Fundamentals of Artificial Intelligence & Cyber Security

By
Sunil Kumar S,
HoD, Dept. of AI & ML,
MITE, Moodabidri

MITE - Institute Vision & Mission

VISION

"To attain perfection in providing **Globally Competitive Quality Education** to all our Students and also benefit the global community by using our strength in **Research and Development**"

MISSION

"To establish world class educational institutions in their respective domains, which shall be **centers of excellence** in their Stated and Implied sense. To achieve this objective we dedicate ourselves to meet the Challenges of becoming **Visionary and Realistic, Sensitive and Demanding, Innovative and Practical and Theoretical and Pragmatic**; All at the same time"

Bloom's Taxonomy

Bloom's taxonomy is a set of three hierarchical models used for classification of educational learning objectives into levels of complexity and specificity. The three lists cover the learning objectives in cognitive, affective and psychomotor domains. The cognitive domain list has been the primary focus of most traditional education and is frequently used to structure curriculum learning objectives, assessments and activities.

The models were named after Benjamin Bloom, who chaired the committee of educators that devised the taxonomy. He also edited the first volume of the standard text, *Taxonomy of Educational Objectives: The Classification of Educational Goals.*[1][2]

Cognitive domain (knowledge-based) [edit]

In the 1956 original version of the taxonomy, the cognitive domain is broken into the six levels of objectives listed below.[10] In the 2001 revised edition of Bloom's taxonomy, the levels have slightly different names and their order was revised: Remember, Understand, Apply, Analyze, Evaluate, and Create (rather than Synthesize).[9][11]

Bloom's Taxonomy

Bloom’s Taxonomy

Level	Description	Example
Knowledge	Knowledge involves recognizing or remembering facts, terms, basic concepts, or answers without necessarily understanding what they mean. Some characteristics may include: • Knowledge of specifics—terminology, specific facts • Knowledge of ways and means of dealing with specifics—conventions, trends and sequences, classifications and categories • Knowledge of the universals and abstractions in a field—principles and generalizations, theories and structures	Name three common varieties of apple.
Comprehension	Comprehension involves demonstrating an understanding of facts and ideas by organizing, summarizing, translating, generalizing, giving descriptions, and stating the main ideas.	Summarize the identifying characteristics of a Golden Delicious apple and a Granny Smith apple.
Application	Application involves using acquired knowledge to solve problems in new situations. This involves applying acquired knowledge, facts, techniques and rules. Learners should be able to use prior knowledge to solve problems, identify connections and relationships and how they apply in new situations.	Would apples prevent scurvy, a disease caused by a deficiency in vitamin C?
Analysis	Analysis involves examining and breaking information into component parts, determining how the parts relate to one another, identifying motives or causes, making inferences, and finding evidence to support generalizations. Its characteristics include: • Analysis of elements • Analysis of relationships • Analysis of organization	Compare and contrast four ways of serving foods made with apples and examine which ones have the highest health benefits.

Bloom's Taxonomy

Synthesis	Synthesis involves building a structure or pattern from diverse elements; it also refers to the act of putting parts together to form a whole or bringing pieces of information together to form a new meaning. Its characteristics include: • Production of a unique communication • Production of a plan, or proposed set of operations • Derivation of a set of abstract relations	Convert an "unhealthy" recipe for apple pie to a "healthy" recipe by replacing your choice of ingredients. Argue for the health benefits of using the ingredients you chose versus the original ones.
Evaluation	Evaluation involves presenting and defending opinions by making judgments about information, the validity of ideas, or quality of work based on a set of criteria. Its characteristics include: • Judgments in terms of internal evidence • Judgments in terms of external criteria	Which kinds of apples are suitable for baking a pie, and why?

Continuous Internal Evaluation (CIE)

- There shall be two mandatory written tests of a duration of 1.5 Hours each
- Each written test shall be conducted for 50 Marks and scaled down to 15 marks
- Marks Distribution of Integrated Course

Sl. No.	Evaluation Method	Marks	Weightage (%)
1	Written Test-1	15	30
2	Written Test-2	15	30
3	Additional assessment: shall be done through any of the two activities suggested above	20	40
	Total	50	100

- In addition to mandatory tests, assessment through assignments, group discussions, case studies, quizzes, seminars, mini-projects, laboratory experiments, etc. as deemed fit by the Course Instructor will be carried out.
- an additional assessment conducted through any of the activities suggested above will be conducted for 50 marks and scaled down to 10 (for both activities) respectively

Semester End Examination (SEE):

- SEE shall be conducted at the end of a semester as per the academic calendar
- This shall be conducted for a duration of 3 hours for a 3 credit course
- SEE for all the courses shall be conducted for 100 marks and the marks secured shall be scaled down to 50
- CIE and SEE are assigned equal (50:50) weightage. A student's performance in coursework shall be judged by taking into account the results of both CIE and SEE individually and together
- Standards for Passing (Absolute Grading)

Evaluation Method	Passing Standard
Continuous Internal Evaluation (CIE)	Score: ≥ 40%
Semester End Examination (SEE)	Score: ≥ 40%
Overall Score for passing (CIE+SEE) ≥ 40%	

- The student shall obtain a minimum of 40% of the marks allotted for CIE in each course to be eligible to appear for the SEE in that course.

FUNDAMENTALS OF AI & CYBER SECURITY			
Semester	I/II	CIE Marks	50
Course Code	23ESCC106	SEE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	Exam Hrs	03
Total Hours	40	Credits	03

Course Learning Objectives: This course intends

1. To provide an insight about basic concepts of Artificial Intelligence and workflow of Intelligent Agents.
2. To impart knowledge of machine learning and its applications.
3. To familiarize Cyber Crimes and Security mechanisms.

Module 1 : Overview of Artificial Intelligence	No. of Hrs: 08
Introduction to AI, Automating Intelligence, Man Vs Computer, Cognitive Science to Computer Modeling, Application Areas of AI, Comparison of Conventional and AI Computing, Intelligent Agents,	
Pedagogy	Chalk and board, Active Learning

Module 2: Problem Solving and Search	No. of Hrs: 08
Problem Solving by intelligent agents; Problem Formulation, State Space Representation, Search Problems: Playing Chess, 8-Puzzle,Water Jug Problem, Problem Reduction, Production Systems, 8-Puzzle Production System.	
Pedagogy	Chalk and board, Active Learning

Module 3: Introduction to Machine Learning	No. of Hrs: 08
Introduction to Human Learning, Types of Human Learning, Machine Learning, Types of Machine Learning, Non-Machine Learning Problems, Applications of Machine Learning.	
Pedagogy	Chalk and board, Active Learning

Module 4: Introduction to Cyberspace and Cyber Crime	No. of Hrs: 08
Defining Cyberspace, Overview Internet and World Wide Web, Introduction to Cyber-Crime, Cyber-crime Classification: Phishing, Email Spoofing, Credit card fraud, Password cracking, Cyber-stalking, Social Engineering, Virus and Trojan Horse.	
Pedagogy	Chalk and board, Case studies

Module 5: Introduction to Cyber Security	No. of Hrs: 08
Cyber security: Need and Importance, CIA Triad, Types of security: Application security, endpoint security, mobile security, data security, Authentication: Authentication factors, two factor authentication, multi-factor authentication, Firewall: hardware and software firewalls.	
Pedagogy	Chalk and board, Case studies

List of Activities:

1. Use Case based discussion of AI application in various domains
2. Use Case based discussion on recent Cyber Attacks, Threats and Data Breaches (2022-2023)

Course Outcomes:

students will be able to:

CO1. Understand the workflow of agents in Artificial Intelligence and design intelligent agent for problem solving.

CO2. Illustrate the concepts of Machine Learning, Applications and its advantages over human learning.

CO3. Analyze the cybercrimes on digital platform and articulate defense mechanism for cyber attacks.

REFERENCE BOOKS

1. Dr. Munesh Chandra Triveni, "A Classical Approach to Artificial Intelligence", 2nd Edition Reprint 2019.
2. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Appriach", 3rd Edition, 2010
3. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, "Machine Learning", Pearson Education India, 2018.
4. Anand Shinde, " Introduction to Cyber Security Guide to the World of Cyber Security", notion press, 2021
5. Sunit Belapure and Nina Godbole, "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", First Edition, Wiley India Pvt. Ltd., 2022.

Program Outcomes

Engineering Graduates will be able to:

Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

Program Outcomes

The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Overview of Artificial Intelligence

- Artificial Intelligence is a broad field
  - Different things to different people
  - Multiple definitions are there by different people
- Main Objective of AI is
  - Computers to do tasks that require human intelligence
- People want to automate human intelligence to following reasons
  - Understand human intelligence better
  - Smarter Programs
  - Useful technique for solving difficult problem
- To a common man Artificial Intelligence are two words
  - **Artificial** : Made as copy of something natural
  - **Intelligence** : Ability to gain and apply knowledge and skills

Overview of Artificial Intelligence

- List of Task that requires Intelligence
  - Speech generation & understanding
  - Pattern Recognition
  - Mathematical Theorem proving
  - Reasoning
  - Motion in obstacle filled space

Reasoning in AI refers to deriving new information from existing information using logical rules and principles. AI systems use Reasoning to make inferences, draw conclusions, and solve problems. Reasoning in AI aims to create machines that can reason like humans, using logic, common sense, and intuition.

Intelligence Vs Artificial Intelligence

	Intelligence	Artificial Intelligence
1.	Natural	Programmed by human beings
2.	Increases with experience and also hereditary.	Nothing called hereditary but systems do learn from experience.
3.	Highly refined and no electricity from outside is required to generate output. Rather knowledge is good for intelligence.	It is in computer system and we need electrical energy to get output. Knowledge base is required to generate output.
4.	No one is an expert. We can always get better solution from another human being.	Expert systems are made which have the capability of many individual person's experiences and ideas.
5.	Intelligence increases by supervi- sed or unsupervised teaching.	We can increase AI's capabilities by other means apart from supervised and unsupervised teaching.

Supervised Ex – Apple / Mango Classification
Unsupervised Ex – Grouping the students based on their IA Marks

Supervised + Unsupervised + Self Learning + NLP + Generative AI Ex

Strong AI Vs Weak AI

- **Weak AI** – Helping people in their intelligence task
- **Strong AI** – Doing intelligence task on its own

The confusion about the word "intelligence", its ill definition and much broad sphere has led people to divide AI into two classes:

(i) Strong AI (ii) Weak AI

Strong AI makes the bold claim that computers can be made to think on a level atleast equal to humans. Strong AI research deals with the creation of some form of computer-based artificial intelligence that can truly reason and solve problems. People advocating strong AI believe that it will eventually lead to computers whose intelligence will greatly exceed than that of human beings. In strong AI the programs are themselves the explanations.

Weak AI simply states that some "thinking-like" features can be added to computers to make them more useful tool. Weak AI research deals with the creation of some form of computer-based artificial intelligence which can reason and solve problems in a limited domain. Hence, such a machine would act in some ways as if it was intelligent but it would not possess true intelligence.

Definition of AI

- "The art of creating machines that perform functions that require intelligence when performed by people" (Kurzweil, 1990).
- "The branch of computer science that is concerned with the automation of intelligent behaviour". (Luger and Stublefield, 1993)

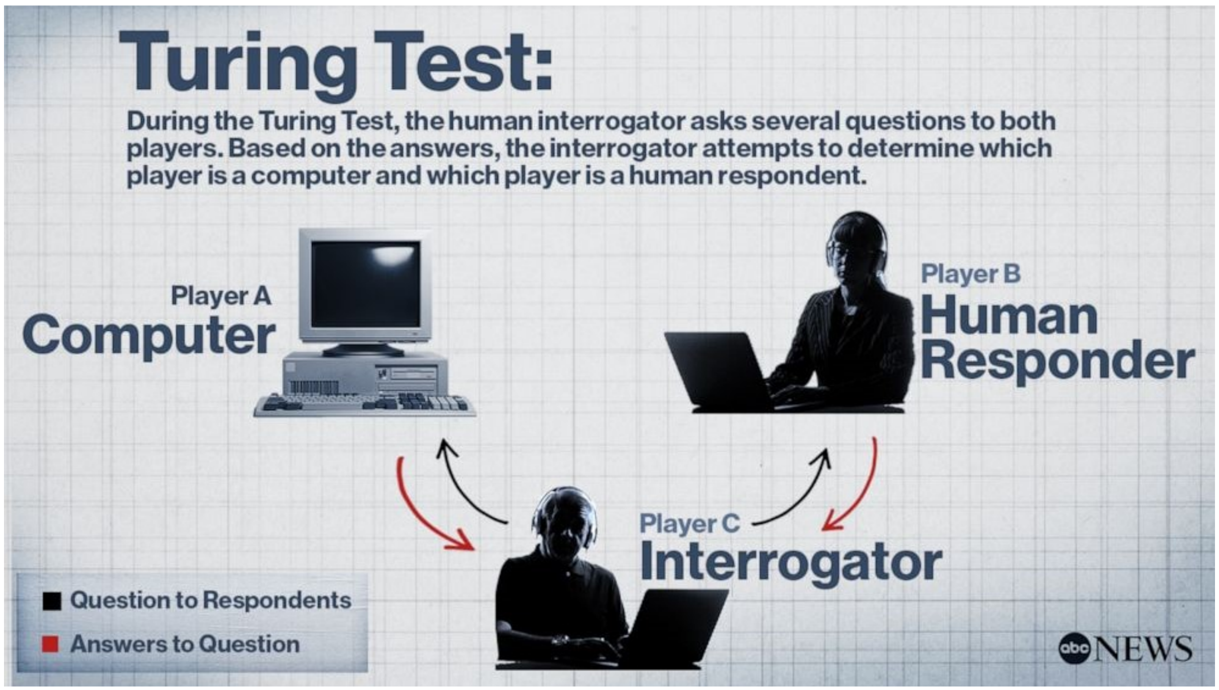
Aftificial Intelligence have the following properties—

Systems that think like humans.	Systems that think/raationally.
Systems that act like humans.	Systems that act rationally.

The Loebner Prize was an annual competition in artificial intelligence that awarded prizes to the computer programs considered by the judges to be the most human-like

Acting Humanly – Turing Test

- The **Turing test** was developed by Alan Turing(A computer scientist) in 1950.
- He proposed that the "Turing test is used to determine whether or not a computer(machine) can think intelligently like humans"?
- The Turing Test is a widely used measure of a machine's ability to demonstrate human-like intelligence.

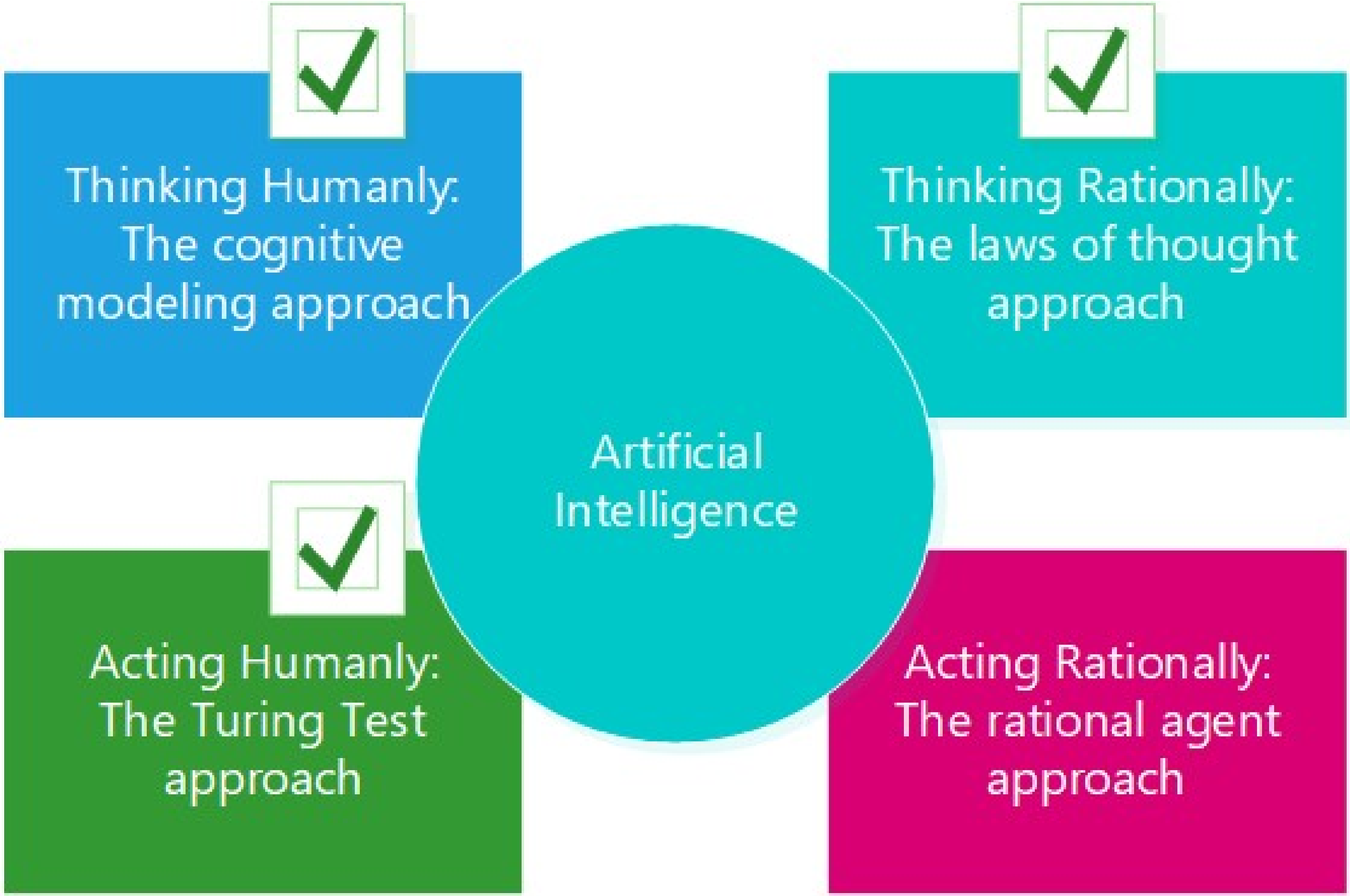


The basic idea of the Turing Test is simple:

•a human judge engages in a text-based conversation with both a human and a machine, and then decides which of the two they believe to be a human.

•If the judge is **unable to distinguish** between the **human and the machine** based on the conversation, then the machine is said to have passed the Turing Test.

Thinking in AI



Rational - based on or in accordance with reason or logic.

Thinking Humanly : Cognitive Modeling

- Once we gather enough data, we can create a model to simulate the human process.
- This model can be used to create software that can think like humans.
- Of course this is easier said than done! All we care about is the output of the program given a particular input.
- If the program behaves in a way that matches human behavior, then we can say that humans have a similar thinking mechanism.
- Within computer science, there is a field of study called **Cognitive Modeling** that deals with simulating the human thinking process.
- It tries to understand how humans solve problems.
- It takes the mental processes that go into this problem solving process and turns it into a software model.
- This model can then be used to simulate human behavior.
- Cognitive modeling is used in a variety of AI applications such as deep learning, expert systems, Natural Language Processing, robotics, and so on.

(ii) Thinking Humanly : Cognitive Modelling

- Method must not just exhibit behaviour sufficient to fool a human judge but must do it in a way demonstrably analogous to human congnition.
- Requires detailed matching of computer behaviour and timing to detailed measurements of human subjects gathered in psychlogical experiments.
- **Cognitive Science** : Interdisiplinary field (AI, psychology, linguistics, philosophy, anthropology) that tries to form computational theories of human cognition.

Thinking Rationally : Laws of Thought

(iii) Thinking Rationally : Laws of Thought

- Formalize "correct" reasoning using a mathematical model (e.g. of deductive reasoning).
- **Logicist Program**. Encode knowledge in formal logical statements and use mathematical deduction to perform reasoning :

Problems:

- Formalizing common sense knowledge is difficult.
- General deductive inference is comutationally intractable.

Idealized or "right" way of thinking

Logic: patterns of argument that always yield correct conclusions when supplied with correct premises

– "Tom is a man; all men are mortal; therefore Tom is mortal."

Acting Rationally : Rational Agents

(iv) Acting Rationally : Rational Agents

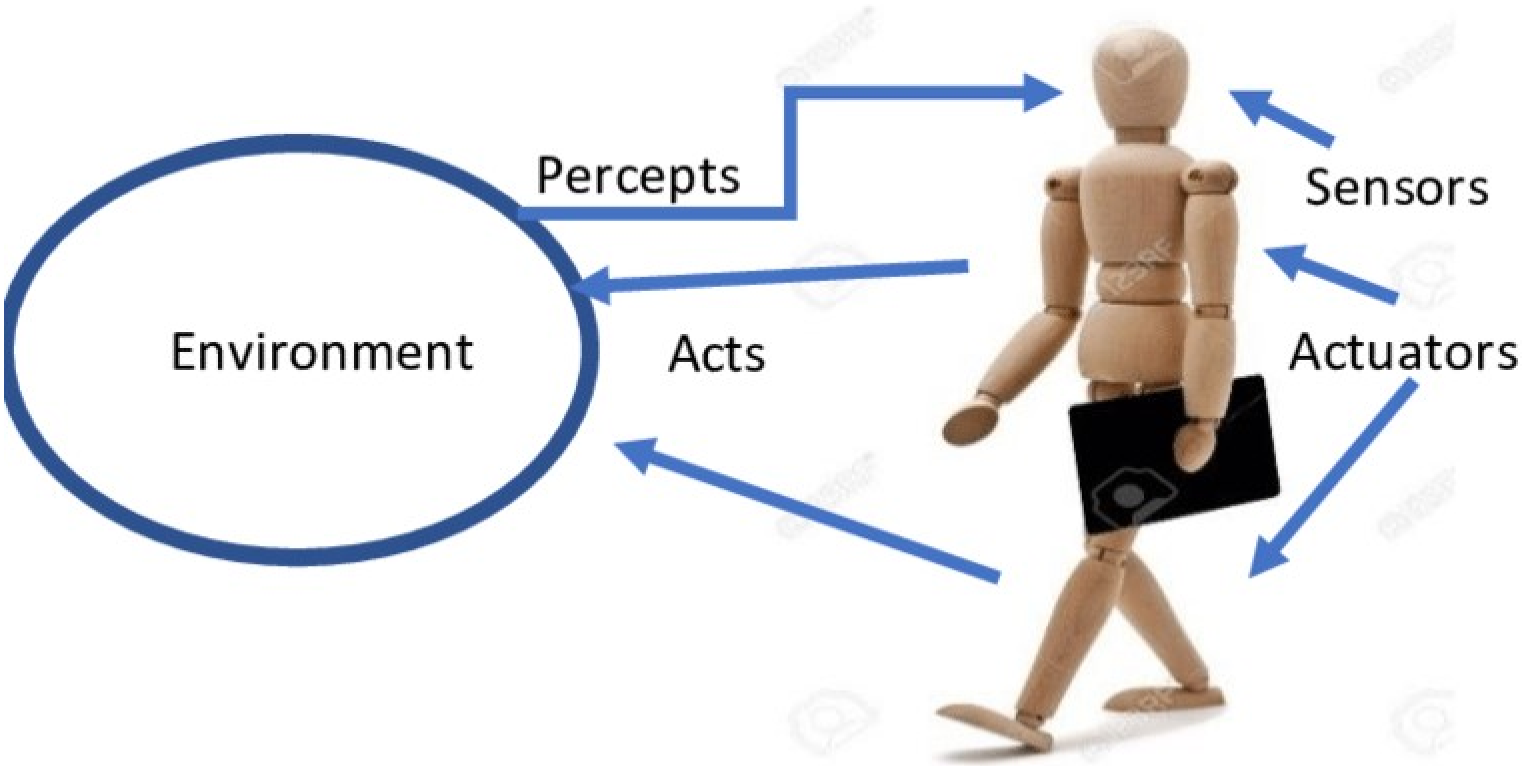
- An agent is an entity that perceives its environment and is able to excute actions to change it.
- Agents have inherent goals that they want to achieve (e.g. survive, reproduce).
- A rational agent acts in a way to maximize the achievement of its goals.
- True maximization of goals requires omniscience and unlimited computational abilities.
- Limited rationality involved maximizing goals within the computational and other resources available

Claim: "Rational" means more than just logically justified. It also means "doing the right thing"

Omniscience - the state of knowing everything.

Rational agents do the best they can given their resources

Acting Rationally : Rational Agents



Foundations of AI

Psychology

Psychology is the study of mind and behavior in humans and non-humans. **Psychology** includes the study of conscious and unconscious phenomena, ...

Philosophy is a systematic study of general and fundamental questions concerning topics like existence, reason, knowledge, value, mind, and language.

Foundations of AI

- Many older disciplines contribute to a foundation for artificial intelligence.
  - Philosophy; logic, philosophy of mind, philosophy of science, philosophy of mathematics
  - Mathematics; logic, probability theory, theory ofcomputability
  - Psychology; behaviorism, cognitive psychology
  - Computer Science & Engineering; hardware, algorithms, computational complexity theory
  - Linguistics; theory of grammar, syntax, semantics

philosophy

/fɪˈlɒsəfi/

noun

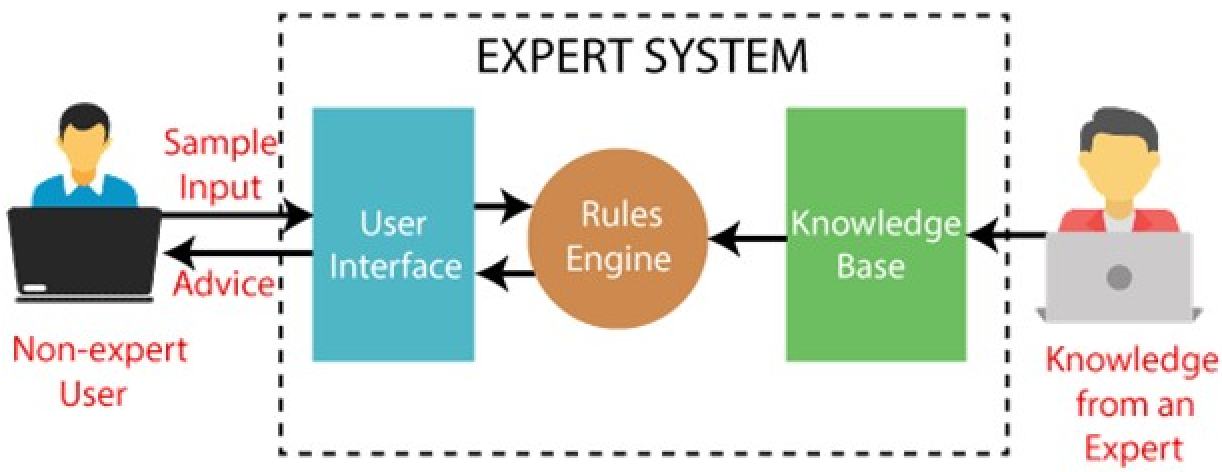
1. the study of the fundamental nature of knowledge, reality, and existence, especially when considered as an academic discipline.
2. a theory or attitude that acts as a guiding principle for behaviour.
   "don't expect anything and you won't be disappointed, that's my philosophy"

Expert Systems

Expert Systems

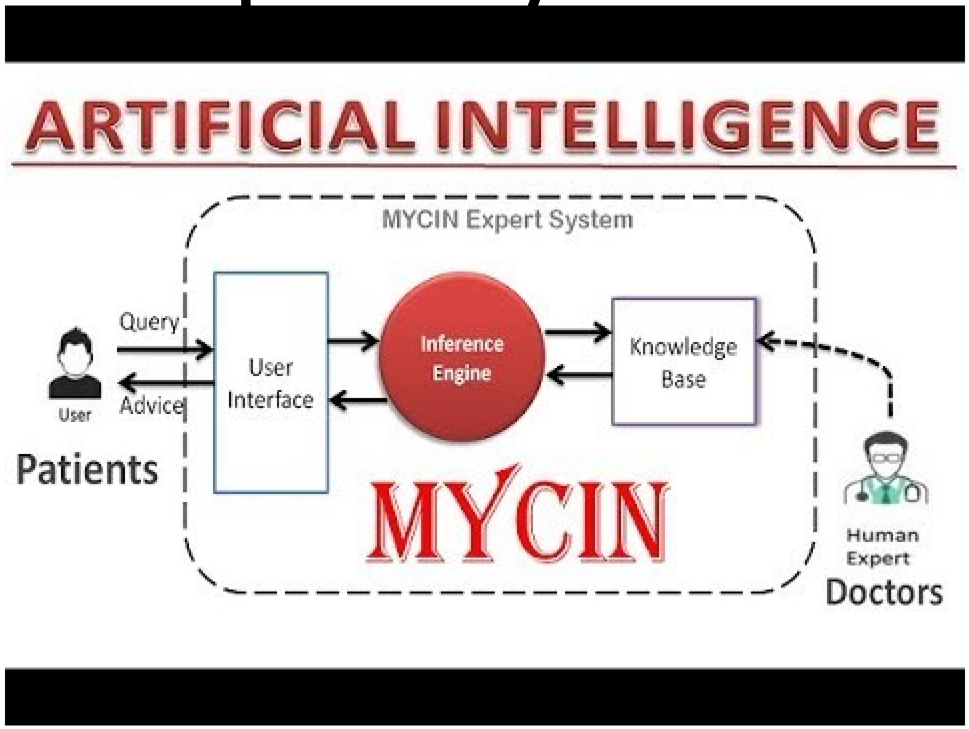
- Discovery that detailed knowledge of the specific domain can help control search and lead to expert level performance for restricted tasks.
- First expert system DENDRAL for interpreting mass spectrogram data to determine molecular structure by Buchanan, Feigenbaum, and Lederberg (1969)
- Early expert systems developed for other tasks;
  - MYCIN: diagnosis of bacterialinfection (1975)
  - PROSPECTOR: Found molybendum deposit based on geological data (1979)
  - R1: Configure computers for DEC (1982)

R1/XCON is an early manufacturing expert system that automatically selects and orders computer components based on customer specifications.



Expert systems	Flight-tracking systems, clinical systems

Expert Systems



MYCIN was a backward chaining expert system which used **artificial intelligence**.
It used this to identify the cause of serious infections and to recommend which antibiotics should be used to treat it.
It could also recommend the correct dosage based on the weight of the patient.
It could also be used to identify and diagnose several blood clotting diseases.
MYCIN operated using a base of approximately 600 rules and various yes/no questions.
It was then able to identify the bacteria based on the high or low probability of each type being responsible.

Dendral

What is an Expert System??

- In artificial intelligence, an expert system is a computer system that emulates the decision making ability of a human expert.

- The first expert systems were created in the 1970s and then proliferated in the 1980s.

What is Dendral??

- **Dendral** was an influential pioneer project in artificial intelligence (AI) of the 1960s, and the computer software expert system that it produced.
- The name Dendral is a portmanteau of the term "Dendritic Algorithm"
- It was written in Lisp (programming language), which was considered the language of AI because of its flexibility

Why it was needed?

- Its primary aim was to help organic chemists in identifying unknown organic molecules, by analyzing their **mass spectra** and using knowledge of chemistry

Mass Spectrum:

- A mass spectrum is a plot of the ion signal as a function of the mass to charge ratio.

The mass-to-charge ratio is a physical quantity relating the mass and the electric charge of a given particle

AI in industry

AI Industry

- Development of numerous expert systems in early eighties.
- Estimated $2 billion industry by 1988.
- Japanese start "Fifth Generation" project in 1981 to build intelligent computers based on Prolog logic programming.
- MCC established in Austin in 1984 to counter japanese project.
- Limitations become apparent, prediction of AI Winter
  - Brittleness and domain specificity
  - knowledge acquisition bottleneck

Fifth Generation Computer Systems

Microelectronics and Computer Technology Corporation

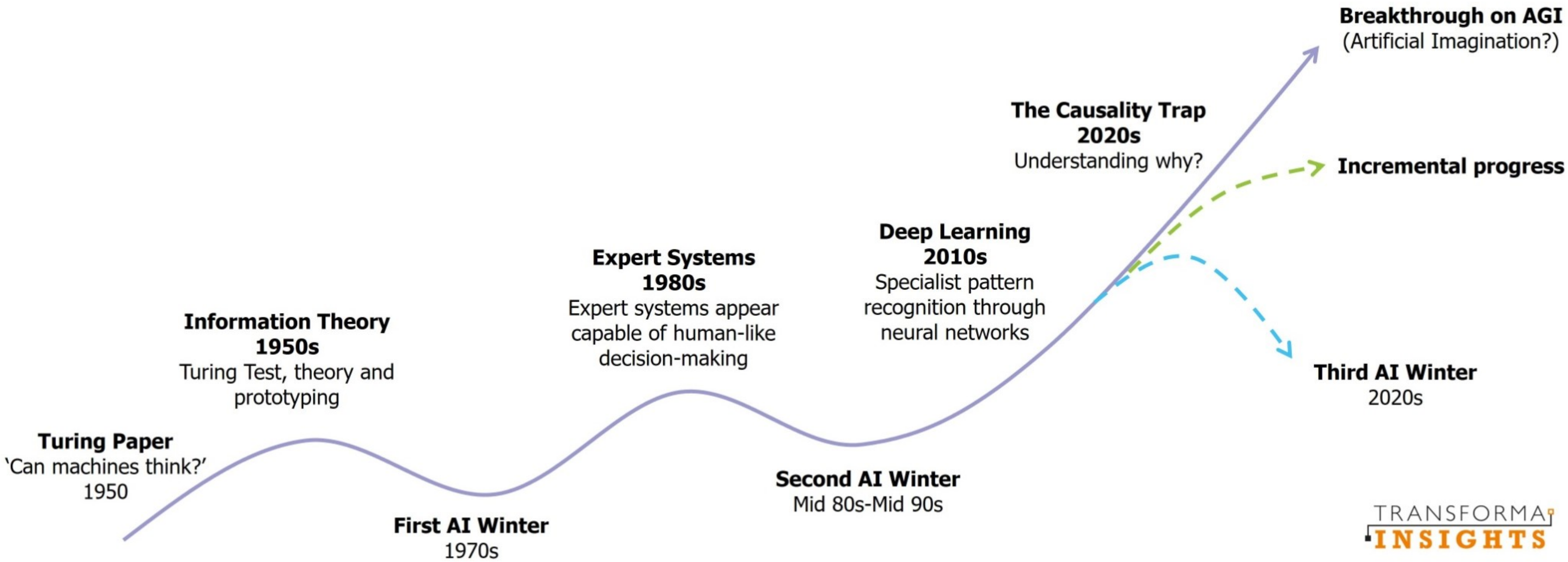
AI winter

https://members.womencorporatedirectors.org/WCD/News/JAN-Feb2020/Reality%20Light.pdf

AI Winter

- Problems with automotive computer vision have been cited as contributing factors in many fatal Tesla crashes (Risen 2016, Crowe 2016) and the death of a pedestrian in an Uber self-driving car accident (Griggs and Wakabayashi 2018).
- Despite years of promises by many companies of full-self driving powered by AI, many companies have walked back their claims in attempt to recalibrate the public’s and funders’ expectations (Bubbers 2019, Elias 2019).

AI Winter



Breakthrough on AGI
(Artificial Imagination?)
The Causality Trap
2020s
Understanding why?
Incremental progress
Deep Learning
2010s
Specialist pattern
recognition through
neural networks
Expert Systems
1980s
Expert systems appear
capable of human-like
decision-making
Information Theory
1950s
Turing Test, theory and
prototyping
Third AI Winter
2020s
Turing Paper
'Can machines think?'
1950
Second AI Winter
Mid 80s-Mid 90s
First AI Winter
1970s
TRANSFORMA
INSIGHTS

Brittleness

- Algorithm brittleness occurs when the environment changes in such a way that the computer vision algorithm can no longer recognize the object due to some small perturbation.
- Brittleness for driverless car computer vision includes an inability to cope with changes caused by weather conditions.
- While important advancements have been made in the last 10 years in computer vision and in the deep learning algorithms that underpin these systems, such approaches to **developing perceptual models** of the real world are plagued by **problems of brittleness**.
- Brittleness occurs when any algorithm cannot generalize or adapt to conditions outside a narrow set of assumptions.
- For example, many natural language processing algorithms are brittle when they can understand a person from New York City but fail to understand the same sentence from someone in Appalachia or who speaks English with a foreign accent (Harwell 2018).

Is automating intelligence is possible ?

- Answer lies in
  - Turing Test
  - Yes – its possible

Man Vs Computer

- AI is the study of how to make computers do things which, at the moment, people do better (Rich and Knight)
- Second definition

AI is the part of computer science concerned with designing intelligent computer systems which exhibit the characteristics we associate with intelligence in human behaviour (Barr and Edward, 1981-82).

- Role of computer changes from something useful to essential

Man Vs Computer

- What computers do better than people?
  - Numerical computation
  - Information Storage
  - Repetitive operations
- What People can do better than computers?
  - People have outperformed computers in activities which involve intelligence.
  - We do not just process information.
  - We understand it , make sense out of what we see and hear
  - Then we come out new ideas

List of characteristics intelligence should possess

Whom to save in accidents

- To respond to situations very flexibly

Bottle fallen on floor vs chair

- Make sense out of ambiguous or contradictory messages

aged person falling

- To attach relative importance to different elements of a situation
- To find similarities between situations despite differences which may separate them
- To draw distinctions between situations despite similarities which may link them. The two situations may look similar on the surface, yet we are able to note the difference and hence adjust our reaction.

Cleaning floor containing both dust & gold ring

Though all these abilities come under common sense, yet these abilities cannot be simulated by the computer.

Now consider some activities such as :

- What did you eat in a friend's marriage? You cannot enlist the mental steps required to remember what you ate in the marriage.
- What are muscular contractions necessary to pick up a cup of tea?
- Can we describe the processes of reading and understanding a book?

The research done by cognitive scientists (scientists who study how human beings learn, reason, store knowledge and use it) helps to explain the workings of human intelligence. This, in turn, has helped workers in the field of AI to simulate that intelligence on a computer.

AI technique for mapping human knowledge

AI technique is a method that exploits knowledge. Workers in AI used many different techniques to make computers more intelligent.

- One commonly used technique is to determine the process used by humans to produce a particular type of intelligent behaviour and then to simulate that process on a computer.
- The other technique which is used by cognitive scientists is to determine those processes which produce human intelligence in a given situation. These processes may be programmed, in an attempt to simulate that behaviour. This AI technique is called *modeling* or *simulation*. (In fact, a model of intelligent human behaviour is an effort to simulate that behaviour on a computer to determine if the computer will exhibit the same intelligent behaviour as does a human.)

Three important AI techniques are :

(1) *Search*—It provides a way of solving problems for which no more direct approach is available as well as a framework into which any direct techniques that are available can be utilized. **Find solution by searching**

(2) *Use of knowledge*—It provides a way of solving problems by exploiting the structures of the objects that are involved. **Example of Mango vs Apple Identification**

(3) *Abstraction*—It provides a way of separating important features and variations from the many unimportant ones. **Cat vs Dog Classification**

Cognitive science to computer modeling

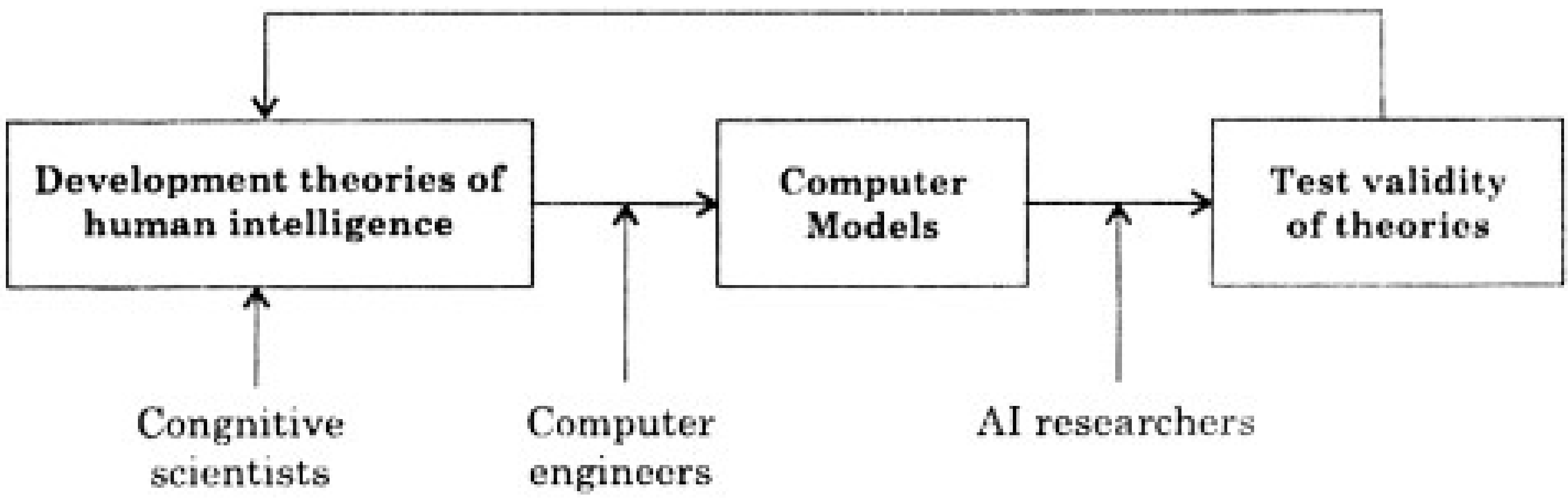


Fig. (1). Relationship between cognitive scientists computer engineers and AI researchers.

Cognitive science to computer modeling

AI is that branch of computer science which deals with symbolic, non algorithmic methods of problem solving. (Buchanen and Shortliffe, 1984)

This definition focuses on two different characteristics of computer programs:

(1) *Numeric vs Symbolic* : Computers were initially designed to process numbers. Consistent researches have shown that people think symbolically rather than numerically and human intelligence is partially based on our mental ability to manipulate symbols, rather than just numbers.

(2) *Algorithmic vs Non-algorithmic* : An algorithm is a step by-step procedure with well-defined starting and ending points, which is guaranteed to reach a solution to a specific problem. Computer architecture readily lends itself to this step-by-step approach since the conventional computer programs are based on algorithms. However, most of the human processes, tend to be non-algorithmic, *i.e.*, our mental activities consist of more than just following logical, step-by-step procedures.

AI research continues to be devoted to symbolic and non-algorithmic processing techniques in an attempt to emulate more closely human reasoning processes by a computer.

Application Areas of AI

- General Problem Solving
  - Water jug, TSP, Tower of Hanoi etc.

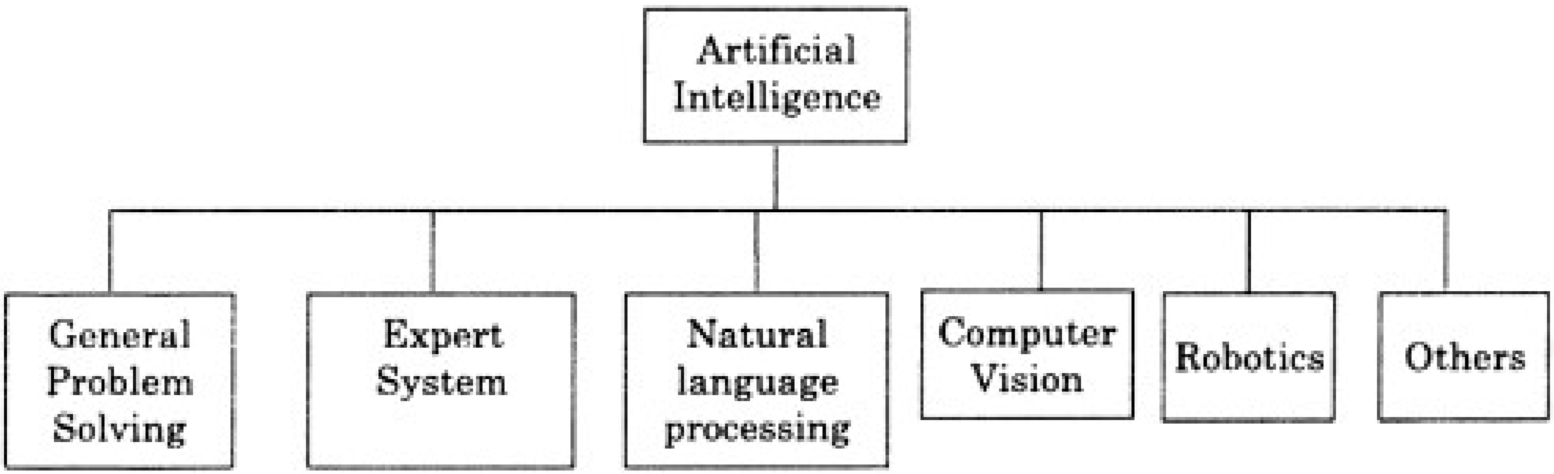


Fig. (5) Application areas of AI.

Application Areas of AI

- Expert Systems

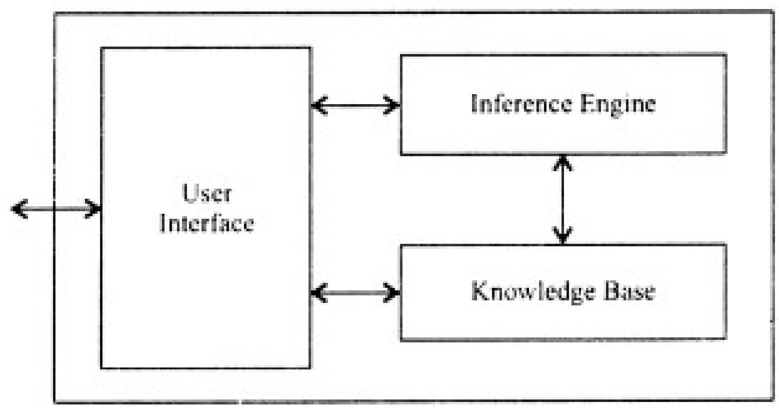


Fig. (6). Block diagram showing structure of an Expert System.

Application Areas of AI

- Natural Language Processing

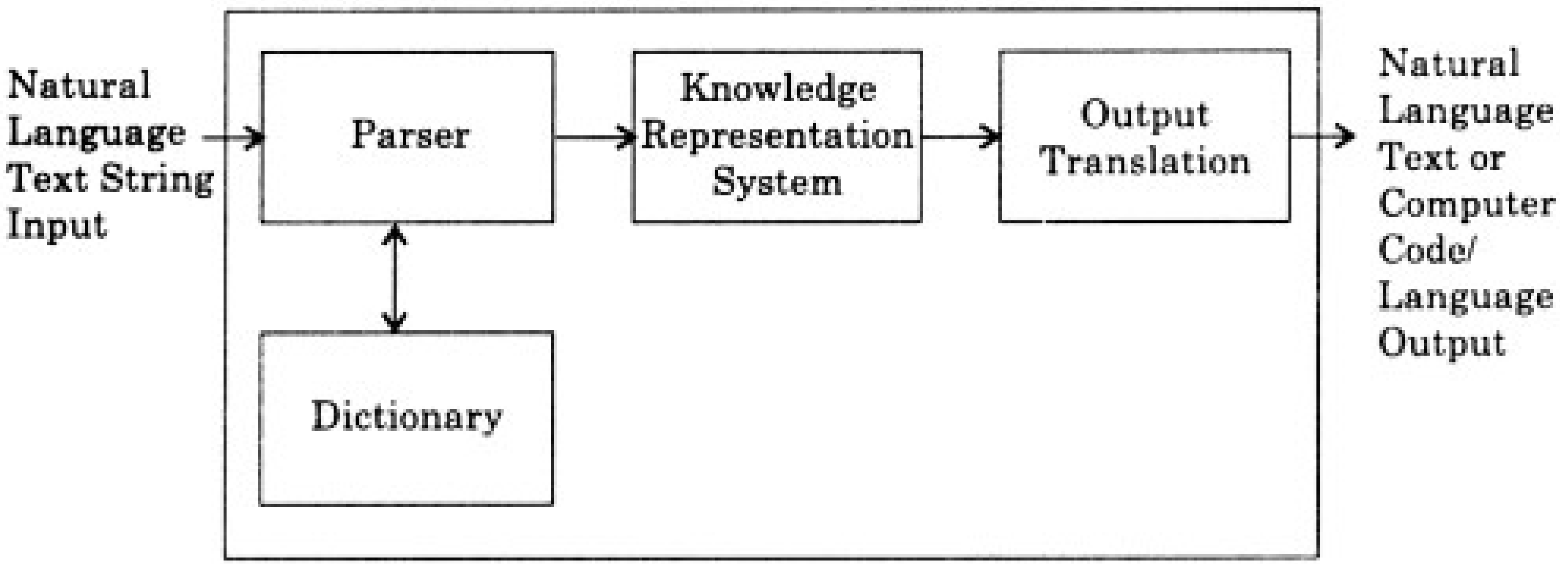


Fig. (7). The major components of a natural language processing system.

Application Areas of AI

- Computer Vision

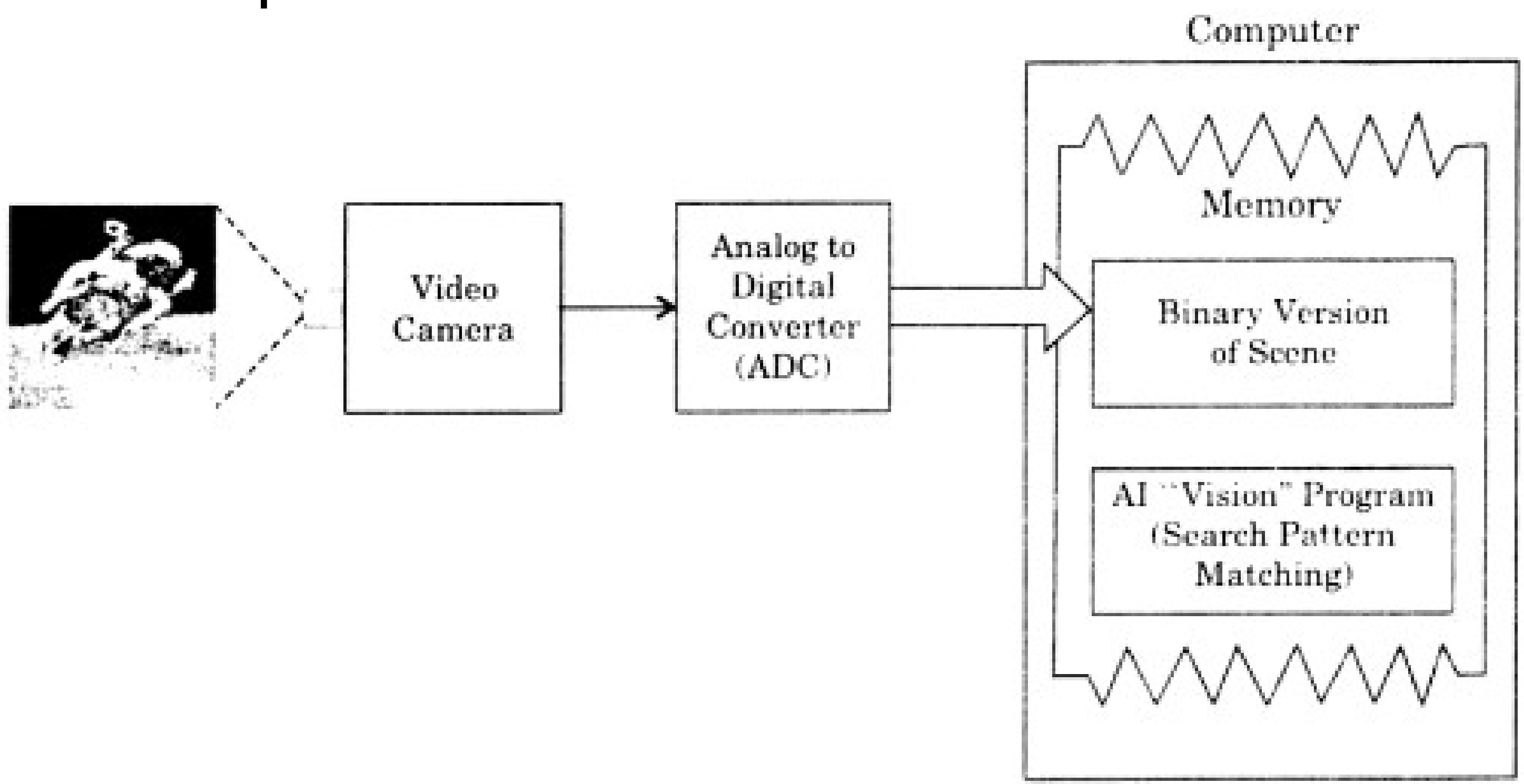


Fig. 8. A computer vision system.

AI Computing

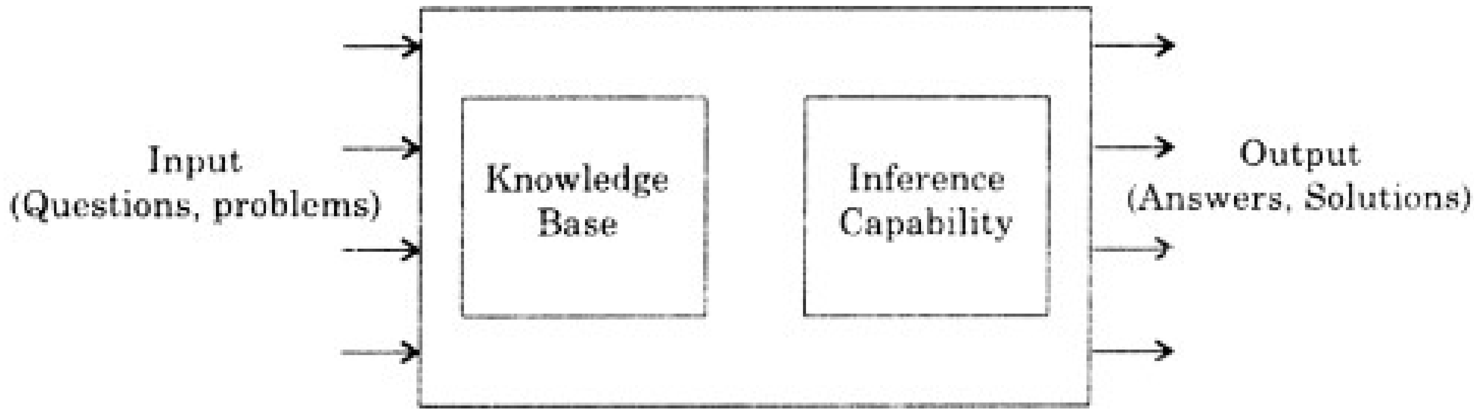


Fig. (2). AI Computing

In order to understand how a computer becomes smarter, let us first know the role of human brain. The human brain address the following queries:

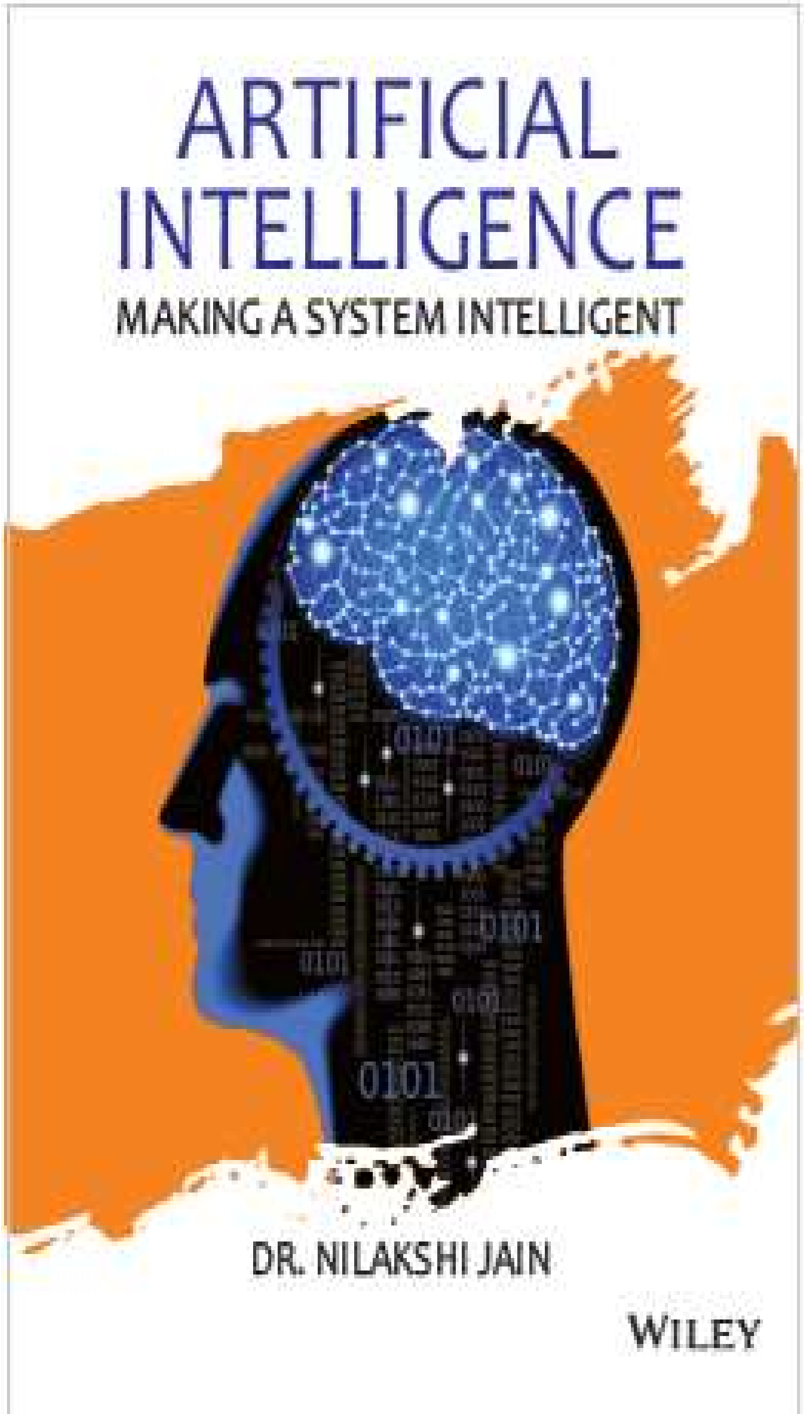
- How does a human being store knowledge?
- How does human being use this knowledge?
- How does a human being learn?
- How does a human being reason?

The art of performing these actions collectively is the aim of AI and is called ***cognitive science***. In conventional computing, computer is given data and computer is told how to solve a problem. Whereas in AI computing computer is given knowledge about a domain and some inferencing capability is inducted. The computer is then told what the problem is but not how to solve it. AI computing is capable of explaining how a particular conclusion was reached and why requested information was needed during the consultation. Hence it gives the user a chance to assess and understand the system's reasoning ability. The differences between conventional and AI computing is depicted in the table 1.2.

AI Computing

Table 1.2 Comparison of Conventional and AI Computing

Dimension	Conventional Computing	AI Computing
Processing	Primarily algorithmic	Includes symbolic[1] conceptualization
Nature of Input	Must be complete	Can be complete
Search Approach	Frequently based on algorithms	Frequently uses rules and heuristics
Explanation	Usually not provided	Provided
Focus	Data, information	Knowledge
Maintenance and update	Usually difficult	Relatively easy, changes can be made in self contained modules.
Reasoning Capability	No	Yes



Chapter Two

Intelligent Agents

Learning objectives.

After this chapter , students will be able to :

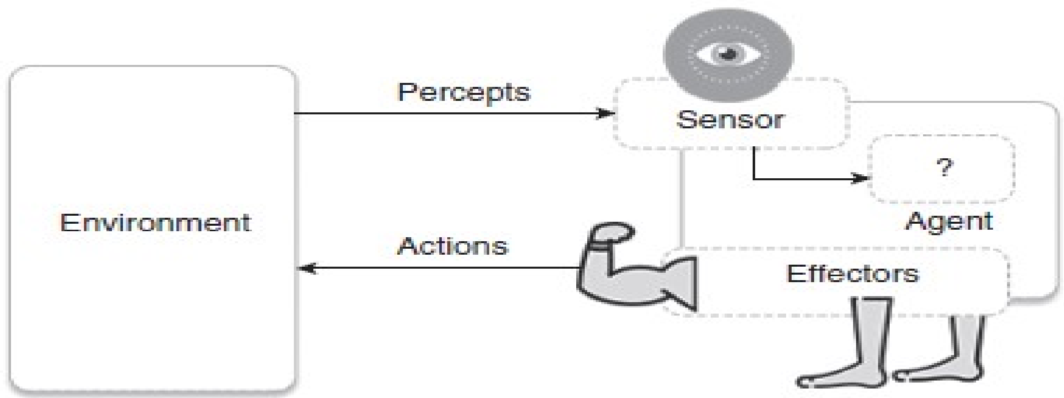
- Understand the concept of artificial intelligent agent
- Understand the basic concept of rationality
- Interpret and apply fundamental of type of agent on current problems related to artificial intelligence.

Intelligent systems

Intelligent systems are the frameworks or a system that may show a type of knowledge. This knowledge was made by preparing the framework with information, however, not utilizing the explicit programming. For example, a less complex smart framework/system could be the Gmail spam classifier.

Intelligent agent

An agent is defined something that sees (perceives) and acts in an environment. IAs perform task that will be beneficial for the business procedure, PC application or a



Human agent	A human agent has sensory organs like eyes, ears, nose, tongue and skin parallel to the sensors, and other organs such as hands, legs, mouth for effectors.
Robotic agent	A robotic agent replaces cameras and infrared range finders for the sensors, and various motors and actuators for effectors.
Software agent	A software agent encodes bit strings as its programs and actions.

Agent terminology

- **Performance measure of agent:** It is the criteria determining the success of an agent.
- **Behaviour/action of agent:** It is the action performed by an agent after any specified sequence of the percepts.
- **Percept:** It is defined as an agent's perceptual inputs at a specified instance.
- **Percept sequence:** It is defined as the history of everything that an agent has perceived till date.
- **Agent function**: It is defined as a map from the percept sequence to an action.

Agent function, $a = F(p)$

where *p is the current percept, a is the action carried out, and F is the agent function.*

Agent terminology

where p is the current percept, a is the action carried out, and F is the agent function. F maps precepts to actions

$$F : P \rightarrow A$$

where P is the set of all precepts, and A is the set of all actions.

Generally, an action may be dependent of all the precepts observed, not only the current percept,

$$a_k = F(p_0\, p_1\, p_2 \ldots p_k)$$

Where p_0 , p_1, p_2, … ,p_k is the sequence of percepts recorded till date, a_k is the resulting action carried out and F now maps percept sequences to action

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

Environment of problem

The vacuum agent perceives in which square it is in and if there is dirt in the square. It may choose to move left, right or suck up the dirt or do nothing. On the basis of the aforementioned task, one of the very simple functions of the agent is the following: If the current square is dirty then suck, otherwise move to other square. Hence, we can write:

Precepts: location and status, e.g., [A, Dirty]

Actions: left, right, suck and no-op (Do Nothing)

Function REFLEX-VACUUM-AGENT([*location*, *status*]) **returns**
Action **if** *status* = *Dirty* **then return** *Suck*
if *location* = A **then return**
Right **if** *location* = B **then**
return *Left* **return action**.

Vacuum cleaner problem

Performance measure of vacuum cleaner agent: All the rooms are well cleaned.

Behaviour/action of agent: Left, right, suck and no-op (Doing nothing).

Percept: Location and status, for example, [A, Dirty].

Agent function: Mapping of percept sequence to an action

Vacuum cleaner problem

Percept sequence

Percept sequence	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Dirty], [A, Clean]	Right
[A, Clean], [B, Dirty]	Suck
[B, Dirty], [B, Clean]	Left
[B, Clean], [A, Dirty]	Suck
[A, Clean], [B, Clean]	No-op
[B, Clean], [A, Clean]	No-op

Structure of agent

The structure of agent can be represented as:

Agent= Architecture + Agent

Program where,

Architecture : The machinery that an agent executes on

Agent program: An implementation of an agent function

PAGE

While designing the intelligent systems, there are following four main factors to be considered

- **P, Percepts:** The inputs to the AI system.
- **A, Actions:** The outputs of the AI system.
- **G, Goals:** What the agent is expected to achieve.
- **E, Environment:** What the agent is interacting with one.

Agent type	Percepts	Actions	Goals	Environ ment
Medical diagnosti c system	Symptoms , test results, patient's answers	Questions, test requests, treatments, referrals	Healthy patients, minimise the costs	Patient, hospital, staff
Satellite image analysis system	Pixels of varying intensity and colour	Display a categorisati on of the scene	Correct image categorisat ion	Images from orbiting satellite
Part-picking robot	Pixels of varying intensity and colour	Pick up parts and sort them into bins	Place parts into correct bins	Conveyor belt with parts, bins
Refinery controller	Temperatu re, pressure and chemical readings	Open and close valves, adjust temperature	Maximise purity, yield, safety	Refinery, staff
Interactiv	Typed	Display		Set of

Agent program

The skeleton of the IAs accept the percept's from the environment and actions are generated.

Firstly, although *agent mapping* is defined as the function from the percept sequence to actions, the agent mapping accepts only a single percept as its input

Secondly, the measure of goal or performance is not the part of skeleton program, since the performance measure is enforced externally to understand the behaviour of an agent, and it is also often possible to attain high performance without an explicit knowledge of measuring the performance.

Percept sequence	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Dirty], [A, Clean]	Right
[A, Clean], [B, Dirty]	Suck
[B, Dirty], [B, Clean]	Left
[B, Clean], [A, Dirty]	Suck
[A, Clean], [B, Clean]	No-op
[B, Clean], [A, Clean]	No-op

Agent program

On each invocation, the memory of the agent is updated to mirror the new percept, the best action selected and the fact that the action was taken is also stored inside the memory. The memory persists from one invocation to the next.

Function **Skeleton-Agent**(Percept) returns Action
    Static : Memory, the agents memory of the world
    Memory ←**Update_Memeory**(Memory, Percept)
    Action ← **Choose_Best_Action**(Memory)
    Memory ← **Update_Memory**(Memory,Percept)
Returns Action

2.1 Introduction

Percept sequence	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Dirty], [A, Clean]	Right
[A, Clean], [B, Dirty]	Suck
[B, Dirty], [B, Clean]	Left
[B, Clean], [A, Dirty]	Suck
[A, Clean], [B, Clean]	No-op
[B, Clean], [A, Clean]	No-op

agent

2.2.3 Autonomy

Agent program

On each invocation, the memory of the agent is updated to mirror the new percept, the best action selected and the fact that the action was taken is also stored inside the memory. The memory persists from one invocation to the next.

Function **Skeleton-Agent**(Percept) returns Action
  Static : Memory, the agents memory of the world
  Memory ← **Update_Memeory**(Memory, Percept)
  Action ← **Choose_Best_Action**(Memory)
  Memory ← **Update_Memory**(Memory,Percept)
Returns Action

Attributes of agent

Autonomy: Agent work without the direct interference of the people or others and have some kind of control over their action and the internal state.

Social ability: An intelligent agent can interact with other agents and humans in a social context. It can understand and respond to natural language, and can also take the initiative to communicate with others..

Reactivity: Agents perceive their condition which might include the physical world, client by the means of graphical user interface, an accumulation of agent.

Proactively: Agents do not simply act in response to their environment; but these are able to exhibit goal–director behavior by taking initiative.

Goal orienteer: An agent is efficient of handling complex high-level tasks. The decision for how such a task is best split into smaller subtasks, and in which order and manner these subtasks should be composed by the agents itself.

Rationality

the quality of being based on or in accordance with reason or logic.

Rationality is defined as the only status of being sensible and with great judgmental feeling. Rationality is only related to the expected activities or actions and this results relying on what the IA has viewed.

Performing activities with the point of obtaining the valuable data is said to be a critical piece of rationality.

Rational Agent

A right action is always performed by a rational agent, where the right action is equivalent to the action that causes the agent to be most successful in the given percept sequence. The problem the agent solves is characterised by the performance measure, environment, actuators and sensors (*PEAS*).

Definition. An Intelligent Agent is a system that can perceive its environment and take actions to achieve a specific goal. A Rational Agent is an Intelligent Agent that makes decisions based on logical reasoning and optimizes its behavior to achieve a specific goal.

Rational agent

Rationality of an agent depends on mainly on the following four parameters:

- The degree of success that is determined by the performance measures.
- The percept sequence of the agent that have been perceived till date.
- Prior knowledge of the environment that is gained by an agent till date.
- The actions that the agent might perform in the environment.

Ideal rational agent

An ideal rational agent is the one, which is competent enough of performing expected actions to expand its performance measure, on the basis of the following:

- Its percept sequence.
- Its built-in knowledge base.

Autonomy

If the behaviour of the system is determined by its own experience, the system is known as *autonomous*. When the agent has very less experience, it will be required to act in a random manner unless some assistance is provided either by the designer or the programmer. An actual autonomous IAs would be able to successfully and efficiently show all types of the environments, given sufficient time to adapt.

Types of agents

1. Simple reflex agent
2. Model based agent
3. Goal based agent
4. Utility based agent
5. Learning agent

Functions

```
// Function to compute the product of p1 and p2
function myFunction(p1, p2) {
  return p1 * p2;
}
```

```
// Function is called, the return value will end up in x
let x = myFunction(4, 3);

function myFunction(a, b) {
// Function returns the product of a and b
  return a * b;
}
```

Types of agent

1. Simple reflex agent

Simple reflex agent is said to be the simplest kind of agent. These agents select an action based on the current percept ignoring the rest of the percept history.

These percept to action mapping which is known as *condition-action rules* (so-called *situation–action rules, productions, or if–then rules)* in the simple reflex agent. It can be represented as follows:

if *{set of percepts}* **then** *{set of actions}*

For example,

if *it is raining* **then** *put up umbrella*

or

if *carin front is breaking* **then** *initiate breaking*

Simple Reflex Agent

function REFLEX-VACUUM-AGENT([*location*,*status*]) **returns** an action

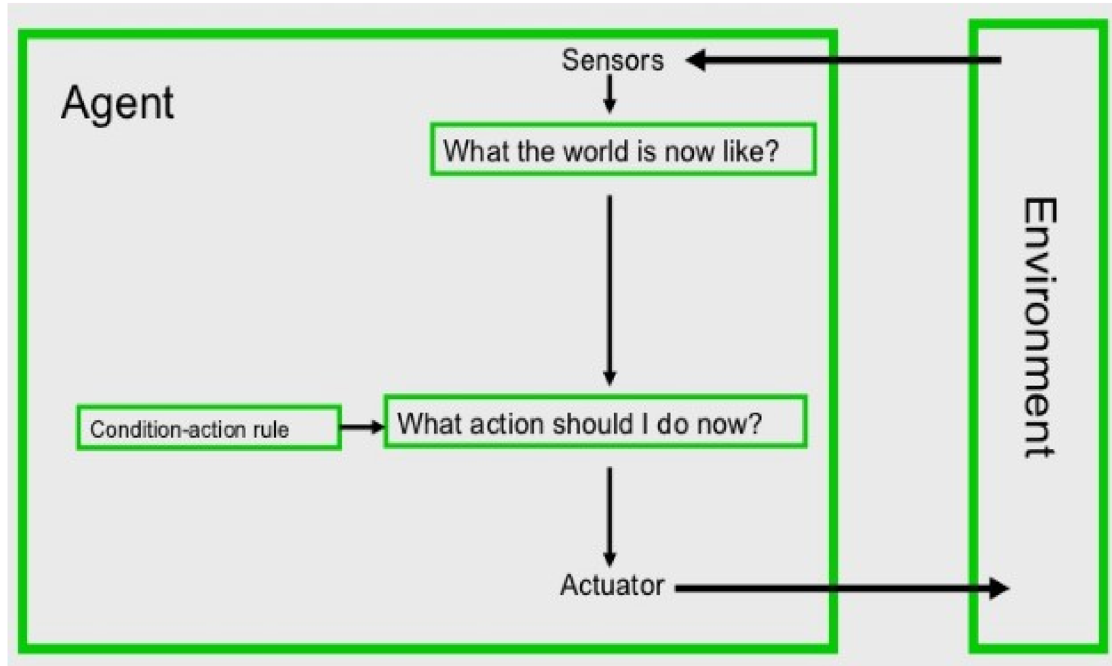
  if *status* = *Dirty* **then return** *Suck*
  else if *location* = *A* **then return** *Right*
  else if *location* = *B* **then return** *Left*

Figure 2.8 The agent program for a simple reflex agent in the two-state vacuum environment. This program implements the agent function tabulated in Figure 2.3.

Percept sequence	Action
[*A*, *Clean*]	*Right*
[*A*, *Dirty*]	*Suck*
[*B*, *Clean*]	*Left*
[*B*, *Dirty*]	*Suck*
[*A*, *Clean*], [*A*, *Clean*]	*Right*
[*A*, *Clean*], [*A*, *Dirty*]	*Suck*
⋮	⋮
[*A*, *Clean*], [*A*, *Clean*], [*A*, *Clean*]	*Right*
[*A*, *Clean*], [*A*, *Clean*], [*A*, *Dirty*]	*Suck*
⋮	⋮

Figure 2.3 Partial tabulation of a simple agent function for the vacuum-cleaner world shown in Figure 2.2.

- The agent in Figure will work only if the correct decision can be made on the basis of only the current percept—that is, only if the environment is fully observable.
- Even a little bit of unobservability can cause serious trouble.
- Some cars are using old model signal lights
- Simple reflex agent may break continuously, (How?)
- No break at all




```
Function SIMPLE-REFLEX-AGENT(percept) returns Action
persistent: rules, a set of condition-action rules
state = INTERPRET-INPUT(percept)
rule = MATCH-RULE(state, rules)
action=RULE-ACTION(rule)
return rule.action
```

Code for simple reflex agent

Simple Reflex Agent - Problems

- We can see a similar problem arising in the vacuum world.
- Suppose that a simple reflex vacuum agent is deprived of its location sensor and has only a dirt sensor.
- Such an agent has just two possible percepts: [Dirty] and [Clean].
- It can Suck in response to [Dirty];
- what should it do in response to [Clean]?
    - Moving Left fails (forever) if it happens to start in square A, and moving Right fails (forever) if it happens to start in square B.
- Infinite loops are often unavoidable for simple reflex agents operating in partially observable environments.

Limitations

- Intelligence level in these agents is very limited.
- It works only in a fully observable environment.
- It does not hold any knowledge or information of non-perceptual parts of state.
- Because of the static knowledge based; it’s usually too big to generate and store.
- If any change in the environment happens, the collection of the rules are required to be updated.

If you put sudden break then what happens to car behind you – Previous State

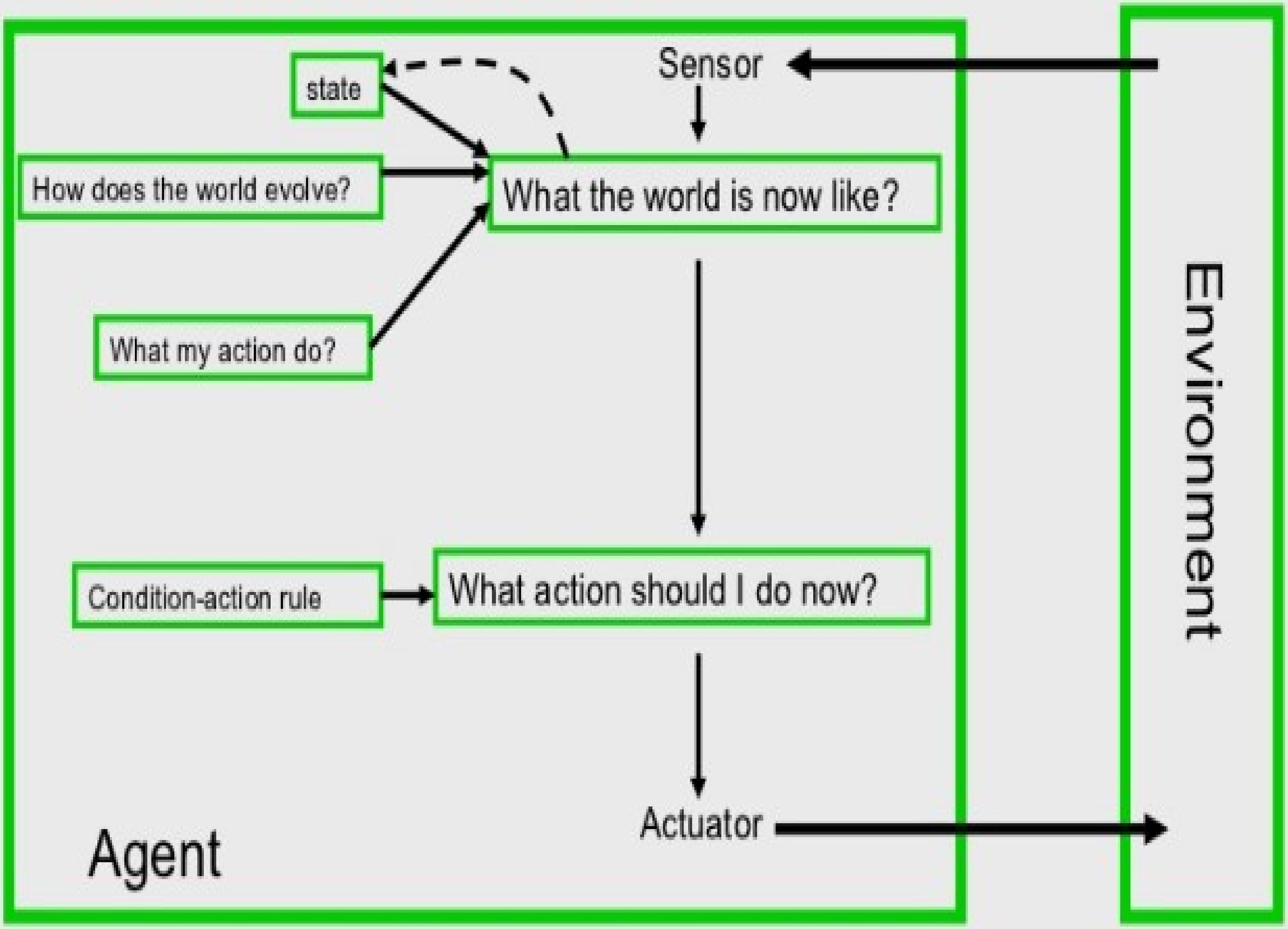
If you turn right, what happens – What my actions do

What will happen when another car overtakes you – How does world evolves

2. Model based agent

Model-based agent is known as *Reflex agents with an internal state.* One problem with the simple reflex agents is that their activities are dependent of the recent data provided by their sensors. On the off chance that a reflex agent could monitor its past states (that is, keep up a portrayal or "model" of its history in the world), and understand about the development of the world.

Keep track of the world it can't see now. Consider lane change.



```
Function REFLEX-AGENT-WITH-STATE(percept) returns action
persistent: state, conception of the world state.
persistent: model, of rule/state transitions.
persistent: rules, a set of condition–action rules.
persistent: action, the most recent action, initially none.
state = UPDATE–STATE (state, action, percept, model)
rule = MATCH–RULE (state, rules)
action=RULE–ACTION(rule)
return action
```

Code for model based agent

Goal – Safety or Speed

What will you do when car infront brakes : Turn right or Break

What is the Goal – Nor hitting other car

How ?

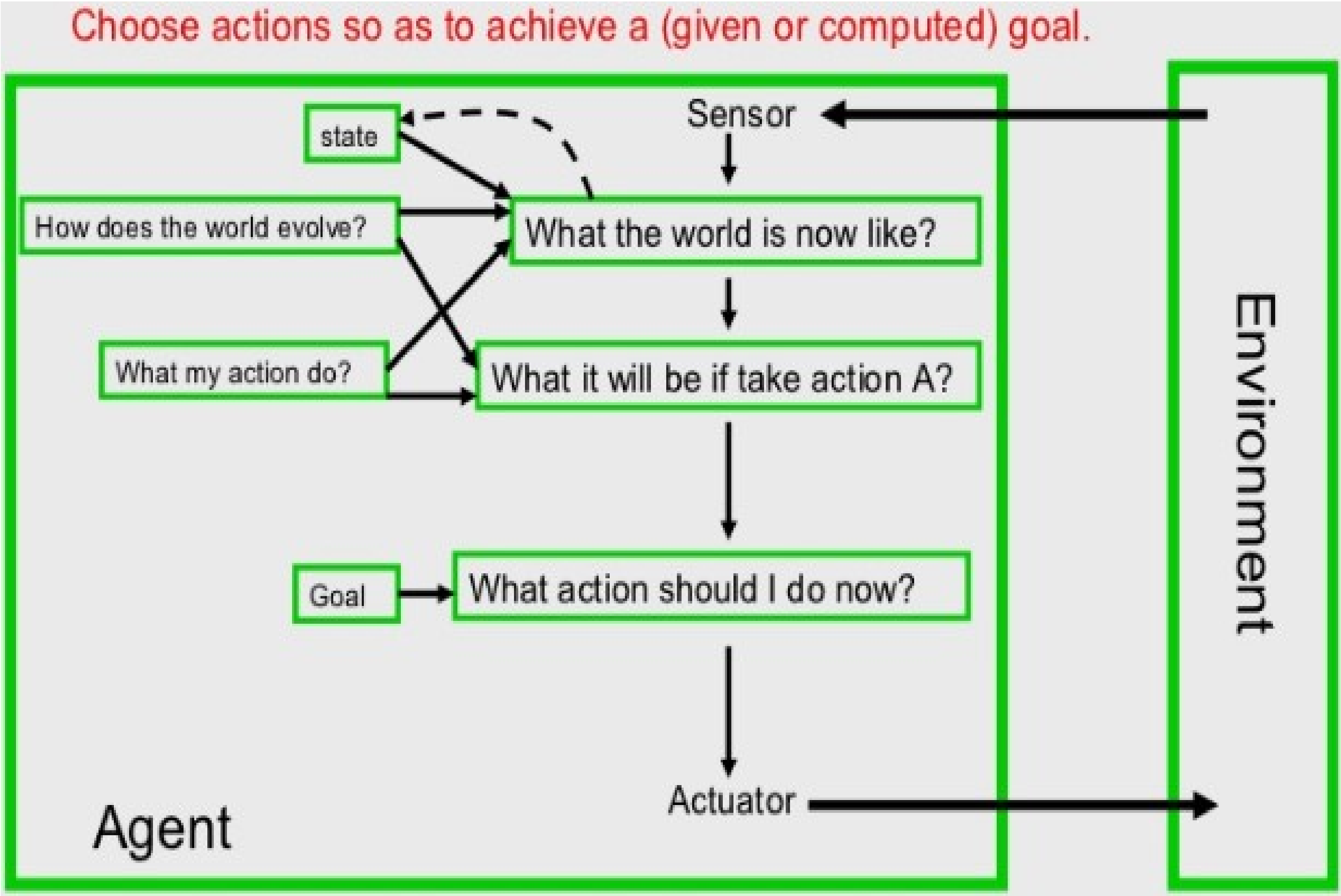
Applying Break ?

Slowing Down ?

Applying Break continuously – Make unhappy

3. Goal based

Indeed, even with the expanded information of the current situation of the world given by an agent's internal state, the agent may not, in any case, have enough data to reveal to it. The proper action for the agent will regularly depend upon its goals .Thus, it must be provided with some goal information.


Choose actions so as to achieve a (given or computed) goal.
state
Sensor
How does the world evolve?
What the world is now like?
What my action do?
What it will be if take action A?
Goal
What action should I do now?
Actuator
Agent
Environment

```
Function REFLEX-AGENT-WITH-GOAL (percept) returns action.
persistent: state, conception of world state.
persistent: rules, a set of condition–action rules.
persistent: action, the most recent action, initially none.
 state = UPDATE-STATE (state, action, percept).
 rule = MATCH-RULE (state, rules).
rule = GOAL-TEST (rules)
action=RULE-ACTION (rule)
return action
```

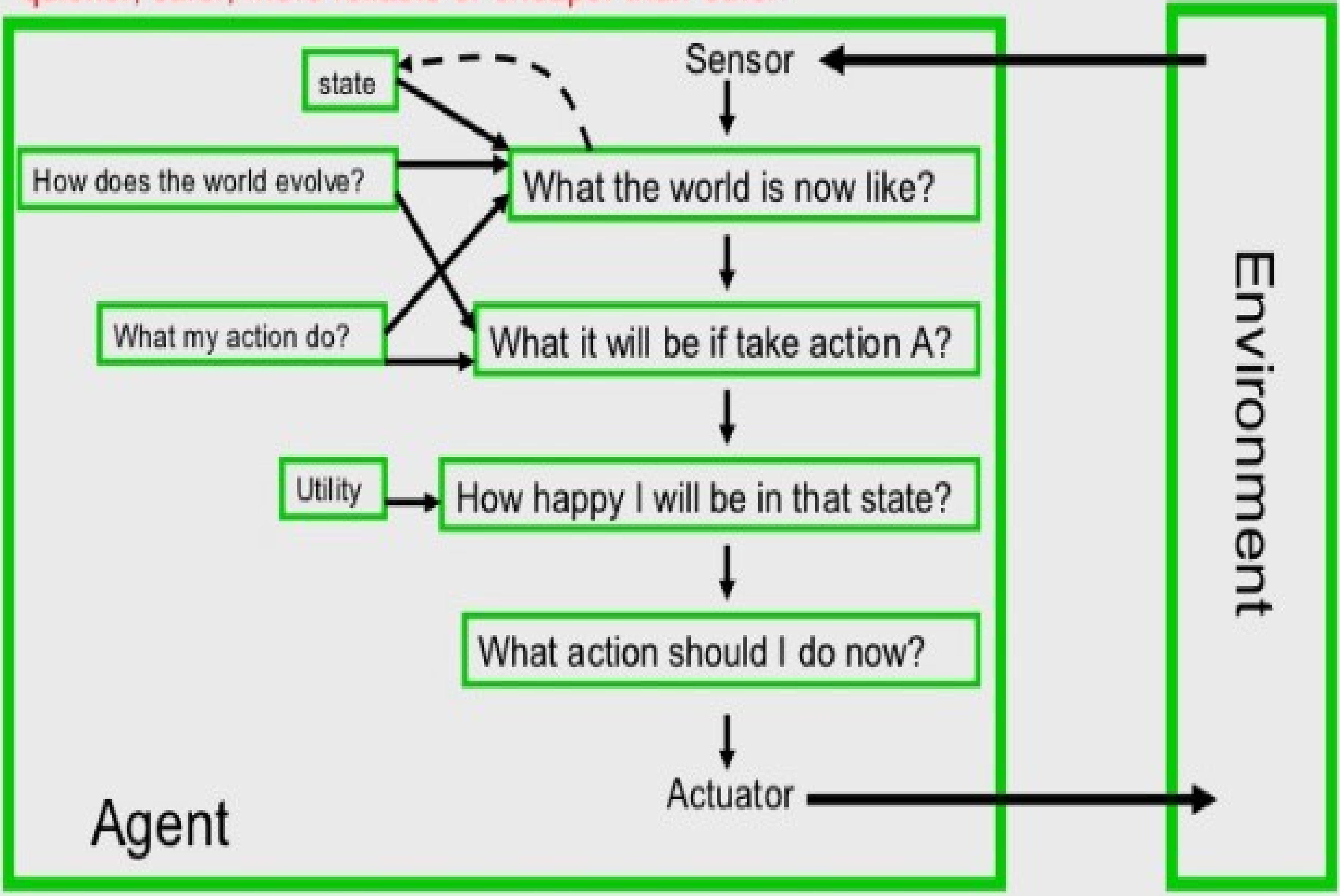
Code for goal based agent

4. Utility based agent

Goals individually are insufficient to produce top high-quality behavior. Frequently, there are numerous groupings of actions that can bring about a similar goal being accomplished. Given proper criteria, it might be conceivable to pick 'best' sequence of actions from a number that all result in the goal being achieved.

Any utility-based agent can be depicted as having an utility capacity that maps a state, or grouping of states, on to a genuine number that speaks to its utility or convenience or usefulness.

Goal alone is not enough. There can be many way to achieve goal but some are quicker, safer, more reliable or cheaper than other.



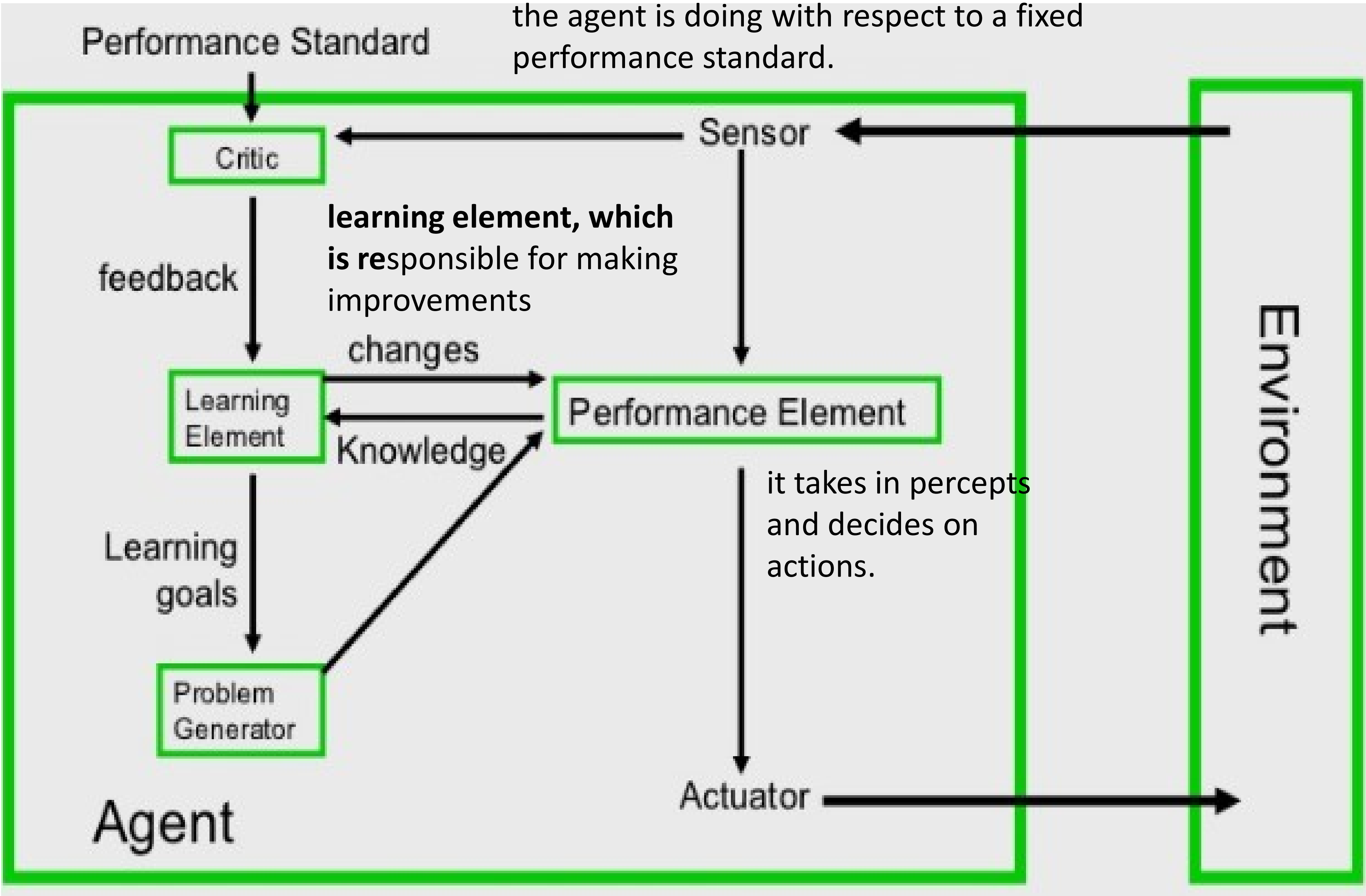
```
Function REFLEX-AGENT-WITH-UTILITY(percept) returns action
persistent: state, conception of world state
persistent: rules, a set of condition-action rules
persistent: action, the most recent action, initially none
 state = UPDATE-STATE(state, action, percept)
 rule = MATCH-RULE(state, rules)
rule = UTILITY-OF-RULE(rules)
action=RULE-ACTION(rule)
return action
```

Code for utility based agent

5. Learning agent

By actively exploring and experimenting with their environment, the most powerful agents are able to learn. A learning agent can be further divided into the four conceptual components

The critic tells the learning element how well the agent is doing with respect to a fixed performance standard.



learning element, which is responsible for making improvements

it takes in percepts and decides on actions.

Learning agent

- A learning agent can be divided into four conceptual components, as shown in Figure
- The most important distinction is between the **learning element, which is re**sponsible for making improvements
- the **performance element, which is responsible for** selecting external actions.
- The performance element is what we have previously considered to be the entire agent: it takes in percepts and decides on actions.
- The 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

Learning agent - Example

- **automated taxi example.**
- The performance element consists of whatever collection of knowledge and procedures the taxi has for selecting its driving actions.
- The taxi goes out on the road and drives, using this performance element.
- The critic observes the world and passes information along to the learning element.
- For example, after the taxi makes a quick left turn across three lanes of traffic, the critic observes the shocking language used by other drivers.
- From this experience, the learning element is able to formulate a rule saying this was a bad action, and
- the performance element is modified by installation of the new rule.
- The problem generator might identify certain areas of behavior in need of improvement and suggest experiments, such as trying out the brakes on different road surfaces under different conditions.

Environment and its properties

1. *Accessible and Inaccessible Environments*

2. *Deterministic Environment and Nondeterministic Environment*

3. *Episodic and Nonepisodic Environment*

4. *Static versus Dynamic Environment*

5. *Discrete versus Continuous Environment*

6. *Single Agent versus Multiagent Environment*

Environment and its properties

- The **Environment** is the surrounding world around the agent which is not part of the agent itself.
- It's important to understand the nature of the environment when solving a problem using **artificial intelligence**.
- For example, program a chess bot, the environment is a chessboard and creating a room cleaner robot, the environment is **Room**.
- Each environment has its own properties and agents should be designed such as it can explore environment states using sensors and act accordingly using actuators. In this guide, we're going to understand all types of environments with real-life examples.

Fully Observable vs Partially-Observable

In a **fully observable environment**, The Agent is familiar with the complete state of the environment at a given time. There will be no portion of the environment that is hidden for the agent.

Real-life Example: While running a car on the road (**Environment**), The driver (**Agent**) is able to see road conditions, signboard and pedestrians on the road at a given time and drive accordingly. So Road is a **fully observable environment** for a driver while driving the car.

in a **partially observable environment**, The agent is not familiar with the complete environment at a given time.

Real-life Example: Playing card games is a perfect example of a **partially-observable environment** where a player is not aware of the card in the opponent's hand. Why partially-observable? Because the other parts of the environment, e.g opponent, game name, etc are known for the player (Agent).

Deterministic vs Stochastic

Deterministic are the environments where the next state is observable at a given time. So there is no uncertainty in the environment.

Real-life Example: The traffic signal is a deterministic environment where the next signal is known for a pedestrian (**Agent**)

The Stochastic environment is the opposite of a deterministic environment. The next state is totally unpredictable for the agent. So randomness exists in the environment.

Real-life Example: The radio station is a stochastic environment where the listener is not aware about the next song or playing a soccer is stochastic environment.

Episodic vs Sequential

Episodic is an environment where each state is independent of each other. The action on a state has nothing to do with the next state.

Real-life Example: A support bot (agent) answer to a question and then answer to another question and so on. So each question-answer is a single episode.

The sequential environment is an environment where the next state is dependent on the current action. So agent current action can change all of the future states of the environment.

Real-life Example: Playing tennis is a perfect example where a player observes the opponent's shot and takes action.

Static vs Dynamic

The **Static** environment is completely unchanged while an agent is precepting the environment.

Real-life Example: Cleaning a room (Environment) by a dry-cleaner reboot (Agent) is an example of a static environment where the room is static while cleaning.

Dynamic Environment could be changed while an agent is precepting the environment. So agents keep looking at the environment while taking action.

Real-life Example: Playing soccer is a dynamic environment where players' positions keep changing throughout the game. So a player hit the ball by observing the opposite team.

Discrete vs Continuous

Discrete Environment consists of a finite number of states and agents have a finite number of actions.

Real-life Example: Choices of a move (action) in a tic-tac game are finite on a finite number of boxes on the board (Environment).

While in a **Continuous environment**, the environment can have an infinite number of states. So the possibilities of taking an action are also infinite.

Real-life Example: In a basketball game, the position of players (**Environment**) keeps changing continuously and hitting (**Action**) the ball towards the basket can have different angles and speed so infinite possibilities.

Single Agent vs Multi-Agent

Single agent environment where an environment is explored by a single agent. All actions are performed by a single agent in the environment.

Real-life Example: Playing tennis against the ball is a single agent environment where there is only one player.

If two or more agents are taking actions in the environment, it is known as a multi-agent environment.

Real-life Example: Playing a soccer match is a multi-agent environment.

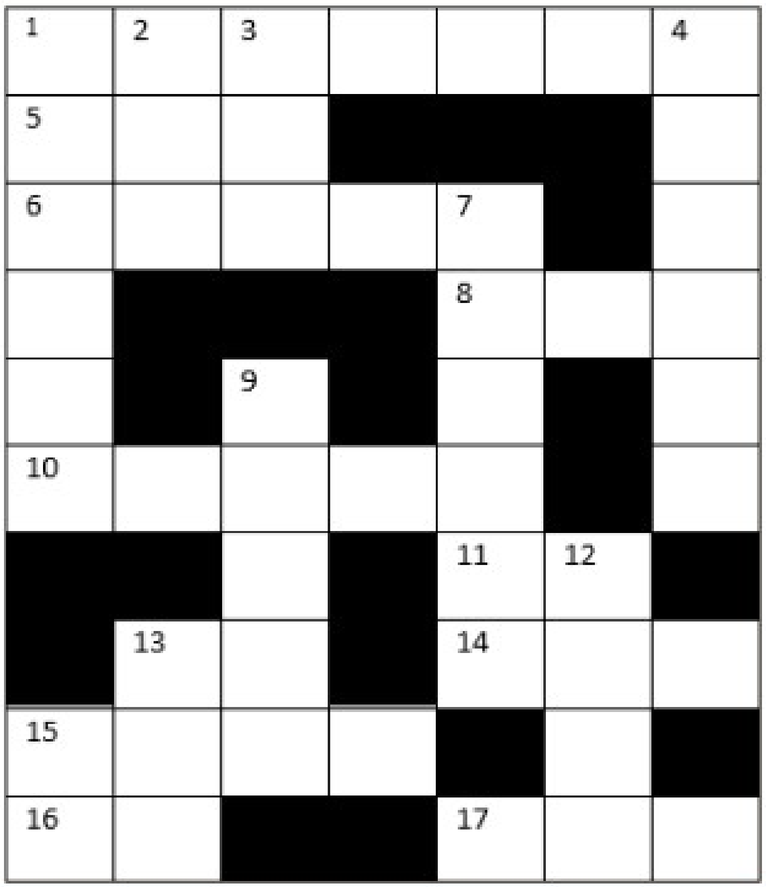
There are mainly six groups of environment and an environment can be in multiple groups. Below are 10 more real-life examples and categories into environment groups.

	Fully vs Partially Observable	Deterministic vs Stochastic	Episodic vs Sequential	Static vs Dynamic	Discrete vs Continuous	Single vs Multi Agents
Brushing Your Teeth	*Fully*	*Stochastic*	*Sequential*	*Static*	*Continuous*	*Single*
Playing Chess	*Partially*	*Stochastic*	*Sequential*	*Dynamic*	*Continuous*	*Multi-Agent*
Playing Cards	*Partially*	*Stochastic*	*Sequential*	*Dynamic*	*Continuous*	*Multi-Agent*
Playing	*Partially*	*Stochastic*	*Sequential*	*Dynamic*	*Continuou*	*Multi Agent*
Autonomous Vehicles	*Fully*	*Stochastic*	*Sequential*	*Dynamic*	*Continuous*	*Multi-Agent*
Order in Restaurant	Fully	Deterministic	Episodic	Static	Discrete	Single Agent

Fully observable vs. partially observable:

- If an agent's sensors give it access to the complete state of the environment at each point in time, then we say that the task environment is fully observable.
- A task environment is effectively fully observable if the sensors detect all aspects that are relevant to the choice of action; relevance, in turn, depends on the performance measure.
- Fully observable environments are convenient because the agent need not maintain any internal state to keep track of the world.
- An environment might be partially observable because of noisy and inaccurate sensors or because parts of the state are simply missing from the sensor data—
  - for example, **a vacuum agent** with only a local dirt sensor cannot tell whether there is **dirt in other squares**, and **an automated taxi** cannot see what other **drivers are thinking**.
- If the agent has no sensors at all then the environment is unobservable.
- One might think that in such cases the agent's plight is hopeless, but, as we discuss in Chapter 4, the agent's goals may still be achievable, sometimes with certainty.

Crossword Puzzle



Down

1. Place without water
2. Rodest pest
3. Anger
4 Scattered all over
7. A small river
9. Not this one but the .... One
12. The sort of house a mother would be proud of
13. Spoil the work
15. Abbreviation

Across

1. Find them in cars
5. Use it to listen
6. Dishes of meat and vegetable cooked together in water
8. Look for one on the end of your foot
10. The name of a book
11. Article
13. Not you
14. Listen to the kitten
15. People like to race them
16. Little connector
17. Past tense, rhymes with a number under ten

Crossword Puzzle Answer

1 D	2 R	3 I	V	E	R	4 S
5 E	A	R				T
6 S	T	E	W	7 S		R
E				8 T	O	E
R		9 O		R		W
10 T	I	T	L	E		N
		H		11 A	12 N	
	13 M	E		14 M	E	W
15 C	A	R	S		A	
16 O	R			17 A	T	E

Single agent vs. multiagent:

- The distinction between single-agent and multiagent environments may seem simple enough.
- For example, an agent solving a **crossword puzzle** by itself is clearly in a single-agent environment,
- whereas an agent **playing chess** is in a two agent environment
- For example, in chess, the opponent entity B is trying to maximize its performance measure, which, by the rules of chess, minimizes agent A's performance measure.
- Thus, chess is a **competitive multiagent** environment.
- In the taxi-driving environment, on the other hand, avoiding collisions maximizes the performance measure of all agents, so it is a **partially cooperative** multiagent environment.
- It is also partially competitive because, for example, only one car can occupy a parking space

having a random probability distribution or pattern that may be analysed statistically but may not be predicted precisely.

Deterministic vs. stochastic

- If the next state of the environment is completely determined by the current state and the action executed by the agent, then we say the environment is **deterministic**; otherwise, it is stochastic.
- **Taxi driving** is clearly **stochastic** in this sense, because one can never predict the behavior of **traffic exactly**; moreover, **one's tires blow out** and **one's engine seizes** up without warning
- The vacuum world as we described it is **deterministic**, but variations can include stochastic elements such as randomly appearing dirt and an unreliable suction mechanism
- We say an environment is **uncertain** if it is not fully observable or **not deterministic**

Episodic vs. sequential

- In an episodic task environment, the agent's experience is SEQUENTIAL divided into atomic episodes.
- In each episode the agent receives a percept and then performs a single action.
- Crucially, the next episode does not depend on the actions taken in previous episodes.
- Many classification tasks are episodic.
- For example, an agent that has to spot defective parts on an assembly line bases each decision on the current part,
  - regardless of previous decisions; moreover, the current decision doesn't affect whether the next part is defective.
- In sequential environments, on the other hand, the current decision could affect all future decisions.
- Chess and taxi driving are sequential: in both cases, short-term actions can have long-term consequences.
- Episodic environments are much simpler than sequential environments because the agent does not need to think ahead

Static vs. dynamic

- If the environment can change while an agent is deliberating, then we say the environment is dynamic for that agent; otherwise, it is static.
- Static environments are easy to deal with because the agent need not keep looking at the world while it is deciding on an action, nor need it worry about the passage of time.
- Dynamic environments, on the other hand, are continuously asking the agent what it wants to do; if it hasn't decided yet, that counts as deciding to do nothing.
- If the environment itself does not change with the passage of time but the agent's performance score does, then we say the environment is semidynamic
- Taxi driving is clearly dynamic: the other cars and the taxi itself keep moving while the driving algorithm dithers about what to do next.
- Chess, when played with a clock, is semidynamic.
- Crossword puzzles are static.

Discrete vs. continuous:

- Discrete vs. continuous:
  - Chess
  - Taxi Driving
- The discrete/continuous distinction applies to the state of the environment, to the way time is handled, and to the percepts and actions of the agent.
- For example, the chess environment has a finite number of distinct states (excluding the clock).
- Chess also has a discrete set of percepts and actions.
- Taxi driving is a continuous-state and continuous-time problem: the speed and location of the taxi and of the other vehicles sweep through a range of continuous values and do so smoothly over time.
- Taxi-driving actions are also continuous (steering angles, etc.).
- Input from digital cameras is discrete, strictly speaking, but is typically treated as representing continuously varying intensities and locations.

Known vs. unknown

- Strictly speaking, this distinction refers not to the environment itself but to the agent’s (or designer’s) state of knowledge about the “laws of physics” of the environment.
- In a known environment, the outcomes (or outcome probabilities if the environment is stochastic) for all actions are given.
- Obviously, if the environment is unknown, the agent will have to learn how it works in order to make good decisions.
- Note that the distinction between known and unknown environments is not the same as the one between fully and partially observable environments
- Example Rover in Moon
- Example Rover in Cricket Stadium

Examples of task environments

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Interactive English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete

Figure 2.6 Examples of task environments and their characteristics.

PEAS for Agent

PEAS Descriptors for automated Taxi Driver	
Performance Measure	**Safety, time, legal drive, comfort**
Environment	**Roads, other cars, pedestrians, road signs**
Actuators	**Steering, accelerator, brake, signal, horn**
Sensors	**Camera, sonar, GPS, Speedometer, odometer, accelerometer, engine**

PEAS for vacuum cleaner	
Performance Measure	cleanness, efficiency: distance traveled to clean, battery life, security
Environment	room, table, wood floor, carpet, different obstacles
Actuators	wheels, different brushes, vacuum extractor
Sensors	camera, dirt detection sensor, cliff sensor, bump sensors, infrared wall sensors

Environment Types-I : Fully observable (accessible) vs. partially observable (inaccessible)	**Environment Types-II: Deterministic vs. stochastic (non-deterministic)**
Fully observable if agents sensors detect all aspects of environment relevant to choice of action Could be partially observable due to noisy, inaccurate or missing sensors, or inability to measure everything that is needed Model can keep track of what was sensed previously, cannot be sensed now, but is probably still true. Often, if other agents are involved, their intentions are not observable, but their actions are E.g chess – the board is fully observable, as are opponent's	· Deterministic = the next state of the environment is completely predictable from the current state and the action executed by the agent Stochastic = the next state has some uncertainty associated with it Uncertainty could come from randomness, lack of a good environment model, or lack of complete sensor coverage Strategic environment if the environment is deterministic except for the actions of other agents Examples: Non-deterministic environment:

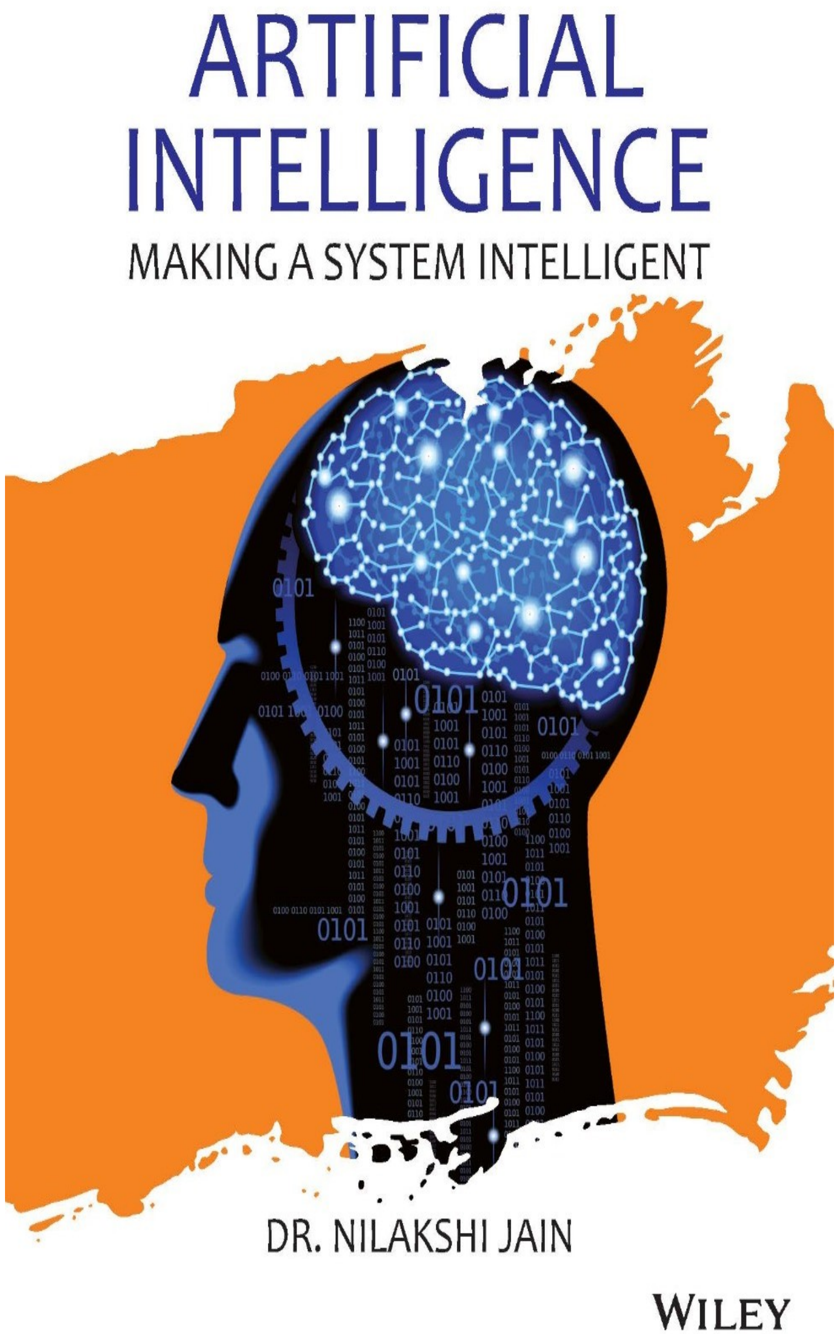
Environment Types-III : Episodic vs. sequential	**Environment Types-IV: Discrete vs. continuous**
The agent's experience is divided into atomic "episodes" (each episode consists of the agent perceiving and then performing a single action) and the choice of action in each episode depends only on the episode itself Sequential if current decisions affect future decisions, or rely on previous ones Examples of episodic are expert advice systems – an episode is a single question and answer Most environments (and agents) are sequential Many are both – a number of episodes containing a number of	Discrete = time moves in fixed steps, usually with one measurement per step (and perhaps one action, but could be no action). E.g. a game of chess Continuous = Signals constantly coming into sensors, actions continually changing. E.g. driving a car

Environment types-V: Static vs. dynamic:	**Environment types-VI Single agent vs. multi agent:**
- Dynamic if the environment may change over time. Static if nothing (other than the agent) in the environment changes - Other agents in an environment make it dynamic - The goal might also change over time - Not dynamic if the agent moves from one part of an environment to another, though it has a very similar effect - E.g. – Playing football, other players make it dynamic, mowing a lawn is static (unless there is a cat…), expert systems usually static (unless	- An agent operating by itself in an environment is single agent! - Multi agent is when other agents are present! - A strict definition of an other agent is anything that changes from step to step. A stronger definition is that it must sense and act - Competitive or co-operative Multi-agent environments - Human users are an example of another agent in a system - E.g. Other players in a football team (or opposing team), wind and waves in a sailing agent, other cars in a taxi driver

	Fully observable?	**Deterministic?**	**Episodic?**	**Static?**	**Discrete?**	**Single agent?**
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes	No
Taxi driving	No	No	No	No	No	No
Internet shopping	No	No	No	No	Yes	No
Medical diagnosis	No	No	No	No	No	Yes
Crossword puzzle	Yes	Yes	Yes	Yes	Yes	Yes
English tutor	No	No	No	No	Yes	No
Image analysis	Yes	Yes	Yes	No	No	Yes

Intelligent agent application

1. Systems and network management
2. Information access and management
3. Work flow and administrative management
4. Customer help desk
5. Personal shopping assistant



Dr. Nilakshi Jain

Email ID : nilakshijain1986@gmail.com

Thank you