CS3243: Introduction to Artificial Intelligence

Semester 1, 2019/2020

Teaching Staff

- Lecturer: Akshay Narayan

  Email: anarayan@comp.nus.edu.sg

  Website: http://www.comp.nus.edu.sg/~anarayan

  Office: COM2-02-03

  Consultation hours: By appointment

Teaching Staff

- TAs:
  - Wesley Joon-Wie Tann (dcsawjt@nus.edu.sg)
  - Devmanyu Hazarika (hazarika@comp.nus.edu.sg)
  - Xing Jifang (e0383347@u.nus.edu)
  - Lee Yiyuan (leeyiyuan@u.nus.edu)
  - Wang Yaofeng (yaofeng@u.nus.edu)
  - Jiang Yue (jiangyue12392@gmail.com)
  - Pakorn Ueareeworakul (e0196816@u.nus.edu)
  - Rong Hua Lui (ronghualui@u.nus.edu)

Teaching Resources: LumiNUS

- Lectures, Tutorials, Supplementary Materials, Homework
- Discussion forum
  - Any questions related to the course should be raised on this forum
  - Emails to me will be considered public unless otherwise specified
- Announcements
- Homework submissions
- Webcasts

A 'Tasting Menu' of AI

Foundational concepts of AI

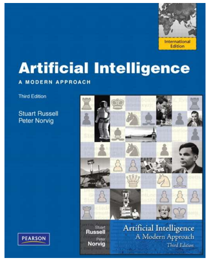
- Search
- Game playing
- Logic
- Uncertainty
- Probabilistic reasoning

Beyond CS3243

Machine Learning	Search & Planning	Logic	... And more!
CS3244	CS4246	CS4248	CS4261
CS5242	CS5446	CS6207	CS6216
CS5339	CS5338, TBA	CS4244	
CS5340			
CS5344			

Readings

- Textbook:
  - Russell and Norvig (2010). Artificial Intelligence: A Modern Approach (3rd Edition ⟵ Important!)
  - Online Resources, Code, and ERRATA: http://aima.cs.berkeley.edu/
  - We will not cover entire book! But it makes for an interesting read…



Syllabus

- Introduction and Agents (chapters 1, 2)
- Search (chapters 3, 4, 5, 6)
- Logic (chapters 7, 8, 9)
- Uncertainty (chapters 13, 14)

Assessment Overview

What	When	Grade Percentage
Midterm Exam (**during lecture, NO make-up**)	**30 September 2019** Venue: **UTown auditorium 2**	20%
Final Exam	**26 November 2019** (morning)	40%
3 Assignments	As announced!	30%
Participation (lecture + tutorial)	-	10%

Collaboration in Solving Tutorial Questions

- Collaboration is acceptable and encouraged
- You must always write the name of your collaborator on your solution sheet. Each of you must submit the write up individually.
- You will never be marked down for working with others, **unless you fail to mention it!**
- You can work in pairs.

Assignments

- You can work in pairs or groups of three; lone wolfs are discouraged!
- All of you should be from the same tutorial group
- Write the names and matric numbers of your team members in the submission
- Only one submission from the team is necessary

On Collaboration

- You are free to meet with fellow student(s) and discuss assignments.
- Writing on a board or shared piece of paper is acceptable during the meeting; however, you **may not take any written (electronic or otherwise) record away from the meeting**.
- Do not solve assignment immediately after discussion; wait a while, **ensure you can reconstruct solution by yourself**!

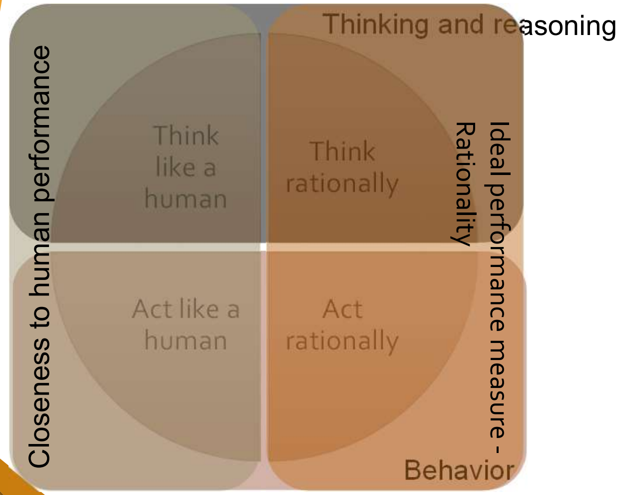
Let's begin!

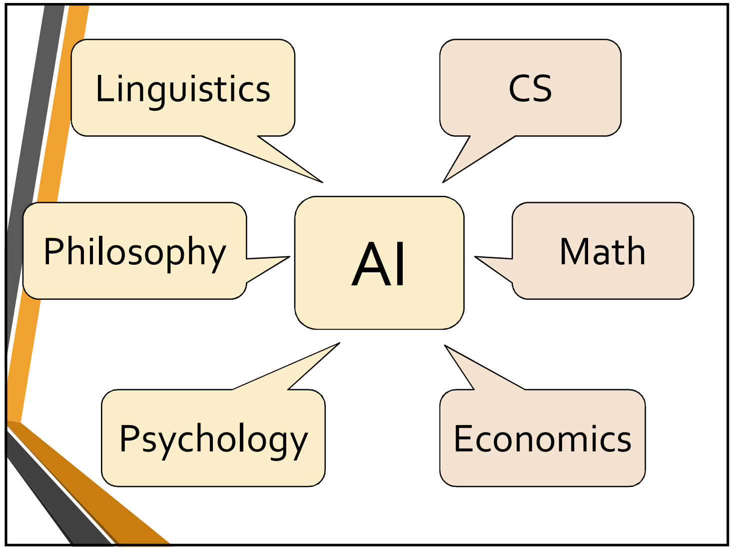
What is AI?

Making Computers (at least as) Good as Humans in Human Tasks

Oxford: The theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages.

- … but not necessarily in the same way a human would!
  - Birds and planes fly very differently
  - AI solves problems very differently.
  - I think!





Philosophy
- Ethics
- Logic
- Learning
- Rationality
- Theory of the Mind

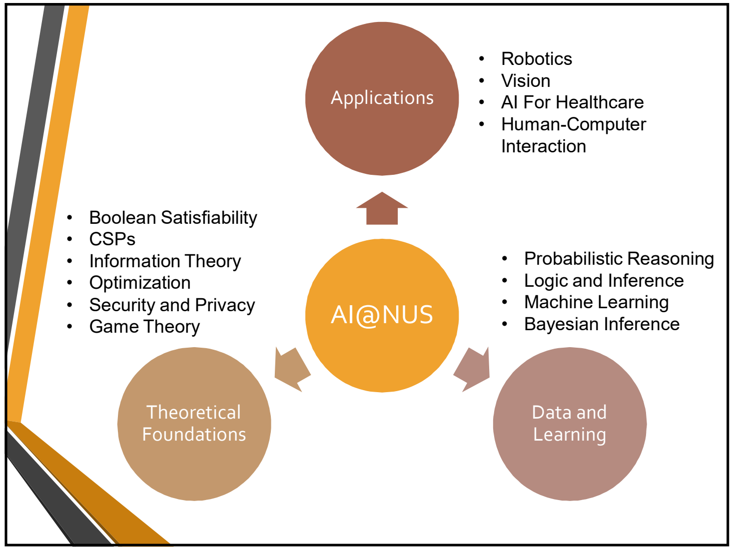
Computer Science
- Theory of Computing
- Hardware
- Control Theory
- Dynamic Systems

Mathematics
- Formal representation
- Probability
- Statistics

Economics
- Utility theory
- Decision theory
- Game theory
- Fair Division

Psychology
- Perception and motor control
- Experiments

Linguistics
- Knowledge representation
- grammar



AI is Getting Better at Gameplay

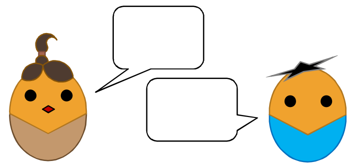
Year	Game	Program	Developer	Techniques
1994	Checkers	Chinook	U. Alberta	Rule Based + search
1997	Chess	Deep Blue	IBM	Search + randomization
2008	Limit Texas Hold'em	Polaris (Cepheus 2015)	U. Alberta	Agent based modeling, game theory
2011	Jeopardy	Watson	IBM	NLP, Information retrieval, data analytics
2015	No Limit Texas Hold'em	Claudico (later Libratus)	Carnegie Mellon Univ.	Game Theory, Reinforcement Learning
2016	Atari Games	DeepMind	Google	Deep Learning
2016	Go	AlphaGo	Google	Deep Learning, search

When Can a Machine Truly Think?

- Turing (1950). Computing Machinery and Intelligence:
  "Can machines think?" → "Can machines behave intelligently?"
- Operational test for intelligent behavior: The Imitation Game

Is the Turing Test a Good Idea?

- What are the advantages of the Turing test?
- What are the disadvantages?



Winograd Schema

The city councilmen refused the demonstrators a permit because they [feared/advocated] violence.

Who [feared/advocated] violence?

Answers: The city councilmen/the demonstrators.

Winograd Schema

- A sentence with an ambiguous pronoun (he/she/it/they/their …).
- Need to decide what the pronoun refers to (offered two options).
- This depends on **context**.
- Context can be easily flipped with a **single word**.

The firemen arrived [after/before] the police because they were coming from so far away.

Who came from far away?

Answers: The firemen/the police.

Winograd Schema Challenge

- You are given m Winograd schema, with the context word chosen uniformly at random.
- Design an AI that can correctly resolve a significant number of them.
- What is a trivial lower bound on the number of schema one can solve?

Best performance achieved: 58%

A Single Test for Intelligence?

- Difficult to resolve
- Tests tend to be
  - over-specified
  - very subjective
- Result will be debatable

Acting Rationally: Rational Agent

- Rational behavior: doing the “right thing”
- What is the “right thing” to do? Expected to achieve best outcome
  - Best for whom?
  - What are we optimizing?
  - What information is available?
  - Unintended effects

- Break through wall to get a cup of coffee
- Prescribe high doses of opiates to depressed patient
- Kill the human who tries to deactivate robot

Rational Agents

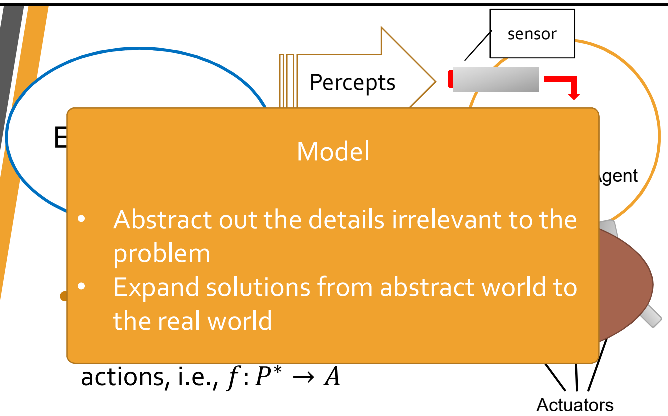
- An agent is an entity that perceives and acts
- **This course:** designing rational agents
- An agent is a function from **percept histories** to **actions**, i.e., $f: P^* \rightarrow A$
- We seek the best-performing agent for a certain task; must consider computation limits!

design best program given resources

Intelligent Agents

Agents

- Anything that can be viewed as perceiving its environment through sensors; acting upon that environment through actuators
- **Human agent:** eyes, ears, skin etc. are sensors; hands, legs, mouth, and other body parts are actuators
- **Robotic agent:** cameras and laser range finders for sensors; various motors for actuators



Model

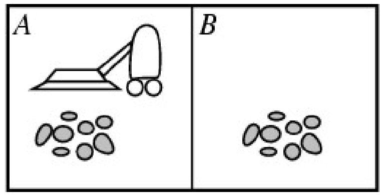
- Abstract out the details irrelevant to the problem
- Expand solutions from abstract world to the real world

actions, i.e., $f: P^* \to A$

- The agent program runs on the physical architecture to perform f

agent = architecture + program

Vacuum-Cleaner World



- Percepts: location and status, e.g., $[A, \text{Dirty}]$
- Actions: Left, Right, Suck, NoOp

Vacuum-Cleaner Agent Function

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

Rational Agents

- An agent should strive to "do the right thing", based on what it can perceive and the actions it can perform. The right action: maximize agent success.
- Performance measure: objective criterion for measuring success of an agent's behavior
- Vacuum-cleaner agent:
  - amount of dirt cleaned
  - time taken
  - electricity consumed
  - noise generated

Rational Agents

- Rational Agent:
  - For each possible percept sequence, 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.

Rational Agents

- Rationality ≠ omniscience (all-knowing with infinite knowledge)
- Agents can perform actions that help them gather useful information (exploration)
- An agent is autonomous if its behavior is determined by its own experience (with ability to learn and adapt)

Specifying Task Environment: PEAS

- PEAS: Performance measure, Environment, Actuators, Sensors
- Must first specify the setting for intelligent agent design
- Consider, e.g., the task of designing an automated taxi driver:
  - Performance measure
  - Environment
  - Actuators
  - Sensors

Specifying Task Environment: PEAS
Automated Taxi

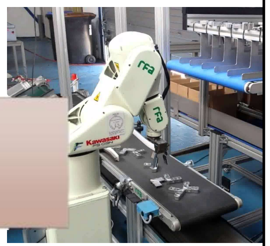
Performance Measure
- Safe
- Fast
- Legal
- Comfort
- Revenue

Environment
- Roads
- Other traffic
- Pedestrians
- Customers

Actuators
- Steering wheel
- Accelerator
- Brake
- Signal
- Horn

Sensors
- Camera
- Sonar
- Speedometer
- GPS
- Engine sensors

Specifying Task Environment: PEAS
Part Picking Robot



Performance Measure
- % parts in correct bins

Environment
- Conveyor belt
- parts
- bins

Actuators
- Jointed arm
- hand

Sensors
- Camera
- joint angle sensors

Specifying Task Environment: PEAS
Medical Diagnosis System

Performance measure
- Healthy patient
- cost
- lawsuits

Environment
- Patient
- hospital
- staff

Actuators
- Screen display (questions, tests, diagnoses, treatments, referrals)

Sensors
- Keyboard
- Medical Readings
- Medical History

Specifying Task Environment: PEAS
Interactive English Tutor

Performance measure
- Student's score on test

Environment
- Set of students
- Testing agency
- Chat platform

Actuators
- Screen display (exercises, suggestions, corrections)

Sensors
- Keyboard entry

Properties of Task Environments

Fully observable (vs. partially observable):
- Sensors provide access to the complete state of the environment at each point in time.

Deterministic (vs. stochastic)
- The next state of the environment is completely determined by the current state and the action executed by the agent.

Episodic (vs. sequential)
- The choice of **current** action does not depend on actions in past episodes.

Properties of Task Environments

Static (vs. dynamic)
- The environment is unchanged while an agent is deliberating.

Discrete (vs. continuous)
- A finite number of distinct states, percepts, and actions.

Single agent (vs. multi-agent)
- An agent operating by itself in an environment.

Properties of Task Environments

Task Environment	Crossword puzzle	Part-picking robot	Taxi driving
Fully observable	Yes	No	No
Deterministic	Yes	No	No
Episodic	No	Yes	No
Static	Yes	No	No
Discrete	Yes	No	No
Single agent	Yes	Yes	No

Properties of task environment largely determine agent design. World is partially observable, stochastic, sequential, dynamic, continuous, multi-agent.

Agent Functions and Programs

- An agent is completely specified by the agent function mapping percept sequences to actions
- One agent function (or a small equivalence class) is rational
- Aim: Find a way to implement the rational agent function concisely

Agent Functions and Programs

Solution: write all possible percepts and optimal actions in a table.

All done!

Table-Lookup Agent

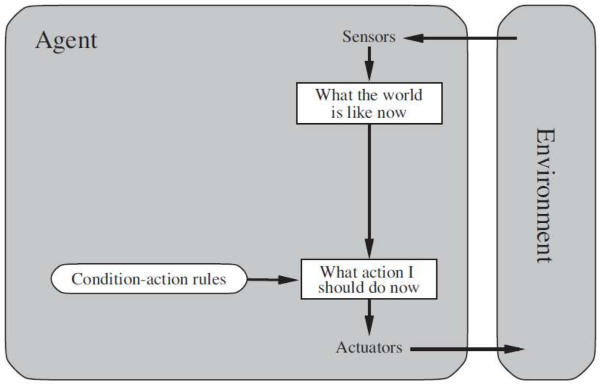
```
function TABLE-DRIVEN-AGENT(percept) returns action
   static: percepts, a sequence, initially empty
           table, a table of actions, indexed by percept sequences, fully specified

   append percept to the end of percepts
   action ← LOOKUP(percepts, table)
   return action
```

- Drawbacks:
  - Huge table to store
  - Take a long time to build the table
  - No autonomy: impossible to learn all correct table entries from experience
  - No guidance on filling in the correct table entries

Agent Types

- Four basic types in order of increasing generality:
  - Simple reflex agent
  - Model-based reflex agent
  - Goal-based agent
  - Utility-based agent

Reflex Agent

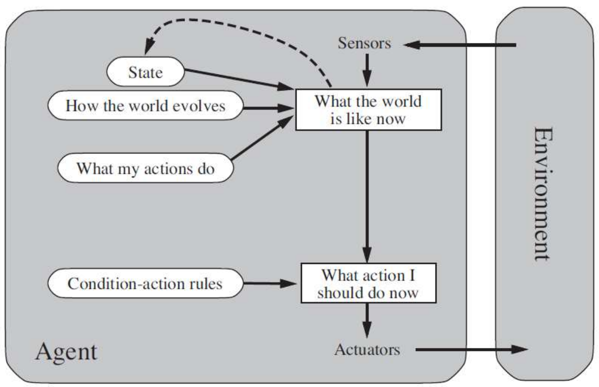


Reflex Agent

- Passive: only acts when observes a percept
- Updates *state* based on *percept* only.
- Easy to implement

Motion sensor triggered.
state ← *WELCOME MODE*

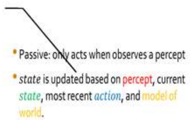
Model Based Reflex Agent



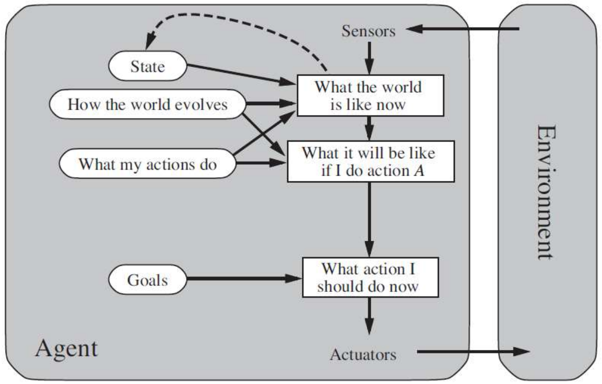
Model Based Reflex Agent

- Passive: only acts when observes a percept
- *state* is updated based on percept, current *state*, most recent *action*, and model of world.

Motion sensor triggered. I am on ACTIVE MODE. I AUTHORIZED ACCESS. ⇒ HUMAN IN HOUSE
state ← *WELCOME MODE*



Goal Based Agent



Goal Based Agent

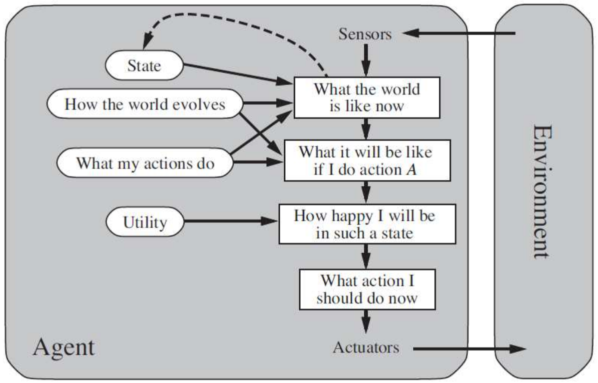
- Has **goals**, acts to achieve them (not passive).
- *state* is updated based on percept, current *state*, most recent *action*, and model of world.

Goal: make human happy (and buy products from my vendor)

Motion sensor triggered. I am on ACTIVE MODE. I AUTHORIZED ACCESS. ⇒ HUMAN IN HOUSE

$state \leftarrow WELCOME\ MODE(1)$

Utility Based Agent



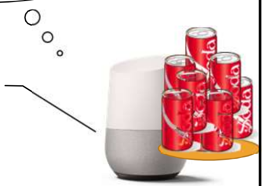
Utility Based Agent

- Has **utility function**, acts to maximize it.
- *state* is updated based on percept, current *state*, most recent *action*, and model of world.

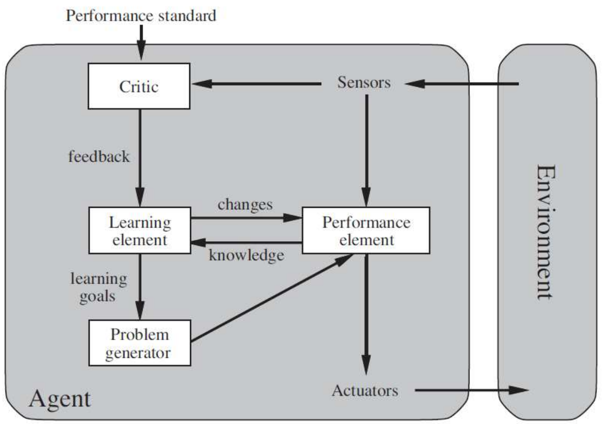
$\max \alpha \times happiness + \beta \times purchases$

Motion sensor triggered. I am on ACTIVE MODE. I AUTHORIZED ACCESS. ⇒ HUMAN IN HOUSE

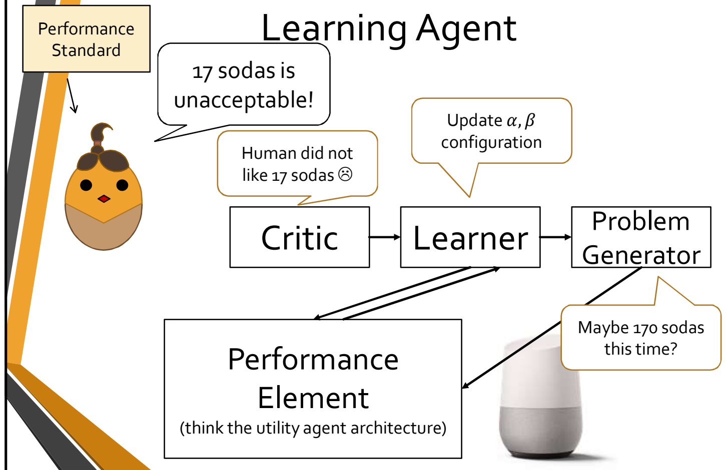
$state \leftarrow WELCOME\ MODE(17)$



Learning Agent



Learning Agent



Exploitation vs. Exploration

- An agent operating in the real world must often choose between:
  - maximizing its expected utility according to its current knowledge about the world; and
  - trying to learn more about the world since this may improve its future gains.

Exploitation vs. ***Exploration***