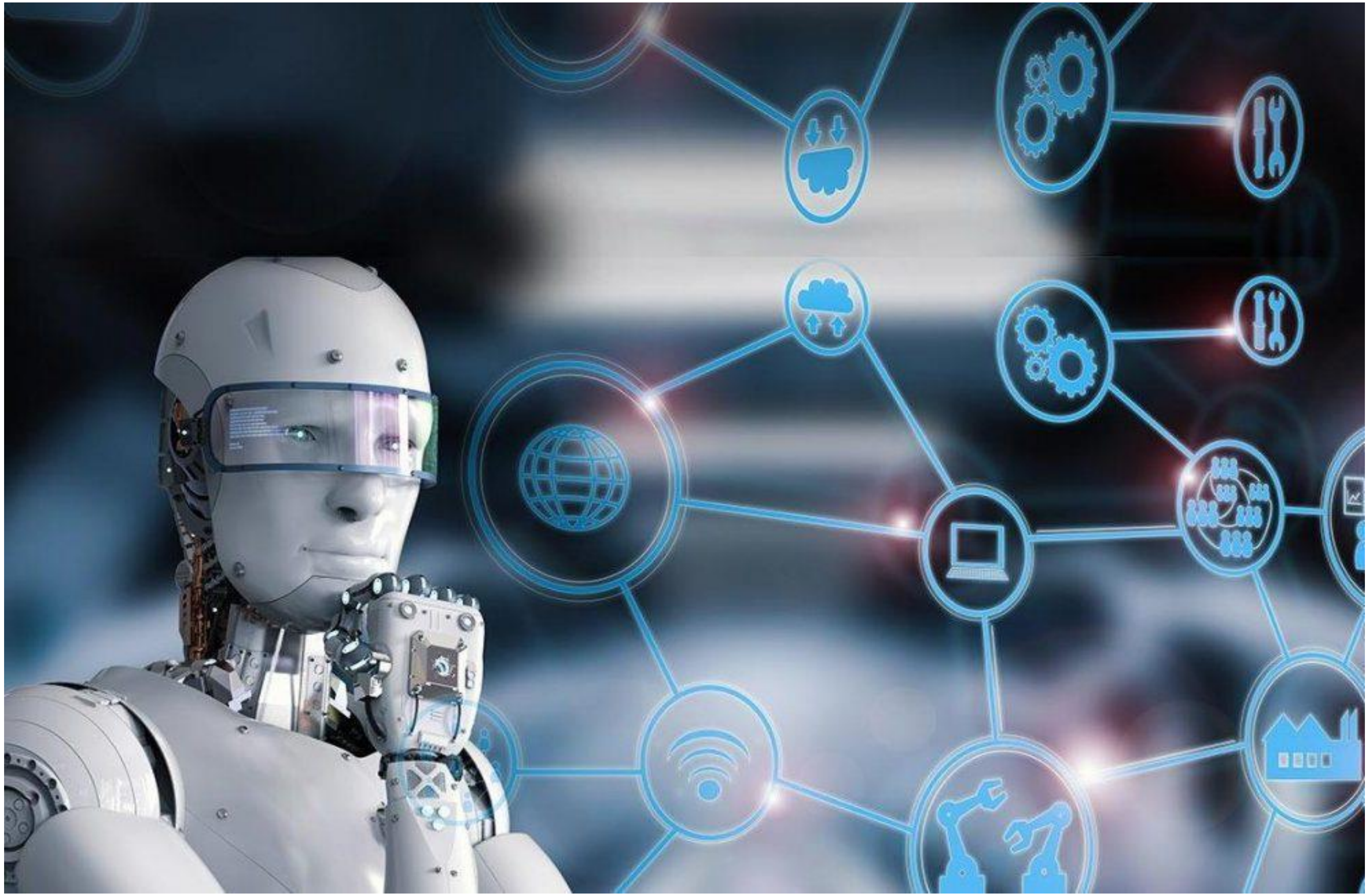
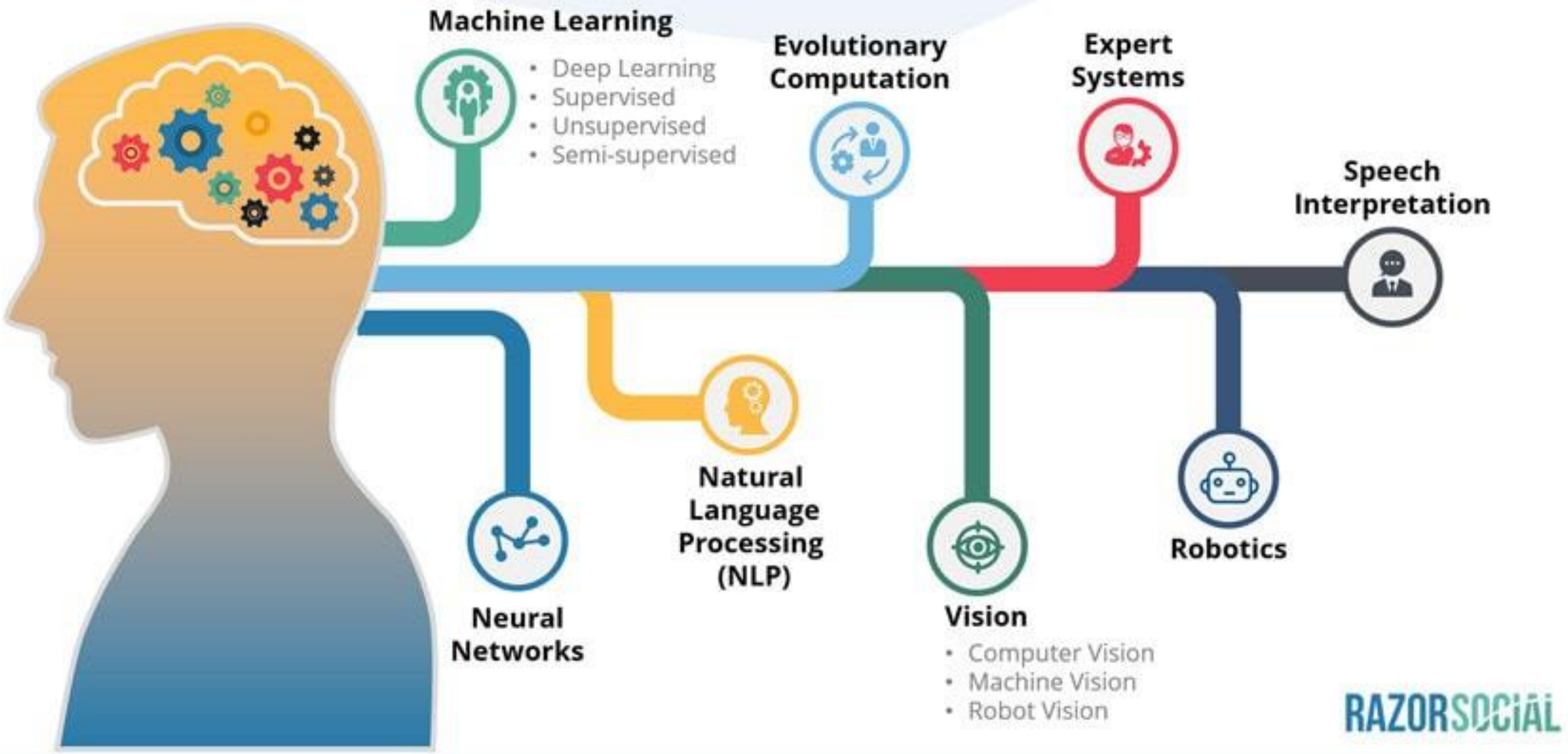
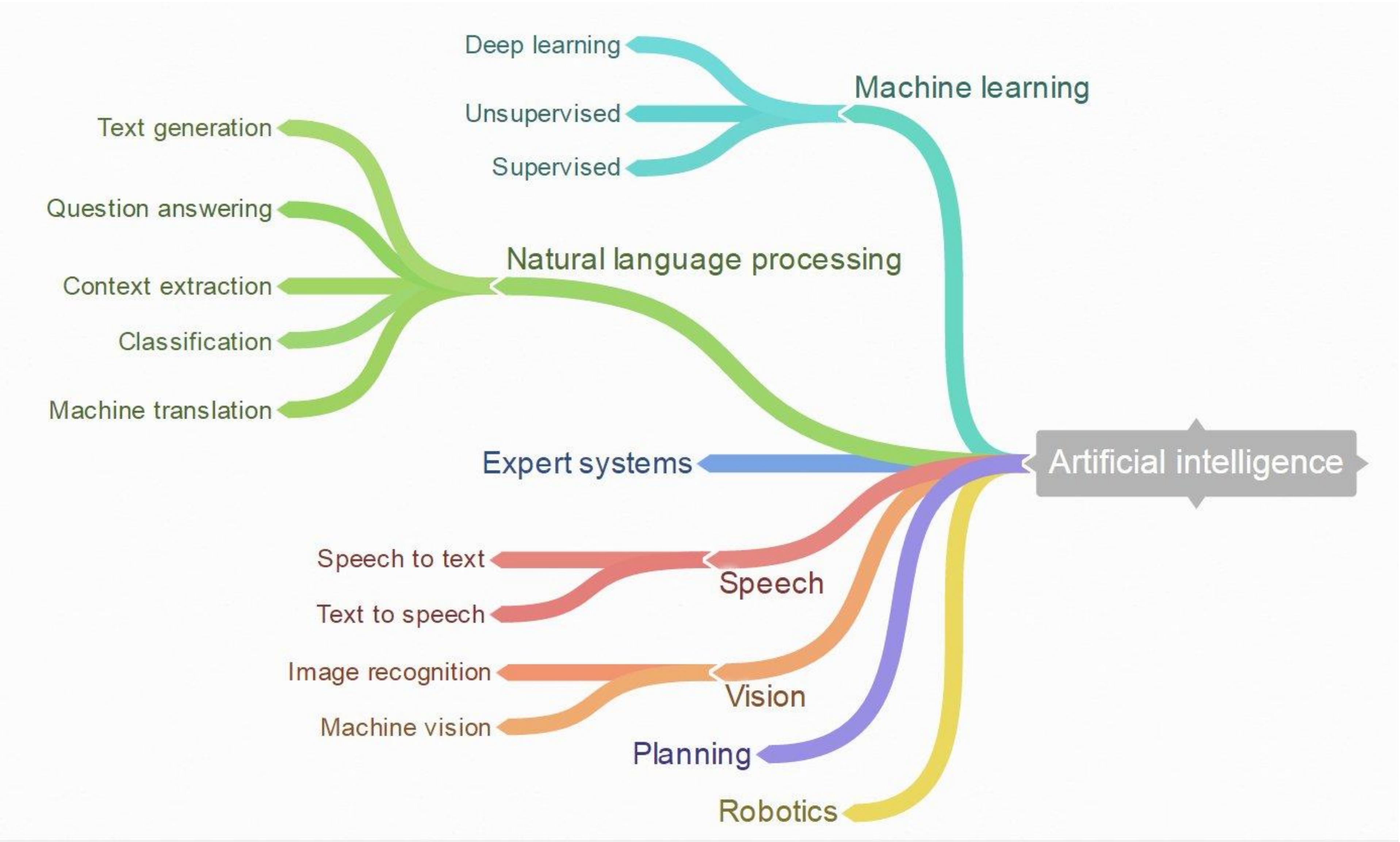
AI

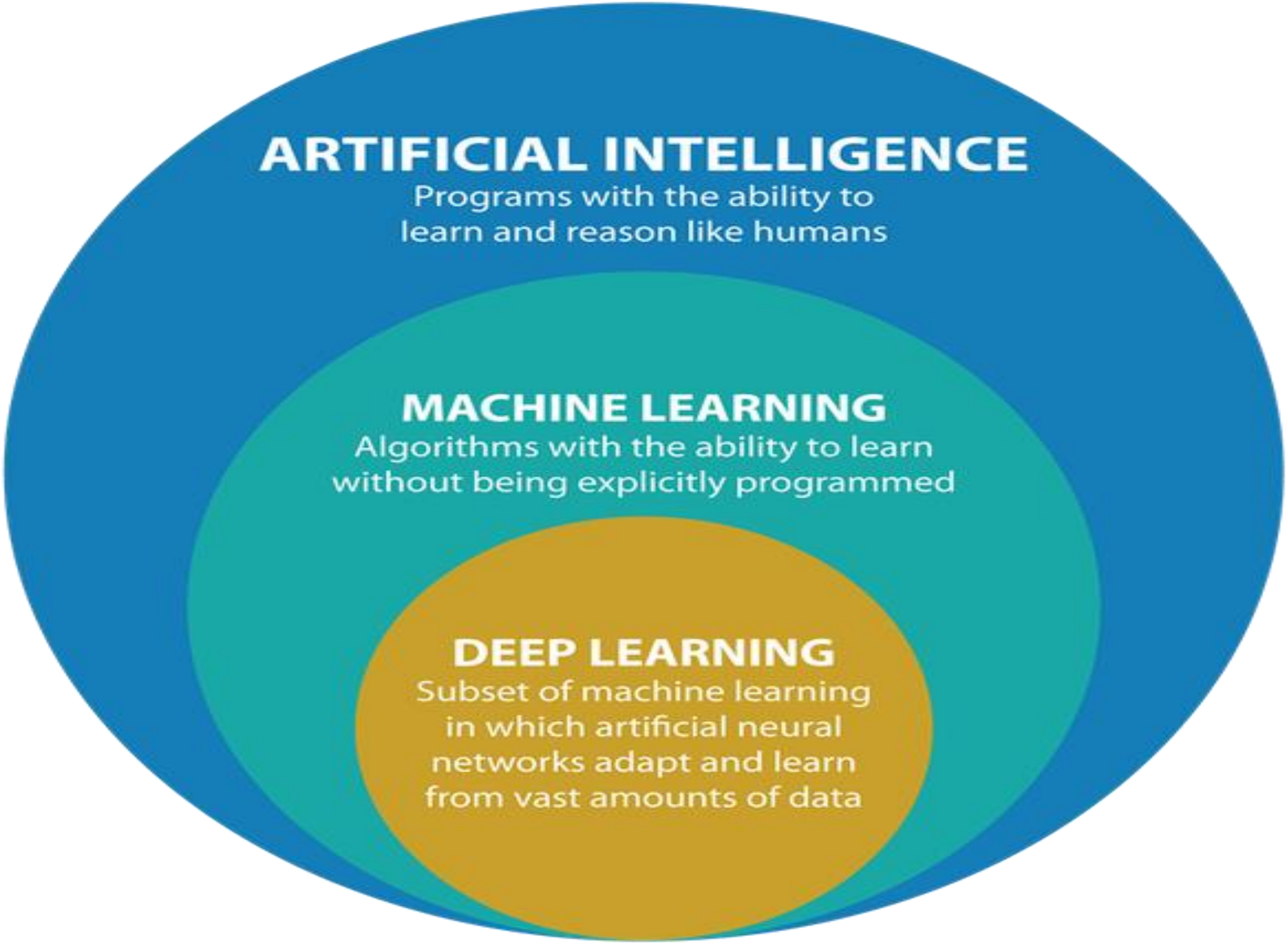
Artificial Intelligence = Making Decision



Elements of AI




Deep learning
Machine learning
Unsupervised
Supervised
Text generation
Question answering
Natural language processing
Context extraction
Classification
Machine translation
Expert systems
Artificial intelligence
Speech to text
Speech
Text to speech
Image recognition
Vision
Machine vision
Planning
Robotics


ARTIFICIAL INTELLIGENCE
Programs with the ability to learn and reason like humans
MACHINE LEARNING
Algorithms with the ability to learn without being explicitly programmed
DEEP LEARNING
Subset of machine learning in which artificial neural networks adapt and learn from vast amounts of data

Acknowledgment

- The slides in this PPT file are composed using the materials supplied by Prof. Stuart Russelland Peter Norvig: They are currently from University of California, Berkeley. They are also the author of the book "*Artificial Intelligence: A Modern Approach*", which is used as the textbook for the course

In this course, we focus on general AI techniques that have been useful in many applications

Foundations of AI

- **Philosophy**
  - 450 BC, Socrates asked for algorithm to distinguish pious from non-pious individuals
  - Aristotle developed laws for reasoning
- **Mathematics**
  - 1847, Boole introduced formal language for making logical inference
- **Economics**
  - 1776, Smith views economies as consisting of agents maximizing their own well being (payoff)
- **Neuroscience**
  - 1861, Study how brains process information
- **Psychology**
  - 1879, Cognitive psychology initiated
- **Linguistics**
  - 1957, Skinner studied behaviorist approach to language learning

What is AI?

- **Intelligence**:
  "Ability to learn, understand and think" (Oxford dictionary)

- **Artificial Intelligence:**
  "Attempts to understand intelligent entities"
  "Strives to build intelligent entities" (Stuart Russel & Peter Norvig)

What is AI?

What is artificial intelligence

It is the science and engineering of making intelligent machines, especially intelligent computer programs

What is intelligence

Intelligence is the computational part of the ability to achieve goals in the world

Why AI?

Competitive Edge

What is AI?

- **Schools of Thought:**

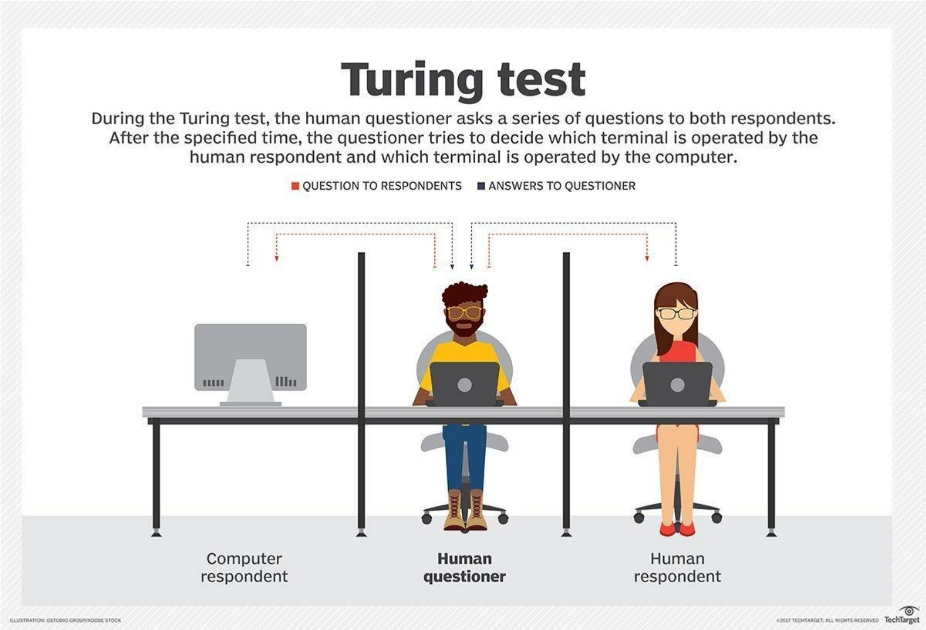
Thinking humanly	**Thinking rationally**
Acting humanly	**Acting rationally**

Acting Humanly: The Turing Test

- Alan Turing (1912-1954)
- "Computing Machinery and Intelligence" (1950)

Acting Humanly: The Turing Test

- Can machines think? Can machines behave intelligently?"
- Operational test for intelligent behavior: **the Imitation Game**



Acting Humanly

Different disciplines

- **Natural language processing:** to enable machine to communicate in human languages
- **Knowledge representation:** to store information provided before or during the interrogation
- **Automated reasoning:** to use the stored information to answer questions and to draw new conclusions
- **Machine learning:** to adapt to new circumstances and to detect and extrapolate patterns.
- **Computer vision:** to perceive objects.
- **Robotics:** to do action in the physical world.

Thinking Humanly: Cognitive Modelling

- **Not content to have a program correctly solving a problem.**

  More concerned with comparing its reasoning steps to traces of human solving the same problem.

- **Requires testable theories of the workings of the human mind: cognitive science.**

  How does a human think?

- **Typical research:**

  General Problem Solver (GPS), Newell & Simon

Thinking Rationally: Laws of Thought

Rational thinking is based on logic inferences.

Aristotle was one of the first to attempt to codify "right thinking", i.e., irrefutable reasoning processes.

✓ Major premise:	**All men are mortal.**
✓ Minor premise:	**Socrates is a man.**
✓ Conclusion:	**Socrates is mortal.**

Formal logic provides a precise notation and rules for representing and reasoning with all kinds of things in the world.

✓ Major premise:	**Men(X) ⇒ Mortal(X)**
✓ Minor premise:	**Men(Socrates)**
✓ Conclusion:	**Mortal(Socrates)**

Acting Rationally

- Acting so as to achieve one's goals, given one's beliefs.
- Does not necessarily involve thinking.
- **Advantages**:
  - More general than the "laws of thought" approach.
  - More amenable to scientific development than human-based approaches.

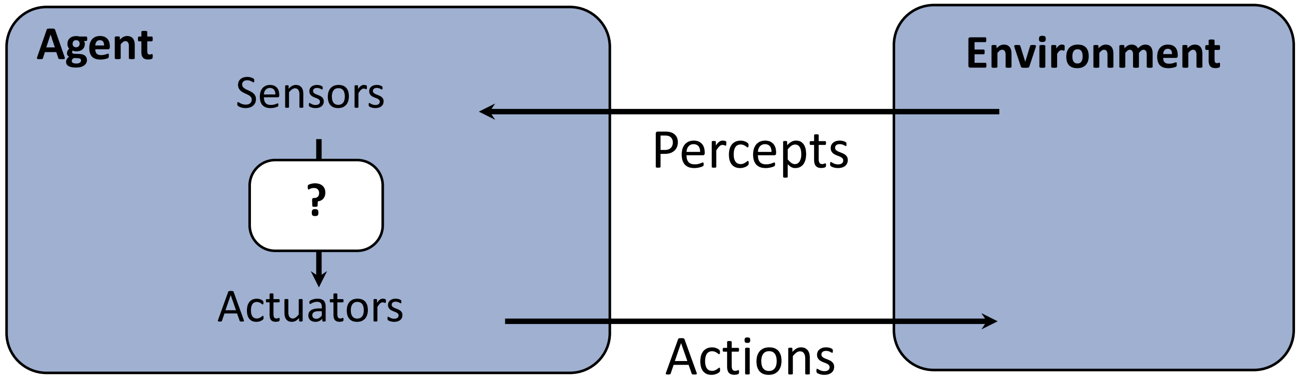
Rational Agent

- **An agent** is anything that can be viewed as perceiving its environment through **sensors** and acting upon that environment through **actuators**.

- **Human agent:**
  - **Sensors**: eyes, ears, …
  - **Actuators**: hands, legs, mouth…

- **Robotic agent**
  - **Sensors**: cameras, range finders, …
  - **Actuators**: motors

Agents and environments



An agent ***perceives*** its environment through ***sensors*** and ***acts*** upon it through ***actuators***

Rationality

A rational agent is an agent which has clear preference, models uncertainty, and acts in a way to maximize its performance measure with all possible actions.

- A ***rational agent*** chooses actions maximize the ***expected*** utility.
  - Today: agents that have a goal, and a cost
    - E.g., reach goal with lowest cost
  - Later: agents that have numerical utilities, rewards, etc.
    - E.g., take actions that maximize total reward over time (e.g., largest profit in $)

- A rational agent does the right thing
- A fixed performance measure evaluates the sequence of observed action effects on the environment

PEAS

- Use PEAS to describe task
  - Performance measure
  - Environment
  - Actuators
  - Sensors

PEAS

- **Agent** : Part picking robot
- **Performance measure** : Percentage of parts in correct bins
- **Environment** : Conveyor belt with parts, bins
- **Actuators** : Jointed arm and hand
- **Sensors** : Camera, joint angle sensors

PEAS

- **Agent** : Medical diagnosis system
- **Performance measure** : Healthy patient, Minimized cost
- **Environment** : Patient, Hospital, Staff
- **Actuators** : Screen display (question, tests, diagnoses, treatment, referrals)
- **Sensors** : Keyboard (for entry of data)

PEAS

- **Agent** : Interactive English tutor
- **Performance measure** : Maximize student's score on test
- **Environment** : Set of students
- **Actuators** : Screen display (exercises, suggestions, corrections)
- **Sensors** : Keyboard, microphone

Features of Environment

As per Russell and Norvig, an environment can have various features from the point of view of an agent:

1. Fully observable vs Partially Observable
2. Static vs Dynamic
3. Discrete vs Continuous
4. Deterministic vs Stochastic
5. Single-agent vs Multi-agent
6. Episodic vs sequential
7. Known vs Unknown
8. Accessible vs Inaccessible

Fully observable vs Partially Observable:

- If an agent sensor can sense or access the complete state of an environment at each point of time then it is **a fully observable** environment, else it is **partially observable**.
- A fully observable environment is easy as there is no need to maintain the internal state to keep track history of the world.
- An agent with no sensors in all environments then such an environment is called as **unobservable**.

Deterministic vs Stochastic:

2. If an agent's current state and selected action can completely determine the next state of the environment, then such environment is called a **deterministic** environment.

- A stochastic environment is random in nature and cannot be determined completely by an agent.
- In a **deterministic**, **fully observable environment**, agent does not need to worry about uncertainty.

Episodic vs Sequential:

- In an **episodic** environment, there is **a series of one-shot actions**, and only the current percept is required for the action.
- However, in Sequential environment, an agent requires memory of past actions to determine the next best actions.

Static vs Dynamic:

- If the environment can change itself while an agent is deliberating then such environment is called a dynamic environment else it is called a static environment.
- Static environments are easy to deal because an agent does not need to continue looking at the world while deciding for an action.
- However for dynamic environment, agents need to keep looking at the world at each action.
  - *Taxi driving is an example of a dynamic environment*
  - *Crossword puzzles are an example of a static environment.*

Single-agent vs Multi-agent

- If only one agent is involved in an environment, and operating by itself then such an environment is called **single agent environment.**
- However, if multiple agents are operating in an environment, then such an environment is called a **multi-agent environment**.
- The agent design problems in the multi-agent environment are different from single agent environment.

Discrete vs Continuous:

- If in an environment there are **a finite number of percepts and actions that can be performed within it**, then such an environment is called a **discrete** environment else it is called continuous environment.
  - *A chess game comes under discrete environment as there is a finite number of moves that can be performed.*
  - *A self-driving car is an example of a continuous environment.*

Known vs Unknown

- Known and unknown are not actually a feature of an environment, but it is an agent's state of knowledge to perform an action.
- In a **known** environment, **the results for all actions are known to the agent.**
- While in unknown environment, agent needs to learn how it works in order to perform an action.
- It is quite possible that a known environment to be partially observable and an Unknown environment to be fully observable.

Accessible vs Inaccessible

- If an agent can **obtain complete and accurate information** about the state's environment, then such an environment is called an **Accessible** environment else it is called inaccessible.
- An empty room whose state can be defined by its temperature is an example of an accessible environment.
- Information about an event on earth is an example of Inaccessible environment.

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock						
Chess without a clock						

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker						

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi

Environment Examples

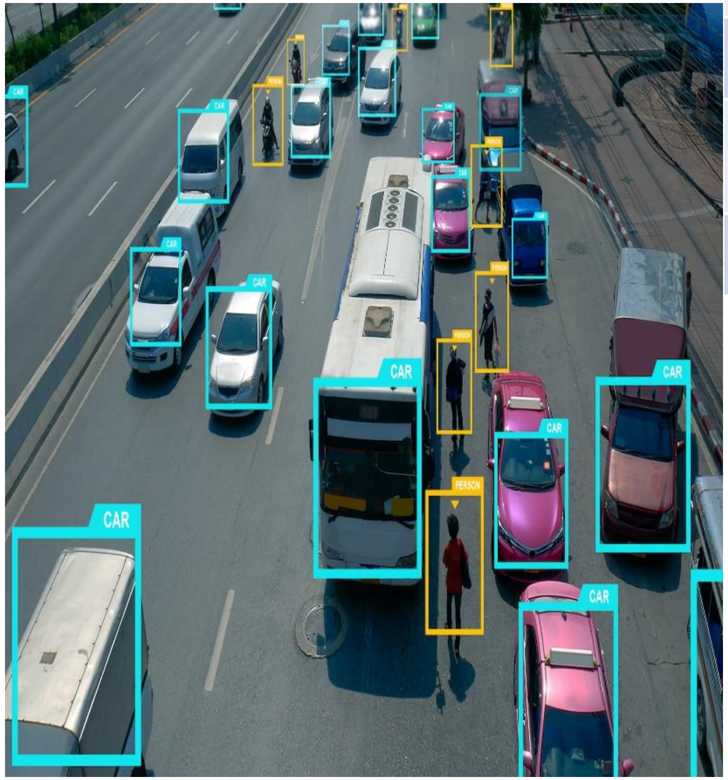
Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon						

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi

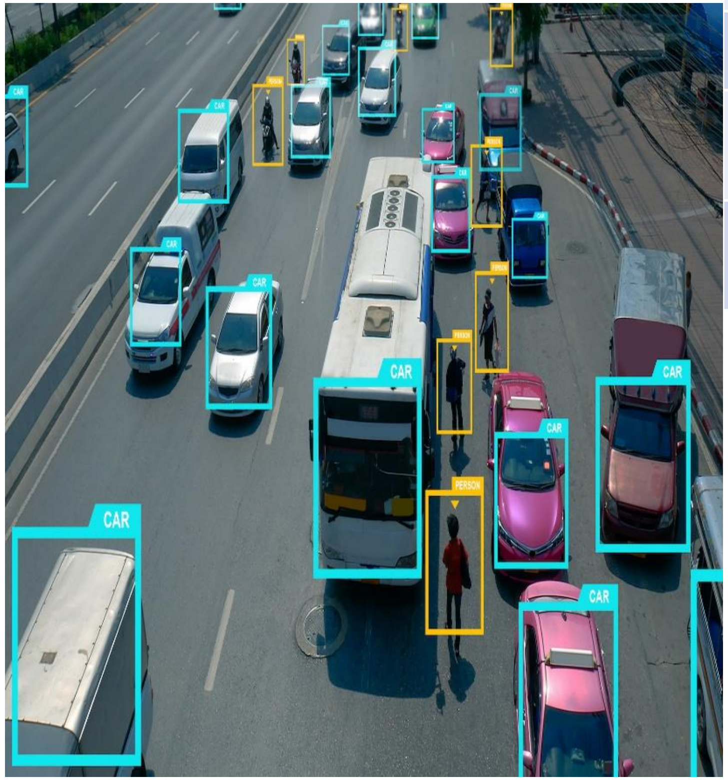
Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving						



Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi



Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis						

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single

Environment Examples

Environment	Observabl e	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis						

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single
Robot part picking	Fully	Deterministic	Episodic	Semi	Discrete	Single

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single
Robot part picking	Fully	Deterministic	Episodic	Semi	Discrete	Single
Interactive English tutor						

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single
Robot part picking	Fully	Deterministic	Episodic	Semi	Discrete	Single
Interactive English tutor	Partial	Stochastic	Sequential	Dynamic	Discrete	Multi

Agent Types

- Types of agents ***(increasing in generality and ability to handle complex environments)***
  - Simple reflex agents
  - Reflex agents with state
  - Goal-based agents
  - Utility-based agents
  - Learning agent

Acting Rationally

The intelligent agent approach

Classification of agents based on intelligent behaviors:

- Simple reflex agents If temperature>27 turn on the air-conditioner.
- Model-based reflex agents Turn on the air-conditioner at 9:00AM. (No thermometer)
- Goal-based agents Keep people in the room comfortable.
- Utility-based agents Measure comfortable with a comfortable utility function. (not just comfortable or uncomfortable.)
- Learning agents Learn the definition of “comfortable” with some feedbacks.

Simple Reflex Agent

- Use simple "**if then**" rules
- Can be short sighted

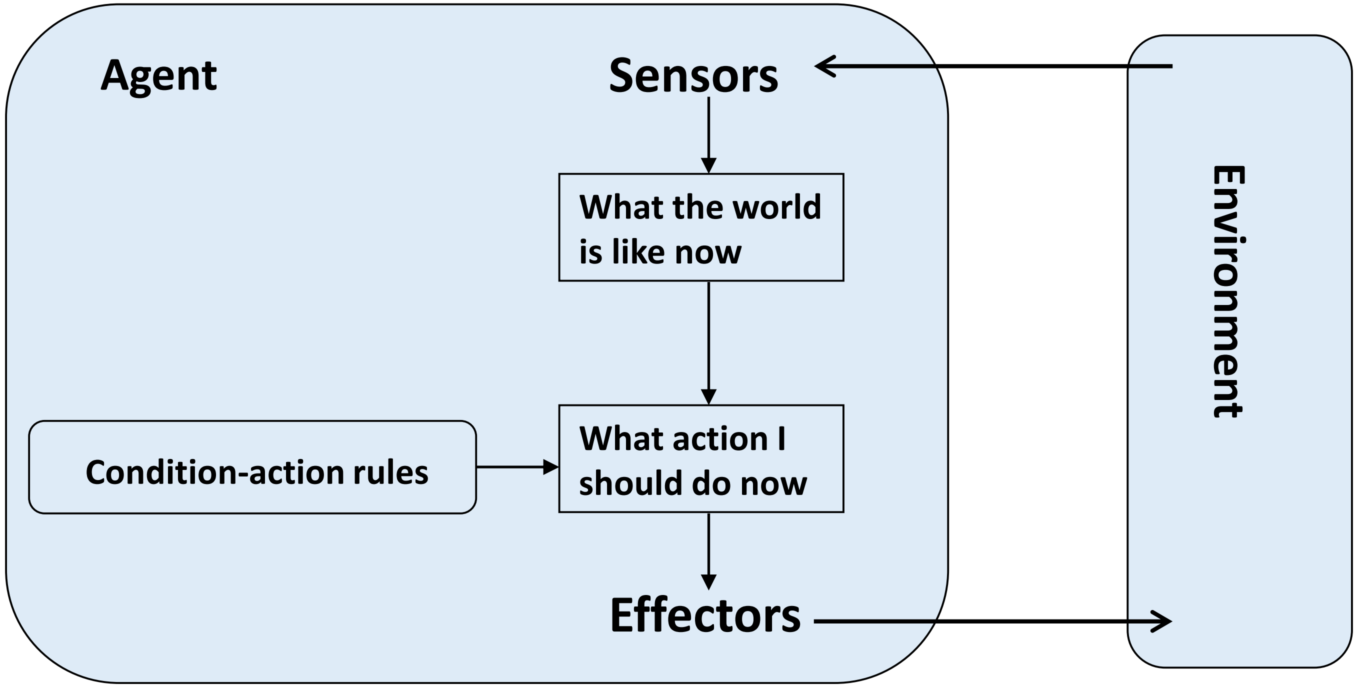
SimpleReflexAgent(percept)

state = InterpretInput(percept)

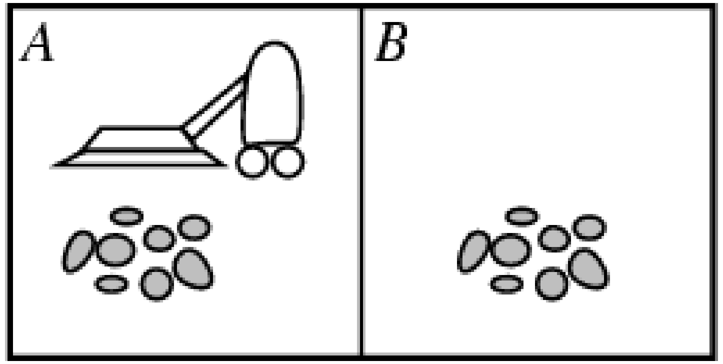
rule = RuleMatch(state, rules)

action = RuleAction(rule)

Return action



Example: Vacuum Agent



- **Performance**?
  - 1 point for each square cleaned in time T?
  - #clean squares per time step - #moves per time step?
- **Environment**: vacuum, dirt, multiple areas defined by square regions
- **Actions**: left, right, suck, idle
- **Sensors**: location and contents [A, dirty]

If status=Dirty then return Suck

else if location=A then return Right

else if location=B then right Left

Rational is not omniscient

Environment may be partially observable

Rational is not clairvoyant

Environment may be stochastic

Thus Rational is not always successful

if car-in-front-is-braking then brake

if light-becomes-green then move-forward

if intersection-has-stop-sign then stop

Reflex Agent With State

- Store previously-observed information
- Can reason about unobserved aspects of current state

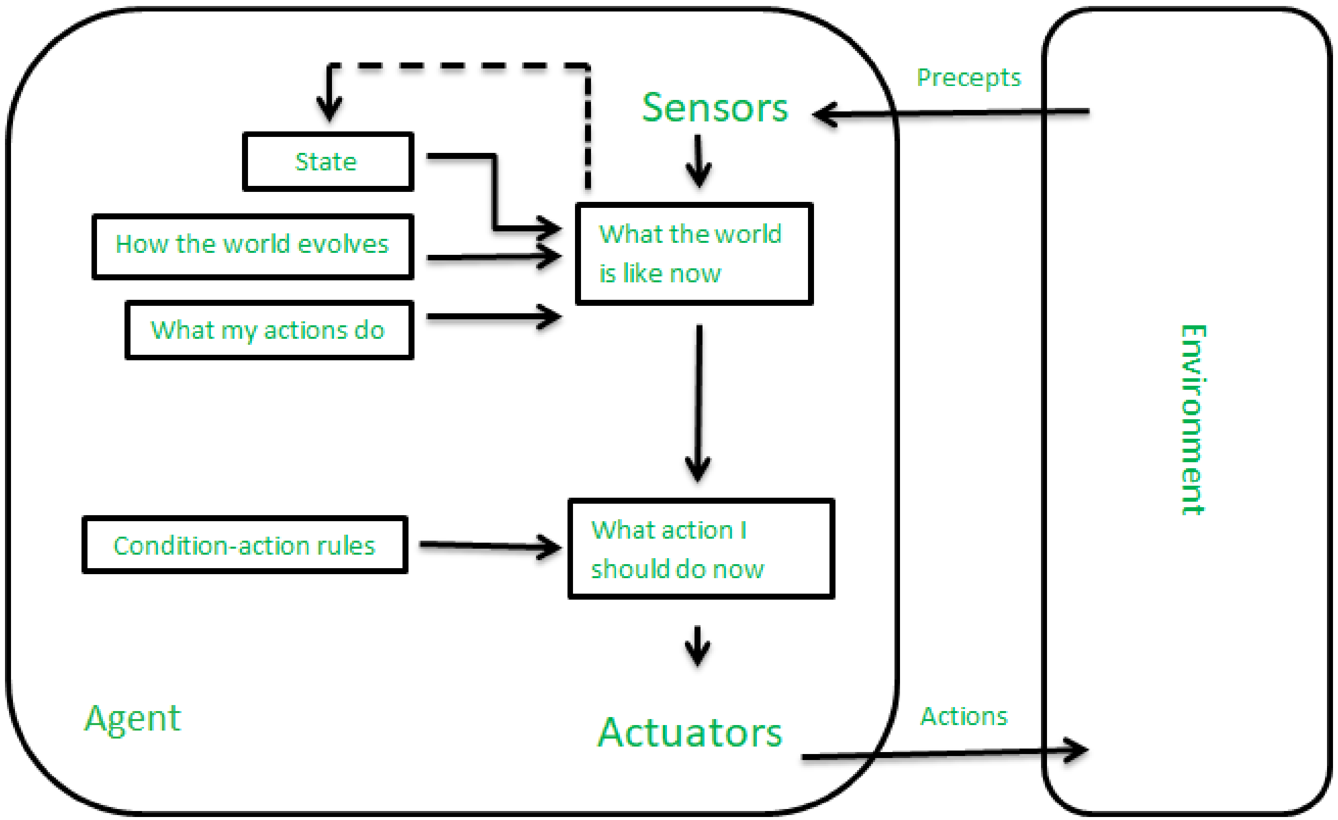
ReflexAgentWithState(percept)

state = UpdateDate(state,action,percept)

rule = RuleMatch(state, rules)

action = RuleAction(rule)

Return action



It must also remember which actions it has taken

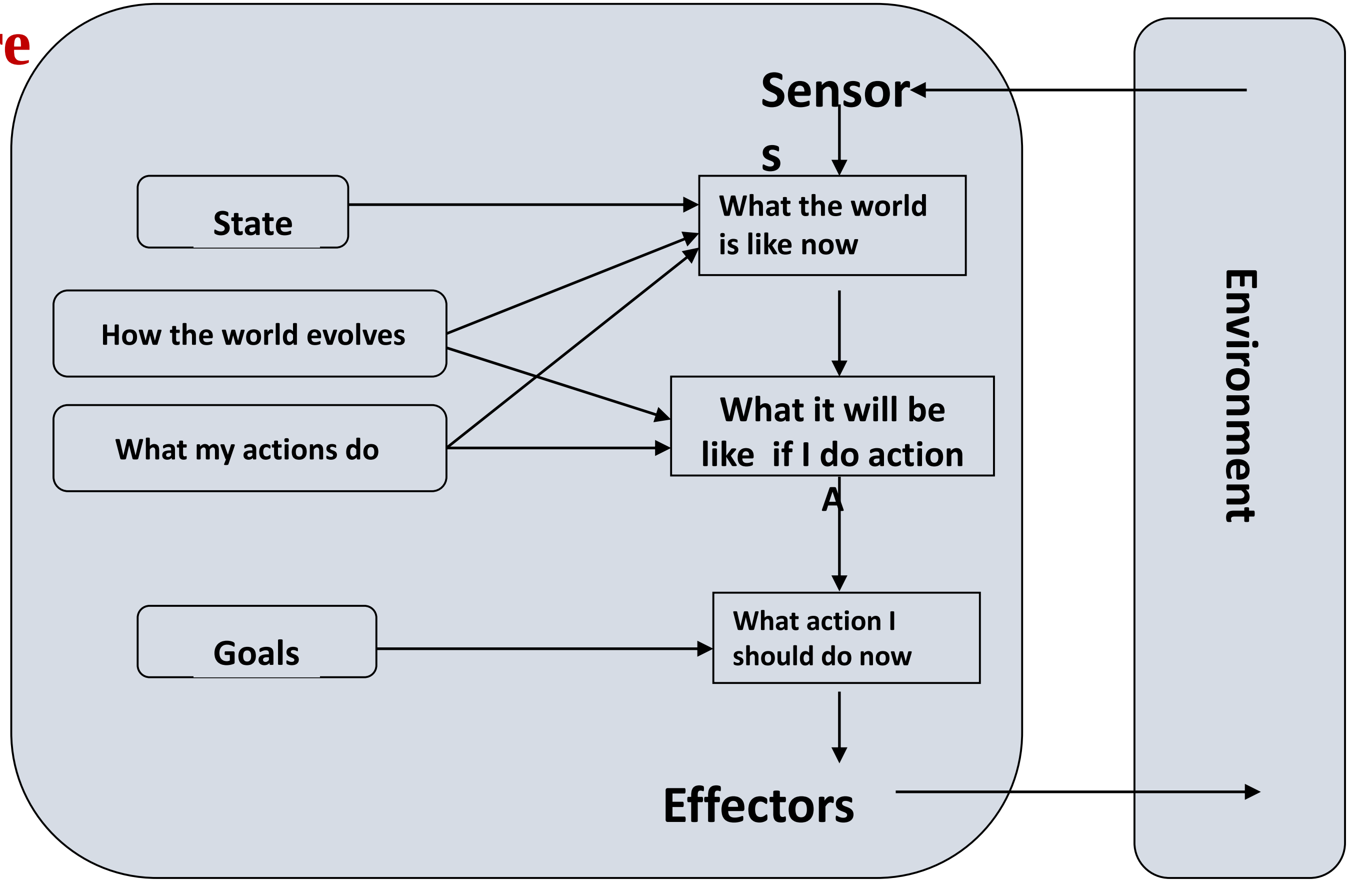
Ex: car in front signals it is turning left

Ex: loaded/unloaded passenger

Goal-Based Agents

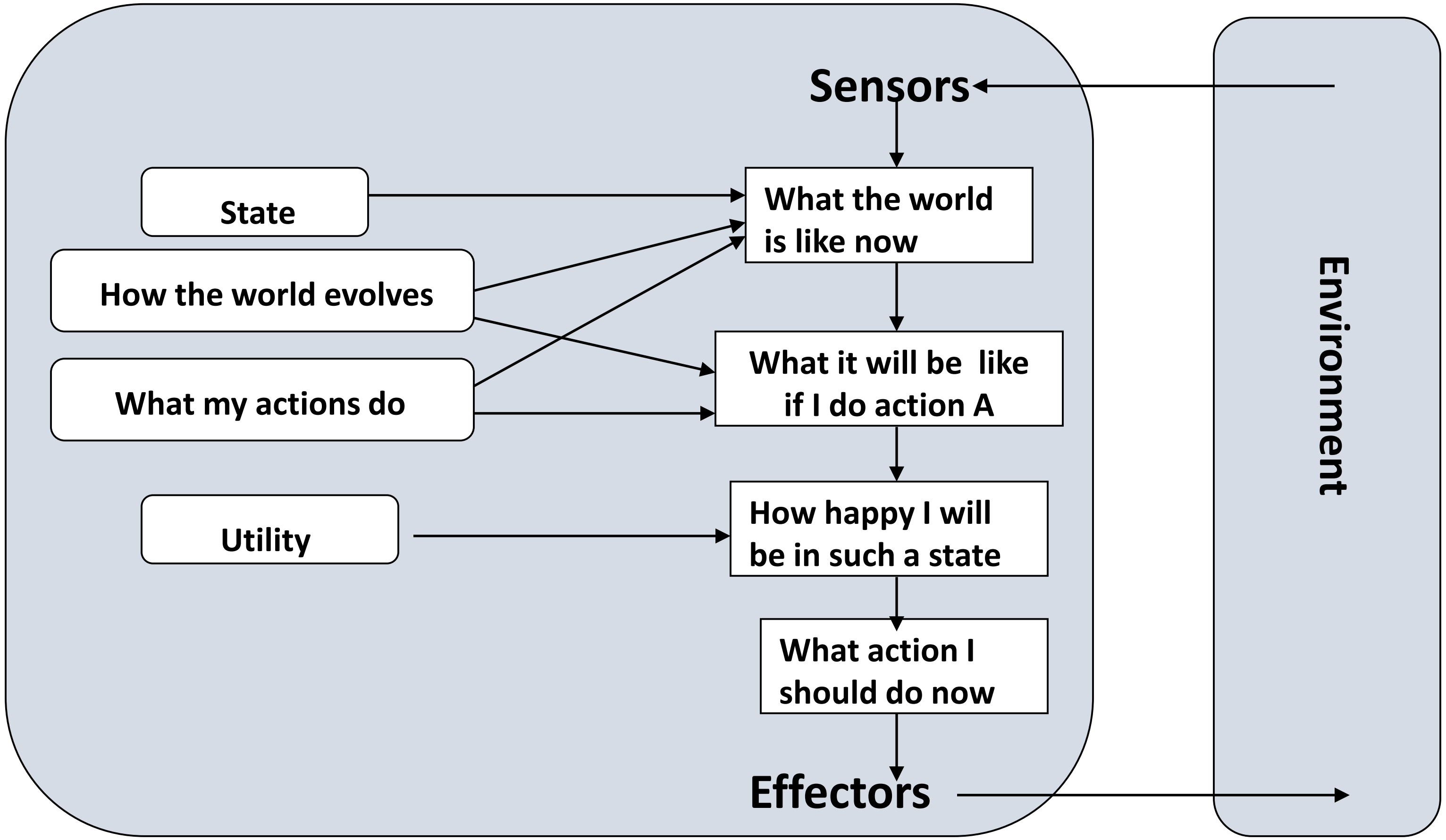
Goal-based Agents are much more flexible in

- responding to a changing environment
- accepting different goals



Utility-Based Agents

- Evaluation function to measure utility f(state) -> value
- Useful for evaluating competing goals



Utility- Based Agent

- 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 --> 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)