14/07/2025

Portion

Intelligent Agents and Problem Solving

Knowledge representation and reasoning

Supervised Learning

Unsupervised Learning and Enscemble Technique

Neural Technique

**Artificial Intelligence**

machine Displaying intelligence that simulates human behavior or overthinking and can be trained to solve

specific Problems.

AI is a combination of machine Learning technique and Deep Learning

Types of Artificial Intelligence models are trained using vast volumes of data and can make intelligence data.

Generative AI:

Advance application AI by Creating Original Content Such as text, images and code based on user prompts unlike traditional AI ,it forces on creativity and human like interactions.

Domains:

E-commerce

Education

Life Style

Robotics

Neutral Language Processing.

Gaming

Automobiles

Social Media

**Artificial intelligence Search Problem:**

**Solve Maze using breath First Search(BFS) Algorithm:**

AI->Agent perform search algorithm to archive the task.

A search problem in Artificial Intelligence consist of:

* State space: refers to all possible state value where the agent can be.
* Start state:
* Goal State:
* Solution State:
* Cost:

Breadth First Search(BFS)

* BFS will visit all current depth level nodes before proceeding to the nodes at the next level of depth.
* Why BFS?
* Explore to all neighbor level by level.

Start @ given starting point

Use the queue to explore neighboring cells.

Marks visited cell to avoid revisiting

When the goal is found stop and backtrack.

**17/07/205**

**Types of Environment in AI**

An environment in artificial intelligence is the surrounding of the agent.

The agent take input from the environment through sensors and delivers the output to the environment through actuators.

There are several types of environments:

* Fully observable Vs partially observable
* Deterministic vs Stochastic.
* Completive vs collaborative
* Single-agent vs Multi agent
* Static vs Dynamic
* Discrete vs continuous
* Episodic vs sequential
* Known vs Unknown

**Fully Observable Vs Partially Observable**

When an agent sensor is capable to sense or access the complete state of an agent at each point in time it is said to be a fully observable environments

**Intelligent agents:**

Perception.

Reasoning and Decision Making

Learning

Action

**Agents and Environment**

Ex: Vacuum cleaner

**PEAS**:

Performance measure

Environment

Actuators

Sensors

Ex: Automated Taxi Driver.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Agent Type** | **Performance Measure** | **Environment** | **Actuator** | **Sensor** |
| Medical Diagnosis System | Healthy Patient  Patient hospital | Staff | Display | Touch Screen |

**Environment In AI / Example**

**Determine Vs Stochastic:** Tic-tac-Toe and Ludo

**Episodic vs Sequential :** Part picking robort

**Static Vs Dynamic :**Cross word puzzle

**Known vs Unknown :**Card Game

**18/07/2025**

**Search Algorithm**

Start Space: Goal State:

**Search Technique:**

**Uninformed Search Informed Search**

Breadth-first Search Best First Search

Depth-First search

Bi-Directional search

Depth limited search

Iterative Deepening DFS

Uniform Cost Search

**Breath-First-Search:**

Explain all nodes at Current Depth before Moving to next level.

**Depth-First-Search:**

Explore as far as possible along each path before backtracking.

**Depth-Limited-search:**

It prevents infinite loop

Useful when the goal is known.

Cannot find solution beyond limit.

**Iterative Deepening Depth-First-Search:**

Combines BFS AND DFS. By running DFS with increasing depth limit until a solution is found.

Ensure completeness and optimality.

**Uniform-Cost-Search:**

Variation of dijikstra’s algorithm.

Used to find the minimum path from the source node to the destination node around a directed Weighted graph.

Uses brute forces approach

|  |  |  |
| --- | --- | --- |
| A-0 |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F |
| F | F | F | F | F | F |

**Application of Uniformed Search Algorithm:**

Path Finding.

Puzzle Solving

Game AI

Robot Navigation

Web Crawling

**21/07/2025**

**Informed Search Algorithm:**

Heuristic Search

Enhance Efficiency

To determine best node to visit Next.

An Evaluation Function f(n) is used.

Based on how f(n) is defined informed search algorithm Is classified into 2 types :

Greedy Search:Select the next node based on lower heuristic cost.

A search\*:

**Greedy-Best-Search:**

Follow a Goal driven approach

Always expanding the node that appears closest to heuristic algorithm.

Steps:

1. Start at initial code
2. Expand the code with lowest A(n)
3. Add neighbor to the open list
4. Repet until

EX:

A

/ \

B C

/ \ / \

D E F G

| **Node** | **h(n)** |
| --- | --- |
| A | 6 |
| B | 5 |
| C | 2 |
| D | 4 |
| E | 3 |
| F | 1 |
| G | 0 (goal) |

**Path:**

**A → C → G**

**A\* Search:**

Combines the best expects of Greedy Search and uniform Search

**A\*** is a popular **informed search algorithm** that finds the shortest path to the goal node using both:

* **g(n):** Cost from the start node to the current node n
* **h(n):** Heuristic estimate of the cost from node n to the goal

It uses the evaluation function:

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

Example:

A

/ \

B C

/ \ / \

D E F G

A-B: 1, A-C: 3

B-D: 3, B-E: 1

C-F: 1, C-G: 5

**A → C → F → G**

* **Total cost = g(G) = 8**

**24-07-2025**

**Local Search and Optimization problem:**

* In many Optimization problems,the path to the goal is irrelevant;
* The goal state itself is the solution
* State space=set of “complete” configurations.

**Local Search Algorithms:**

“The path cost Does not matter the focus is on only solution State.”

* Keep a single “current” state, or small set of states.
* Iterative try to improve it.
* Very memory efficient.

**Key Features:**

1. Objective function
2. Search space
3. Neighbourhood structure
4. Elevation criteria

**Working of Local search Algorithm**

**Global minimum:**

If the elevation corresponds to the cost then the task is to find the lowest valley. Which is known as global minimum.

**Global Maxima:**

If the elevation corresponds to an objective function then its finds the highest peak which is called Global maxima. It’s the highest point in the valley.

**Types of Local Search:**

1. Hill-climbing Search
2. Simulated Annealing
3. Local Beam Search

**i. Hill climbing Algorithm in AI**

* To climb and reach the topmostn peak/point of that hill
* Its based on the heuristic search technique.

**How hill climbing Works:**

* Initialization
* Evaluation
* Neighbor selection
* Termination

**State-space Landscape of Hillclimbing algorithm.**

1. Global minimum
2. Local maximum
3. Flat Local maximum
4. Shoulder
5. Current space

**Types of Hill Climbing Search Algorithm:**

1. Simple Hill Climbing
2. Steepest Hill climbing
3. Stochastic Hill climbing

**Simple Hill climbing Algorithm**

1. Create a Current node, neighbor node and goal node
2. If the current node=goal node return goal node
3. Else current node<=neighbor node =move Ahead.

**Steepest ascent hill climbing**

1. Create a Current node, neighbor node and goal node
2. If the current node=goal node return goal node
3. Loop until a better node is not foiund to reach the solution
4. If there is a better successor node expand it

**Stochastic hill climbing:**

1. Stochastic hill does not focus on all nodes.
2. It select one node at a random and decides whether it should expand or search better one.

**Limitation of Hill climbing Algorithm:**

**1.Local Maxima:** May get stuck in suboptimal solutions

**2.Plateau**:Flat regions in the search may slow down the process

**3.Ridges:**Narrow peak can be challenging to navigate

**Example**: Robot trying to reach the highest point on a hill.

**Stimulated Annealing:**

The algorithm introduces randomness to overcome the limitation of traditional local search methord,such as getting stuck in the local optima.

**Example:** In scheduling problem such as staff shifts simulated annealing explores various schedule.

**Local beam search:**

* Algorithm that maintains multiple candidate solution simantaneously.
* Analysing the neighborhood of all candidate and select the most promising ones for further exploration.

Example: Root optimization.

**Extra Technique:**

1.Tabu search

2.Genetic Algorithm

**Constrain Satisfaction Problem(CSP):**

Is a problem where the goal is to find the constrains

Key components of CSP:

1. Variable
2. Domains
3. Constrains

**Standard search formation**

* Initiate state
* Sucessive state

Example:

STEPS:

1. define graph node as csp variable

2.Assign colour set as adomain

3. Add constrain adjacent node not equal to size of csp.

Constrain Graph:

Binary CSP:Each constrain relates at most two variable

Constrain Map

**Variable and value Ordering:**

Minimum Remaining Values:

Choose the variable with the Fewest legal values.

Degree heuristic:

If there are several MRV variable, We can use degree heuristic.

Least constraining Value:

When we are using MRV and degree heuristic we will be using least constraint values.

**Inter Leveling Search and Interfaces:**

* Forward checking