UNIT- 1

Artificial Intelligence

Artificial Intelligence is a machine to work and behave like human in the recent past, AI has been able to accomplish this by creating machine and robotics that are being used in wide range of field including health care, robotics, and marketing.

AI has found it's way into our daily life. It has become so general that we don't realise, we use it, all the time for instance. Example-

1) Google is able to give you such accurate such result.
2) Your Facebook field always gives you content based on interest.

NOTE → Copy ~~it~~ of something ~~n~~ natural (human ~~beig~~ being) is called AI

OR

The science of getting machine to mimic behaviour of humans.

Categories of AI

1) Narrow AI → * It is also known as weak AI.
   * It involves AI only to specific task.

for eg:- Alexa- Alexa is a very good example of narrow AI. It operates within a limit predefined function.

(ii) face verification - That you see in your Iphone, ~~or autopilot feature at Tesla~~

(iii) Google map.

2) Artificial General Intelligence -(AGI)-

* It is also known as ~~a~~ strong AI
* It involves machines that posses the ability to perform any intellectual task that are human being can.

Eg → (i) Sensory perception → AGI would ~~ex~~ excel in colour recognation; it would also able to percieve depth and 3D in static images.

(ii) Advanced Robotics

(iii) Autopilot feature of Tesla.

3.) Artificial Super Intelligence

The time when capability of computers will surpass human. Eg →

i) Science fiction books, where machine will take over the world.

(ii) It is presently seen in hypotetical situation as depicted in movies.

Real life examples of AI

1) Google AI eye, doctor is another ~~But~~ initiative taking by google to develop a AI system which can examine retina scan and diabetic Retinopaty with causes blindness.

2.) Social Media Paltform, like facebook AI is used for face verification where in machine language and deep language concept are used to detect facial features and tag your friends

3.) Twitter's - It is being used to identify hate speech and wrong words in tweet It marks use of Machine learning, ~~te~~ deep learning and Natural ~~latt~~ language Processing to filter out offensive content.

4.) Google predictive Search - 98% found by mean human artificaly intelligence machine, the google predictive search is one of the most famous application.

Demand for AI.

1) More computation power - AI requires a lot of computing power Since we have more computational power. Now it is possible to Implement AI in our daily aspects.

2) We have a lot of data at present.
e.g:- Big data enables us to do this more efficiently.

There are basically effective algorithm which are based on the idea of neural network. Neural network is nothing but the concept behind Deep learning.

Since we have better algorithm which can do better computation with more accuracy and the demand of AI has increase.

Disadvantages of Artificial Intelligence.

1) AI can do something devasting. for eg:- Weapons and missile
2.) High cost of creation
3.) Unemployement
4.) No human replication due to lack of emotions.
5.) Zero creativity.

Intelligent Agent - Intelligent agent is composed of:

1 Reasoning
2 Learning
3 Problem Solving
4 Perception
5 Linguistic Intelligent.

Intelligent agent is a system that percieves its environment (through sensors effectors) and take action (through actuators) to maximize the chance of success.

Different types of agent

1) Human agent - It has eye and other organ for sensor and hand, leg, mouth and other body parts (hand, leg) for actuators.

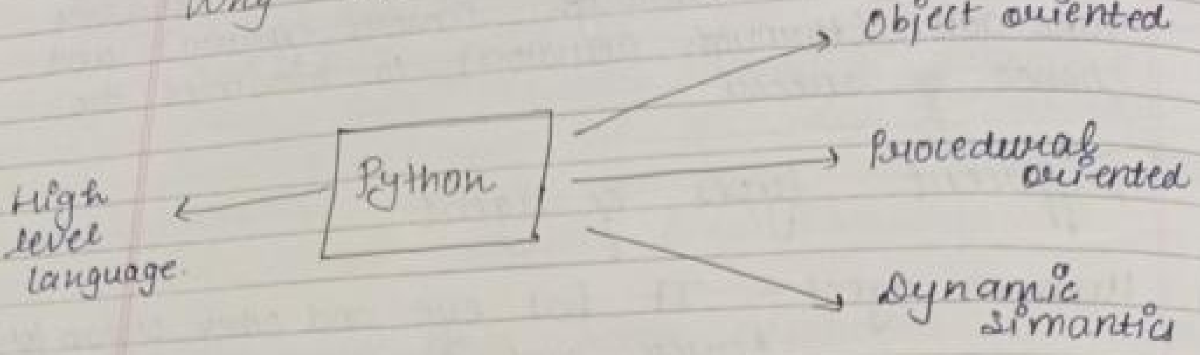
2) Robotics agent - It might have camera and Infra-red range finder for sensors and various motor for actuators.

3.) Software agent - It recieves keystrokes, file content, network packet as sensory input and act on the environment by displaying on the screen, writing file and sending network packet.

Rationality depends on four things

1.) The performance measure that defines the criteria of success.
2.) The agent prior knowledge of environment
3.) The action that the agent can perform
4.) The agent percept sequence to date.

Why we are using Python in AI?



It is a vary graphs this language. It is also free and open source. It is portable supported by many platform like linux, windows free (BSD) and maintos etc.

NOTE → This language to develop data science, algorithm, machine learning algorithm and [IOT] ~~proo~~ projects.

*. Agents - Agents is a software agent that ~~ass~~ assist user and act in ~~pe~~ performing computer related task.

Intelligent behaviour -

1.) Perceiving once environment.
2.) Acting in complex environment.
3.) learning and understanding from expeirence.
4.) Reasoning → To solve problems and discover

hidden knowledge
5) Knowledge → Applying successfully in new situation
6) Thinking → abstractly using analogies
7) Communicating → with others and more like

Omniscience, learning & Autonomy

An omniscience agent know the actual outcome of its action and can act accordingly, but omniscience is impossible in reality.

* Doing action in order to modify future percept sometimes called information gathering. It is an important part of rationality.

* A rational agent is not only to gather information but also to learn as much as possible from what it percieve.

* To the extent that the agent relies on prior knowledge of its designer rather than on its own percept we say that the agent lacks autonomy.

* A rational agent should be autonomous. It should learn what it can compensate or ~~parichtial~~ partial or incorrect prior knowledge.

NOTE → A agent perceive an act in environment has an architecture and is implemented by an agent program.

* imp "Task environment should be PEAS"
P → Performance
E → Environment
A → Actuators
S → Sensor.

All agent can improve their performance through learning.

1) Simple Reflex Agent

The simplest kind of agent is the simple reflex agent, these agent ~~t~~ select actions on the basis of the ~~c~~ current percept, ignoring the rest of the percept history.

* Select action on the basis of only the current percept.
* implement through condition action rule.

Characteristics -

a) Only work if the environment is fully observable
lacking of history easily get stuck in infinite loop

and

c.) One solution is to randomize action.

2.) Model-Based Reflex Agent

The most effective way to handel ~~pust~~ partial observability is for the agent to keep track of the part of the world it can't see now i.e the agent should maintain some short internal state that depend on the percent history and their by reflect at least some of the ~~th~~ unobservable aspect of current state. An agent that uses such a model, based mode is called a model based agent.

3.) Goal Based Agent.

Goal Based Agent know the current state of the environment is not always enough to decide what to do. In other words, as well as current state discription the agent need some short Goal information that describe the situation that are disirable.

4.) Utility Based Agent

It is also known as Happy and ~~H~~ unhappy agent. Goals alone are not really enough generate high quality behaviour in most environment.

Some are better, have a utility.

* Improve on goals
* selecting between conflicting goals
* Select appropriate between several goals
* based on likely hood of success

Unit 2

Search strategies

Uninformed

It is also called blind search, Brute force or exhaustive search

Informed

heuristic search and intelligent search

Difference b/w uninformed and informed search

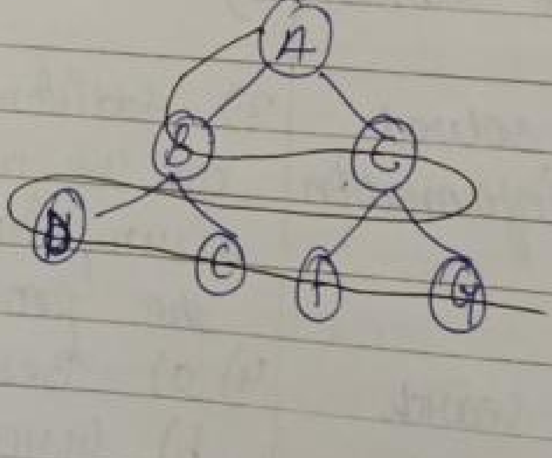
Uninformed Search	Informed Search
1) No information about the past cost from the current stage to the goal stage.	1) The ~~past~~ path cost from the current stage to the goal stage is calculated, to select the minimum past cost as the next stage
2.) less effective in search method	2.) more effective
3.) Problem to be solved with a given information	3.) Additional information can be added as assumptions to solve the problem.
4.) a) Breadth first search b) Depth first search	4) a) Best first search b) Greedy search

c) Uniform cost Search
d) depth limited search
e) Interactive deepening Search
f) Bidirectional search

c) A* Search

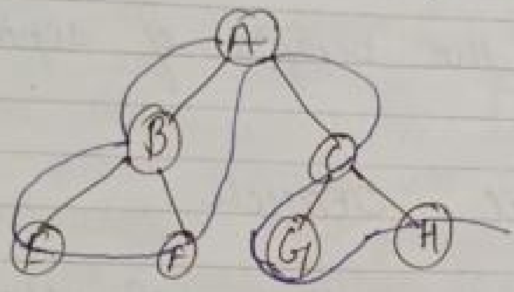
1) Breadth first Search (BFS)

* Root node is expanded first then all the successors of the root node are expanded next, and then their successors and so on.
* It is implemented by using queue (FIFO) (First in first out)
* Every node that is generated must remain in memory.
* Guarante to find the single solution if exist
* The memory requirement are a bigger problem.



2) Depth first Search

* Expand the deepest node in the fringe (branch) of the search tree.
* If dead end occurs backtracking is done, to the next immediate previous node for node to be expanded.
* Implemented by using stack
* Not guaranted to find a solution.
* It trapped into infinite loops.



3) Uniform cost Search

* The root node is expanded first
* Then the next node to be expanded is selected as the lowest cost node on the fringe rather than lowest depth node
* $\boxed{g(n) = \text{path cost}}$

* ~~Bread~~ BFS is equivalent to uniform cost search where $g(n) =$ depth ~~of~~ 0

$$g(n) = DEPTH(n)$$

* Guaranteed to find the single solution at mean minimum path cost
* Only suitable for smallest instance problem

4) Interactive Deepening Search

* It is a general ~~stare~~ strategy that side steps the issue of choosing the best depth limit.
* It combines the benifits of depth fs and Bfs

5) Bidirectional Search

* It is a strategy that simultaneously searchs both the directional.
* forward from the initial state and backward from the goal and stop when the two searches ~~in th~~ meet in the middle.

6) Hill climbing.

* Estimate how far away the ~~g~~ goal is
* It is neither optimal nor complete
* Can be very fast.
* Expand the ~~p~~ cheapest node where the cost is path ~~cog~~ cost g(n)

* Expand the node you think is nearest to the goal where the estimate of distance to goal is h(n)

NOTE → Uninformed search are all to slow for ~~most~~ more real world problem

Heuristic Search (Informed Search)

A heuristic is a function that, when applied to a state, return a number that is an estimate of the ~~merit~~ merit of the state, with respect to the goal. In other words, we can say that it is approximately how far the state is forward from the goal ~~stage~~. state.

i) Best first Search (OR graph)

* Dfs is good – solution is found without all competing branches having two be expanded.
* Bfs is good – it dosen't get trapped ~~in~~ on dead and point.

v.v. imp
* combinging these two is to follow a single path at a time but switch whenever some competing path looks more promising then current path.
* A key component of these algorithm is a heuristic function h(n).

* A ~~kn~~ node is selected for expansion based on an evolution function f(n)

$$\boxed{f(n) = h(n)}$$

(ii) Greedy Best first Search-

* Greedy Best first search try to expand the node that is ~~goo~~ closest to the goal.
* It is not optimal and it is incomplete
* It lead to a solution quickly.
* It evaluates node by using just the heuristic function. $f(n) = h(n)$

vvv Imp

A* Star Search

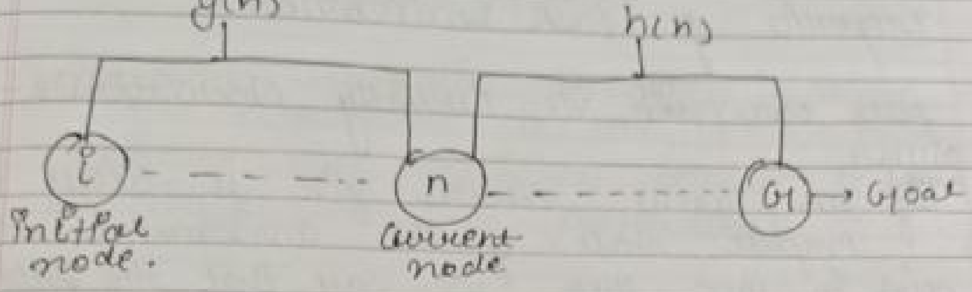
* A* can be used wether we are interested in finding a minimal cost over all path or simply path as quickly as possible
* ~~I~~ The cost to get from the node to the goal $\boxed{f'(n) = g(n) + h'(n)}$ vvImp

$g(n) \rightarrow$ gives the path cost from the start node to node (n)

$h'(n) \rightarrow$ is the estimated cost of the cheapest path from ~~e~~ n to goal.

$f'(n) \rightarrow$ estimated ~~Goo~~ cost of the cheapest solution through (n)

* "A* Search is both complete and optimal"



Role of G →

Imp
* G = 0, we care about getting ~~two~~ to solution somehow (always choose the node that seems ~~closet~~ closest to the goal)

Imp
* G=1, (constant) if we want to find a path with fewest number of steps

NOTE → If we guarantee (h') never over estimate h, then A* guarante to find optimal path to goal determine by G.

* A* is the fastest search algorithm that is, for any given heuristic, No algorithm can expand fewer node than A*.

* A* depends of the quality of the heuristic

Task Environment -

Properties of task Environment.

1) fully observable v/s Partially observable -

fully -
If an agent sensors give it access to the complete state of the environment at each point in time then we say that the ~~fully~~ task environment is fully observable.

If task environment is effectively fully observable if the ~~so~~ sensor detect all ~~aleep~~ aspect that are relevant to the choice of action.

Partially observable → An environment might be partially observable because of noisy and inaccurate sensors or because part of the state are ~~st~~ simply missing from the sensors data.

2) Deterministic v/s Stocastic

If the next ~~sate~~ state of the environment is completly determine by current state and the action executed by the agent then we say that the environment is Deterministic.

Otherwise it is stocastic.

3.) Episodic v/s sequential.

If an episodic task environment the agent ~~expo~~ experienced is divided into atomic episode. each episodes consist of the agent perceiving and then performing a single action.

~~5~~ The next episode doesn't ~~epis~~ depend on the actions taken in ~~a~~ previous episode
for example →

An agent that has to spot defective part on an assembly ~~time~~ line basis each decision on the current part, regardless of previous decision.

Sequential environment → The current decision could effect all ~~fut~~ future decision.
for example →
Chess, taxi driving.

4.) Discrete V/S continous

The discrete / continuous ~~distingui~~ distinction can be applied to the state of the environment to the way time is handled and to percept and action of the agent.
for example → 1) Chess is a discrete set of

percept and action.

2) Taxi driving is a continous state and continuous time problem the speed and ~~tore~~ location of the taxi and of the other vehicles sweep through a range of continuous values and do so smoothly over time

3) ~~Fat~~ Taxi driving action are also continuous (Steering angle).

5.) Single agent v/s Multi agent

An agent solving a crossword ~~pzu~~ puzzle by itself is ~~clea~~ clearly is in a single agent whereas an agent playing chess is in a multi agent environment.

NOTE =)

The most challenging environment are inaccessible, non deterministic, dynamic and continuous.

~~Th~~ An ideal agent always choose the action which maximizes its ~~ele~~ expected performance given its percept sequence so far.

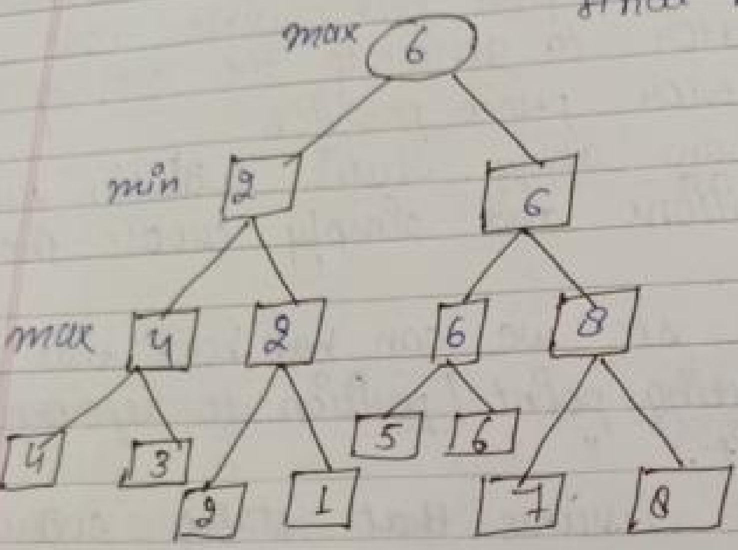
MiniMax Search Procedure

* The minimax search is a depth first search and depth limited procedure.
* The idea is to strat at the ~~st~~ current position. and use ~~poss pot~~ plasible move ~~ao~~ generator to generate the set of possible successor ~~proce~~ position
* Now we can apply static evalution function to those positions and simply choose the best one.
* After doing so we can ~~bak~~ back that value upto the starting point / position to represent our evolution of it.
* Here we assume that static evalution function returns larger values to indicate ~~to~~ good situation for us.
* So our goal is to maximize the value of static evalution function to the next ~~goated~~ board position
* The opponants goal is to minimize the value of the static evalution function.
* The alteration of maximizing and minimizing alternate when evolutions are to be pushed back ~~up~~ correspond to the opposite strategies of the two players. is called minimax

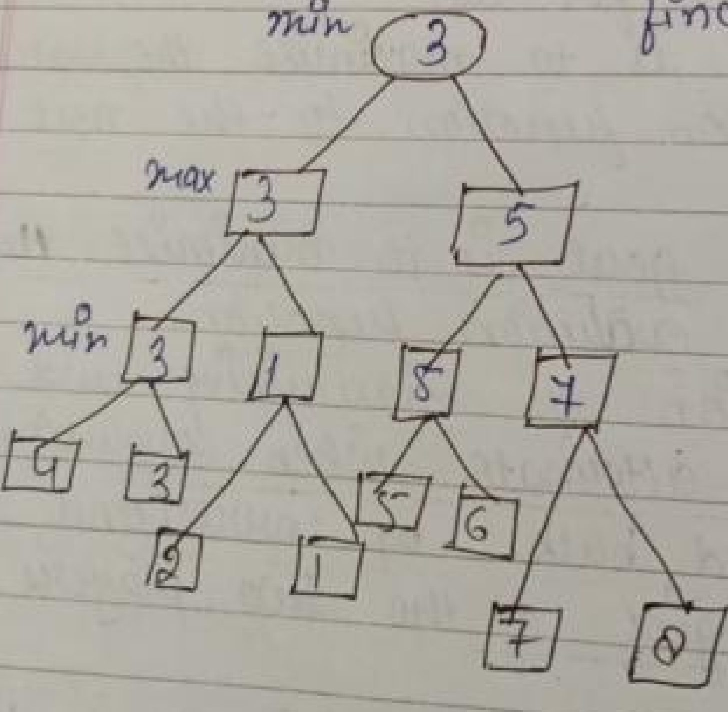
The minimax procedure as it stand does not need to treat maximizing and minimizing level differently. Since it simply negates to slow evaluation each time it changes level.

lower Bound = α = max
upper Bound = β = min.

final max $\Rightarrow$ 6



final min $\Rightarrow$ 3



Search Space

State space search is a problem-solving technique used in AI to find the solution path from the initial state to the goal state by exploring the various states.

The state space search approach searches through all possible states of a problem to find a solution. It is an essential part to artificial intelligence and is used in various applications, from game-playing algorithms to natural language processing.

A State space is a way to mathematically represent a problem by defining all the possible states in which the problem can be. This is used in search algorithms to represent the initial state, goal state, and current state of the problem. Each state in the state space is represented using a set of variables.

The efficiency of the search algorithm greatl[illegible] depends on the size of the ~~st~~ state space and it is important to choose an appropr[illegible] representation and search strategy to sear[illegible] the state space efficiently.

One of the most well-known state space search algorithms is the A* algorithm.

Features of State Space Search-

State space search has several features that makes it an effective problem-solving technique in ~~AI~~ AI. These features include:

- Exhaustiveness :- State space search explores all possible states of a problem to find a solution.

- Completeness -: If a solution exists, state ~~spo~~ space search will find it.

- Optimality -: Searching through a state space results in an optimal solution.

- Uninformed and Informed Search -:

State space search in AI can be classified as uninformed if it provides additional information about the problem.

State Space Representation -: It involves defining an Initial State and a Goal State and then determining a sequence of actions, called states, to follow.

* State - A state can be an initial state, a go goal state or any other possible state that can be generated by applying rules between them.

* Space - In an AI Problem, space refers to the exhaustive collection of all conceivable states.

* Search - This ~~t~~ technique moves from the beginning state to the desired state by applying good rules while traversing the space of all possible states.

* Search Tree - To visualize the search issue, search tree is used, which is a tree-like structure that ~~pr~~ represent the problem. The initial state is represent by the root node of the search tree, which is the starting point of the tree.

* Transition Model - This describe what each action does, while path cost assigns a cost value to each path, an activity sequences that connects the beginning node to the end node. The optimal option has the lowest cost among all alternatives.

Applications of State Space Search -

* State space search algorithms as robotic, game playing, computer networks, operations research, bioinformatics, cryptography and supply chain management. In ~~an~~ AI, state space search algorithms can AI, state space search algorithm can solve problems like pathfinding, planning, scheduling

- They are also useful in planning robot motion and finding the best sequence of actions to achieve a goal. In games, state space search algorithms can ~~be~~ help determine the ~~a~~ best move ~~fo~~ for a player given a particular game state.

- State space ~~a~~ search ~~z~~ algorithms can optimize routing and resource ~~allo~~ allocation in computer networks and operations research.

In Bioinformatics - State space search algorithms can help find patterns in biological data and predict protein structures.

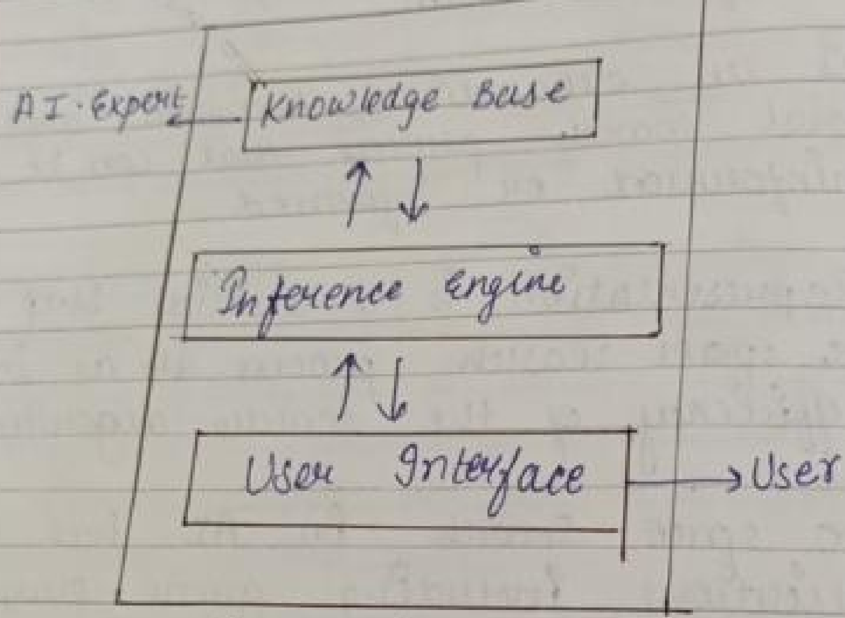
In cryptography, State space search algorithms are used to break codes and find cryptographic keys.

Conclusion.

- State space search is a ~~plo~~ problem-solving technique used ~~to~~ in AI to find a solution to a problem by exploring all possible states of the problem.
- It is an exhaustive, complete, and optimal search process that can be classified as uninformed or informed
- SS Representation is a crucial step ~~in~~ in the state space search process ~~as~~ as it determines the efficiency of the search algorithms.
- State space search in AI has several applications including game-playing ~~algo~~ algorithm, NLP, robotics, planning and computer vision.

UNIT- 3

Expert System – In expert ~~sy~~ system or an intelligent system is a group of instruction that can output a result using the knowledge embedded.



Expert System.

The performance, accuracy of any expert system is interdependent with the accumulation of knowledge which are authentic and specific.

Knowledge can be defined as a set of 3 important elements.

1.) Statistics
2.) information
3.) Previous exposure

knowledge basis can be divided under two categories

1.) Heuristic
2.) Algorithms

1.) Heuristics - It is the ability to draw influences and evaluate data all of which comes with practice. Major portion of knowledge base of an expert system is Heuristic in ~~m~~ nature.

2.) Algorithm knowledge - It is a sort of factual knowledge which is widely ~~a~~ accepted among specialist of a task domain.

Application of Expert System.

- Process control and Monitoring
- Decision management.
- ~~Daiganis~~ Daignosis and trouble and trouble-shooting
- Planning
- configuration of object
- Finanical decision
- knowledge publishing.

Difference between Expert System and human.

* Human expert are perishable but an expert system is permanent

* Expert system consist of knowledge from multiple human expert, so the solution developed ~~o~~ are more efficient

* A cost effective expert system can be developed for medical daignosis ~~a~~ application

* Expert system can solve complex problems by deducing new fact through exiting fact of knowledge, represented mostly as If- then rules rather than through conventional procedural code.

Knowledge Based Agent (KB)

Knowledge Based is a set of representation of fact about the world each individual representation is called sentence. ~~the~~ The sentence are expressed in a language is termed as knowledge representation language.

Knowledge based agent are best understood as agent that know about their world and reason about their ~~cores~~ course of action

Basic concept:

1.) The knowledge based agent → ~~it~~ It is a set of representation of fact about the world.

2.) Knowledge representation language - It is a language whose sentences represents facts about the world.

There ~~must~~ must be way to add new sentences to the knowledge based agent and to query what is known. The standard name for these ~~so~~ task are: to

(i) ~~tell~~ TELL

(ii) ASK.

The fundamental requirement that is imposed on ~~to~~ TELL and ASK is that when one ask a question ~~at~~ of KB agent, the answer should follow from what has been told to knowledge based previously. The agent maintains a knowledge based, which may initially contain some back-~~gou~~ ground knowledge. Each time the agent program is called, it does two things

1) It ~~tell~~ TELLS knowledge based what ~~it~~ is perceives

2.) It ~~AK~~ ASK knowledge based what action it should perform.

conjuction → and ($\wedge$)
disjuction → OR ($\vee$)

Logic- Logic is concerned with reasoning and the validity of arguments. Generally in logic, we are not concerned with the truth of statement but their validity.

For ex → ~~to~~ All lemons are blue
Nary is a lemon

conclusion →

Therefore Nary is blue.

This set of statement is considered to be valid because the conclusion (Nary is blue) follow logically from the other two statement which we often called the ~~pri~~ premises

Why logic is used in AI?

Logic is widely used as a representation ~~meth~~ method of AI. Unlike some other representation logic allow us to easily reason about negatives (This book is not blue) and disjuction (such as He is either a soldier or a ~~a~~ sailor)

imp

Types of Preposition

For eg → ($P \vee \sim P$) is ~~also~~ always true regardless the value of preposition P.

$\wedge \rightarrow$ And
$\vee \rightarrow$ OR
$\rightarrow \rightarrow$ Implies
$\leftrightarrow \rightarrow$ Bidirectional
$\sim \rightarrow$ Negation

P	~P	P ∨ ~P
0	1	1
1	0	1

Tato Tautology

* A preposition that is always true is known tautology

* There are also preposition that are always false such as $(P \wedge \sim P)$. This such a preposition is termed as contradiction

P	~P	P ∧ ~P
0	1	0
1	0	0

* A ~~pres~~ preposition that is neither a tautology nor a contradiction is called contingency

P_1	P_2	$P_1 \cup P_2$
0	0	0
0	1	1
1	0	1
1	1	1

Imp $\Rightarrow$ $P \rightarrow q$
$\sim P \vee q$

[For bidirectional [$\leftrightarrow$]]

$\Rightarrow$ $(P \rightarrow q) \cap (q \rightarrow P)$
$(\sim P \vee q) \cap (\sim q \vee P)$

* A sentence can be formed by combining simple sentence with on of the five logical connectives

- ∧ (AND) → A sentence whose main connective is ∧, such as $P \wedge q$ is known conjunction logic.

- ∨ (OR) → A sentence using ∧ such $P \wedge q$ is a disjunction of the disjoint P and Q

- → (Implies) → A sentence such as $P \rightarrow q$ is known as implication (or conditional). Implications are also known as rules or If-THEN statements.

- ↔ (Equivalent/ Bidirectional) → This sentence $P \rightarrow q$, $(P \rightarrow q) \wedge (q \rightarrow P)$, $(\sim P \vee q) \wedge (\sim q \vee P)$.

- ~ (NOT, Negation, Complement) → A sentence such as $\sim P$ is termed as negation of P.

First Order Predicate Logic

First order logic is formal logical system used in mathematics, linguistic and computer science. First order logic is

$\forall \rightarrow$ forall

distinguish from the ~~prop~~ propositional logic by it's used of quantifier.

First order logic and their primitives are
1.) Variables symbols : x, y
2.) connectives : $\vee, \wedge, \rightarrow, \leftrightarrow, \sim$
3.) Quantifiers : a (universal quantifier), b (existential quantifier)

Quantifiers →

* Universal quantification ~~are~~ correspond to conjuction (and) in this $(\forall x)$ ~~for~~ $P(x)$ means that P hold for all values of x. in the domain associated with that variable.
for eg → $(\forall x)$ dolphins $(x) \longrightarrow$ mammal (x)

* Existential quantification correspond to disjunction (OR) in that $(\exists x) P(x)$ means that P hold for some values of x in the domain associated with that variable.
for eg → $(\exists x)$ mammal $(x) \wedge$ lays egg (x)

* ~~Unif~~ Universal Quantifier are ~~usely~~ usually used with "~~implies~~", to ~~perf~~ form "IF-THEN RULES"
for eg → $(\forall x)$ Students $(x) \longrightarrow$ smart (x)

* ~~Exist~~ Existential quantifier are ~~usual~~ usually used with "and". to specify a list of properties or fact about an individual.

Eg→ $(\forall x)$ Students $(x) \wedge$ Smart(x) (there is some student who is smart)

* Switching the order of universal and existential does change the meaning.

Rule of Inference.

Modus Ponens – $\dfrac{\begin{array}{c}p \\ p \to q\end{array}}{\therefore\ q}$

Modus tollens – $\dfrac{\begin{array}{c}\sim q \\ p \to q\end{array}}{\therefore\ \sim p}$

Hypothetical syllogism – $\dfrac{\begin{array}{c}p \to q \\ q \to r\end{array}}{\therefore\ p \to r}$

Disjunctive syllogism – $\dfrac{\begin{array}{c}p \vee q \\ \sim p\end{array}}{\therefore\ q}$

Addition – $\dfrac{p}{\therefore\ p \vee q}$

simplification – $\dfrac{p \wedge q}{\therefore\ p}$

conjunction – $\dfrac{\begin{array}{c}p \\ q\end{array}}{\therefore\ p \wedge q}$

Resolution – $\dfrac{\begin{array}{c}p \vee q \\ \sim q \vee r\end{array}}{\therefore\ q \vee r}$

1) NOT (~) →

P	~P
0	1
1	0

2) (∧) AND ⇒

P	Q	P ∧ Q
0	0	0
0	1	0
1	0	0
1	1	1

3.) (V) OR ⇒

P	Q	P V Q
0	0	0
0	1	1
1	0	1
1	1	1

4.) → (Implies) ⇒

~P	P	Q	P → Q	~P V Q
1	0	0	1	1
1	0	1	1	1
0	1	0	0	0
0	1	1	1	1

5) (↔) Bidirectional ⇒

P	$\sim P$	Q	$\sim Q$	$\sim P \vee Q$	$\sim Q \vee P$	$(\sim P \vee Q) \wedge (\sim Q \vee P)$	$P \leftrightarrow Q$
0	1	0	1	1	1	1	1
0	1	1	0	1	0	0	0
1	0	0	1	0	1	0	0
1	0	1	0	1	1	1	1

Planning And Learning.

1. Planning- The task of coming up with a sequence of actions that will acheive a goal is known as planning. In planning we see how an agent can take advantage of the structure of a problem to construct complex plan of action.
The Planning problem is artificial intelligence is about the decision making done by intellegent creatures such as robot, human or computer programs while ~~trying~~ tring to achevie a goal.
In includes choosing a sequence of actions that will (with a high likelihood) transform the state of the word, step by step, so as to ~~t~~ satisfy the goal.

Imp NOTE→ Only environment that are fully observable, deterministic, finite, static (change happen only when the agent action) and discrete (in time, action, object and effective) are to be considered, these are called classical planning environment.

In contrast non classical planning is for partially observable, or stochastic environment and involves a different ~~st~~ set of algorithms and agent design.

History.

Planning has a long history in AI. Strong Interaction with logic based knowledge representation and reasoning scheme are the basic planning environment.

- Given→ ~~Stas~~ Start state, goal conditions and actions.
- find- Sequence of actions, leading from Strat to goal

Unit 4

Knowledge Representation

Knowledge Representation and reasoning

Knowledge Representation and reasoning define the limits of how an intelligent agent interacts with its environment. and uses its knowledge to decide what to do wether it is to decide any action at all.

It is a field where we study

- Representation of knowledge and fact about the world
- ~~A~~ Reasoning that can be performed using the aforementioned knowledge.

The goals for representing knowledge in an efficient manner are manifold (several copies). The aim is to achieve a good representation that can . (together)

1.) ~~Asmi~~ Assimilate information /knowledge to be able to provide an . accurate solution to the problem at hand.

2.) be Amenable to efficient computation will being mintable, compact (short) and natural

3.) Be expressive ~~&~~ ~~e~~ enough to incorporate exploitable features of the problem to achieve computational gain

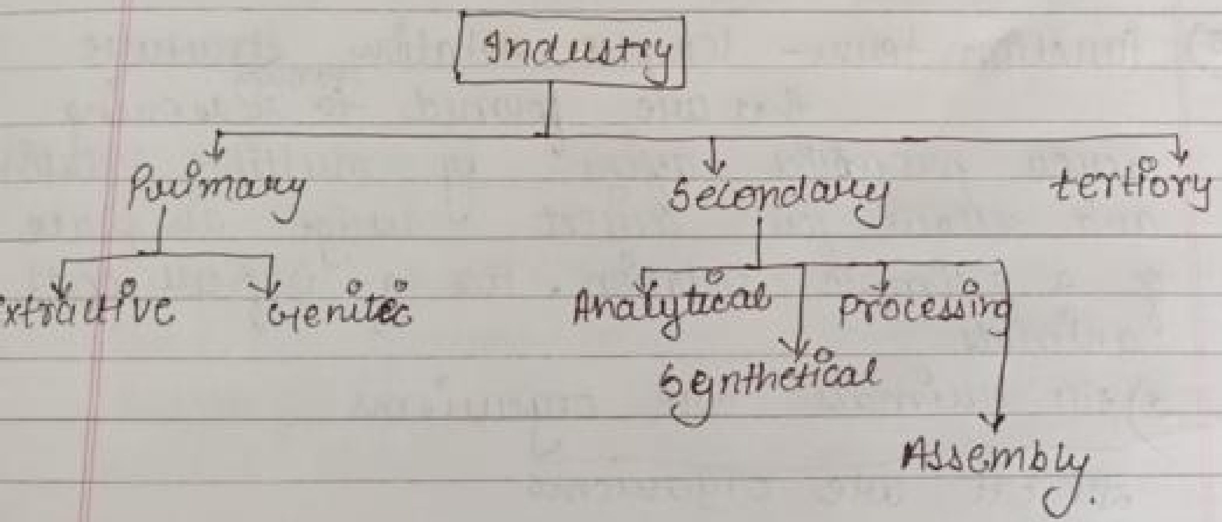
4.) Efficiently trade of computational, complexity and accuracy.

* Imp ⇒ • In the field of AI there exists a multitude of knowledge representation

that can be chosen to allow for more specific and powerful problem solving models to be applied to the same problem.

Ontologies, Object and Events.

Ontologies - In ontological engineering concept are generally represented in a tree like fashion. where the parent node represent ~~the~~ a generalised concept ~~which~~ while the children represent more specific classes. belonging to that parent. This general frame work of the concept is called ontology.



Most of these ontologies have multiple components which are as follows.

1.) Object - Individual components in a problem for eg → bat and ball.

2) Categories - A set or a collection of a similar object. for eg → subjects and vehicle.

3.) Attributes - ~~pr~~ Properties, features, characteristics and parameter an object can have for eg → colour and shape.

4) Relations - Rules that define which object are put into what category. for eg → all cats are animal.

5.) function term - Complex relation structure that are formed through ~~to~~ reasoning across properties define by multiple relations and allow for direct usage in place of a simple relation. for eg → 1) If all cats are animals

2) All animals are organisms

All cat are organisms

6.) Restrictions - ~~forming~~ Formally stated fact and assertion and the condition underwhic assumption hold true. for eg → sampling

7.) Rules - ~~If~~ IF-THEN statement that describe logical inferences that define how ~~a~~ oject of different categories are related for eg→ If the day is sunny, people go ~~to~~ for ~~finish~~ fishing.
This rule link ~~to~~ two categories weather and people and relate them by the relation that state that if a particular weather ~~is~~ occurs, then people perform a particular action.

8.) Axioms - Assertions which comprise the over all theory which a particular ontology describes in the domain of ~~its~~ its applications. They are the ~~t~~ super sets of rules and ~~tt~~ restrictions and define a prior knowledge i.e known and has to be asserted.

9.) Events - Process that change attribute and relations of object. For eg→
"If John goes fishing on a Sunny day", and the person object contain the attribute "hasgonefishing", it will be set to true
Event → fishing Object → John.

* Categories & Object.

Categories - Categories are set and collection of similar object. The collection of object is a component kr of knowledge of representation. Generally, an agent with the world is through the use of object, but most of the reasoning is done by general rules and at categorical level. Categories can be used to make predictions about object

Once rules are either learned or define. this can be done by determining the category membership of some object on which rules are already define and then generalizing these rules to the categorical level. Once this is done the behaviour of the new object is predicted by infering their membership to the classes and then predicting the outcome.

Events - We need to define event in order to be able to put the object in an ~~oto~~ ontology in context of time.

Event are embedded in time and can define different properties and relationship of object at different point in time.

Predicate - ~~Pred~~ Predicate are means to symbolize and event being true at some time so they take time as an argument.