ABHYASMITRA.IN
STUDY NOTES

ARTIFICIAL INTELLIGENCE

Unit I — Introduction to AI and Intelligent Agents

Topics Covered (09 Hours)

Introduction & Foundations of Artificial Intelligence
History of Artificial Intelligence (Milestones & Winters)
Limits of AI (Computational, Philosophical & Practical)
Ethics of AI (Bias, Transparency & Accountability)
Future of AI (Narrow vs. General vs. Super AI)
AI Components & AI Architectures
Intelligent Agents & Environments
Good Behavior & Concept of Rationality
Nature of Environments & Properties
Types of Agents & Structure of Agents

Compiled in a simple, easy-to-understand, book style format for students of Computer Science and Engineering

TABLE OF CONTENTS

1.1 Introduction to Artificial Intelligence

Artificial Intelligence (AI) is one of the most exciting and transformative fields of modern science. At its core, AI is about creating machines or computer programs that are capable of performing tasks that would normally require human intelligence. These tasks include learning from experience, reasoning to solve problems, understanding natural human language, recognizing visual patterns, and making decisions.

While humans are born with natural intelligence that grows through education and life experiences, machines must be provided with synthetic or artificial intelligence. We achieve this by writing specialized algorithms and processing massive amounts of data so that machines can simulate cognitive functions.

Definition

Artificial Intelligence is the branch of computer science concerned with building smart machines capable of performing tasks that typically require human intelligence, particularly by simulating human cognitive processes such as learning, reasoning, and self-correction.

Four Dimensions of Defining AI

Over the years, researchers have defined AI in different ways. In their famous textbook, Stuart Russell and Peter Norvig organized these definitions into a 2x2 matrix based on two main axes: whether the definition focuses on thought processes versus behavior, and whether it measures success against human performance versus an ideal standard of rationality (known as 'acting or thinking rationally').

Approach	Thinking Dimension	Acting Dimension
Human-Centric	Thinking Humanly: Cognitive modeling approach. Focuses on mimicking the inner thought processes, logic, and decision paths of the human brain.	Acting Humanly: The Turing Test approach. Focuses on creating machines that perform actions indistinguishable from those of a human.
Rational-Centric	Thinking Rationally: 'Laws of thought' approach. Focuses on formalizing logic and rules of inference to reach correct, logical conclusions.	Acting Rationally: The Rational Agent approach. Focuses on building agents that act in a way that maximizes their chances of achieving the best outcome.

1. Acting Humanly: The Turing Test Approach

Proposed by Alan Turing in 1950, the Turing Test (originally called the Imitation Game) provides an operational definition of artificial intelligence. Under this test, a human interrogator sits in a separate room and sends text-based questions to two entities: a human and a computer. If the interrogator cannot

reliably tell which responder is the machine and which is the human after a conversation, the computer passes the test and is considered intelligent.

To pass the Turing Test, a computer needs to possess several core capabilities:

- **Natural Language Processing (NLP):** To communicate successfully in a human language like English.
- **Knowledge Representation:** To store what it knows or hears.
- **Automated Reasoning:** To use the stored information to answer questions and draw new conclusions.
- **Machine Learning:** To adapt to new circumstances and detect and extrapolate patterns.

Later, researchers proposed the Total Turing Test, which adds physical interaction. To pass the Total Turing Test, the machine also needs Computer Vision (to perceive objects) and Robotics (to manipulate objects and move around).

2. Thinking Humanly: The Cognitive Modeling Approach

If we want to build a machine that thinks like a human, we must first understand how humans actually think. We can observe human thoughts in three ways: by catching our own thoughts as they occur (introspection), by observing a person in action (psychological experiments), and by looking at the physical brain in action (brain imaging). Once we have a sufficiently precise theory of the human mind, we can express it as a computer program. This intersection of AI and cognitive psychology is called Cognitive Science.

3. Thinking Rationally: The 'Laws of Thought' Approach

Greek philosopher Aristotle was among the first to attempt to codify 'right thinking'—that is, irrefutable thought processes. His famous syllogisms provided patterns for argument structures that always yielded correct conclusions when given correct premises (e.g., 'Socrates is a man; all men are mortal; therefore, Socrates is mortal'). These laws of thought were supposed to govern the operation of the mind, and their study initiated the field of formal logic.

4. Acting Rationally: The Rational Agent Approach

An agent is simply something that acts (the word comes from the Latin 'agere', meaning 'to do'). In this book, we focus on rational agents. A rational agent is one that acts so as to achieve the best outcome, or, when there is uncertainty, the best expected outcome. The rational-agent approach has two advantages over the other approaches: first, it is more general than the 'laws of thought' approach because correct inference is only one of several possible ways to achieve rationality; second, it is more amenable to scientific development than approaches based on human behavior or human thought.

1.2 Foundations of Artificial Intelligence

Artificial Intelligence did not appear out of nowhere. It is a cross-disciplinary field that sits on the shoulders of giants. Several key disciplines have contributed ideas, viewpoints, and technical tools to the development of AI over hundreds of years.

Key Disciplines and Their Contributions

- **Philosophy (428 BC - Present):** Philosophers made AI conceivable by suggesting that the mind is in some ways like a machine, that it operates on knowledge encoded in some internal language, and that thought can be used to choose the right actions. Major questions include: How does a physical brain give rise to a mental mind? How does knowledge come from experience, and how does it lead to action?
- **Mathematics (c. 800 - Present):** Mathematics provided the tools to manipulate statements of logical certainty as well as uncertain, probabilistic statements. It also set the ground rules for algorithms and computation. The key areas are logic (the formal rules of deduction), computation (what can and cannot be computed), and probability (how to handle uncertainty and make decisions).
- **Economics (1776 - Present):** Economists formalized the theory of rational decision-making. They asked: How should we make decisions to maximize our payout or utility? How should we do this when others are acting against us, or when the payoff is in the far future? Game theory and decision theory are crucial mathematical frameworks inherited from economics.
- **Neuroscience (1861 - Present):** Neuroscience is the study of the nervous system, particularly the brain. It provides insights into how the human brain processes information. A human brain contains about 100 billion neurons, which communicate via electrical and chemical signals. This biological structure directly inspired the creation of Artificial Neural Networks (ANNs) in computer science.
- **Psychology (1879 - Present):** Scientific psychology investigates how humans perceive, think, and act. Cognitive psychology views the brain as an information-processing device, which fits perfectly with the computer science view of AI. Psychology provides models of human behavior and learning that AI researchers attempt to replicate in software.
- **Computer Engineering (1940 - Present):** To build AI, we need powerful machines. Computer engineering provides the silicon hardware, operating systems, and memory capacity required to run heavy AI calculations. Every doubling of hardware speed (historically described by Moore's Law) has opened new doors for what AI systems can realistically process.
- **Control Theory and Cybernetics (1948 - Present):** Control theory focuses on designing machines that regulate their own behavior based on feedback from their environment. For example, a thermostat uses feedback to maintain a stable temperature. Early cybernetics researchers recognized that logical reasoning could be combined with feedback control to build adaptive, goal-seeking machines.
- **Linguistics (1957 - Present):** Modern linguistics began with understanding how language and thought relate. Natural Language Processing (NLP) is a core part of AI that relies heavily on computational linguistics to understand syntax, grammar, semantics, and context in human speech and writing.

1.3 History of Artificial Intelligence

The history of AI is a fascinating journey of high expectations, sudden disappointments, and revolutionary breakthroughs. It can be divided into several distinct eras.

1. The Gestation of AI (1943–1955)

The first work that is now generally recognized as AI was done by Warren McCulloch and Walter Pitts in 1943. They drew on three sources: knowledge of the basic physiology and function of neurons in the brain, a formal analysis of propositional logic, and Alan Turing's theory of computation. They proposed a model of artificial neurons, showing that any computable function could be computed by a network of connected neurons. In 1950, Alan Turing published his seminal paper 'Computing Machinery and Intelligence', introducing the Turing Test, machine learning, genetic algorithms, and reinforcement learning.

2. The Birth of AI (1956)

John McCarthy organized a two-month workshop at Dartmouth College in the summer of 1956. This workshop brought together key figures who would dominate the field for decades, including Marvin Minsky, Claude Shannon, Herbert Simon, and Allen Newell. It was here that McCarthy convinced the attendees to adopt the name 'Artificial Intelligence' for the field, rather than names like 'Cybernetics' or 'Automata Studies'.

3. Early Enthusiasm and Great Expectations (1952–1969)

During this era, computers were writing algebraic proofs, playing checkers, and learning to speak English. Intellectuals predicted that machines would be fully intelligent within a generation. Programs like the Logic Theorist and the General Problem Solver (GPS) simulated human problem-solving methods. John McCarthy invented the Lisp programming language, which became the standard language for AI development.

4. A Dose of Reality and the First AI Winter (1966–1973)

Early researchers failed to realize the 'combinatorial explosion' of search spaces. Methods that worked on simple 'toy problems' completely failed when applied to complex real-world situations. Machine translation efforts funded by the US government were shut down when computers could not resolve simple grammatical ambiguities. In 1969, Marvin Minsky and Seymour Papert published the book 'Perceptrons', proving that single-layer neural networks could not compute simple functions like XOR. This killed funding for neural network research for over a decade, leading to the first 'AI Winter'.

5. Knowledge-Based Systems and Expert Systems (1969–1979)

AI researchers realized that general-purpose search algorithms were too weak. Instead, programs needed domain-specific knowledge. This led to 'Expert Systems', which encoded the rules used by human experts (like doctors or chemists) to solve specific problems. Notable examples include DENDRAL (which analyzed molecular structures) and MYCIN (which diagnosed blood infections and prescribed antibiotics).

6. AI Becomes an Industry (1980–Present)

In the 1980s, expert systems became a commercial success. Large corporations built systems like R1/XCON to help configure computer orders, saving millions of dollars. The Japanese government launched the massive 'Fifth Generation' computer project to build intelligent computers, prompting the US and Europe to set up competing research consortiums.

7. The Return of Neural Networks (1986–Present)

In the mid-1980s, researchers reinvented and popularized the 'backpropagation' learning algorithm. This algorithm allowed multi-layer neural networks to learn internal representations, overcoming the limitations pointed out by Minsky and Papert in 1969. Neural networks once again became highly popular.

8. AI Becomes a Science (1987–Present)

AI moved away from informal 'hack' solutions toward rigorous mathematical and statistical foundations. Researchers began using probability theory, data mining, and machine learning to build systems based on real-world data rather than hand-coded logical rules. Bayesian networks emerged as a powerful tool to handle uncertainty.

9. Deep Learning and Big Data (2011–Present)

The availability of massive datasets (like ImageNet) and cheap parallel compute power (GPUs) led to the rise of Deep Learning. Deep neural networks broke all records in image classification, speech recognition, game playing (such as Google DeepMind's AlphaGo defeating the world champion in Go), and Natural Language Processing. The emergence of the Transformer architecture in 2017 led to Large Language Models (LLMs) like GPT-4, shifting AI into mainstream daily use.

1.4 Limits of AI

While modern AI is incredibly powerful, it has absolute limits that computer scientists, philosophers, and mathematicians have identified. These can be categorized into computational limits, philosophical limits, and practical limits.

1. Computational and Technical Limits

- **Combinatorial Explosion:** Many real-world planning and search problems are NP-complete or NP-hard. As the number of variables increases, the number of possible states grows exponentially, making them impossible to solve exactly, even for supercomputers.
- **Catastrophic Forgetting:** Unlike humans, when artificial neural networks are trained on a new task, they tend to overwrite and completely forget the knowledge acquired from previous tasks.
- **Data Dependency:** Modern Deep Learning models require huge amounts of labeled training data. If high-quality data is unavailable, the AI's accuracy degrades rapidly.

2. Philosophical Limits

- **Gödel's Incompleteness Theorem:** Mathematician Kurt Gödel proved that in any consistent formal mathematical system, there are statements that are true but cannot be proven within the system. Since computers are formal symbol-manipulating systems, some philosophers (like J.R. Lucas) argue that there are mathematical truths that human minds can see but computers can never compute.
- **The Chinese Room Argument:** Proposed by John Searle in 1980, this thought experiment challenges the idea of 'Strong AI'. Imagine a person who doesn't speak Chinese sitting in a closed room with a book of English rules for manipulating Chinese symbols. When Chinese characters are slipped under the door, the person uses the rules to write a response in Chinese characters and slips it back. To people outside, the room appears to understand Chinese. However, the person inside has zero understanding of Chinese—they are simply manipulating symbols. Searle argues that computers are like this person: they have syntax (symbol manipulation) but no semantics (real meaning or understanding).
- **Consciousness and Qualia:** Machines process binary signals but do not have subjective experiences (qualia). A computer can analyze the wavelength of red light, but it does not know what it feels like to see the color red.

3. Practical and Everyday Limits

- **Lack of Common Sense:** AI systems do not have the intuitive background knowledge about how the physical and social world works that a five-year-old child possesses. For example, an AI might know that water is wet, but fail to realize that if you pour water on a book, the book will be ruined.
- **Brittleness:** AI models are often fragile. A tiny, imperceptible change to an image (called an adversarial perturbation) can trick a state-of-the-art computer vision model into misclassifying a stop sign as a speed limit sign.

1.5 Ethics of AI

Because AI systems are increasingly being used to make decisions that affect human lives—such as who gets a job, who gets a loan, or who is arrested—the ethics of AI has become an urgent area of study.

Core Ethical Challenges in AI

- **Fairness and Bias:** AI systems learn from historical data. If the historical data contains human biases (e.g., gender or racial discrimination), the AI will learn and perpetuate these biases. For example, hiring algorithms trained on historical data have been shown to favor male candidates because historically more men were hired.
- **Transparency and Explainability:** Deep neural networks are often described as 'Black Boxes'. We know the inputs and we get the outputs, but we cannot trace exactly *why* the network made a specific decision. Explainable AI (XAI) is a major research area aimed at making AI decisions transparent so that humans can verify and trust them.
- **Accountability and Liability:** If an autonomous self-driving car hits a pedestrian, who is responsible? Is it the software developer, the car manufacturer, the safety driver inside, or the pedestrian? Resolving legal and moral accountability is a major hurdle.
- **Privacy and Surveillance:** AI enables mass surveillance through facial recognition, predictive policing, and tracking of online activity. This raises deep concerns about the erosion of individual privacy rights.
- **Job Displacement:** As AI automates white-collar tasks (like writing, accounting, and basic coding) alongside blue-collar tasks, there is a serious risk of large-scale structural unemployment and widening wealth inequality.
- **Autonomous Weapon Systems (AWS):** The development of military drones and robots that can select and engage targets without human intervention (sometimes called killer robots) raises profound moral questions about letting algorithms decide who lives and who dies.
- **The Alignment Problem:** How do we ensure that superintelligent AI systems will always share human values and act in our best interests? If we give an AI a poorly phrased goal, it might pursue it literally with disastrous consequences (e.g., Nick Bostrom's Paperclip Maximizer thought experiment, where an AI instructed to make paperclips converts the entire planet into paperclips).

1.6 Future of AI

To understand where AI is heading, we must distinguish between the three levels of Artificial Intelligence.

The Three Levels of AI

- **1. Artificial Narrow Intelligence (ANI):** Also known as 'Weak AI', this is AI designed to perform a single, specific task exceptionally well. Examples include search engines, recommendation systems, language translators, and game-playing bots. All existing AI today is Narrow AI.
- **2. Artificial General Intelligence (AGI):** Also known as 'Strong AI', this refers to hypothetical machines that possess human-like intelligence across a broad range of cognitive tasks. An AGI could learn new skills, transfer knowledge from one domain to another, reason abstractly, and solve unfamiliar problems just like a human. AGI does not exist yet.
- **3. Artificial Superintelligence (ASI):** This represents an entity that is vastly smarter than the best human brains in practically every field, including scientific creativity, general wisdom, and social skills.

Key Trends for the Future

- Human-AI Collaboration: Rather than replacing humans, AI will act as a cognitive partner (often called 'augmented intelligence' or 'Centaur systems'), helping doctors diagnose diseases faster, helping programmers write code, and helping scientists discover new materials.
- Quantum Machine Learning: Combining quantum computing with AI algorithms could allow us to process complex computations in seconds that would take traditional supercomputers thousands of years, potentially unlocking breakthroughs in physics and medicine.
- The Technological Singularity: Some futurists (like Ray Kurzweil) predict a point in the future where AGI becomes capable of self-improvement. It would design even smarter versions of itself, leading to an uncontrollable, exponential intelligence explosion, completely altering human civilization.

1.7 AI Components

Just as the human brain has different specialized areas for vision, speech, and planning, a fully realized AI system is built from several interconnected components. These components allow the machine to perceive its environment, process information, make decisions, and take actions.

The Core Components of an AI System

Component	Description / Sub-fields	Role in AI
Perception	Computer Vision, Automatic Speech Recognition, Sensor fusion.	Collects and interprets raw signals from the physical world (images, sounds, depth data) into structured data.
Knowledge Representation	Ontologies, Knowledge Graphs, Semantic Networks.	Stores facts, rules, relationships, and concepts about the world in a format that the computer can manipulate.
Reasoning & Inference	Logical deduction, probabilistic inference, search algorithms.	Processes stored knowledge and current inputs to solve problems, prove theorems, and draw logical conclusions.
Learning	Supervised Learning, Unsupervised Learning, Reinforcement Learning.	Analyzes feedback and data to improve the system's performance over time without being explicitly reprogrammed.
Planning & Decision Making	State-space search, Markov Decision Processes (MDPs).	Formulates a sequence of future actions that will guide the agent from its current state to a desired goal state.
Natural Language Processing	Machine translation, sentiment analysis, text generation.	Enables the system to comprehend and generate human languages, bridging the gap between humans and machines.
Execution / Action	Robotics, motor control, text-to-speech, digital UI.	Translates decisions into physical movements (via actuators) or digital outputs (via screens or API calls).

1.8 AI Architectures

An AI architecture is the underlying system structure that defines how data flows, how different AI components interact, and how computational hardware is organized. It represents the physical or structural framework that runs the AI programs.

To put it simply, Russell and Norvig define the relationship as:

> **The Agent Equation**
> *Agent = Architecture + Program*
>
> *Here, the 'Architecture' is the physical device with sensors and actuators (e.g., a computer, a robot, or a self-driving car). The 'Program' is the software code containing the algorithms that map percepts to actions.*

Common AI Architectural Styles

- **1. Simple Compute Architecture:** A traditional setup where a central processing unit (CPU) or graphics processing unit (GPU) executes a single, sequential agent program that takes inputs, runs them through model weights, and produces outputs. This is typical for deep learning inference on a server.
- **2. Data-Flow (Pipe-and-Filter) Architecture:** Data moves through a linear chain of processing elements. For example, in a voice assistant, raw audio flows to a speech-to-text filter, which outputs text; this text flows to a semantic parser, which outputs an intent; this intent flows to a database filter, which outputs a response; and finally, the response flows to a text-to-speech synthesizer.
- **3. Blackboard Architecture:** Multiple specialized, independent AI modules (often called 'knowledge sources') share a common, global database called the 'blackboard'. When a module sees data on the blackboard that it can work on, it processes it and writes the result back. This architecture is excellent for solving complex, ill-defined problems where multiple different techniques (e.g., rule-based, neural networks, search) must collaborate.
- **4. Layered (Subsumption) Architecture:** Popularized by roboticist Rodney Brooks, this architecture organizes behaviors into hierarchical layers. Lower layers handle urgent, primitive behaviors (like avoiding collisions), while higher layers handle complex, goal-oriented behaviors (like navigating to a room). Higher layers can 'subsume' or override the inputs/outputs of lower layers. This allows robots to react extremely fast to environmental changes.

1.9 Intelligent Agents

In modern AI literature, the concept of the Intelligent Agent is the central unifying theme. Instead of studying isolated algorithms (like pathfinding or neural networks) in a vacuum, we study them as the brains of agents that interact with a world.

What is an Agent?
An Agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.

Sensors, Actuators, Percepts, and Actions

- **Sensors:** The inputs of the agent. For a human, these are eyes, ears, skin, and nose. For a robot, they are cameras, infrared range finders, microphones, and GPS receivers. For a software agent, they are keystrokes, network packets, or file inputs.
- **Actuators:** The outputs of the agent. For a human, they are hands, legs, vocal cords, and fingers. For a robot, they are wheels, robotic arms, display screens, and speakers. For a software agent, they are commands written to a disk, packets sent over a network, or pixels displayed on a monitor.
- **Percept:** The agent's perceptual inputs at any given instant.
- **Percept Sequence:** The complete history of everything the agent has perceived so far. An agent's choice of action can depend on the entire percept sequence, but not on anything it hasn't perceived.

The Agent Function

Mathematically, we describe an agent's behavior by an agent function that maps any given percept sequence to an action:

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

Where P* represents the set of all possible percept sequences (the asterisk means any length from 0 to infinity), and A represents the set of all possible actions.

It is critical to distinguish between the agent function and the agent program. The agent function is an abstract mathematical description of what the agent does. The agent program is a concrete implementation of this function that runs on the physical architecture.

A Simple Example: The Vacuum-Cleaner World

Imagine a simple environment with two rooms, A and B. A vacuum agent can perceive which room it is in (Location: A or B) and whether there is dirt in that room (Status: Dirty or Clean). It can take three actions: Left (move to room A), Right (move to room B), or Suck (vacuum the dirt).

Percept Sequence	Action
[A, Clean]	Right

Percept Sequence	Action
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Clean], [B, Clean]	Left
[A, Clean], [B, Dirty]	Suck

1.10 Agents and Environments

An agent does not exist in a vacuum; it operates inside an environment. The interaction between the agent and its environment is a continuous feedback loop: the agent gathers data from the environment through its sensors, processes this data, and executes actions via its actuators. These actions, in turn, modify the state of the environment, which leads to new sensory inputs.

The Perception-Action Cycle

- Step 1: The environment is in a certain state (which may be fully or partially observable).
- Step 2: The agent's sensors capture the current state as a percept.
- Step 3: The agent's program processes this percept, comparing it with past percept sequences.
- Step 4: The agent selects and triggers an action via its actuators.
- Step 5: The action changes the state of the environment.
- Step 6: The cycle repeats.

> **NOTE**
> The boundary between the agent and the environment is not always the same as the physical boundary of the agent. For example, a self-driving car's actuators (like its engine and brakes) are physically part of the car, but from the perspective of the AI controller, the mechanical response of the brakes is actually part of the environment—if the road is icy, the brakes behave differently, which the AI must perceive and adapt to.

1.11 Good Behavior: Concept of Rationality

What makes an agent's behavior 'good' or 'rational'? Rationality is not about being omniscient (knowing everything) or being perfect. It is about making the best decisions possible with the information you have.

Performance Measures

To evaluate the success of an agent, we use a performance measure. A performance measure is an objective, mathematical criterion used to evaluate how successful an agent is in achieving its goals within a given environment.

An important rule in AI is that performance measures should be designed based on what is actually desired in the environment for the user, rather than how the agent behaves. For example, if we measure a vacuum agent's performance by how much dirt it collects, a rational agent could clean up a room, dump the dirt back onto the floor, and clean it up again to maximize its score. Instead, the performance measure should reward a clean floor over time.

What Determines Rationality?

Rationality at any given time depends on four factors:

1. The performance measure that defines the criterion of success.
2. The agent's prior knowledge of the environment.
3. The actions that the agent can perform.
4. The agent's percept sequence to date.

> **Definition of a 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.*

Omniscience vs. Rationality

We must carefully distinguish between rationality and omniscience. An omniscient agent knows the *actual* outcome of its actions and acts accordingly; but omniscience is impossible in the real world. Rationality maximizes *expected* success, not actual success. For example, if I look both ways before crossing a street, see no cars, and step onto the road, I am acting rationally. If a quiet piano suddenly falls from a window and hits me, my action was not successful, but it was still rational based on my percepts.

Information Gathering, Learning, and Autonomy

A rational agent is not static. To act rationally, it must perform two main tasks:

- **Information Gathering (Exploration):** Sometimes, the rational action is to gather information rather than immediate payoff. For example, looking before crossing the street is an information-gathering action.
- **Learning:** An agent must learn from what it perceives so that it can adapt to new environments. A vacuum agent that learns where dirt is most likely to appear is more rational than one that simply guesses.
- **Autonomy:** An agent is autonomous if its behavior is determined by its own experience (learning and adapting) rather than relying completely on the built-in instructions provided by the programmer at the start.

1.12 Nature of Environments

To design a rational agent, we must first understand the environment it will operate in. We describe the agent's task environment using the PEAS framework, and analyze it using seven key properties.

The PEAS Framework

PEAS stands for Performance, Environment, Actuators, and Sensors. Before writing any code, we must specify these four aspects for our agent.

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi Driver	Safe, fast, legal trip, comfortable, maximize profits.	Streets, highways, pedestrians, other vehicles, weather.	Steering wheel, accelerator, brakes, horn, display.	Cameras, radar, GPS, speedometer, engine sensors.
Medical Diagnosis	Healthy patient, minimize costs, avoid lawsuits.	Patient, hospital staff, laboratory.	Screen displays (diagnoses, treatments, tests).	Keyboard (symptoms, test results, history).
Refinery Controller	Maximize purity, yield, safety, minimize energy.	Pipes, valves, heaters, chemical inputs.	Valves, pumps, heating elements, alarms.	Temperature, pressure, flow rate sensors.

Properties of Task Environments

We classify environments using the following dimensions:

- **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, the environment is fully observable (e.g., Chess). Otherwise, it is partially observable (e.g., Taxi driving, due to blind spots and weather).
- **Single-Agent vs. Multi-Agent:** An agent operating by itself is in a single-agent environment (e.g., Crossword puzzle). If there are other agents in the environment, it is multi-agent. Multi-agent environments can be competitive (e.g., Chess) or cooperative (e.g., driving).
- **Deterministic vs. Stochastic:** If the next state of the environment is completely determined by the current state and the action executed by the agent, the environment is deterministic (e.g., Chess). If there is uncertainty or random chance, it is stochastic (e.g., Taxi driving).
- **Episodic vs. Sequential:** In an episodic environment, the agent's experience is divided into independent atomic episodes. The action taken in one episode does not affect the next (e.g., classifying images). In sequential environments, current decisions affect future states (e.g., Chess, where a bad move now ruins your game later).

- **Static vs. Dynamic:** If the environment can change while the agent is deliberating, it is dynamic (e.g., Taxi driving). If it doesn't change, it is static (e.g., Chess). If the environment doesn't change but the agent's performance score does, it is semi-dynamic (e.g., Chess with a clock).
- **Discrete vs. Continuous:** If the environment has a finite, distinct number of states, actions, and percepts, it is discrete (e.g., Chess). If it varies smoothly over real numbers, it is continuous (e.g., Taxi driving).
- **Known vs. Unknown:** This refers to the agent's knowledge of the 'laws of physics' of the world. In a known environment, the outcomes of all actions are mathematically defined (e.g., solitaire). In an unknown environment, the agent must learn how the world works (e.g., a new video game).

1.13 Types of Agents

AI programs can be structured in five basic ways, depending on how they process information and select actions. These agent programs grow progressively more complex as they are given harder environments.

1. Simple Reflex Agents

A Simple Reflex Agent is the simplest type of agent. It selects actions based *only* on the current percept, completely ignoring the history of past percepts. It operates using condition-action rules: 'if condition, then action'. For example, if a self-driving car sees brake lights ahead, it applies the brakes.

Limitation: They only work if the environment is fully observable. If the environment is partially observable, a simple reflex agent will often get stuck in infinite loops.

2. Model-Based Reflex Agents

To handle partial observability, an agent needs to maintain an internal state. A Model-Based Reflex Agent keeps track of the part of the world it cannot see right now. It does this by combining its current percept with its knowledge of how the world evolves independently and how the agent's own actions affect the world. This knowledge of 'how the world works' is called a model of the world.

3. Goal-Based Agents

Knowing the current state of the environment is not always enough to choose an action. An agent often needs a goal—a description of desirable situations. A Goal-Based Agent combines its knowledge of the current state with its goals to decide which actions will help it achieve those goals. This involves searching and planning, making these agents highly flexible.

4. Utility-Based Agents

Goals are binary: either the goal is reached or it isn't. However, in the real world, there are often many different ways to achieve a goal, some of which are faster, safer, or cheaper. A Utility-Based Agent uses a utility function to map states to a real number representing the agent's 'happiness' or utility. This allows the agent to make rational trade-offs (e.g., choosing a slightly longer route that is much safer).

5. Learning Agents

All the agents described above can be turned into learning agents. A Learning Agent is divided into four main conceptual components:

- **Learning Element:** Responsible for making improvements. It takes feedback from the critic and determines how to modify the performance element to do better.
- **Performance Element:** The component that actually selects external actions. It is what we previously considered to be the entire agent program.
- **Critic:** Evaluates the agent's behavior against an external performance standard and tells the learning element how well the agent is doing.

- **Problem Generator:** Responsible for suggesting new, exploratory actions that will lead to new experiences rather than just repeating safe, known actions.

1.14 Structure of Agents

Below, we show the pseudo-logical structures of the four main reflex-style agent programs, illustrating how their internal logic is written.

1. Simple Reflex Agent Program Structure

Step	Logical Execution Logic
Input	Receive current percept from sensors.
State	Interpret current percept into state representation.
Rule Lookup	Match state against database of Condition-Action rules.
Action Selection	Select the action associated with the matching rule.
Output	Send action command to actuators.

2. Model-Based Agent Program Structure

Step	Logical Execution Logic
Input	Receive current percept from sensors.
Update State	Update internal state based on: (1) previous state, (2) last action taken, (3) how the world changes, (4) current percept.
Rule Lookup	Match updated state against Condition-Action rules.
Action Selection	Select action associated with the matching rule.
Record Action	Store selected action as 'last action' for the next step.
Output	Send action command to actuators.

3. Goal-Based Agent Program Structure

Step	Logical Execution Logic
Input	Receive current percept from sensors.
Update State	Update internal state using percept and model of the world.
Analyze Goals	Predict the future states resulting from candidate actions.
Planning / Search	Select the action that moves the state closer to the goal.
Output	Send action command to actuators.

4. Utility-Based Agent Program Structure

Step	Logical Execution Logic
Input	Receive current percept from sensors.
Update State	Update internal state using percept and model of the world.
Evaluate Utility	Calculate utility value for each possible future state resulting from candidate actions.
Select Best Action	Choose the action that yields the highest expected utility (highest score).
Output	Send action command to actuators.

Unit Summary

This unit introduced the foundational concepts of Artificial Intelligence and Intelligent Agents. The key takeaways are summarized below.

- Artificial Intelligence can be defined in four ways: thinking humanly, thinking rationally, acting humanly, and acting rationally. We focus on acting rationally.
- The Turing Test provides an operational definition of intelligence based on indistinguishability from human behavior.
- AI has deep roots in philosophy, mathematics, economics, neuroscience, psychology, computer engineering, control theory, and linguistics.
- The history of AI has cycled through periods of optimism (Dartmouth workshop, expert systems) and stagnation (AI Winters).
- AI has computational limits (NP-hardness), philosophical limits (Searle's Chinese Room argument, Gödel's incompleteness), and ethical challenges (bias, transparency, job loss).
- An Agent is an entity that perceives its environment through sensors and acts via actuators. The Agent function is defined as f: P* -> A.
- Rationality means choosing actions that maximize the expected value of a performance measure, based on the current percept sequence and prior knowledge.
- Environments are specified using the PEAS (Performance, Environment, Actuators, Sensors) framework.
- Task environments are categorized along seven dimensions: observability, agents, determinism, episodic nature, dynamism, discreteness, and laws (known/unknown).
- The five basic agent types are simple reflex, model-based reflex, goal-based, utility-based, and learning agents.