CSC 830 - Advanced Artificial Intelligence

Lecture 01: Introduction to Artificial Intelligence

Prof. Nwojo Agwu Nnanna

nagwu@nileuniversity.edu.ng

March 3, 2025

Nile University of Nigeria, Abuja

Table of contents

Introduction

Introduction

- Can machines think?
- And if so, how?
- And if not, why not?
- And what does this say about human beings?
- And what does this say about the mind?

What is artificial intelligence?

- There are no clear consensus on the definition of AI
- Here's one from John McCarthy, (He coined the phrase AI in 1956) - see http:// www. formal. Stanford. EDU/ jmc/ whatisai/)

- Q. What is artificial intelligence?
- A. (John McCarthy) It is the science and engineering of making intelligent machines, especially intelligent computer programs. It is related to the similar task of using computers to understand human intelligence, but AI does not have to confine itself to methods that are biologically observable.

- Q. Yes, but what is intelligence?
- A. Intelligence is the computational part of the ability to achieve goals in the world. Varying kinds and degrees of intelligence occur in people, many animals and some machines.

- Q. Yes, but what is intelligence?
- A. Intelligence is the computational part of the ability to achieve goals in the world. Varying kinds and degrees of intelligence occur in people, many animals and some machines.

Intelligence

Intelligence is an umbrella term used to describe a property of the mind that encompasses many related abilities, such as the capacities

- to reason,
- to plan,
- to solve problems,
- to think abstractly,
- to comprehend ideas,
- to use language, and
- to learn.

Intelligence

Intelligence can be defined as the ability for solving problems

- Problem solving is to find the "best" solution in the problem space.
- Reasoning is to interpret or justify solutions or sub solutions.
- Planning is to find ways for solving the problem.
- Thinking abstractly is to simulate the problem solving process inside the system (brain).
- Idea/language comprehension is a way (or means) for data/problem/knowledge representation;
- Learning is the process to find better ways for solving a problem (or a class of problems).

Other possible definitions of AI

- Artificial Intelligence is the study of the *computations* that make it possible to perceive, reason, and act.
    - Hence differs from most psychology because of the greater emphasis on computation, and differs from most computer science because of the emphasis on perception, reasoning, and action.

- AI is a collection of hard problems which can be solved by humans and other living things, but for which we don't have good algorithms for solving.
    - e. g., understanding spoken natural language, medical diagnosis, circuit design, learning, self-adaptation, reasoning, chess playing, proving math theories, etc.

From the perspective of goals

- AI can be viewed as part engineering, part science
  - Engineering goal – to solve real-world problems using AI as a toolkit of ideas about representing knowledge, using knowledge, and assembling systems.
  - Scientific goal – to determine which ideas about representing knowledge, using knowledge, and assembling systems explain various sorts of intelligence.

What is easy? what is hard?

- It's been easier to mechanize many of the high level cognitive tasks we usually associate with "intelligence" in people
  - e. g., symbolic integration, proving theorems, playing chess, some aspect of medical diagnosis, etc.
- It's been very hard to mechanize tasks that animals can do easily
  - walking around without running into things
  - catching prey and avoiding predators
  - interpreting complex sensory information (visual, aural, ...)
  - modeling the internal states of other animals from their behavior
  - working as a team (ants, bees)

- Is there a fundamental difference between the two categories?
- Why some complex problems (e.g., solving differential equations, database operations) are not subjects of AI

History of AI

- AI has roots in a number of scientific disciplines
    - computer science and engineering (hardware and software)
    - philosophy (rules of reasoning)
    - mathematics (logic, algorithms, optimization)
    - cognitive science and psychology (modeling high level human/animal thinking)
    - neural science (model low level human/animal brain activity)
    - linguistics

- The birth of AI (1943 – 1956)
  - Pitts and McCulloch (1943): simplified mathematical model of neurons (resting/firing states) can realize all propositional logic primitives (can compute all Turing computable functions)
  - Alan Turing: Turing machine and Turing test (1950)
  - Claude Shannon: information theory; possibility of chess playing computers
  - Tracing back to Boole, Aristotle, Euclid (logics, syllogisms)

Early enthusiasm (1952 – 1969)

- 1956 Dartmouth conference
  - John McCarthy (Lisp);
  - Marvin Minsky (first neural network machine);
  - Allen Newell and Herbert Simon (GPS);
- Emphasize on intelligent general problem solving
  - GSP (means-ends analysis);
  - Lisp (AI programming language);
  - Resolution by John Robinson (basis for automatic theorem proving);
  - heuristic search (A^*, AO^*, game tree search)

- Emphasis on knowledge (1966 – 1974)
  - domain specific knowledge is the key to overcome existing difficulties
  - knowledge representation (KR) paradigms
  - declarative vs. procedural representation
- Knowledge-based systems (1969 – 1999)
  - DENDRAL: the first knowledge intensive system (determining 3D structures of complex chemical compounds)
  - MYCIN: first rule-based expert system (containing 450 rules for diagnosing blood infectious diseases)
  - EMYCIN: an ES shell
  - PROSPECTOR: first knowledge-based system that made significant profit (geological ES for mineral deposits)

- AI became an industry (1980 – 1989)
  - wide applications in various domains
  - commercially available tools
- Current trends (1990 – present)
  - more realistic goals
  - more practical (application oriented)
  - distributed AI and intelligent software agents
  - resurgence of neural networks and emergence of genetic algorithms

Intelligent Agents

Intelligent agents

- Definition: An Intelligent Agent perceives its environment via sensors and acts rationally upon that environment with its effectors.
- Hence, an agent gets percepts one at a time, and maps this percept sequence to actions.
- Properties
    - Autonomous
    - Interacts with other agents plus the environment
    - Reactive to the environment
    - Pro-active (goal- directed)

Rationality

- An ideal rational agent should, for each possible percept sequence, do whatever actions that will maximize its performance measure based on
  - (1) the percept sequence, and
  - (2) its built-in and acquired knowledge.
- Hence it includes information gathering.
- Rationality $=>$ Need a performance measure to say how well a task has been achieved.
- Types of performance measures: payoffs, false alarm and false dismissal rates, speed, resources required, effect on environment, etc.

Autonomy

- A system is autonomous to the extent that its own behavior is determined by its own experience and knowledge.
- To survive agents must have:
  - Enough built- in knowledge to survive.
  - Ability to learn.

Agent Types

- Table-driven agents
  - use a percept sequence/ action table in memory to find the next action. They are implemented by a (large) lookup table.
- Simple reflex agents
  - are based on condition- action rules and implemented with an appropriate production (rule-based) system. They are stateless devices which do not have memory of past world states.
- Agents with memory
  - have internal state which is used to keep track of past states of the world.
- Agents with goals
  - are agents which in addition to state information have a kind of goal information which describes desirable situations. Agents of this kind take future events into consideration.
- Utility-based agents
  - base their decision on classic axiomatic utility-theory in order to act rationally.

Simple Reflex Agents

- Table lookup of percept- action pairs defining all possible condition-action rules necessary to interact in an environment
- Problems
  - Too big to generate and to store (Chess has about 10^{120} states, for example)
  - No knowledge of non- perceptual parts of the current state
  - Not adaptive to changes in the environment; requires entire table to be updated if changes occur
- Use condition-action rules to summarize portions of the table

A Simple Reflex Agent

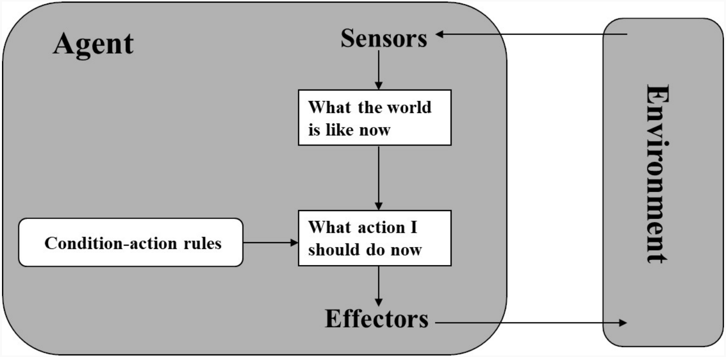

Figure 1: A simple reflex agent

Simple reflex agent with internal state

- Encode "internal state" of the world to remember the past as contained in earlier percepts
- Needed because sensors do not usually give the entire state of the world at each input, so perception of the environment is captured over time. "State"
- used to encode different "world states" that generate the same immediate percept.
- Requires ability to represent change in the world; one possibility is to represent just the latest state, but then can't reason about hypothetical courses of action

Goal-based Agents

- Choose actions so as to achieve a (given or computed) goal.
- A goal is a description of a desirable situation
- Keeping track of the current state is often not enough – need to add goals to decide which situations are good
- Deliberative instead of reactive
- May have to consider long sequences of possible actions before deciding if goal is achieved – involves consideration of the future, “what will happen if I do...?”

Agents with explicit goals

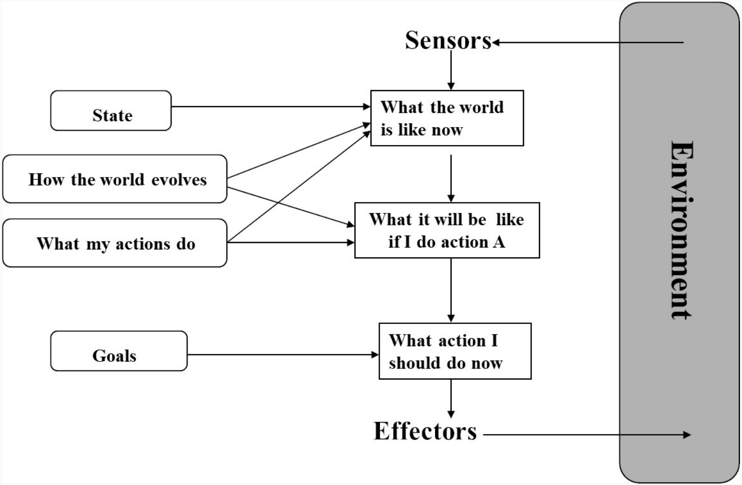

Figure 2: reflex agent with goals

Utility-based Agents

- When there are multiple possible alternatives, how to decide which one is best?
  - A goal specifies a crude distinction between a happy and unhappy state, but often need a more general performance measure that describes "degree of happiness"
  - Utility function $U:$ States $\rightarrow$ Reals indicating a measure of success or happiness when at a given state
  - Allows decisions comparing choice between conflicting goals, and choice between likelihood of success and importance of goal (if achievement is uncertain)

Properties of environments

Properties of environments

- Accessible/ Inaccessible.
  - If an agent's sensors give it access to the complete state of the environment needed to choose an action, the environment is accessible.
  - Such environments are convenient, since the agent is freed from the task of keeping track of the changes in the environment.
- Deterministic/ Nondeterministic.
  - An environment is deterministic if the next state of the environment is completely determined by the current state of the environment and the action of the agent.
  - In an accessible and deterministic environment the agent need not deal with uncertainty.
- Episodic/ Nonepisodic.
  - An episodic environment means that subsequent episodes do not depend on what actions occurred in previous episodes.
  - Such environments do not require the agent to plan ahead.

Properties of environments

- Static/ Dynamic.
  - An environment which does not change while the agent is thinking is static.
  - In a static environment the agent need not worry about the passage of time while he is thinking, nor does he have to observe the world while he is thinking.
  - In static environments the time it takes to compute a good strategy does not matter.
- Discrete/ Continuous.
  - If the number of distinct percepts and actions is limited the environment is discrete, otherwise it is continuous.
- With/ Without rational adversaries.
  - If an environment does not contain other rationally thinking, adversary agents, the agent need not worry about strategic, game theoretic aspects of the environment
  - Most engineering environments are without rational adversaries, whereas most social and economic systems get their complexity from the interactions of (more or less) rational agents.
  - As example for a game with a rational adversary, try the Prisoner's

Summary

Summary

- An agent perceives and acts in an environment, has an architecture and is implemented by an agent program.
- An ideal agent always chooses the action which maximizes its expected performance, given percept sequence received so far.
- An autonomous agent uses its own experience rather than built- in knowledge of the environment by the designer.
- An agent program maps from percept to action & updates its internal state.
  - Reflex agents respond immediately to percpets.
  - Goal-based agents act in order to achieve their goal(s).
  - Utility-based agents maximize their own utility function.
- Representing knowledge is important for successful agent design.
- Some environments are more difficult for agents than others. The most challenging environments are inaccessible, nondeterministic, nonepisodic, dynamic, and continuous.

Possible Research Areas

- Search and optimization
- Knowledge representation
- Reasoning and automatic proving
- Learning and understanding
- Pattern classification / recognition
- Planning
- Problem solving

Search

- Brute force search
  - Depth first search
  - Breadth first search
- Heuristic Search
  - Hill climbing search
  - Best first search
  - A* Algorithm
- Intelligent search
  - Genetic algorithms
  - Meta heuristics