If L=2, DFS explores nodes only upto depth 2.

Iterative Deepening Search (IDS)

Iterative Deepening Search combines the space efficency of DFS and completeness of BFS by repeatdly running DLS with increasing depth limits L= 0,1,2.. dots L= 0,1,2 _ until goal is found.

Example.

1. DLS with L=0: only check S, goal not found.
2. DLS with L =1 Expand S, explores A,B,C goal not found
3. DLS with L = 2: Expand A,B,C explores D,G goal found at Depth 2

Advantage:- Guarantees finding the shortest Path while using less memory than BFS

12 Explain Hill climbing and its drawbacks in detail with example. Also state limitations of steepest - ascent hill climbing.

Hill Climbing Algorithm:- Hill climbing is a local search algorithm thant contionusly moves towards the best neighboring state with a higher heuristic value, aiming to reach a global optimum.

Steps :-
1. Start from an intial state
2. Evaluate neighboring states and move to the one with the highest heuristic value
3. Repeat until no better neighbor exists (local max)

Example :-
Imagine a mountain climbing scenario where a hiker moves uphill based on the steepest slope.

Drawbacks of Hill Climbing :
1. Local Maximum :
The algorithm may stop at a peak that is not the global optimum.
Eg :- A small hill before a taller mountain.
2. Plateau Problem :-
A flat region where all neighboring states have the same heuristic value, causing the search to halt.
3. Ridges :
The algorithm may fail to climb diagonally when only direct moves are considered.

Limitations of Steepest-Ascent Hill Climbing:
1 Slow Progress in narrow Ridges - If the optimal path requires small incremental changes, the algorithm struggles.

2. More likely to Get Stuck in Local Maxima :- Choosing only the steepest path can lead to premature convergence.

13 Explain Simulated annealing and write its algorithm.

Simulated Annealing :- is a probablistic optimization algorithm inspired by the annealing process in metallurgy. It helps in finding a global optimum by allowing occasional moves solutions to escape local optima

Algorithm :-

1 Intalize the current State and Temperature T.

2 Repeat until Stopping cond$^n$.
Select a random neighbhor State.
Compute energy diff $\Delta E = E_{new} - E_{current}$
If $\Delta E < 0$, accept the new state.

3 return the best foundation Sol$^n$.

14 Explain $A^*$ Algorithm with an example.

The $A^*$ algorithm is an informed search algorithm that finds the optimal path by consedering both the cost to reach a node g(n) and the

estimated cost to the goal $h(n)$

$$f(n) = g(n) + h(n)$$

$g(n)$ = actual cost from start node $n$
$h(n)$ = estimated cost from $n$ to goal
Eg:-
In a graph search problem, if S is the start G is the Goal, $A^*$ expands nodes based on the lowest $f(n)$ value, ensuring the shortest path is found efficently.

15. Explain MiniMax Algorithm and draw game tree for tic tac toe.

The MiniMax Algorithm is used to in adversarial search to determine the optimal move by assuming both plays optimally.
Maximizer (eg X):- tries to get the higest score
Minimizer (eg. O):- tries to reduce the score.
Algo:-
1. Generate the game tree up to depth limit
2. Assign heuristic values to leaf nodes.
3. Backpropgate values:-
   Maximizer picks the max value
   Minimizer pick the min value.
4. the root node selects the best move based on propgated values.

16. Explain Alpha - Beta pruning Algorithm for adversarial search with an example.

Alpha- Beta pruning optimizes Minimax by skipping unnecessary branches, reducing computations. It uses two values:-

Alpha ($\alpha$) = Best score Maximizer can achive.

Beta ($\beta$) = Best score Minimizer can achive.

Algorithm :-

1 Perform Minimax search

2 Prune branches where :-

If Maximizer's best ($\alpha$) > Minimizers best ($\beta$), further evalution is skipped

If Minimizers best ($\beta$) ≤ Maximizers best ($\alpha$) further evalution is skipped.

3 This reduces time complexity

Eg -

In a game tree, if a branch has already provided a worse outcome that the best know move, it is ignored to save time.

17. Explain WUMPUS world Environment giving its PEAS description. Explain how percept sequence is generated?

The WUMPUS World is a grid-based AI environment where an agent explores a cave while avoiding hazards like pits and the Wumpus monster.

PEAS

Performance Measure :- Reaching the gold safely, minimizing steps.

Environment : A 4x4 grid with pits, gold and WUMPUS.

Actuators :- Move forward, turn, grab, shoot, climb.

Sensors :- Perceive Stench, breeze, glitter.

Percept Sequence Generation :-

1. Agent starts (1,1) sensing its surrounding
2. If Stench is detected, the WUMPUS is nearby.
3. If breeze is detected, a pit nearby.
4. The agent infers safe paths and navigates toward the gold while avoiding hazards.

18 Solve the following crypto-arithmetic problem SEND + MORE = MONEY

```
  S E N D
+ M O R E
---------
M O N E Y
```

```
  9 5 6 7
+ 1 0 8 5
---------
1 0 6 5 2
```

S=9, E=5, N=6, D=7, M=1, O=0, R=8, Y=2

19

Represent these axioms in (FOPL):-

1. $\forall x$ (Graduating $(x) \rightarrow$ Happy $(x)$)
2. $\forall x$ (Happy $(x) \rightarrow$ Smiling $(x)$)
3. $\exists x$ (Graduating $(x)$)

Convert each formula to clause form.

1. Graduating $(x) \vee$ Happy $(x)$
2. Happy $(x) \vee$ Smiling $(x)$
3. Graduating $(A)$

Prove that "Is someone smiling" using resol technique.

Graduating $(A)$

1 Happy $(A)$

2 Smiling $(A)$

Resol$^n$ Tree.

1 Graduating $(A)$

2 Graduating $(A) \rightarrow$ Happy $(A) \rightarrow$ Happy $(A)$

3 Happy $(A) \rightarrow$ Smiling $(A) \rightarrow$ Smiling $(A)$

20 Explain Modus Ponens with a suitable example.

Modus Ponens is a rule of inference Stating :-

$P \rightarrow Q$ , $P \Rightarrow Q$

Eg:-

1 If it rains, the ground gets wet.
2 It's raining.
3 Conclusion :- The ground is wet.

21. Explain Forward Chaining and Backward Chaining algorithm with an eg.

Forward chaining

Data driven inference:- Starts from known facts and applied rules to reach goal.

Used in expert systems.

Eg.

Fact "sore throat"

Rule : " If sore throat → infection".

New fact " If infection → need antibiotics"

Conclusion:- "Need antibiotics"

## Backward Chaining:-

Goal driven interference : starts from the goal and works backword to find supporting facts.

Used in AI reasoning and theorem proving.

Eg:-

Goal: Does the patient and antibiotics

1 Check Dose patient have an infection.

2 Check Dose the patient have some tohart?

3 If both hold, concluded "need antibiotics"

This reduces unneccesary complulations by exploring relevent facts.