Machine Learning Engineering

Welcome

- CS 5781 - Machine Learning Engineering
- A new class for an emerging topic
- Completely designed for virtual teaching

Class Context

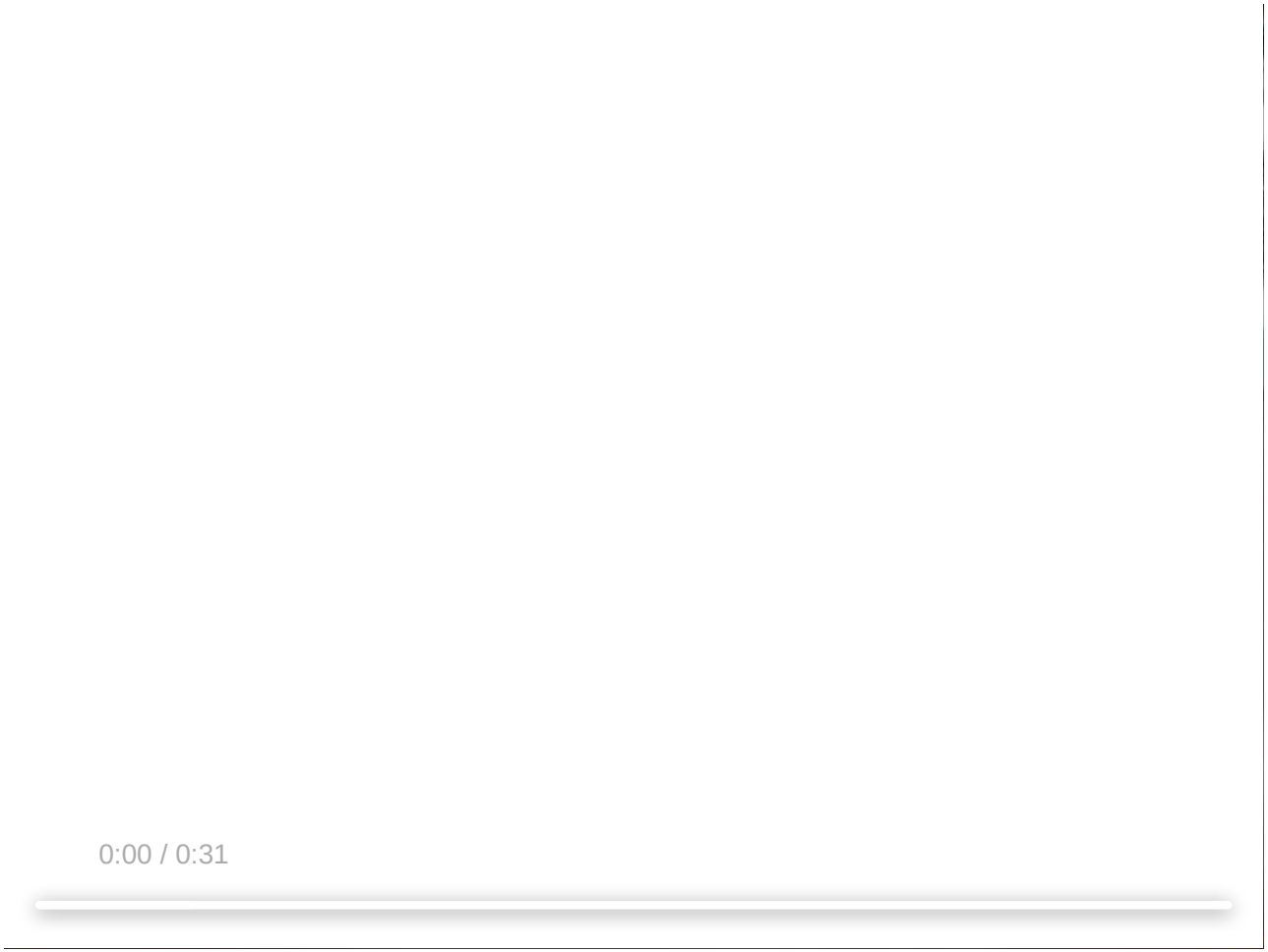
- Development of deep learning models
- Deep learning models in industrial context
- Programming large systems

Current Context: MLE

Indeed's best jobs of 2019

Rank	Job title	% growth in # of postings, 2015–2018
1	Machine Learning Engineer	344%
2	Insurance Broker	242%
3	Full-stack Developer	206%
4	Insurance Advisor	190%
5	Litigation Attorney	168%
6	Litigation Associate	165%
7	Dental Hygienist	157%
8	Associate Attorney	149%
9	Realtor	138%

Future Context: Self-Driving



Future Context: Code Gen

Future Challenges: ML in Society

A Case for Banning Facial Recognition

Question

What does it look like to be an engineer in this context?

Personal Introduction

Hi!

Academic Work

- Website: **http://rush-nlp.com/**
- Area of Study: Natural Language Processing (NLP)
- Area of Study: Deep Learning

Academic Work: Projects

- Automatic text summarization
- Accurate math OCR
- Machine learning on cell phones

My Path

- Coder -> Student -> Industry -> Professor
- Professor at Harvard for 5 years
- Moved to Cornell Tech last year!

Intro: Engagement

- Open-source development projects for NLP
- Part-time at **Hugging Face**

TA

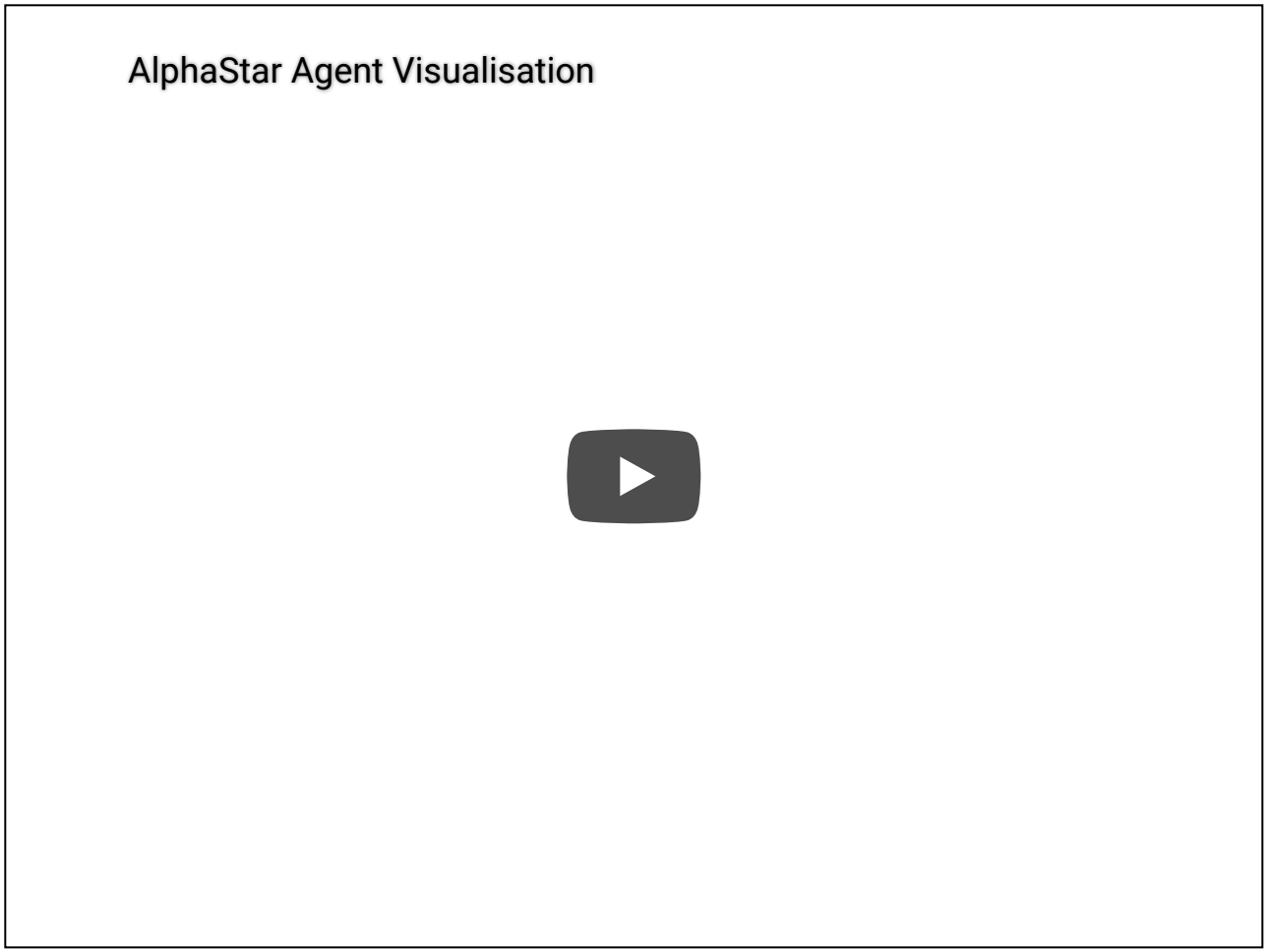
- Ge Gao
- PhD Student in NLP / Machine Learning

Class Introduction

Class Focus

- **Machine Learning** Engineering
- Machine Learning **Engineering**
- Focus: software engineering behind machine learning

Future Context: Multi-Agent



Applied Machine Learning

- Coverage of different models and learning setups
- Focus on algorithms and mathematical underpinnings
- Broad coverage of the field and its future

Machine Learning Engineering

- Coverage of only one model (neural network)
- Focus on implementation details and design
- Deep dive into its positives / negatives
- (For those who care about the weeds)

Machine Learning

- Rich and interesting field
- Buildnig models is a core skill
- Probabilistic reasoning for decision making

Hidden Factor

Many recent successes based on:

- Hardware
- Tooling
- Brute-force search

Skill Set of a ML Engineer

- Math
- Experimentation
- *Systems*

Machine Learning Systems

Machine Learning Engineering

- ML practitioners build large-scale mathematical systems.
- Tooling has been key to speed up ML development.
- Most work done in *Deep Learning frameworks*.

Deep Learning Frameworks

- Implement mathematical functions as efficient code
- Provide organization and structure to ML projects
- Allow for easy training and deployment
- Think: "Programming language for machine learning"

Popular Frameworks

- **TensorFlow**

- **PyTorch** (Tool used at Tech)

Deep Learning Frameworks

Example of code in PyTorch.

```
class Network(torch.nn.Module):
  def __init__(self):
    super().__init__()
    self.layer1 = Linear(2, HIDDEN)
    self.layer2 = Linear(HIDDEN, HIDDEN)
    self.layer3 = Linear(HIDDEN, 1)

  def forward(self, x):
    h = self.layer1.forward(x).relu()
    h = self.layer2.forward(h).relu()
    return self.layer3.forward(h).sigmoid()
```

Deep Learning Frameworks

- Used for all the major projects shown.
- Provide easy user programming interface
- Connect to fast hardware under the hood

ML Day-to-Day

- Data scientist or ML practitioners and use these systems
- However, an ML Engineer should really know what is going on...

CS 5781 - 2020

We're going to build PyTorch.

Course Outline

My Learning Philophy

- Engineering is learned through implementing
- You don't understand it until the tests pass
- Build your own demos

Future Context: Image Gen

Learning Objectives

- Reason about the requirements for large system systems
- Be comfortable designing and testing mathematical code
- Gain confidence reading large open-source codebases

Learning Non-Objectives

- Rigorous understanding of mathematical foundations
- Development of new or creative models
- Details of state-of-the-art ML systems

Course Style

- Highly applied, focus on building
- Project directed, questions from students
- Interactive and grounded in the project

Course Outline

PyTorch

- Big codebase on CPU and GPU
- Large team of professional developers
- Used in thousands of academic papers
- Deployed by Facebook, Uber, Tesla, Microsoft, OpenAI ...

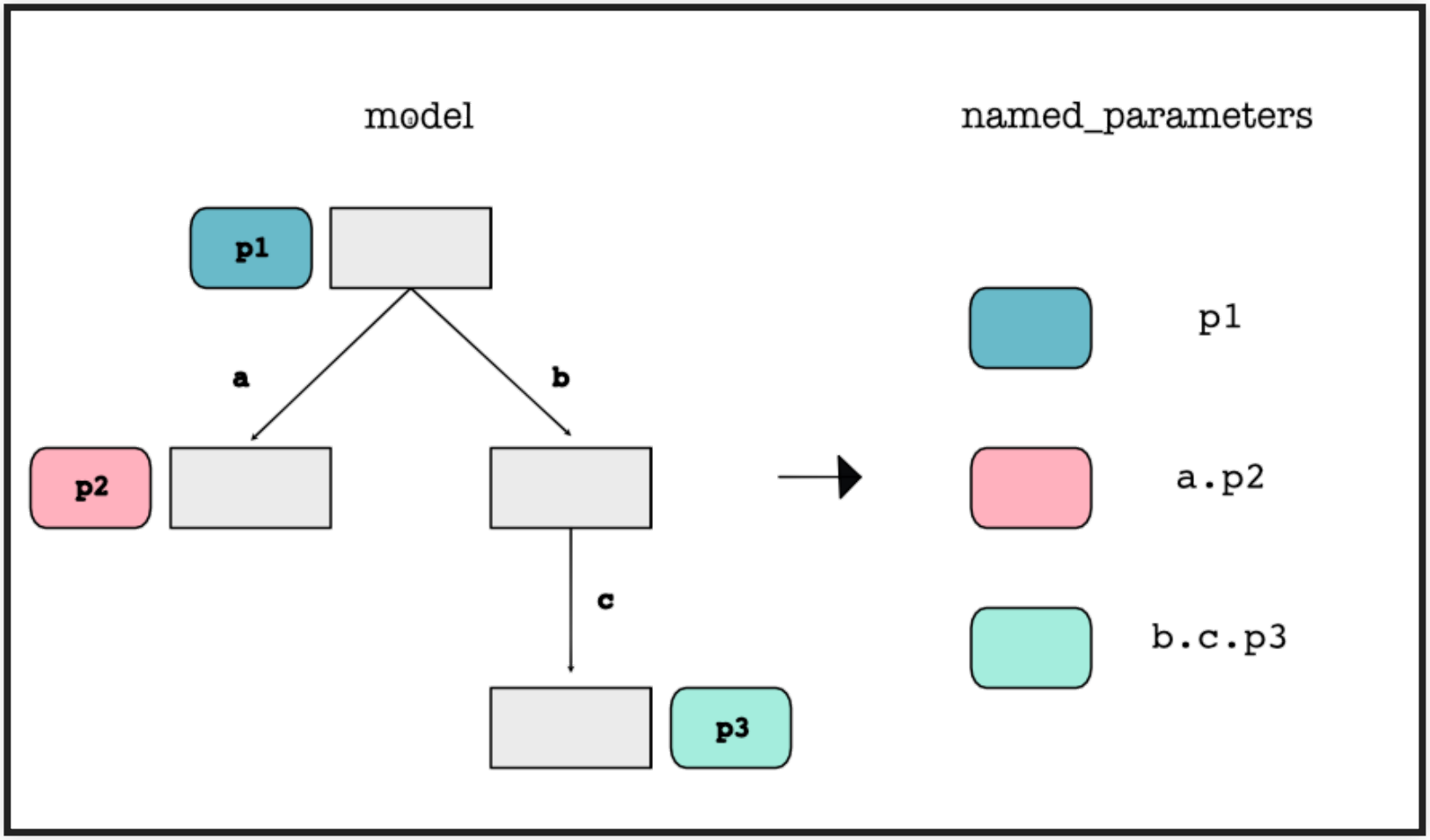
Challenge

- Q: How are you going to build PyTorch?
- A: One commit at a time

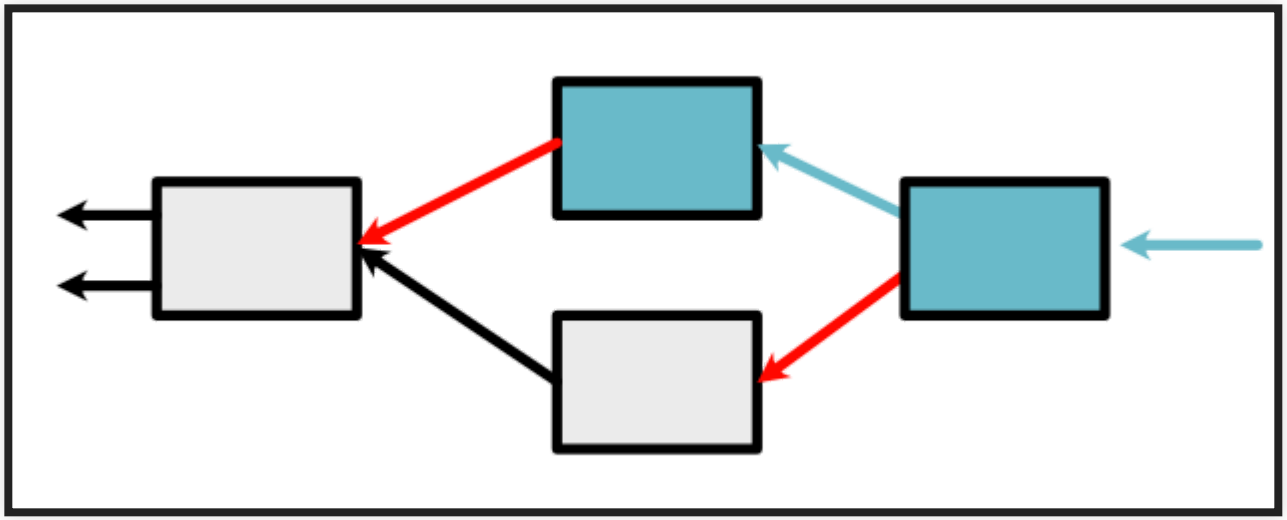
Course Project

- 5 modules walking you through the process
- Each covers a different topic in MLE
- Final module yields a full image recognition system.

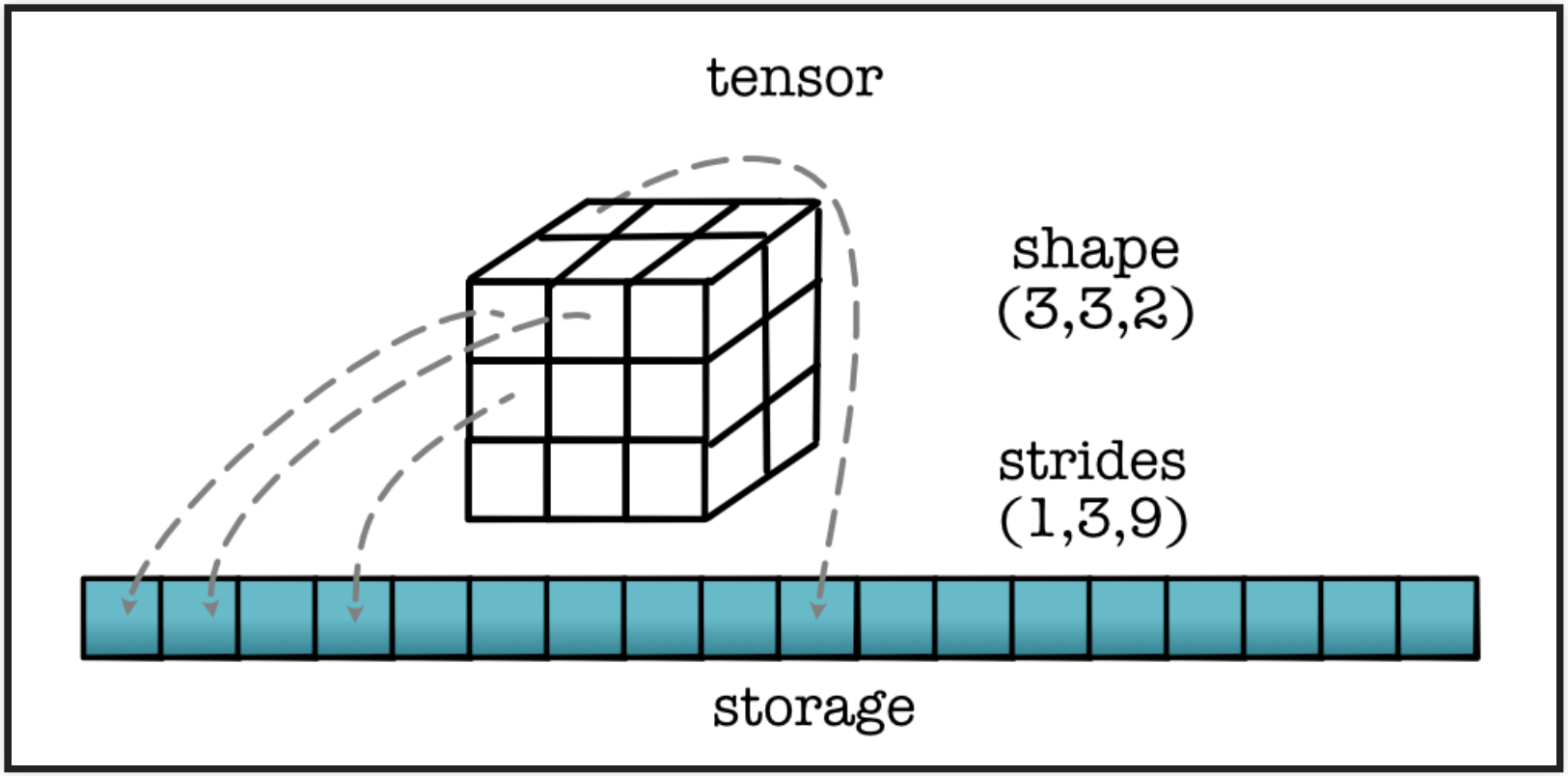
Module 0 - Foundations



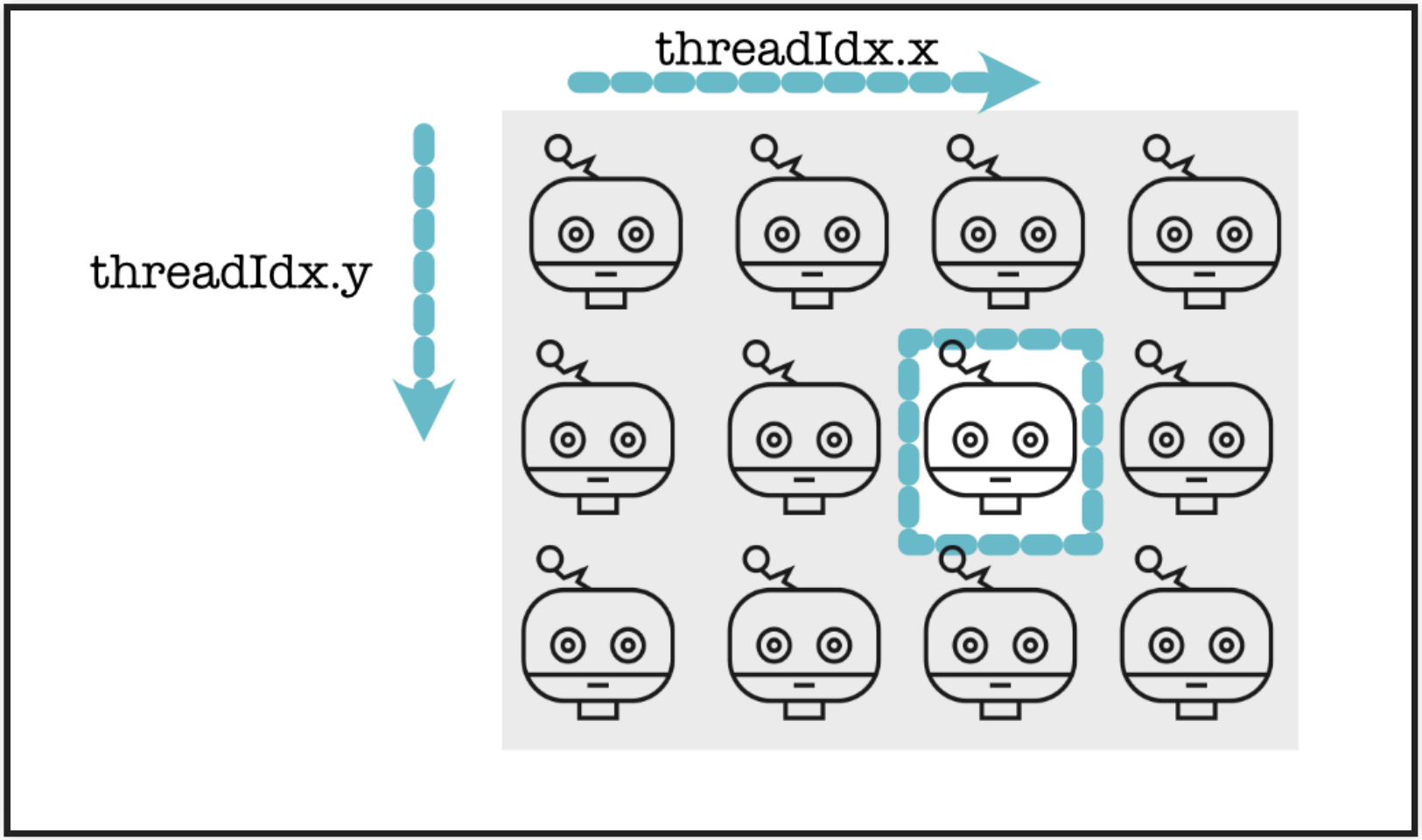
Module 1 - Autodifferentiation



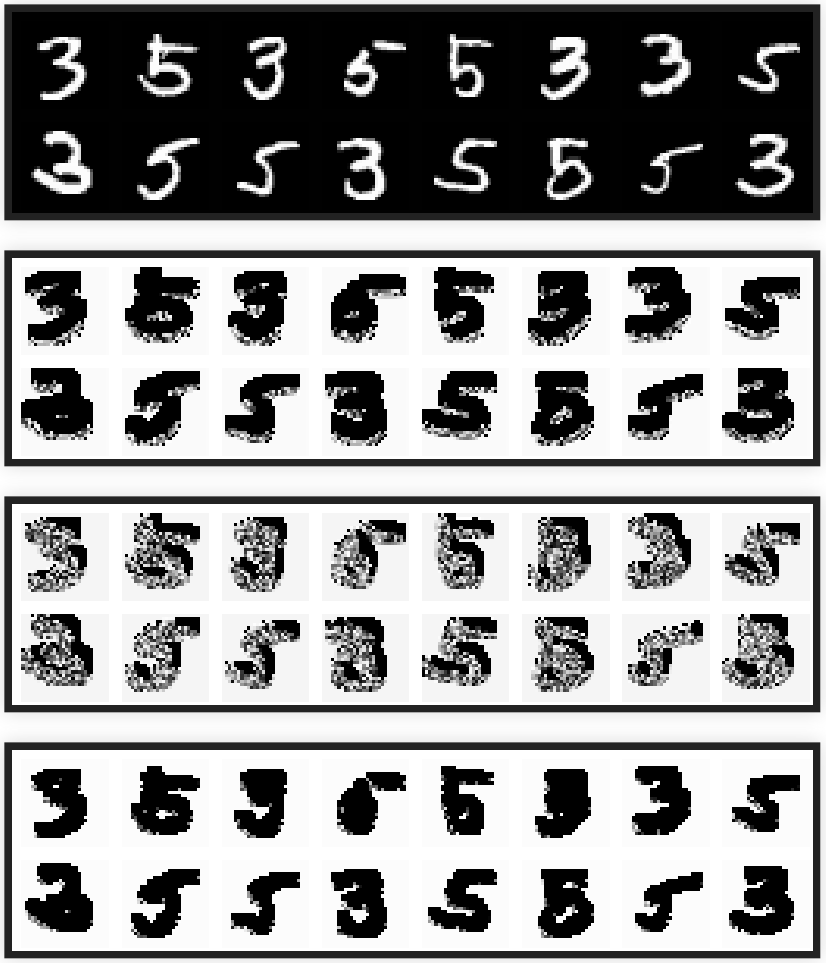
Module 2 - Tensors



Module 3 - Efficiency



Module 4 - Networks



Grading

- Assignments - Completion and Correctness
- In-Class Quizzes
- Assignments are done individually

Course Structure

- Co-working session on Fridays
- Lectures Tuesday Morning and Thursday Evening
- Thursday Morning practical session

Caveats

Course Prerequisites

- Programming experience
- Mathematical notation / calculus experience
- Willingness to debug
- **https://forms.gle/j1VZjwDUVCEqubi36**

Next Lecture

- Getting dev setup
- Getting started for Module-0
- Come ready to program.

Q & A