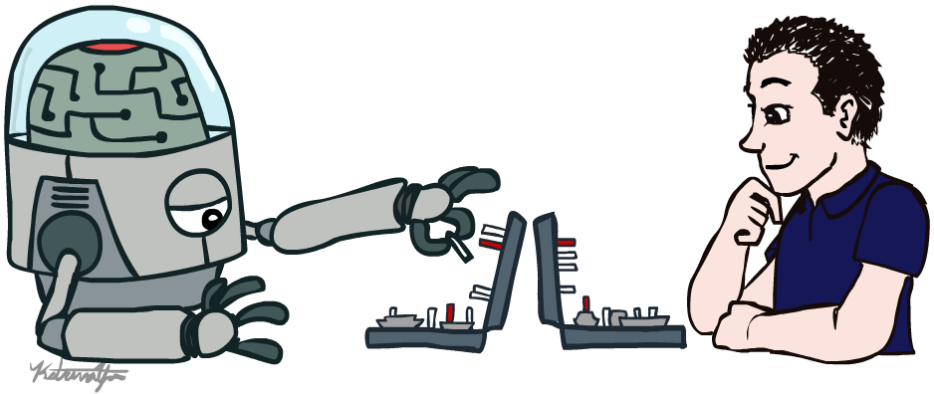
CSE 473: Introduction to Artificial Intelligence

Introduction



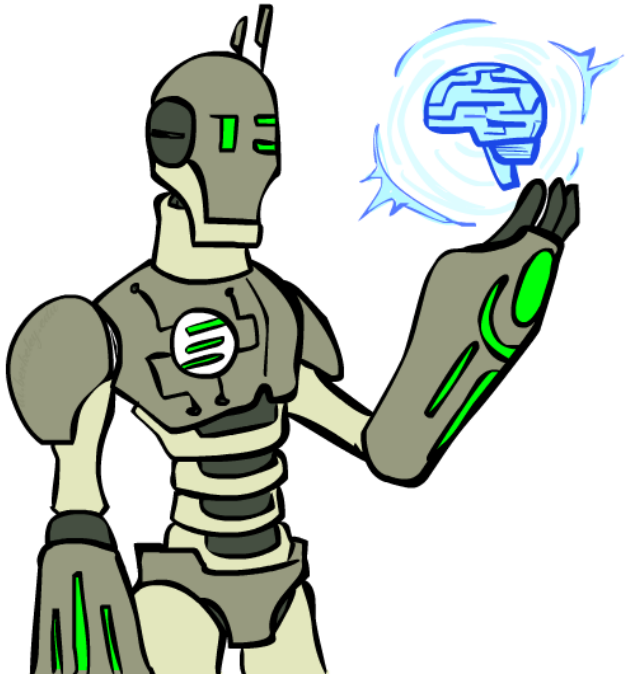
Luke Zettlemoyer

University of Washington

[These slides were adapted from Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All materials at http://ai.berkeley.edu.]

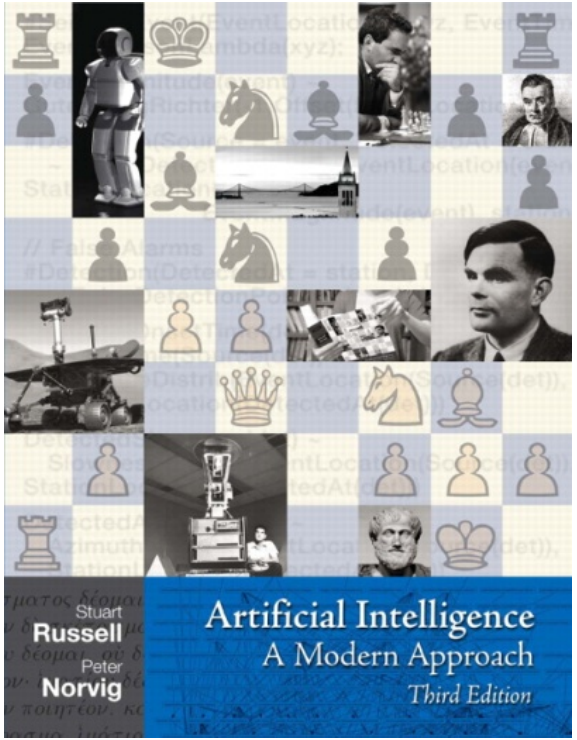
Today

- Course Overview
- What is artificial intelligence?
- What can AI do?
- What is this course?



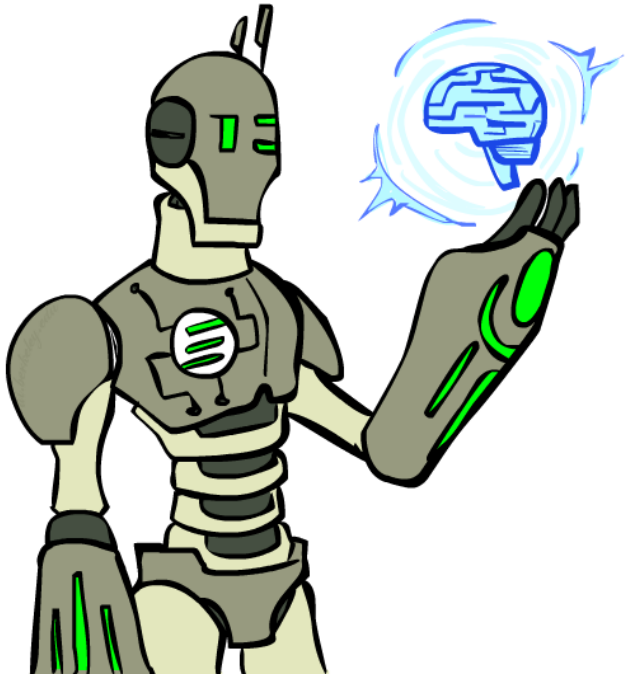
Textbook

- Not required, but for students who want to read more we recommend
  - Russell & Norvig, AI: A Modern Approach, 3rd Ed.
  - Warning: Not a course textbook, so our presentation does not necessarily follow the presentation in the book.

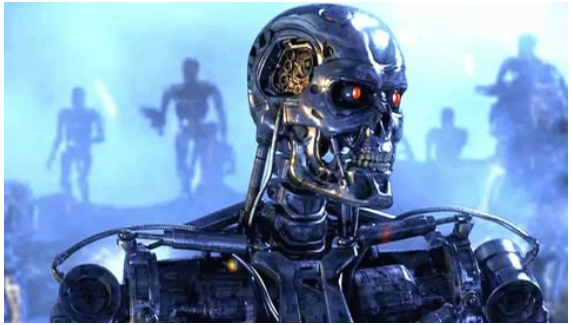


Today

- What is artificial intelligence?
- What can AI do?
- What is this course?



Sci-Fi AI?



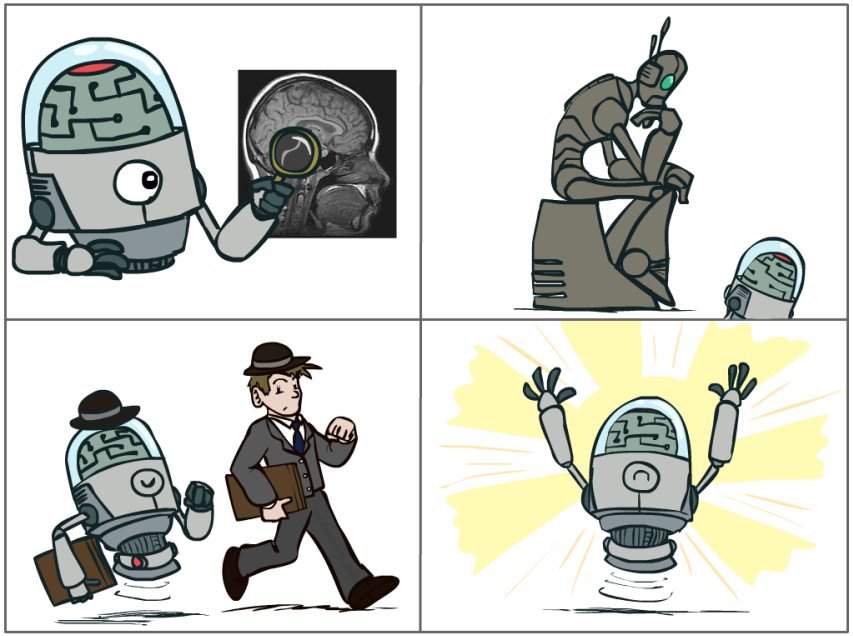
What is AI?

The science of making machines that:

What is AI?

The science of making machines that:

Think like people



Think rationally

Act like people

Act rationally

Rational Decisions

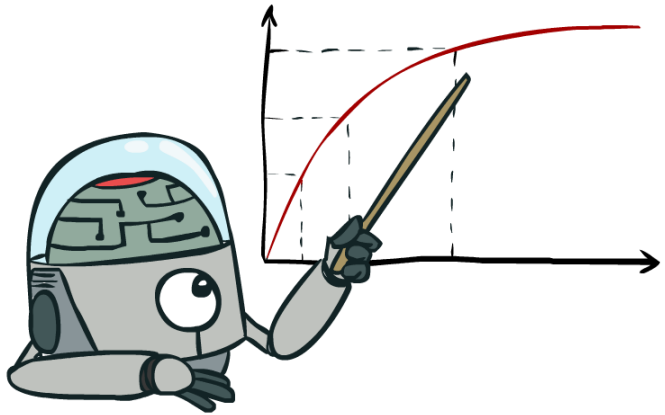
We'll use the term **rational** in a very specific, technical way:

- Rational: maximally achieving pre-defined goals
- Rationality only concerns what decisions are made (not the thought process behind them)
- Goals are expressed in terms of the **utility** of outcomes
- Being rational means **maximizing your expected utility**

A better title for this course would be:

Computational Rationality

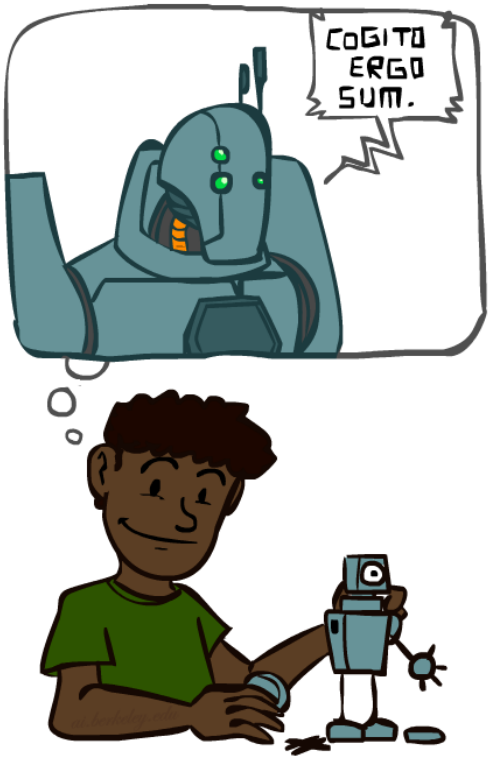
Maximize Your Expected Utility



What About the Brain?

- Brains (human minds) are very good at making rational decisions, but not perfect
- Brains aren't as modular as software, so hard to reverse engineer!
- "Brains are to intelligence as wings are to flight"
- Lessons learned from the brain: memory and simulation are key to decision making

A (Short) History of AI



Demo: HISTORY – MT1950.wmv

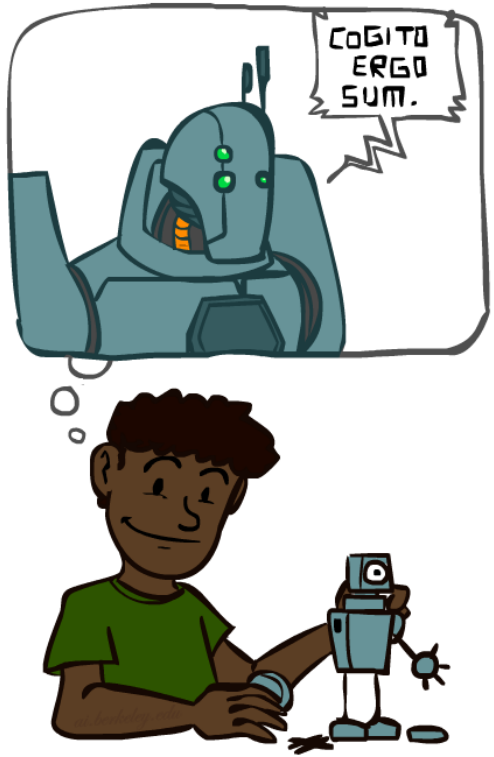
A Historic Idea….

A (Short) History of AI

- 1940-1950: Early days
  - 1943: McCulloch & Pitts: Boolean circuit model of brain
  - 1950: Turing's “Computing Machinery and Intelligence”

I propose to consider the question, "Can machines think?" This should begin with definitions of the meaning of the terms "machine" and "think." The definitions might be framed...

-Alan Turing

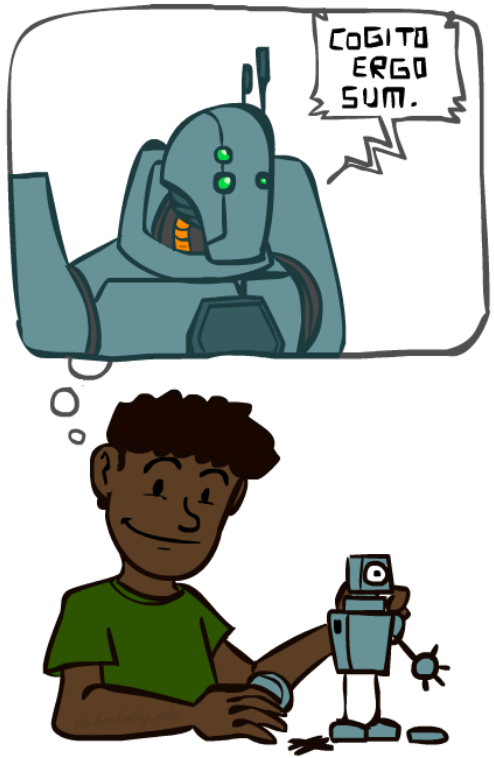


A (Short) History of AI

- 1940-1950: Early days
  - 1943: McCulloch & Pitts: Boolean circuit model of brain
  - 1950: Turing's "Computing Machinery and Intelligence"
- 1950—70: Excitement: Look, Ma, no hands!
  - 1950s: Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: Dartmouth meeting: "Artificial Intelligence" adopted
  - 1965: Robinson's complete algorithm for logical reasoning

"Over Christmas, Allen Newell and I created a thinking machine."

-Herbert Simon

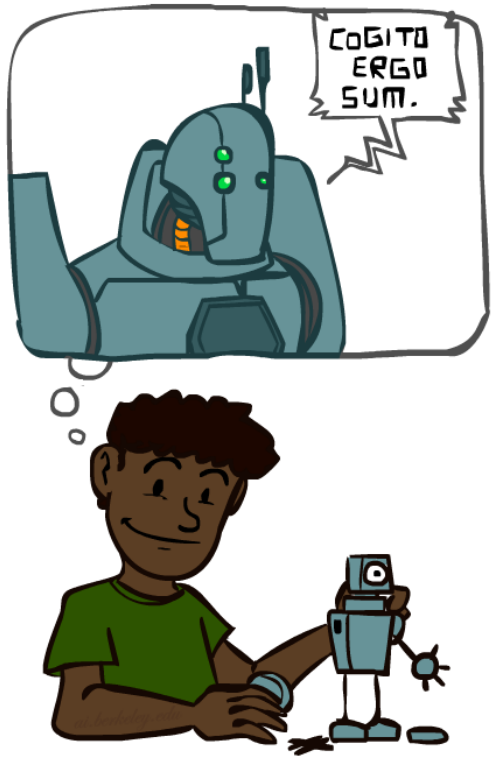


A (Short) History of AI

- 1940-1950: Early days
  - 1943: McCulloch & Pitts: Boolean circuit model of brain
  - 1950: Turing's "Computing Machinery and Intelligence"
- 1950—70: Excitement: Look, Ma, no hands!
  - 1950s: Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: Dartmouth meeting: "Artificial Intelligence" adopted
  - 1965: Robinson's complete algorithm for logical reasoning
- 1970—90: Knowledge-based approaches
  - 1969—79: Early development of knowledge-based systems
  - 1980—88: Expert systems industry booms
  - 1988—93: Expert systems industry busts: "AI Winter"

The knowledge engineer practices the art of bringing the principles and tools of AI research to bear on difficult applications problems requiring experts' knowledge for their solution.

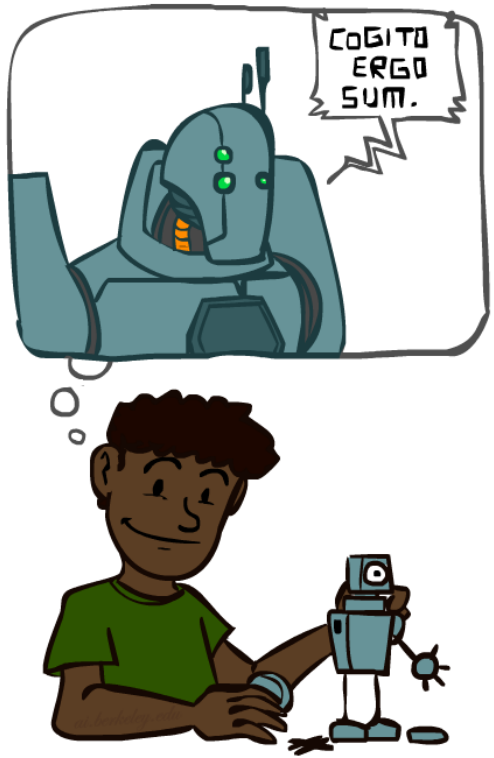
- *Edward Felgenbaum* in "The Art of Artificial Intelligence"



A (Short) History of AI

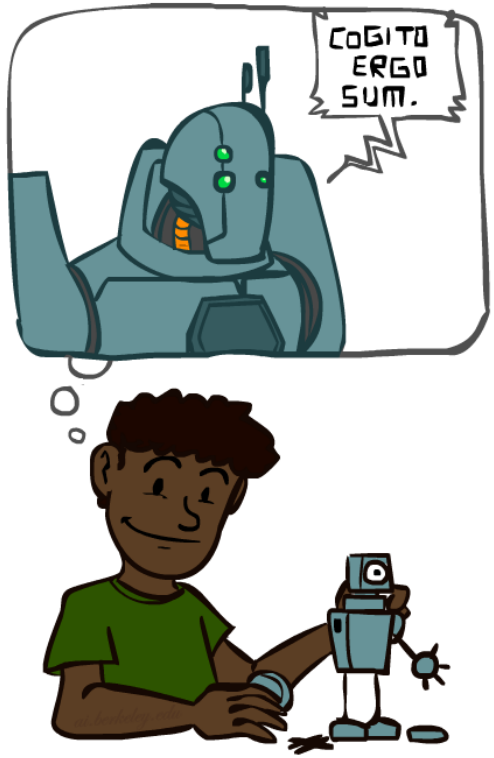
- 1940-1950: Early days
  - 1943: McCulloch & Pitts: Boolean circuit model of brain
  - 1950: Turing's "Computing Machinery and Intelligence"
- 1950—70: Excitement: Look, Ma, no hands!
  - 1950s: Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: Dartmouth meeting: "Artificial Intelligence" adopted
  - 1965: Robinson's complete algorithm for logical reasoning
- 1970—90: Knowledge-based approaches
  - 1969—79: Early development of knowledge-based systems
  - 1980—88: Expert systems industry booms
  - 1988—93: Expert systems industry busts: "AI Winter"
- 1990—: Statistical approaches
  - Resurgence of probability, focus on uncertainty
  - General increase in technical depth
  - Agents and learning systems… "AI Spring"?

Every time I fire a linguist, the performance of the speech recognizer goes up. – *Frederick Jelinek*, IBM



A (Short) History of AI

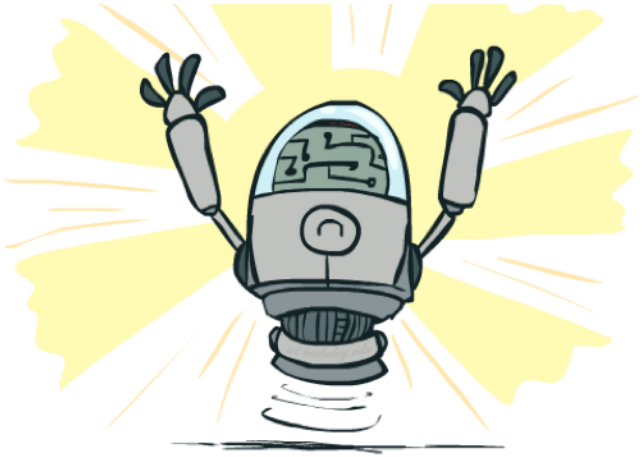
- 1940-1950: Early days
  - 1943: McCulloch & Pitts: Boolean circuit model of brain
  - 1950: Turing's "Computing Machinery and Intelligence"
- 1950—70: Excitement: Look, Ma, no hands!
  - 1950s: Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: Dartmouth meeting: "Artificial Intelligence" adopted
  - 1965: Robinson's complete algorithm for logical reasoning
- 1970—90: Knowledge-based approaches
  - 1969—79: Early development of knowledge-based systems
  - 1980—88: Expert systems industry booms
  - 1988—93: Expert systems industry busts: "AI Winter"
- 1990—: Statistical approaches
  - Resurgence of probability, focus on uncertainty
  - General increase in technical depth
  - Agents and learning systems… "AI Spring"?
- 2010—: Where are we now?



What Can AI Do?

Quiz: Which of the following can be done at present?

- ✓ Play a decent game of table tennis?
- ✓ Play a decent game of Jeopardy?
- ✓ Drive safely along a curving mountain road?
- ? Drive safely along University Avenue?
- ✓ Buy a week's worth of groceries on the web?
- ✗ Buy a week's worth of groceries at QFC?
- ? Discover and prove a new mathematical theorem?
- ✗ Converse successfully with another person for an hour?
- ? Perform a surgical operation?
- ✓ Put away the dishes and fold the laundry?
- ✓ Translate spoken Chinese into spoken English in real time?
- ✗ Write an intentionally funny story?

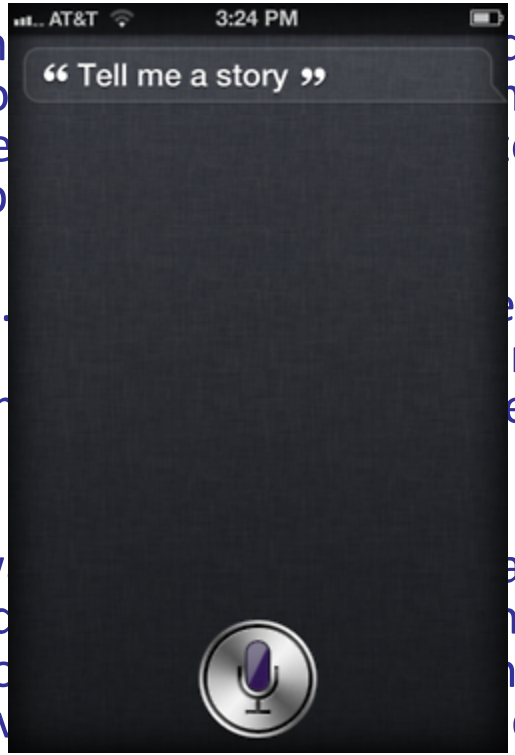


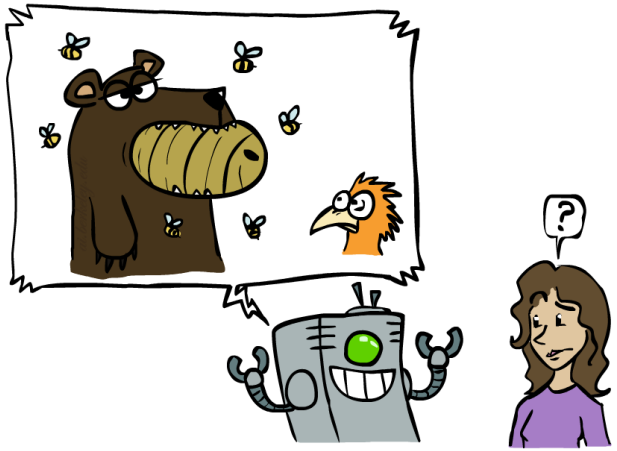
Unintentionally Funny Stories

- One day Joe Bear was hun d Irving Bird where some ho m there was a beehive in the o the oak tree. He ate the b

- Henry Squirrel was thirsty. river bank where his good ng. Henry slipped and fell in th ed. The End.

- Once upon a time there w a vain crow. One day the crow was sitting in his tree, hold his mouth. He noticed that he was holding the piece c hungry, and swallowed the cheese. The fox walked ov d.

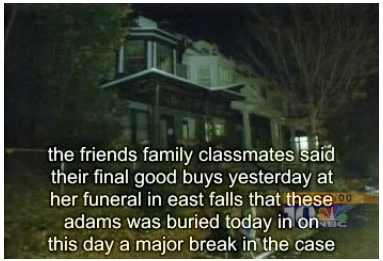




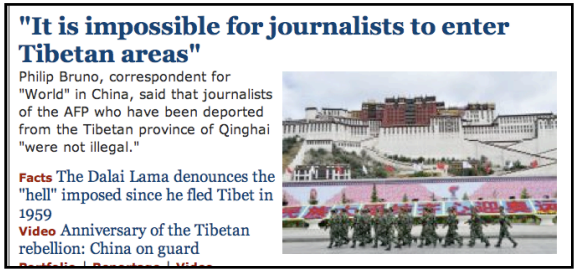
[Shank, Tale-Spin System, 1984]

Natural Language

- Speech technologies (e.g. Siri)
  - Automatic speech recognition (ASR)
  - Text-to-speech synthesis (TTS)
  - Dialog systems



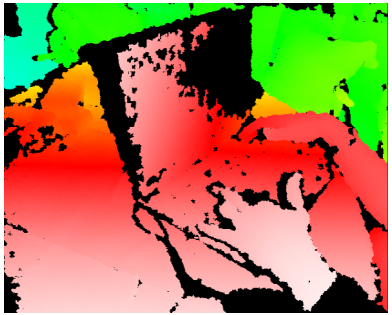
- Language processing technologies
  - Question answering
  - Machine translation



  - Web search
  - Text classification, spam filtering, etc…

Vision (Perception)

- Object and face recognition
- Scene segmentation
- Image classification

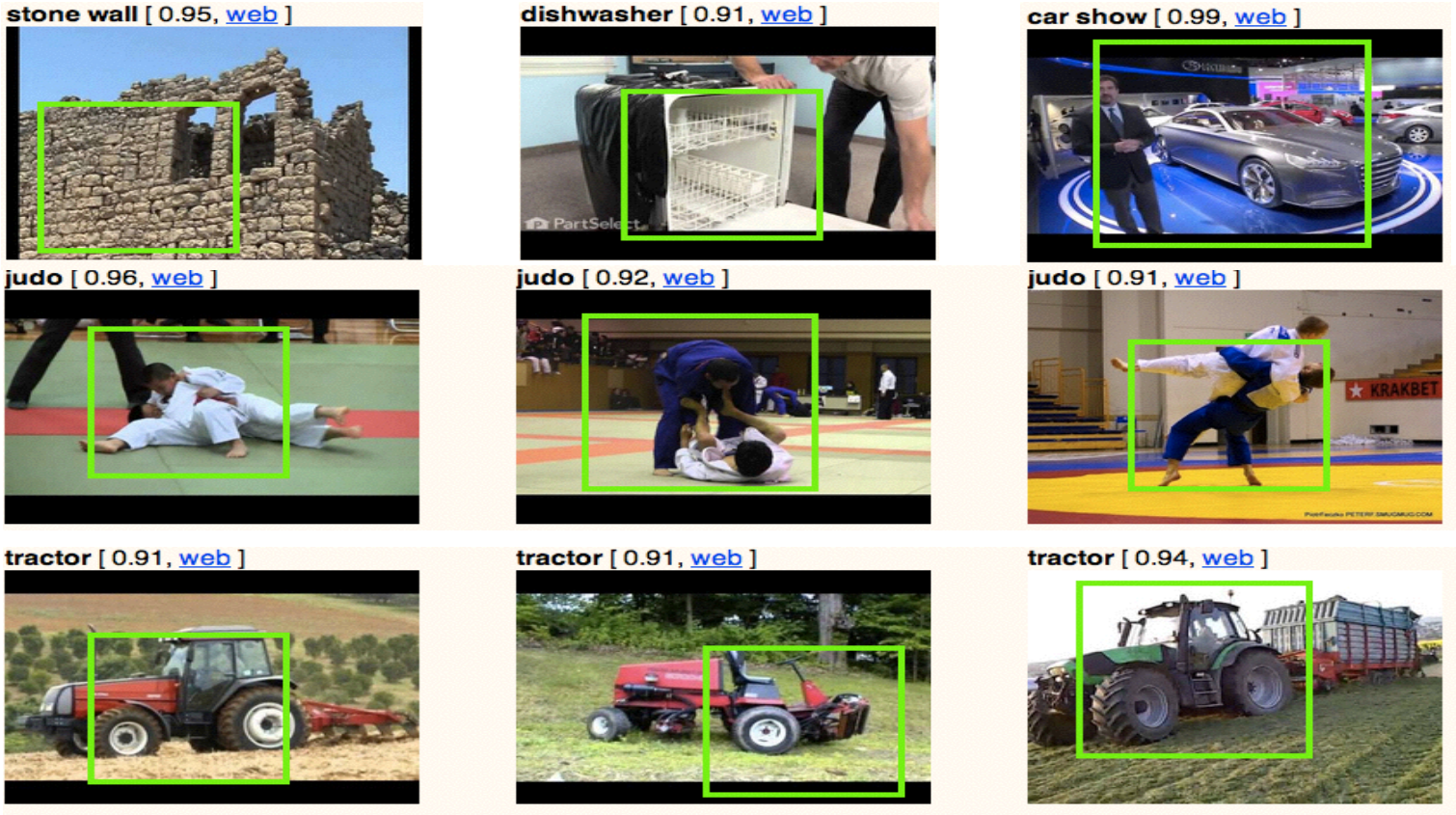


Images from Erik Sudderth (left), wikipedia (right)

Demo1: VISION – lec_1_t2_video.flv

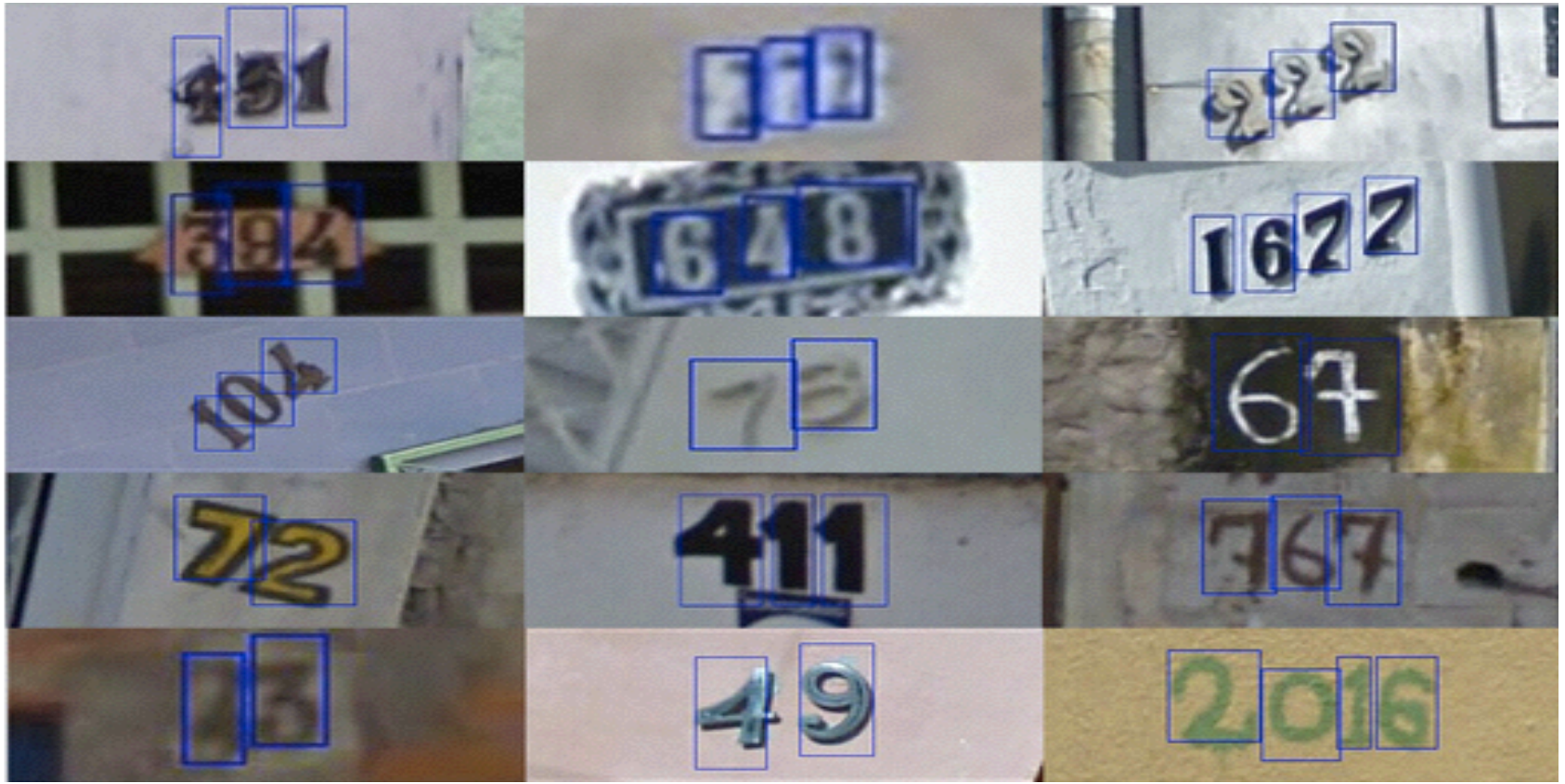
Demo2: VISION – lec_1_obj_rec_0.mpg

Object Some Recent Results



Slides from Jeff Dean at Google

Number Detection



Slides from Jeff Dean at Google

Good Generalization

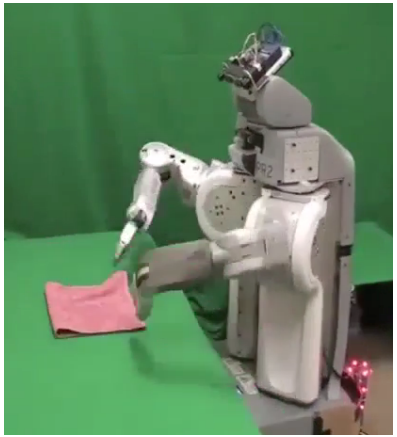
Both recognized as a
"meal"

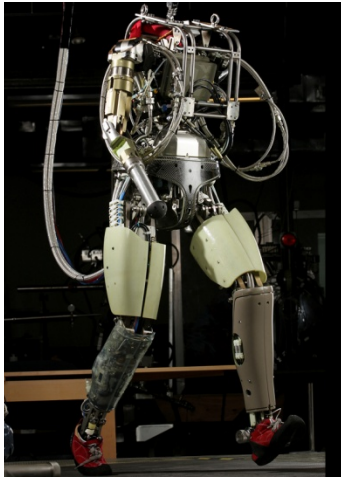
Slides from Jeff Dean at Google

Robotics

Demo 1: ROBOTICS – soccer.avi
Demo 2: ROBOTICS – soccer2.avi
Demo 3: ROBOTICS – gcar.avi
Demo 4: ROBOTICS – laundry.avi
Demo 5: ROBOTICS – petman.avi

- Robotics
  - Part mech. eng.
  - Part AI
  - Reality much harder than simulations!
- Technologies
  - Vehicles
  - Rescue
  - Soccer!
  - Lots of automation…
- In this class:
  - We ignore mechanical aspects
  - Methods for planning
  - Methods for control

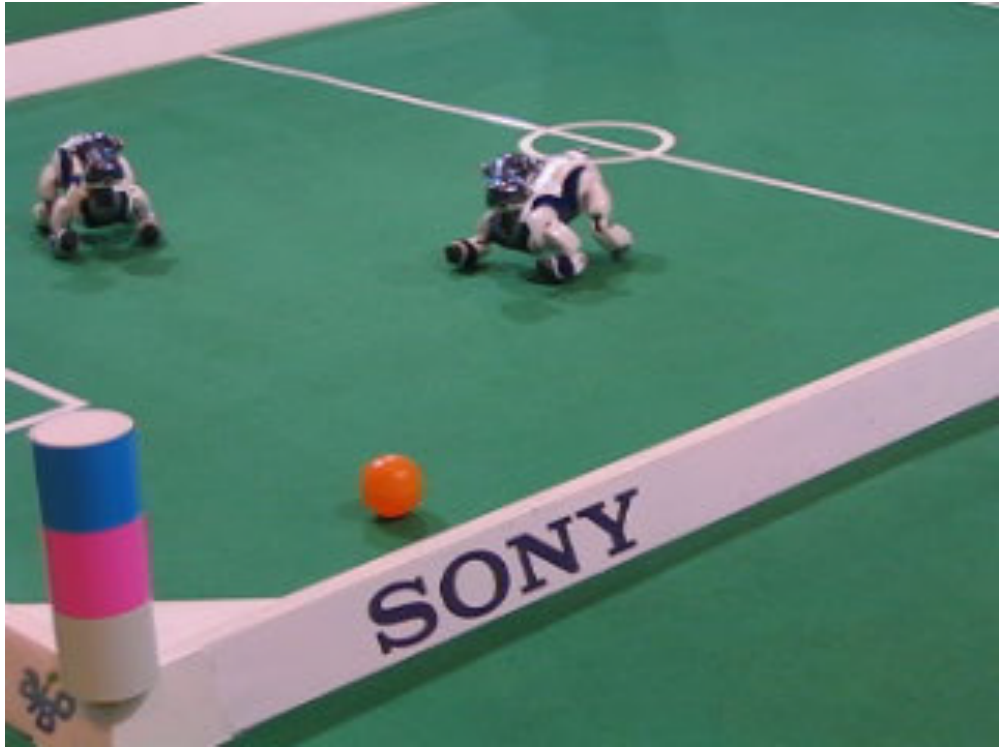




Images from UC Berkeley, Boston Dynamics, RoboCup, Google

Robot Soccer

Robot Soccer



Google Car

Logic

- Logical systems
  - Theorem provers
  - NASA fault diagnosis
  - Question answering
- Methods:
  - Deduction systems
  - Constraint satisfaction
  - Satisfiability solvers (huge advances!)

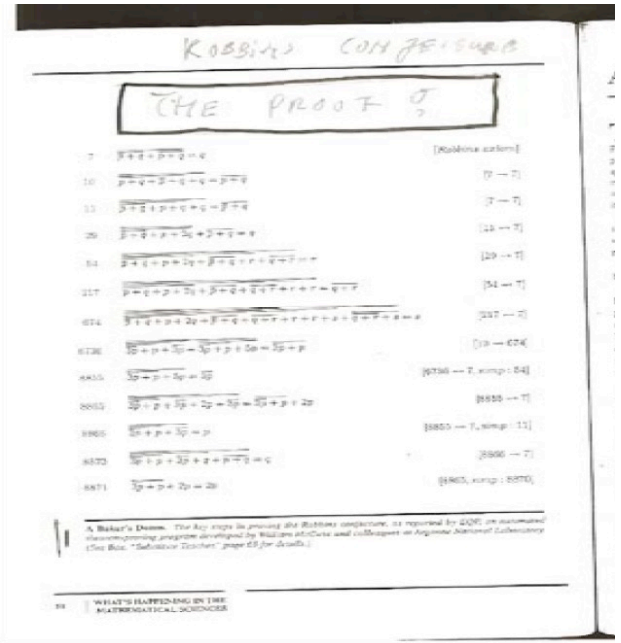
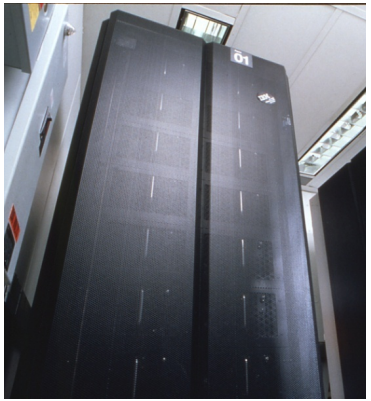


Image from Bart Selman

Game Playing

- Classic Moment: May, '97: Deep Blue vs. Kasparov
  - First match won against world champion
  - “Intelligent creative” play
  - 200 million board positions per second
  - Humans understood 99.9 of Deep Blue's moves
  - Can do about the same now with a PC cluster



- Open question:
  - How does human cognition deal with the search space explosion of chess?
  - Or: how can humans compete with computers at all??

- 1996: Kasparov Beats Deep Blue
  “I could feel --- I could smell --- a new kind of intelligence across the table.”
- 1997: Deep Blue Beats Kasparov
  “Deep Blue hasn't proven anything.”
- Huge game-playing advances recently, e.g. in Go!

Text from Bart Selman, image from IBM’s Deep Blue pages

AlphaGo versus Lee Sedol

From Wikipedia, the free encyclopedia

Coordinates: 37.5706°N 126.9754°E

AlphaGo versus Lee Sedol or **Google DeepMind Challenge Match** was a five-game Go match between South Korean professional Go player Lee Sedol and AlphaGo, a computer Go program developed by Google DeepMind, played in Seoul, South Korea between 9 and 15 March 2016. AlphaGo won all but the fourth game;[1] all games were won by resignation.[2] The match has been compared with the historic chess match between Deep Blue and Garry Kasparov in 1997.

AlphaGo versus Lee Sedol
4–1

Seoul, South Korea, 9–15 March 2016	
Game one	AlphaGo won.
Game two	AlphaGo won.
Game three	AlphaGo won.
Game four	Lee Sedol won.
Game five	AlphaGo won.

The winner of the match was slated to win $1 million. Since AlphaGo won, Google DeepMind stated that the prize will be donated to charities, including UNICEF, and Go organisations.[3] Lee received $170,000 ($150,000 for participating in all the five games, and an additional $20,000 each game won).[4]

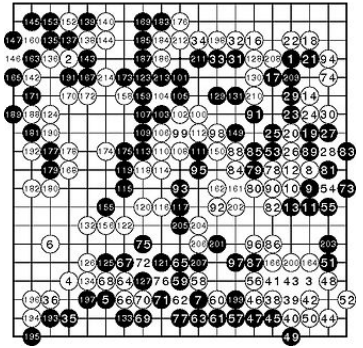
After the match, The Korea Baduk Association awarded AlphaGo the highest Go grandmaster rank – an "honorary 9 dan". It was given in recognition of AlphaGo's "sincere efforts" to master Go.[5]

Contents [hide]

1 Background

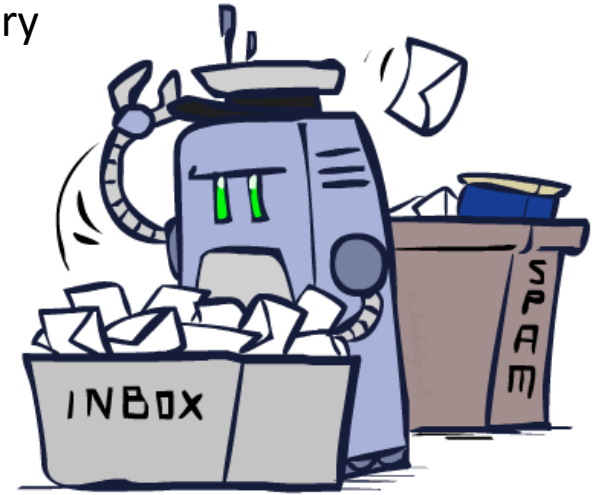
AlphaGo

"I misjudged the capabilities of AlphaGo and felt powerless.", quote after game 3



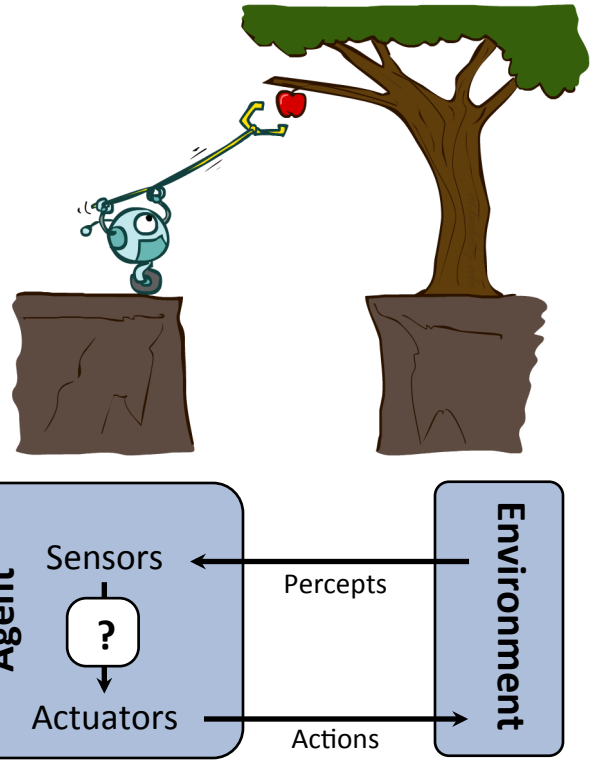
Decision Making

- Applied AI involves many kinds of automation
  - Scheduling, e.g. airline routing, military
  - Route planning, e.g. Google maps
  - Medical diagnosis
  - Web search engines
  - Spam classifiers
  - Automated help desks
  - Fraud detection
  - Product recommendations
  - … Lots more!

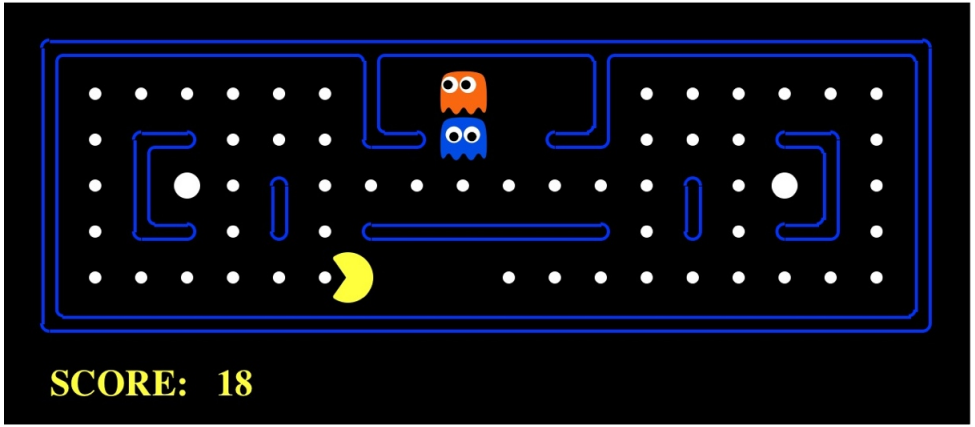


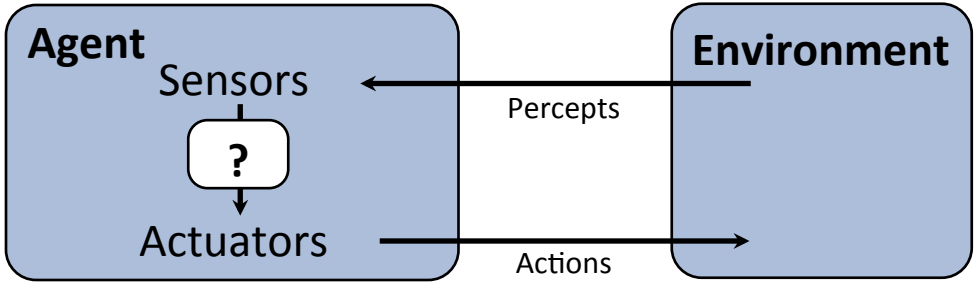
Designing Rational Agents

- An **agent** is an entity that *perceives* and *acts*.
- A **rational agent** selects actions that maximize its (expected) **utility**.
- Characteristics of the **percepts, environment,** and **action space** dictate techniques for selecting rational actions
- **This course** is about:
  - General AI techniques for a variety of problem types
  - Learning to recognize when and how a new problem can be solved with an existing technique



Pac-Man as an Agent



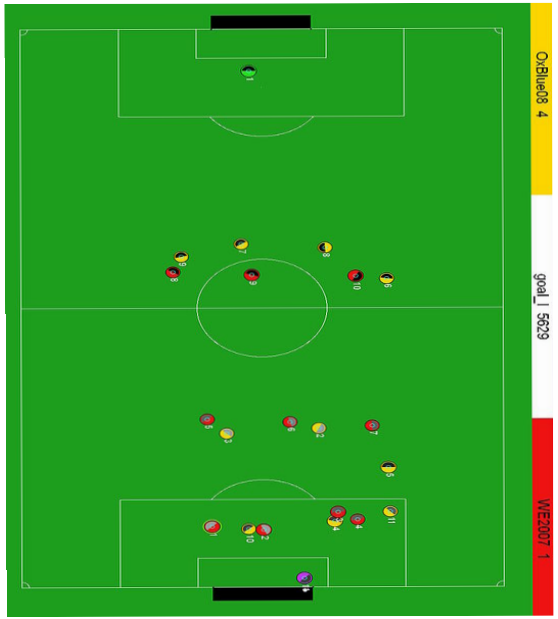


Types of Environments

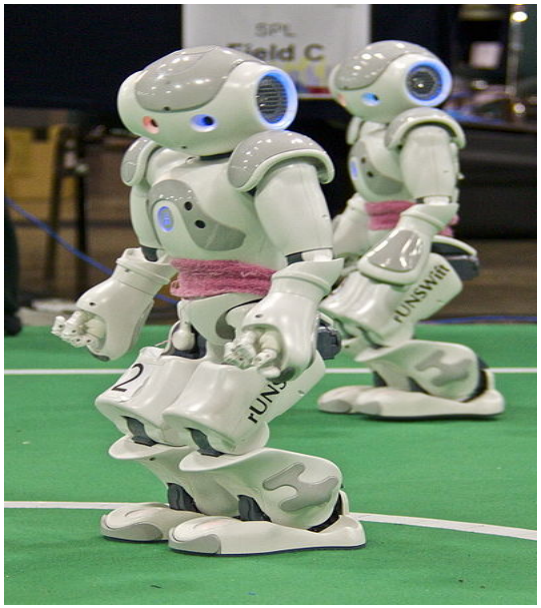
- Fully observable *vs.* partially observable
- Single agent *vs.* multiagent
- Deterministic *vs.* stochastic
- Static *vs.* sequential
- Discrete *vs.* continuous

Fully observable vs. Partially observable

Can the agent observe the complete state of the environment?



vs.



Single agent vs. Multiagent

Is the agent the only thing acting in the world?

vs.

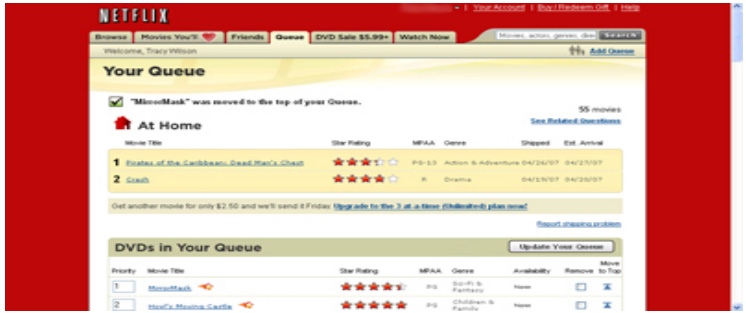
Deterministic vs. Stochastic

Is there uncertainty in how the world works?

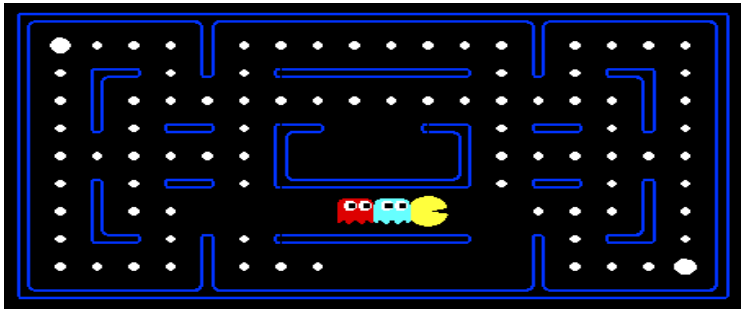
vs.

Static vs. Sequential

Does the agent take more than one action?



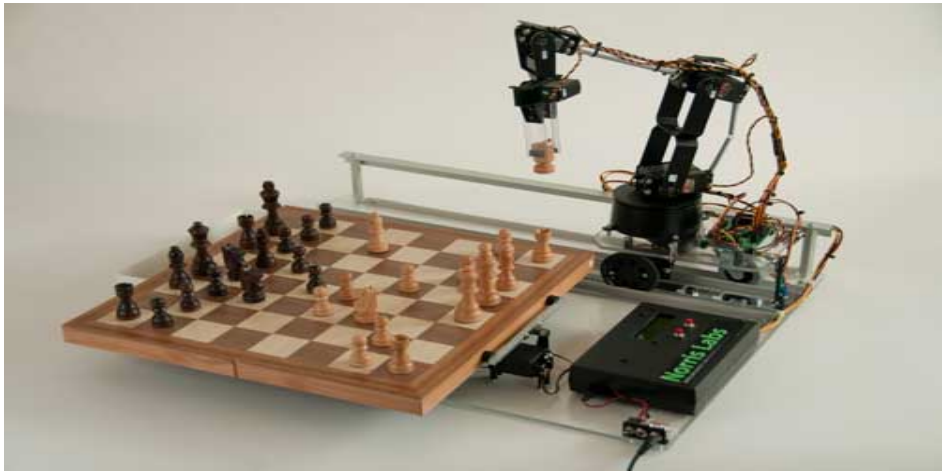
vs.



Discrete vs. Continuous

- Is there a finite (or countable) number of possible environment states?

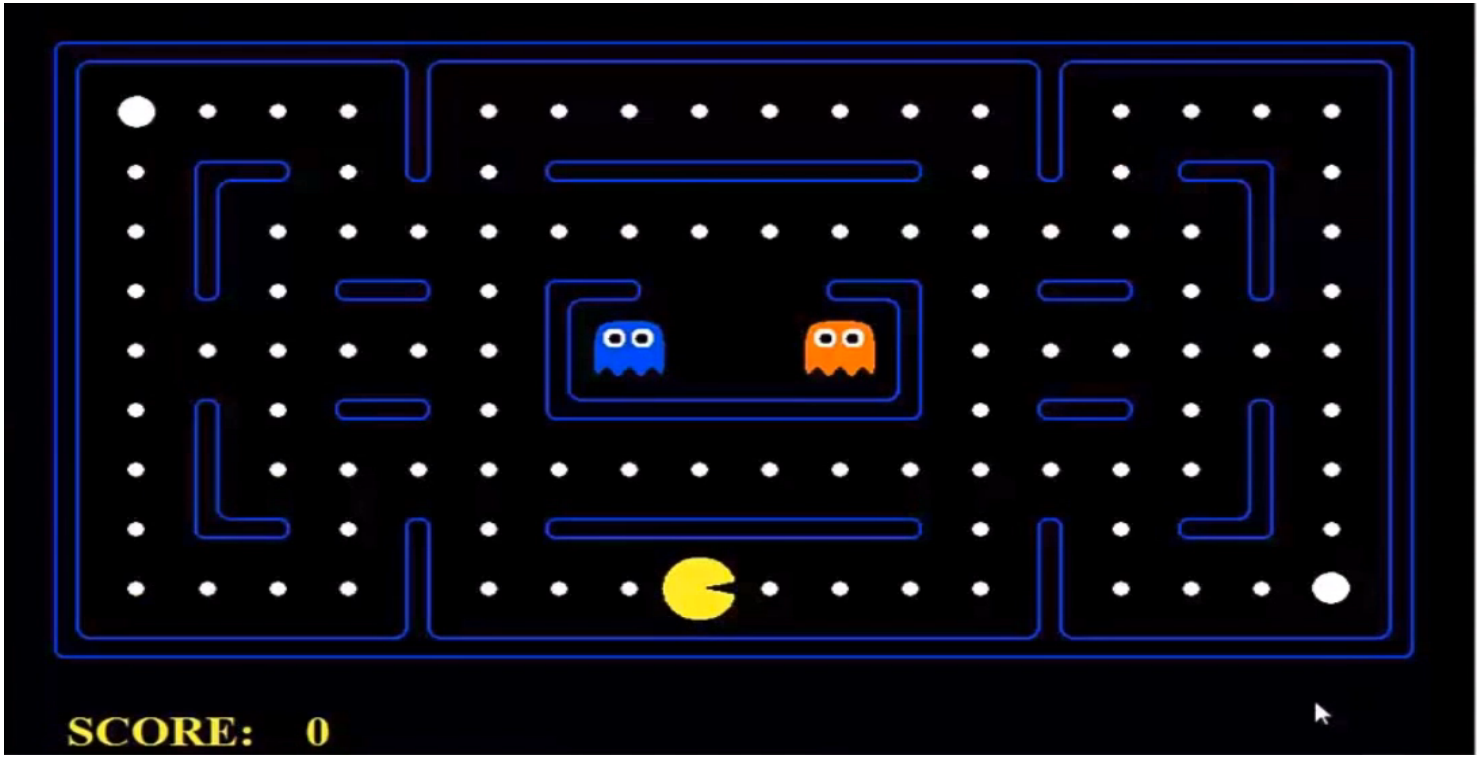
vs.



Course Topics

- Part I: Making Decisions
  - Fast search / planning
  - Constraint satisfaction
  - Adversarial and uncertain search
- Part II: Reasoning under Uncertainty
  - Bayes' nets
  - Decision theory
  - Machine learning
- Throughout: Applications
  - Natural language, vision, robotics, games, …

Assignments: Pac-man



Originally developed at UC Berkeley:
http://www-inst.eecs.berkeley.edu/~cs188/pacman/pacman.html

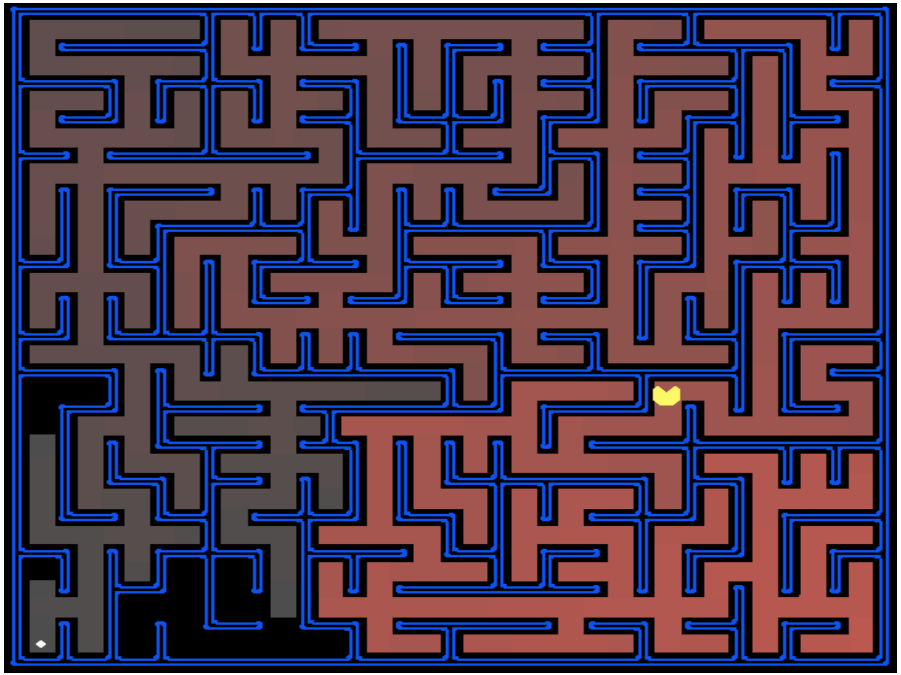
PS1: Search

Goal:

- Help Pac-man find his way through the maze

Techniques:

- Search: breadth-first, depth-first, etc.
- Heuristic Search: Best-first, A*, etc.



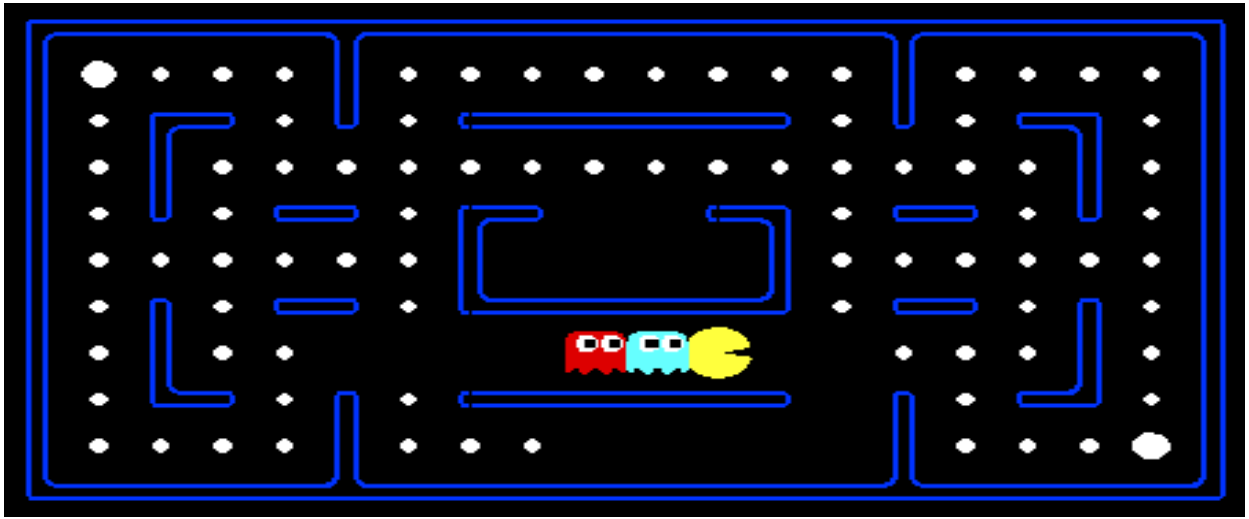
PS2: Game Playing

Goal:

- Play Pac-man!

Techniques:

- Adversarial Search: minimax, alpha-beta, expectimax, etc.



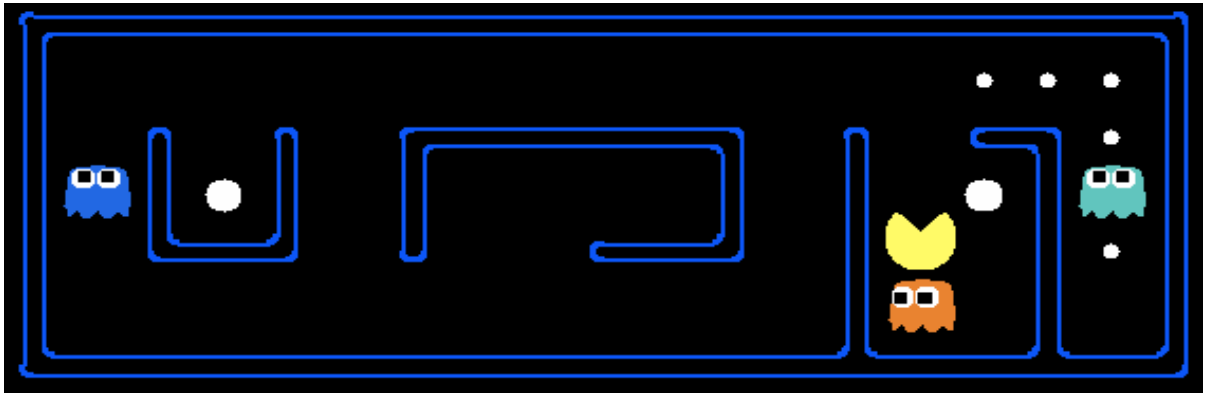
PS3: Planning and Learning

Goal:

- Help Pac-man learn about the world

Techniques:

- Planning: MDPs, Value Iterations
- Learning: Reinforcement Learning



PS4: Ghostbusters

Goal:

- Help Pac-man hunt down the ghosts

Techniques:

- Probabilistic models: HMMS, Bayes Nets
- Inference: State estimation and particle filtering

To Do

- Look at the course website: https://courses.cs.washington.edu/courses/cse473/17wi/
- Do the python tutorial (not graded)