
PLURALSIGHT

Introduction to Artificial Intelligence

Welcome!

Greg Swanson
Instructor, Pluralsight

Agenda

- Today:
  - RL & Agents Discussion
  - AI Applications
  - Survey
  - Final Project & Course Wrap Up

Today’s Schedule

9:00 - 10:30 AM PT RL & Agents Discussion:

Break 9:50 - 10:00 AM PT

10:30 - 11:30 AM PT Applications & Resources

11:30 - 12:00 PM PT Survey

Lunch 12:00 - 1 PM PT

1:00 - 5:00 PM PT Final Project Lab Time & Course Wrap Up

How we're going to work together

- You'll have a copy of the course materials shortly
  - We'll be using Jupyter notebooks (explained shortly)

- You'll be following along in the notebook and..
  - doing coding exercises/labs inside the notebook as well

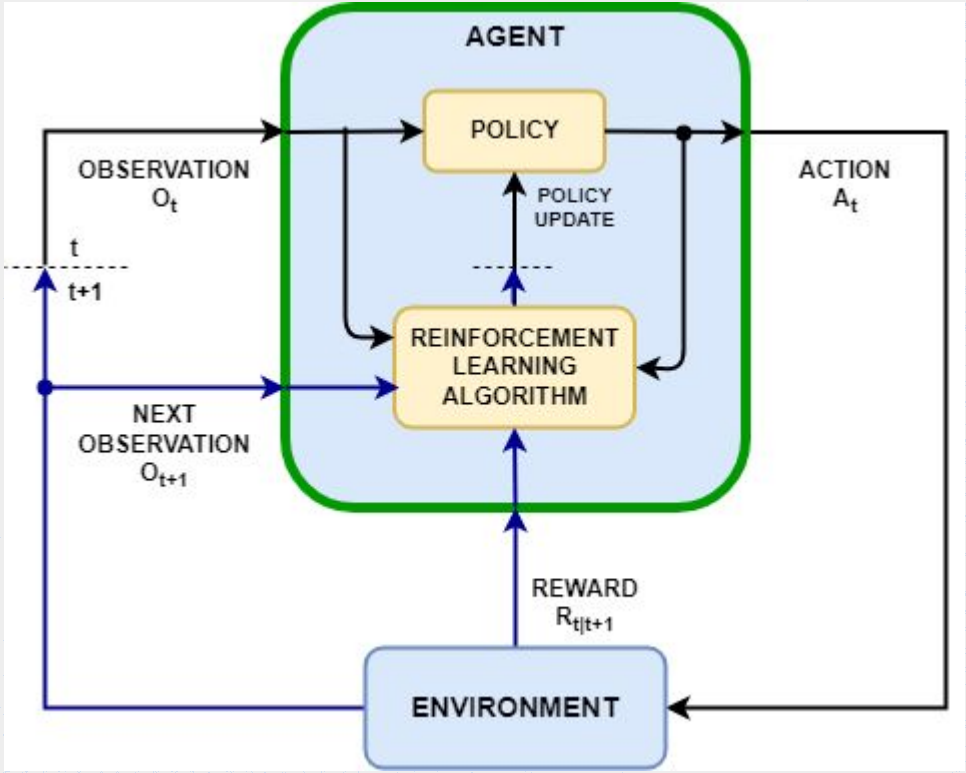
Today’s Key Learning Objectives

- Identify different agents and search mechanisms and their uses
- Demonstrate AI applications with SOTA Models (state-of-the-art)
- Use AI in the completion of a project on your own

Reinforcement Learning

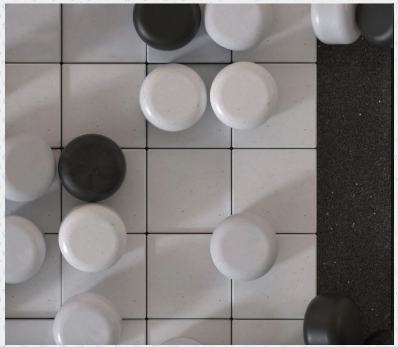
Reinforcement Learning: an area of machine learning that trains an agent to take a suitable action to maximize reward in a particular situation

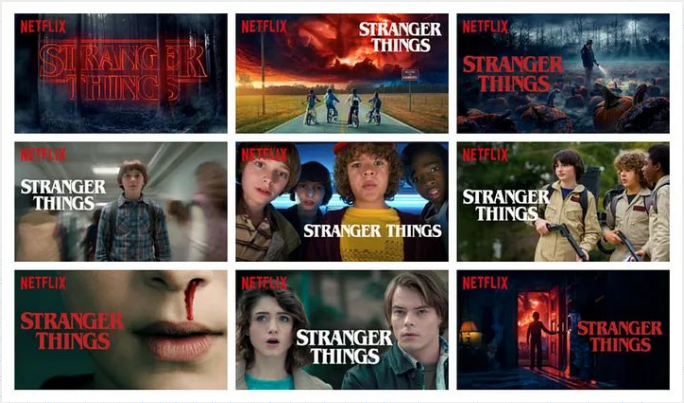
Agents?

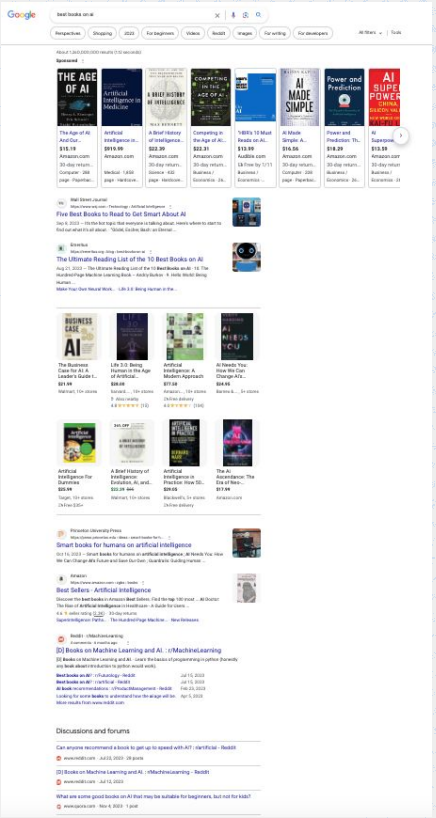


Examples of reinforcement learning

- Netflix Thumbnail Optimization
- New web pages in google search
- Alpha go



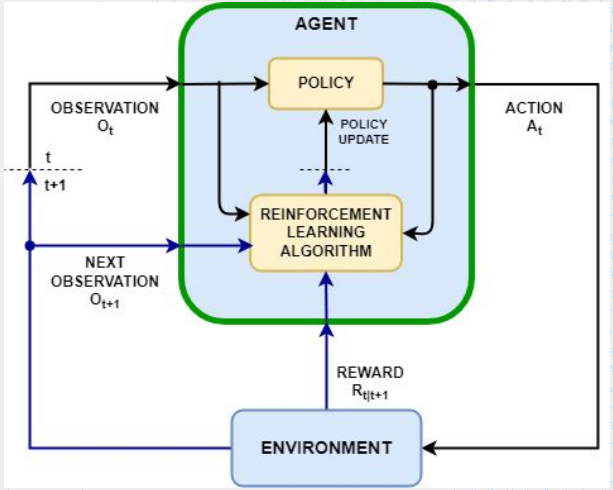




Reinforcement Learning & AI Agents Part II

Reformulating RL as a Markov Decision Process

- **Markov Decision Process (MDP):** a stochastic decision-making process that uses a mathematical framework to model the decision-making of a dynamic system
- **Can you think of any examples?**



Examples of MDP

Routing Problems

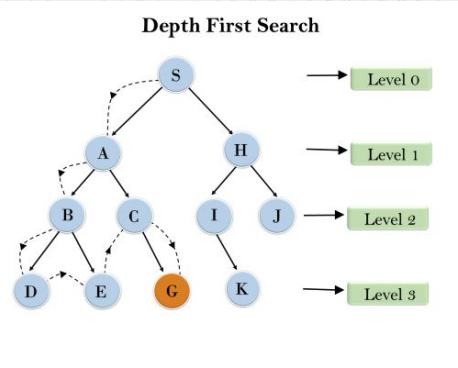
Managing phone wait time

More on Agents

- There are multiple types of agents from Simple Reflex agents to Intelligent Agents
  - **Simple Reflex:** Rules based approach (can't learn)
  - **Intelligent:** an agent that can learn from its environment to achieve its goals e.g. a thermostat

- **Which type of agent are we using in our RL examples?**

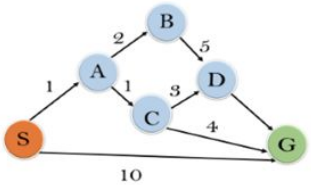
Agents Search for Optimal Solutions



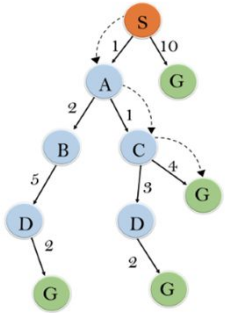
- **Uninformed Search**
  - This “Blind Search” does **NOT** contain any domain knowledge and will search every possible option until it finds what it’s looking for
  - Examples: Breadth-first Search, Depth-first search
- **Informed Search**
  - This “Heuristic Search” **DOES** contain domain knowledge and can find a solution more efficiently than an uninformed search
  - Examples: Greedy Search, A* Search

Informed Search Examples: Greedy Search

- **Step 1:** Place the starting node into the OPEN list.
- **Step 2:** If the OPEN list is empty, Stop and return failure.
- **Step 3:** Remove the node n, from the OPEN list which has the lowest value of h(n), and places it in the CLOSED list.
- **Step 4:** Expand the node n, and generate the successors of node n.
- **Step 5:** Check each successor of node n, and find whether any node is a goal node or not. If any successor node is goal node, then return success and terminate the search, else proceed to Step 6.
- **Step 6:** For each successor node, algorithm checks for evaluation function f(n), and then check if the node has been in either OPEN or CLOSED list. If the node has not been in both list, then add it to the OPEN list.
- **Step 7:** Return to Step 2.

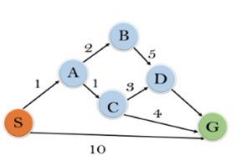


State	h(n)
S	5
A	3
B	4
C	2
D	6
G	0

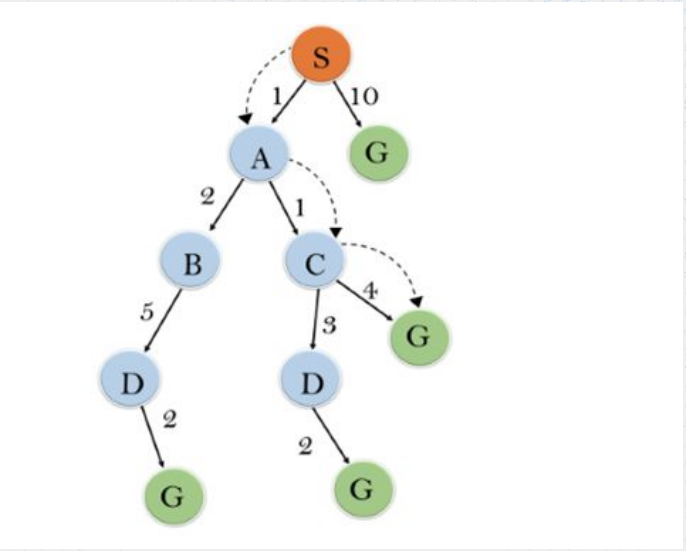


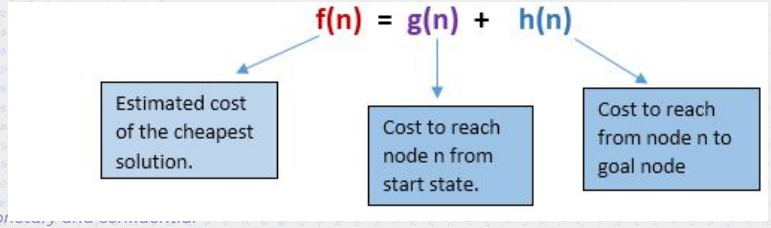
Informed Search Examples: A* Search

- **Step 1:** Place the starting node in the OPEN list.
- **Step 2:** Check if the OPEN list is empty or not, if the list is empty then return failure and stops.
- **Step 3:** Select the node from the OPEN list which has the smallest value of evaluation function (g+h), if node n is goal node then return success and stop, otherwise
- **Step 4:** Expand node n and generate all of its successors, and put n into the closed list. For each successor n', check whether n' is already in the OPEN or CLOSED list, if not then compute evaluation function for n' and place into Open list.
- **Step 5:** Else if node n' is already in OPEN and CLOSED, then it should be attached to the back pointer which reflects the lowest g(n') value.
- **Step 6:** Return to **Step 2**.



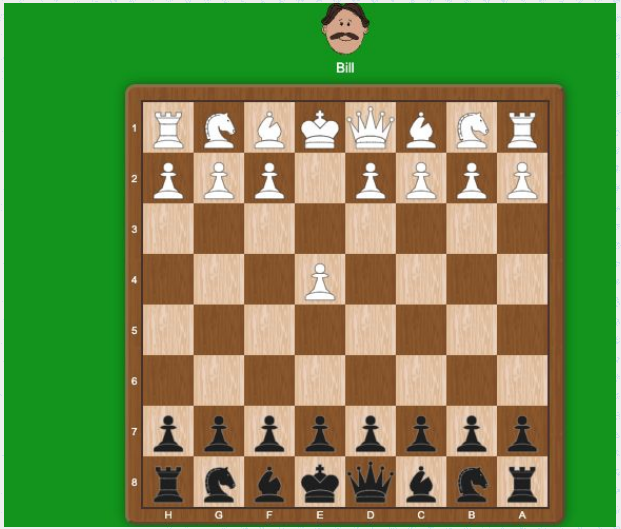
State	h(n)
S	5
A	3
B	4
C	2
D	6
G	0

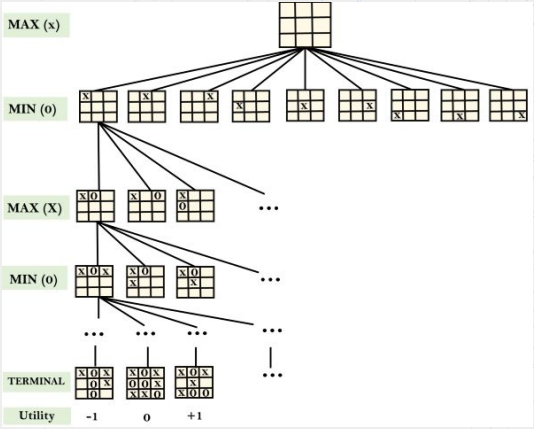




Adversarial Search

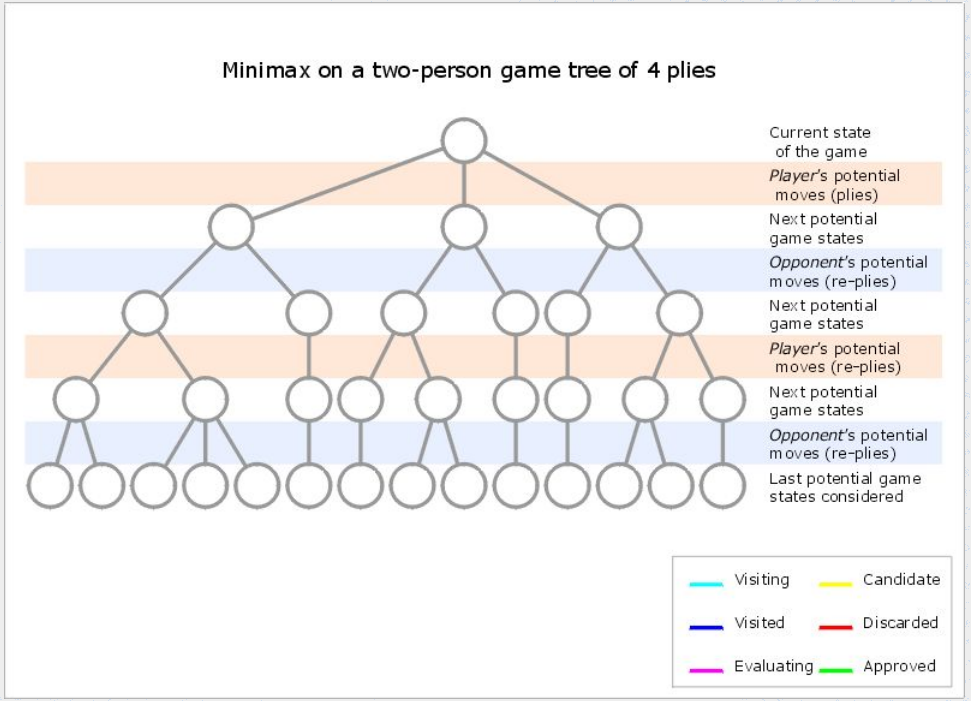
- **Adversarial Search:** a search, where we examine the problem which arises when we try to plan ahead of the world and other agents are planning against us
- Think Game AI
  - Perfect vs Imperfect information
  - Deterministic vs Chance





MiniMax Algorithm

- **MiniMax Algorithm:** a recursive or backtracking algorithm which is used in decision-making and game theory. It provides an optimal move for the player assuming that opponent is also playing optimally
- Searches through the game tree to maximize benefit for agent while minimizing opponent benefit



AI Applications (See Notebook)

Let’s review what we’ve learned

Why study this subject?

- To enhance your understanding of Artificial Intelligence
- Get hands-on experience working with AI in a project
- So that you can better work with AI products within Intuit

Objectives

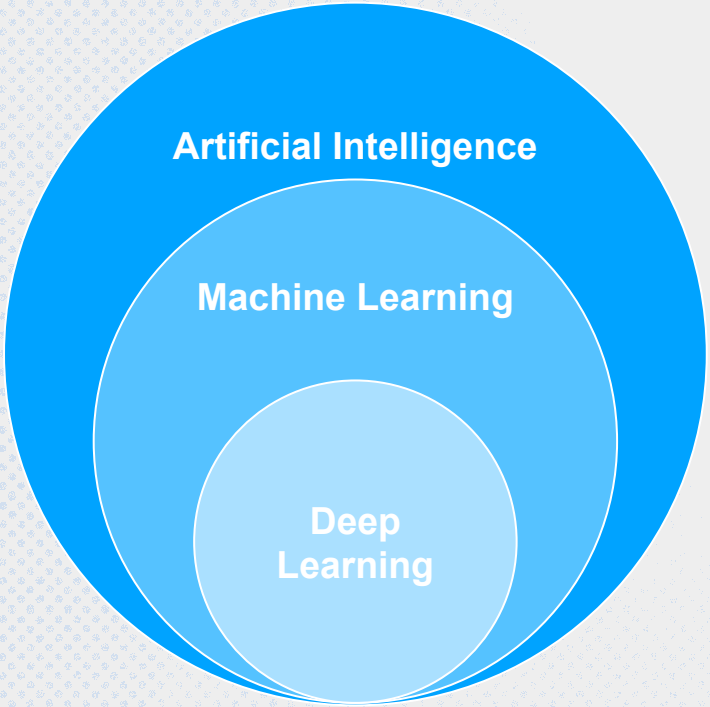
At the end of this course, you will be able to:

- Describe what artificial intelligence is and how it can be used in business applications
- Identify the different agents and search mechanisms and their specific uses
- Use AI in the completion of a capstone project

(Applied) Artificial Intelligence

The ability of a computer system to deal with ambiguity, by making predictions using previously gathered data, and learning from errors in those predictions in order to generate newer, more accurate predictions about how to behave in the future

How it all Relates



Artificial Intelligence
Teaching machines to behave like humans

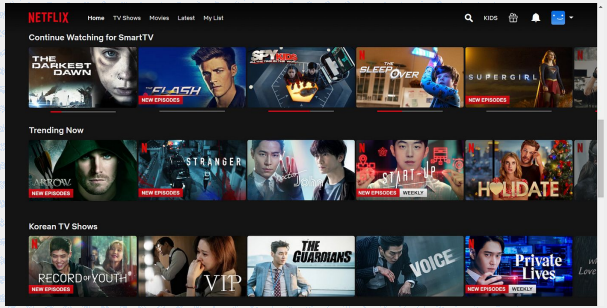
Machine Learning

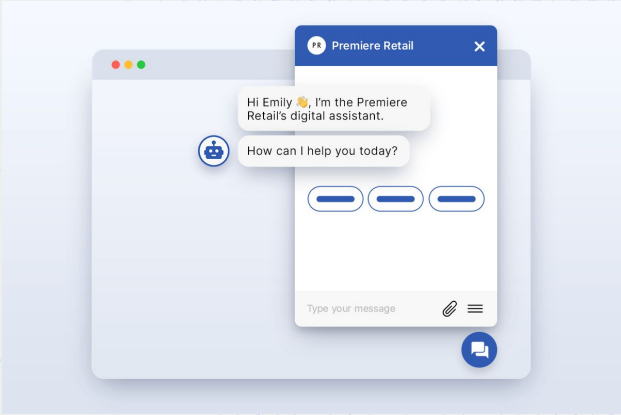
A model that learns to do one task and improves with more data (i.e. examples).

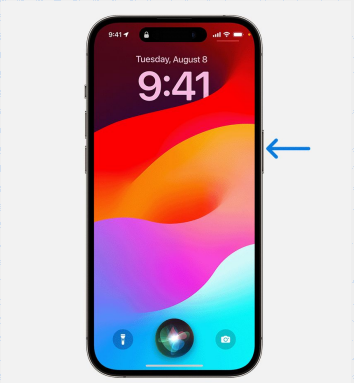
Deep Learning

Branch of Machine Learning that can handle many nuances and challenging data (e.g. text, image)

AI is ubiquitous today







Narrow VS Broad AI

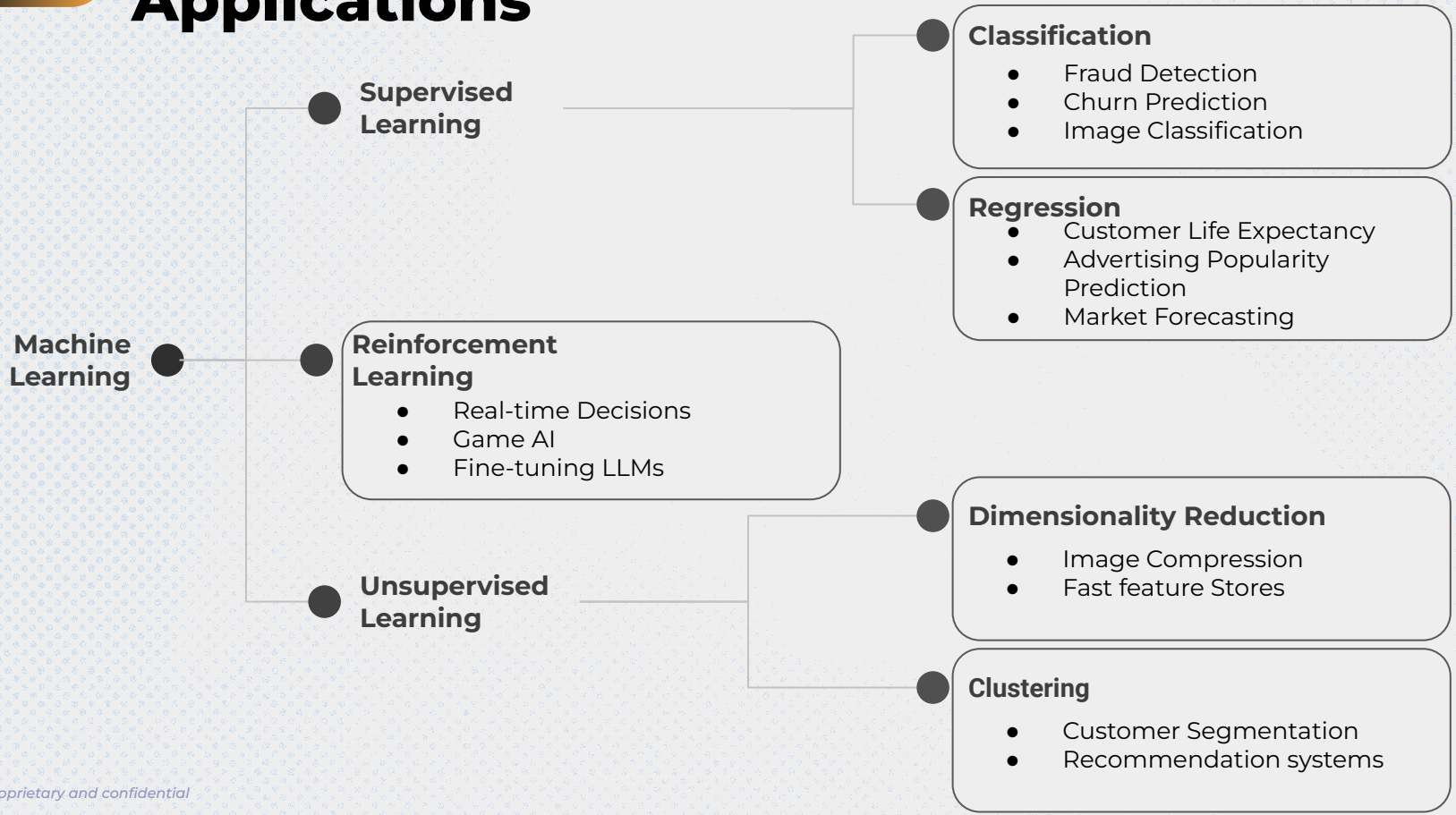
Narrow AI:

- Already exists
- Less than human level intelligence
- Really good at focused set of tasks
- Also called “weak AI”
- Self-driving vehicles
- Digital Assistants
- Game AI

Broad AI:

- Doesn’t exist yet
- Human level (or better) intelligence
- Able to perform unseen tasks very well
- Also called “General AI”
- Think Sci-fi movies e.g. C-3PO, HAL, Smarthouse, I, Robot, etc.

Translating Machine Learning to Applications



We used quite a few technologies in class:

VOWPAL WABBIT

HUGGING FACE

NumPy

SciPy

pandas

Resources for the future

Going Forward:
Learning Resources for You

Libraries:

- Pandas
- Sci-Kit Learn
- Keras
- Vowpal Wabbit
- Pytorch
- Tensorflow
- NLTK
- Gensim

Textbooks & Guides:

- Intro to Statistical Learning
- Deep Learning Notebooks

SOTA Models, implementations & comps:

- Papers with code
- Huggingface
- Kaggle

Ethics of AI:

- MIT Articles
- UNESCO Perspective
- IBM's perspective
- OpenAI Senate Hearing

Survey Time

Before completing the course, you need to fill out this survey

What was your favorite part of the course?

Is there anything you would change or wish we covered?

Final Lab (See Notebook)

Any last questions?

Thank you!

If you have any additional questions, please ask!If