AI Unit 1

→ What is AI?

1] AI in four different aspects / approach:

1] Acting humanly: The Turing Test Approach
2] Thinking humanly: The cognitive modeling approach
3] Thinking rationally: The 'laws of thought' approach.
4] Acting rationally: The rational agent approach.

2]

Think like people	Think rationally
Act like people	Act rationally

3] Thinking humanly is the automation of activities that we associate with human thinking, activities such as decision-making, problem solving, learning, etc.

4] Acting humanly is the art of creating machines that perform functions that require intelligence when performed by people.

5] Acting rationally means AI is concerned with intelligent behaviour in artifacts.

6] Definition:

(i) Artificial intelligence is ~~the~~ an area of computer science that emphasizes the creation of intelligent machines that work and react like humans.

(ii) AI is the intelligence exhibited by machines, rather than the humans or animals.

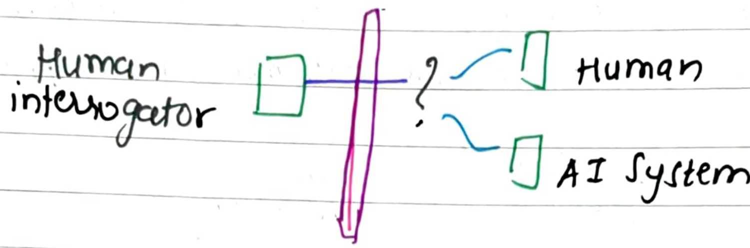
(iii) AI is the technology that enables computers and machines to simulate human learning, comprehension, problem solving, decision making, creativity, autonomy.

→ Acting Humanly : The Turing Test

Turing (1950) 'computer machinery and intelligence'

1] 'can machines think?' → 'can machines behave intelligently?'

2] To answer this, Turing proposed an operational test of intelligence known as the imitation game.



- Here, an interrogator asks questions to the human and machine through text only (so voice / appearance doesn't give it away.
- If the interrogator cannot reliably tell which is the machine and which is the human, then the machine is said to have passed the Turing Test.

3] He predicted that by 2000, a machine will have 30% chance of fooling a lay person for 5 mins.

4] The major AI components suggested by him are: knowledge, reasoning, language, understanding, learning.

5] Total Turing Test also includes computer vision and robotics.

→ Thinking Humanly: Cognitive Science

1] In the 1960s, the 'cognitive revolution' began where information processing psychology replaced the earlier dominance of behaviourism.

2] The goal was to study the inner workings of the human mind using methods like:

Page No.
Date / /

(i) Introspection
(ii) Psychological experiments.
(iii) Brain Imaging.

3] Scientific Theories of brain activity require:
(i) Choosing the correct level of abstraction (eg: knowledge vs. neural circuits).
(ii) Methods of validation such as:
- Predicting and testing human behaviour (top-down)
- Analyzing neurological data directly (bottom-up approach).

4] Both of these approaches (cognitive science and cognitive neuroscience) are separate fields from AI.

5] They share the following characteristics with AI: Despite progress, all three fields still lack theories that explain human-level general intelligence. This indicates a common challenge.

→ Thinking Rationally: Laws of Thought

1] Focus is normative (or prescriptive) rather than descriptive.

2] Aristotle raised the question: what are the correct arguments and thought processes?

3] Ancient greek schools developed forms of logic, creating notations and rules of reasoning for thought. These may or may not have proceeded to the idea of mechanization.

4] The logicist tradition in AI builds on such programs to create intelligent systems.

5] Problems:
(i) Not all intelligent behaviour is meditated by logical deliberation.

(ii) Raises deeper questions like : What is the purpose of thinking? What thoughts should I have of all thoughts that I could have?

→ Acting Rationally

1] Rational behaviour means doing the right thing.
2] The right thing is the action that is expected to maximize goal acheivement, given the available info.
3] It doesn't always require thinking (eg: blinking reflex), but thinking should be in the service of rational action.
4] Aristotle (Nicomachean Ethics) stated:
Every art and every inquiry, and similarly every action and pursuit, is thought to aim at some good.

→ Applications of AI

1] Agriculture:
(i) Use of computer vision to identify weeds and spray herbicide only where needed, reducing chemical use.
(ii) AI image recognition to detect plant diseases, soil nutrient deficiencies, and pest damage from photos.

2] Gaming:
(i) DeepMind developed a chess game that defeated chess champ Garry Kasparov.
(ii) AI-driven NPCs adapt to player strategies instead of following fixed patterns.

3] Space Exploration:
(i) Analyze space data (lots of data) explanets, cosmic events, etc.
(ii) AI is used to identify a solar planet system from

the data of Kepler telescope. or other telescopes.
(iii) Mars 2020 Rover uses AEGIS for performing AI-based investagation on MARS.

4] Self driving vehicles:
(i) Tesla's autopilot and waymo use deep learning, computer vision, and sensors (LiDAR, Radar, cameras) to detect objects, pedestrians, and traffic signals.
(ii) AI algorithms handle path planning, obstacle avoidance, and adaptive cruise control to ensure safe driving.

5] Education:
(i) AI is used for personalised recommendation algorithm to adapt learning paths to each student.
(ii) Automated grading system ~~used~~ uses NLP to check essays and provide instant feedback.
~~(iii)~~ (iii) AI tutors use chatbots and speech recognition.

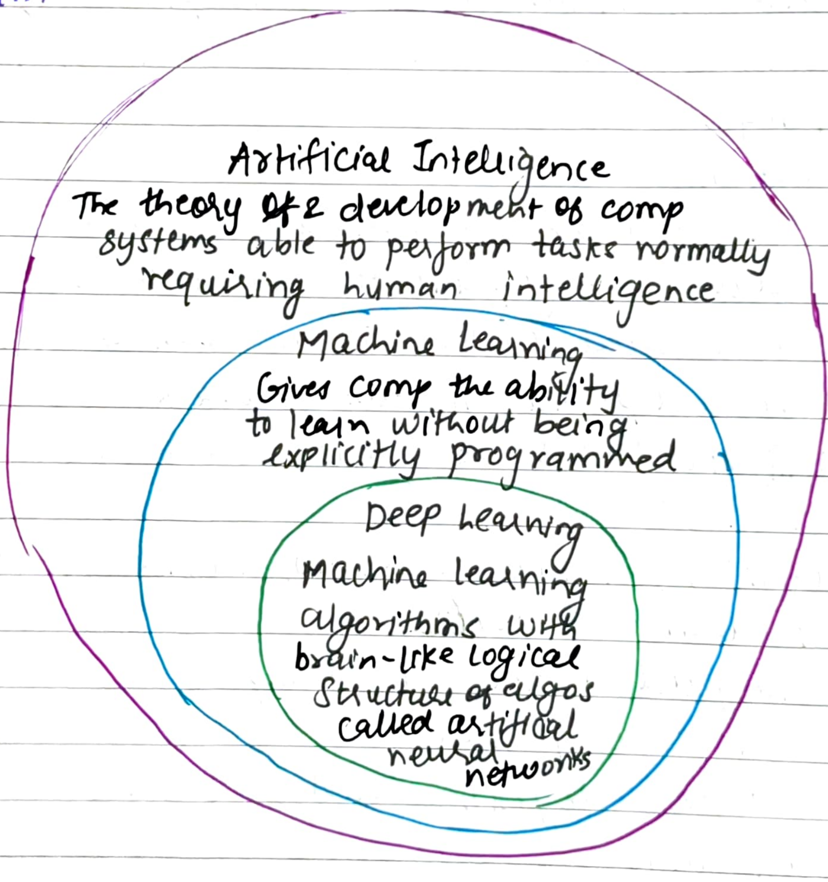
6] Business:
(i) Chat bots, virtual assistants, business intelligence bots
(ii) Predictive analytics
(iii) Targeted online advertising
(iv) Voice recognition
(v) Pattern recognition and recommendation using clustering.

7] Healthcare:
(i) Telemedicine
(ii) Nanorobots for operations
(iii) Medical chatbots
(iv) Image processing for detection of diseases
(v) Medical data analysis for prediction
(vi) Automated prescription reading app.

8] In everyday life:
(i) Virtual assistants (Alexa, Siri) use speech recognition and NLP for answering queries, set reminders, etc.
(ii) Google Translate uses neural machine translation.

9] (iii) Smart Homes, Cities, Hotels and Industries:
(i) Traffic, waste management, etc.

→ Diagram



→ Machine Learning

1] Machine learning is a class of algorithms which is data driven i.e. unlike normal algorithms, it is the data that tells what the good answer is.

2] Getting computers to program themselves and also teaching them to make decisions using data

3] 'Where writing software is bottleneck, let the data do the work instead.'

4] Features:

(i) It uses the data to detect patterns in a dataset and adjust program actions accordingly.

(ii) It focuses on the development of computer programs that can teach themselves to grow and change when exposed to new data.

(iii) It enables the computer to find hidden insights using iterative algorithms without being explicitly programmed.

(iv) ML is a method of data analysis that automates analytical model building.

5] (i) Traditional computer:

Data → Computer → Output
Program → Computer

(ii) Machine Learning

Data → Computer ⇄ Model
Program → Computer

- Learn from data
- Find hidden insights
- Train & Grow

6] ML Lifecycle:

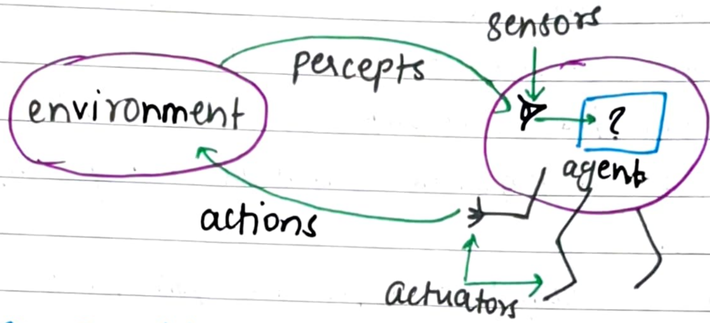
Step 1: Collect data
Step 2: Data Wrangling
Step 3: Analyze Data
Step 4: Train Algorithm
Step 5: Test Algorithm
Step 6: Deployment

7] Agents

1] An agent is anything that can be viewed as:
(i) perceiving its environment through sensors.
(ii) acting upon that environment through actuators.

2] Assumption: Every agent can percieve its own actions (but not always the effects).

3]



4] Human agent:
(i) Sensors: Eyes, ears, other organs.
(ii) Actuators: Hands, legs, mouth, other body parts.

5] Robotic agent:
(i) Sensors: Cameras and infrared range finders.
(ii) Actuators: various motors

6] A software agent:
(i) Sensors: Keystrokes, file contents, recieved network packages.
(ii) Actuators: Displays on screens, files, sent network packets.

Agents and Environments

1] Percept: Agent's perceptual input at any given instant.
2] Percept sequence: complete history of everything the agent has perceived.
3] An agent's choice of action at any given instant can depend on the entire percept sequence observed to date.

4] An agent's behaviour is described by the agent function which maps from percept histories to actions:

$$f : P^* \rightarrow A$$

5] The agent function describes any given agent (external characterization).

6] Internally, the agent function will be implemented by an agent program which runs on the physical architecture to produce f.

agent = architecture + program.

→ Rational Agents

1] An agent must strive to do the right thing, based on what it can percieve and the actions it can perform.

2] The right action is the one that will cause the agent to be most successful.

3] Performance measure is an objective criterion for success of an agent's behaviour.

4] Eg: performance measure for a vacuum cleaner can be the amount of dirt cleaned up, amount of time taken, amt of electricity consumed, amt of noise generated. etc.

5] As a general rule, it is better to design performance measures according to what one actually wants in the environment rather than according to how one thinks the agent should behave (amount of dirt cleaned vs. a clean floor).

6] A more suitable measure would reward the agent for having a clean floor.

→ Rationality

1] Rationality at any given time depends on:
(i) The performance measure that defines the criterion for success.
(ii) The agent's prior knowledge of the environment.
(iii) The actions that the agent can perform.
(iv) The agent's percept sequence to date.

2] Rational Agent: For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

3] Rationality is different from omniscience (all-knowing with infinite knowledge).

4] Rationality maximizes expected performance while perfection maximizes actual performance.

5] Agents can perform actions in order to modify future percepts so as to obtain useful information (information gathering, exploration).

6] An agent is autonomous if its behaviour is determined by its own experience (with the ability to learn and adapt).

→ Specifying the Task Environment (PEAS)

1] PEAS:
(i) Performance measurement
(ii) Environment
(iii) Actuators
(iv) Sensors

2) In designing an agent, the first step must be to

specity the task environment (PEAS) as fully as possible.

→ Environment Types

1] Fully observable vs. Partially observable:

1] An environment is fully observable if its sensors give it access to the complete state of the environment at any point in time.

2] These are convinient because the agent need not maintain any internal state to keep track of the world.

3] An environment may be partially observable because of noisy and inaccurate sensors or because parts of the state are simply missing from the sensor data.

4] Examples: vacuum cleaner with local dirt sensor, taxi driver.

2] Deterministic vs. Stochastic:

1] The environment is deterministic if the next state of the environment is completely determined by the current state and the action executed by the agent.

2] In principle, an agent need not worry about uncertainty in a fully observable, deterministic environment.

3] If the environment is partially observable then it could appear stochastic

4] Eg: vacuum world is deterministic while taxi driver is not.

5] If the environment is deterministic except for the actions of other agents, then it is strategic.

EcoNāma Page No
Date / /

3] Episodic vs. Sequential

1] In episodic environments, the agent's experience is divided into atomic episodes (each episode ~~contains~~ consists of the agent perceiving and then performing a single action), and the choice of action in each episode depends only on the episode itself.
2] Eg: classification tasks.
3] In sequential environment, the current decision could affect the future decisions.
4] Eg: chess and taxi driver.

4] Static vs. Dynamic:

1] Static environment is unchanged while an agent is deliberating.
2] These are easy to deal with as the agent need not keep looking at the world while it is deciding on the action ~~or~~ need to worry about the passing time.
3] Dynamic environments continuously ask the agent what it wants to do
4] The environment is semi-dynamic if ~~it~~ the env itself does not change with the passage of time but the agent's performance score does.
5] Eg: Taxi driving → dynamic
Chess played with a timer → semi dynamic
crossword puzzles → static

5] Discrete vs. Continuous

1] A limited number of distinct, clearly defined states, percepts and actions.
2] Eg: Chess has a finite number of discrete states,

and has discrete set of percepts. and actions. Taxi driving has continuous states and actions.

6] Single Agent vs. Multi Agent Environment :

1] An agent operating by itself in an env is single agent.
2] Eg: Crossword is single agent while chess is two-agents.
3] Agent A should treat another entity B as an agent if B is maximizing its own performance measure. and that performance depends on A's actions.
4] If not, then A must treat B as a stochastic object.
5] Eg: Chess → competitive multiagent env.
Taxi driving → partially cooperative multiagent env.

[Refer Table from PPT].

7] The env type largely determines the agent design.
8] The real world is partially observable, stochastic, sequential, dynamic, continuous, multi-agent.

→ Agent functions and programs

1] Agent Program :
1] Takes the current percept as input from the sensors.
2] Return an action to the actuators.
3] While agent function takes the whole percept history, agent program takes just the current percept as input which is the only available input from the environment.
4] The agent needs to remember the whole percept

EcoNāma Page No.
Date / /

sequence, if it needs it.

→ Table-lookup agent

1] A trivial agent program: Keeps track of the percept sequence and then uses it to index into a table of actions to decide what to do.

2] The designers must construct the table that contains the appropriate action for every possible percept sequence.

3] function TABLE-DRIVEN-AGENT (percept) returns an action
static: percepts, a sequence, initially empty
table, a table of actions, indexed by percept sequences, initially fully specified
append percept to the end of percepts
action ← LOOKUP (percepts, table)
return action.

4) Drawbacks:
(i) Huge Table (P^T, P: set of possible percepts, T: lifetime)
(ii) Space to store the table
(iii) Takes a long time to build the table
(iv) No autonomy
(v) Even with learning, need a long time to learn the table entries.

→ Agent Types

1] Rather than a table how we can produce rational behaviour from a small amount of code

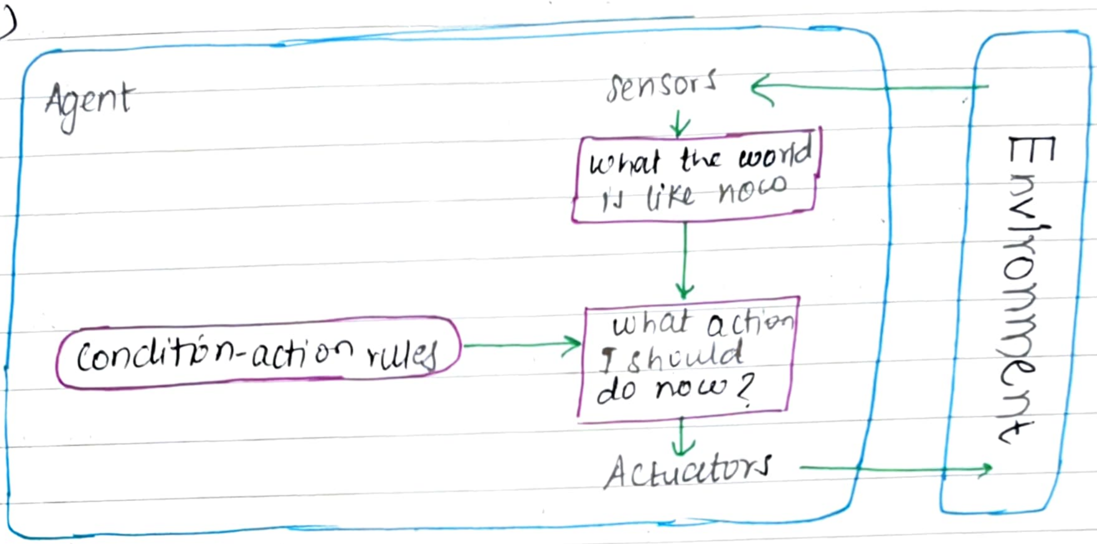
2] Four basic types in order of increasing generality:
1] Simplex reflex agents:
(i) Selects actions on the basis of the current percept

ignoring the rest of the percept history.

(ii) Eg: simple reflex vacuum cleaner agent.

(iii) They use condition-action-rule.

(iv)



(v) They are simple, but they turn out to be of very limited intelligence.

(vi) The agent will work only if the correct decision can be made on the basis of current percept - that is only if the environment is fully observable.

(vii) Infinite loops are often unavoidable - escape could be possible by randomizing.

(viii) function SIMPLE-REFLEX-AGENT (percept) returns an action

```
    static: rules, a set of if condition-action rules
    state ← INTERPRET_INPUT (percept)
    rule ← RULE_MATCH (state, rules)
    action ← RULE_ACTION (rule)
    return action
```

2] Model-based reflex agents

(i) The agent should keep track of the world it can't see now.

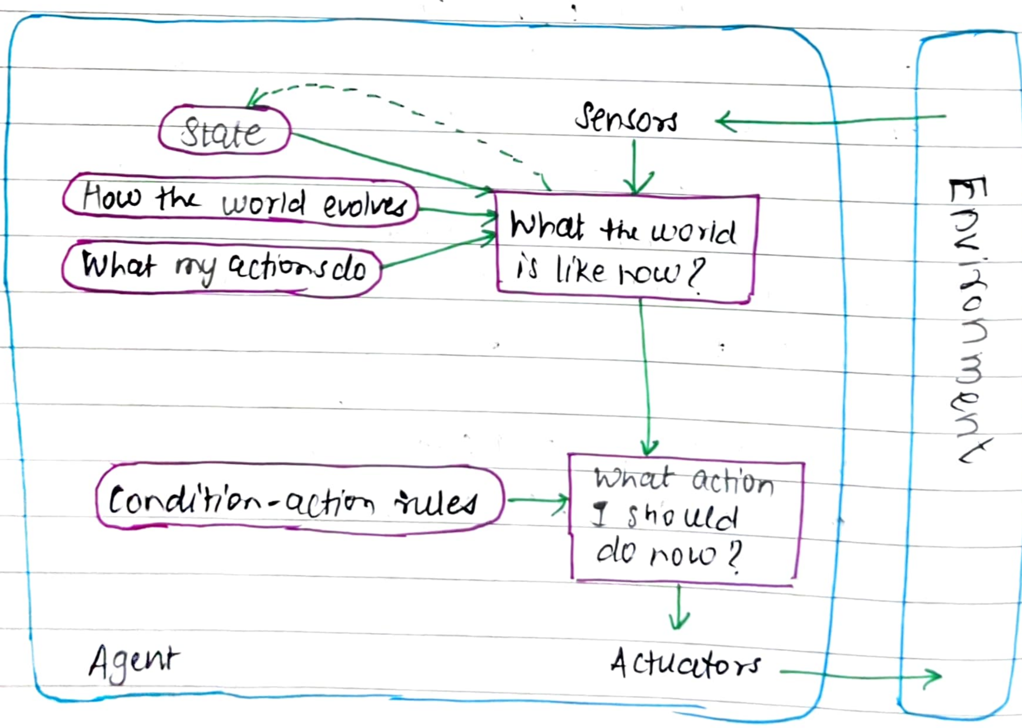
(ii) The agent should maintain some sort of internal state that depends on the percept history and reflects atleast some of the unobserved aspects of the current state.

(iii) Updating the internal state information as time goes by requires two kinds of knowledge to be encoded in the agent program:

- Information on how the world evolves independently of the agent.
- Information about how the agent's own actions affects the world.

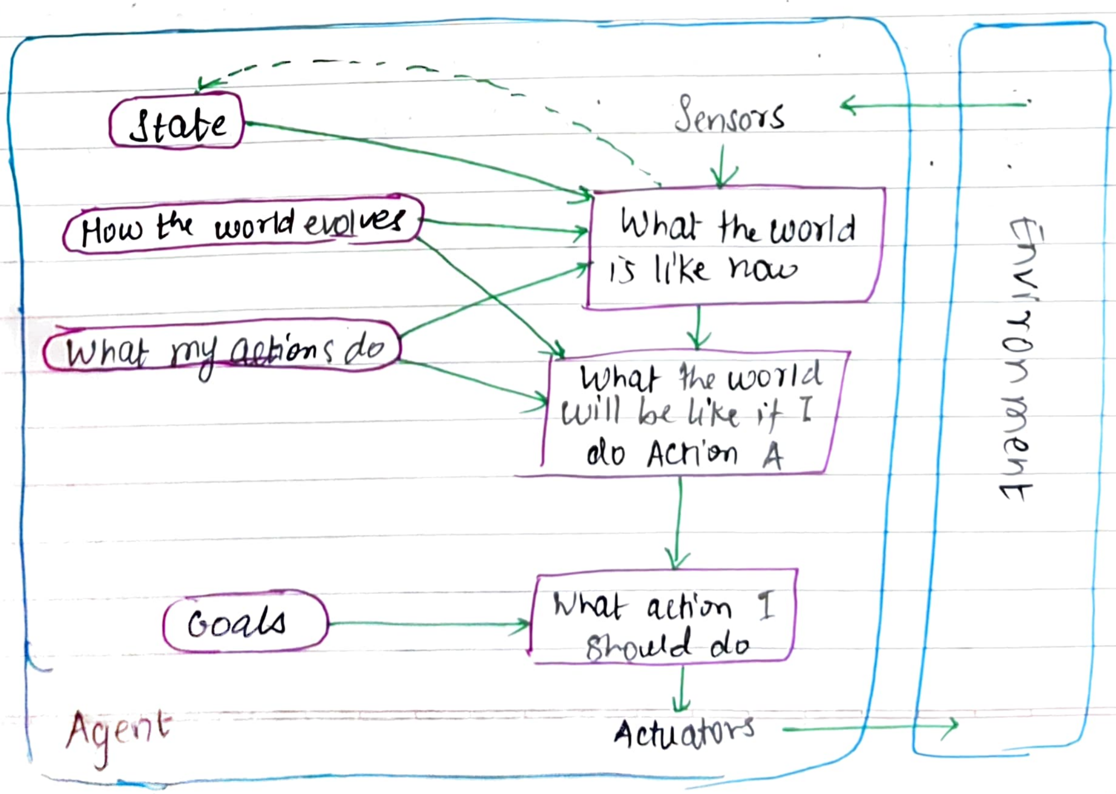
(iv) Model of the world- model based agents.

(v)



(vi) function REFLEX-AGENT-WITH-STATE (percept) returns action
static: state, a desc of the current world
rules, a set of condition-action rules
action, the most recent action, initially none
state ← UPDATED_INPUT (state, action, percept)
rule ← RULE_MATCH (state, rules)
action ← RULE_ACTION [rule]
return action

3] Goal-based agents
(i) Knowing about the current state of the env is not always enough to decide what to do
(ii) The agent needs some sort of goal information that describes the situations that are desirable.
(iii) The agent program can combine this with information about the results of possible actions in order to choose actions that achieve the goal.
(iv) usually requires search and planning.
(v)



EcoNāma Page No.
Date / /

→ Goal-based agents vs. Reflex-based agents

1] Although goal based agents appear less efficient, it is more flexible because the knowledge that supports its decision is represented 'explicitly' and can be modified.

2] On the other hand, for the reflex-agent, we would have to rewrite many condition-action rules.

3] The goal based agent's behaviour can easily be changed.

4] The reflex agent's rules must be changed for a new situation.

→ Utility-based agents

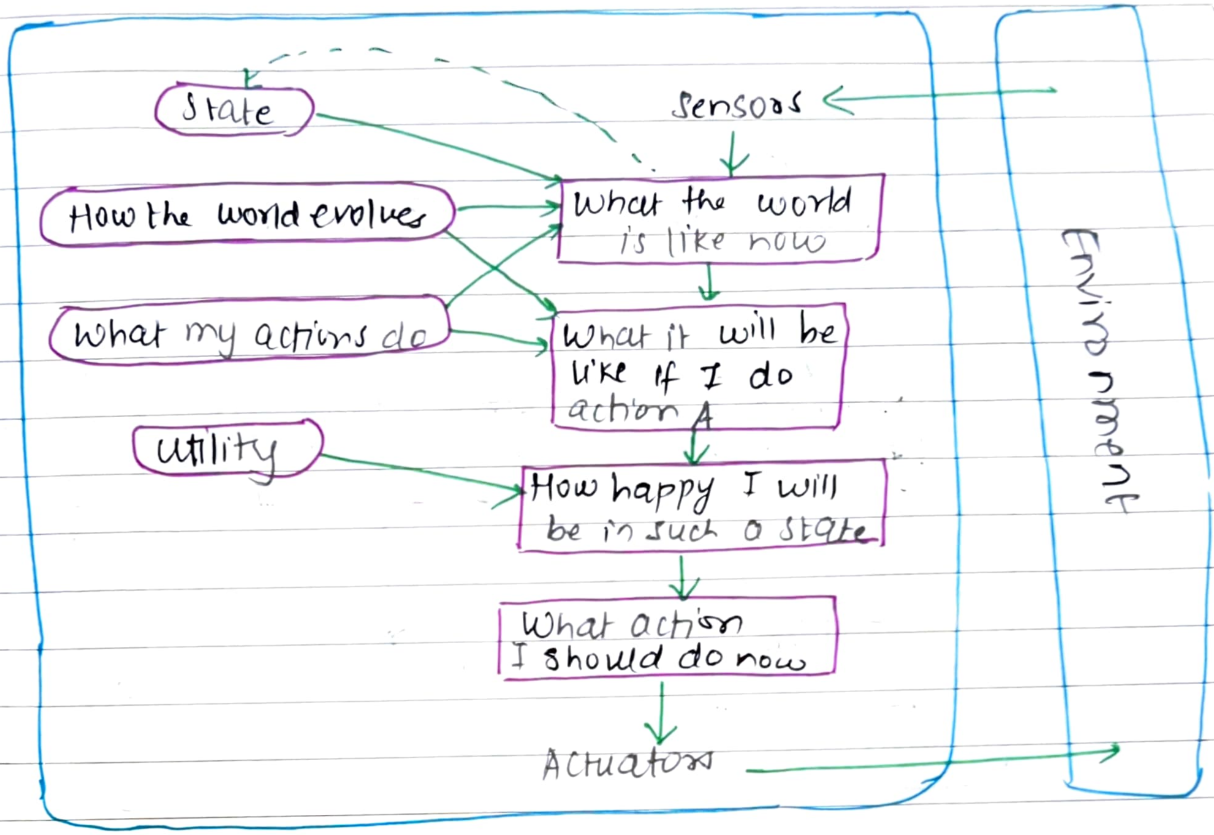
1] Goals alone are not really enough to generate high quality behavior in most envs, they just provide a binary distinction between happy and unhappy states.

2] A more general performance measure should allow a comparison of different world states according to exactly how happy they would make the agent if they could be achieved.

3] Happy- utility (the quality of being useful)

4] A utility function maps a state onto a real number which describes the associated degree of happiness

5] Diagram :

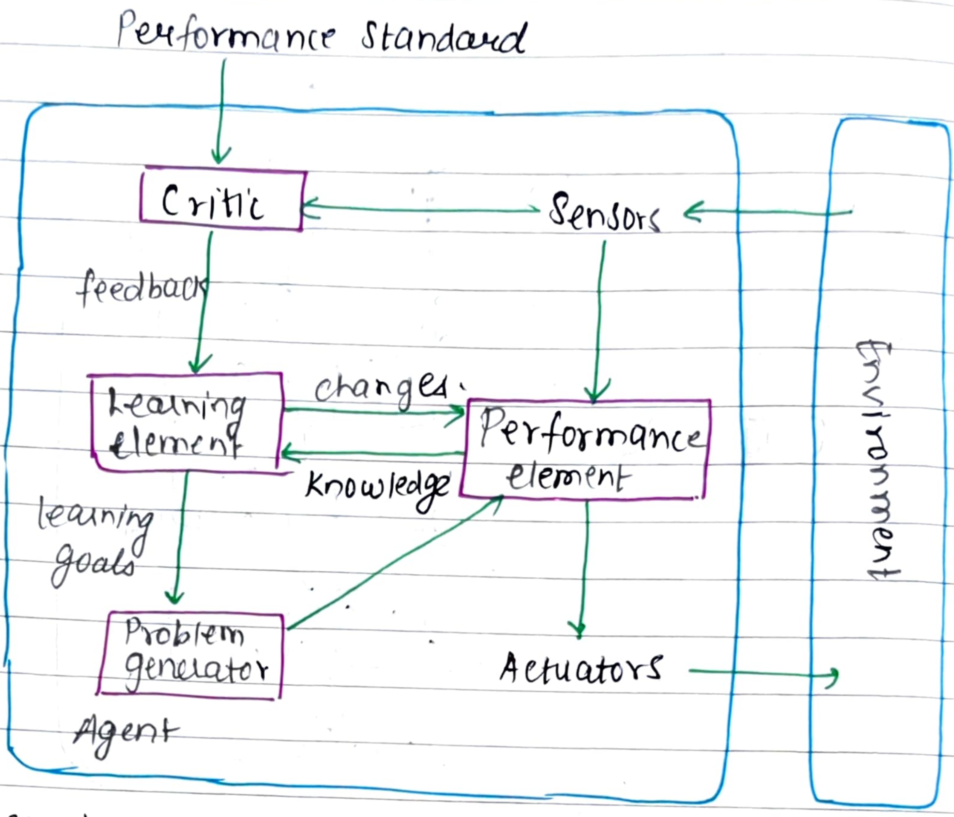


→ Learning Agents

1] Turing - instead of ~~building~~ actually programming intelligent machines by hand, which is too much work, build learning machines and then teach them.

2] Learning also allows the agent to operate in initially unknown environments and to become more competent than its initial knowledge alone might allow.

3] Diagram:



4] Learning element is responsible for making improvements.

5] Performance elements is responsible for selecting external actions (it is what we had defined as the entire agent before).

6] Learning element uses feedback from the critic on how the agent is doing and determines how the performance element should be modified to do better in the future.

7] Problem generator is responsible for suggesting actions that will lead to a new and informative experiences