Overview

- Administrivia
- History/applications
- Modeling agents/environments

What can we learn from the past?

Pre-AI developments

Philosophy: intelligence can be achieved via mechanical computation (e.g., Aristotle)

Church-Turing thesis (1930s): any computable function is computable by a Turing machine

Real computers (1940s): Heath Robinson, Z-3, ABC/ENIAC

Birth of AI, early successes

Birth of AI (1956): Workshop at Dartmouth College (John McCarthy, Marvin Minsky, etc.); aim for general principles

Every aspect of learning or any other feature of intelligence can be so precisely described that a machine can be made to simulate it.

Checkers (1952): Samuel's program learned weights and played at strong amateur level

Problem solving (1955): Newell & Simon's Logic Theorist: prove theorems in Principia Mathematica using search + heuristics; later, General Problem Solver (GPS)

Overwhelming optimism...

Machines will be capable, within twenty years, of doing any work a man can do. —Herbert Simon

Within 10 years the problems of artificial intelligence will be substantially solved. —Marvin Minsky

I visualize a time when we will be to robots what dogs are to humans, and I'm rooting for the machines. —Claude Shannon

...underwhelming results

Example: machine translation:

The spirit is willing but the flesh is weak.

(Russian)

The vodka is good but the meat is rotten.

1966: ALPAC report cut off government funding for MT
### Summary

**Problems:**
- **Limited computation:** search space grew exponentially, outpacing hardware ($100! \approx 10^{10^{57}} > 10^{80}$)
- **Limited information:** complexity of AI problems (number of words, objects, concepts in the world)

**Contributions:**
- Lisp, garbage collection, time-sharing (John McCarthy)
- **Key paradigm:** separate modeling (declarative) and algorithms (procedural): program has internal model of the external world, search for goal using model

---

### Knowledge-based systems (1970s-1980s)

**Knowledge is power**

**Expert systems:** elicit specific domain knowledge from experts in form of rules:

```
if [premises] then [conclusion]
```

**DENDRAL:** infer molecular structure from mass spectrometry

**XCON:** convert customer orders into parts specification; save DEC $40 million a year by 1986

---

### Knowledge-based systems

**Contributions:**
- First real application that impacted industry
- Knowledge helped curb the exponential growth

**Problems:**
- Knowledge is not deterministic rules, need to model uncertainty
- Requires considerable manual effort to create rules, hard to maintain

---

### Modern AI (1990s-present)

**Better models:**
- Pearl (1988): promote probability, Bayesian networks in AI to model uncertainty coherently (Bayes rule in 1700s)
- Speech recognition using HMMs

**More data:**
- Trillions of words in English, billions of images on Web
- Tune million of parameters using statistical principles, e.g., maximum likelihood (Gauss in 1800s, Fisher in 1910s)
- Key: use learning to solve the lack of information

---

### Big milestones

- 1997: IBM's Deep Blue chess computer defeats world champion Gary Kasparov
- 2005: Stanford's Stanley drives 132 miles in desert to win DARPA Grand Challenge
- 2011: IBM's Watson defeats humans at Jeopardy!

---

### Search/planning

**Route planning:** (e.g., Google maps); search + heuristics

**Logistics planning:** hospitals organize bed schedules, staff rotations

**Formal verification:** prove correctness of hardware/software (e.g., NASA, Intel); logic/theorem proving
Prediction

**Recommendation systems**: users rate/buy products (e.g., Netflix Prize)

**Medical diagnosis**: given symptoms, predict diseases

Computer vision

**Check reading**: automatic tellers widespread

**Face detection/recognition**: widespread on digital cameras

**Object recognition**: 10 million labeled images, 100,000 object categories

**Scene understanding**: partition image and label regions with building, sky, road, etc.

**Activity recognition**: infer high-level concept from low-level data (UIUC)

Robotics

**Disaster areas**: after earthquakes, surveillance robots check for survivors and structural integrity

**Household chores**: towel folding [Abbeel at Berkeley]

**Robotic surgery**: less invasive, can perform some actions better than humans

**Autonomous vehicles**: (e.g., Google Car)

Natural language processing

**Spam filtering**: 80-90% of all messages are spam; adversarial

**Information retrieval**: rank web pages based on relevance to query

**Machine translation**: Google Translate handles 64 languages

**Speech recognition**: personal assistants (Siri, Google Now)

Summary

**in vitro**

**reasoning/search**

**in vivo**

**perception/uncertainty**

AI: the study and design of intelligent **agents**

Ingredients:

- **Computation**: exponential search space
- **Information**: tons of **noisy** data
- **Tools**: logic, probability, statistics, optimization
**Framework**

- **Environment**
- **Agent**
- **Percepts**
- **Program**
- **Actions**
- **Sensors**
- **Actuators**

**Utility**: measure performance on desired task  
*Our goal*: build an agent that obtains high utility

**Examples**

**Robotics**:
- Percepts: sensor measurements (cameras, microphones, laser range finders, sonar, GPS)
- Actions: move, turn, grasp, etc.

**Computer vision**:
- Percepts: pixels of an image
- Actions: produce description of objects in image

**Natural language**:
- Percepts: request in context (e.g., *Where is the nearest airport?*)
- Actions: response (e.g., *San Jose*)

**Games**:
- Percepts: state of a chess board
- Actions: make legal chess moves

**Human agents**

**Brain (hardware)**: 100 billion neurons, 7,000 connections per neuron; topic of neuroscience; inspiration for some models (neural networks)

**Mind (software)**: cognitive science studies human intelligence and behavior; share some of same techniques as AI (probabilistic models)

**Analogy**: brains : intelligence :: wings : flight

**Rational agents**

**Ideally**: obtain agent that maximizes expected utility!

\[
\alpha^* = \arg \max_{\alpha:Rational} \text{ExpectedUtility}(\alpha)
\]

**Issue**:
- Real-world tasks are too complex to formalize exactly
- Example: what are utility (performance measure) and percepts (input) for machine translation?
- Example: in chess, board is fully-observed but opponent is not

**Model-based agents**

**Model**: a simplification of the original task (environment, utility)

**Methodology for solving AI tasks**

1. **Real-world task**
2. **Modeling**: make simplifications / assumptions
   - **Formal task**
3. **Algorithms**: find rational agent in simplified task
   - **Solution**
Making decisions

Task: I give you 2 dollars if you raise your left hand, 5 dollars if you raise your right hand.

Model:
- Environment: I'm telling truth
- Utility: amount of money
- Rational agent: raise right hand

Making decisions under uncertainty

Task: You choose a number $n$. I flip two coins. If $n$ heads show up, you get $n^2$ dollars.

Model:
- Environment: I'm telling truth, fair coin
- Utility: amount of money
- Rational agent:
  - Action $n = 2$: $\text{ExpectedUtility} = \frac{1}{2} \cdot 2^2 = 1$
  - Action $n = 1$: $\text{ExpectedUtility} = \frac{1}{2} \cdot 1^2 = 0.5$
- Therefore, choose $n = 2$

Flip coins, get HT; got 0 instead of 1; still rational?

Lesson: under uncertainty, must think about expected utility

A clinical task

- Three drugs (A, B, C), each with some probability of success.
- Conduct 20 trials; in each trial, choose one of the drugs.
- Goal: maximize number of successes.

Desiderata / course topics

Reason about goals: what will I get if I try this sequence of actions?
  - Search, planning, minimax

Deal with uncertainty: don't know what will happen, ambiguity in language, noise in sensor readings
  - MDPs, probabilistic graphical models

Learn from experience: results of actions provide information to improve utility over time
  - Machine learning, reinforcement learning

Interface with the human world: tasks involve humans
  - Vision, robotics, language

Summary

Diverse real-world applications: language, vision, robotics, planning

Challenges: limited computation, limited information

Methodology: modeling + algorithms