Welcome to 15-849: Machine Learning Systems

Zhihao Jia

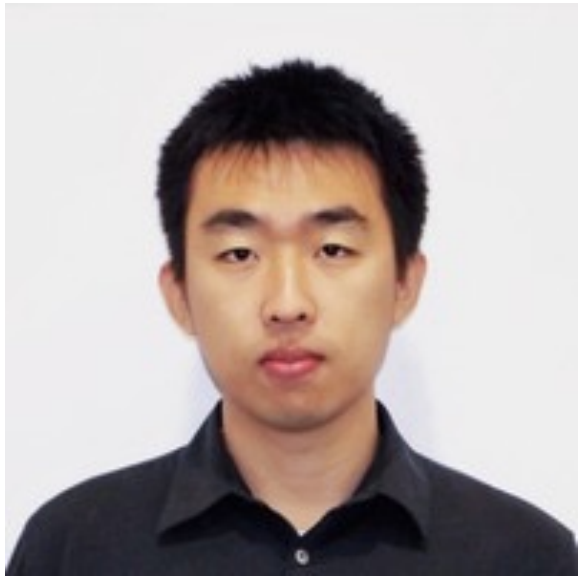
Computer Science Department

Carnegie Mellon University

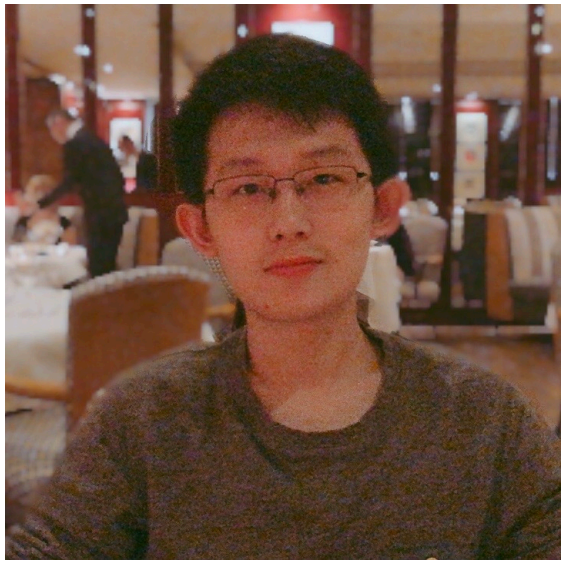
Course Information

- **Website**: https://www.cs.cmu.edu/~zhihaoj2/15-849/
  - Contains links to all resources
- **Piazza**: discussions and announcements
- We will use zoom for the first two weeks of the semester, and resume to in-person meetings in Feb
- **Canvas**: all lecture recordings will be available on Canvas
- **Gradscope**: submit assignments, project proposals, final papers

Instructor



Zhihao Jia
Office hours: upon request



Zhihao Zhang
Office hours: Tue 4-5pm

Giulio Zhou
Office hours: Thu 4-5pm

Zoom links available on the website

Carnegie Mellon University
School of Computer Science

Zoom

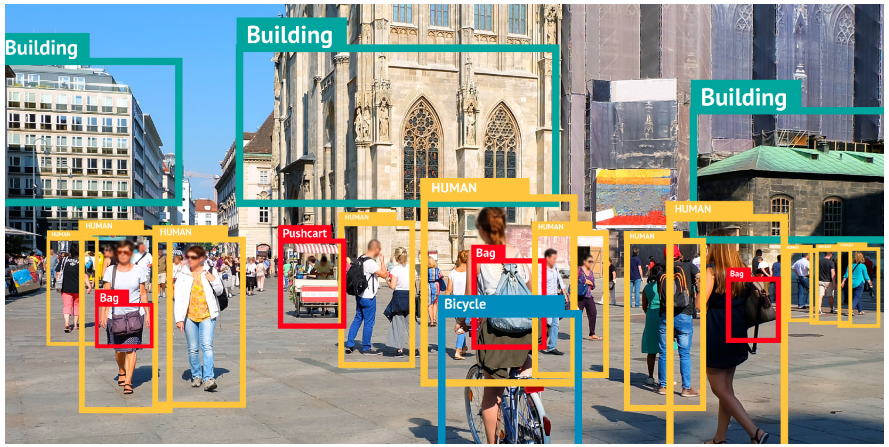
- All lectures will be recorded and posted on Canvas

- Questions and discussions are extremely welcomed during lectures
  - Speak out
  - Use the raise-hand feature
  - Type questions into the chat window

- Keep muted when not talking

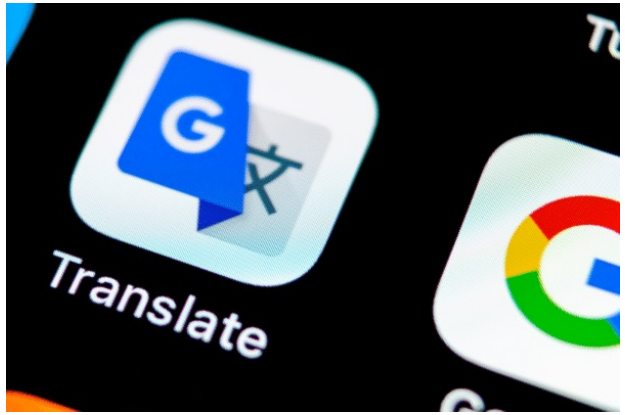
What is this course about?

The Success of Machine Learning Today

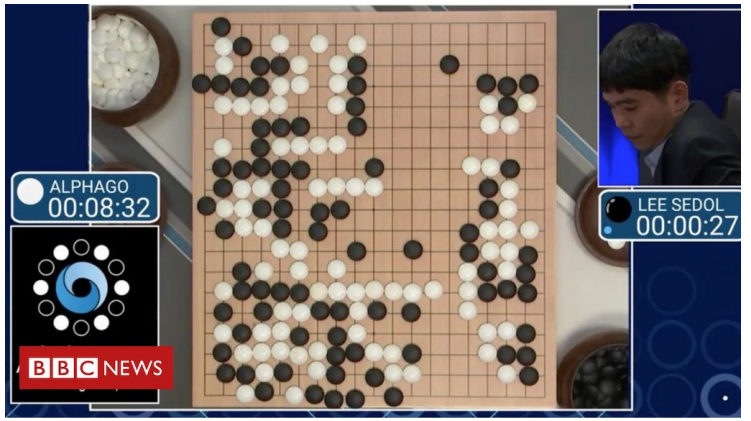


Object detection

Autonomous vehicles

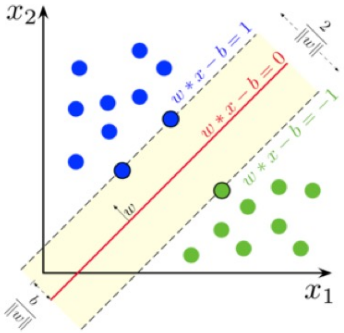


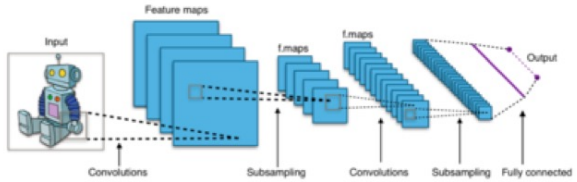
Machine translation

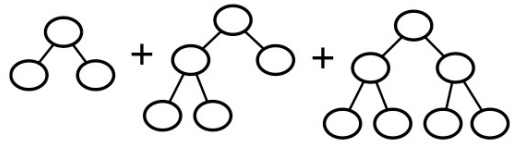


Game playing

Most ML techniques invented in 1980s and 1990s







Perceptron Algorithm 1958

Backprop 1986

Support Vector Machine (SVM) 1992

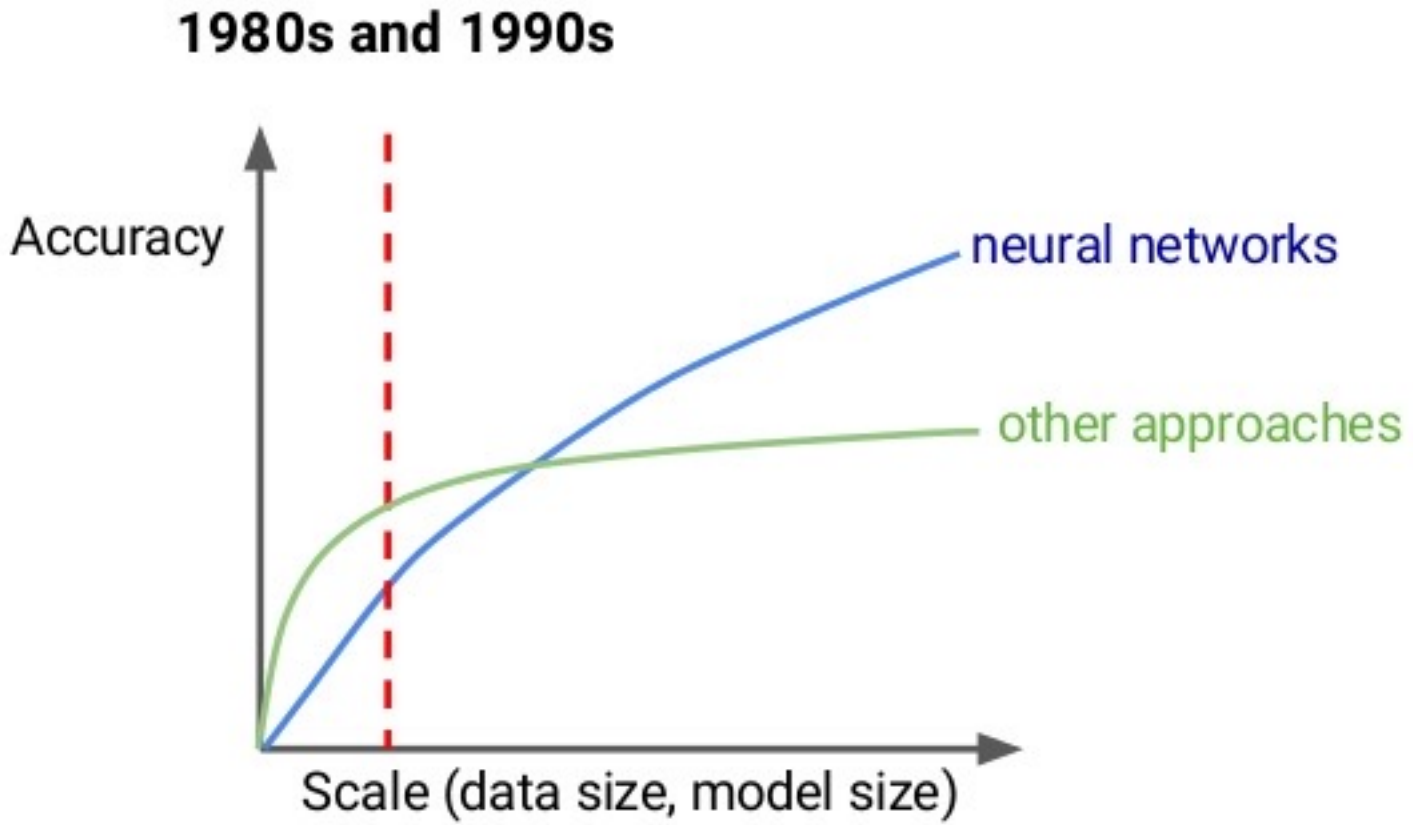
ConvNet 1998

Gradient Boosting Machine (GBM) 1999

Why didn't the success of ML happen in 1990s?

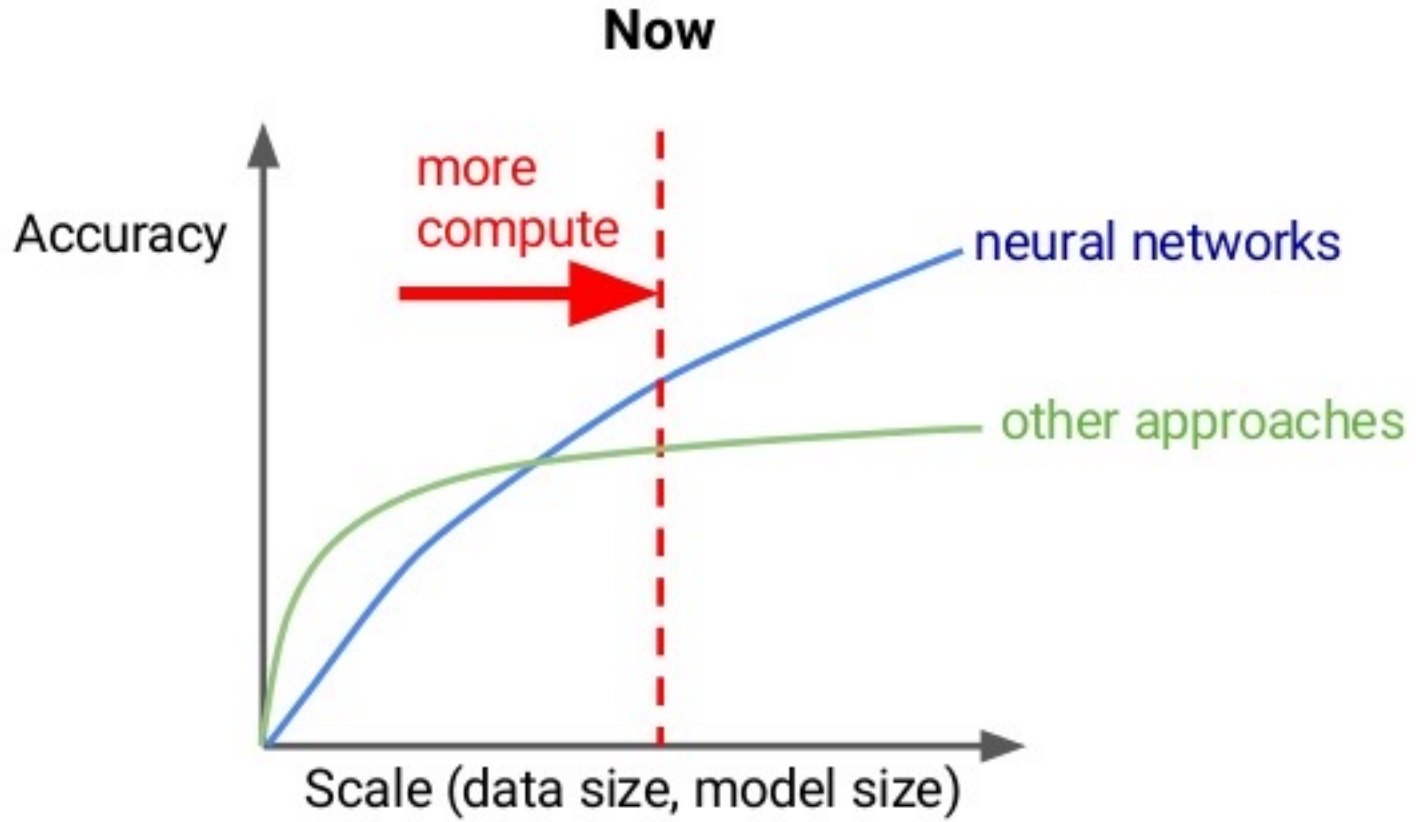
Adapted from TQ's slide

The Rise of ML and Neural Networks



Adapted from Jeff Dean, HotChips 2017

The Rise of ML and Neural Networks

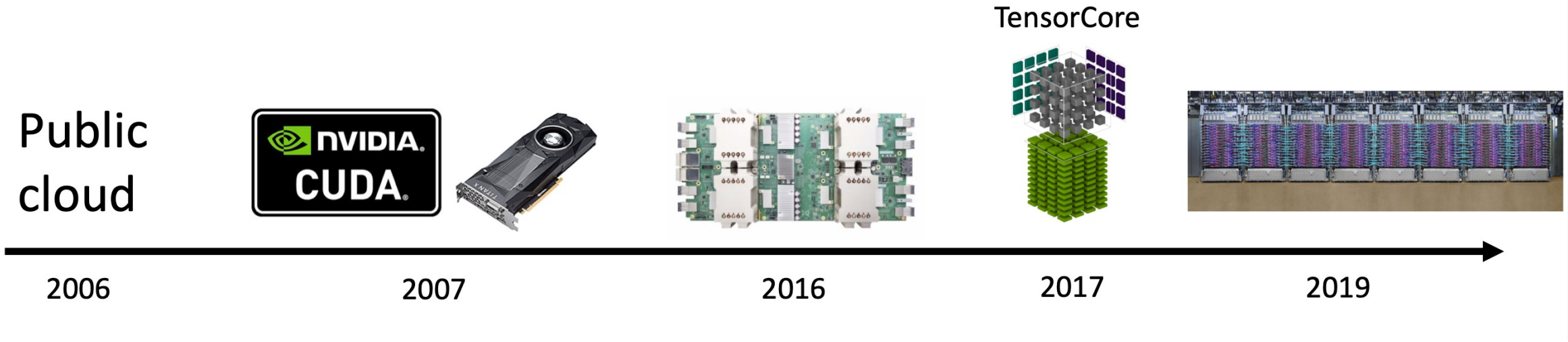


Adapted from Jeff Dean, HotChips 2017

Big data arrives in early 2000

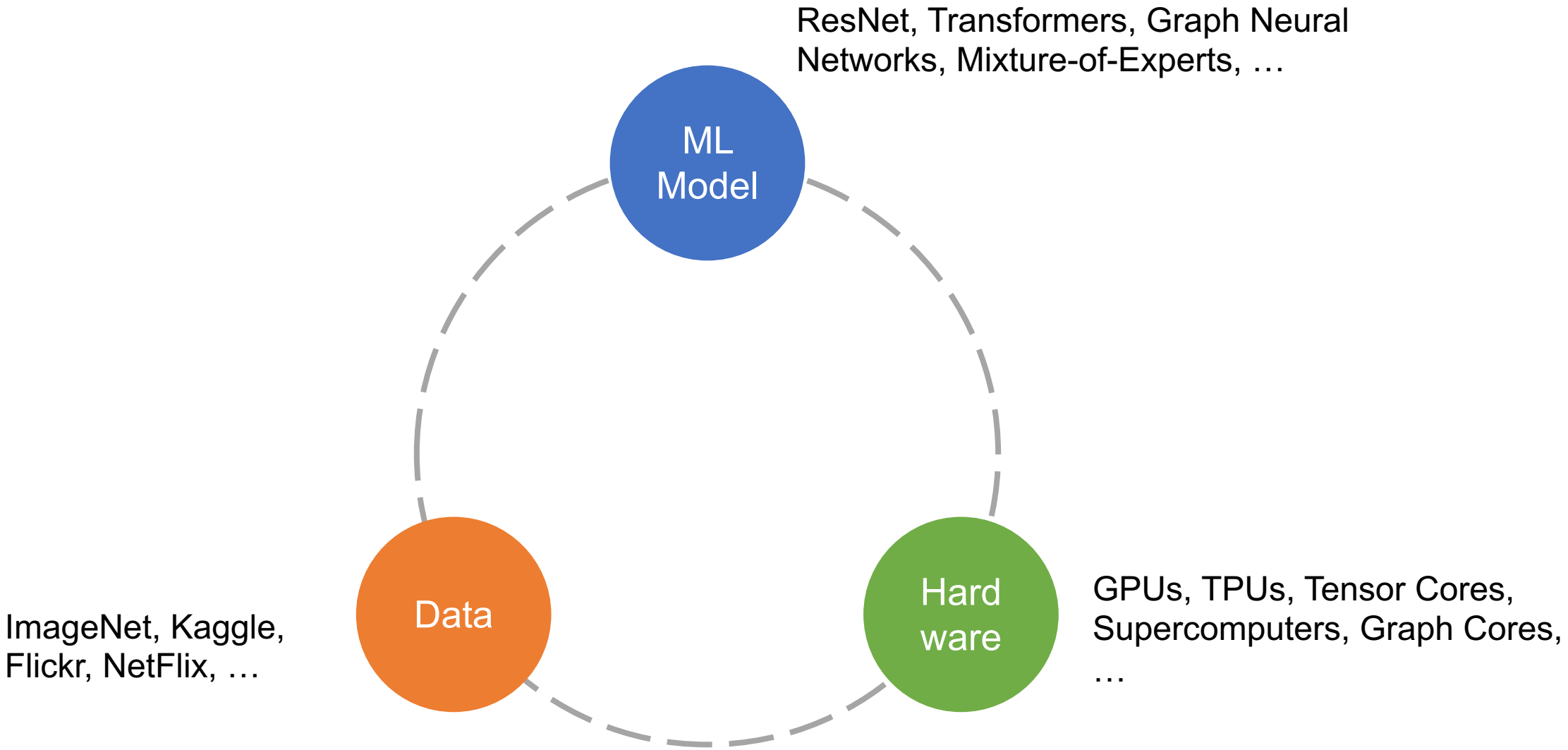
Large-scale training datasets become available

AI hardware becomes widely available in 2010s

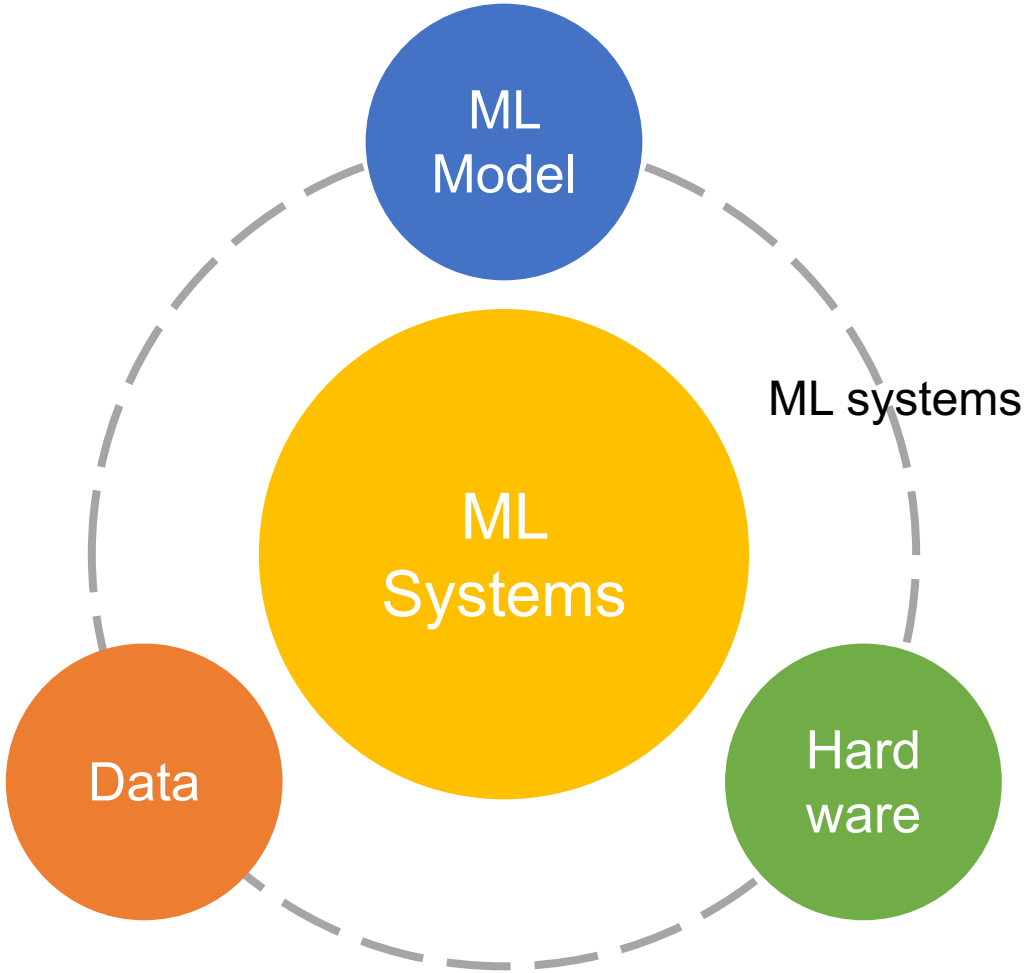


Distributed Heterogeneous Hardware Platforms

The Secret Ingredients in ML Success



Where do ML systems fit into the picture?



ML systems bridge model, data, and hardware

ML Systems are a Key Ingredient in ML

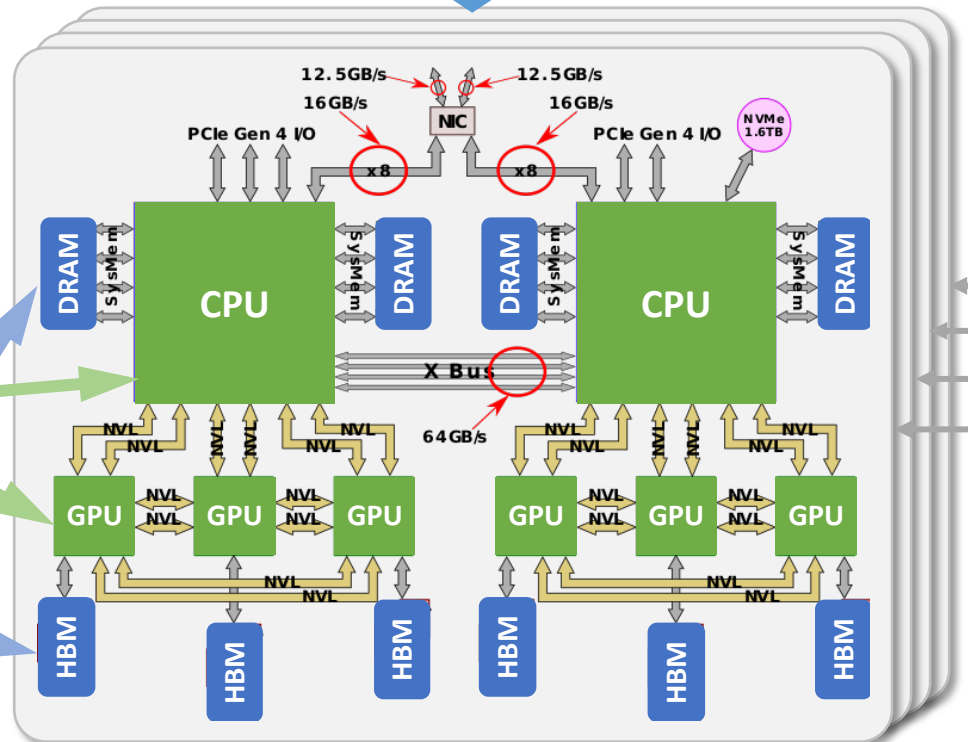
Model

Data

Provide high-level programming interfaces to prototype different ML applications

ML Systems

Effectively deploy ML computations on modern hardware



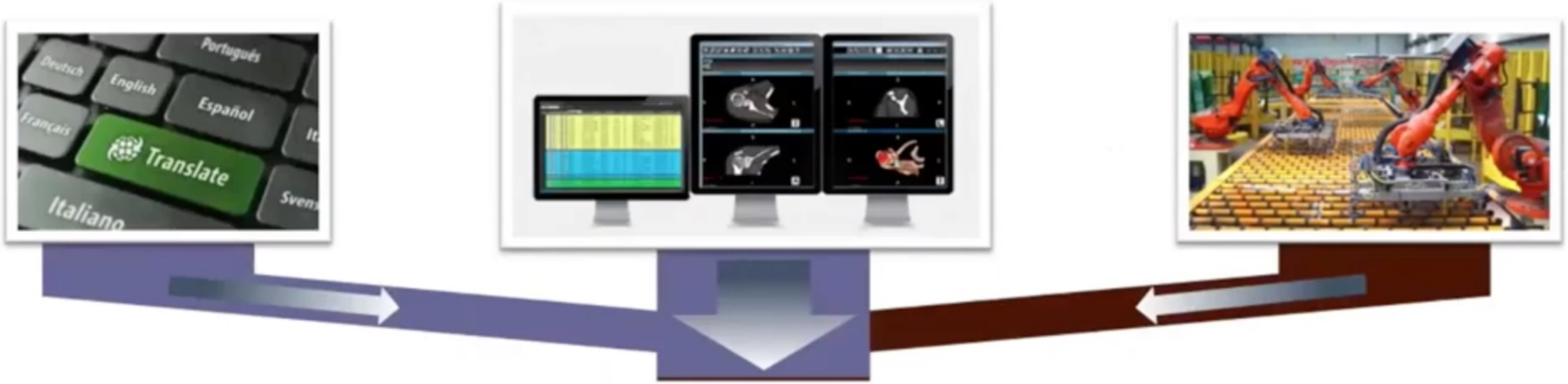
Heterogenous Processors

Heterogenous Memories

High Performance Network

Distributed Heterogenous Hardware Platforms

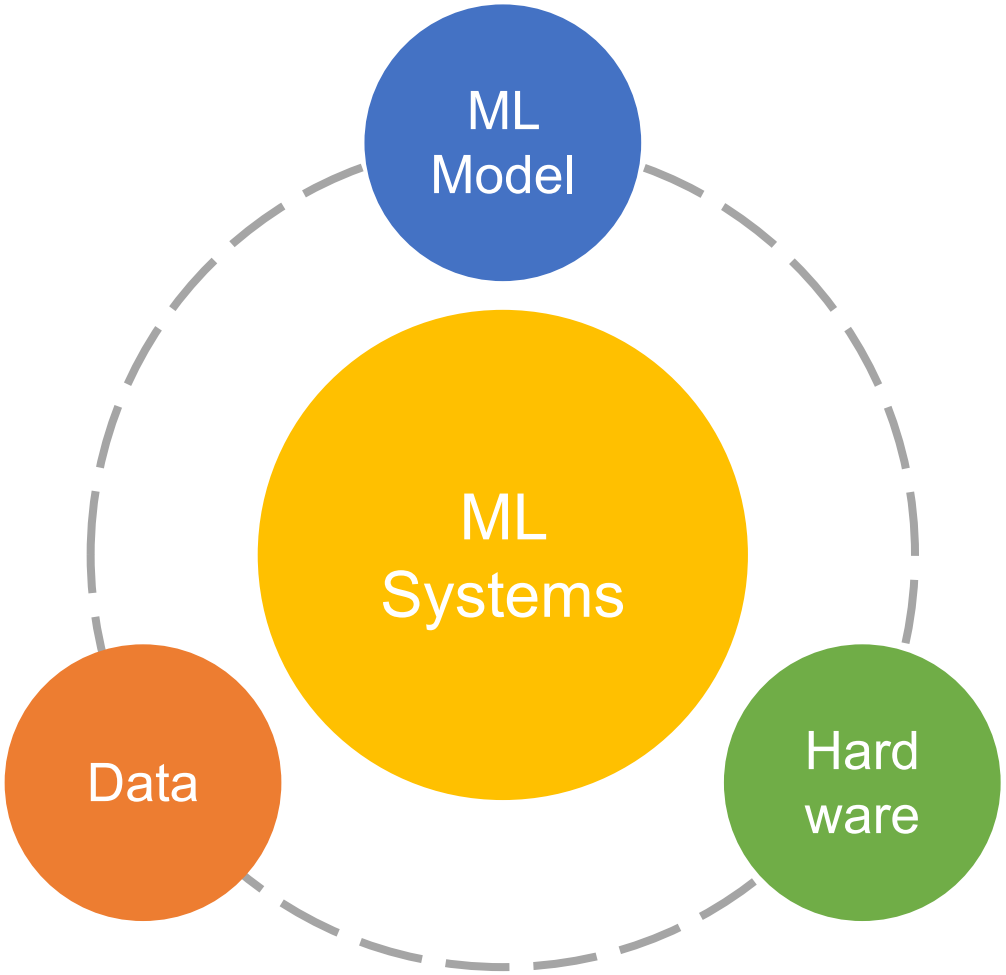
Example: Deep Neural Networks for Machine Translation



1000x Productivity

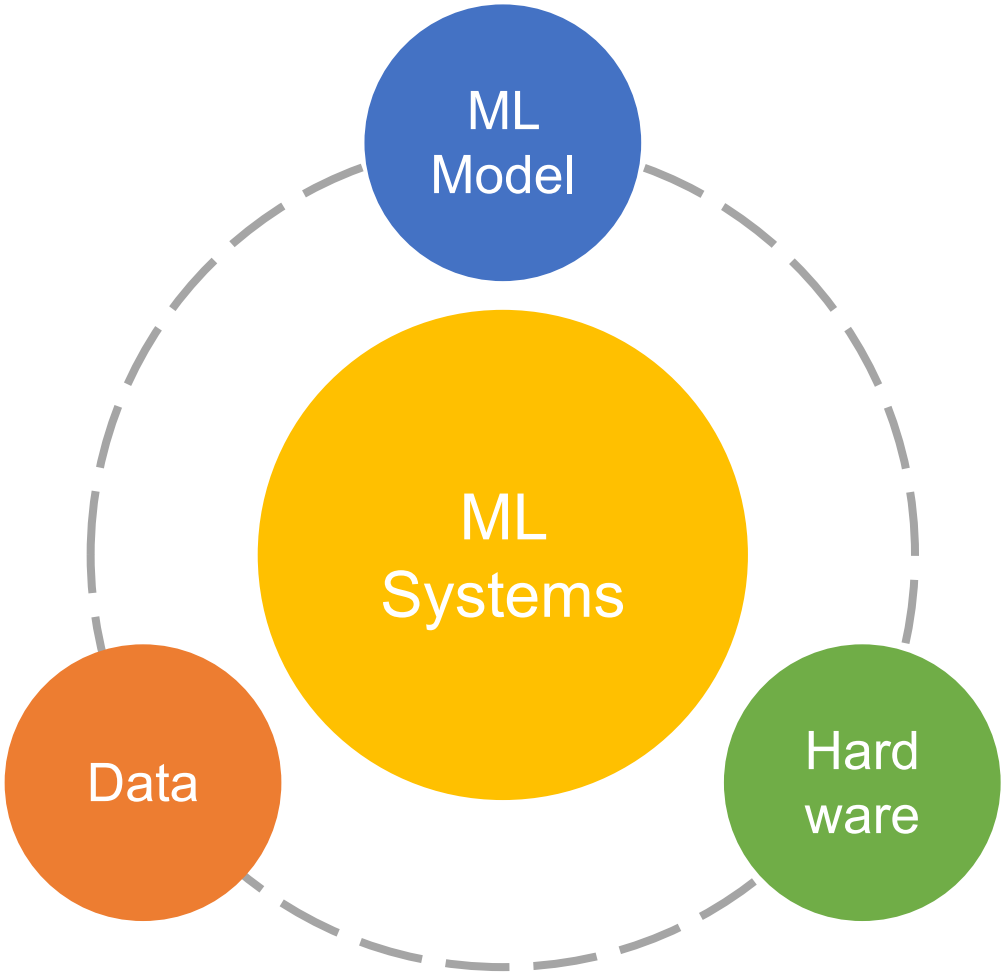
Google shrinks language translation code
from 500k imperative LoC to **500 lines of dataflow**

ML Systems as an Emerging Research Field

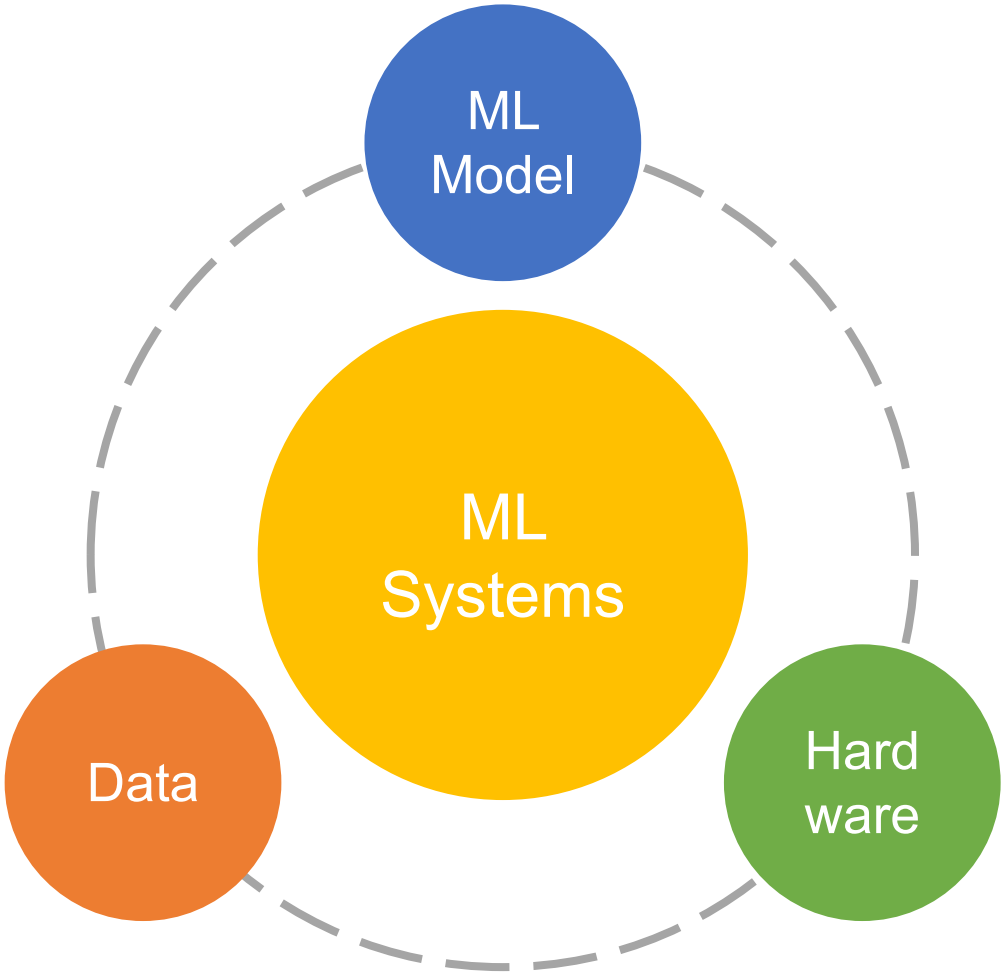


- MLSys papers at major ML and systems venues
- MLSys workshops at these venues
- mlsys.org: a new conference at the intersection of ML and systems

How is MLSys research different from typical ML and systems research?

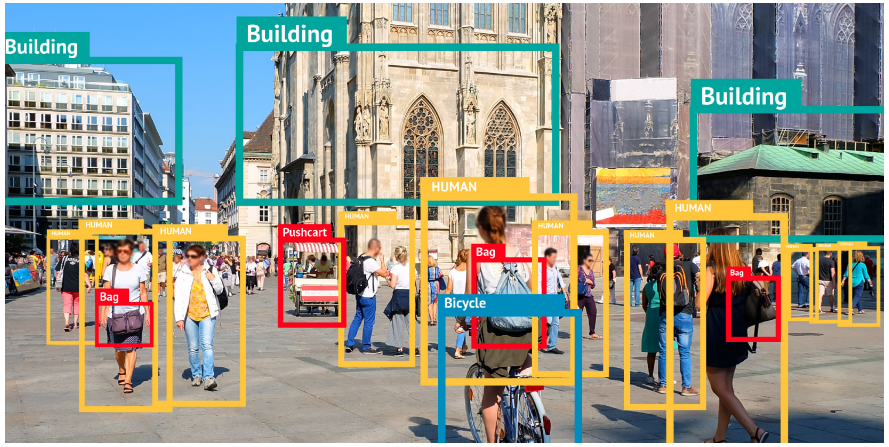


How is MLSys research different from typical ML and systems research?



MLSys provides a holistic approach to combining ML, data, systems, and hardware techniques to solve problems.

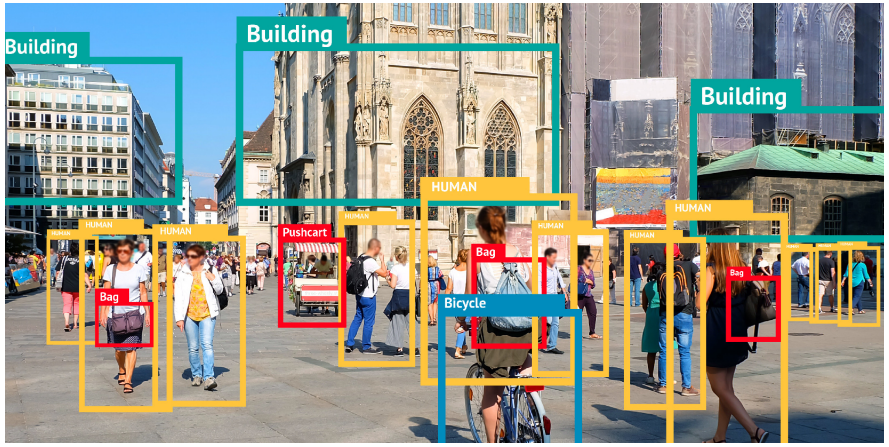
Exercise: Object Detection on surveillance camera



We want to deploy an object detection model on surveillance cameras:

- Accuracy >= 90%
- Latency <= 10ms
- Memory requirement <= 100 MB

A Typical ML Approach



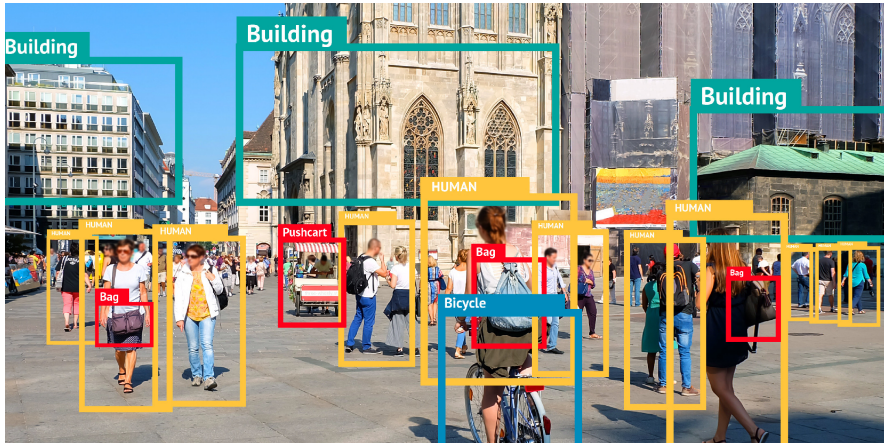
We want to deploy an object detection model on surveillance cameras:

- Accuracy >= 90%
- Latency <= 10ms
- Memory requirement <= 100 MB

Design models with better accuracy and smaller sizes

- Model pruning, quantization, distillation, low-rank approximation, etc..

A Typical Systems Approach

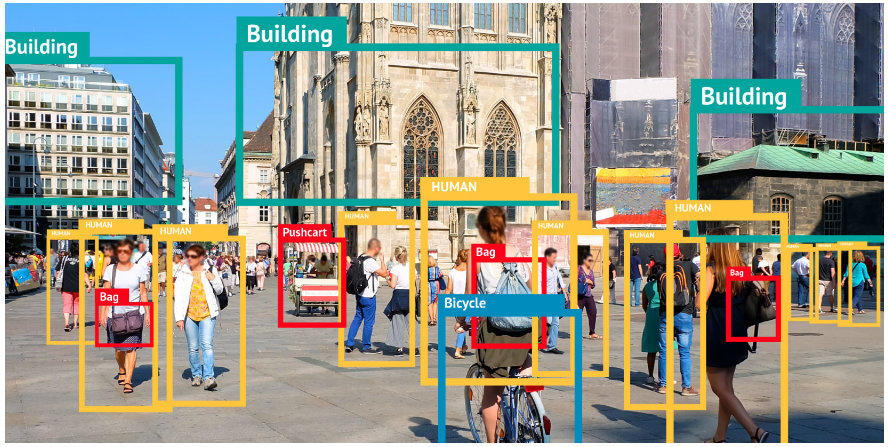


We want to deploy an object detection model on surveillance cameras:

- Accuracy >= 90%
- Latency <= 10ms
- Memory requirement <= 100 MB

Build a fast and memory-efficient inference engine with better resource utilization and runtime performance

An MLSys Approach



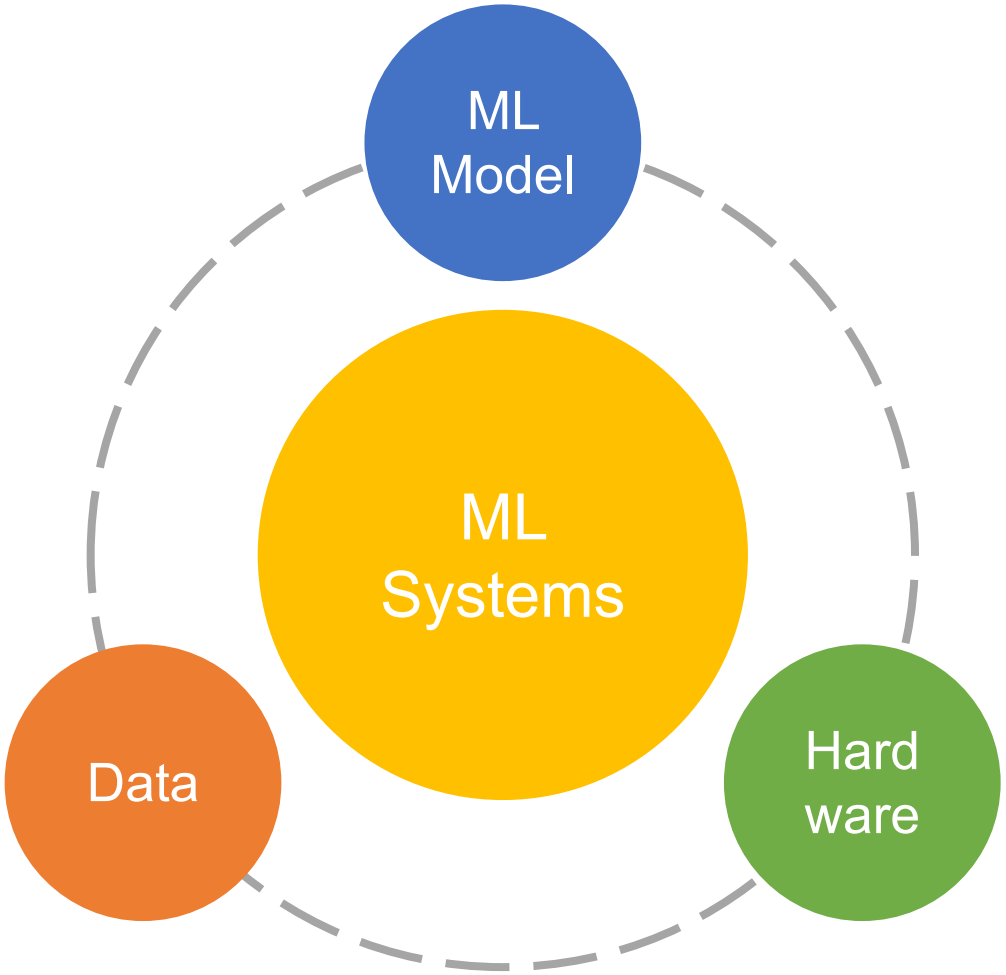
We want to deploy an object detection model on surveillance cameras:

- Accuracy >= 90%
- Latency <= 10ms
- Memory requirement <= 100 MB

Models and systems co-design and co-optimization

- Exploit specialized AI **hardware**
- Develop **models** optimized for the specific hardware
- Build ML **systems** that make use of the above points

How is MLSys research different from typical ML and systems research?

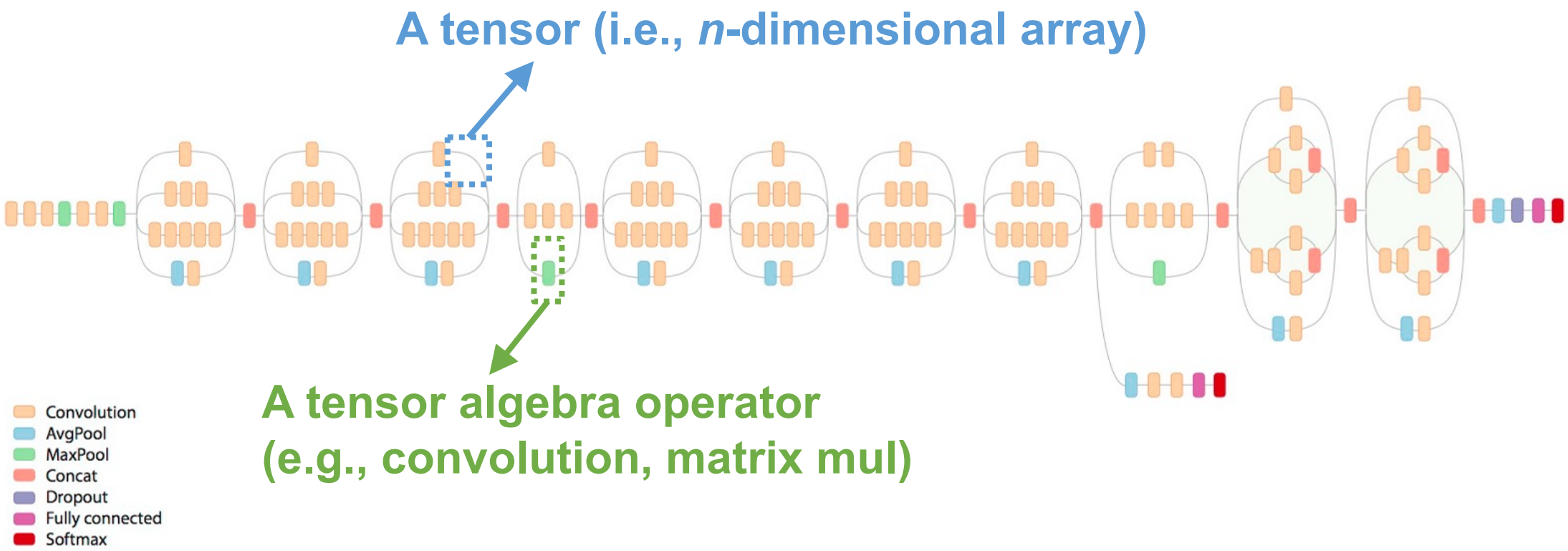


MLSys provides a holistic approach to combining ML, data, systems, and hardware techniques to solve problems.

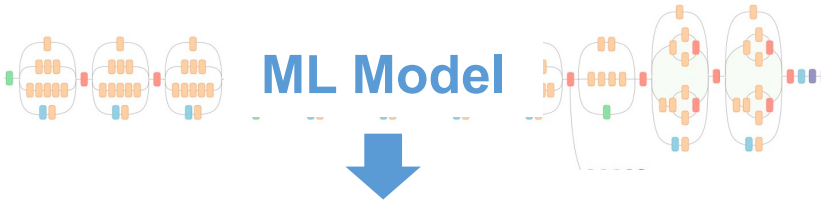
What will this course cover?

- **Systems for Machine Learning**
- **Machine Learning for Systems**
- **Systems and ML Co-design and Co-optimization**

Computation Graph of ML Model



Systems for Machine Learning



Automatic Differentiation

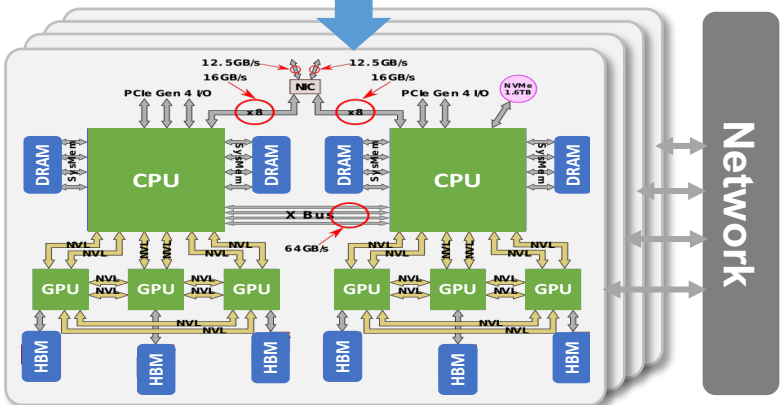
Graph-Level Optimization

Parallelization / Distributed Training

Data Layout and Placement

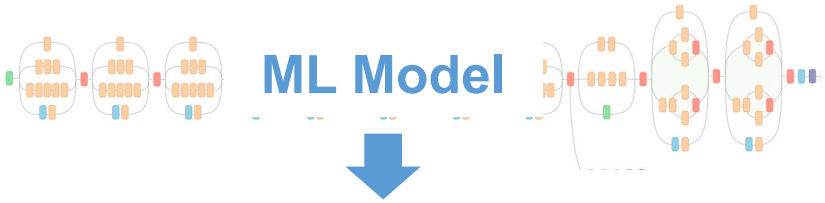
Kernel Optimizations

Memory Optimizations



We will learn the current design and key techniques of each stack in ML systems

Systems for Machine Learning



Automatic Differentiation

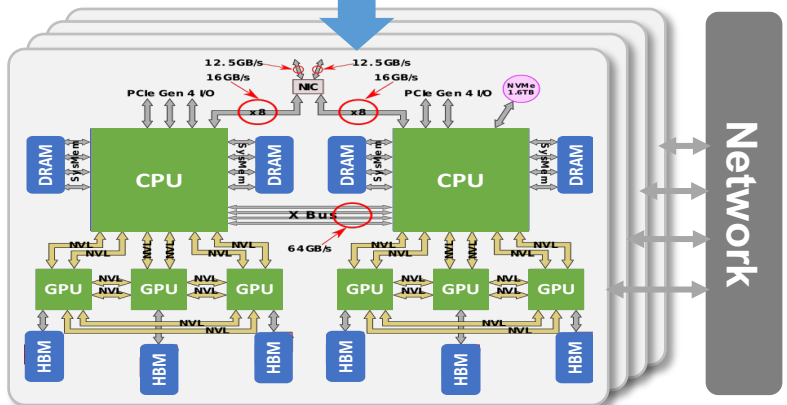
Graph-Level Optimization

Parallelization / Distributed Training

Data Layout and Placement

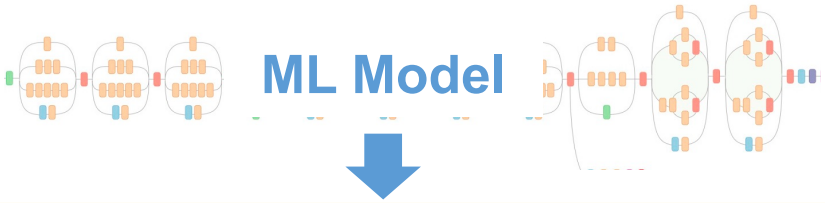
Kernel Optimizations

Memory Optimizations



Automatically construct backpropagation graph and compute gradients for model training

Systems for Machine Learning



Automatic Differentiation

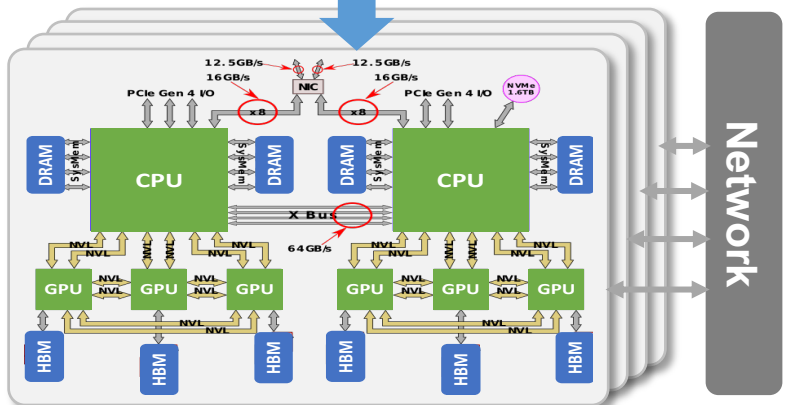
Graph-Level Optimization

Parallelization / Distributed Training

Data Layout and Placement

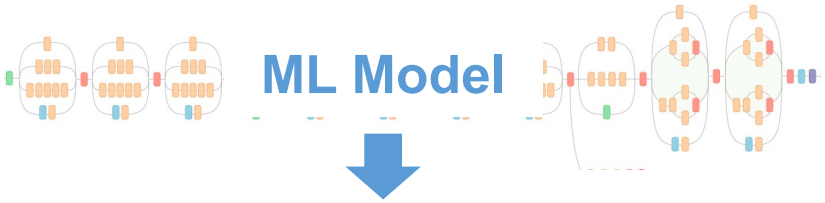
Kernel Optimizations

Memory Optimizations



Optimize the computation graphs of ML models;
Apply possible arithmetic transformations

Systems for Machine Learning



Automatic Differentiation

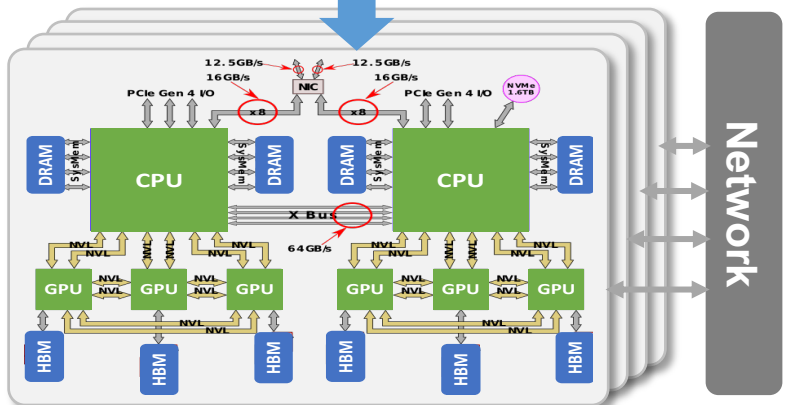
Graph-Level Optimization

Parallelization / Distributed Training

Data Layout and Placement

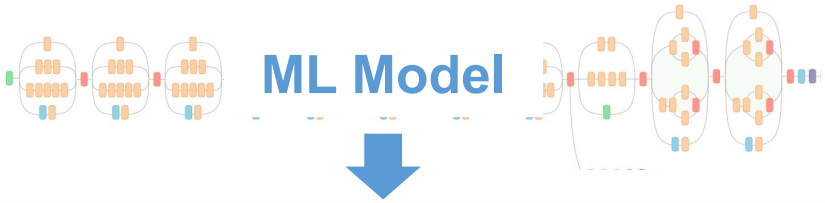
Kernel Optimizations

Memory Optimizations



Decide how to best parallelize ML computation across distributed heterogenous machines

Systems for Machine Learning



Automatic Differentiation

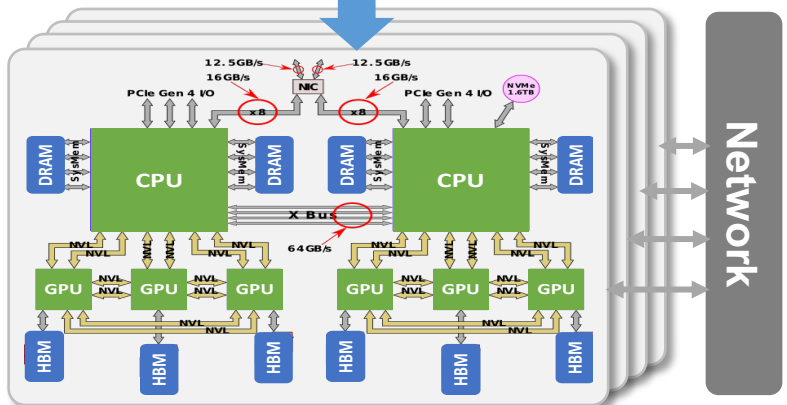
Graph-Level Optimization

Parallelization / Distributed Training

Data Layout and Placement

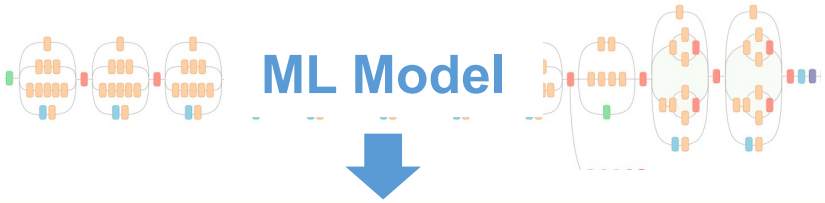
Kernel Optimizations

Memory Optimizations



Where to place the intermediate tensors in the memory hierarchy?
Which data layouts should we use?

Systems for Machine Learning



Automatic Differentiation

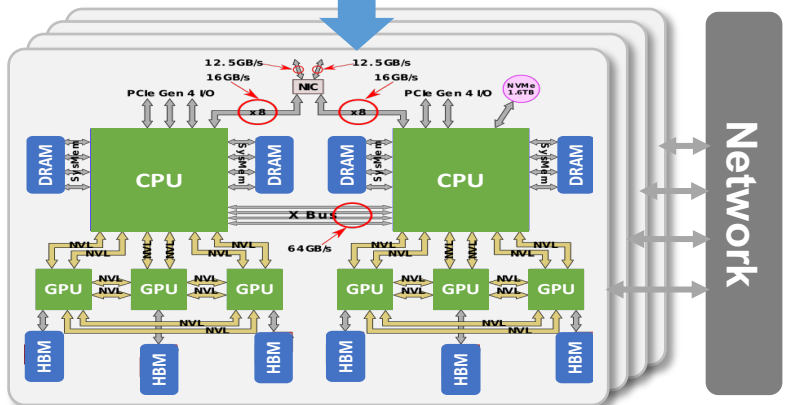
Graph-Level Optimization

Parallelization / Distributed Training

Data Layout and Placement

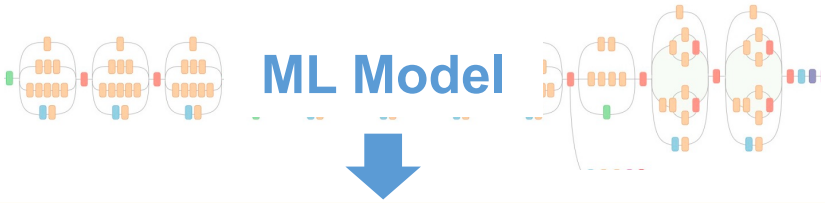
Kernel Optimizations

Memory Optimizations



Generate high performance kernels and executables for different hardware backends

Systems for Machine Learning



Automatic Differentiation

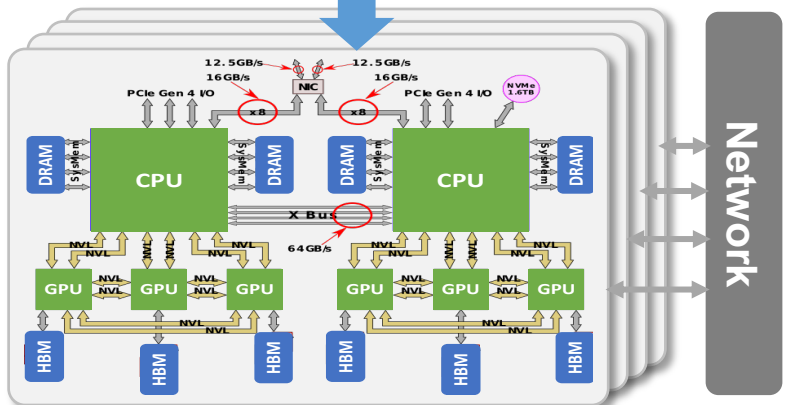
Graph-Level Optimization

Parallelization / Distributed Training

Data Layout and Placement

Kernel Optimizations

Memory Optimizations



Minimize memory requirements for ML computations on AI hardware

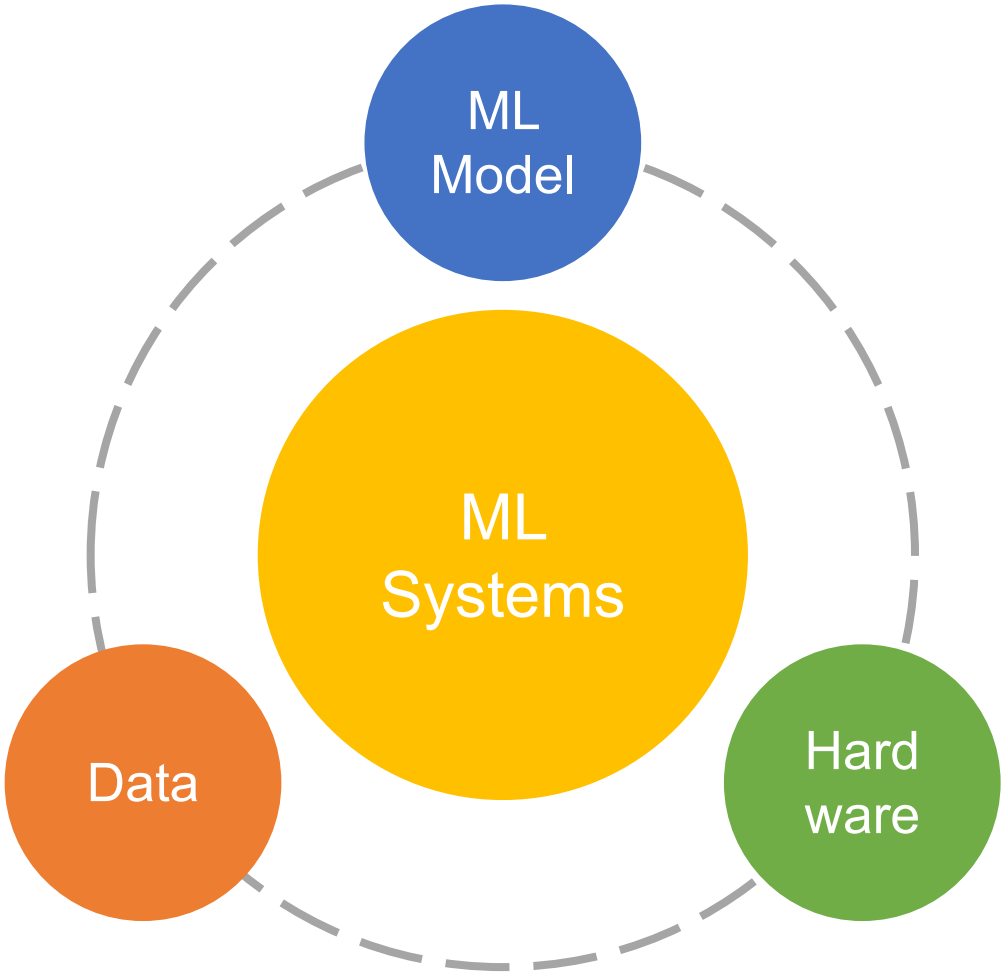
Machine Learning for Systems

- Automatic tensor program optimizations
- Learnt device placement
- Learnt data structures
- Learnt compiler optimizations

ML and Systems Co-Design and Co-Optimizations

- Hardware-specific neural architecture search
- Leverage emerging hardware infrastructures for ML
  - Use serverless computing for ML
  - Train ML models on spot virtual machines

What will you learn?



A holistic view and approach to combining ML, data, systems, and hardware techniques to solve MLSys problems

Lectures

- Overview lectures of areas in machine learning and systems
  - Learn basic concepts of ML systems
- Paper reading, presentation, and discussion
  - Learn from recent MLSys works
  - Understand the layers in ML systems and how they interact
- Write paper reviews
  - Critical thinking
  - Understand their strength and limitations
  - Learn and generalize ideas
- Final project (groups of 1-3 students)
  - Build your own ML systems

Paper Reading (Starting from Week 3) How to Read a Paper

In each lecture, we will discuss two MLSys papers

Read these papers before the class and write a review

- Review details in the next slide

Keep in mind:

- What problem does this paper try to solve?
- Why is this an important and hard problem?
- Why can't previous work solve this problem?
- What is novel in this paper?
- Does it show good results?

Paper Review (due before each class)

- One short paragraph summarizing the first paper, in your own words
- One short paragraph summarizing the second paper, in your own words
- One short paragraph on any connections between the papers, such as
  - Compare and contrast: how one work is better than the other
  - Apply the ideas from one paper to solve the problem in the other
  - A new idea that can incorporate results from both papers

Paper Presentation

- You will present one paper in the paper discussion class once a semester
- 15 mins presentation + 5 mins QA
- Presenters submit slides to **gradescope** before the class

What should be covered in the presentation?

- **Summary** slides for high-level ideas
- **Problems**: what problem does this paper solve?
- **Challenges**: why is this problem hard to solve?
- **Methods**: what are the key techniques in the paper?

Paper Discussion

10 mins group discussion + 10 mins class discussion

Group discussion: students will be randomly divided into groups of three-four people. Discuss the following points with your group members:

- What are the main contributions of the paper?
- What are the limitations of the proposed method?
- How the method can be improved?
- Your own thoughts

Class discussion: a member from each group summarizes the discussion

Signup for Paper Presentations

- Each student will present at least once.

- Paper presentations will start on week 3 (systems for deep learning)
- Sign-up link will be posted to Piazza after the class
  - Please list at least five topics you are interested in presenting

Final Course Project

- Team of 1-3 students (sign up in week 5), find your teammates early
- We will provide a list of potential project ideas. You are more encouraged to bring your own MLSys topics and ideas

Milestones:

- 1-page proposal
- Informal mid-term check-in with TAs
- Final presentation
- Paper writeup

Grading

- Course project: 50%
- Paper review: 20%
- Paper presentation: 20%
- Class participation: 10% + 5%

- All reviews and reports are submitted on Gradscope
- Ask questions and discuss on Piazza

Always refer to the website for more info:
https://www.cs.cmu.edu/~zhihaoj2/15-849/

Stay safe and have a great semester!