

CSCI 3202: Intro to Artificial Intelligence

Lecture 2: Agents

Rachel Cox
Department of Computer Science

Starting at 10:10 am



What is an agent?

An **agent** is an entity that perceives and acts.

- Perceives via sensors (percepts)
- Acts via actuators (actions)

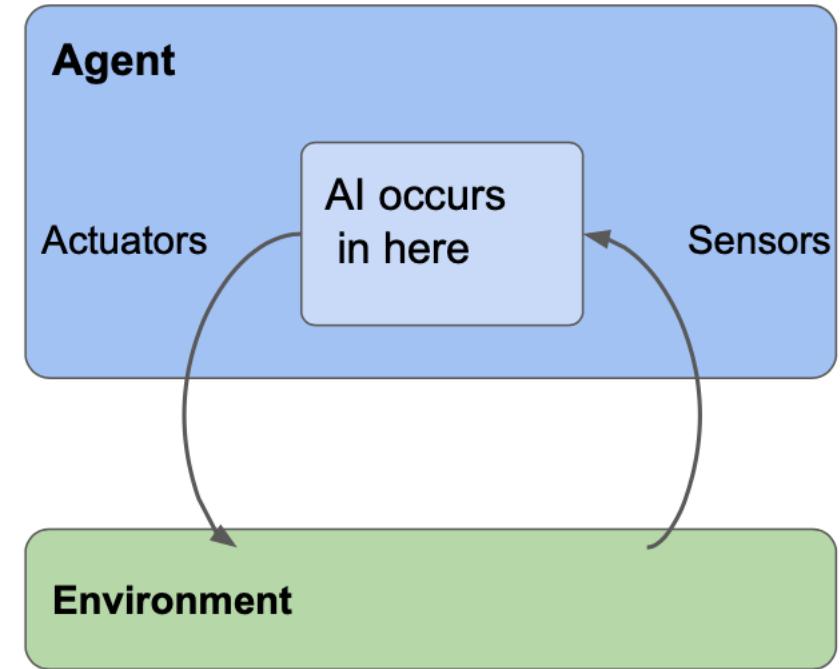
A **percept** is the agent's perceptual inputs at any given instant.

A **percept sequence** is the complete history of everything the agent has ever perceived.

→ **Agent function:** maps a percept sequence to an action

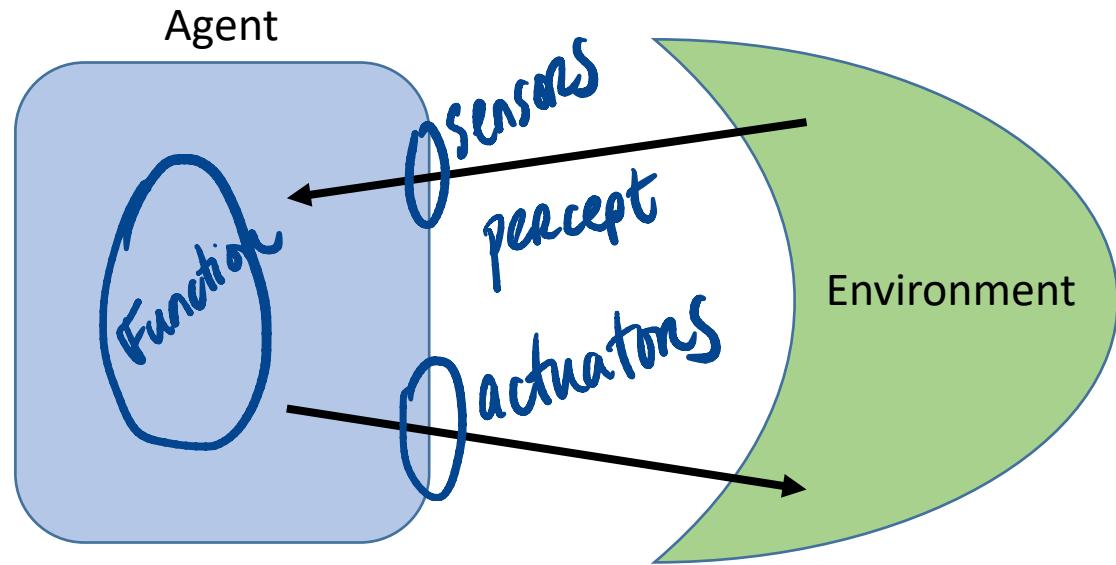
- External characterization is the function.
- Implemented by the **agent program**

A **rational agent** is one that does the “right” thing.



What is an agent?

Perception-Action Cycle

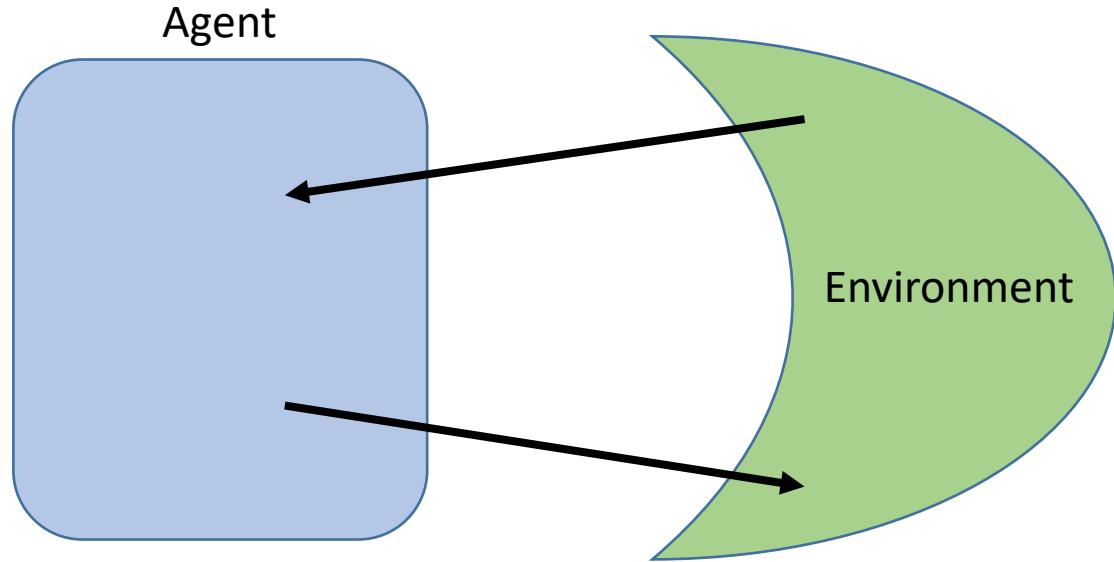


How does the agent make decisions based on the sensory data?

- ❖ Can't yet build one system that does it all, but we can build a system that does one thing well.

What is an agent?

Example: Human Agent!



sensors: eyes, ears, nose
5 senses
touch

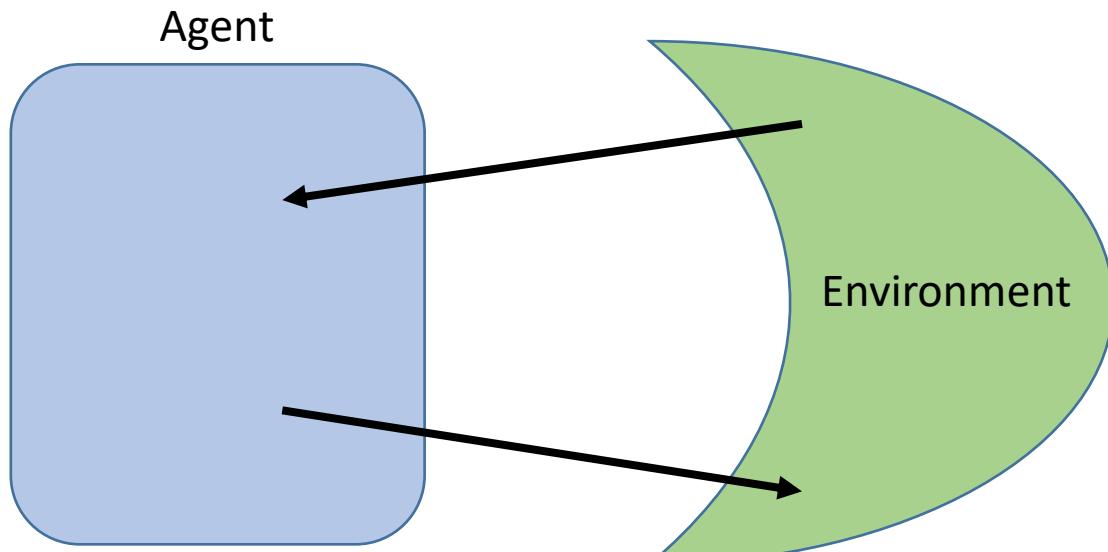
actuators: muscles
feet
hands
;

What is an agent?

In this class - we will work on the function that drives the agent's behavior. You can't generally control the environment, but you can control the agent.

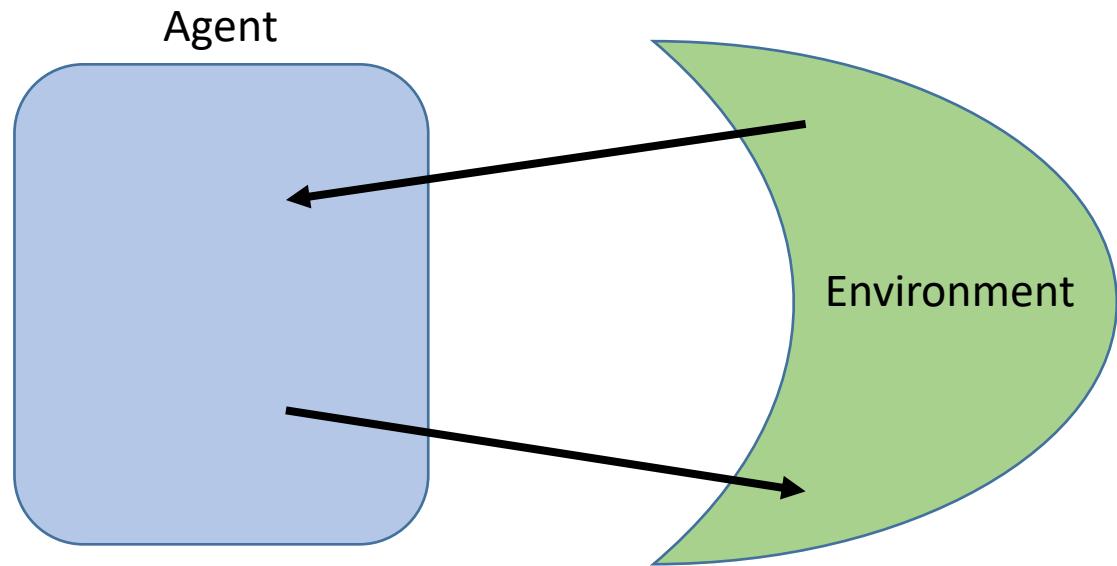
Stock

Example: Trading Agents



sensors: Stock prices, news
actuations: buy/sell

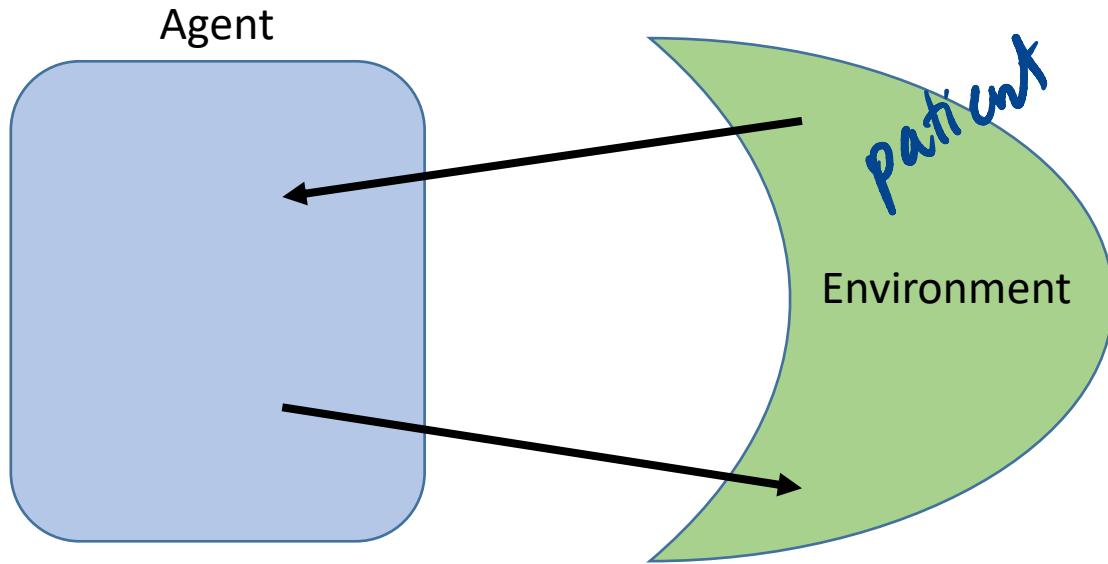
Example: Robotics



Sensors: camera, gyroscope
actuators: pistons, gears, wheels, levers.

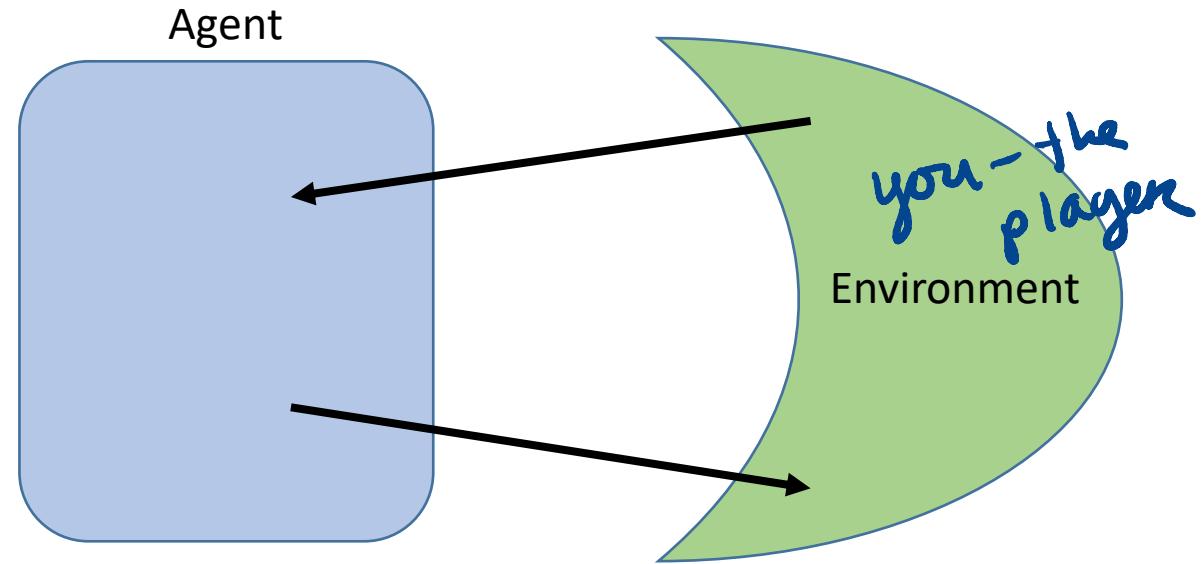
What is an agent?

Example: Medicine



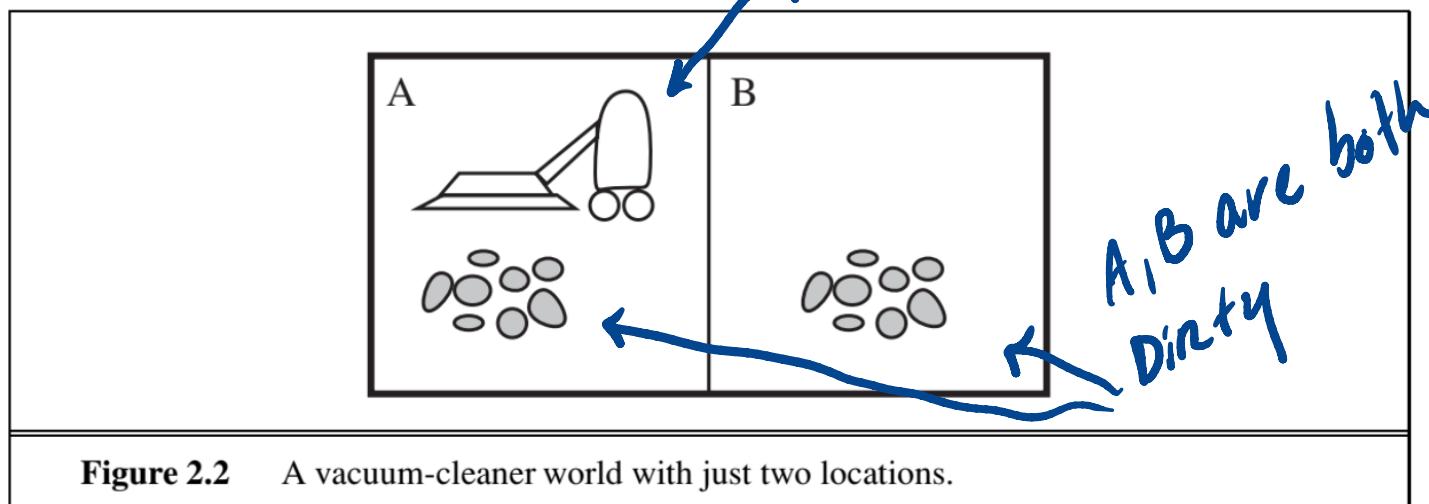
sensors: medical data, respiratory rate, oxygen level
actuators: diagnostics, availability of rooms

Example: Games



sensors: actions
actuators: move of computer

What is an agent?



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.



What is an agent?

The **performance measure** evaluates the behavior of the agent in an environment. A rational agent acts so as to maximize the expected value of the performance measure given the precept sequence it has seen so far.

Example: Vacuum-cleaner agent

- 1) Suppose we measure performance strictly by the amount of dirt cleaned up in a single eight-hour shift.
 - if floor is already clean, it might keep running unnecessarily
 - cat litter box
 - repeatedly dump dirt + clean it up
- 2) Reward agent for having clean floor. One point for each clean square, with a penalty for electricity usage.



What is an agent?

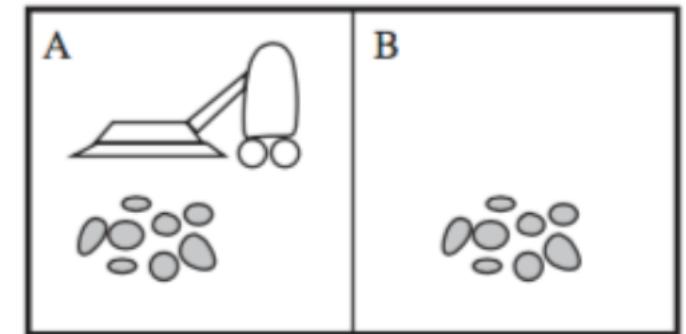
Design performance measures to track what we actually want in the environment.

Example: Suppose our vacuum agent is programmed such that:

If the current square is dirty, vacuum it.

If the current square is clean, move to the other square.

Is this agent rational?



What is an agent?

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 ~~precept~~ sequence and whatever built-in knowledge the agent has.

perce~~p~~t

Rationality based on the following four principles:

- 1) performance measure (utility)
- 2) environment familiarity
- 3) actions that are possible
- 4) sequences of percepts (memory)

What is an agent?

Example: Suppose our vacuum agent is programmed such that:

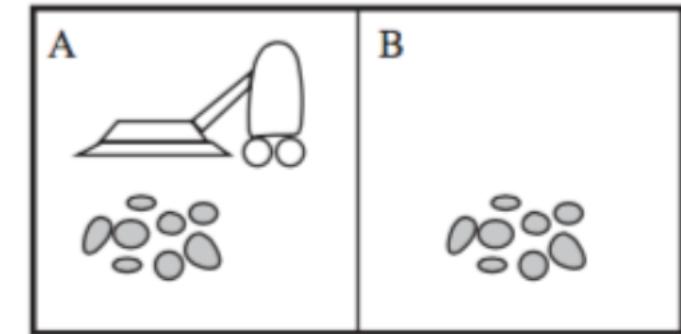
If the current square is dirty, vacuum it.

If the current square is clean, move to the other square.

Is this agent rational?

- Suppose both squares are dirty?
 - 1) clean current square
 - 2) move to other square
 - 3) clean

} 3 Δt to get the room clean



Task Environments

Task environments are essentially the “problems” to which rational agents are the “solutions”.

P E A S : Performance, Environment, Actuators, Sensors

Example: PEAS description of the task environment for a taxi

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi Driver	safe, fast, legal, comfortable trip, maximize profit	roads, other traffic, pedestrians, customers	steering, accelerator, brake, signal, horn, display	cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors, keyboard



radar

Task Environments

Agent can observe all relevant aspects of the environment.

Fully Observable

No need to maintain any internal state to keep track of the world.

What your agent can sense at any given time is completely sufficient to make the optimal decision.

Some parts missing from sensor data.

Partially Observable

e.g. noisy environment, inaccurate sensors

Need memory on the part of the agent to make the optimal decision.

Can't make any observations but still must make a choice.

Unobservable

Task Environments

Deterministic vs. Stochastic

- No randomness when you move a piece
- Effect of moving a piece is completely predetermined.
- Can't predict the outcome of the dice.
- There is randomness present in the environment.



Task Environments

Discrete vs. Continuous

- Finitely many action choices.
- Infinitely many action choices.



Task Environments

Benign vs. Adversarial

- Environment might be stochastic, but there is no objective that would contradict your own objective.
- An opponent is out to get you. (like in games)

Task Environments

Single Agent vs. Multi-Agent



Task Environments

Episodic vs. Sequential

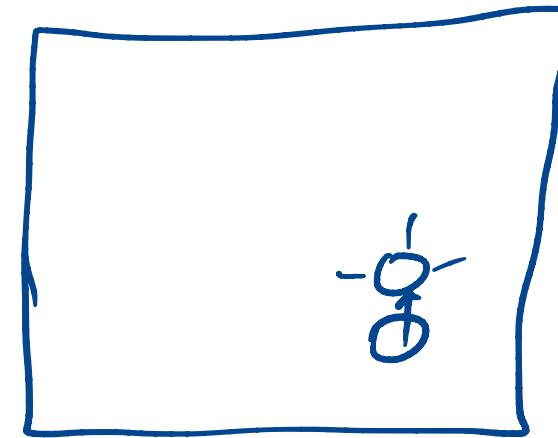
Agents experience is divided into atomic episodes.

In each episode, the agent receives a percept and then performs a single action.

Next episode is not dependent on the actions taken in previous episode.

e.g. classification tasks

Current decision could affect all future decisions.



Task Environments

Static vs. Dynamic

- The environment remains constant.
- The environment can change while the agent is deliberating.

Task Environments

Example: Checkers

Partially or Fully Observable

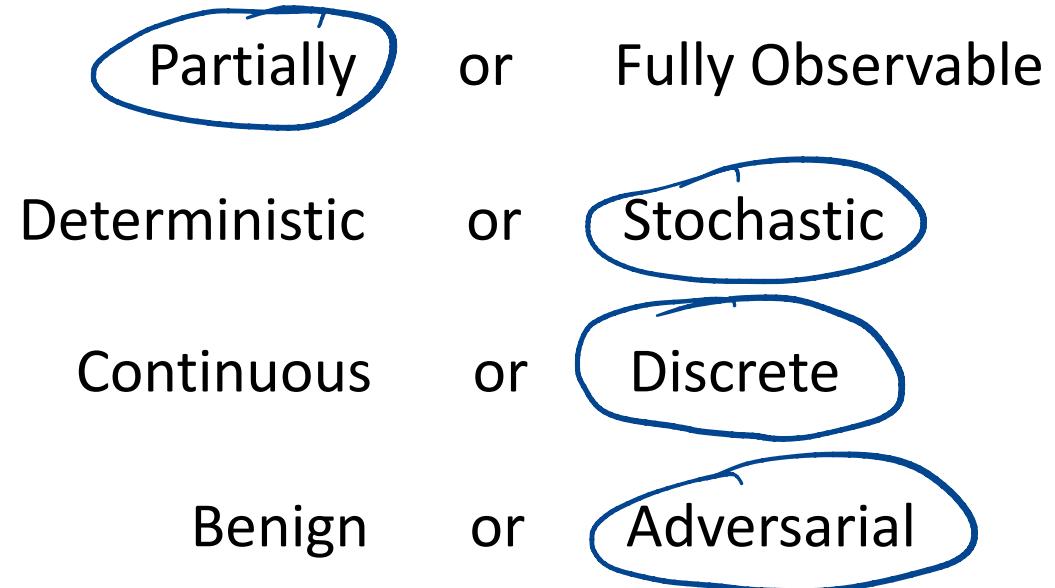
Deterministic or Stochastic

Continuous or Discrete

Benign or Adversarial

Task Environments

Example: Poker



Task Environments

Example: Robotic Car

- Partially or Fully Observable
- Deterministic or Stochastic
- Continuous or Discrete
- Benign or Adversarial

a case can be made for either

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.

Agent Types

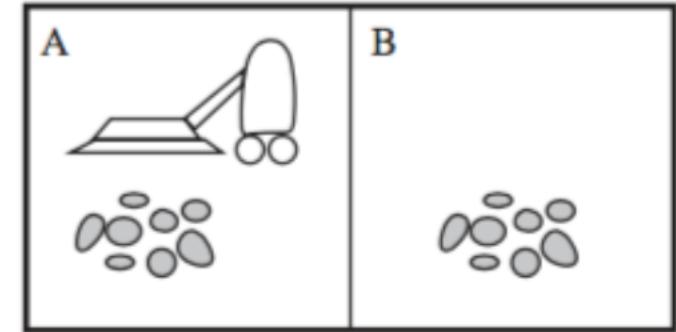
- 1) Simple reflex agents
- 2) Model-based reflex agents
- 3) Goal-based agents
- 4) Utility-based agents



❖ Boundaries between these groups are not cut and dried; an agent can be multiple types.

Simple Reflex Agent

```
In [ ]: def reflexVacuum(location, status):
    if status=='dirty':
        # clean it
    elif location==1:
        # move right
    elif location==2:
        # move left
```



- Select an action based on the present.
- Do not consider future consequences.
- Encode preferences as a function of the percepts and action.

Model-based Reflex Agent

```
In [ ]: def updateState(percept, status_old):
    # update agent's idea of what the world looks like
    return status

def ruleMatch(status, rules):
    # match which action (reflex) should follow the given state
    # according to the rules
    return action

def reflexVacuum(location, percept, status_old):
    status = updateState(percept, status_old)
    action = ruleMatch(status, rules)
```

- Select an action based on the past and present.
- Still not thinking about the future (still a reflex agent).
- Maintain an internal state to keep track of what the world looks like.

Goal-based Agent

- Select an action based on the past, present, and future.
- Towards a pre-specified goal.
- Decisions based on hypothesized consequences of proposed actions.
 - Requires a model
- Search (first half of the semester) and planning (not explicitly covered)

Utility-based Agent

- Select an action based on the past, present, and future.
 - As opposed to goals, utility quantifies the benefit of various actions.

performance measure

- different world states yield different utilities

- 1) electricity output
- 2) reliability ↗
- 3) public good ↗

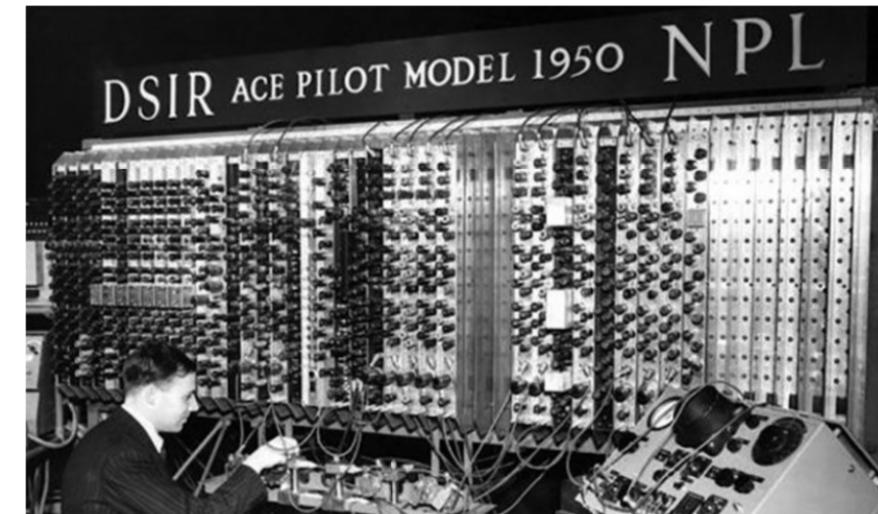


Learning Agent

Turing (1950)

- too hard to program an intelligent machine by hand
- instead, build a learning machine and then teach it

Separate percepts into a component of reward
(or penalty)



Example: Spam filters

Bayesian Reasoning

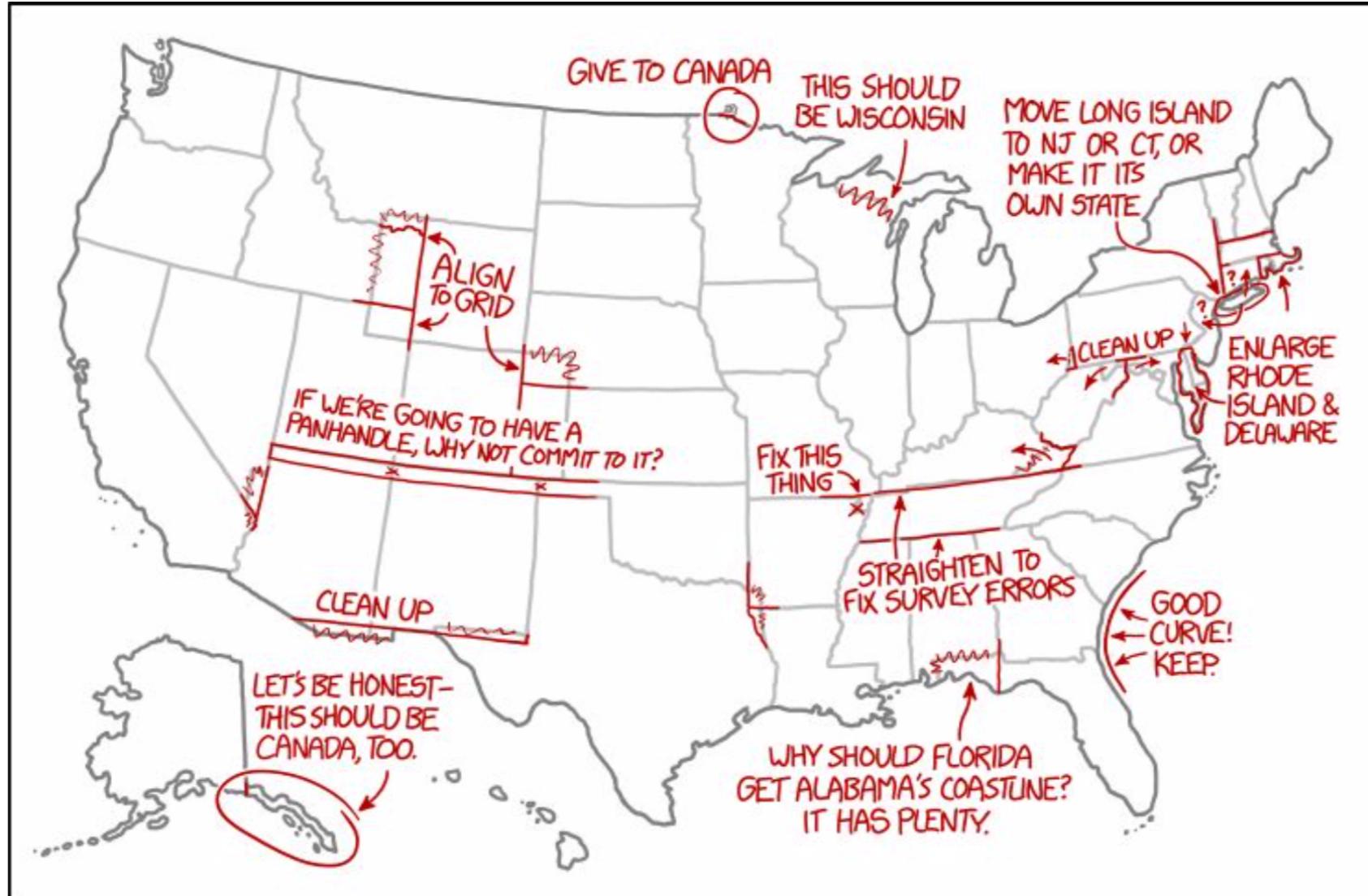
Learning Agent

Example: Spam Filters and Bayes' Theorem:

$$P(\text{spam} | w) = \frac{p(w | \text{spam})}{p(w)} \cdot p(\text{spam})$$

probability
Bayes Nets

Next Time – Agents & States



IT WAS SCARY WHEN THE GRAPHIC DESIGNERS SEIZED CONTROL OF THE COUNTRY, BUT IT TURNED OUT THEY JUST WANTED TO FIX SOME THINGS ABOUT THE STATE BORDERS THAT HAD ALWAYS BOthered THEM.

Extra Practice

Task Environments

Environment characteristic	Solitaire	Backgammon	Bidding on eBay	Taxi
Fully/partially observable?				- - -
Deterministic/stochastic?		-		
Episodic/sequential task?				
Static / dynamic environment?				
Discrete / continuous?				
Single/multi-agent?				

Task Environments

Environment characteristic	Solitaire	Backgammon	Bidding on eBay	Taxi
Fully/partially observable?	partially	fully	partially	partially
Deterministic/stochastic?	deterministic	stochastic	(semi-) deterministic	stochastic
Episodic/sequential task?	sequential	sequential	sequential	sequential
Static / dynamic environment?	static	static	semi-static/dynamic	dynamic
Discrete / continuous?	discrete	discrete	discrete	continuous
Single/multi-agent?	single	multiple	multiple	multiple