CMSC818Q: Special Topics in Cloud Computing

Introduction

Instructor: Alan Liu

DEPARTMENT OF COMPUTER SCIENCE

Class Information

- Website: https://zaoxing.github.io/teaching/2026-cloud-network
  - Bookmark this, contains links all resources
- ELMS-Canvas: discussions and announcements
- Email: always happy to chat

Instructor / Your Collaborator

Alan Liu

Office hours:
upon request

Prof.

DEPARTMENT OF
COMPUTER SCIENCE

UMIACS
University of Maryland Institute for Advanced Computer Studies

Research

SYSTEMS&
NETWORKING

I work with

Akamai
intel
amazon

Foodie and …

Welcome: What is this class about

Three Goals (How to do systems research):

- Learning latest research in Systems for AI domain:
  Reading, Reviewing, Presenting, Reproducing

- Finding an interesting problem to explore, how?

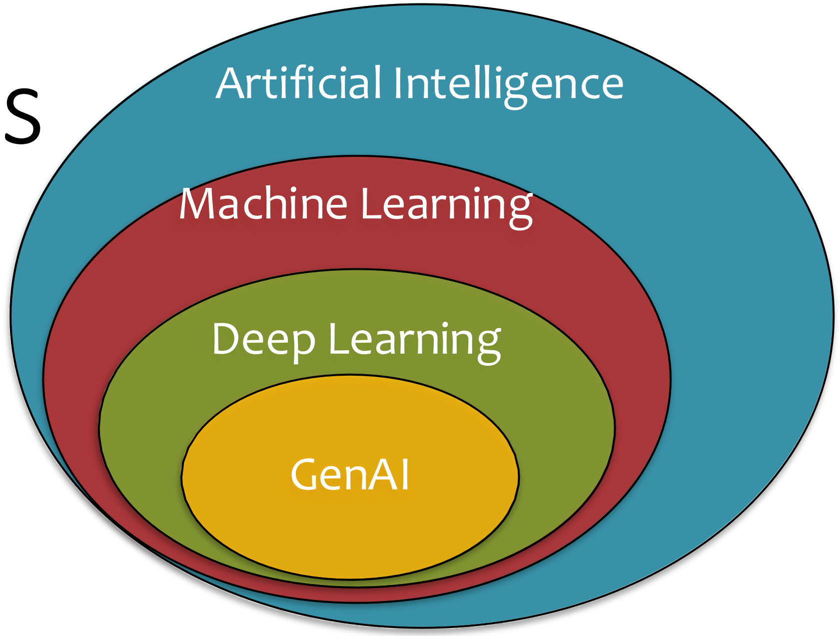
- Playing a role in an open-source project.

Two main themes this semester:

- How a LLM is trained and served in the cloud infra?

- How (Shall) we build efficient systems in the AI wave?
E.g., How AI agent systems work?

Artificial Intelligence Workloads



The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

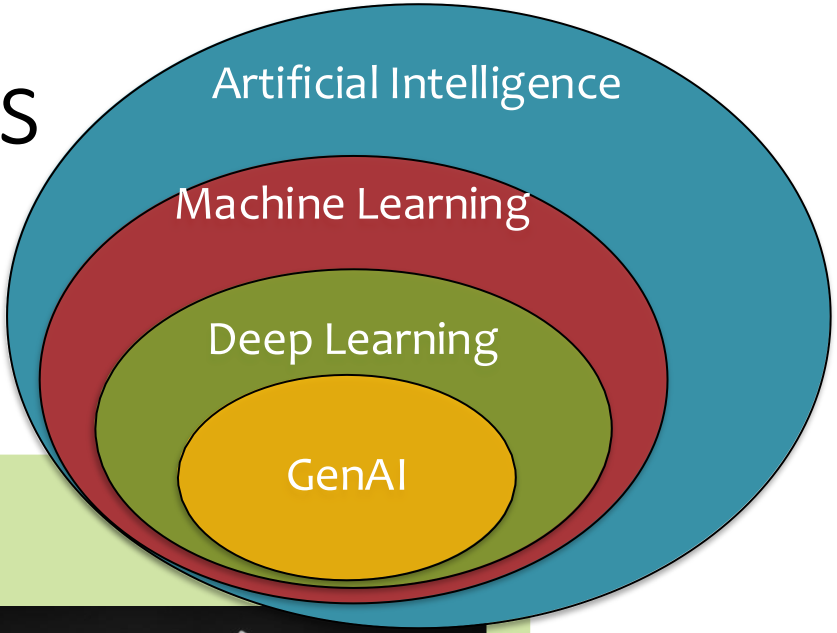
- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

Artificial Intelligence Workloads

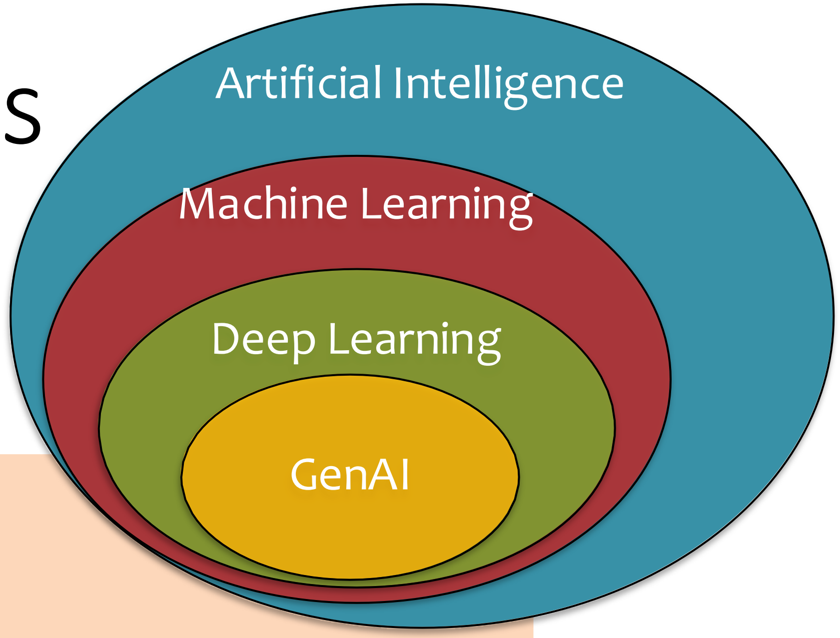
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning



Artificial Intelligence Workloads

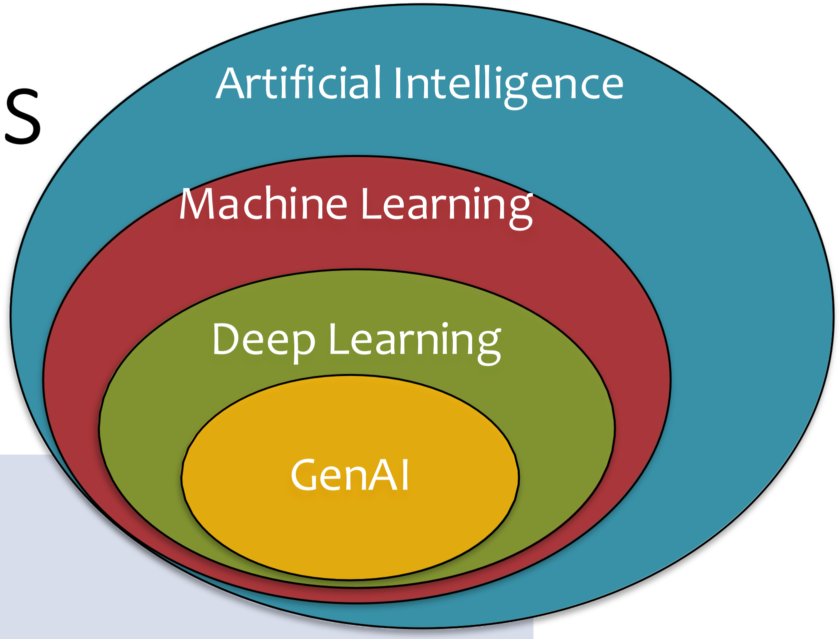


The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

Artificial Intelligence Workloads

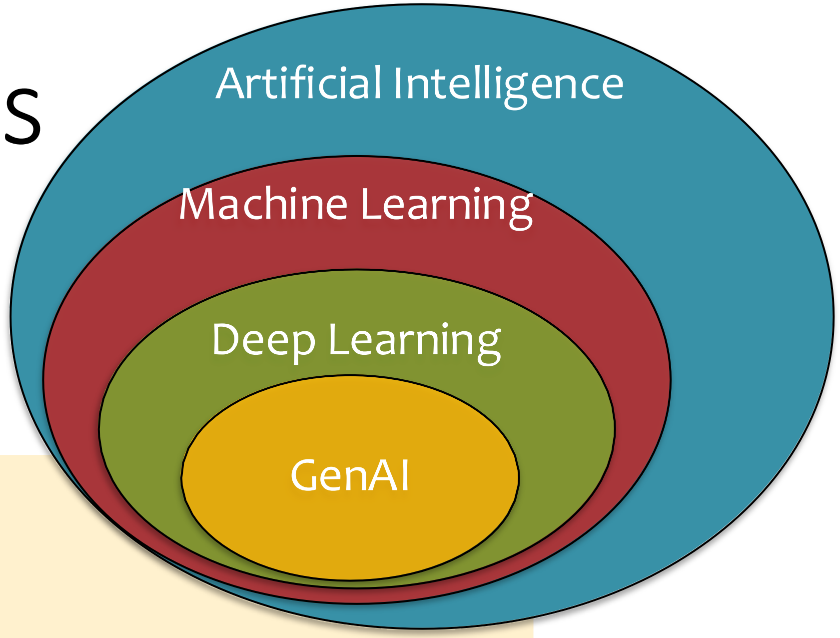


The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

Artificial Intelligence Workloads

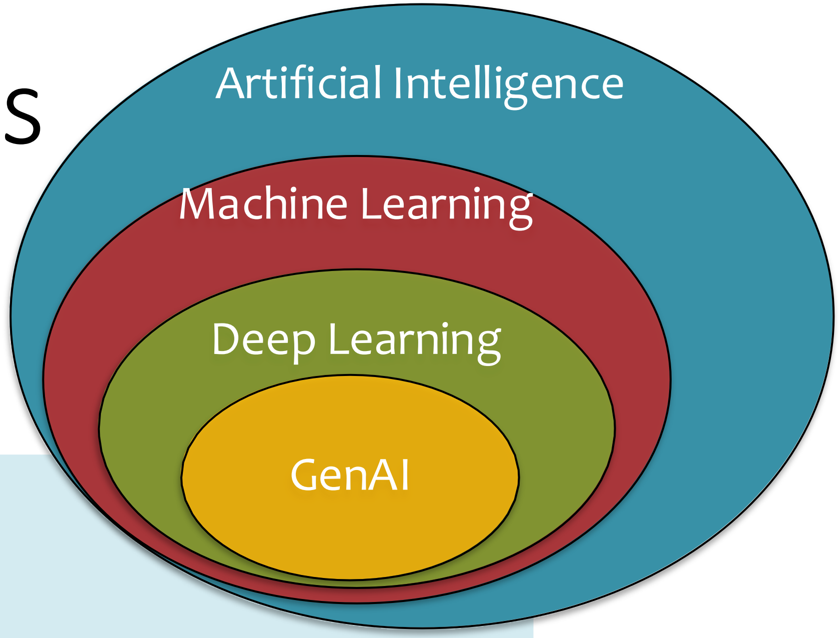


The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

Artificial Intelligence Workloads

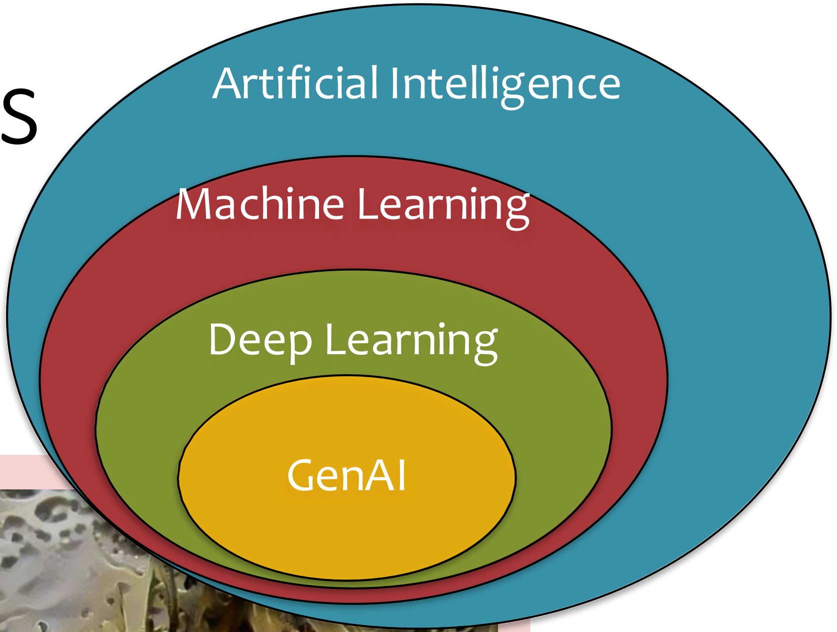


The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

Artificial Intelligence Workloads



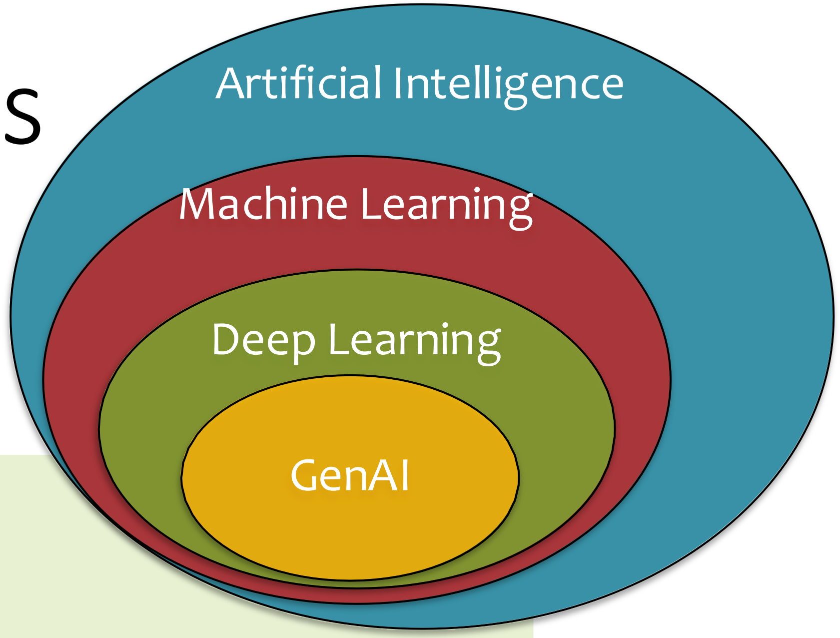
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

"Deep Style" from https://deepdreamgenerator.com/#gallery

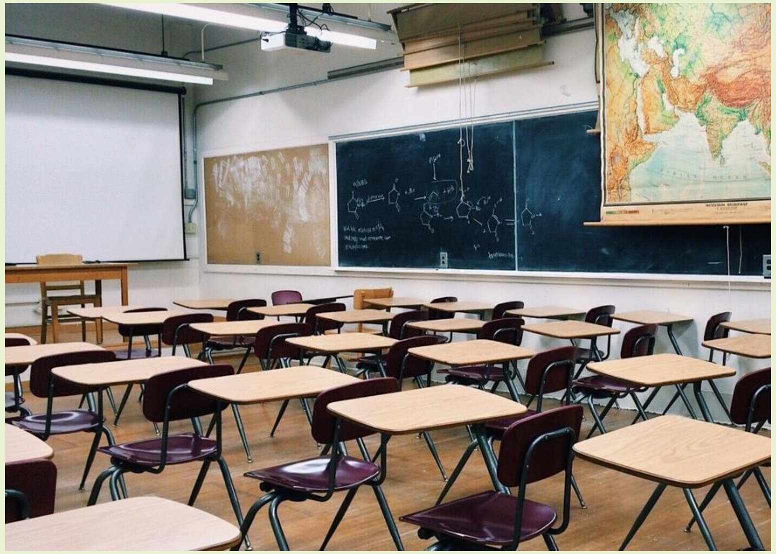
Artificial Intelligence Workloads



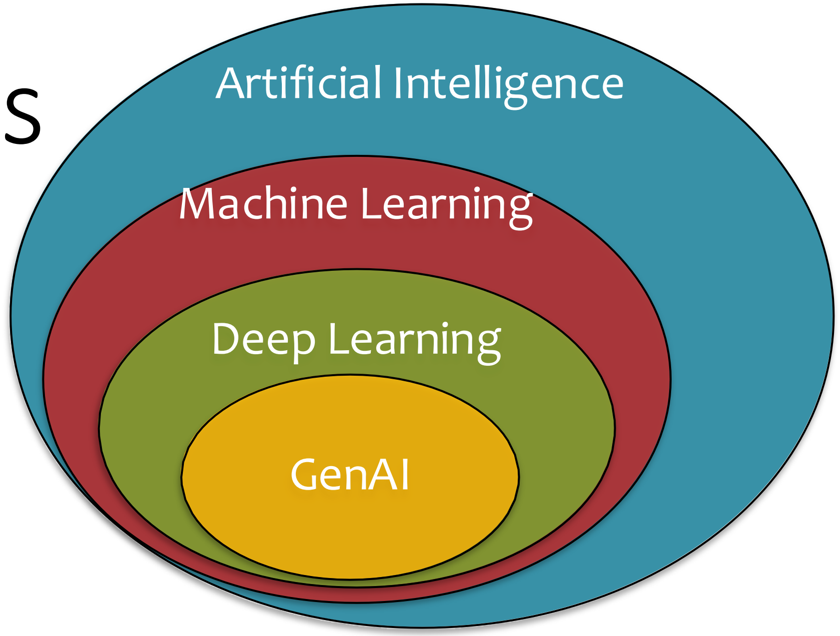
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning



Artificial Intelligence Workloads



The basic goal of AI is to develop intelligent machines.

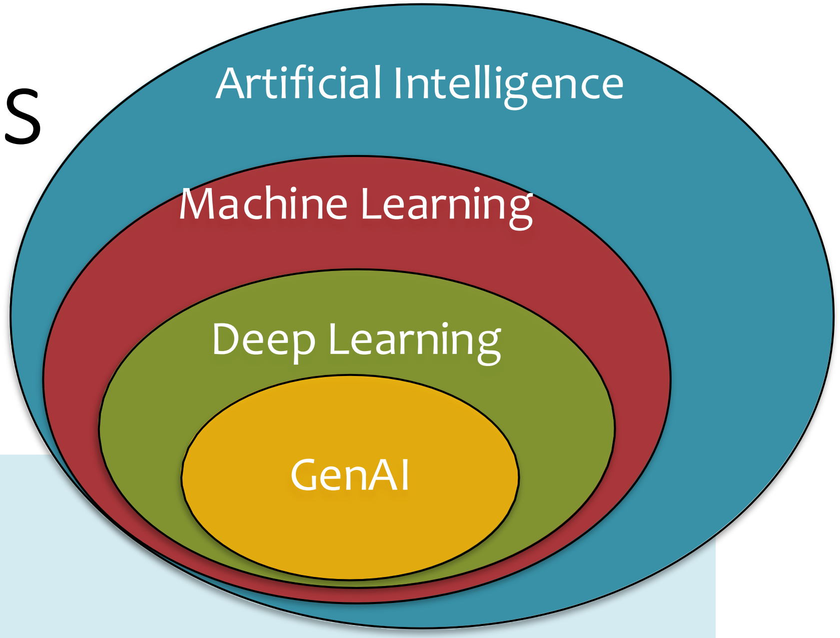
This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

OQ: What does Generative AI have to do with **any of these goals**?

OA: It's making in-roads into **all of them.**

Artificial Intelligence Workloads



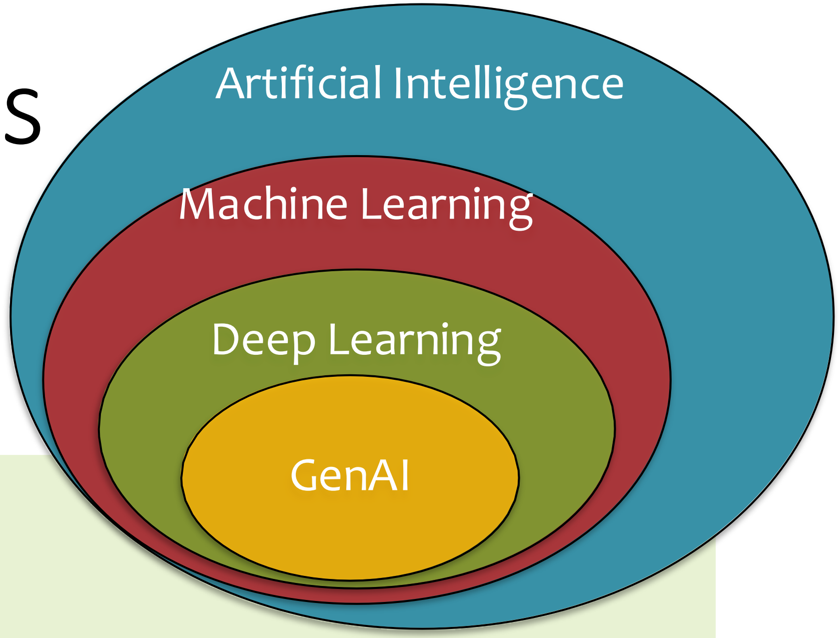
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

- ☐ Communication comprises the comprehension and generation of human language.
- ☐ Large language models (LLMs) excel at both
- ☐ (Even though they are most often trained autoregressively, i.e. to **generate** a next word, given the previous ones)

Artificial Intelligence Workloads



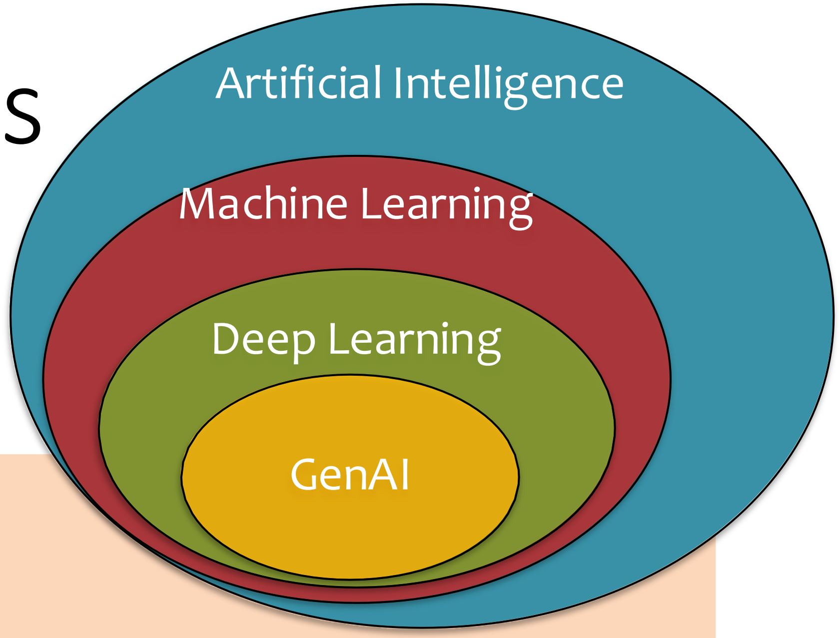
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

- The traditional way of learning in ML is via **parameter estimation**
- But **in-context learning** (i.e. providing training examples as context at test time) shows that learning can also be done via **inference**

Artificial Intelligence Workloads



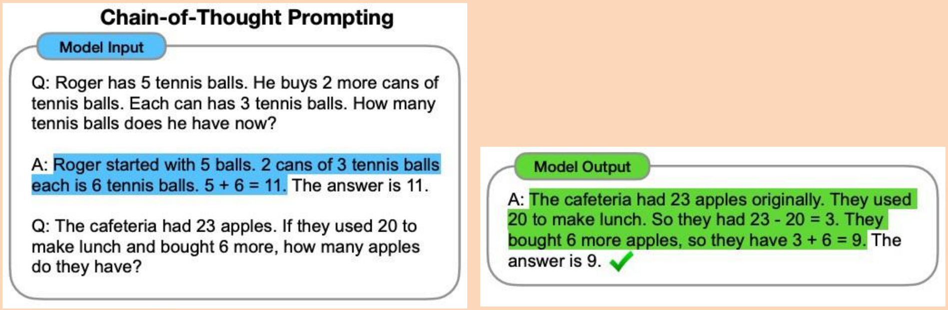
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

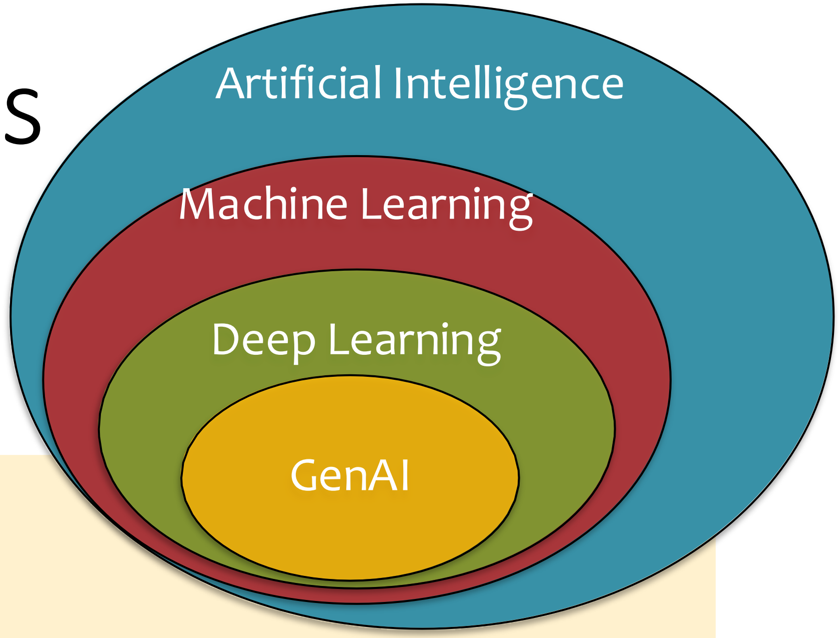
- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

□ LLMs are also (unexpectedly) good at certain reasoning tasks

□ cf. Chain-of-Though Prompting (an ex. of in-context learning)



Artificial Intelligence Workloads



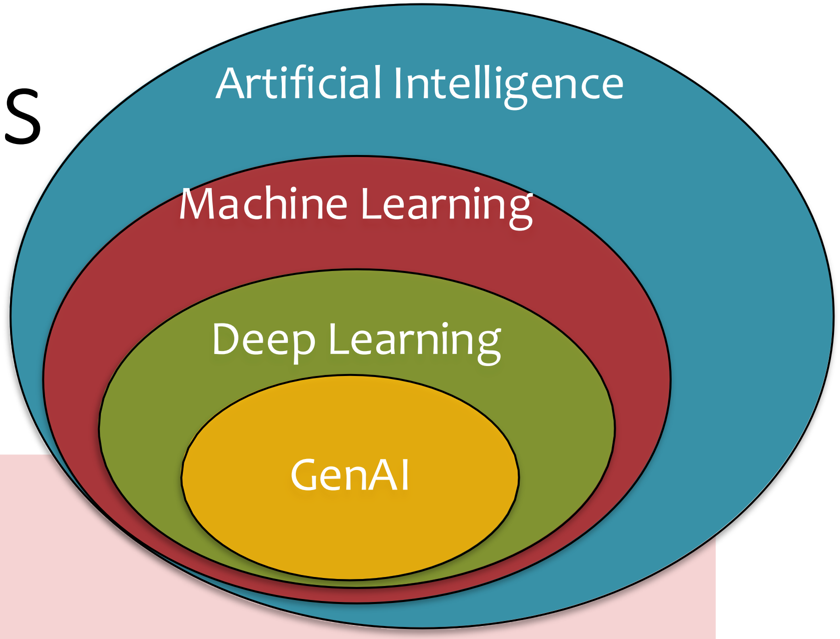
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

- ☐ LLMs are already being used for grounded planning for embodied agents, c.f. LLM-Planner
- ☐ Planning is a key step for agentic code assistants

Artificial Intelligence Workloads



The basic goal of AI is to develop intelligent machines.

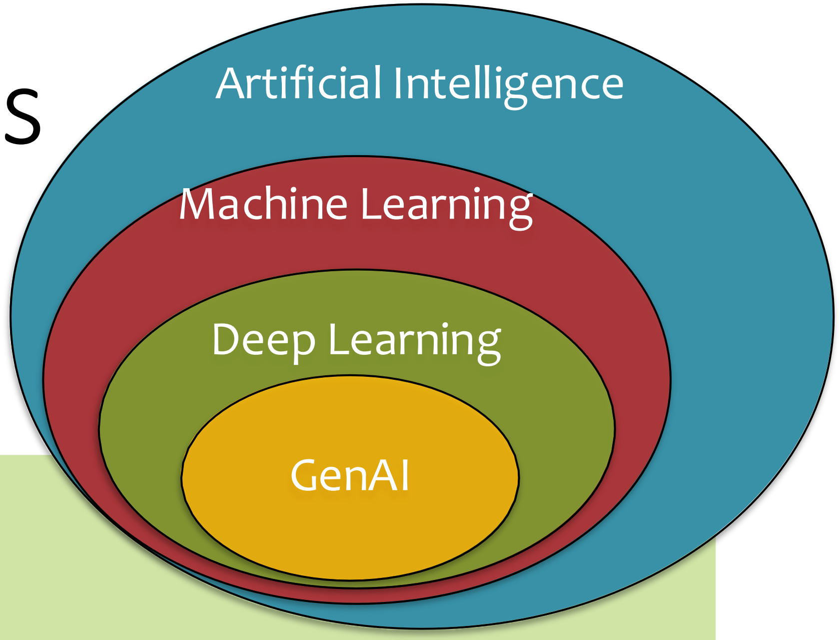
This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

- ☐ Text-to-image models [Midjourney's Discord server has 18 million members (1.7 million were online this morning)]
- ☐ Text-to-music models [MusicGen capable of conditioning on text and audio sample]

"Deep Style" from https://deepdreamgenerator.com/#gallery

Artificial Intelligence Workloads



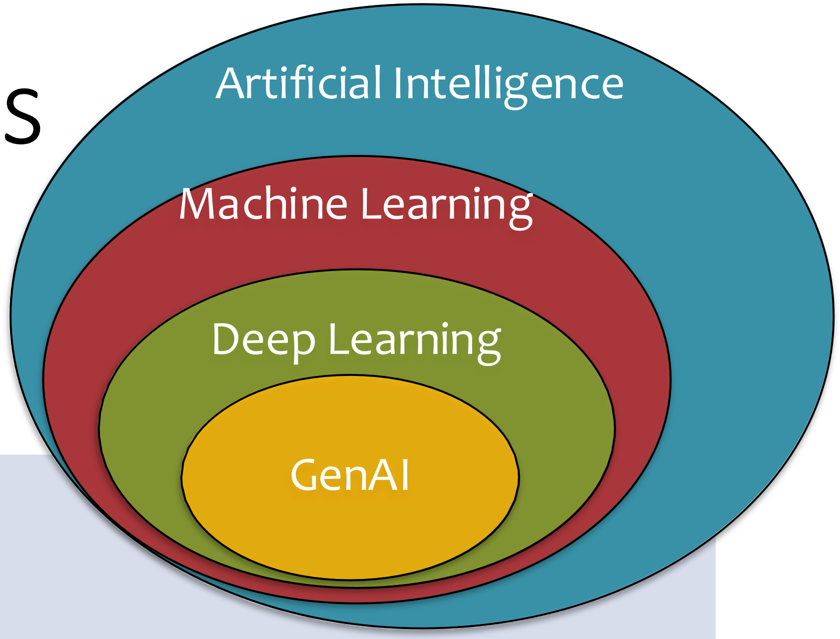
The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

- □ Multimodal foundation models learn to answer questions about images (and text in images)
- □ Diffusion models can be used as zero-shot classifiers

Artificial Intelligence Workloads

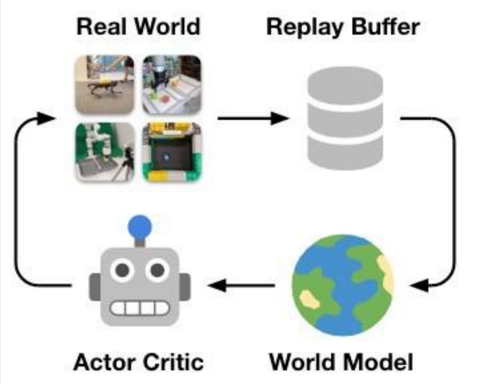


The basic goal of AI is to develop intelligent machines.

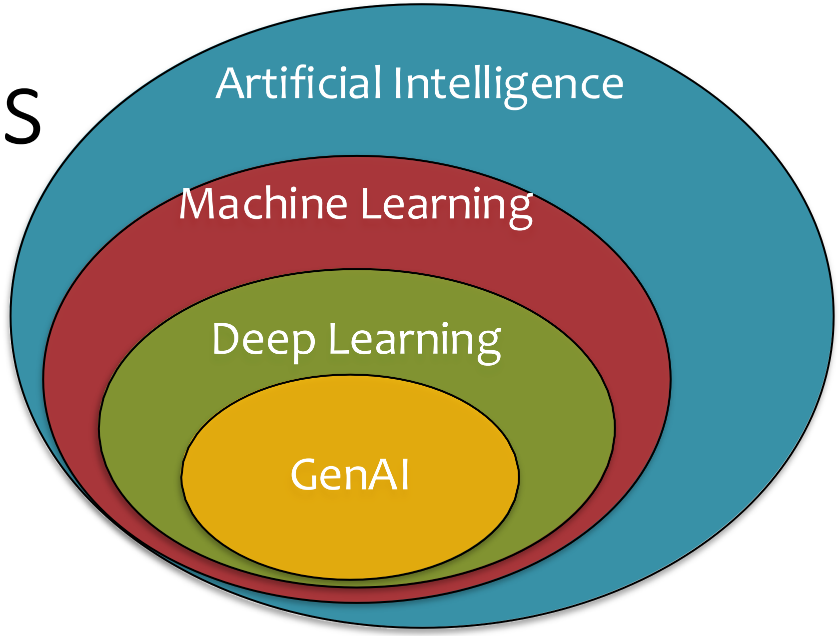
This consists of many sub-goals:

- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

- ☐ DayDreamer learns a generative model of experiences for RL, i.e. a World Model, without simulation
- ☐ Quadruped robot learns to walk in under 1 hour



Artificial Intelligence Workloads



The basic goal of AI is to develop intelligent machines.

This consists of many sub-goals:

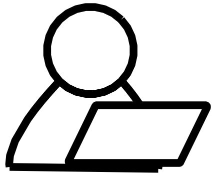
- Perception
- Reasoning
- Control / Motion / Manipulation
- Planning
- Communication
- Creativity
- Learning

OQ: What does Generative AI have to do with **any of these goals**?

OA: It's making in-roads into **all of them.**

Where can Systems fit into the picture

Machine Learning Systems



Researcher

Transformer ....

New Models

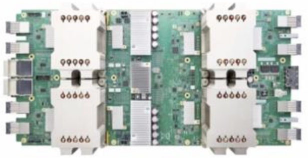
ML Research

44k lines of code

Six months

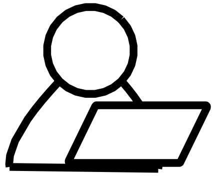
IMAGENET

Data



Compute

Machine Learning Systems



Researcher

ResNet ....

Transformer

ML Research

100 lines of python | A few hours

System Abstractions

Systems (ML Frameworks)

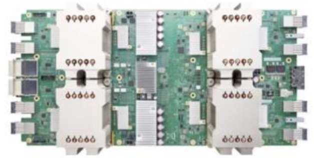
TensorFlow

mxnet

IMAGENET

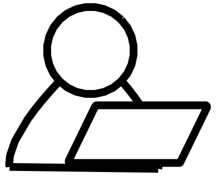
Data

NVIDIA CUDA



Compute

Machine Learning Systems



Researcher

ResNet ....

Transformer

ML Research

Model

100 lines of python

A few hours

System Abstract

Systems (ML Frameworks)

ML Systems

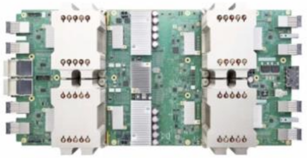
mxnet

IMAGENET

Data

Data

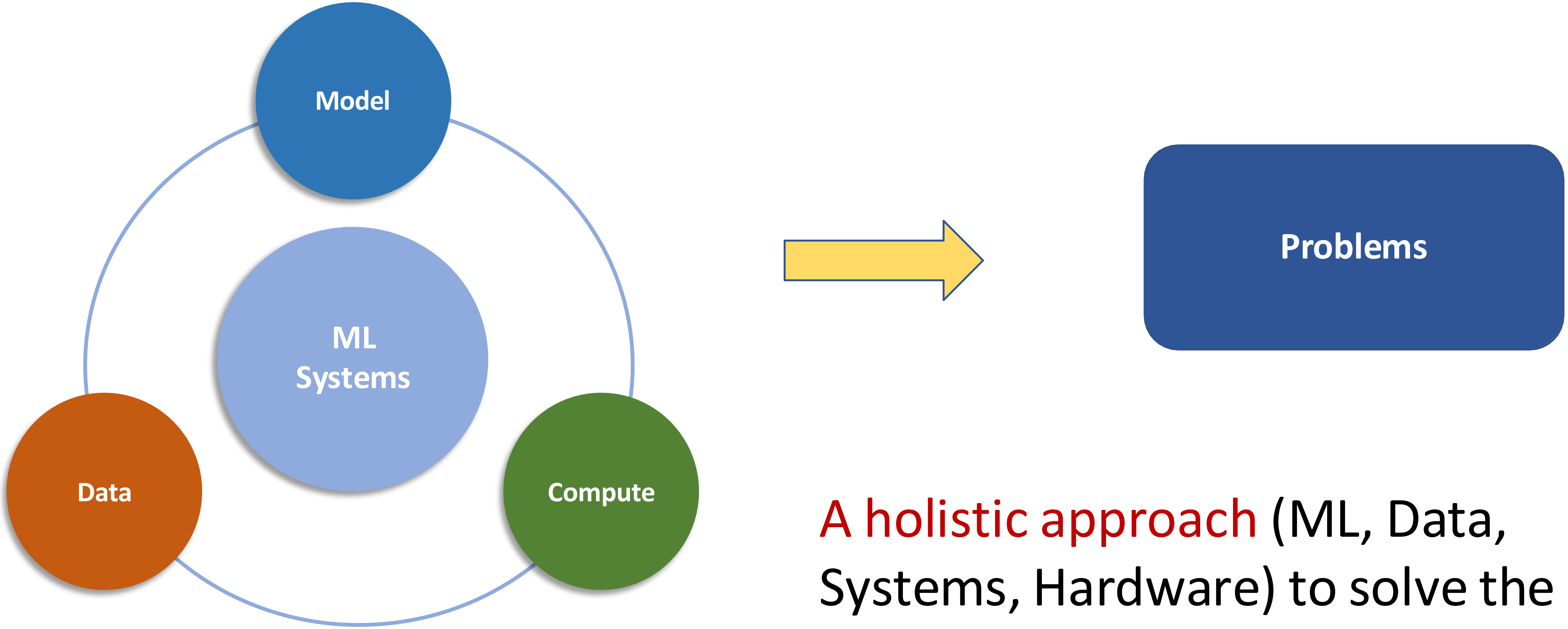
NVIDIA CUDA



Compute

Compute

MLSys as a Research Field



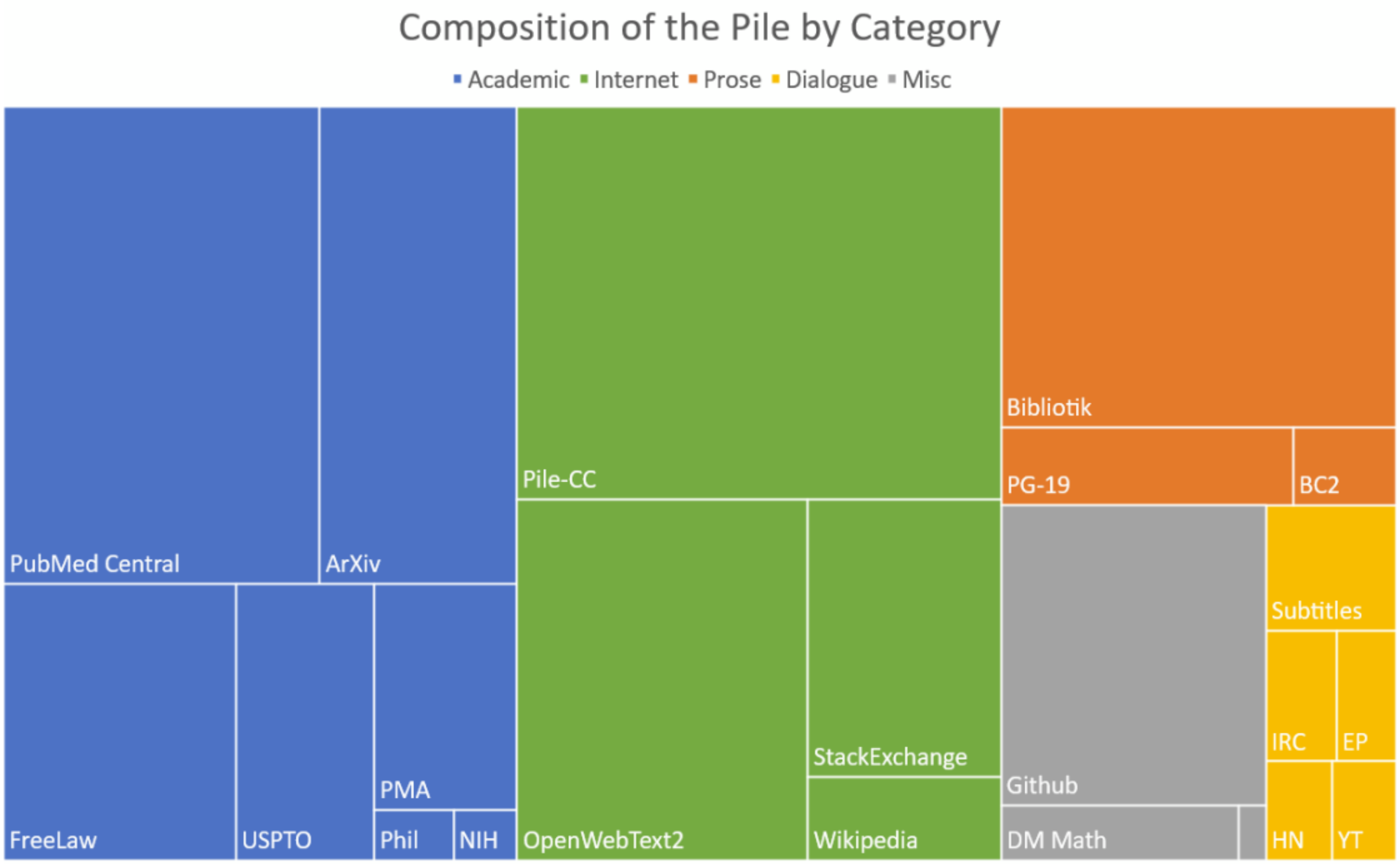
A holistic approach (ML, Data, Systems, Hardware) to solve the problem of interest.

SCALING UP

Training Data for LLMs

The Pile:

- An open-source dataset for training language models
- Comprised of 22 smaller datasets
- Favors high quality text
- 825 Gb ≈ 1.2 trillion tokens



RLHF

- **InstructGPT** uses Reinforcement Learning with Human Feedback (RLHF) to **fine-tune** a **pre-trained** GPT model
- From the paper: "In human evaluations on our prompt distribution, outputs from the 1.3B parameter InstructGPT model are preferred to outputs from the 175B GPT-3, despite having 100x fewer parameters."

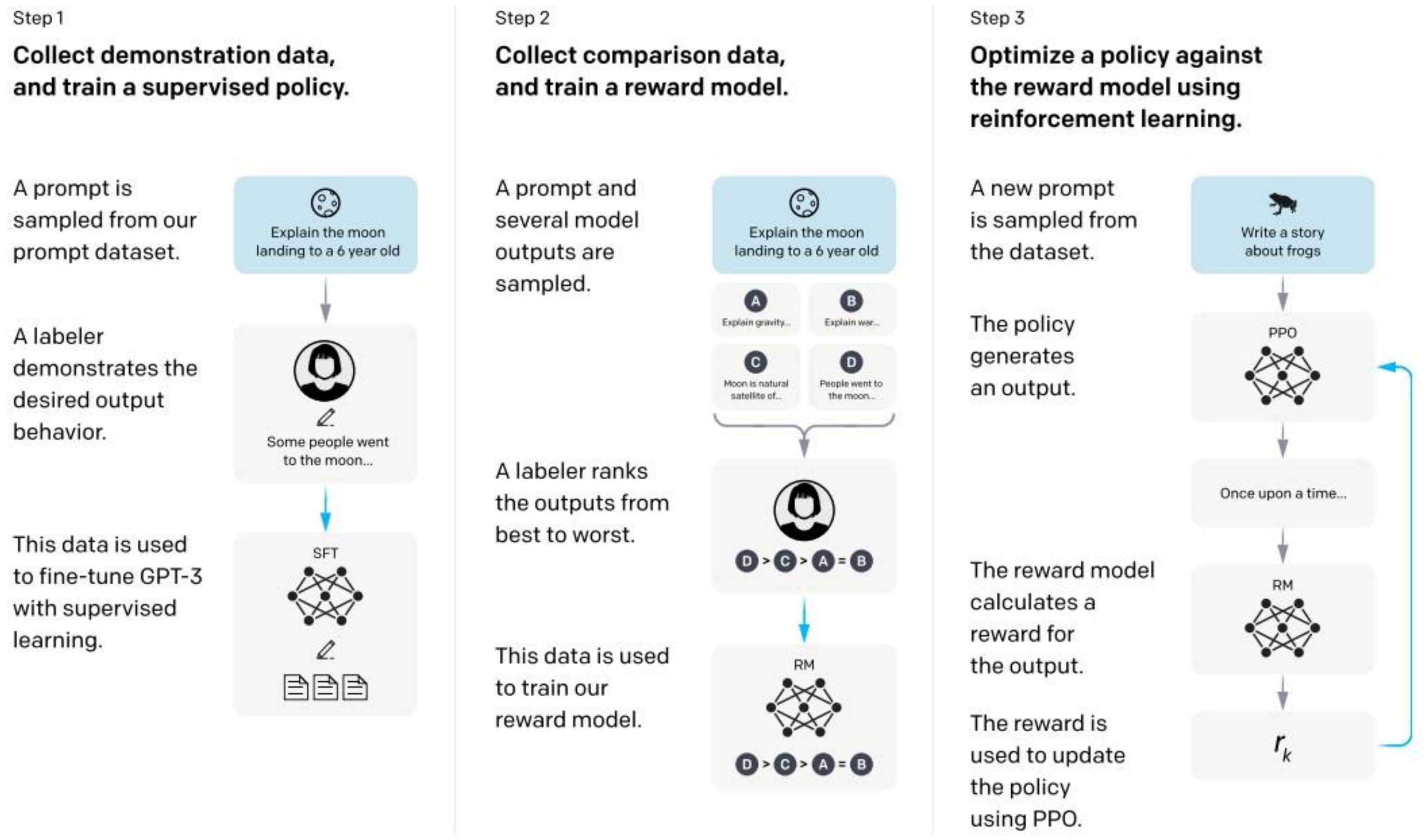


Figure 2: A diagram illustrating the three steps of our method: (1) supervised fine-tuning (SFT), (2) reward model (RM) training, and (3) reinforcement learning via proximal policy optimization (PPO) on this reward model. Blue arrows indicate that this data is used to train one of our models. In Step 2, boxes A-D are samples from our models that get ranked by labelers. See Section 3 for more details on our method.

Figure from https://arxiv.org/pdf/2203.02155.pdf

Memory Usage of LLMs

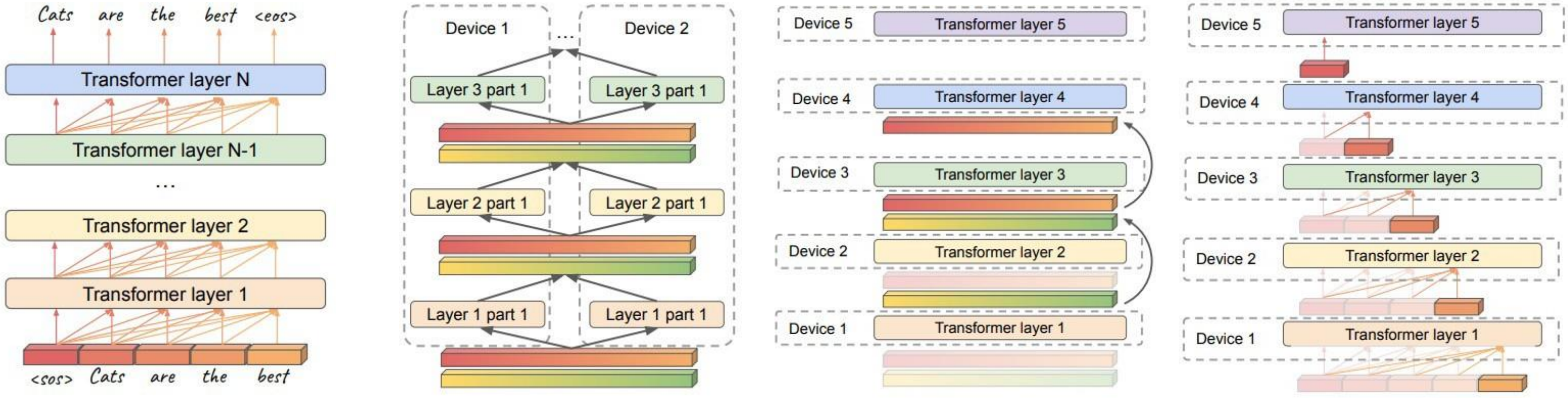
How to store a large language model in memory?

- **full precision:** 32-bit floats
- **half precision:** 16-bit floats
- Using half precision not only **reduces memory**, it also **speeds up** GPU computation
- "*Peak float16 matrix multiplication and convolution performance is 16x faster than peak float32 performance on A100 GPUs.*" [from Pytorch docs]

Model	Megatron-LM	GPT-3
# parameters	8.3 billion	175 billion
full precision	30 Gb	651 Gb
half precision	15 Gb	325 Gb

GPU / TPU	Max Memory
TPU v2	16 Gb
TPU v3/v4	32 Gb
Tesla V100 GPU	32 Gb
NVIDIA RTX A6000	48 Gb
Tesla A100 GPU	80 Gb

Distributed Training: Model Parallel



(a) Transformer-based LM

(b) Operation partitioning (Megatron-LM)

(c) Microbatch-based pipeline parallelism (GPipe)

(d) Token-based pipeline parallelism (TeraPipe)

There are a variety of different options for how to distribute the model computation / parameters across multiple devices.

Matrix multiplication comprises most Transformer LM computation and can be divided along rows/columns of the respective matrices.

The most natural division is by layer: each device computes a subset of the layers, only that device stores the parameters and computation graph for those layers.

A more efficient solution is to divide computation by token *and* layer. This requires careful division of work and is specific to the Transformer LM.

Figure from https://arxiv.org/pdf/2102.07988.pdf

Cost to train

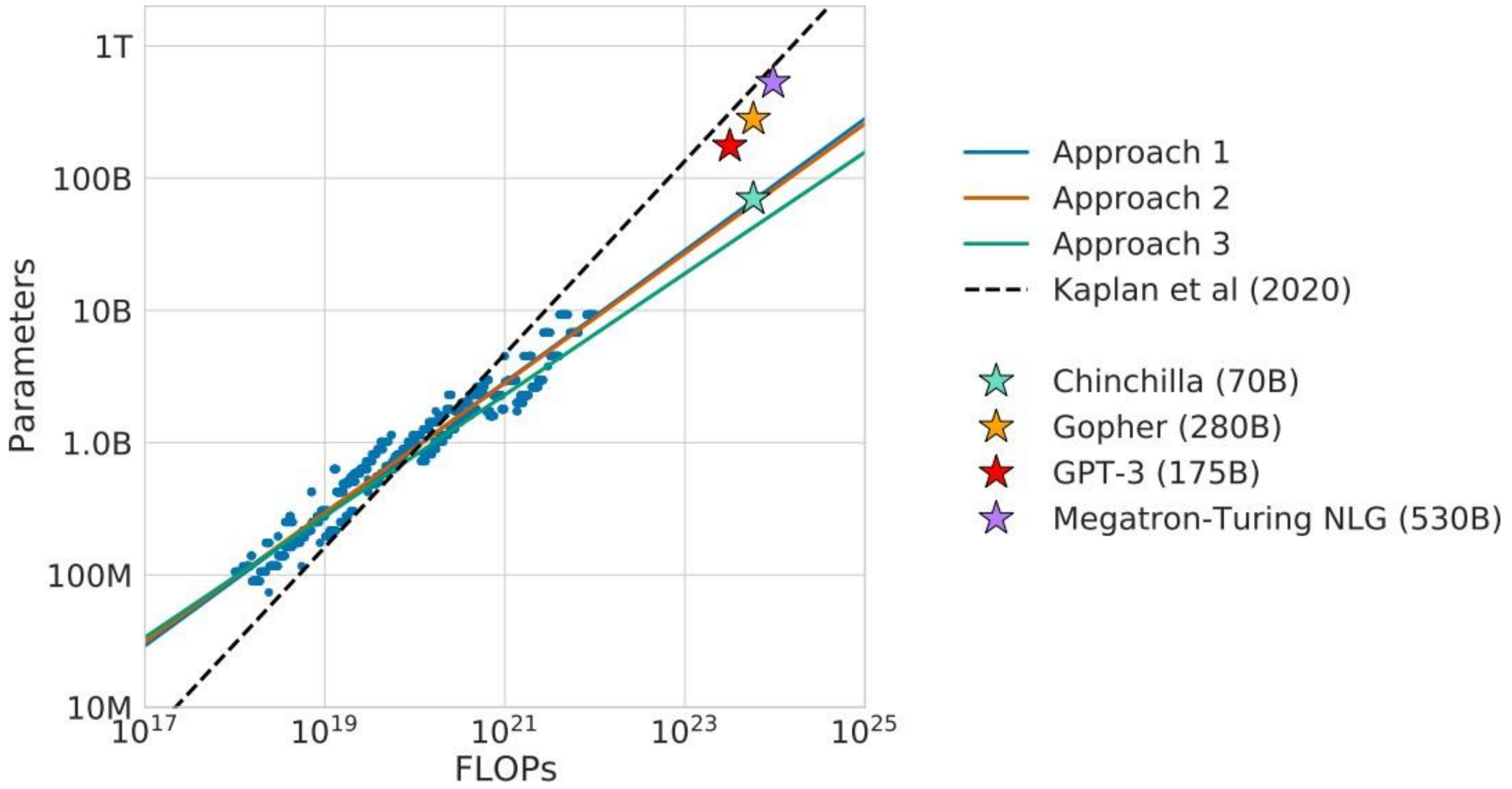
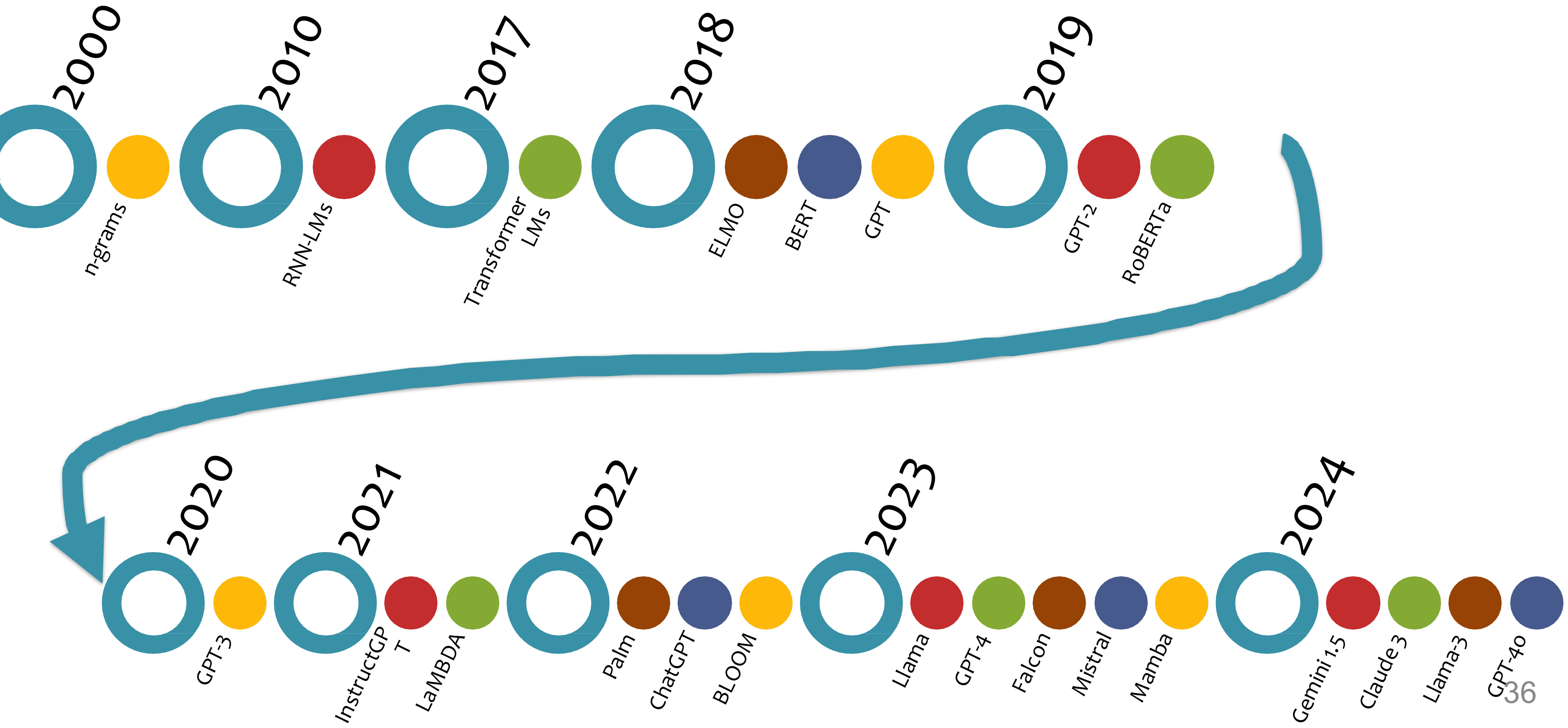


Figure from https://arxiv.org/pdf/2203.15556.pdf

Timeline: Language Modeling



Timeline: Image/Video Generation

What defines good

ML-Systems

Research Today?

What defines good

ML-Systems

Research Today?

Big Ideas in ML Research

- Generalization (Underfitting/Overfitting)
  - What is being “learned”?
- Inductive Biases and Representations
  - What assumptions about domain enable efficient learning?
- Efficiency (Data and Computation)
  - How much data and time are needed to learn?
- Details: Objectives/Models/Algorithms

What makes a great (accepted) paper?

State of the art results

Accuracy, Sample Complexity, Qualitative Results **…**

Novel settings, problem formulations, and **benchmarks**

Innovation in **techniques**: architecture, training methodology, …

Theoretical results that provide a deeper understanding

Narrative and **framing** in prior work and **current trends?**

Parsimony? Are elaborate solutions rejected? If they work better?

Verification of prior results?

What defines good

ML-Systems

Research Today?

Big Ideas in Systems Research

Managing Complexity

Abstraction, modularity, layering, and hierarchy

Tradeoffs

What are the fundamental constraints?

How can you reach new points in the trade-off space?

Problem Formulation

What are the requirements and assumptions?

What makes a great (accepted) paper?

State of the art results

throughput, latency, resource reqs., scale, …

Problem formulations and **benchmarks**

Innovation in **techniques**

Algorithms, data-structures, policies, software abstractions.

What you **remove** or **restrictions** often more important

Narrative and **framing** in prior work and **current trends?**

Verification of prior results?

Open source? Real-world use?

Goals: What can you get from this class

What Can You Get From This Class

- Ability to identify important problems
  - Identify new important problems in ML and Systems.
  - Formalize problems to measurable goals.

- MLSys approach of problem solving
  - Take a holistic approach (ML, different systems layers) to solve the problem.
  - Understand each part of the learning systems and how do they interact with each other.

Example: Problem Identification and Formalization

Safety is a critical problem in autonomous driving

Pedestrian detection is the bottleneck and impact the fail-safe system

Need to improve self-driving car's pedestrian detection to be X-percent accurate, at Y-ms latency budget

Example: MLSys Approach to Problem Solving

Need to improve self-driving car's pedestrian detection to be X-percent accurate, at Y-ms latency budget

- Collect more data
- Incorporate specialized compute hardware
- Develop models that optimizes for the specific hardware
- Built compilation solution to automate code optimization on the target hardware.

What Can You Get From This Class

- You won't be asked to build an end-to-end self-driving system
  - You are more than welcome to do so :)
- We will be looking at sub-problems (e.g., model training, inference)
- The same principle of MLSys approach applies

How Can We Achieve the Goals

- Overview lectures of areas in systems and ML
- Paper reading and presentation
  - Learn from existing examples of problem formalization.
  - Understand the layers of ML systems and how do they interact with each other.
- Write short paper reviews
  - Critical thinking
  - Learn and generalize ideas
- Final project
  - A MLSys project

Additional Tips

There are better classes to take if you want to learn

- General ML methods (take intro to ML)
- Data science toolkits (take practical in data science)

For students with ML background

- Take this class if you want to learn what is behind the scene and how to design model to take full advantage of systems.

For students with Systems background

- Understand the problems in systems field, solve the right problem.

Problems:

What makes a good problem?

What makes a **good problem?**

Impact: People care about the solution

… and progress advances our understanding (**research**)

Metrics: You know when you have succeeded

Can you **measure progress** on the solution?

Divisible: The problem can be divided into smaller problems

You can identify the first sub-problem.

Your Edge: Why is it a good problem ***for you?***

Leverage your strengths and imagine a new path.

Can You Solve a Solved Problem?

Ideally you want to solve a **new** and **important** problem

A **new solution** to a solved problem can be impactful if:

- It supports a **broader set of applications** (users)
- It **reveals a fundamental trade-off** or
- Provides a **deeper understanding** of the problem space
- **10x Better?**
  - Often publishable…
  - Should satisfy one of the three above conditions.

Logistics

Overview of the Course

- Overview lectures of areas in machine learning and systems
- Paper reading and presentation
  - Learn from existing examples of problem formalization.
  - Understand the layers of ML systems and how do they interact with each other.
- Write short paper reviews
  - Critical thinking
  - Learn and generalize ideas
- Final project
  - Build Something!

Class Format

- Overview Lecture: given by the instructor, overview of a sub-area
- Paper discussions: led by students, present and discuss paper reading materials
  - Usually follows the overview lecture
- Guest Lecture: given by external speakers on systems topics
  - Might be in different time, announcements will come before the class

Paper Readings and Reviews

Due before each paper discussion session.

- Papers from the reading list (~ two per week)
- One short review summarizing the first paper, in your own words
- One short review summarizing the second paper, in your own words
- One short paragraph on any connections between the papers, such as:
  - Compare and contrast
  - How one could apply ideas from one paper to solve the problem in the other paper
  - A new idea that would incorporate results from both papers etc

Discussion Session

- Paper presentations: 60 minutes (25 minutes per paper * 2)
    - 20 mins - presentation, 5 mins – question, 5 mins – buffer
- Presenters:
    - Submit slides before the class.
    - Prepare discussion questions and lead the discussions
- Discussion: 15 min
    - Class discussion about the two papers

Signup for Paper Presentations

Pick one paper from the list, present by one student. Each student is expected to present two times in the semester.

- Sign-up link will be posted to course website

Paper Presentation

Big Ideas(Overview/Motivation) — High level summary

Problem

Why is it important?

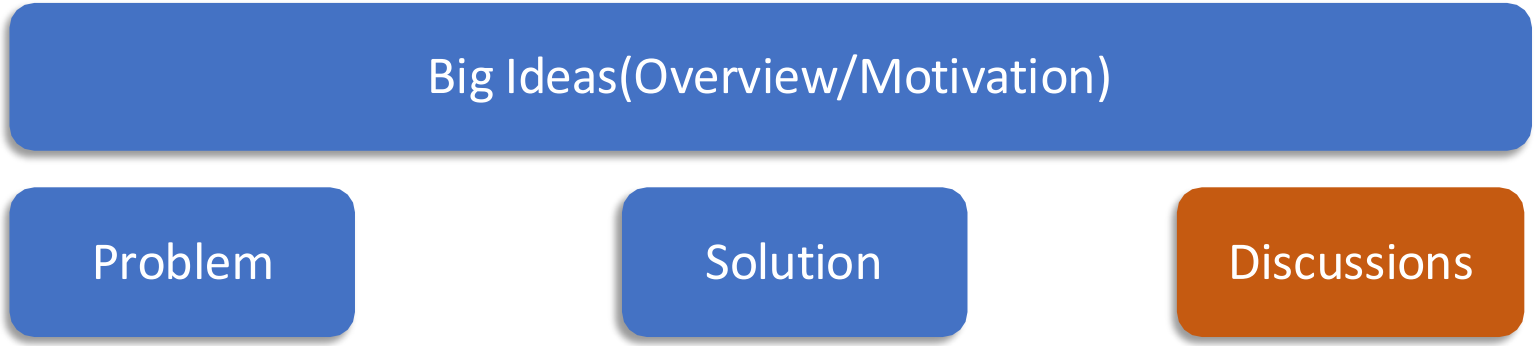
Solution

Key techniques

Discussions

Points for discussion:
- pros, cons
- connections

Discussions Session



Presenter needs to lead the discussion.

- The instructor will facilitate the Q&A.

Course Project

- Team of 1-X students (sign up in next week), find your team-mates early
- Discuss your project ideas. You are more than welcomed to bring your own topic.
- Initial 1-page proposal
- Informal mid-term check-in
- Final lightning presentation and writeup

Grading

- Participation: 10%
- Paper review: 20%
- Paper presentation: 20%
- Open-source engagement: 10%
- Project: 40%

All reviews/reports are submitted via HotCRP.

Ask Questions, Anytime

- You are more than welcomed to lead your own discussion thread
- Cloud Sys+AI/ML is an open field, there may not be definitive answers, let us explore the field together.

Always refer to the website for more details

https://zaoxing.github.io/teaching/2026-cloud-network

DEPARTMENT OF
COMPUTER SCIENCE