Artificial Intelligence and Neural Networks

16/01/2024

Q What is AI?

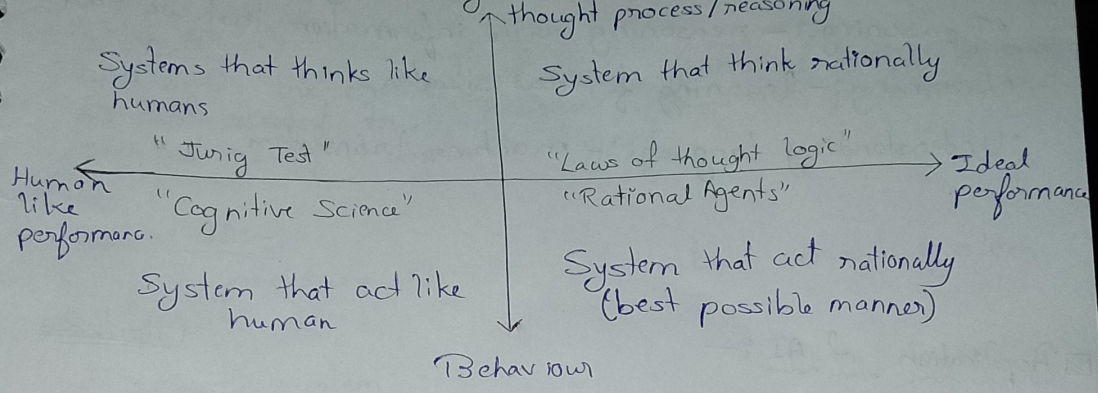
Ans AI is concerned with the design of intelligence in an artificial desire. AI is discovered by John McCarthy in 1956.

Intelligence means having behaviour which is like a human.

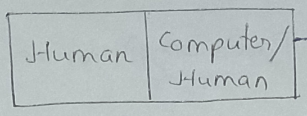
There are two types of behaviour

① Thinking (reasoning property) ② Acting (performed action)

Thought process/reasoning vs behaviour :-



Type question & answer receive on the screen



→ process question then return answer

If the interrogator cannot reliably distinguish the human from the computer then the computer does process (artificial) intelligence.

Problems of AI :- Intelligent entities (or "agent") need to be able to do both mudane and expert tasks

expert tasks
- medical diagnosis
- Mathematical problem solving approach

mudane
- planning reuse
- activity, recognising vision people, objects
- communicating (NLP)

Intelligent Behaviours :-

- Perceptence :- able to see, hear, sensory information
- Reasoning :- reasoning with information that we have.
- Learning :- learning for new situation.
- Understanding language :- machine translation

17/01/2023

Application of AI :-

- Computer Vision
- Image recognition
- Robotics
- NLP (Natural Language Processing)
- Speech processing

Impact of AI :-

- Self driving car
- Machine translation

Internet Agent :-

The explosive growth of the internet has also led to growing interest in internet agent to

i) Monitor users task
ii) Seek needed information
iii) Learn which information is most useful.

Approaches to solve AI tasks:-

i) **Strong AI:-**
Aims to build machine that can truly reason and solve problem which is self aware and whose overall intellectual ability is indistinguishable from that of a human being
- Human like
- Non-human like

ii) **Weak AI:-**
Deal with the creation of some form of computer based AI that cannot truly reason and solve problems but can act as if it were intelligent

iii) **Applied AI:-**
Aims to produce commercial viable smart system such as -

i) Security system

iv) **Cognitive AI:-**
How the human mind works

Q What AI system can do?
→ Computer vision
→ Robotics (IoT)
→ NLP (Natural Language Processing)
→ Spoken Language
→ Learnig text categorization.

Q what cannot AI system?
Q Limitation of AI system.

Software Agents (Softbot):—

→ Functions are Key strokes, file contents, network packet as input.
→ Functions are actuators (writing files and sending network packet)

- Types of Agents—
  1. Robots
     - Xavier (CMo)
     - Cog (MIT)
     - Aibo (Sony)
  2. Expert System Space craft
  3. Intelligent building

Fundamental faculties of intelligence – ~~cost~~ (without sense its not intelligence)
① Acting ② Sensing ③ Understanding, ④ Reasoning, Learning (operate on changing environment)

23/01/2024

Rational Agent:—

→ AI is about building rational agent.
→ A rational agent always does the right thing.
→ What are the fundamental goals?
What are the components?
How do we build them?

(i) Perfect Rationality:—

- Assume that the ~~traditions~~ rational agents knows all and will take the action that maximize his/her utility
- Human do not satisfy this definition of rationality

(ii) Bounded Rationality:—

- Because of the limitations of human mind. Human must use approximate method to handle many tasks.
This are also known as Realistic Agent.

Rational Action:—

The action that maximize the expected value of the performance measure given the percept sequence to date

Rational = Best?

→ Yes to the best of its knowledge

• Does rational is optimal?

→ Yes, to the best of its ability and its constraints

• Is Rational agent omniscient?

→ A rational agent is not omniscient

Agent Environment:—

• Environment : Observability

(i) Fully observable:—

All of the environment relevant to the action being considered is observable

Example — Chessboard.

(ii) Partial observable:—

→ The relevant feature of the environment are only partially observable.

→ Might be partially because of noisy and inaccurate sensors.

Example — Cards, Poker, Automated vehicles/self-driving car

• Environment : Deterministic

(i) Deterministic:—

The next state of the environment is completely described by the current state and the agents action

(ii) Stochastic:—

→ If an element of interference or uncertainity occurs then the environment is stochastic

→ If you cannot observe the ~~envio~~ environment fully, the environmd may appear as stochastic where it is actually deterministic, ~~to the~~ if you access the entire environment.

Example — Poker, Ludo

(iii) Strategic :—

24/01/2024

⊛ Environment —
→ Fully observable or partially
→ Deterministic, stochastic, Strategic
→ Episodic, sequential
→ static, dynamic
→ Single Agent, Multi Agent

• Episodic Environment:—
An episodic environment means that subsequent episode do not depend on what actions occured in previous episode. In each episode receive a percept and then perform a single action.

• Sequential Environment:—
In a sequential environment the agent engage in a series of connected episode.
Example – Chessboard, Taxi driver

• Static and Dynamic:—
If the environment can change while an agent is deliberating then we say the environment is dynamic for the agent otherwise it is static.

• Single Agent, Multi Agent:—
If the environment contain other intelligent agent i.e, multiagent environment.
Example → Tic-tac-toe

Agent plays chess is a two agent environment
Taxi driving is a multiagent environment.

- Complex environment:—

i) Knowledge Rich:—
Enormous amount of information that the environment contains

ii) Input Rich:—
Enormous amount of input the environment can send to an agent.
Agent must have a way of managing the complexity often such consideration led to the development of—
- sensing strategies
- attentional mechanism

Architecture:—

→ The job of AI is to design an agent program that implements agent function, the mapping from Percept to action.

→ This program will run on some sort of computing device with physical sensor and actuator, we call this the architecture

Agent = Architecture + Program

- Table-driven approach:—

Percept	Action

→ Information comes from sensor, percept.
→ Trigger actions through the effector (Reactive Agent)

- Goal based Agent
- Utility based Agent
- Learning Agent

Goal Based Agent:-

→ Knowledge of current state environment is not always sufficient to decide for an agent to what to do.

→ The agent needs to know its goal which describes desirable situations.

→ They choose an action, so that they can achieve the goal.

→ These agents may have to consider a long sequence of possible actions before deciding whether the goal is achieved or not.

Utility-based Agents:-

→ These agents are similar to the goal-based agent but provide an extra component of utility measurement which makes them different by providing a measures of success at a given state.

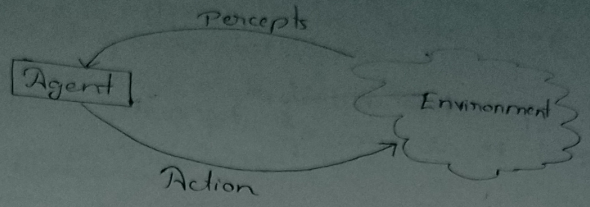
→ Utility-based agent act based not only goals but also the best way to achieve the goals.

→ Utility-based agent is useful when there are multiple possible alternatives, and an agent has to choose in order to perform best action.

→ Utility function maps each state to a real number to check how efficiently each action achieves the goal.

Problem Solving Using Search Algorithm :-

25/01/2024



i) Goal - directed Agent :-

→ A goal directed Agent need to achieve certain goals. Many problems can be represented as a set of state and set of rules of how one state is transferred to another.

→ The agent must choose a sequence of actions to achieve the desired goal

→ Each state is an abstract representation of the agent environment

→ Components :-

- initial state – starting state
- Action / operation
- Transition model / plan – 

- Goal test
- Path Cost

A problem can be defined by five components –

i) Initial state :-
The description of the starting configuration of the agent

ii) Action / operation :-
It takes agent from one state to another state.
A state can have number of successor state

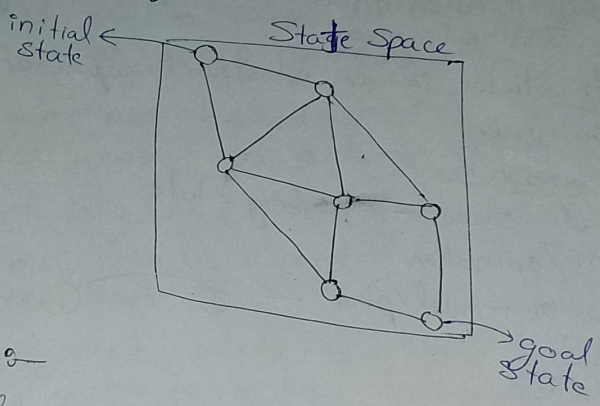
iii) Transition model / plan :-
It is a sequence of actions

iv) Goal Test:-
A goal is a description of a set of desirable state of the world. Goal states are often specified by a goal test which any goal state must satisfy

v) Path Cost:-
Path is always a positive number
Usually, path cost is a sum of step costs

State Space:-

→ Initial state, action and transition model implicity define the state space of the problem
→ The set of all state reachable from the initial state where any sequence of actions



Problem Formulation:-

→ It is a process of deciding what action and state to consider, given a goal.
→ Problem formulation means choosing a relevant set of state to consider, and a feasible set of operators for moving from one state to another.

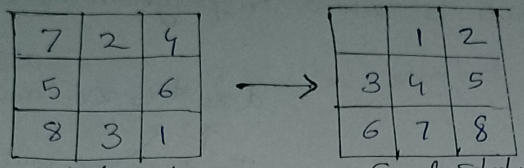
Search:-

→ It is the process of imagining sequence of operators applied to the initial state and checking which sequence ~~which is a goal~~ reaches a goal state.

29/01/2024

S: set of state
S_0: Initial state, $S_0 \in S$
A: $S \rightarrow S$
Set of operator / set of actions
G: Set of final state, $G \subseteq S$

8 Puzzles Problem :-



Initial State Goal State

- State – State description specify the location of each of the eight tiles and the blank in one of 9 square.
- Inital State – Any state can be designated as the initial state.
- Actions – Four actions → up, down, left, right.
- Transition Model – Sequence of action
- Goal Test – Compare each state to the goal state.
- Path Cost –

8 Queen Problem :-

- State – State description specify the location of 8 queens in ~~one of~~ 64 squares.
- Initial State – Empty boards
- Action – Five actions → left, right, up, down, diagonal.
- Transition Model – 8 directions, so that it can slide in all directions.
- Goal Test –

Measuring Problem Solving Performance :—

1. Completeness
2. Optimality
3. Time complexity
4. Space complexity

30/01/2024

Route Finding Problem :—

- Tractable → Problem which is solved in polynomial time.
- Intractable

$d \Rightarrow 4^d \Rightarrow 2d^2$

For $d = 20$ → it will give value in trillion

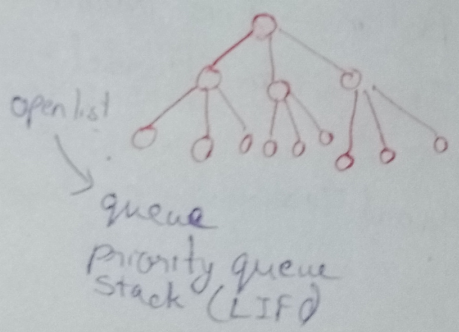
- Frontier :—
  Set of all leaf node available for expansion at any given point ie. called frontier (open list)

(Airline System)

- State :— State description specifies every airport.
- Initial State :— User query (where he/she need to travel)
- Action :— ~~Time constraint~~, location along with time constraint
- Transition Model :—
- Goal Test :— User destination.
- Path Cost :— Path, money, time scheduled.

IW

Graph-Search (implement)



☑ Branching Factor:-
Maximum number of successor of any node.

☑ Depth:- (d)
Number of steps along the path from the route

- m - The maximum length of any path in the state space
- Time:-

Time is often measured in terms of no. of nodes generated during the search.

- Space:-

Maximum no. of nodes stored in memory.

☑ (Assignment) → tic-tac-toe

☑ Basic Search Algorithm:-

Let L be a list containing initial state (L-fringe)

Loop
  if L is empty return failure
  Node ← Select (L)
  if Node is a goal
    then return Node (path from initial state to Node)
  else
    apply all applicable operator to Node,
    and merge the newly generated state into

☑ Key issue:- (of Binary Search)

i) Search tree may be unbounded because of loops
ii) Because state space is infinite.
iii) Return a path or a node.
iv) How are merge and select done?
v) (a) Is the graph weighted or unweighted?
   (b) How much is known about the quality of intermediate s

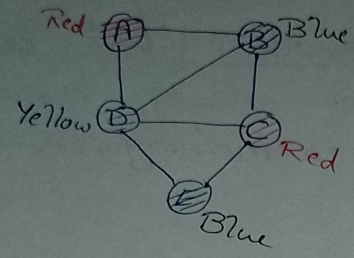
- Heuristic Search

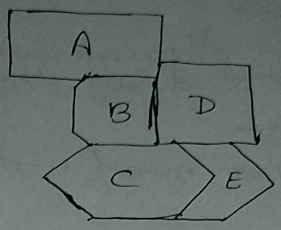
- ~~Heuristic Search~~.

(c) Is the aim to find a minimal cost path or any path as soon as possible.

Uninform Search:— (Blind Search)

→ Graph Coloring Problem:—





(NP hard problem)

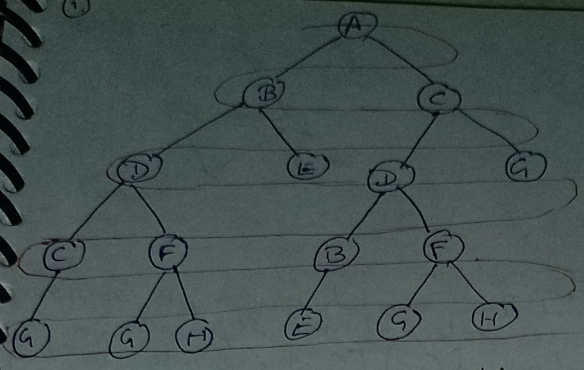
Algorithm not use any information about the domain.
Most important client search strategies are.
- Breadth first search (BFS)
- Depth first Search (DFS)

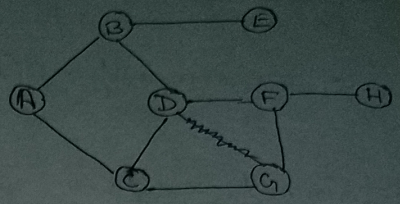
Basic Search Algorithm:—

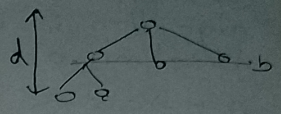
① BFS
② DFS

Let fringe be a list contain the initial state.
Loop
if fringe is empty return failure
Node ← Remove_first (fringe)
if Node is goal
then return the path from initial state to Node
else
generates all successors of node and
merge generated node into fringe

BFS) ↓
Expand shallowest node first and add generate node to the back of fringe







- Time complexity – $O(b^d)$
- Space complexity – $O(bd)$

- ~~DFS~~

2

DFS:–

Expand deepest node first.
Add ~~deepest~~ generated node to the front.

- Time complexity – $O(b^d)$
- Space complexity – $O(b\,m)$ → m – maximum depth, b – no. of branches

Depth limited Search:–

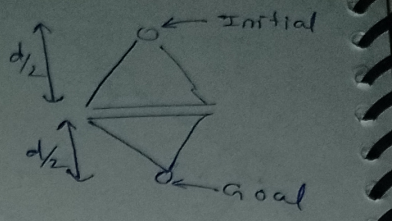
Let fringe be a list contain initial state

Loop
  if fringe is empty return failure
  Node ← Remove-first(fringe)
  if Node is goal
    then return the path from initial state to Node.
  if depth of Node = limited return cutoff

~~Time~~ Space complexity – $O(b \cdot d)$

Bi-directional Search :—

- Time complexity $= O(b^{d/2})$

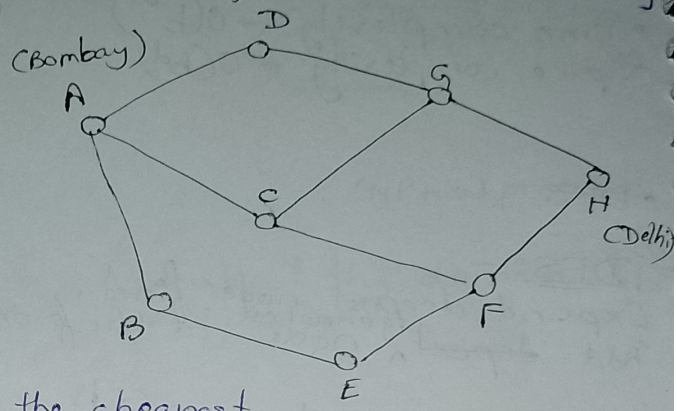


Informed Search :—

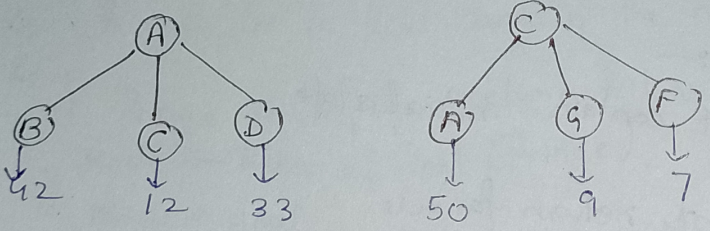
- **Best first Search :—** (Greedy Best First Search)

It is an instance of the general tree search or graph search algorithm in which a node is selected for expansion based on an evaluation function $f(n)$ [Cost Estimation function]

A–H ⇒ cost = 50
B–H ⇒ 42
E–H ⇒ 30
F–H ⇒ 7
C–H ⇒ 12
D–H ⇒ 33
G–H ⇒ 9
H–H ⇒ 0



Heuristic—
$h(n)$ = Estimated cost of the cheapest path from the state at node n to a goal state

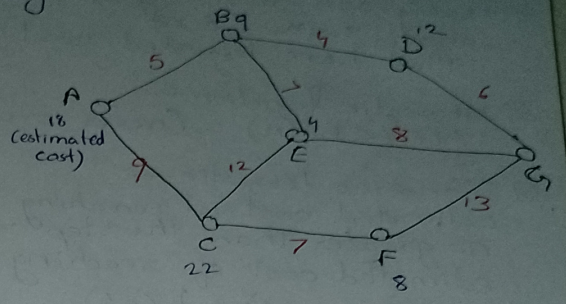


Best first search has given one of best solution but not optimal

- A* algorithm:—
~~It's objective is to find~~
Above solution donot give optimal solution that is why we use A* algorithm.

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

where,
- F(n) estimated cost of cheapest solution or optimal solution through n.
- g(n). is path cost
- h(n) is estimated cost or heuristic function.



A → B
5+9=14

A → C
9+22=31

AB → D
(5+9)+12
=21

AB → E
(5+7)+4
=16

A → B → E → G

☑ A* Algorithm Analysis:—
1. Optimal
2. Completeness.
3. No. of node search is still exponential in the worst case

- If h(n) is admisible then search will find optimal solution if underestimate cost of any solution which can reach from node

- Admissibility of A*:—
Provided solution exist. A first solution found ~~is~~ is an optimal solution

•Condition:—

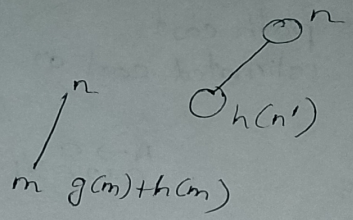
① State space search—
→ Every node has a finite no. of successor
→ Every arc in the graph has a cost greater than sum ($\varepsilon > 0$)
Must have bounded cost

② Heuristic function—
$h(n) < h^*(n)$ (never overestimate)

③ Monotone heuristic—
Estimated cost of reaching the goal from n is no greater than the step cost getting n'* + the estimated cost of reaching the goal from n'

$h(n) \leq c(n, a, n') + h(n')$

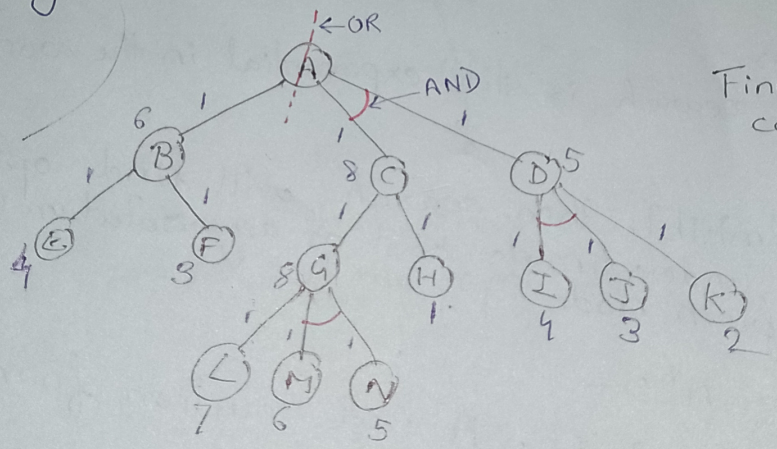
$f(m) = \max(f(n), g(m)+h(m))$



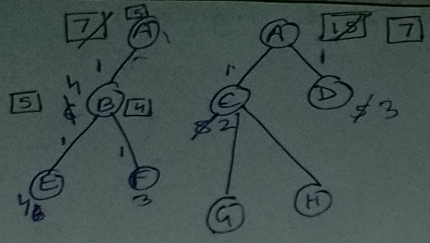
AO* Algorithm:—

① Informed search
② AO* (AND-OR)
③ $f(n) = g(n) + h(n)$
④ All edges default value = 1

Q

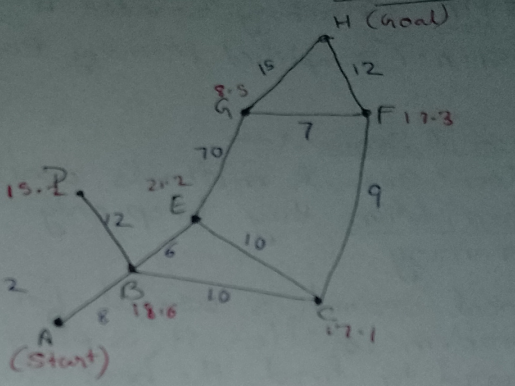


Find Heuristic cost of A



08/02/2024

A* algorithm



A → B
8+18.6 = 26.6

AB → C
(8+10)+17.1 = 35.1

AB → E
(8+6)+21.2 = 35.2

AB → D
(8+12+15.2) = 35.2

ABC → F
(8+10+9)+17.3
= 44.3

ABC → E
(8+10+10)+21.2
= 49.2

ABE → G
(8+6+70)+8.5
= 92.5

ABCF → H
(8+10+9+12)
= 39

Memory Bounded Heuristic —

i) Iterative Deepening A* (IDA*)

- through f-limit
  (If goal state is not reached then we need to increase f-limit)
- prune any node

f(node) > f-limit

☒ RBFS (Recursive Breadth First Search Algorithm)
- Mimic of Best First Search.

☒ MA* (Membory Bounded A*)
☒ SMA* (Simplified Memory Bounded A*)

☒ Local Search:—
→ Local search methods work on complete state formulation they keep only small no. of ~~nc~~ node in memory
→ Local search is useful for solving optimization problem. Often it is easy to find a solution but hard to find the best solution.