



# Module 4.4 - What's Next



# Outline

- Review: Course
- What's to Come



# What should I do after MLE?

ChatGPT



# Review





# Class Focus

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



# MLE

- Systems course disguised as ML
- Algorithms implemented Fast
- Testing / Debugging / Scaling



# Module 0 - Foundations

Module



# Module 0 - Foundations

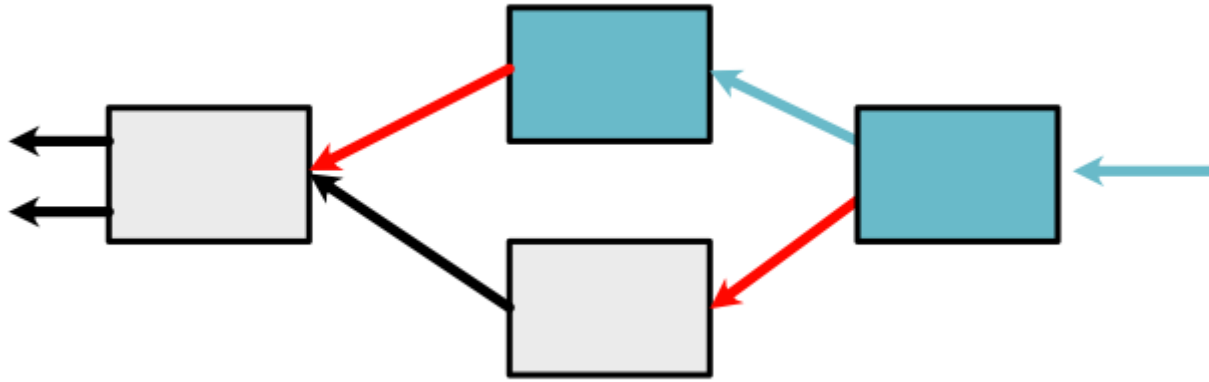
- Testing
- Higher-Order Functions
- Data Structures





# Module 1 - Auto-Diff

Module





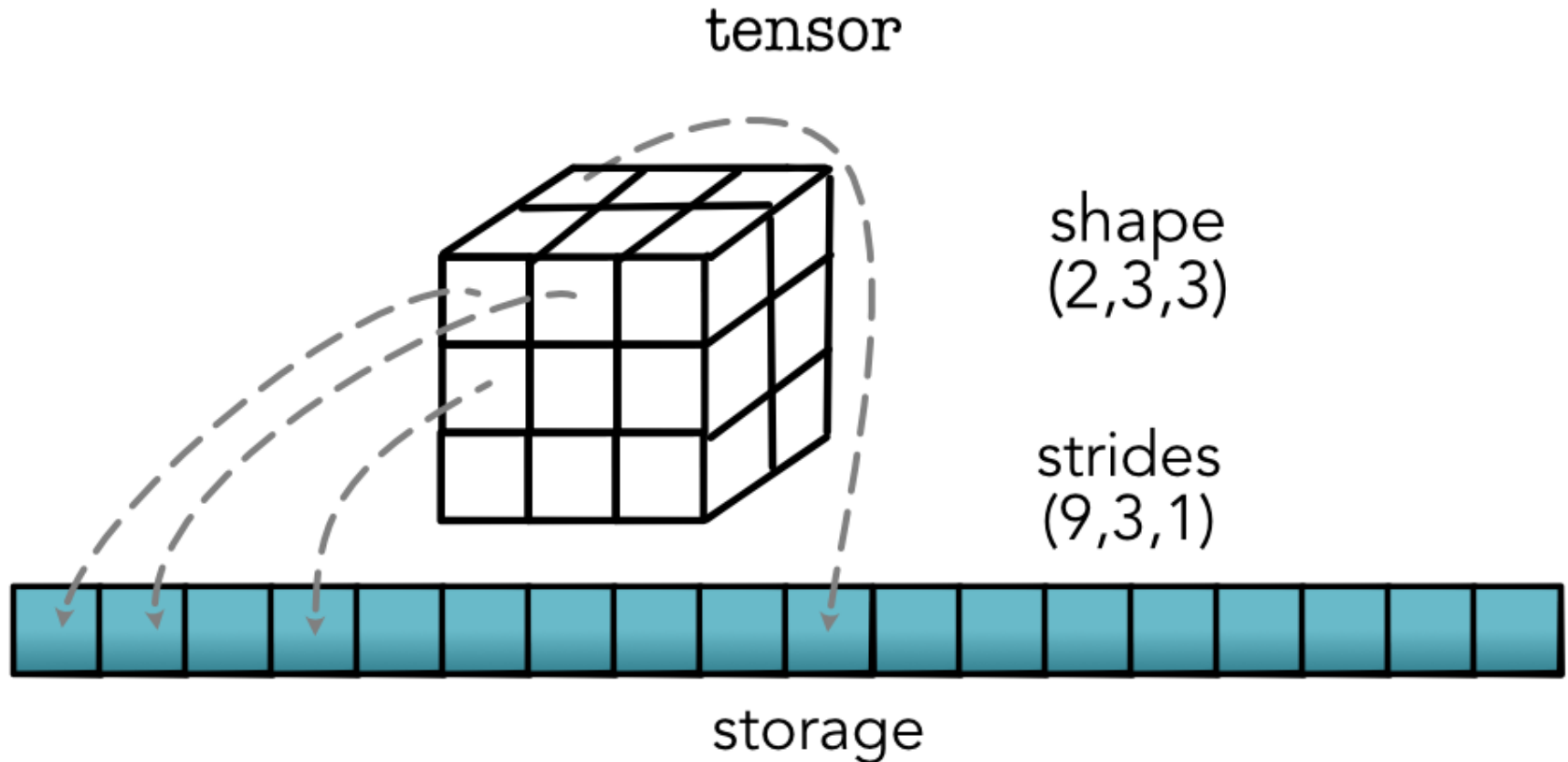
# Module 1 - Auto-Diff

- Variables
- Autodifferentiation
- ML Basics



# Module 2 - Tensors

## Module





# Module 2 - Tensors

- Multidimensional Arrays
- Map / Zip / Reduce
- Broadcasting





# Module 3 - Efficiency

Module



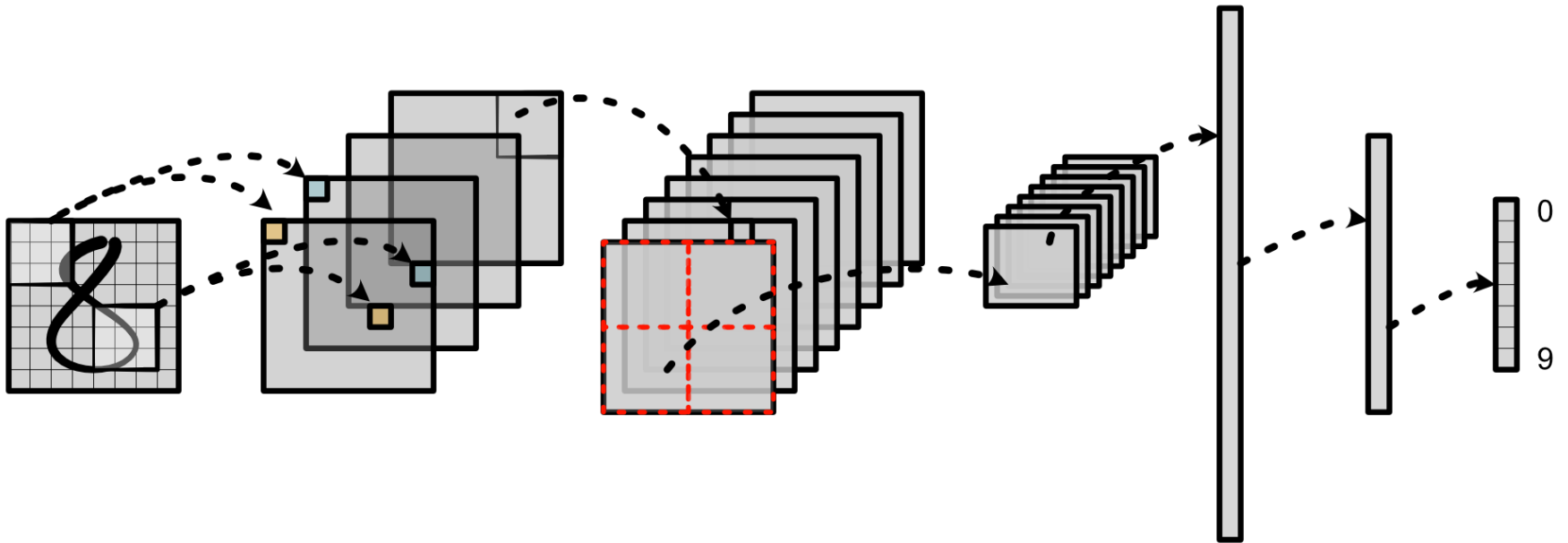
# Module 3 - Efficiency

- JIT / Types
- Parallel
- CUDA / Shared Memory



# Module 4 - Networks

## Module





# Module 4 - Networks

- Convolutions
- Tiling
- Softmax





# What do you know?

- How neural networks works...
- How autodifferentiation works...
- How it all scales ...



# What did you learn?

- Systems are made by humans
- Debugging, testing, organization
- Filing bugs and asking questions



# EdStem

- Justin Chiu - 143 answers!
- Top Students:
  - Hariharan Vijayachandran
  - Courtney Beckham
  - Abhinav Girish



# Course Reviews

<https://apps.engineering.cornell.edu/CourseEval/>





# What's Next



# Many Forks

- ML Engineering
- ML Systems
- ML Models



# ML Engineering

- Many questions beyond training
- Data sets and data availability
- Data collection / preprocessing / robustness



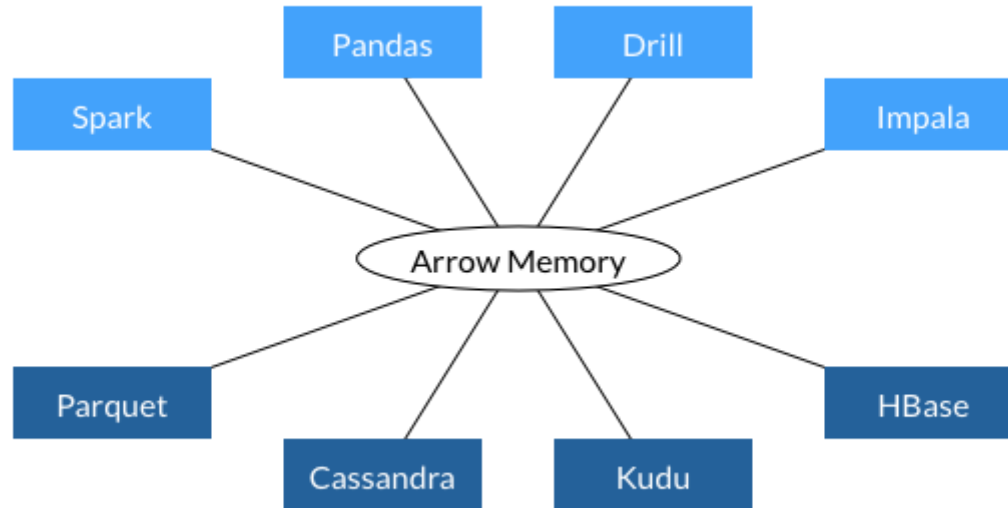
# Tools of the Trade : PyTorch 2

PyTorch 2.0 offers the same eager-mode development and user experience, while fundamentally changing and supercharging how PyTorch operates at compiler level under the hood. We are able to provide faster performance and support for Dynamic Shapes and Distributed.



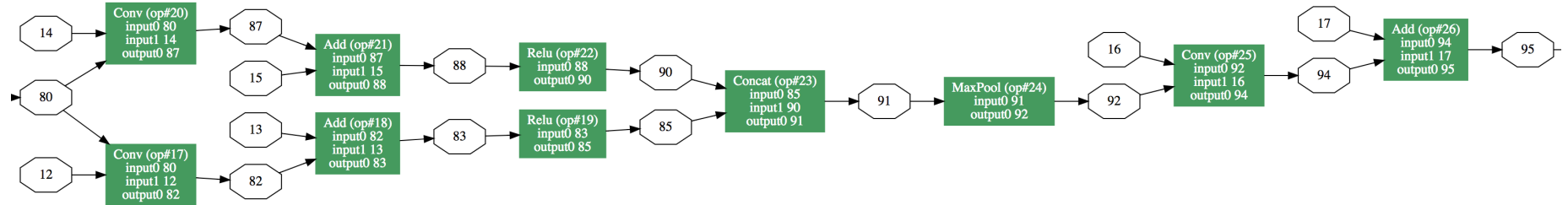


# Tools of the Trade : Arrow





# Tools of the Trade : ONNX





# ML Systems

- Many languages beyond torch
- Different Tensor access forms



# Tools of the Trade: Jax

- Introduces `vector map` in addition to broadcasting
- Can apply a function to a tensor across an entire dimension. `vmap(model.forward, x)`





# Tools of the Trade: Julia

- Programming language for mathematical code
- Pluto -> <https://mybinder.org/v2/gh/fonsp/pluto-on-binder/master?urlpath=pluto>



# ML Models

- Modern models are complicated, but made up of the parts we have seen
- Many are open source and available to play with.



# My Bet

- You can read models!



# NLP Models

- Transformer

<https://github.com/huggingface/transformers/blob/master/src>

---





# Protein Folding

- Distance Prediction

[https://github.com/Urinx/alphafold\\_pytorch/blob/master/netv](https://github.com/Urinx/alphafold_pytorch/blob/master/netv)

---



# Tips of the Trade

- Fancy models are not always necessary
- Build something robust and fast on your hardware.
- Develop multiple expertise to be flexible to users



# Future Steps



# Courses

- Mohamed
- Machine Learning Hardware and Systems





# Deep Learning

- New Version of the course offered in the spring
-



# Independent Study

- I will be taking students for an applied Independent Study
- Topic will be information extraction for zoning



# Graduate Seminar CS 6741

- <https://forms.gle/jvjN8A6YkYZNbaCF9>



# Q & A

- What is it like to work in industry?
- What does academic ML look like?
- How do I contribute to open-source?



