LECTURE NOTES
On

Artificial Intelligence & Machine Learning

B. Tech, 6thSemester, CE, EEE, MECH

Prepared by:

Ms. Monali Patel

Asst. Professor

Computer Science & Engineering

Vikash Institute of Technology, Bargarh

(Approved by AICTE, New Delhi & Affiliated to BPUT, Odisha)

Barahaguda Canal Chowk, Bargarh, Odisha-768040

www.vitbargarh.ac.in

DISCLAIMER

- This document does not claim any originality and cannot be used as a substitute for prescribed textbooks.
- The information presented here is merely a collection by Monali Patel with the inputs of students for their respective teaching assignments as an additional tool for the teaching- learning process.
- Various sources as mentioned at the reference of the document as well as freely available materials from internet were consulted for preparing this document.
- Further, this document is not intended to be used for commercial purpose and the authors are not accountable for any issues, legal or otherwise, arising out of use of this document.
- The author makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and specifically disclaims any implied warranties of merchantability or fitness for a particular purpose.

****

COURSE CONTENT

B. Tech, 6thSemester, CE, EEE, MECH

Module-I: (12 hours) INTRODUCTION –The Foundations of Artificial Intelligence; - INTELLIGENT AGENTS – Agents and Environments, Good Behavior: The Concept of Rationality, the Nature of Environments, the Structure of Agents, SOLVING PROBLEMS BY SEARCH – Problem-Solving Agents, Formulating problems, Searching for Solutions, Uninformed Search Strategies, Breadth-first search, Depth-first search, Searching with Partial Information, Informed (Heuristic) Search Strategies, Greedy best-first search, A* Search, CSP, Means-End-Analysis.

Module-II: (12 hours) ADVERSARIAL SEARCH – Games, The Mini-Max algorithm, optimal decisions in multiplayer games, Alpha-Beta Pruning, Evaluation functions, Cutting off search, LOGICAL AGENTS – Knowledge-Based agents, Logic, Propositional Logic, Reasoning Patterns in Propositional Logic, Resolution, Forward and Backward chaining - FIRST ORDER LOGIC – Syntax and Semantics of First-Order Logic, Using First-Order Logic , Knowledge Engineering in First-Order Logic - INFERENCE IN FIRST ORDER LOGIC – Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution

Module-III: (6 hours) UNCERTAINTY – Acting under Uncertainty, Basic Probability Notation, The Axioms of Probability, Inference Using Full Joint Distributions, Independence, Bayes' Rule and its Use, PROBABILISTIC REASONING – Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distribution, Exact Inference in Bayesian Networks, Approximate Inference in Bayesian Networks

Module-IV: (10 hours) LEARNING METHODS – Statistical Learning, Learning with Complete Data, Learning with Hidden Variables, Rote Learning, Learning by Taking Advice, Learning in Problem-solving, learning from Examples: Induction, Explanation-based Learning, Discovery, Analogy, Formal Learning Theory, Neural Net Learning and Genetic Learning. Expert Systems: Representing and Using Domain Knowledge, Expert System Shells, Explanation, Knowledge Acquisition.

****

REFERENCES

Artificial Intelligence & Machine Learning

B.Tech, 6thSemester, CE, EEE, MECH

Books:

[1] Elaine Rich, Kevin Knight, & Shiva Shankar B Nair, Artificial Intelligence, McGraw Hill,3rd ed.,2009

[2] Introduction to Artificial Intelligence & Expert Systems, Dan W Patterson, PHI.,2010

[3] S Kaushik, Artificial Intelligence, Cengage Learning, 1st ed.2011

[4] Stuart Russell, Peter Norvig, *Artificial Intelligence -A Modern Approach*, 2/e, Pearson, 2003.

Digital Learning Resources: Course Name:	Artificial Intelligence Search Methods For Problem Solving
Course Link:	https://swayam.gov.in/nd1_noc20_cs81/preview
Course Instructor:	Prof. D. Khemani, IIT Madras

****

Friday • September 2014

05 Week 36 Day (248-117)

Artificial Intelligence

Man Made → thinking Power

→ System having thinking Power.

(or) ⇒ Fetch intelligence to computer System artificially by using algo., pseudo code etc.

(or)

⇒ AI is a branch of computer Science by which we can create intelligent machine which can behave like a human & think like human & able to make decision.

⇒ Phone is Preprogram (receiving call or not) but AI do not need to Preprogram because we create intelligent machine).

→ We have to create a machine with program algo. which can work with own intelligence

Why AI ?

1) We can create Such S/W or devices which Solve real world Problem easily & accurately (health issues, marketing, traffic issue) etc.
2) You can create personal virtual Assistant)
3) Build robots which can work in a environment where Survival of human beat Risk

Goals of AI

1) Replicate human intelligence (machine work like human)

NOTES

SEPTEMBER 2014

Wk	M	T	W	T	F	S	S
36	1	2	3	4	5	6	7
37	8	9	10	11	12	13	14
38	15	16	17	18	19	20	21
39	22	23	24	25	26	27	28
40	29	30					

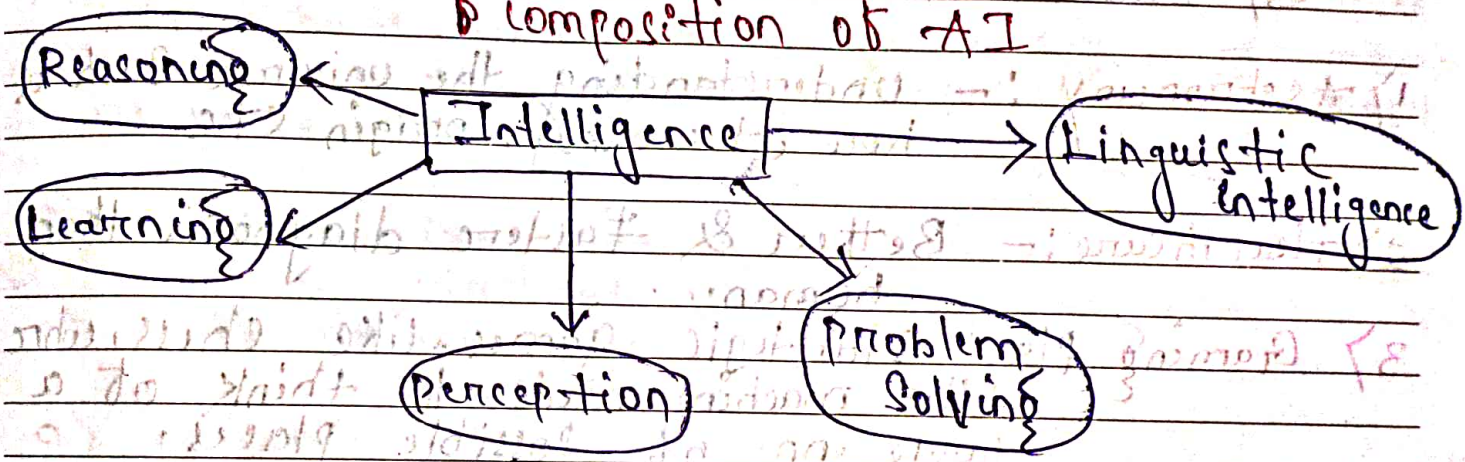
2014

Saturday • September

Week 36 (249-116) 06

2) Solve Knowledge-intensive tasks
3) connection of perception & action
4) Build machine which perform human intelligence.
   → providing thm
   → playing chess
   → Surgical operation

Composition of AI



Reasoning :⇒ Judgement, making dicision & prediction
Learning :⇒ Gaining knowledge
Perception ⇒ acquiring / interpreting, selecting or organizing sensore info. (Sensory organ)
Problem Solving :⇒ working through details of problem to reach soln
Linguistic intelligence :- comprehend, speak, write verbal, written language.

Sunday 07

All the above-mentioned examples define the impact of artificial intelligence on today's life and show how intelligence can be developed artificially. Artificial Intelligence is composed of two words "Artificial" and "Intelligence" where Artificial defines "Man-made" and Intelligence defines *"Thinking Power"*, hence AI means ***"a man-made thinking power".***

"It is a branch of computer science by which we can create intelligent machines which can behave like humans, think like humans, and be able to make decisions like humans."

AI can also be defined as:

(a) A computer system that attempts to model and apply intelligence the human mind.

1

(b) A branch of computer science that deals with the simulation of intelligent behaviour in computers.

(c) The capability of a machine to imitate human behaviour.

To create AI, first, we should know how intelligence is composed, as intelligence is an intangible part of our brain which is a combination of reasoning, learning, problem-solving, perception, language understanding, etc.

To achieve the above factors for a machine or software Artificial Intelligence requires all the following disciplines:

- Mathematics
- Biology
- Psychology
- Sociology
- Computer Science
- Neurology
- Statistics

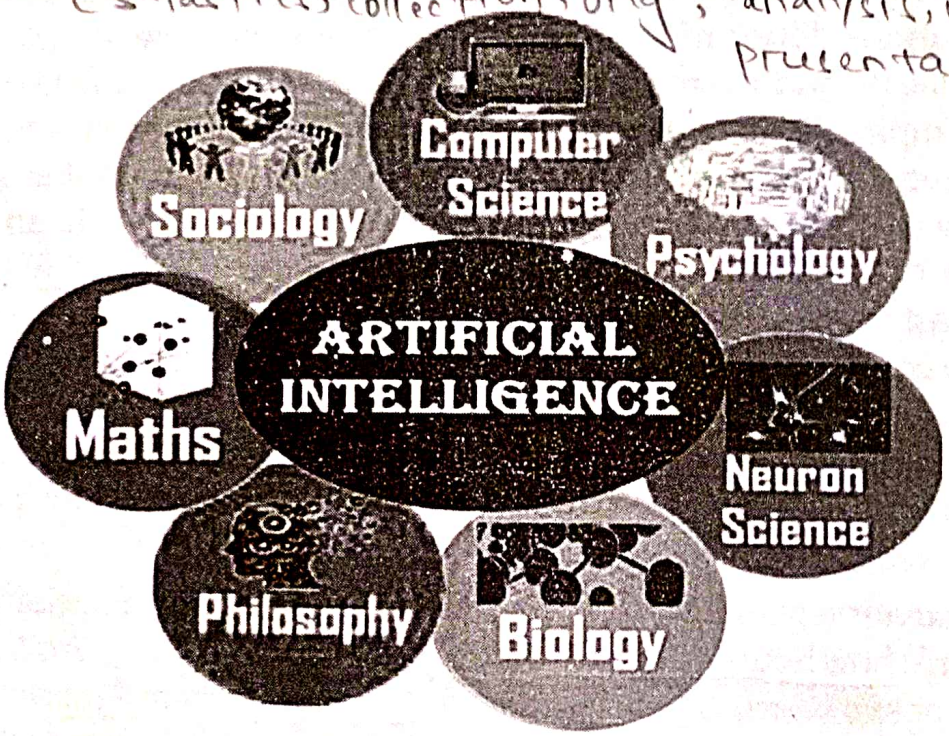


Fig. 1.1. Various Disciplines that constitute artificial intelligence

These are the disciplines that are to be focused on while creating artificial intelligence in the following ways:

(a) We have different types of models for which basic mathematics like linear algebra, linear equations, concepts of coordinate geometry like distance formula, in statistics, probability, correlations, mean, median, mode, etc. are to be focused on, as the field of data science completely depends on the analysis of collected data.

(b) Basic knowledge of computer science, instructions are to be given to implement a model for which, a suitable programming language should be known.

(c) Just as signals in the human brain are processed using neurons, to develop artificial intelligence in computers, an artificial neural network has to be created.

(d) A human response depends on emotions and surroundings. So, to develop a humanoid it is required that it should have a sense of emotions and adaptability.

** Humanoid is an artificially intelligent human, for example, Sophia, the robot.

Examples of AI:

- *Google Assistant*
- *Alexa, Siri, Cortana*
- *Chatbots, for example, Endurance, Casper and Disney*
- *Face detection and recognition*
- *Recommender Systems, like suggestions on YouTube, Netflix and Amazon*

All these examples will be discussed in detail in later chapters. Let us first discuss when and how this term came into existence.

1.1. THE EVOLUTION OF AI TO THE PRESENT

Artificial Intelligence is not a new term and not a new technology for researchers. This technology is much older than you would imagine. There are references of Mechanical men in Greek and Egyptian mythology. Following are some milestones in the history of modern AI which define the journey from its generation to development till date.

John McCarthy, Alan Turing, Marvin Minsky, Allen Newell, and Herbert A. Simon are named as the "founding fathers" of artificial intelligence.

1.1.1. Maturation of Artificial Intelligence (1943-1952)

- **Year 1943:** The first work which is now recognized as AI was done by Warren McCulloch and Walter Pits in 1943. They proposed a model of artificial neurons.
- **Year 1949:** Donald Hebb demonstrated an updating rule for modifying the connection strength between neurons. His rule is now called Hebbian learning.
- **Year 1950:** Alan Turing was an English mathematician who pioneered Machine learning in 1950. Alan Turing published "Computing Machinery and Intelligence" in which he proposed a test that could check the machine's ability to exhibit intelligent behaviour equivalent to human intelligence. It was called Turing test.

1.1.2. Birth of Artificial Intelligence (1952-1956)

- **Year 1955:** Allen Newell and Herbert A. Simon created the "first artificial intelligence program" which was named "Logic Theorist". This program proved

38 of 52 Mathematics theorems and found new and more elegant proofs for some theorems.

- **Year 1956:** The term "Artificial Intelligence" was first coined by an American computer scientist John McCarthy at the Dartmouth Conference. For the first time, AI was called as an academic field.

That was the time when high-level computer languages such as FORTRAN, LISP and COBOL were invented. The enthusiasm for AI was very high at that time.

1.1.3. Golden years—Early enthusiasm (1956-1974)

- **Year 1966:** The researchers emphasized on developing algorithms that could solve mathematical problems. Joseph Weizenbaum created the first chatbot in 1966, which was named ELIZA.
- **Year 1972:** The first intelligent humanoid robot was built in Japan. It was named WABOT-1.

1.1.4. First AI Winter (1974-1980)

The duration between the years 1974 to 1980 was the first AI winter. AI winter refers to the period where computer scientists dealt with a sevre shortage of funding from the government for AI research.

During AI winters, an interest in publicity on artificial intelligence was decreased.

1.1.5. Boom Time for AI (1980-1987)

- **Year 1980:** After the AI winter, AI came back with an "Expert System". Expert systems were programs that emulate the decision-making ability of a human expert. In 1980, the first national conference of the American Association of Artificial Intelligence was held at Stanford University.

1.1.6. Second AI Winter (1987-1993)

The duration between the years 1987 to 1993 was the second AI Winter. Investors and the government stopped funding for AI research due to high costs but no concrete results coming in. The expert system such as XCON was very cost-effective.

1.1.7. Emergence of Intelligent Agents (1993-2011)

- **Year 1997:** In the year 1997, IBM Deep Blue beat world chess champion Gary Kasparov, and became the first computer to beat a world chess champion.
- **Year 2002:** For the first time, AI entered homes in the form of Roomba, a vacuum cleaner.
- **Year 2006:** AI came into the Business world in the year 2006. Companies like Facebook, Twitter, and Netflix also started using AI.

1.1.8. Deep Learning, Big Data and Artificial General Intelligence (2011-Present)

- **Year 2011:** In the year 2011, IBM's supercomputer Watson won Jeopardy, a quiz show, where it had to solve complex questions as well as riddles. Watson had proved that it could understand natural language and can solve tricky questions quickly.
- **Year 2012:** Google launched an Android app feature "Google now", which was able to provide information to the user as a prediction.
- **Year 2014:** In the year 2014, Chatbot "Eugene Goostman" won a competition in the infamous "Turing test."
- **Year 2018:** The "Project Debater", the first AI system from IBM debated on complex topics with two master debaters and performed extremely well.

Google demonstrated an AI program "Duplex" which was a virtual assistant and which gave hairdresser appointments on telephone call, and the lady on the other side didn't notice that she was talking with a machine.

1.2. VARIOUS APPROACHES TO AI → Science + Engineering

Artificial intelligence is both science and engineering. It is the science of understanding intelligent entities—of developing theories that attempt to explain and predict the nature of such entities, and it is the engineering of intelligent entities.

There are four main views of AI in the scientific literature as listed below. –

(a) AI means thinking humanly, i.e., thinking like a person. The field of Cognitive Science delves into this topic, trying to model how humans think.

(b) AI means acting humanly, i.e., acting like a person. The classic example of this is the "Turing test".

(c) AI means thinking rationally, i.e., modelling thinking as a logical process, where conclusions are drawn based on some type of symbolic logic.

(d) AI means acting rationally, i.e., performing actions that increase the value of the state of the agent or environment in which the agent is acting. For example, an agent that is playing a game will act rationally if it tries to win the game.

Let us consider a real-life situation to understand the four views of AI better. Suppose you ask a person to bring a glass of water to you. According to the four views, it is elaborated below:

(a) The other person can grasp your command and understand that he/she has to bring water. This is Thinking Humanly.

(b) The person grasps the command and brings you water whether in a glass or a bowl. This is Acting Humanly.

(c) The person after knowing the command can picture detailed possibilities of

outcomes, like, the path they have to follow, the amount of water that will be sufficient, selecting a proper container for water, etc. This is Thinking Rationally.

(d) The person does what detailed possibilities he thought rationally is an example of acting rationally.

***The difference between "acting humanly" and "thinking humanly" is that the first is only concerned with the actions, the outcome, or product of the human's thinking process; whereas the latter is concerned with modeling human thinking processes.*

Appl^n of AI :⇒

1) Astronomy
2) Healthcare
3) Gaming
4) Finance
5) Data Security
6) Social media

NOTES

OCTOBER 201

Wk	M	T	W	T	F	S
40			1	2	3	4
41	6	7	8	9	10	11
42	13	14	15	16	17	18
43	20	21	22	23	24	25
44	27	28	29	30	31	

4) Travel & transport
8) Automotive industry
9) Robotics
10) Entertainment
11) Agriculture
12) E-commerce
13) Education

1) Astronomy :- Understanding the universe such as how it works, origin etc

2) Healthcare :- Better & faster diagnosis than human.

3) Gaming & play strategic games like chess, where the machine needs to think of a large no. of possible places.

4) Finance :- Automation, chartbot, adaptive intelligence, algo. trading

5) Data Security :- Make data more secured safe eg :- AEG bot, AI2 platform are used to determine SIVI bug & cyberattacks in a better way

6) Social Media :- fb, twitter, Snapchat. AI can organize & manage massive amt. of data.
(AI can analyze lots of data

NOTES

SEPTEMBER 2014

Wk	M	T	W	T	F	S	S
36	1	2	3	4	5	6	7
37	8	9	10	11	12	13	14
38	15	16	17	18	19	20	21
39	22	23	24	25	26	27	28
40	29	30					

to identify the latest trends & requirement users.

4) Travel & Transport :→
AI is capable of doing various travel related works such as travel arrangement to suggesting the hotels, flights & best routes to the customers. (AI-powered chatbots which can make human like interaction with customers for better & fast response)

8) Automotive Industry
virtual assistance, (Tesla bot).
→ Self-driven car

9) Robotics Humanoid robot (Eg:- Erica, Sophia

10) Entertainment Netflix, Amazon (recommendation)

11) Agriculture:— Agriculture robotics, Solid & crop monitoring, Predictive analysis

12) E-commerce :— AI helping Shoppers to discover associated products with recommended size, color or brand.

13) Education AI chatbot can communicate with students as a teaching assistant.

NOTES

① Intelligent Agent :⇒

↳ machine / Robots / anything depending on the situation.

Intelligent :⇒

dictionary def^n

i) Someone's intelligence is their ability to understand & learn things.

ii) Intelligence agent is the ability to think & understand instead of doing things by instinct or automatically.

Intelligent Agent :-

In AI intelligent agent that perceives / sense environment (what you see) & act upon that environment through actuators (through which we perform action).

→ OR

→ In AI, intelligent agent is anything which perceives its environment, takes actions autonomously in order to achieve goals, & may improve its performance with learning or may use knowledge.

→ An agent runs in the cycle of perceiving (notice / realize something / think of something in a particular way), thinking & action.

OR It is an entity that make decision & enable AI to be put into action.

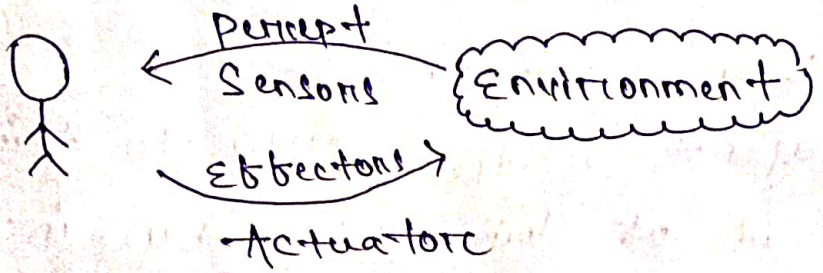
Important terminology

Sensors : Device which detects the change in the environment & sends info. to other device.

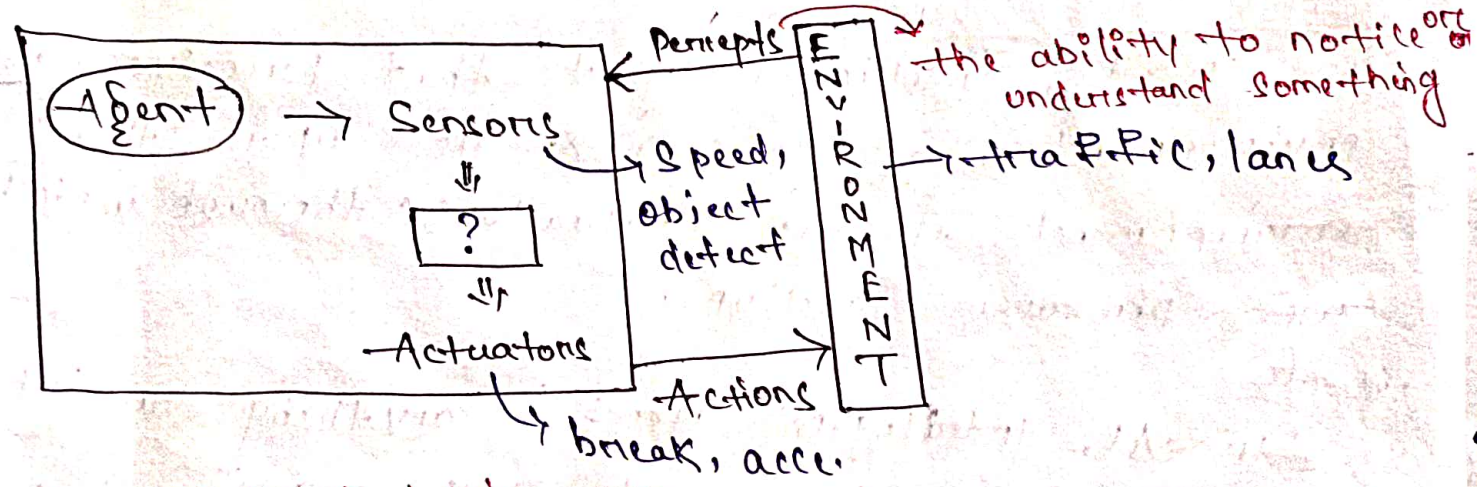
(2) (An agent observe its environment through sensor.)

Actuators:- component of machine that converts energy into motion. It is only responsible for moving & controlling machine

↳ Motor, Gears etc.



Agent → Perception → Decision → Actions.



(Self driving car)

Types of Agent

a) Human Agent

Sensor - eye, ears & other organs

Actuator - Hand, Legs, Vocal tract.

b) Robotic Agent

Sensor - NLP, camera, IRFC (Infrared range finder)

↳ ability to understand text & spoken words like human

↳ calculate the distance of object

(3) c) Software Agent

Key strokes, file contents as sensory i/p & act on those i/p & display o/p on the screen.

Good Behaviour/ The concept of Rationality :-
↳ work as per the desired actions.

→ whatever AI agent we are developing we want that agent work as per desired action.

Rationality depends on
a) Performance measures
b) Agent Prior Knowledge (Environment)
c) Actions that agent can perform (Actuators)
d) Agent's Percept Sequence (Sensors)
↳ History of everything agent has ever perceived.

(PEAS)

Rationality :⇒

→ State of being reasonable, sensible, & having good sens of judgment.

→ conformity of one's belief with one's reason to believe or one's action with one's reasons for action.

Rational Agent

→ An agent which has clear Preferences & model uncertainty via expected values.

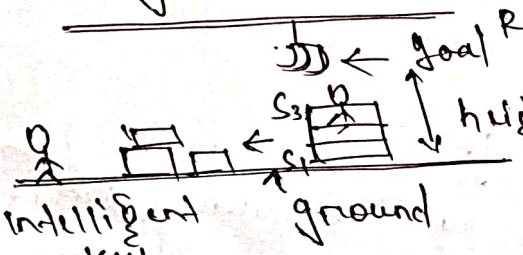
→ It can be anything that make decision. typically a Person, firm, machine, or s/w.

⇒ Rational agent always performs right action where the right action means ~~the action with which the~~ that causes the agent to be most successful

→ RA is capable of taking best possible action in any situation.

Problem Solving in AI / State space Search (How humans are solving Problem)

Eg :- Monkey - Banana (Prolog) ← logical Problem



⇒ Problem Solving is agent performs precisely by defining Problem & several Sol's.

Steps Problem - Solving in AI

The Problem of AI is directly associated with the nature of humans & their activities. So we need no. of finite steps to solve a problem -

a) Goal Formation :-

1st & simple step in problem solving.

→ It organizes finite steps to formulate a target/ goals which requires some action to achieve the goal.

b) Problem formulation :-

It is one of the core steps of problem solving which decide what action should be taken to achieve the formulated goal.

→ components to formulate the associated Problem.

i) Initial State :

This State requires an initial state for the problem which starts the AI agent towards a specific goal.

ii) Action :-

This stage of problem formulation works with fun with a specific Klass taken from the initial state & all possible actions done in this stage.

iii) Transition :-

This stage of problem formulations integrates the actual action done by the previous action stage. &

iv) Goal test :- This stage determines that the specified goal achieved by the integrated

(Sensors)

PEAS → Task Environment

(Intelligence power)

transition model or not.

Path costing :-

This component of problem solving numerical assigned what will be the cost to achieve the goal.

General way to represent any Problem in AI using State space Approach :-

1) Define state space that contain all possible configuration of relevent objects
2) Define some initial state & some goal state.
3) Specify rules as possible actions.
4) Good control strategy → causes motion
→ Systematic step.

Learning Agent Architecture :-

→ It allows agents to operate in initially unknown environment & to become more competent than its initial knowledge alone might allow.

→ The learning element is responsible for making improvement.

→ The performance element is responsible for making selecting external actions. The performance element takes in percepts & decide on actions.

→ The learning element uses feedback from the critic on how agent is doing & determine how Performance element

transition model or not.

Path costing :-

This component of problem solving numerical assigned what will be the cost to achieve the goal.

General way to represent any problem in AI using State space Approach :-

1) Define state space that contain all possible configuration of relevent objects
2) Define some initial state & some goal state.
3) Specify rules as possible actions.
4) Good control strategy → causes motion
   → Systematic strategy.

Learning Agent Architecture :-

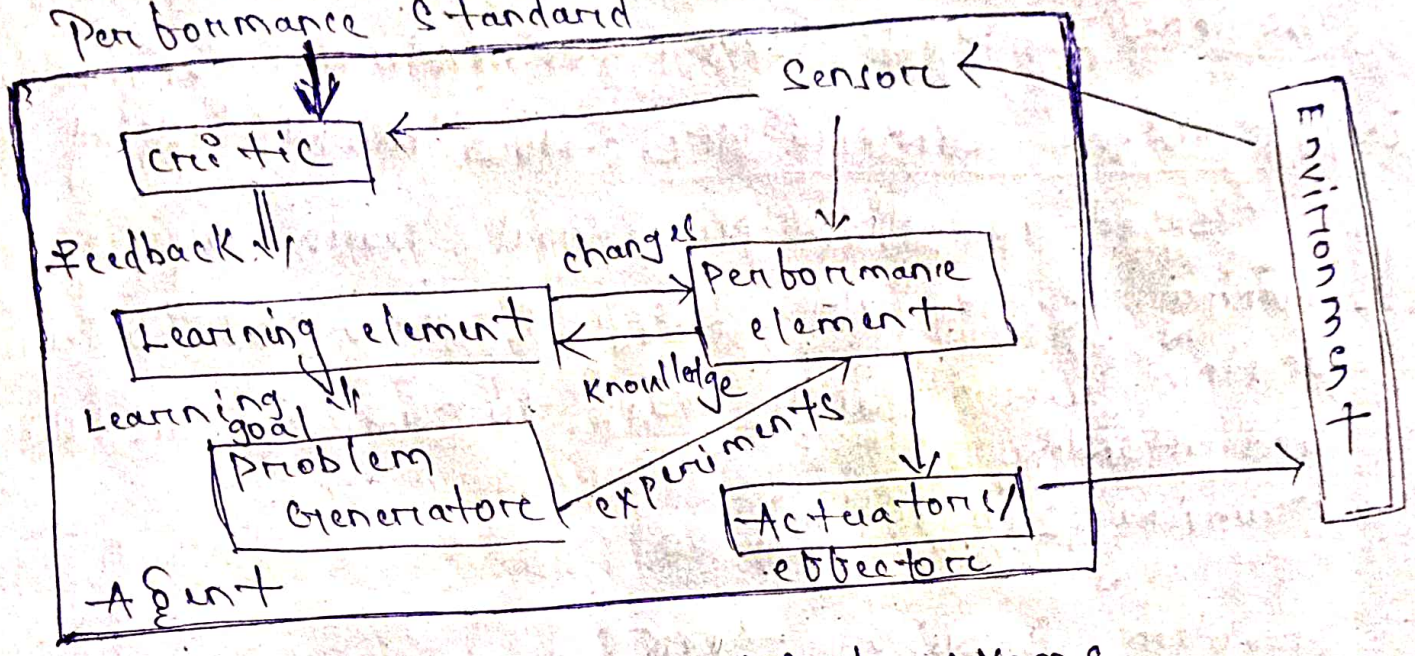
→ It allows agents to operate in initially unknown environment & to become more competent than its initial knowledge alone might allow.

→ The learning element is responsible for making improvement.

→ The performance element is responsible for making selecting external actions. The performance element takes in percepts & decide on actions.

→ The learning element uses feedback from the critic on how agent is doing & determine how performance element should be modified to be better in future.

→ The last component of the learning problem agent is the Problem generator. It is responsible for Suggesting actions that will lead to new informative experiences.



Eg :- class test before final exams
↳ find out our & mistakes.
(learn from its own mistakes)

i) Critics :→ (who tells us our mistake)
→ It is the one who compares Sensors i/p specifying effects of agent's action on the environment with performance Standards & generate feedback for learning team element. ↳ expectation

ii) Learning element :- (Heart of learning Agent)
→ This component is responsible to learn from the difference bet' Performance Standard & the feedback from the critics.
→ According to current percept it is suppose to understand the expected behaviours & enhance its Standards

iii) Performance Element (who Perform action ?)

→ Based on the current Percept received from Sensors & the input obtained by the LE.

→ Performance element is responsible to choose the actions to act upon the external environment.

iv) Problem Generator (How multitasking u are)

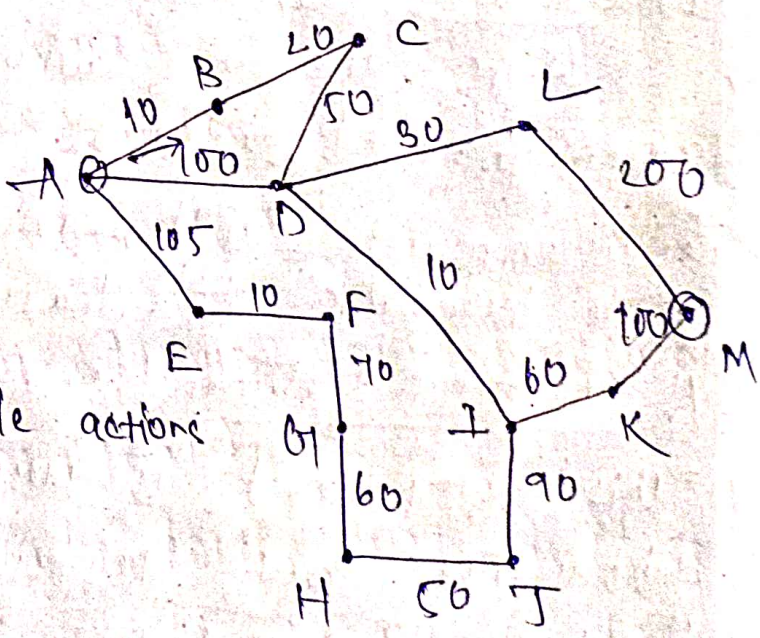
Based on the new Goal learn by the learning agent, PG suggest new or alternative action which will leads to new & instructive understanding.

→ (Proving Suggestion)

Problem Solving Agent

Problem Solving agent in AI is goal based agents that focus on goals.

→ The agents Perform to arrive at a desired state or goal are.



i) Initial State

The initial state A.

ii) Actions

From A the applicable actions are

{Go(B), E, D)}

iii) Succession fun?

Result (In(A), Go(B)) = In(B)

iii) Goal state
The agent goal state in M

iv) Path cost
Distance bet^n 2 transition is the path cost

→ In problem solving agents AI mostly used search startegies or algorithm to solve a specific problem & provide the best result.

Search Algo. & Terminology

a) Search : Step by step procedure to solve a search problem in a given search space
i) Search space - set of possible sol^n, which a system have.
ii) Start state - where Agent begins the search
iii) Goal test - Returns whether the goal state is achieve or not.

~~Types of Search Algo.~~

b) Search Tree : A Tree representation of search problem.

c) Actions d) Transition model

e) Path cost f) Sol^n g) optimal sol^n

Properties of Search Algorithm

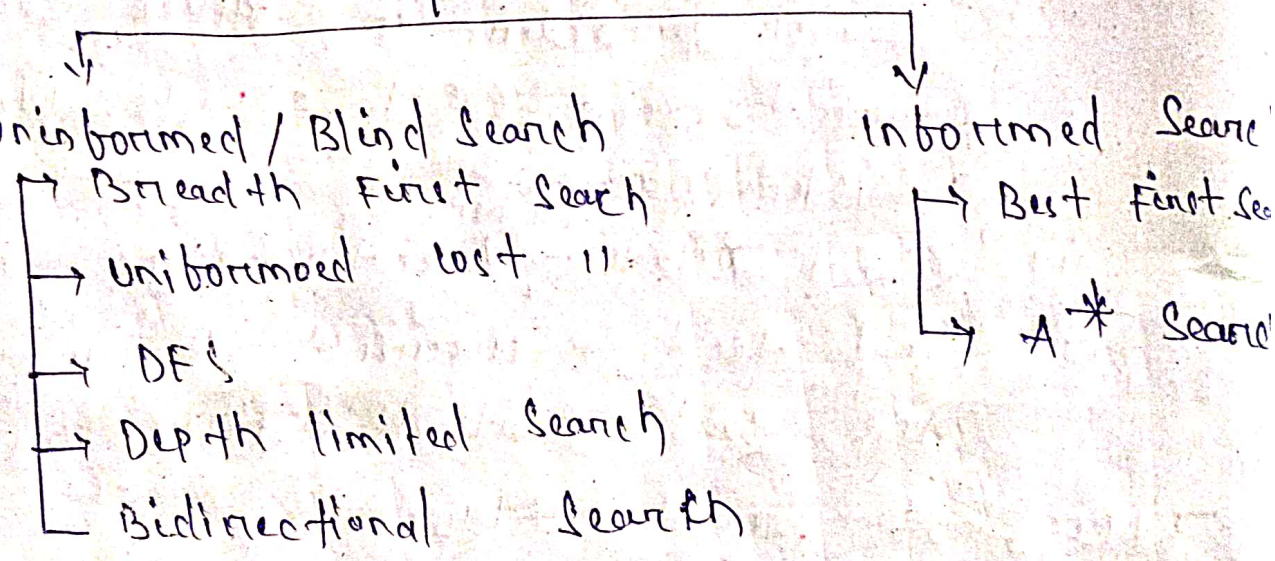
a) completeness = If it gurantees to return a sol^n at least one)

b) optimality :-
The sol^n found for an algo. is guaranteed to be best (lowest path cost)

c) Time complexity
Time complexity is a measure of time for an algo. to complete its task

d) Space complexity
Max^m storage space required at any point during the search

Types of Search Algo.



Uniformed Search / Blind Search

→ It does not contain any domain knowledge (such as closeness, location of the goal).
→ It operates in a brute force way as it only include in so. about how to traverse the tree & how to identify leaf & goal node.

In short

→ Searching without information
→ No information
→ Time consuming
→ More complexity

Breadth-First Search

→ BFS is the most common search strategy for traversing a tree or graph.
→ This algo. searches breadthwise in a tree or graph so, it is called BFS
→ BFS algo. starts searching from the root node of the tree & expands all successors node at the current level by moving to nodes to next level
→ Implemented using FIFO queue datastructure.
↑ The element inserted at first in the list, is the first element to be removed.

BFS Algo. :-

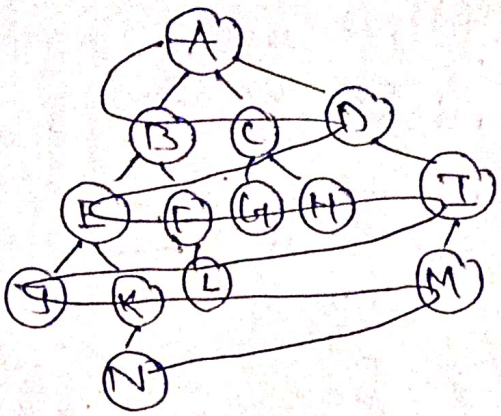
1) Start from Root Node & start traversing the data from it.
2) BFS traverse all the node in the Graph/tree & keep dropping them as completed.
3) BFS visits an adjacent/child unvisited node, & insert it in queue.
4) Remove previous vertex if no adjacent/child node is there.
5) Iterate until all the vertices in the graph/tree marked as complete.

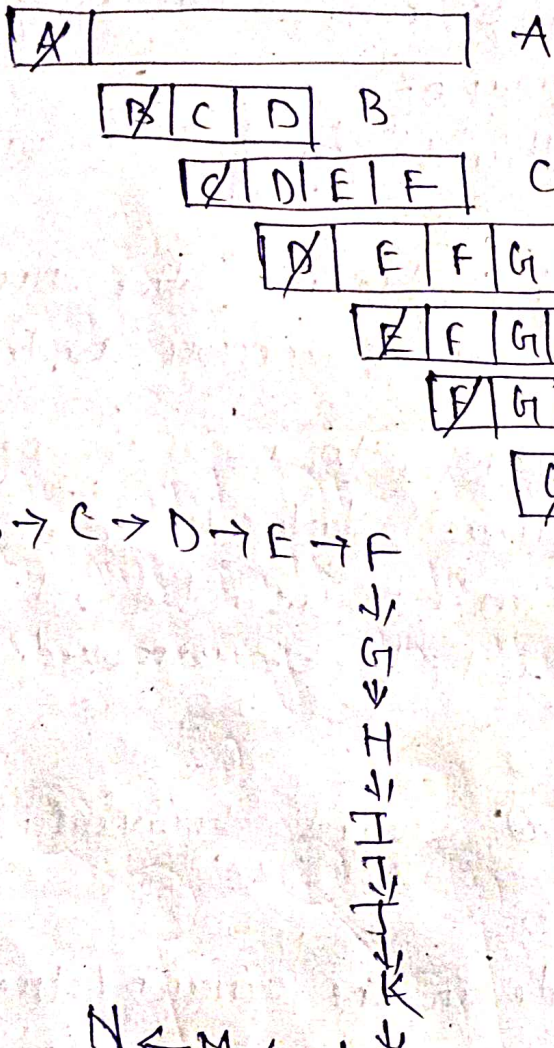
PEAS ⇒ Task

$1 + b^3 + b^5 + b \cdots$

$O(b^d)$

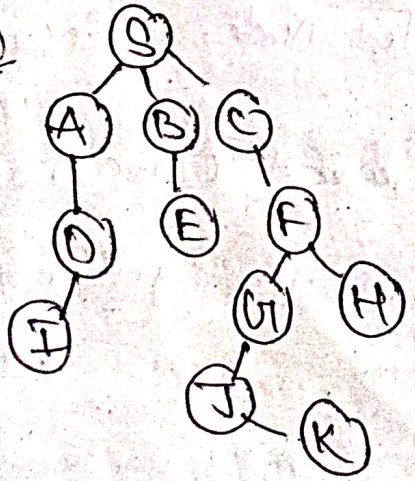
Q9





A → B → C → D → E → F → G → H → I → J → K → L → M → N

Q9



Find the route from S to K using BFS

obj-detect

Advantages

→ BFS will provide a sol^n if any sol^n exists.
→ If there are more than one sol^n for a given problem, BFS will provide the minimal sol^n which requires the least no. of steps.

Disadvantages.

→ Lots of memory required
→ BFS Time consuming if the sol^n is far away from the root node

~~Depth first Search~~

Q)

Depth First Search

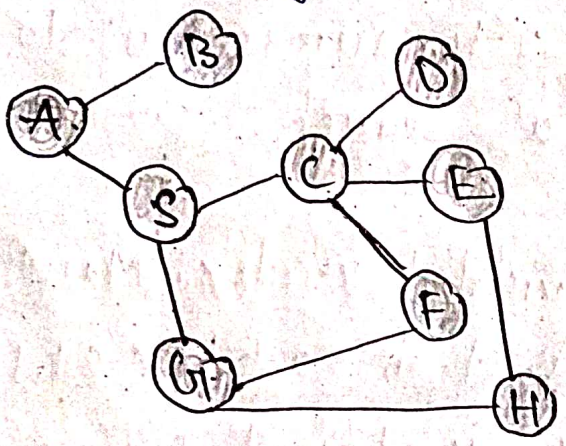
→ DFS algo. traverses a graph in a depthward motion & uses a stack to remember to get the next vertex to start a search.

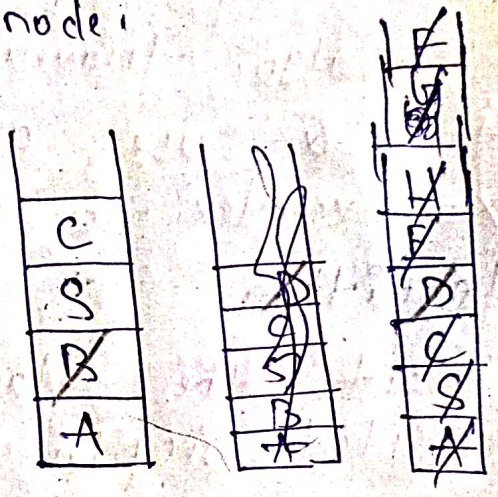
Algorithm

1) visits the adjacent unvisited vertex. Mark it as visited. Display it. Push it in a stack (LIFO)
2) If no adjacent vertex is found, pop up a vertex from the stack (it will pop up all the vertices from the stack, which do not have adjacent vertices)
3) Repeate 1 & 2 step-1 & step-2 until the stack is empty

→ It is called DFS because it starts from the root & follows path to its greatest depth node before moving to the next node.

eg :-





O/P : A B S C D E H G F

Advantages

→ It requires very less memory

→ It takes less time to reach the goal node.

Disadvantage:

→ There is possibility that many state keeps reoccurring & there is no guarantee of finding the sol^n.

→ DFS algo. goes for deep down searching & sometime it may go to the infinite loop.

Informed Search

→ It uses domain knowledge the problem info. is available which can the search.

NOTES

OCTOBER 2014

Wednesday • September

24 Week 39 Day (267-098)

→ Informed Search Strategies can find a soln more efficient than uninformed

→ Informed Search is also called as Heuristic Search.

(A way which might not always be guaranteed for best optimal soln but guaranteed to find a good soln in reasonable time.

→ It can solve much complex Problem which could not be solved in another way

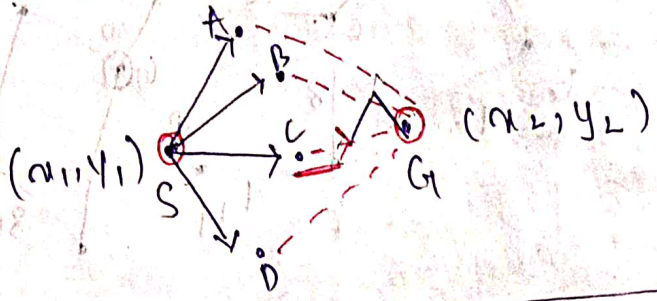
Heuristic in AI

↳ quick guess / soln (eg:- Let value = x)

→ It is a technique designed to solve a problem quickly. (good soln but we may / may not get optimal soln)

→

eg:- Euclidean distance (s.t. line distance)



$$Ed = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

* min^m path

Manhatten distance

1	3	2
6	5	4
	8	7

→

1	2	3
4	5	6
7	8	

a) 0 + 1 + 2 + 2 + 0 + 2 + 2 + 0 ...

b) Misplace tile

Why

→ we want quick soln

Best First Search (informed Heuristic)

Algorithm

→ Let 'OPEN' be a priority (ult.) queue containing initial state

Loop

if open is empty return failure

Node ← Remove-first (open)

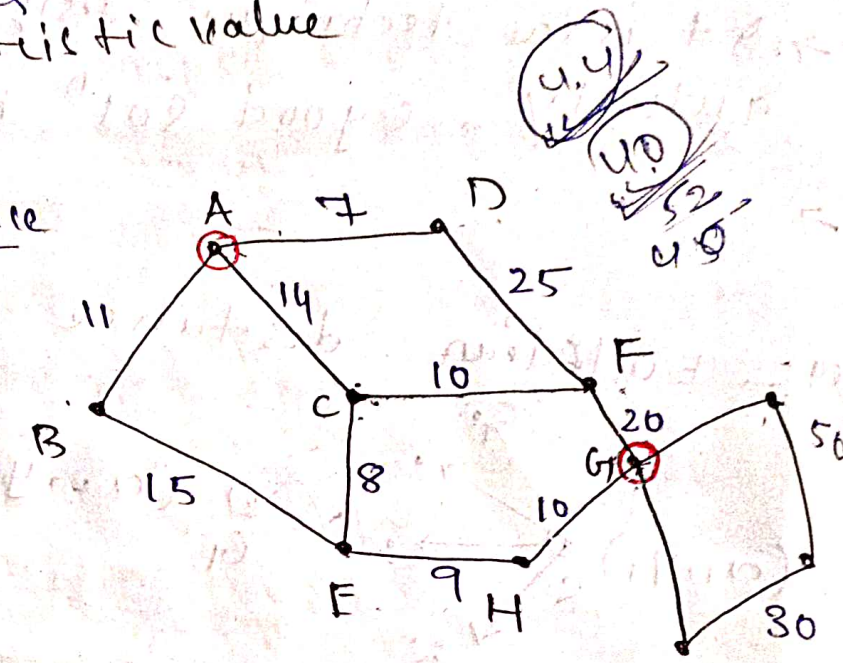
else

Generate all successors of Node & put newly generated Node into OPEN according to their f values
↳ Heuristic value

END Loop

Straight Line distance

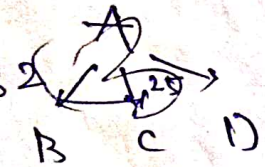
Eucledian distance

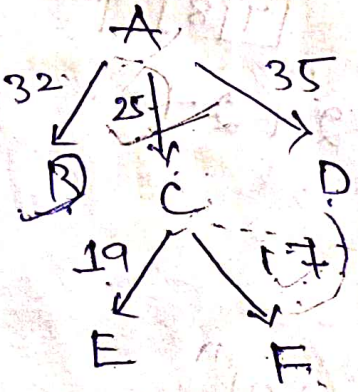


A → G = 40
B → G = 32
C → G = 25
D → G = 35
E → G = 19
F → G = 17
H → G = 10
G → G = 10

(Less heuristic value is better)

↳ Heuristic value





f(n) = h(n)

A, C, B, D

A closed List)

F E

14 – 16 – 20

A → C → F → G

Algorithm for A*

Step-1 Place the starting node in the 'OPEN' list.

Step-2 check if the OPEN list is empty or not. if the list is empty then return failure & stops.

Step-3 Select the node from the open list which has the smallest value of evaluation fun^n (q th), if node n is goal then return success & stops, otherwise

Step-4 Expand node n & generate all of its successor & put n into the closed list. For each successor 'n' check whether 'n' is already in the open or closed list. if not then compute the evaluation.

Step-5 Else if node n is already in open & closed, then it should be attached to the back pointer. which reflects the lowest g(n) value.

Step-6 return Step-2

NOTES

A* - Informed Searching

↳ Best heuristic value + optimal sol^n

→ We already know about the problem / problem statement / domain. (have knowledge)

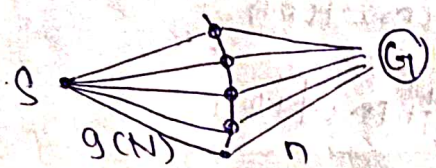
Admissable (Acceptable)
⇓
100% we get optimal sol^n

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

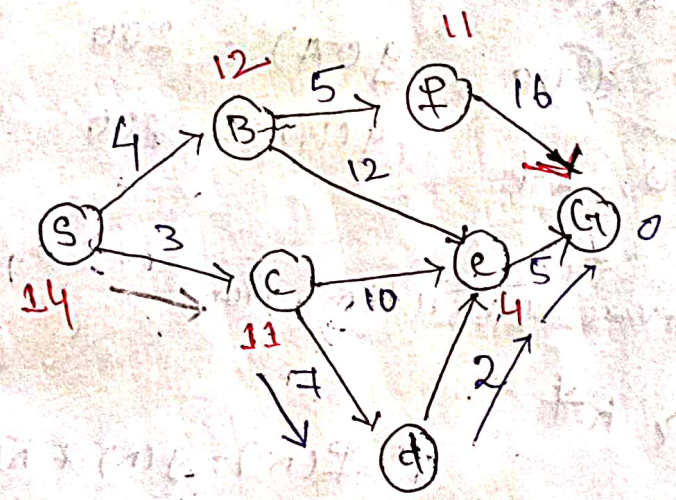
where, $f(N)$ = cost

$g(N)$ = Actual cost from start node to n

$h(N)$ = estimation cost from n to goal node (Heuristic value)



Eg :→



• → Heuristic value

• → Actual cost

5^2

6^2

① $f(S) = 0 + 14 = 14$ 6

~~S → B~~

~~4 + 12 = 16~~

S → C

3 + 11 = (14) SC

② S → B

4 + 12 = (16)

③ SC → e

3 + 10 + 4 = (17)

SC → d

3 + 7 + 6 = (16) SCd

④ SB → F

4 + 5 + 11 = (20)

SB → e

4 + 12 + 4 = (20)

SCd → e

3 + 7 + 2 + 4 = (16)

⑤ Sde → G

3+7+2+5+0

= 17

SBP → G

9+16+0 = 25

Time complexity :→

$= O(b^d)$

Space complexity $= O(b^d)$

Mean End Analysis :

→ In AI, we have studied many search strategies which traverse either in frwd or backword, but a mixturee of these 2 is usually appropriate to solve a complex & large problem.

⇓

Solve the major part of a problem 1st and then go back & solve the small problem which arises while combining the majore parts of the problem.

→ The MEA analysis find the difference bet^n the current state & goal state & applying the operatorrs to reduce the difference.

⇒ To solve MEA, we need to apply the MEA recursively.

→ Following arce the major steps which describe the working principle of MEA algo :-

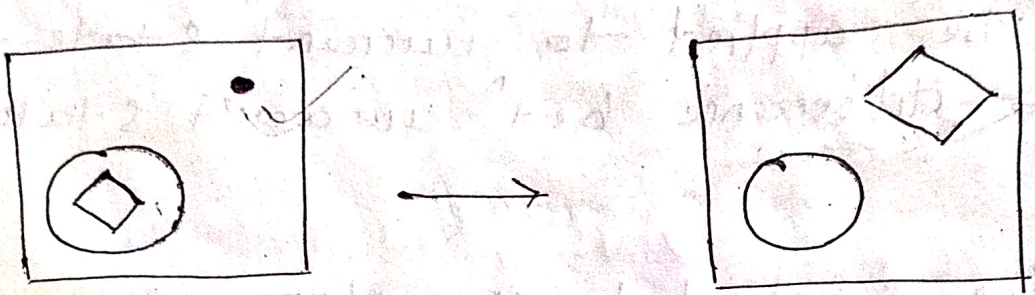
1) Find the difference bet^n initial state & Goal state.

2) From the available operatorrs, select an operatore which can be applied to currcent state to reduce the difference bet^n currcent state & goal state.

3) Apply the selected operatorr.

* if the operatorr can't be applied to the currcent state, in that case we divide the currcent state into sub problems & then we apply an operatorr on sub problem, such type of analysis is called operatorr subgoaling.

eg : ⇒



Initial

Goal

operators we have

i) MOVE - It move the object

ii) Delete — Delete the object

iii) Expand - Increase the size

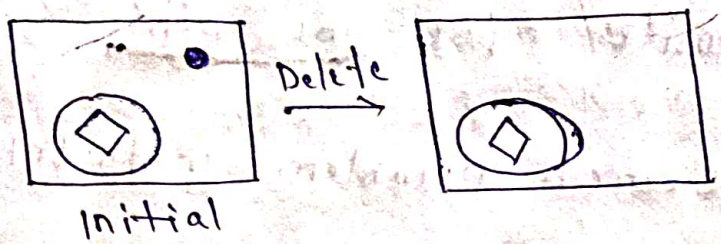
Sol^n

1) Evaluating the inital State :⇒

We will evaluate the inital state & will compare the state to find the difference.

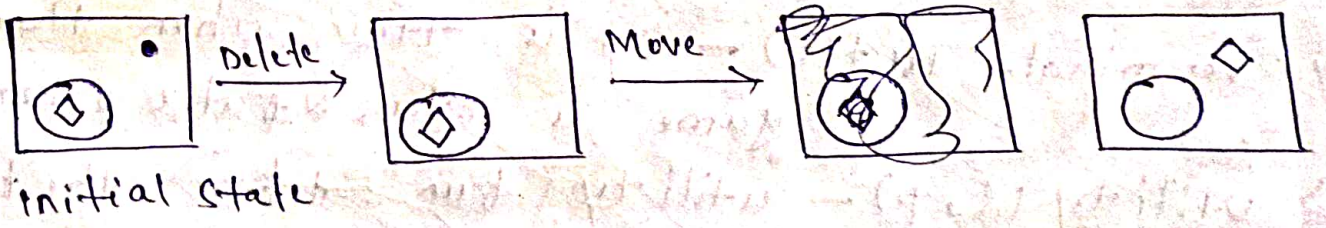
2) Applying Delete operatore :-

There is no dot symbol in goal State, so Ist we have to delete that



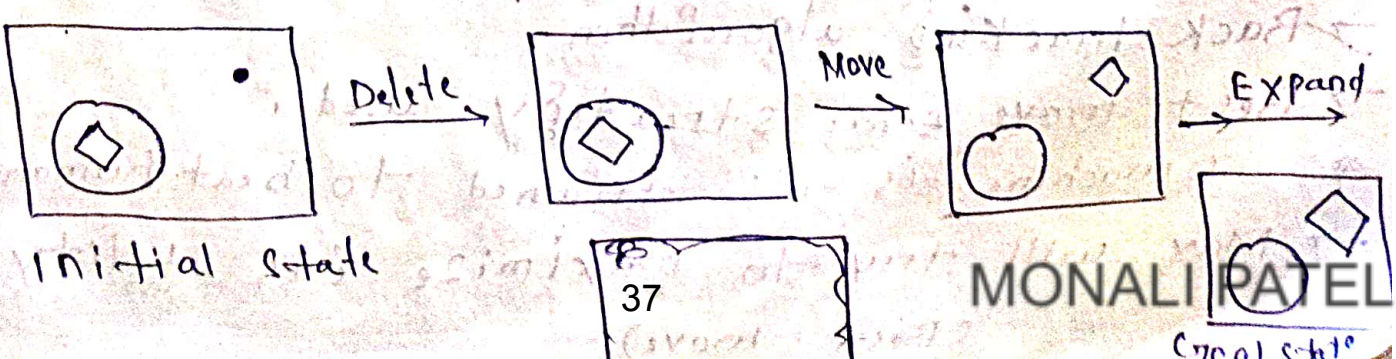
3) Applying Move operatore :-

After applying delete operatore, the new state occure which we will again compare with goal state. After comparcing, we get to know that the dimond is in the outside of the circle, so we will apply Move operatore.



4) Applying Expand operatore :-

The size of the dimond is still occure difference, so, we will apply Expand the operator to increase the size of the dimond.



Adversarial Search

→ It relates to competitive environment in which the agent (multiple agent) goals are in conflict giving rise to adversairal search.

→ In AI the most common games are turen taking two player game i.e 2 agents act alternately utility value at the end of the game is always equal & opposite i.e one wins & other loses.

→ As a kind of search problem with following elements :

i) S_0 – Initial state

ii) Player(s) – Define which player has the turen

iii) Action(s) – Return the set of legal moves in state

iv) Result(s, a) – Transition model which defines the result of the move.

v) Terminal-Test(s) – It is true when the game is over & false otherwise

vi) utility(s, p) – utility fun^n define the final numeric value for game that ends in a terminal state for player p

CSP (constraint satisfication problem)

→ Sudoku, Map coloring

→ CSP consists of 3 components V, D, C

V = Set of variable {V1, V2 ... Vn} → finite variable

D = Set of domain {D1, D2 D3 ... Dn} or domain

for each variable specify

C = Set of constraints/rules (allowable combination of value

{C1, C2 ...}

Ci = (Scope, rule) → defines the values that variable can take

Set of variables that participate in constraint. → binary

V1 V2 → A B

(1,2) (2,4)

C1 = ((V1, V2), (V1 ≠ V2)) or

C1 = ((V1, V2), (A, B))

C1 = ((V1, V2), (1, 2), (1, 4), (2, 4))

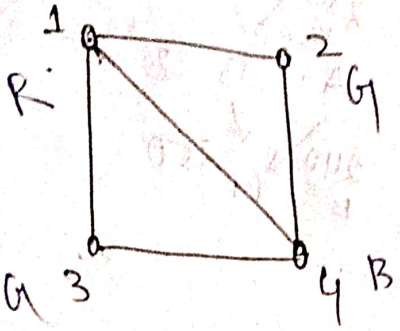
→ 81 (9 x 9)

(3 x 3) → (1, 9)

(81) (1 to 9) with constraint (range)

Eg:⇒ CSP use Backtracking (intelligent)

Constraint Graph



Time efficient

V = {1, 2, 3, 4}

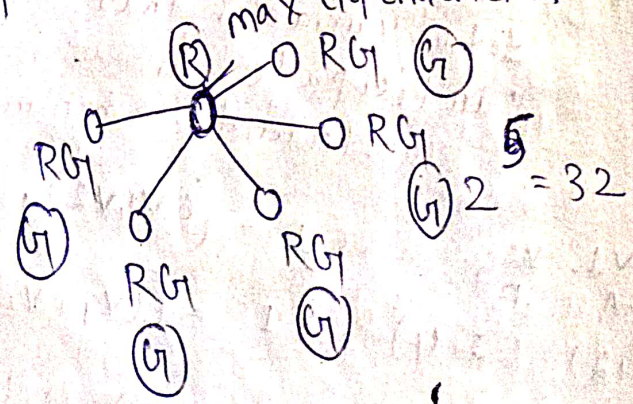
D = {Red, Green, Blue}

C = {1≠2, 1≠3, 1≠4, 2≠4, 3, ≠4}

Ans: ⇒

	1	2	3	4
Initial Domain	R,G,B	R,G,B	R,G,B	R,G,B
1 = R	R	GB	GB	GB
2 = G	R	G	GB	B
✗ 3 = B	R	G	B	(B) empty value
3 = G	R	G	G	B

⇒ Time efficient



$+3 = 8$

Wednesday • November

26 Week 48 Day (330-035)

Constraint Satisfaction Problem

Crypt-Arithmetic Problem

⇒ It is a type of csp where game is about digits & its unique replacement either with alphabets or other symbol.

→ The task in cryptarithmetic problem is to substitute each digit with an alphabet to get the result arithmatically correct.

The rules or constraints on a cryptarithmetic problem are as follows:

i) There should be a unique digit to be replaced with a unique alphabet.

ii) The result should be satisfied by the predefined arithmetic rules i.e 2+2=4, nothing else.

iii) Digit should be only from (0-9)

iv) There should be only one carry forward.

v) The Problem can be solved from both side i.e LHS or RHS.

eg

```
  SEND
+ MORE
--------
 MONEY
```

S-1 Starting from the left hand side the term are S & M assign a digit which could give a satisfactory result.

let's assign S → 9 & M → 1

NOTES

NOVEMBER 2014

Wk	M	T	W	T	F	S	S
44						1	2
45	3	4	5	6	7	8	9
46	10	11	12	13	14	15	16
47	17	18	19	20	21	22	23
48	24	25	26	27	28	29	30

$$\begin{array}{r} S \\ +\ M \\ \hline M\ O \end{array} \rightarrow \begin{array}{r} 9 \\ +\ 1 \\ \hline 10 \end{array}$$

→ Now we move ahead to the next terms E & O to get N as its O/P

$$\begin{array}{r} E \\ +\ O \\ \hline N \end{array} \not\rightarrow \begin{array}{r} 5 \\ +\ 0 \\ \hline 5 \end{array}$$

Not possible bcoz, we can not assign the same digit to 2 letters, So we need to think more & assign some other value

① ← carry

$$\begin{array}{r} E \\ +\ O \\ \hline N \end{array} \quad \begin{array}{r} 5 \\ +\ 0 \\ \hline 6 \end{array}$$

→ Furthere adding the next two terms N & R

① ①

$$\begin{array}{r} N \\ +\ R \\ \hline E \end{array} \not\rightarrow \begin{array}{r} 6 \\ +\ 8 \\ \hline 14 \end{array} \rightarrow \begin{array}{r} 6 \\ +\ 8 \\ \hline 15 \end{array}$$

Let's move ahead

Again on adding the last two terms i.e the right most terms D & E we get Y as its result

$$\begin{array}{r} D \\ +\ E \\ \hline Y \end{array} \quad \begin{array}{r} 7 \\ +\ 5 \\ \hline 12 \end{array}$$

NOTES

Friday • November

28 Week 48 Day (332-033)

Below is the ~~digits~~ representation of the assignment of the digits to the alphabets.

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

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

M O N E Y

Natural Language Processing (NLP)

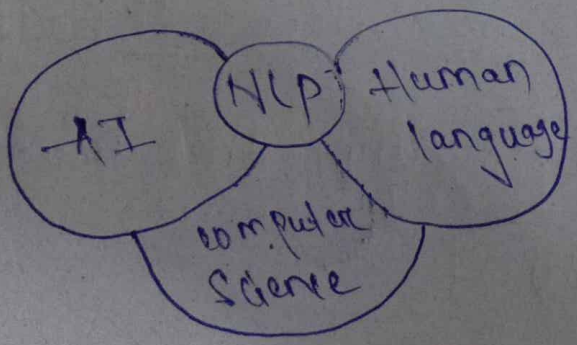
→ NLP is a branch of AI that enables machine to understand the human language.

→ Goal Its goal is to build system that can make sense of text & automatically perform task like translation, spell check or topic classification.

Eg:- Google assistant, Siri, Alexa.

→ Text recommendation when writing emails, offering to translate a facebook post written in diff. language or filtering unwanted promotional emails into your spam.

Note NLP deals with how computers understand & translate human language. To automate this process & deliver accurate response we'll need ML - ML is a process of applying algo. that teach machine how to automatically learn & improve from experience)



Advantages of NLP :-

→ It helps to understand to ask Questions abt any subject & get direct response within second.
→ It offers exact info., i.e it does not offers unnecessary & unwanted info.
→ It helps computers to communicate with human in their language.
→ It is time efficient.

Cons of NLP

→ It may lack in understanding the context.
→ Sometimes it may give some vague ~~conf~~ not clear. ans due to misunderstanding.
→ It is unable to adapt to the new domain as it has a limited vocabulary.

Application of NLP

① Question Answering :- This system answeres the questions asked in the Natural language like Siri, Alexa etc.

② Spam Detection : It detect unwanted emails that can directly go to spam.

③ Sentiment Analysis - opinion mining, like the emotion of a person through text

Eg Is india going to win this match?

YES | NO

4) Machine Translation :- It is used to translate text or speech from one natural lang. to another.

eg:- Google translator.

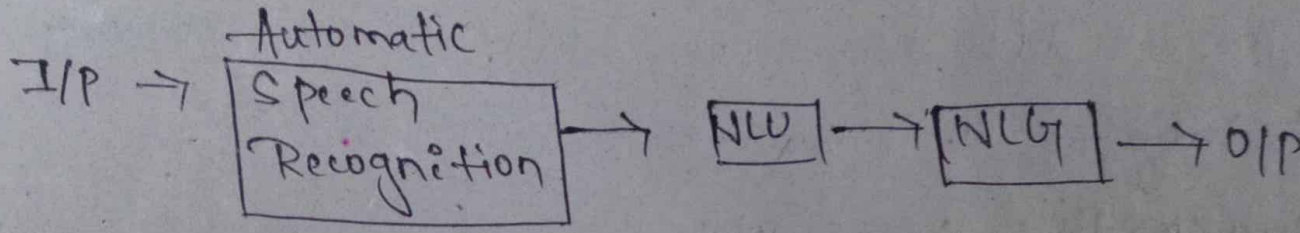
5) Spelling correction: Auto-correction of spelling is a predefined feature MS word, whatsapp etc.

6) Speech Recognition: convert speech into text.

7) Chatbot:

8) Information Extraction

Components of NLP / Architecture of NLP



Natural Language understanding :-

collecting all words & what's his/her intention.

(what do you want to say ?)

challenges

i) Lexical Ambiguity

Two words have the same form or word has more than one meaning

eg The tank was full of water

- water tank
- Army tank

Resolve The water tank was full of water.

ii) Syntatic Ambiguity / Structural ambiguity

Presence of two or more possible meaning within single sentence or sequence of words.

eg - old men & women were taken to safe place
↳ may or may not old

→ The man saw the girl with the telescope
(whether the man saw the girl carrying a telescope / he saw her through telescope

iii) Semantic Ambiguity

→ A word or words with multiple meaning

eg The car hit the pole while it was moving

Resolve lemanization, stanning, Naming entity

iv) Pragmetic Ambiguity

The words which have multiple interpretation (words of a sentence is not specific, it include different meaning)

→ The police are coming

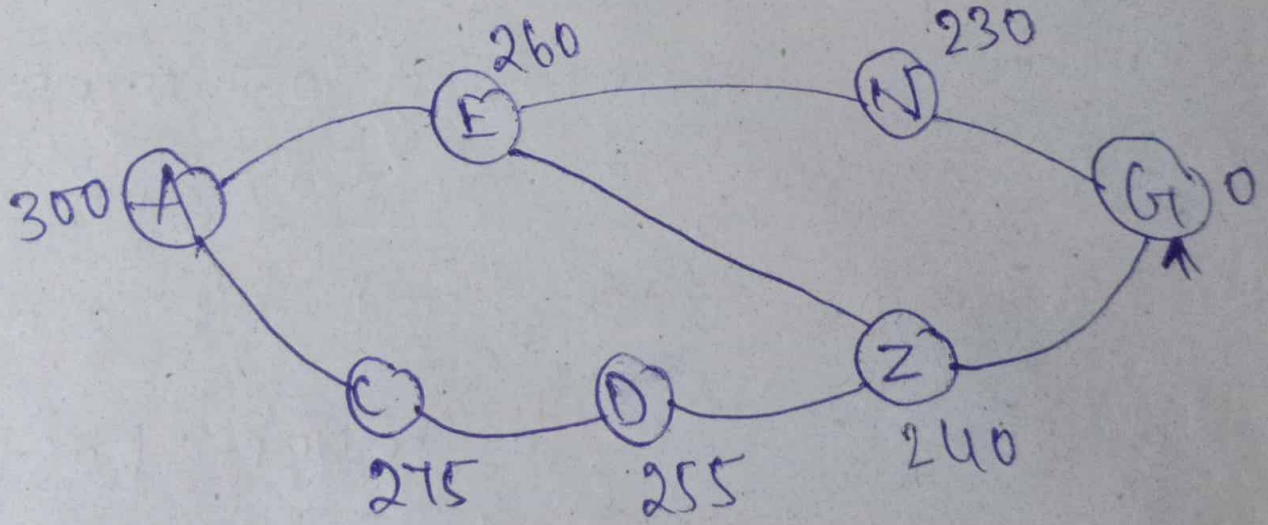
Natural language ~~under~~ Generation (NLG)

what should we say ?

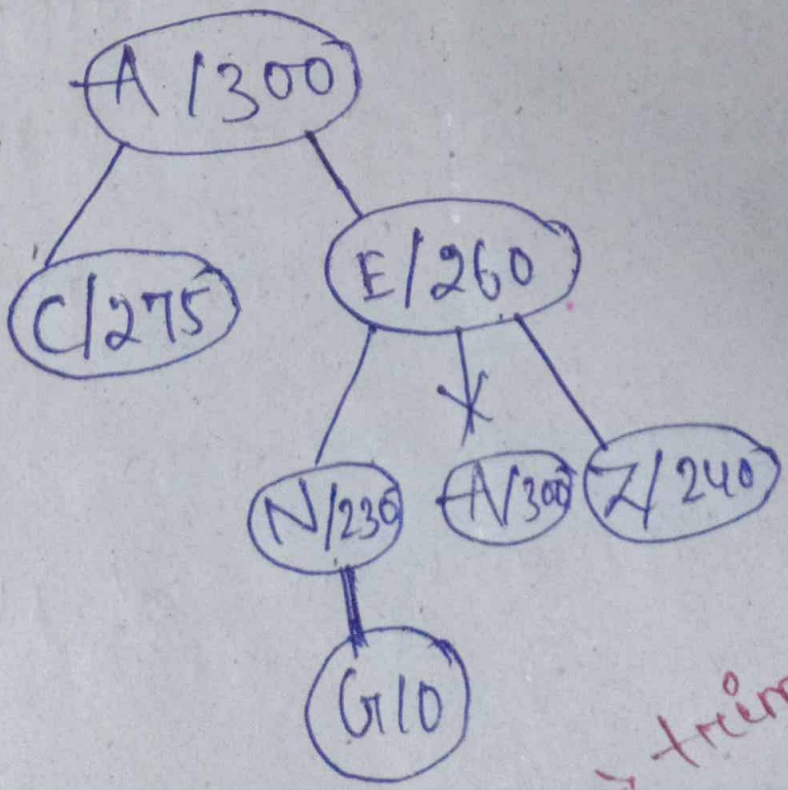
Greedy Best First Search

EXpand the node i.e. closest to the goal on the grounds that this is likely to lead to a sol^n quickly.

heuristic fun $\Psi(n) = h(n)$



⇒



→ trim/cutting away (R

→ α-β Pruning

Adversarial Search

→ It relates to competitive environment in which the agent (multiple agent) goals are in conflict giving rise to adversairal search.

→ In AI the most common games are turn take two player game i.e 2 agents act alternate utility value at the end of the game is always equal & opposite i.e one wins & other loses.

→ As a kind of search problem with follow elements :

i) S_0 – Initial State

ii) Player(s) – Define which player has the turn

iii) Action(s) – Return the set of legal moves in state

iv) Result(s, a) – Transition model which defin the result of the move.

v) Terminal-Test(s) – It is true when the game is over & false otherwise

vi) Utility(s, p) – utility fun^n define the final numeric value for game that ends in a terminal state for player p

Min-Max Algorithm :⇒

→ Back tracking algorithm.

→ Best move strategy used.

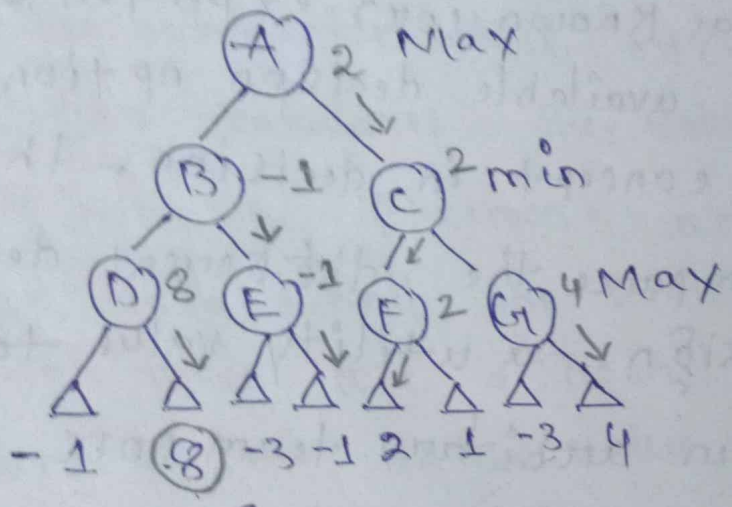
→ (machine is also trained to beat human)

→ Max will try to maximize its utility (Best move)

→ Min will try to minimize utility (score + mor

↓
winning
reward / probabili

eg :-



(Game tree)

$3^2 = 9$

→ Time complexity → $O(b^d)$

on chess → avg. 35 moves / choices

100 (50 + 50)

→ Tictac ...

Alpha-Beta pruning → Removing / cutting away / trim

→ α-β pruning is a modified version of minmax algorithm (optimize technique of min max algo.)

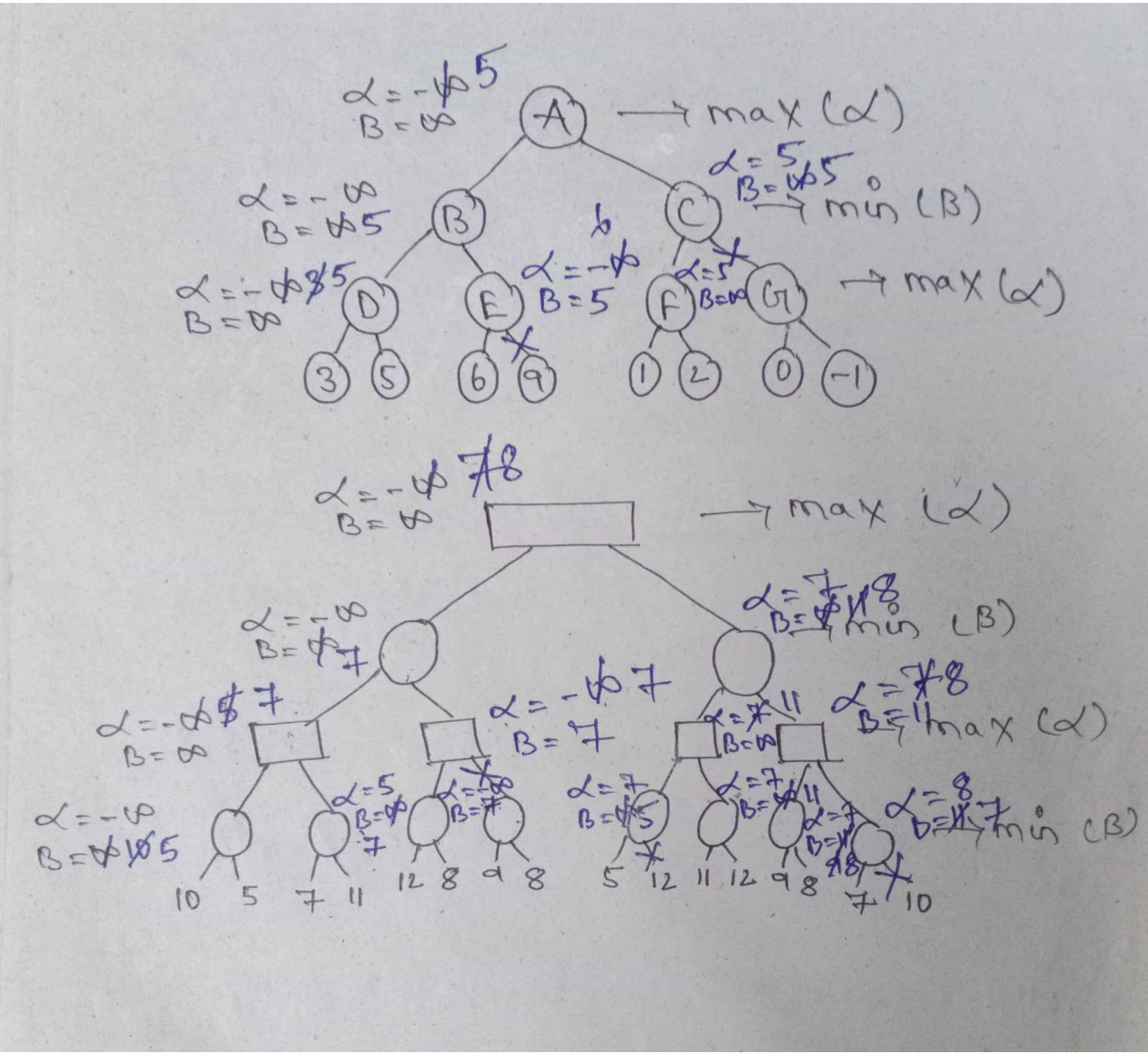
→ It reduces the computation time by a huge factor. This allow us to search much faster & even go into deeper levels in the game tree.

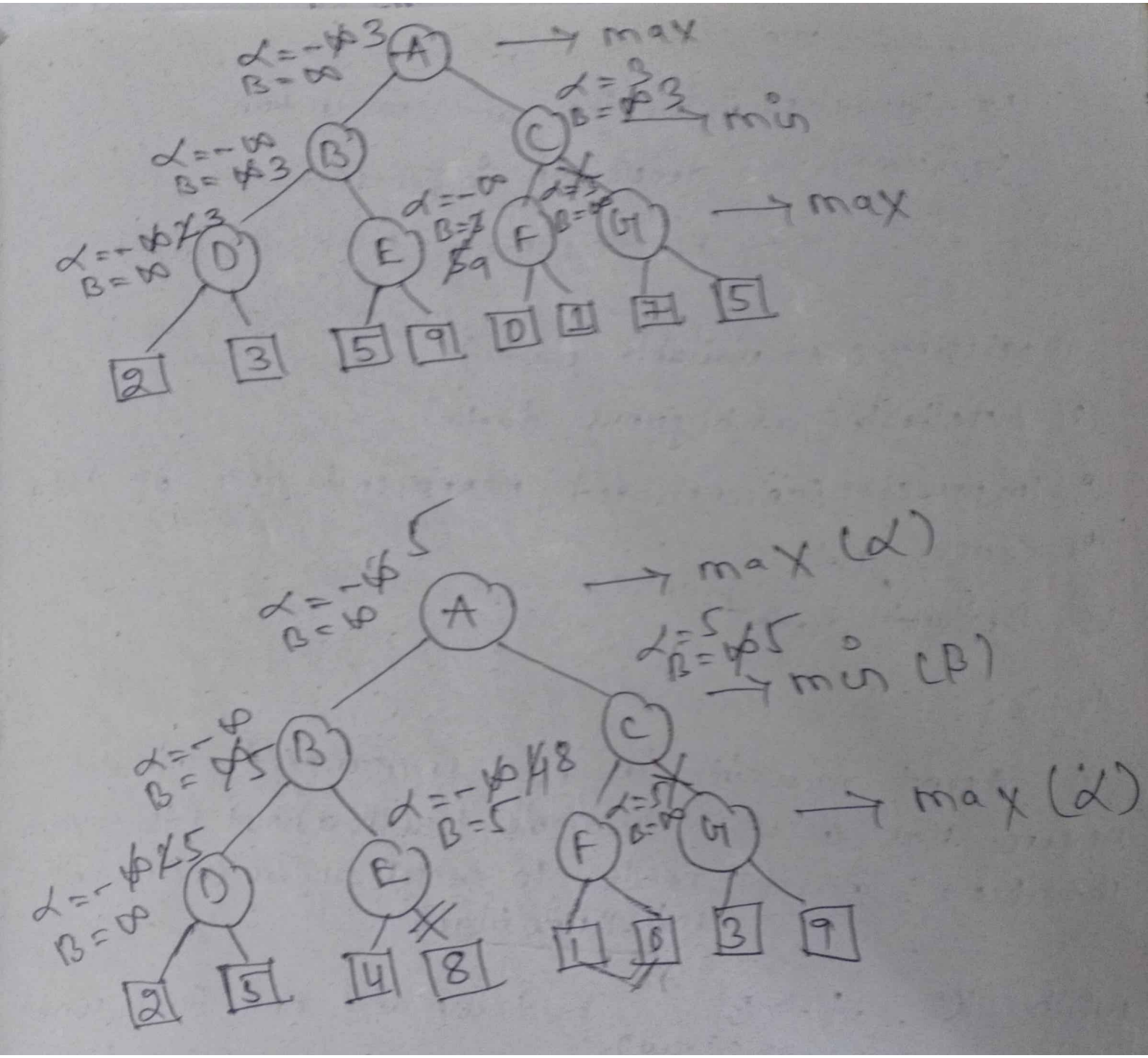
→ It cuts off branches in the game tree which need not be searched bcoz there already exists a better move.

→ This involve 2 threshold parameter α Alpha & Beta

Alpha :- The best (Highest value) value we have found so far at any point along the path of maximizer (max player) $\alpha = -\infty$

Beta : The best (lowest value) value we have found so far at any point along the path of minimizer (min player). $\beta = +\infty$





Optimal Decision in Games:—

→ Find best way to attack / beat the opponent in a game. (good outcome).

→ An optimal decision is a decision that leads to at least as good as known (or) expected outcome as well all other available decision option.

→ It is an imp. concept in decision-theory / AI in order to compare the different decision outcome to assign a utility value to each utility → It is an arbitary term for quantifying the desirability of a particulare decision outcome. (probability)

Mathematically,

Each decision d is a set 'D' of available op^n will lead to an outcome $O = F(d)$. All possible outcome from the set D.

$$U_0(O) = U_0(F(d))$$

Evaluation Function

→ An evaluation function, also know as a heuristic evaluation fun^n, or static evaluation fun^n, is a fun^n used by game playing computere programe to estimate the value or goodness of a position (usually at a leaf terminal node) in a game tree.

→ In Best-First Search algo., each node is expanded using $F(n) = h(n)$

→ In A* Search algo., each node expanded using the evaluation fun^n $F(n) = g(n) + h(n)$.

→ The evaluation fun^n incorporate an estimate of the path cost from the state to a goal.

→ MinMax or Alphabeta purning, each leaf node had a value associated with it.

→ In real world, when we are creating a program to play tic-tac-toe, chess etc. we need to implement a fun^n that calculates the value of the board depending on the placement on the board.

→ The basic idea behind the Evaluation fun^n is to give high value for a board if maximizer's (X) turn or a low value value for minimizer. ↳ O

→ Evaluation fun^n for Tic-Tac-Toe

1) If X wins on the board we give it a +ve value of +10.

X	O	O
	X	
		X

+10

2) If O wins on the board we give it a -ve value -10

O	O	O
	X	X
		X

-10

3) If no one won or the game result is draw then we give a value of +0

X	O	X
O	X	X
O	X	O

+0

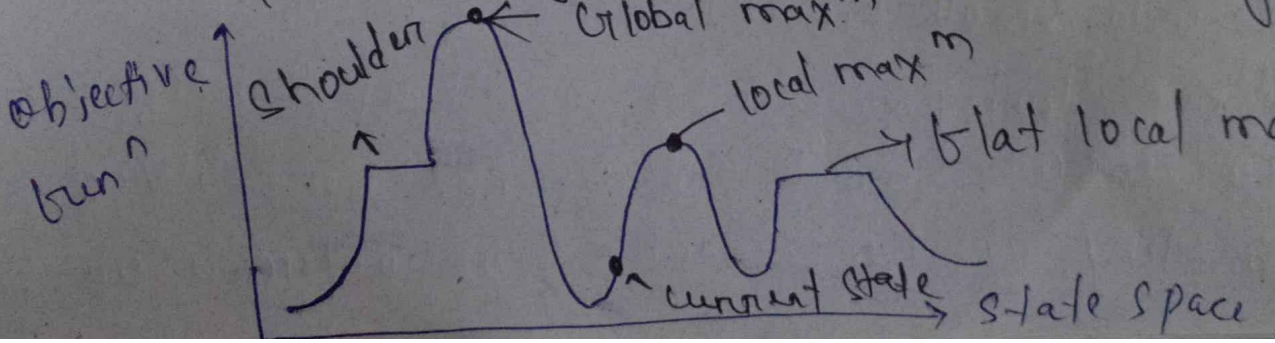
→ Processing of

Hill climbing

Hill climbing algo. is a local search (It can return a valid soln even if its interrupted at any time after finding 1st valid soln) algo which continuously moves in the direction of increasing evaluation/value to find the peak of the mountain or best soln to the problem.

→ It terminates when it reaches peak value where no neighbour has a higher value.

State space diagram of Hill climbing



i) Local Max^m :- It is a state which is better than its neighbor state, but there is also another state which is higher than it.

ii) Global Max^m :- Best possible state (Highest value)

iii) current ~~state~~ State : It is a state in a landscape diagram where an agent is currently present.

iv) Flat local max^m :- All the neighboure state of current state have same value

v) Shoulder :- Region which has an uphill edge.

TYPES

a) Simple Hill climbing (examine only one neighbor at a time ~~select only one~~)

b) Steepest - Ascent HC (examine all neighbor node & select one node which is closer to goal)

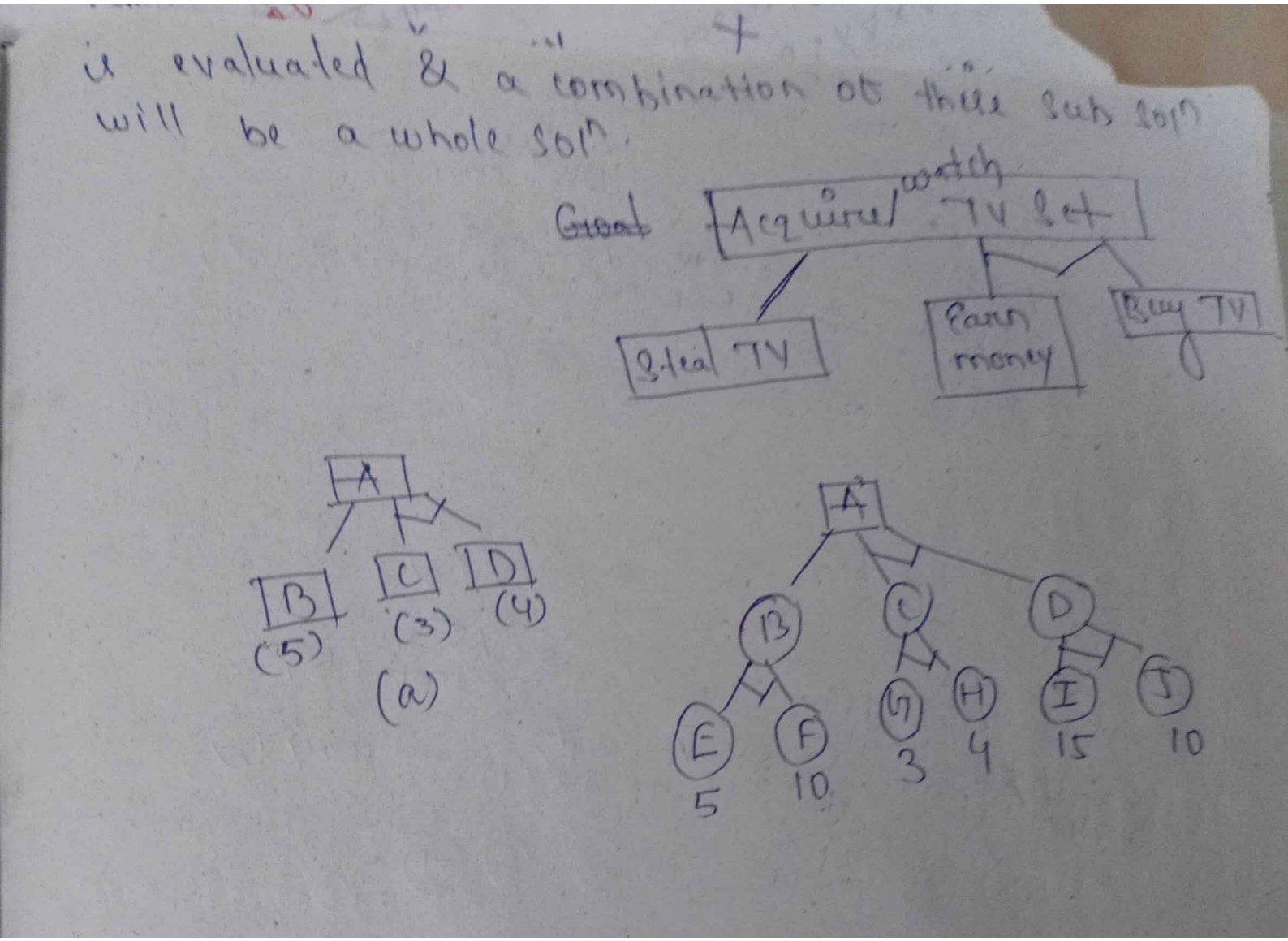
c) Stochastic HC (select one neighbor node randomly)

~~Ao*~~ algorithm

~~And-or~~ Algo.

~~Ao*~~ is based on problem decomposition (breakdown problem into small pieces) when a problem can be divided or decomposed into a set of subproblem, where each sub problem can be solved separately & ~~bor~~ each subproblem is ~~sol~~

it evaluated & a combination of this sub soln will be a whole soln.



iii) Implementation Level

Final part/actions

Propositional Logic :-

1. What is Knowledge Representation

Machine understandable form

It is of 2 types

i) Propositional logic

ii) First orders predicate Logic (FOL)

i) Propositional Logic :⇒ either true or false not both

Symbol	Words
¬	not
∧	and
∨	or
→	implies (if then)
↔	iff

eg X : It is cold
Y : It is sunny
Z : It is Breezy

i) It is not cold ⇒ ¬X

ii) It is cold and it is breezy ⇒ X ∧ Z

iii) It is cold or it is breezy ⇒ X ∨ Z

iv) If it is breezy then it is cold ⇒ Z → X

v) If it Breezy and cold then it is not sunny ⇒ Z ∧ X → ¬Y

vi) It will be cold iff it is Breezy ⇒ X ↔ Z

vii) Anurag did not kill Kajal

~~Kill(Kajal)~~

Kill (Kunal, Kajal)

↓ Action ↓ who ↓ Action performed on whom

viii) Manan eats organge Biscuits

eats (Manan, Biscuits)

↓ Action ∧ color (Biscuit, orange)

color

ix) Anurag watches Elite on Jamtara

→ watches(Anurag, Elite) ∨ watches(Anurag, Jamtara)

x) If Debmalya buys mobile then color is pink

→ Buys(Debmalya, Mobile) → color(mobile, pink)

xi) Anmol becomes happy if and only if anmol eats Donacakes.

2

Becomes(Anmol, happy) ↔ eats(Anmol, Donacake)

Disadvantage of Proposition logic :→

A : Gorcilla is black

B : Gorcilla is Hungry

~~Only one~~ ∀ Gorcilla is Black ✗ in PL

But it can be done in FOL

FOL ⇒ More expressive because it also use Quantifiers

Symbol	Word
¬	not
∧	and
∨	or
→	implies (if then)
↔	iff (is & only if)

1) i) Gorcilla is Black

→ Gorcilla(x) → ∀ Black(x)

Quantifiers :→

i) Universal Quantifiers

∀ for all such that (→) →Limited hant.

ii) Existential Quantifiers (~~Exa~~ Some)

∃ for ~~Such~~ Some such that (∧)

1) All Boys like cricket
$\forall X : Boys(X) \rightarrow likes(X, cricket)$

2) Some Boys like football
$\exists X : Boys(X) \wedge likes(X, football)$

3) Some Girls hate football
$\exists Y : Girls(Y) \wedge likes(Y, football)$

4) All girls love pink
$\forall Y : Girls(Y) \rightarrow love(Y, pink)$

5) Every person who buys a policy is smart
$\forall X \forall Y : person(X) \wedge person(Y) \wedge buys(X, policy) \rightarrow$
~~No person buys expensive policy~~
$Smart(X)$

6) No person buys expensive policy
$\forall X \forall Y : person(X) \wedge policy(Y) \wedge expensive(Y) \rightarrow$
~~buys(X)~~ $\neg buys(X, Y)$

Techniques of Knowledge Representation

There are mainly 4 to ways of knowledge representation

a) Logical Representation → Syntax & Semantic
b) Semantic n/w representation
c) Frame representation
d) production Rule

a) Logical Representation :- (proposition)

→ It is a lang with some concrete rules which deals with propositions & has no ambiguity in representation.
→ It consists of precisely defined

Syntax & Semantics

~~→ which~~ ↓ well-formed sentence ↓ truth meaning of sent

→ Propositional Logic
→ First order predicate Logic (confusion statement)

① PL

→ PL is the simplest logic.
→ A proposition is a declarative statement that's either ~~or~~ True or False
→ It can not predict

Symbol	Words
¬	not
∧	and
∨	or
→	implies
↔	iff (if and only if)

Truth table

① Negation

P	¬P
T	F
F	T

② Conjuction (∧)

P	Q	P ∧ Q
T	T	T
T	F	F
F	T	F
F	F	F

③ Disjunction

P	Q	P ∨ Q
T	T	T
T	F	T
F	T	T
F	F	F

④ Implication

P	Q	P → Q
T	T	T
T	F	F
F	T	T
F	F	T

5) iff & only if

P	Q	P ↔ Q
T	T	T
T	F	F
F	T	F
F	F	T

Forward & Backword chaining

→ data is already given at initial ~~state~~ stage. just you have to find your goal stage or conclusion

→ Move forward
→ Data driven → Data is available

Backward chaining

→ Goal is given you have to find the initial stage.

→ Goal driven

→ Goal state is given

Eg:→ Forward

data

$x = 1$

$y = 2$

Rule

if ($x == 1$ && $y == 2$)

Then $z = 3$;

if ($z == 3$)

then $a = 4$

conclusion

$a = 4$

Backward

Backward

→ Missile(x) → weapon(x)

→ American(colonel)

Forward chainning

→ forward chainning starts from known facts & apply inference rule to extract more data until it reaches the goal.

→ It is bottom-up approach

→ It is known as data-driven. inference technique as we reach the goal using the available data.

→ It applies BFS strategy

→ It test all available rule

→ appln Monitoring, control & interpretation

→ generate an infinite no. of possible conclusion.

→ operates in forward direction

Backward chainning

→ starts from the goal & works backward through inference rules to find the required facts that support the goal.

→ It is top-down approach

→ It is known as goal driven technique as we start from the goal & divide into sub goals to extract the facts.

→ DFS strategy

→ only tests for few required rules

→ appln diagnostic, perscription

→ finite no. of possible conclusion.

→ operates in backward direction

Forward chaining

Sky is blue

Sky → blue

John is a King

King(John)

Everyone like icecream

∀x: likes(x, icecream)

John likes ice cream

likes(John, Icecream)

Q consider the following sentence

a) John likes all kinds of food

b) Apples are food.

c) Chicken is food

d) Anything any one eats & is not killed by is food

e) Bill eats peanuts & is still alive

f) Sue eats everything Bill eats.

① Translate these sentence into formula in Predicate logic.

② Prove that John likes peanuts using forward & Backward chaining.

1)

a) $\forall x : food(x) \rightarrow likes(John, x)$

b) $Apple(x) \rightarrow food(x)$

c) $chicken(x) \rightarrow food(x)$

d) $\forall x \forall y : eats(x, y) \wedge \neg killed(y) \rightarrow food(x)$
(x → any thing eaten by y; food person)

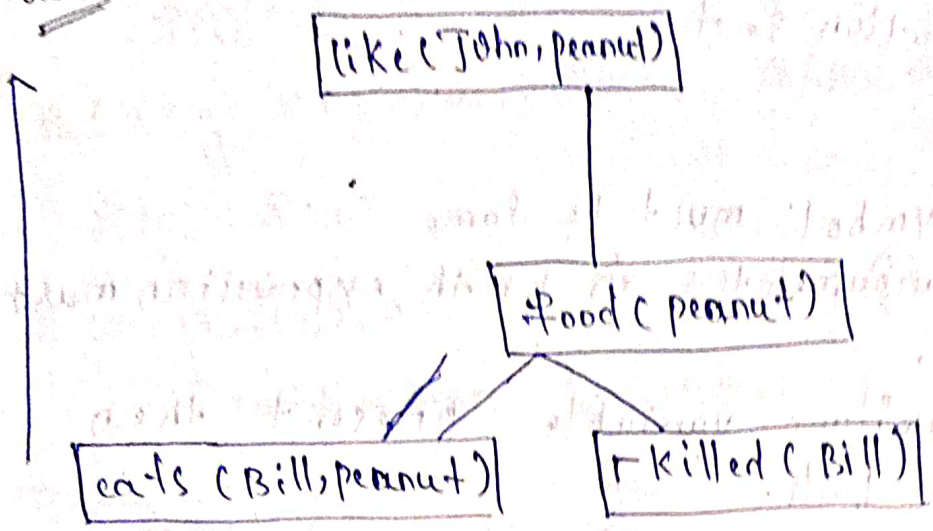
e) $eats(Bill, peanuts) \wedge \neg killed(Bill)$

f) $\forall x\, eat(Bill(x) \rightarrow eat(Sue, x)$

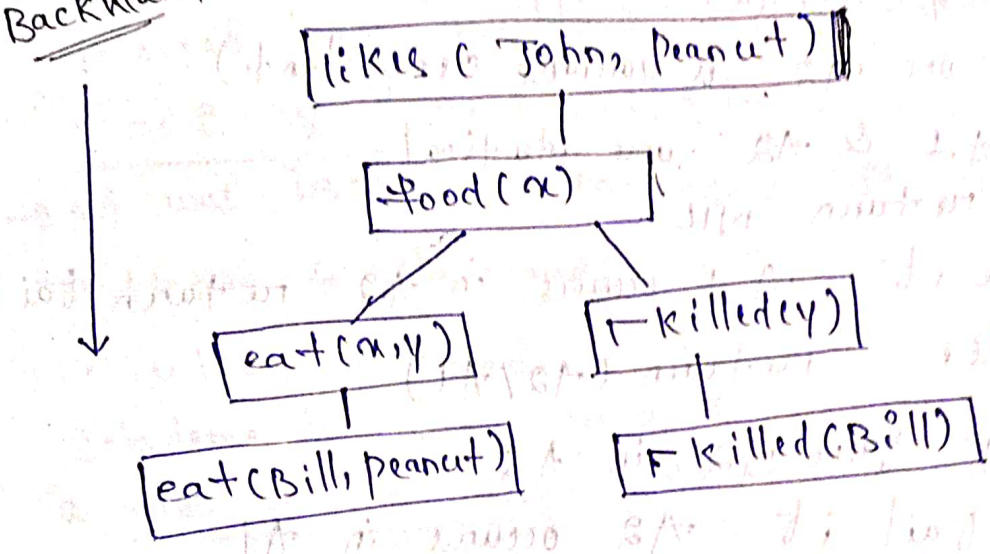
Forward :

Q) John likes peanuts ?

Forward



Backward



Describe unification Algo. with an example

unification :→ Making expression looks identical

→ can be done with the process of substitution

eg :- $P(a, f(y))$ — ①

$P(a, f(g(x))$ — ②

① & ② are identical if x is replace with a

$P(a, f(y))$ — ③

& $f(y)$ is replace with $g(x)$

$P(a, f(g(x)))$ —

[a/x, g(z)/y]

Substitution set

Cond?

→ Predicate Symbol must be same
→ No. of arguments in both expression must be identical.
→ If two similare variable present then unification fails.

Algo. unify (A1, A2)

① if A1 or A2 is variable (constant)
↳ if A1 & A2 are identical
return NIL
↳ Else if A1 occurs in A2 return fail
↳ else return (A2/A1)
↳ check for A2 in A2
↳ fail if A2 occurs in A1
↳ else return (A1/A2)
② if Predicate not same
③ if diff. argument
④ else Subset to Nill
⑤ Loop
⑥ Return SubsT

Q(a, g(x,a), f(y)), Q(a, g(f(b), a), x)

replace with
f(b) / y

Q(a, g(b,a), f(y)), Q(a, g(f(b), a), f(b))

Q. {a(g(a,a)), f(y)), Q(a, g(f(b), x), x}

Sol^n {f(b)/x}

Q(a(g(f(b), a), f(y)), Q(a, g(f(b), a, f(b)

f(y)/f(b)

Q(a, g(f(b), a), f(b)}, Q(a, g(f(b)), a, f(b)

unification successful

Resolution :⇒

→ Resolution is a th^m proving technique that proceeds by building refutation proofs i.e proof by contradiction.

→ It was invented by mathematician John Alan Robinson in the year 1965.

→ Resolution is used, if there are various statements are given & we need to prove a conclusion of those statement.

Steps to convert FOL to CNF ↳ conjuctive NF

① Eliminate implication

② Standardize variable

③ Move Negation inwards

④ Skolemization

⑤ Drop universal Qualifier

① Eliminate implication

$\alpha \rightarrow \beta \equiv \neg\alpha \vee \beta$

$\alpha \leftrightarrow \beta \equiv (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$

② Standardize variable

$\exists(x)$ Smile(x) | $\exists(x)$ Smile(x)

$\exists(x)$ Graduating(x) | $\exists(y)$ Graduating(y)

$\forall x\, happ(x) \mid \forall(z) happy(z)$

③ Move Negation inwards

$\neg(\forall(x) p(x)) \equiv \exists(x) \neg p(x)$

$\neg(\exists(x) p(x)) \equiv \forall(x) \neg p(x)$

$\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$

$\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$

$\neg\neg(\neg\alpha) \equiv \alpha$

④ Skolemization (Remove Existential Qualifiers & replace it by Skolem constant)

$\exists(x)$ Smile(x)

$\exists(y)$ Graduating(y)

After Skolemization

Smile(A)

Graduating(B)

⑤ Drop universal Quantifiers

$\forall x$(Smile(x))

$\forall y$ Graduating(y)

After dropping

Smile(x)

Graduating(y)

Q:-

1) All people who are graduating are happy
2) All happy people Smile
3) Some one is Graduating

i) Convert to FOL

ii) Convert FOL to CNF

iii) Prove that "Is someone Smiling?" using resol

iv) Draw Resolution tree

i) 1) $\forall x (Graduating(x) \rightarrow happy(x))$

2) $\forall x (happy) \rightarrow Smile(x))$

3) $\exists x\ Graduating(x)$

iii) Someone is Smiling $\Rightarrow$

$\exists x\ smile(x)$

contradiction

4) $\vdash \exists x\ Smile(x)$

ii) Convert FOL to CNF $\Rightarrow$ Eliminate implication

1) $\forall x (\neg Graduating(x) \vee happy(x))$

2) $\forall x (\neg happy \vee Smile(x))$

3) $\exists x\ Graduating(x)$

4) $\vdash \exists x\ Smile(x)$

$\Rightarrow$ Standardize variable Apart

$\forall x (\neg Graduating(x) \vee happy(x))$

$\forall y (\neg happy \vee Smile(y))$

$\exists z\ Graduating(z)$

$\vdash \exists w\ Smile(w)$

$\Rightarrow$ Move Negation Inward

$\forall x (\neg Graduating(x) \vee happy(x))$

$\forall y (\neg happy \vee Smile(y))$

$\exists z\ Graduating(z)$

$\forall w \neg Smile(w)$

$\Rightarrow$ Skolemization:-

$\forall x (\neg Graduating(x) \vee happy(x))$

$\forall y (\neg happy \vee Smile(y))$

Graduating(A)

$\forall w$ Smile(w)

③ → Drop universal Quantifier

Graduating(x) ∨ happy(x)

happy(y) ∨ Smile(y)

Graduating(A)

Smile(w)

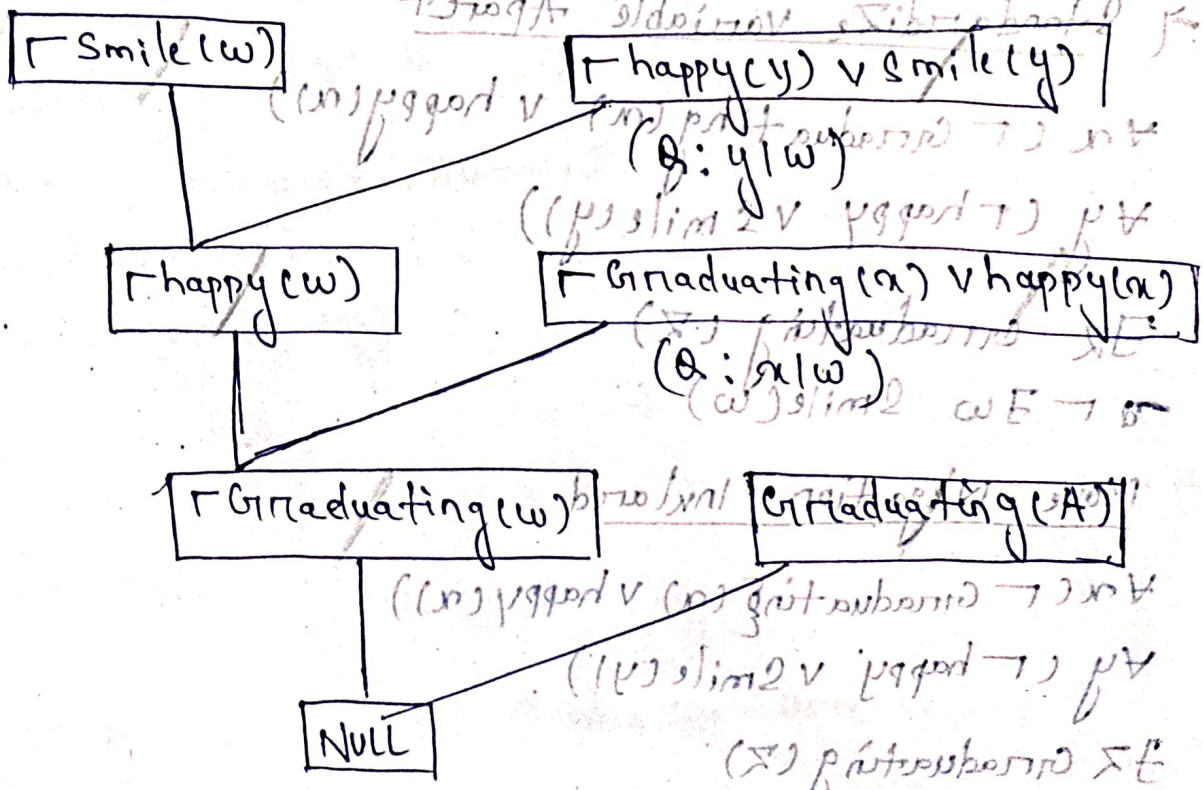
④ Now Sentences are in CNF

⑤ Resolution tree

→ If fact "F" is to be proved then it start with ¬F.

→ It contradicts all the other rule in KB

→ The process Stop when it return NULL clause



Hence someone is Smiling □

Uncertainity

Uncertainity is defined as the lack of exact info; or knowledge that helps us to find correct conclusion.

Source of uncertainity

1. Uncertain input
   - Missing data
   - Noisy data.
2. Uncertain knowledge
   - Multiple cause leads to multiple effects
   - incomplete knowledge casuality in domain
     → Theoritical / practical ignorance

3) uncertain o/p

↳ Abduction, induction are uncertain

↳ incomplete deduction inference

Uncertainity may be caused by problems with data such as -

① Missing (unavailable data)

② unreliable (ambiguous data)

③ Imprecize / inconsistent representation of data

④ Gross data] data.

⑤ Default data

Acting under uncertainity

→ A Agent working in real environment almost never has access to whole truth about its environ. Therefore, agent needs to work under uncertainity (missKnowledge)

→ with KR A → B but we are not sure whether (true) (true)

it's true or false.

→ But when agent works with uncertain knowledge then it might be impossible to construct a complete correct description.

Sol^n for uncertainity

→ Probabilistic reasoning is a way of knowledge representation where we apply the concept of probability to indicate the uncertainity in knowledge.

Probabilistic Reasoning

It is a way of knowledge representation to ~~indicate~~ where we apply concept of probability to indicate the uncertainity in knowledge.

→ we combine probability theory with logic to handle uncertainty

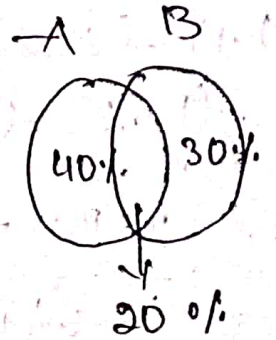
1. Bayes's Rule
2. Bayesian statistics.

Representation of conditional distribution

$$P(A|B) = \frac{P(A \cap B)}{P(B)}, \quad P(B|A) = \frac{P(A \cap B)}{P(A)}$$

↓ probability of A given that B already occured

$P(A|B) = ?$



$$P(B) = \frac{30}{100} = 0.3$$

$$P(A \cap B) = \frac{20}{100} = 0.2$$

$$P(A|B) = \frac{0.2}{0.3} = 0.67$$

Q Mother, father & daughter line up at random in a queue find $P(A|B)$ here

A = daughter on 1 end

B = father in middle

mother = M

father = F

daughter = D

S = { MFD, FMD, MDF, DMF, DFM, FDM }

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{2}{2} = 1$$

Q 2 Dice are thrown & sum of no. is observed to be 4 what is the probability that no. of 2 has appread atleast once $P(B|A)$,

$6 \times 6 = 36$

A = Sum of no. is 4

B = 2 has appread once (1 or more)

$A = \{(2,2), (3,1), (1,3)\} = 3$

$B = \{(2,1), (2,2), (2,3), (2,4), (2,5), (2,6), (3,2), (1,2), (4,2), (5,2), (6,2)\} = 11$

$$P(B|A) = \frac{P(B \cap A)}{P(A)} = \frac{1}{3} = 0.33$$

Probability
Basic Notation :⇒

⇒ Probability can be defined as a chance that an uncertain event will occure. It is the numerical measure of the likehood that an event will occure. The value of probability always remains bet^n 0 & 1

$0 \leq P(A) \leq 1$; where $P(A)$ is the probability of an event A

ideal uncertain

$P(A) = 0$, indicates total uncertainty in an event A

$P(A) = 1$, " " certainty in an event A

We can find the probability of an uncertain event by using the below formula.

probability of occurance = No. of desired outcomes / Total no. of outcomes

$P(\neg A) + P(A) = 1$

Basic Notation of Probability

1) Propositions
2) Atomic event
3) Unconditional (prior) probability
4) Conditional probability
5) Inference using full joint distribution
6) Independence
7) Bayes's Rule.

1) **Propositions** :⇒ (either statement is True/False)
complex proposition can be formed using standard logical connectivites

eg :- 1) [[cavity = true) ∧ (toothache = false)]
2) [[cavity ∧ ¬toothache]] or ((c ∧ ¬t))

Random variable

→ It is used to represent the events & objects in the real world.

→ $P(a) = 1 - P(\neg a)$

2) **Atomic Event**

→ It is a complete specification of the state of the world about which agent uncertain

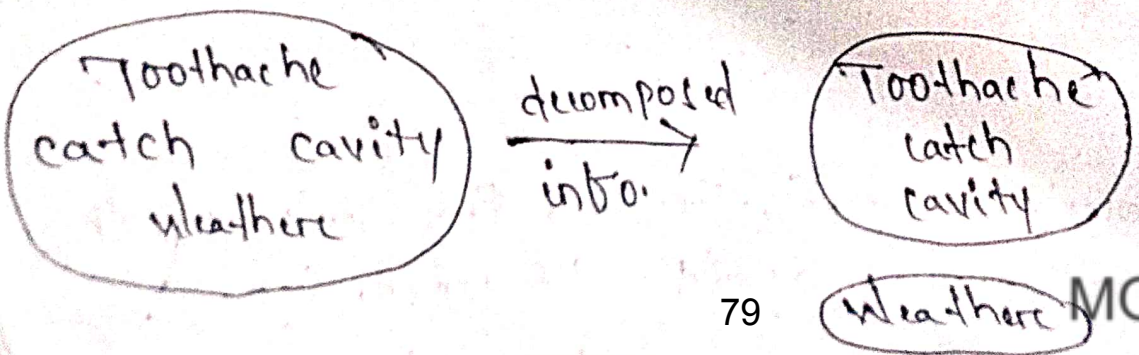
eg
1) cavity = True ∧ toothache = True
2) cavity = False ∧ toothache = False
3) cavity = False ∧ toothache = True
4) cavity = True ∧ toothache = False

3) Unconditional Probability
It is the degree of belief accorded to a proposition in the absence of any other info.
→ Written as P(a)

eg Abc has cavity
P(cavity = true) = 0.1 or P(cavity) = 0.1 or P(¬C) = 0.9

P(weather) = ⟨0.7, 0.2, 0.08, 0.02⟩
P(weather = Sunny) = 0.7
P(weather = Rain) = 0.2
P(weather = cloudy) = 0.08
P(weather = cold) = 0.02

4) Independence
→ It is relation between 2 diff. set of full joint distributions. It is also called as marginal or absolute independence.
→ The weather is independent of once dental problem
P(toothache, catch, cavity, weather) = P(toothache, catch, cavity) P(weather)



5) Inference using full Joint Distribution

Probability inference means computation from observed evidence of posterior probabilities, for query propositions.

	Toothache		~Toothache	
	catch	~catch	catch	~catch
cavity	0.108	0.012	0.072	0.008
~cavity	0.016	0.064	0.144	0.576

P(cavity) = 0.108 + 0.012 + 0.072 + 0.008
= 0.2

Computing probability of cavity,

$$P(\text{cavity} \mid \text{toothache}) = \frac{P(\text{cavity} \wedge \text{toothache})}{P(\text{toothache})}$$

$$= \frac{0.108 + 0.012}{0.108 + 0.012 + 0.016 + 0.064}$$

$$= 0.6$$

$$P(\sim\text{cavity} \mid \text{toothache}) = \frac{P(\sim\text{cavity} \wedge \text{toothache})}{P(\text{toothache})}$$

$$= \frac{0.016 + 0.064}{0.108 + 0.012 + 0.016 + 0.06}$$

$$= 0.4 \quad (1 - P(\text{cavity} \mid \text{toot}$$

P(cavity | ~toothache)

P(cavity | ~toothach

Axioms of probability

→ It is just another way to describing the probability of an event. (principles of probability)

→ 3 axioms are three kinds of universal truth

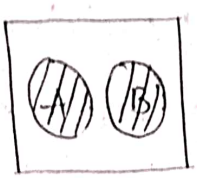
→ Probability law is a fun^n that assigns a no. to events.

① $0 \leq P[A]$

② $P[S] = 1$ S = certain event $0 \leq P(A) \leq 1$

③ If $A \cap B = \phi$ then $P[A \cup B] = P[A] + P[B]$

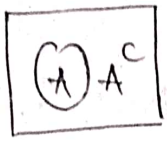
If $A \cap B = \phi$, $P(A \cup B) = P(A) + P(B)$



$A + B$

C-1 : $P(A^c) = 1 - P(A)$

$\Rightarrow P(A) + P(A^c) = 1$



$\Rightarrow P(A \cup A^c) = P(S) = 1$

C-2 $P(A) \leq 1$

C-3 $P(\phi) = 0 \Rightarrow P(S \cup \phi) = P(S) + P(\phi)$

$= 1$

$\Rightarrow P(\phi) = 0$

C-4

$P(A \cup B) = P(A) + P(B) - P(A \cap B)$

$= P(A) + P(B) - \{\phi\}$ it null set

Bayes' Thm

→ It determine the probability of an event with uncertain knowledge.

→ It is a way to calculate the value of P(B|A) with knowledge of P(A|B)

→ It allows updating probability prediction of an event by observing new info. of the real world

From conditional probability

$$P(A|B) = \frac{P(A \cap B)}{P(B)}, \quad P(B|A) = \frac{P(A \cap B)}{P(A)}$$

$$\Rightarrow P(A \cap B) = P(A|B) \cdot P(B) \quad \text{--(i)} \quad \text{or} \quad P(A \cap B) = P(B|A) \cdot P(A) \quad \text{--(ii)}$$

From eq^n (i) & (ii)

$$P(A|B) \cdot P(B) = P(B|A) \cdot P(A)$$

$$* \boxed{P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}}$$ Bayes' Rule or Thm

→ The above eq^n is basic of most modern AI systems for probabilistic inference (simple relationship bet^n Joint & conditional probabilities)

where, P(A|B) = Posterior (Probability of hypothesis A when we have occurred an evidence B)

P(B|A) = Likelihood (Hypothesis is true)

P(A) = Prior Probability (Hypothesis considering the evidence)

P(B) = Marginal Probability (Pure Probability of the evidence)

9 :-

$$P(\text{King}|\text{Face}) = \frac{P(\text{Face}|\text{King}) \cdot P(\text{King})}{P(\text{Face})}$$

(King face card)

K, J, Q

$$= \frac{1 \cdot 4/52}{12/52}$$

$$= \frac{1 \cdot 1/13}{3/13} = \frac{1}{\cancel{12}} \times \frac{\cancel{12}}{3} = \frac{1}{3}$$

Q What is the probability that a patient has disease meningitis with a stiff neck

A doctor is aware that disease meningitis causes a patient to have a stiff neck & it occurs 80% of time o

$P(D) = \frac{1}{30000}$, $P(S) = 2\%$.

Sol)

$P(S|D) = 80\% = 0.8$

$P(S) = 0.02$

$$P(D|S) = \frac{P(S|D) \cdot P(D)}{P(S)} = \frac{0.8 * \frac{1}{30000}}{0.02} = 0.00133$$

2004 Instant

SL :⇒ It is the process where the AI system gathered, organised, and analyse, & interprete numerical info. of data

7th week Tuesday FEBRUARY 10

Statistical Learning :-

→ It is based on the learning of uncertainty in real environment.

→ The methods Probability & decision theory are used to handle uncertainty by the agents

→ First the agent must learn its probabilistic theories of the world from experience.

→ A bayesian view of learning is extremely powerful providing general sol^n to the problems of noise, overfitting & optimal prediction.

Statistical Learning Methods

→ Statistical learning is about inferences

→ The idea is generated from the data & hypothesis & these are called as key terms of statistical learning.

→ Data (samples & population) are Evidence

Eg :- Surprise Candy

Our favourite Surprise candy comes in 2 flavors
a) cherry (yum) &
b) Lime (ugh)

→ The manufacturer has peculiare Sense of humor & wraps each piece of candy in the same opaque wrapper regardless of flavor

MARCH

M	T	W	T	F	S	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

The candy is sold in very large bags, of which there are known to be 5 kinds (again can't identify from the outside):

1) h1 : 100% cherry (10%)
2) h2 : 75% cherry + 25% Lime (20%)
3) h3 : 50% cherry + 50% Lime (40%)
4) h4 : 25% cherry + 75% Lime (20%)
5) h5 : 100% Lime (10%)

Then we observe candies drawn from some bag:
→ what kind of bag is it? What flavors will the next candy be?

⇒ Given a new bag of candy, the random variable H denotes the type of the bag, with possible values h1 through h5
⇒ H is not directly observable.
⇒ As the pieces of candy are opened & inspected
→ Data are revealed - D1, D2 ... DN
→ where each Di is a random variable with possible values cherry & lime

→ The basic task faced by the agent is to predict the flavor of the next piece of candy

→ Bayesian Learning simply calculate the probability of each hypothesis, given the data, & makes predictions on that basis

→ Let D represent all the data, with observed value d; then the probability of each hypothesis is obtained by bayes' rule:

$$P(h_i/d) = \alpha P(d/h_i) \cdot P(h_i) :$$

Genetic Algorithm (John Holland)

→ Abstraction of Real Biological Evaluation

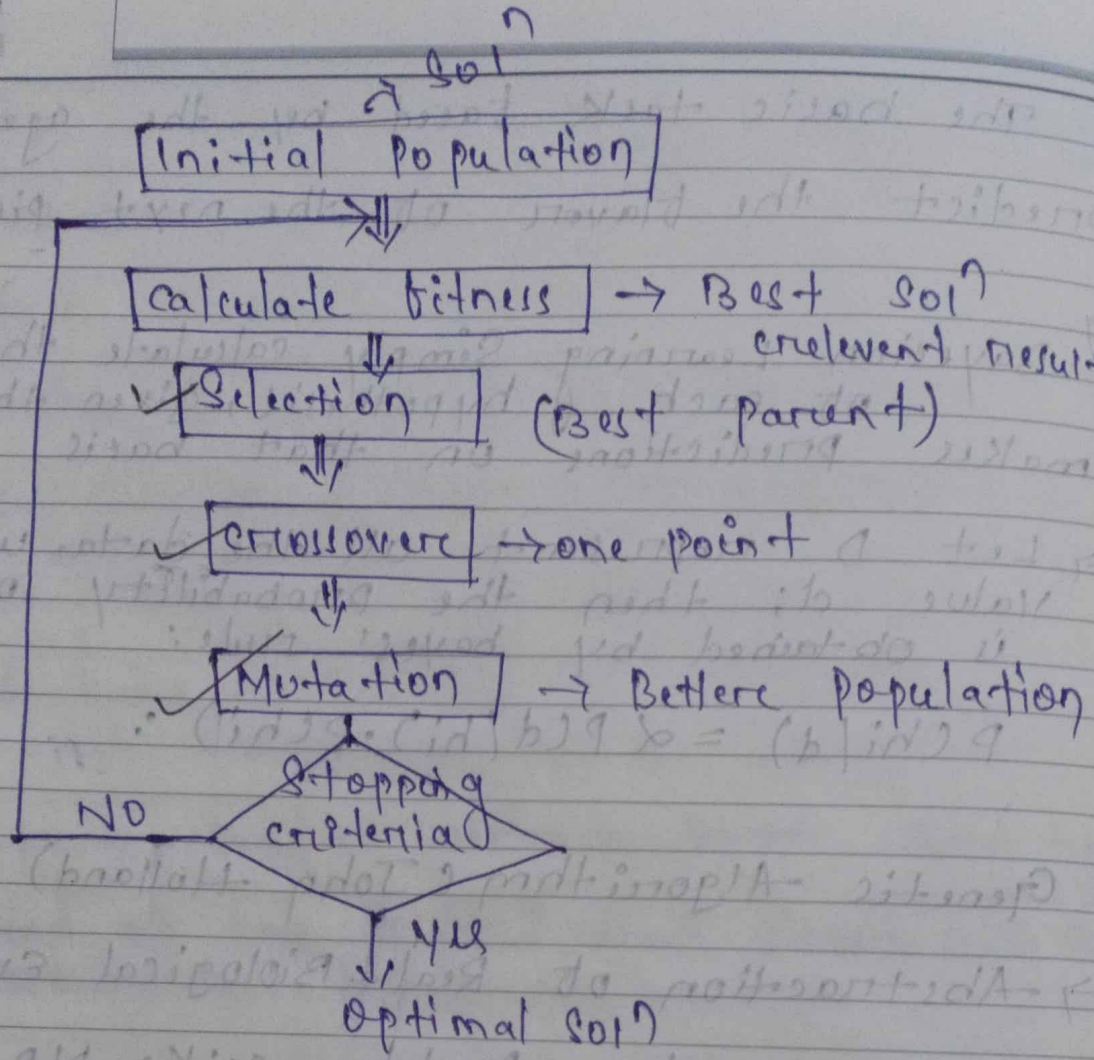
→ Solve complex problem (like NP hard)

→ focus on optimization (optimal soln)

→ Population of possible soln for a given problem

→ From a group of individuals, the best will survive

phenotype —Encode→ Genotype (propere Representation)
(search space) ←Decode— raw soln



cake
make
Take

abcde
efghi
swap

abchi
efgde

Genetic operators

1) Selection operators

2) Mutation

3) crossovers

1) Selection operators

⇒ Process of selecting two or more Parents form the population for crossing.

⇒ Purpose of selection is its emaphasize fitter individuals in the population in hopes that offspring have higher fitters

Methods - Roulette-wheel Selection, Boltzman Selection, rank Selection, tournament selection

Roulette wheel Selection

Depending on % contribution to the total population fitness string is selected for ~~meeting~~ mating to form the next generation

Sunday 15

Eg Maximize the fun $f(x) = x^2$ with x in interval $[0, 31)$ i.e $x = 0, 1, \dots 30, 31$

Soln 1. Generate initial population at random. they are chromosome or genotypes.

2) 01101(13), 11000(24), 01000(8), 10011(19)

2. calculate fitness

a) Decode into an integers (called Phenotype)
01101 → 13, 11000 → 24, 01000 → 8, 10011 → 19

b) Evaluate fitness $F(x) = x^2$

13 → 169, 24 → 576, 8 → 64, 19 → 361

3) Select Parents (2 individuals) based on the fitness in P:

$$P_i = f_i / \left(\sum_{j=1}^{n} f_j\right)$$

f_i = fitness for string i in population
P_i = Prob. of string i being selected
n = no. of individual in the population

String No.	Initial / Mutation	x value	Fitness F_i $F(x)=x^2$	P_i	Expected count $n * P_i$
1	01101	13	169	169/1170 = 0.14	4*0.14 = 0.56
2	11000	24	576	0.49	1.97
3	01000	8	64	0.06	0.22
4	10011	19	361	0.31	1.23
Sum			1170	1.00	4.00
Average			293	0.25	1.00
Max			576	0.49	1.97

2) crossovere operatore can be of eithere one point or 2-point crossovere

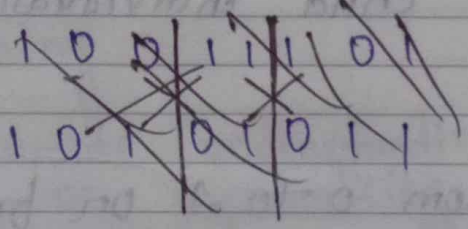
→ In 1 point crossovere, selected pairs of string is cut at some random population position & segement are swapped in form new pairs of strings.

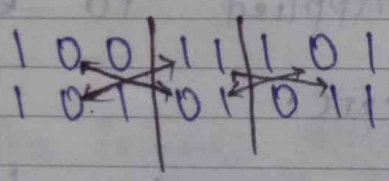
→ In 2 points, there will be 2 break points

eg 1 0 0 | 1 1 1 0 1
1 0 1 | 0 1 0 1 1

offsprings 1 0 0 0 1 0 1 1
1 0 1 1 1 1 0 1

2 points





offspring ~~0 0 0 0 1 1 1 0~~ 1 0 0 0 1 1 0 1
1 0 1 1 1 0 1 1

~~String No.~~

→ PTO

Sunday 22

String No.	Mating pool	crossover site P[c] (random)	offspring	x value	fitness $f(x) = x^2$
1	01101	4	01100	12	144
2	11001	4	11001	25	625
2	11000	2	11011	27	729
4	10011	2	10000	16	256
Sum					1754
Avg					439
Max					

3) Mutation Operators:

Applied to each child individually after crossovers.

→ Bits are changed from 0 to 1 or form 1 to 0 at randomly choosen position of randomly selected Strings.

Strings	offspring (after crossover)	Offspring (after Mutation)	x value	Fitness $f(x) = x^2$
1	01100	11100	26	676
2	11001	11001	25	625
2	11011	11011	27	729
4	10000	10100	18	324
Sum				2354
Avg				588
max				729

Rote Learning

→ Learning by Memorization

Why memorization ?

→ We store the computed values so that we don't have to recompute them later.

eg Table, Square, & cubes, formulas etc

→ Simplest type of learning without any modification is, Simply copied into the Knowledge base.

→ Whenever a computer stores a piece of info. it is performing basic form of learning i.e Rote learning (cache data)

Rote Learning

It avoids understanding the inner complexity but focuses on memorizing the materials so that it can be recalled exactly the way it was read.

① Learning by Memorization

② Learning by Something by Repetation - Over & over again method is used

① Saving Knowledge So it can be used again

② Retrival is the only problem

③ No Replead computation, inference of query

eg Caching, cookie

Learning with complete Data

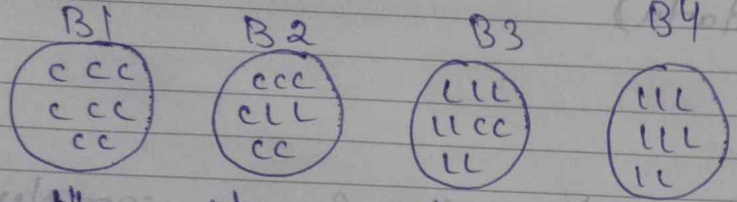
1) Maximum Likelihood Parameter learning

Final simplification is provided by assuming a uniform prior over the space of hypothesis

→ In that case, MAP learning reduces to choosing h, that maximize $P(d|h_i)$.

→ This is called ~~Maximiz~~ max^m likelihood Hypothesis

eg



Learning with Hidden Data : EM algorithm
↳ Expection maximization algo.

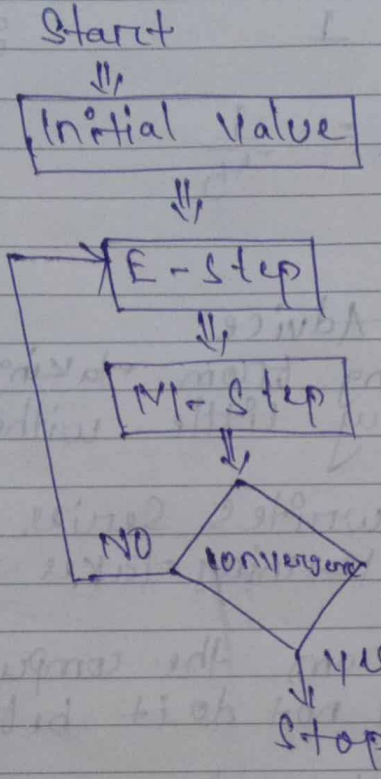
(1) What is Estimation

Estimate our expection from machine & then classify the data into some classes.

(2) What is Maximization

What ever we estimated, should be maximized.

3) Convergence
Repeat step1 & step2 until convergence



EM algo.
It is a iterative estimation algo. that can derive the max^m likelihood estimation in the presence of missing data.

eg :-
1. Assume we have 2 coins C1 & C2
2. θ_1 → probability of getting head with C1
θ_2 → " " " " with C2

				Head	Tail
C1 →	H	T	H	2	1
C2 →	T	H	H	1	2
C1 →	T	H	T	1	2

$\theta_1 = \frac{3}{6}$ $\theta_2 = \frac{1}{5}$

Learning By Taking Advice
(Learning from taking guidance)

→ Compute can do very little without program

→ When programmer writes series of program a rudimentry kind of learning takes place.

→ After learning program the computers can do something which could not do it before.

→ Interpreter or compiler is needed to interprute to change the teacher simulation into code that machine can directly executed.

Request → interpret → operationlize → integrate
(what I have to do) applicable or not (the sol^n)

2004

Saturday 28 FEBRUARY

Learning By Problem Solving

→ No teacher - Advices but learning by experiences

→ It doesn't involves increase in knowledge just method of using knowledge.

Learning by Parameters - Adjustment

→ Many programs rely on evaluation procedure that combines info. from several sources into a single summary statistic.

→ Pattern classification often combine with Statistic

→ It is difficult to know on how much weight has to be attached to each feature. (evaluation function)

→ One way to find is experience.

Learning By Example - Induction

→ classification: process of assigning to a particular input.

Sunday 29

→ It is simplest form - It is present as straight forward recognisition task.

① Isolate a set of features relevent to the task domain define each class.

$c_1 t_1 + c_2 t_2 + \cdots c_n t_n$

MARCH

M	T	W	T	F	S	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

01 10th week Monday MARCH 2004

Instant

19. Tackle Weather Prediction

t1 = rainfall
t2 = Humidity
t3 = temp.
t4 = location of cold.

(2)

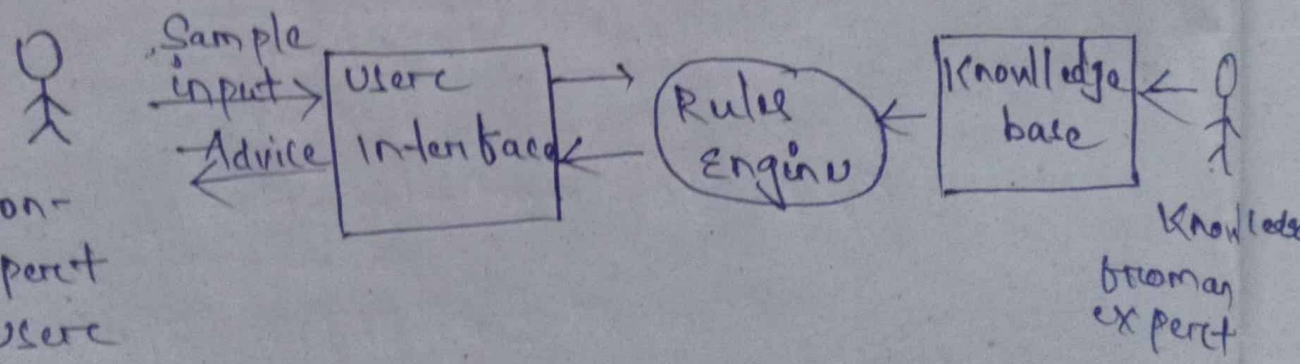
Expert System
by Domain expert

→ An expert system is a computer program i.e designed to solve complex problem & to provide decision making ability like human. It performs this by extracting knowledge from its knowledge base using reasoning & inference rules)

→ ES is a part of AI & was developed in the year 1970. It solves the most complex issue as an expert.

→ The system helps in decision making for complex problems using both facts & heuristics like a human expert.

working system of Expert System



Note :- Expert system is not used to replace the human experts; instead, it is used to assist the human in making complex decision.

characteristics of Expert system

a) High performance:
The expert system provides high performance for solving any type of complex problem of a specific domain with high efficiency & accuracy.

b) understandable:
It responds in a way that can be easily understandable by the user. It
→ It can take input in human language & provides the o/p in the same way.

c) Reliable:
It is much reliable for generating & efficient & accurate o/p

d) Highly responsive: It provides the result for any complex query within a very short period of time

components of Expert System
It consists of 3 components.

a) UI
b) Inference Engine
c) Knowledge base.

Limitation of ES

i) Do not have human like decision making powers.
ii) Can not Posses human capabilities.
iii) Can not Produce correct result from less amt of knowledge.
iv) Require excessive training.

Advantages

i) Low accessibility cost.
ii) Fast Response
iii) Not affected by emotions, as like humans.
iv) Low error rate.

Disadvantages

→ The ES has no emotions.
→ common sense is the main issue of the ES.
→ It is developed for a specific domain
→ Need to be updated manually. It does not learn itself.
→ Not able to explain the logic behind the decision.

Some popular example of the expert system:

a) Dendral :- Made as a chemical analysis expert system.
→ It was used in organic chemistry to detect unknown organic molules with the help of their mass spectria & knowledge base of chemistry.

b) MYCIN : Earliest bckward chaining ES that was designed to find the bacteria causing infection.
→ It was also used for recommendation of antibiotics & the diagnosis of blood clotting diseases.

c) PXDES : Determine the type & level of lung cancer.

→ To determine the disease, it takes a picture from the upper body, which looks like the shadow, which identifies the type & degree of harm.

d) CaDeT : It is diagnostic support system that can detect cancer at early stages.