Machine Learning Systems Design

Lecture 1: Understanding ML production

CS 329 | Chip Huyen | cs329s.stanford.edu

Reply in Zoom chat:

Where are you (physically)?

Agenda

1. Course overview
2. ML in research vs. production
3. Breakout exercise
4. ML systems vs. traditional software
5. ML production myths

1. Short class today
2. Lecture note is on course website / syllabus
3. I'm in Vermont, sorry in advance about the bad Internet

1. Course overview

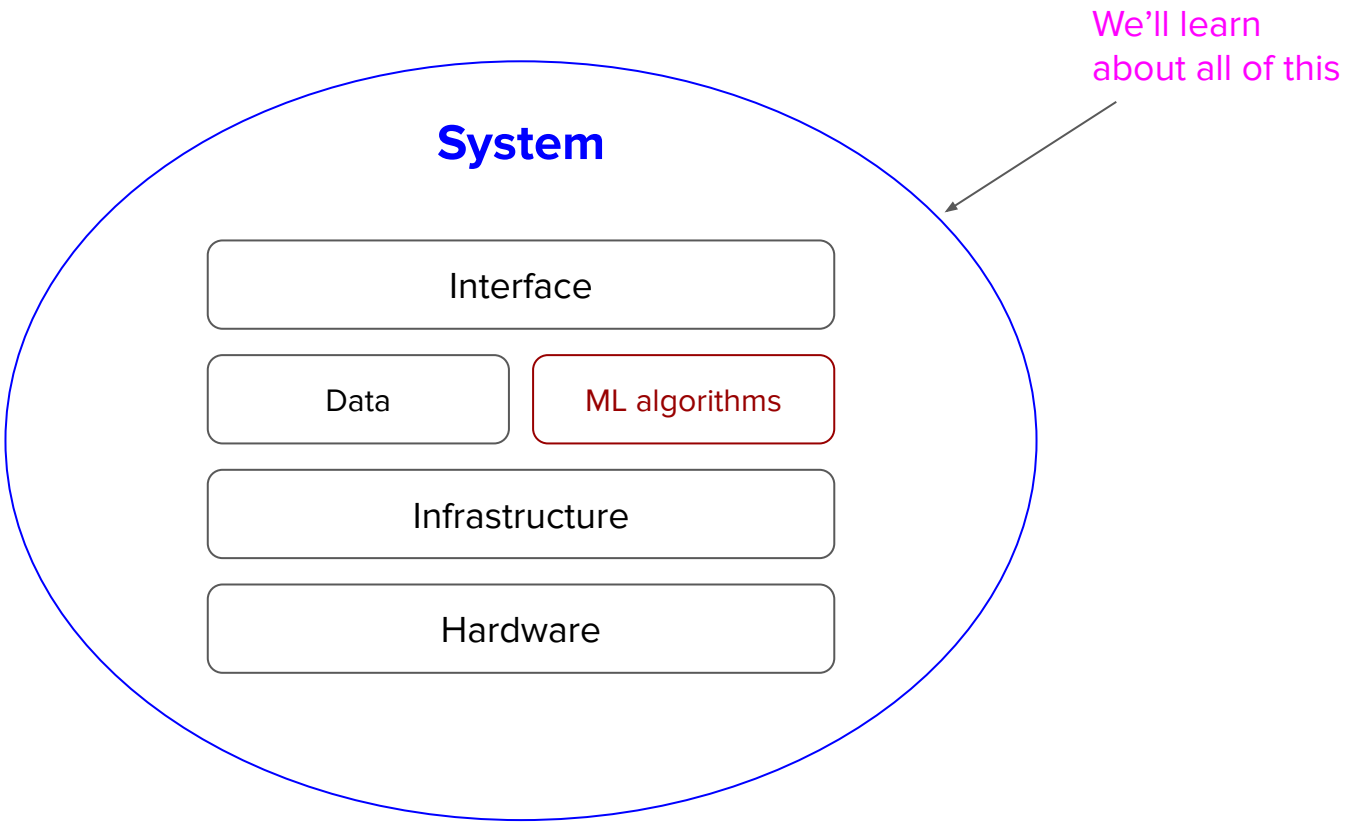
What’s machine learning systems design?

The process of defining the **interface, algorithms, data**, **infrastructure**, and **hardware** for a machine learning system to satisfy **specified requirements**.

What’s machine learning systems design?

The process of defining the **interface, algorithms, data**, **infrastructure**, and **hardware** for a machine learning system to satisfy **specified requirements**.

reliable, scalable, maintainable, adaptable


We'll learn
about all of this
System
Interface
Data
ML algorithms
Infrastructure
Hardware

This class will cover ...

- ML production in the real-world from software, hardware, business perspectives
- Iterative process for building ML systems at scale
  - project scoping, data management, developing, deploying, monitoring & maintenance, infrastructure & hardware, business analysis
- Challenges and solutions of ML engineering

This class will not teach ...

- Machine learning/deep learning algorithms
  - CS 229: Machine Learning
  - CS 230: Deep Learning
  - CS 231N: Convolutional Neural Networks for Visual Recognition
  - CS 224N: Natural Language Processing with Deep Learning
- Computer systems
  - CS 110: Principles of Computer Systems
  - CS 140E: Operating systems design and implementation
- UX design
  - CS 147: Introduction to Human-Computer Interaction Design
  - DESINST 240: Designing Machine Learning: A Multidisciplinary Approach

Machine learning: expectation

This class won't teach you how to do this

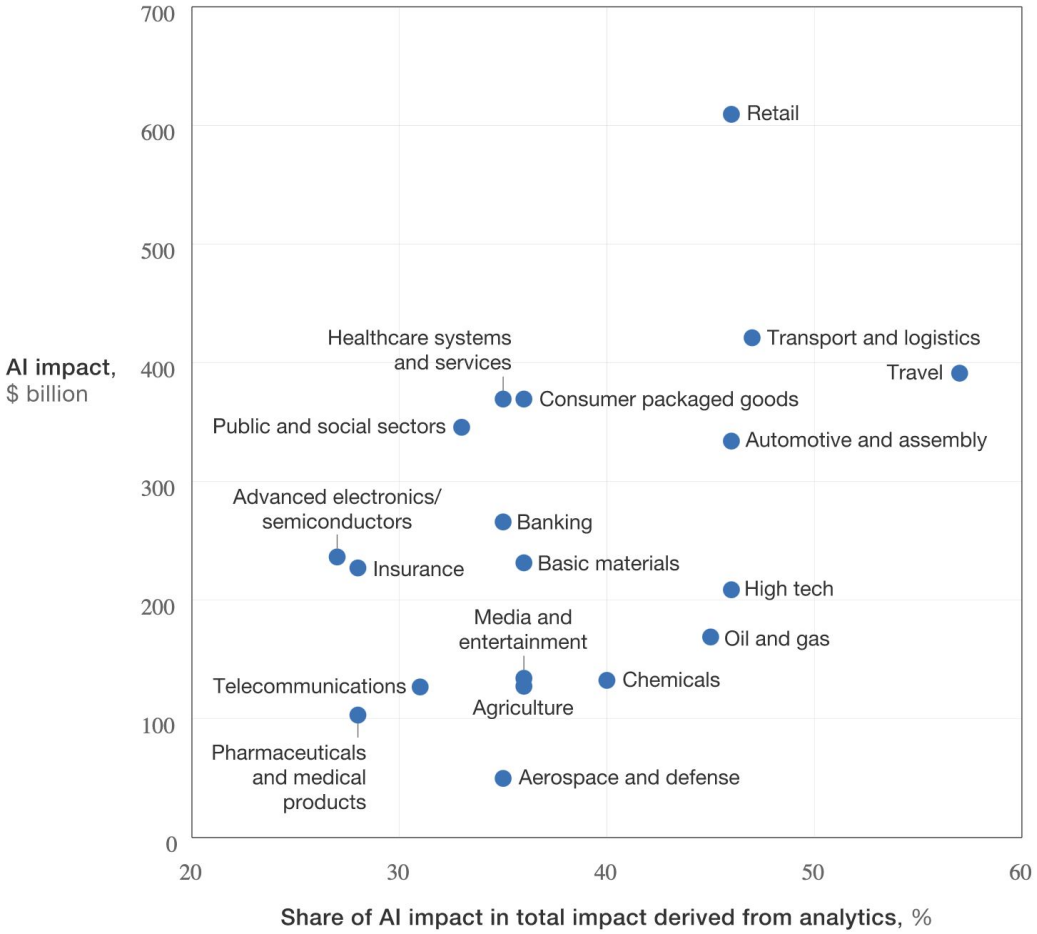
Machine learning: reality

This class will teach you how to build something like this (buggy but cool)

Prerequisites

- Knowledge of CS principles and skills (CS 106B/X)
- Understanding of ML algorithms (CS 229, CS 230, CS 231N, or CS 224N)
- Familiar with at least one framework such as TensorFlow, PyTorch, JAX
- Familiarity with basic probability theory (CS 109/Stat 116).

Artificial intelligence (AI) has the potential to create value across sectors.



AI value creation by 2030

13 trillion USD

Most of it will be outside the consumer internet industry

We need more people from non-CS background in AI!

McKinsey&Company | **Source:** McKinsey Global Institute analysis

Zoom etiquettes

- Write questions into Zoom chat
  - Feel free to reply to each other — TAs will also reply
- I will stop occasionally for Q&A
  - TAs will re-share some of the questions with me

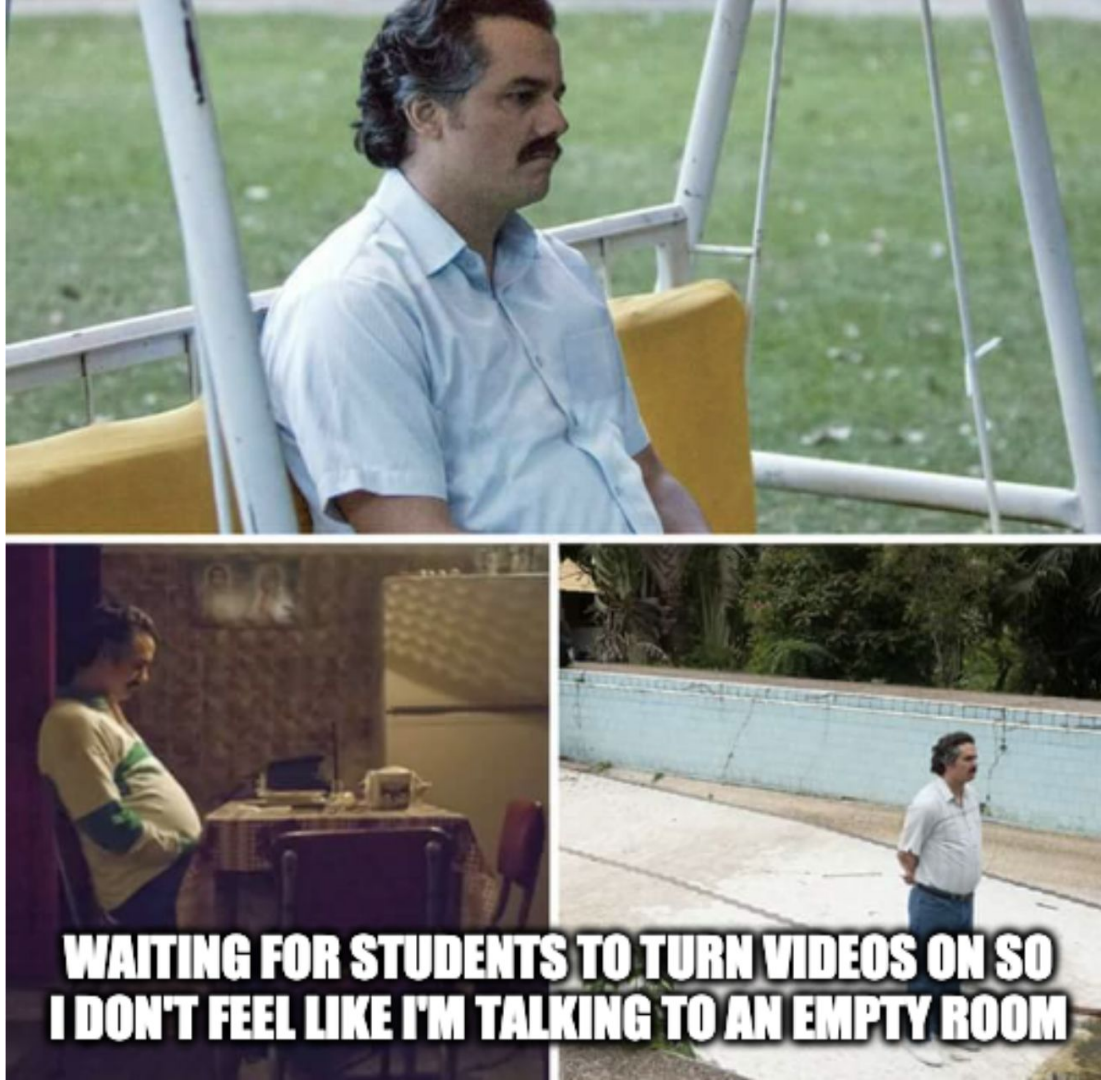
Zoom etiquettes

- Write questions into Zoom chat
  - Feel free to reply to each other — TAs will also reply
- I will stop occasionally for Q&A
  - TAs will re-share some of the questions with me
- After each lecture, a random question will get a random reward

Ping Karan if you want to opt out

Zoom etiquettes

We appreciate it
if you keep your video on!



Grading

- Assignments (30%)
  - 2-3 assignments
- Final project (60%)
- Class participation (10%)
  - Zoom questions + Piazza
  - Bad sign if by the end of the quarter, we still don’t know who you are

See course website: https://cs329s.stanford.edu

Final project

- Build an ML-powered application
- Must work in group of three
- Demo + report (creative formats encouraged)
- Evaluated by course staff and industry experts

See course website: https://cs329s.stanford.edu

Honor code: permissive but strict - don't test us ;)

- OK to search, ask in public about the systems we're studying. Cite all the resources you reference.
  - E.g. if you read it in a paper, cite it. If you ask on Quora, include the link.
- NOT OK to ask someone to do assignments/projects for you.
- OK to discuss questions with classmates. Disclose your discussion partners.
- NOT OK to copy solutions from classmates.
- OK to use existing solutions as part of your projects/assignments. Clarify your contributions.
- NOT OK to pretend that someone's solution is yours.
- OK to publish your final project after the course is over (we encourage that!)
- NOT OK to post your assignment solutions online.
- **ASK the course staff if unsure!**

Course staff

Karan Goel++

Michael Cooper

Xi Yan

Chip Huyen

See course website: https://cs329s.stanford.edu

⚠️⚠️ Work in progress ⚠️⚠️

- First time the course is offered
- First time Chip's taught a course online
- The subject is new, we don't have all the answers
  - We are all learning too!
- We appreciate your:
  - **enthusiasm** for trying out new things
  - **patience** bearing with things that don't quite work
  - **feedback** to improve the course

Inspired by Michael Bernstein's CS 278

- https://cs329s.stanford.edu
- OHs start next week
- If you enrolled without submitting an application, send us an email!
- Questions so far?

2. ML in research vs. production

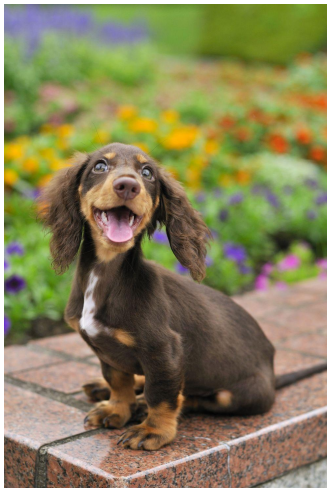
ML in research vs. in production

	Research	**Production**
Objectives	Model performance*	Different stakeholders have different objectives

"*" It's actively being worked. See Utility is in the Eye of the User: A Critique of NLP Leaderboards (Ethayarajh and Jurafsky, EMNLP 2020)

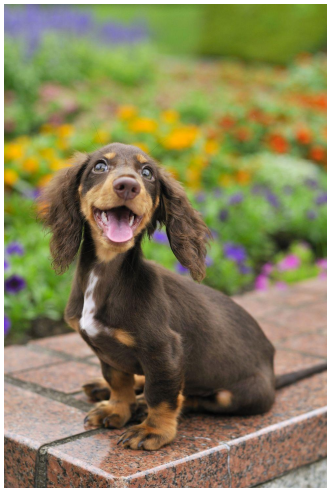
Stakeholder objectives

ML team
highest accuracy

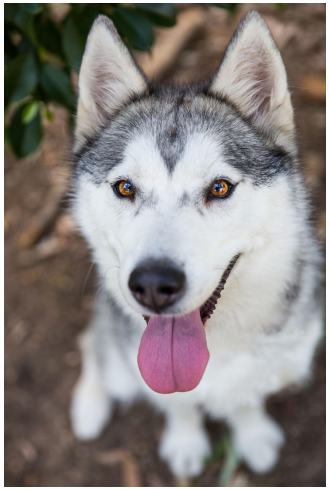


Stakeholder objectives

ML team
highest accuracy

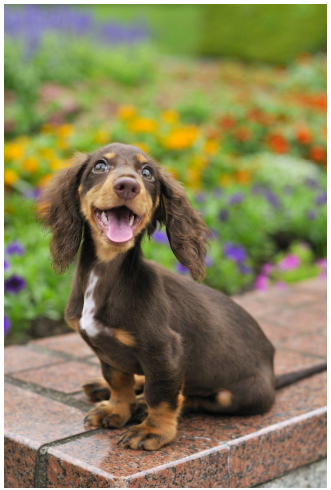


Sales
sells more ads

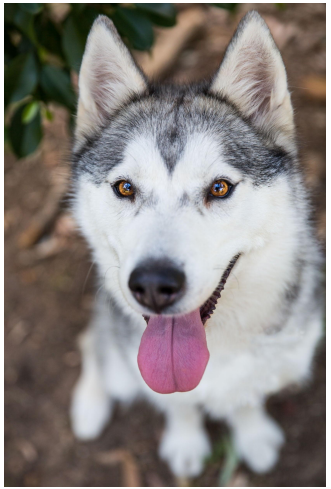


Stakeholder objectives

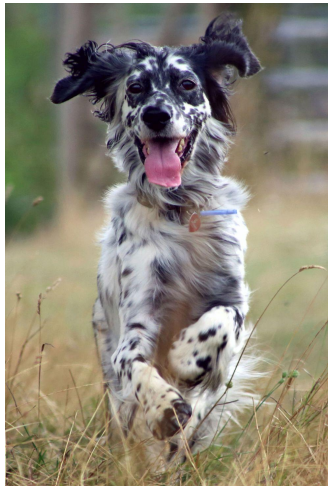
ML team
highest accuracy



Sales
sells more ads

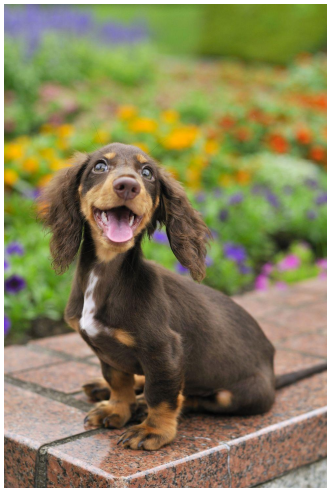


Product
fastest inference

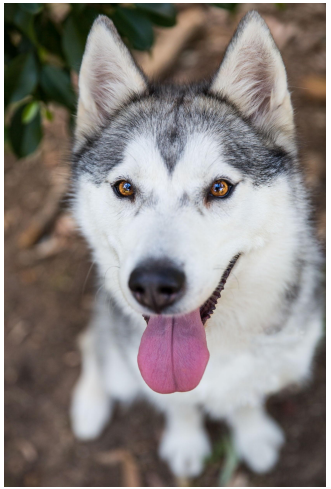


Stakeholder objectives

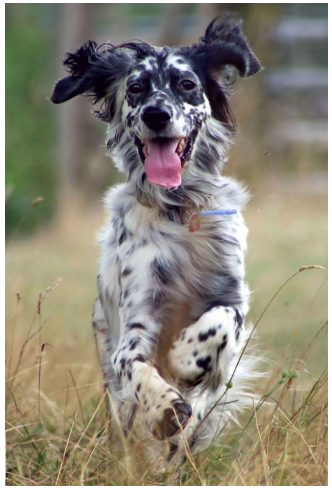
ML team
highest accuracy



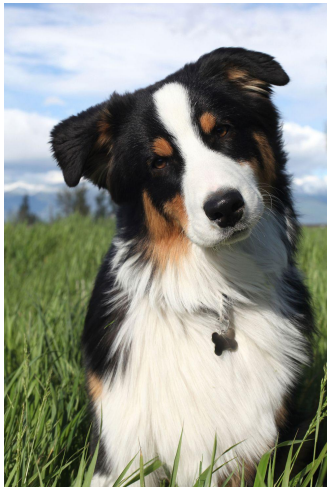
Sales
sells more ads



Product
fastest inference

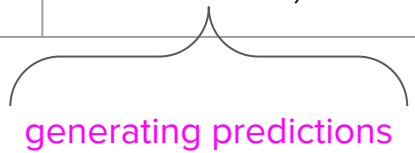


Manager
maximizes profit
= laying off ML teams



Computational priority

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency



Latency matters

Latency 100 -> 400 ms reduces searches 0.2% - 0.6% (2009)

30% increase in latency costs 0.5% conversion rate (2019)

- Latency: time to move a leaf
- Throughput: how many leaves in 1 sec

- Real-time: low latency = high throughput
- Batched: high latency, high throughput

ML in research vs. in production

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency
Data	Static	Constantly shifting

Data

Research	Production
● Clean ● Static ● Mostly historical data	● Messy ● Constantly shifting ● Historical + streaming data ● Biased, and you don’t know how biased ● Privacy + regulatory concerns

THE COGNITIVE CODER

By Armand Ruiz, Contributor, InfoWorld | SEP 26, 2017 7:22 AM PDT

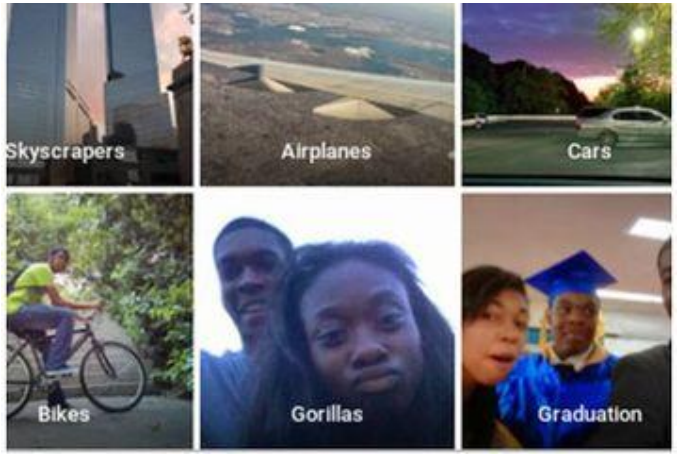
The 80/20 data science dilemma

Most data scientists spend only 20 percent of their time on actual data analysis and 80 percent of their time finding, cleaning, and reorganizing huge amounts of data, which is an inefficient data strategy

ML in research vs. in production

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency
Data	Static	Constantly shifting
Fairness	Good to have (sadly)	Important

Fairness

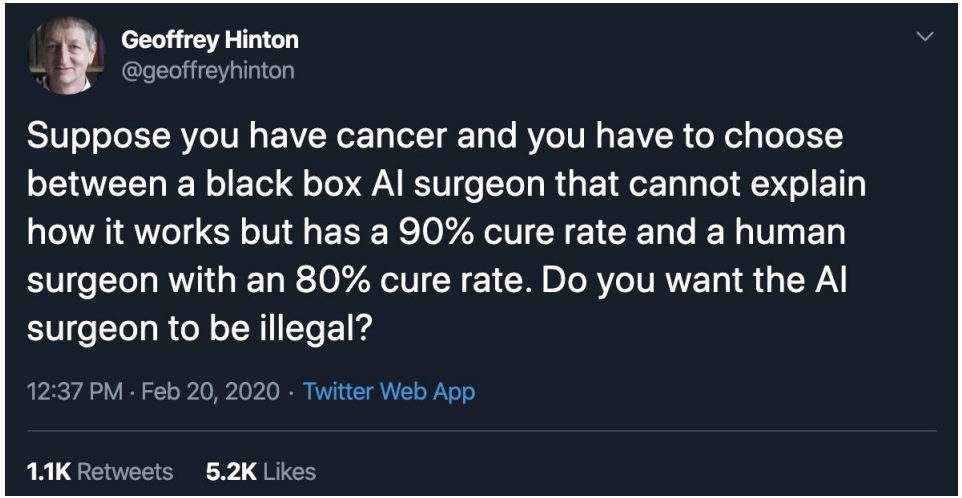


The Berkeley study found that both face-to-face and online lenders rejected a total of 1.3 million creditworthy black and Latino applicants between 2008 and 2015. Researchers said they believe the applicants "would have been accepted had the applicant not been in these minority groups." That's because when they used the income and credit scores of the rejected applications but deleted the race identifiers, the mortgage application was accepted.

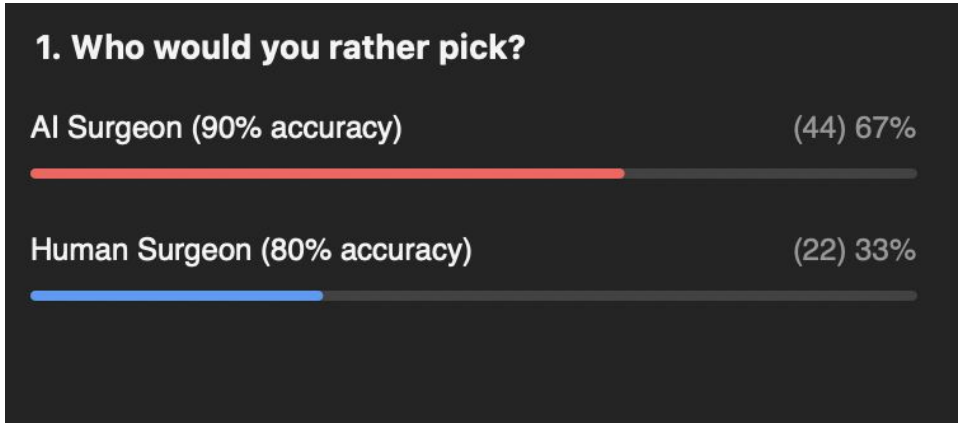
ML in research vs. in production

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency
Data	Static	Constantly shifting
Fairness	Good to have (sadly)	Important
Interpretability*	Good to have	Important

Interpretability



Result from the Zoom poll



ML in research vs. in production

	Research	**Production**
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast inference, low latency
Data	Static	Constantly shifting
Fairness	Good to have (sadly)	Important
Interpretability	Good to have	Important

3. Breakout exercise

Each lecture, you'll be randomly assigned to a group

7 mins - no one right answer!

1. How can academic leaderboards be modified to account for multiple objectives? Should they?
2. ML models are getting bigger and bigger. How does this affect the usability of these models in production?

Don’t forget to introduce yourself to your classmates!

Future of leaderboards

- More comprehensive utility function
  - Model performance (e.g. accuracy)
  - Latency
  - Prediction cost
  - Interpretability
  - Robustness
  - Ease of use (e.g. OSS tools)
  - Hardware requirements
- Adaptive to different use cases
  - Instead of a leaderboard for each dataset/task, each use case has its own leaderboard
- Dynamic datasets
  - Distribution shifts

Dynamic datasets

WILDS (Koh and Sagawa et al., 2020): 7 datasets with evaluation metrics and train/test splits representative of distribution shifts in the wild.

Dataset	Data (x)	Target (y)	Examples	Domains (d)	Domain count	Train/test domain overlap
FMoW	satellite images	land use	523,846	time	16	✗
				regions	5	✓
PovertyMap	satellite images	asset wealth	19,669	countries	23	✓
				urban/rural	2	✗
iWildCam2020	camera trap photos	animal species	217,609	trap locations	324	✗
Camelyon17	tissue slides	tumor	455,954	hospitals	5	✗
OGB-MolPCBA	molecular graphs	bioassays	437,929	molecular scaffolds	120,084	✗
Amazon	product reviews	sentiment	1,400,382	users	7,642	✗
CivilComments	online comments	toxicity	448,000	demographics	16	✓

4. ML systems vs. traditional software

Traditional software

Separation of Concerns is a design principle for separating a computer program into distinct sections such that each section addresses a separate concern

- Code and data are separate
  - Inputs into the system shouldn't change the underlying code

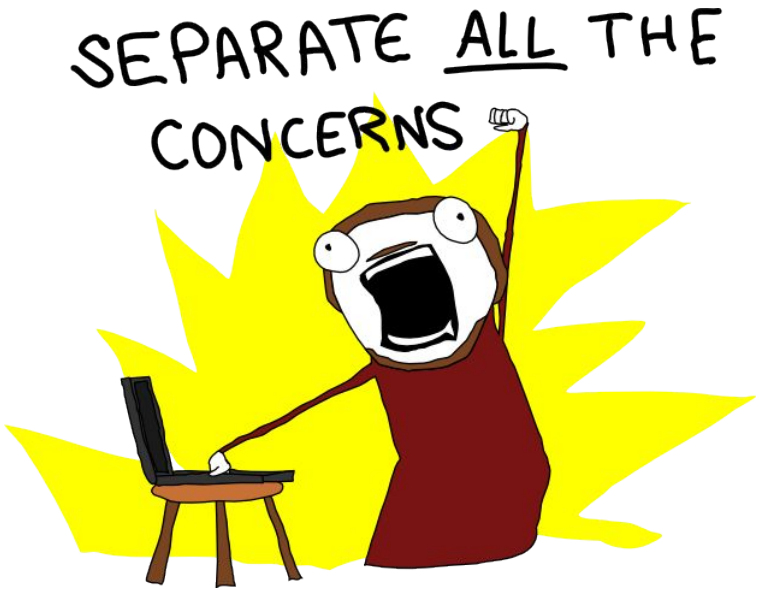


Image by Arda Cetinkaya

ML systems

- Code and data are tightly coupled
  - ML systems are part code, part data
- Not only test and version code, need to test and version data too

the hard part

ML System: version data

- Line-by-line diffs like Git doesn’t work with datasets
- Can’t naively create multiple copies of large datasets
- How to merge changes?

ML System: test data

- How to test data correctness/usefulness?
- How to know if data meets model assumptions?
- How to know when the underlying data distribution has changed? How to measure the changes?
- How to know if a data sample is good or bad for your systems?
    - Not all data points are equal (e.g. images of road surfaces with cyclists are more important for autonomous vehicles)
    - Bad data might harm your model and/or make it susceptible to attacks like data poisoning attacks

ML System: data poisoning attacks

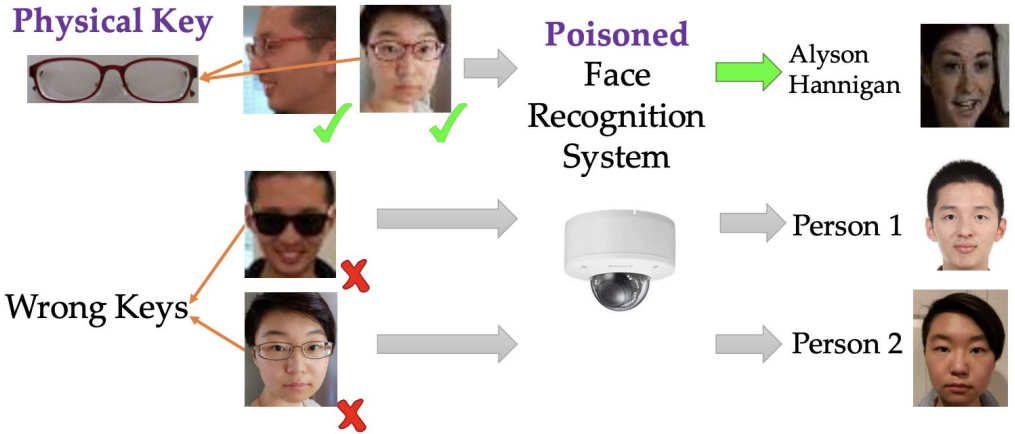


Fig. 1: An illustrating example of backdoor attacks. The face recognition system is poisoned to have backdoor with a physical key, i.e., a pair of commodity reading glasses. Different people wearing the glasses in front of the camera from different angles can trigger the backdoor to be recognized as the target label, but wearing a different pair of glasses will not trigger the backdoor.

Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning (Chen et al., 2017)

Engineering challenges with large ML models

- Too big to fit on-device
- Consume too much energy to work on-device
- Too slow to be useful
  - Autocompletion is useless if it takes longer to make a prediction than to type
- How to run CI/CD tests if a test takes hours/days?

5. ML production myths

Myth #1: Deploying is hard

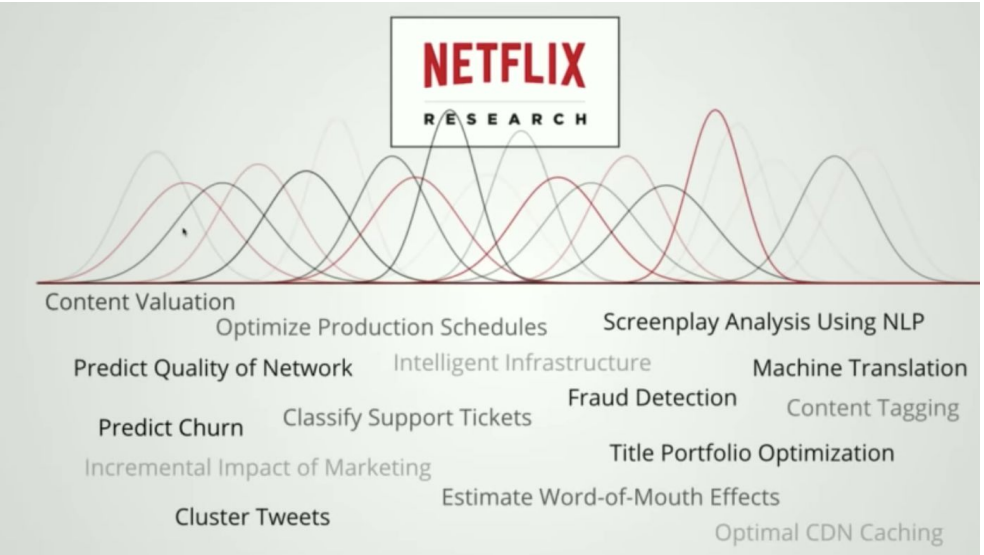
Myth #1: Deploying is hard

Deploying is easy. Deploying reliably is hard

Myth #2: You only deploy one or two ML models at a time

Myth #2: You only deploy one or two ML models at a time

Booking.com: 150+ models, Uber: thousands



Myth #3: If we don’t do anything, model performance remains the same

Myth #3: If we don’t do anything, model performance remains the same

Concept drift

Myth #3: If we don’t do anything, model performance remains the same

Concept drift
Tip: train models on data generated 2 months ago & test on current data to see how much worse they get.

Myth #4: You won’t need to update your models as much

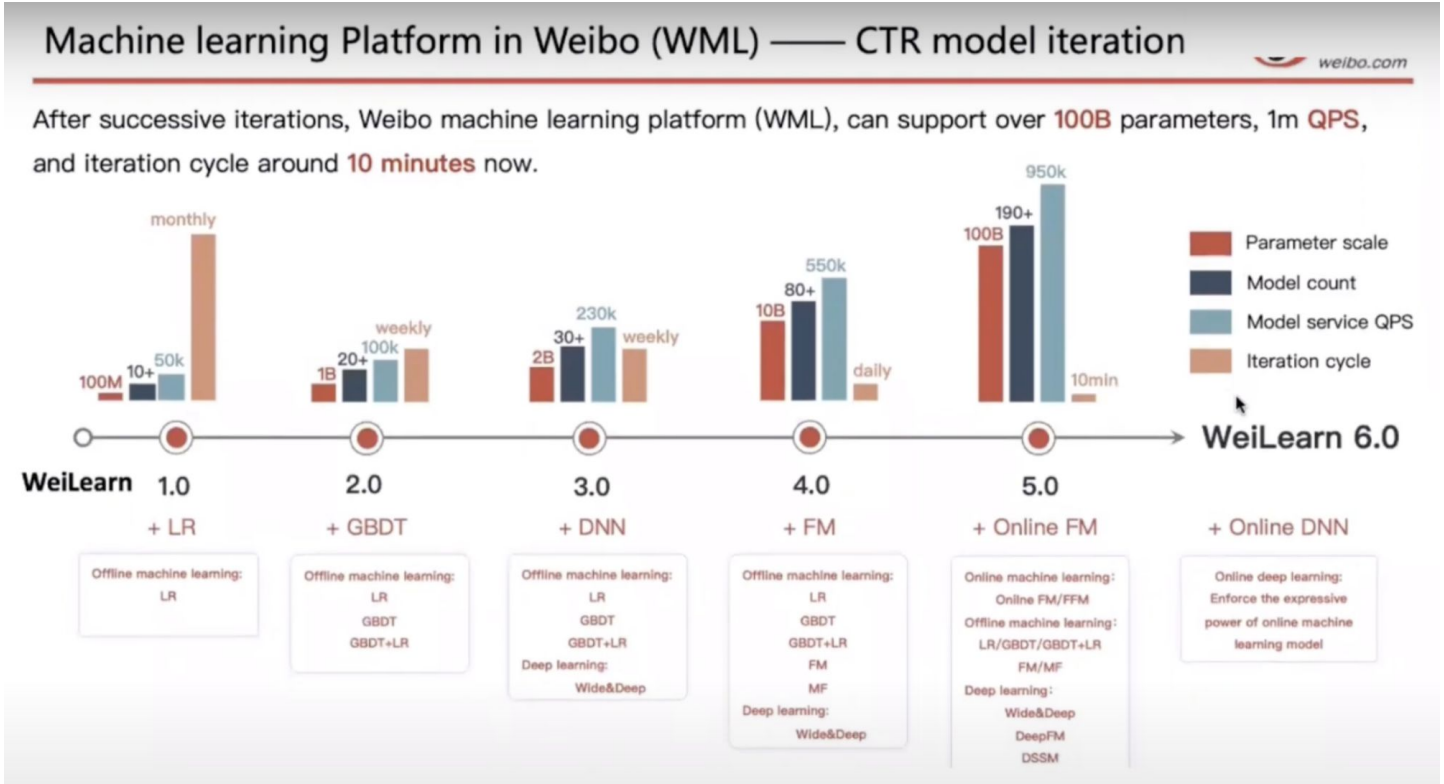
Myth #4: You won't need to update your models as much

DevOps standard

- Etsy deployed 50 times/day
- Netflix 1000s times/day
- AWS every 11.7 seconds

Weibo's ML iteration cycles: 10 minutes

Weibo's iteration cycle: 10 mins

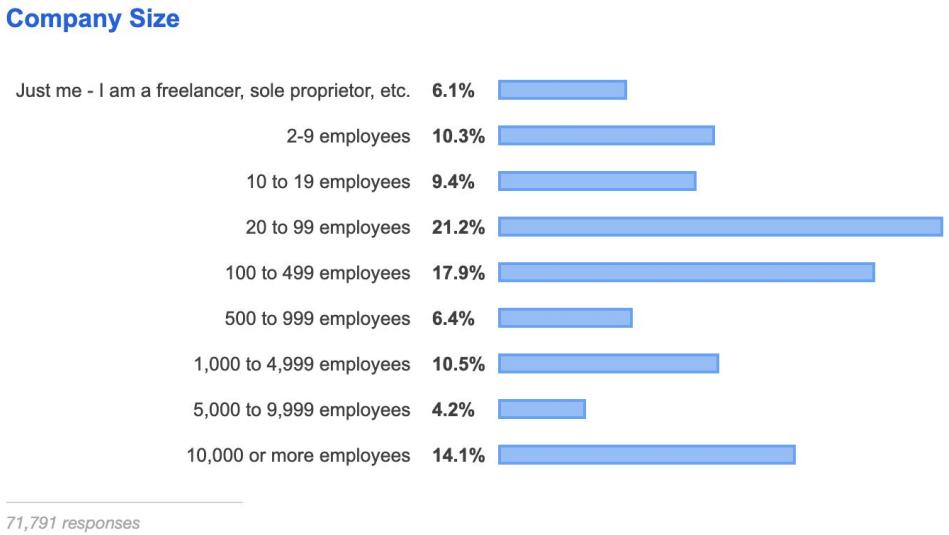


Machine learning with Flink in Weibo (Qian Yu, QCon 2019)

ML + DevOps = 

Myth #5: Most ML engineers don’t need to worry about scale

Myth #5: Most ML engineers don’t need to worry about scale



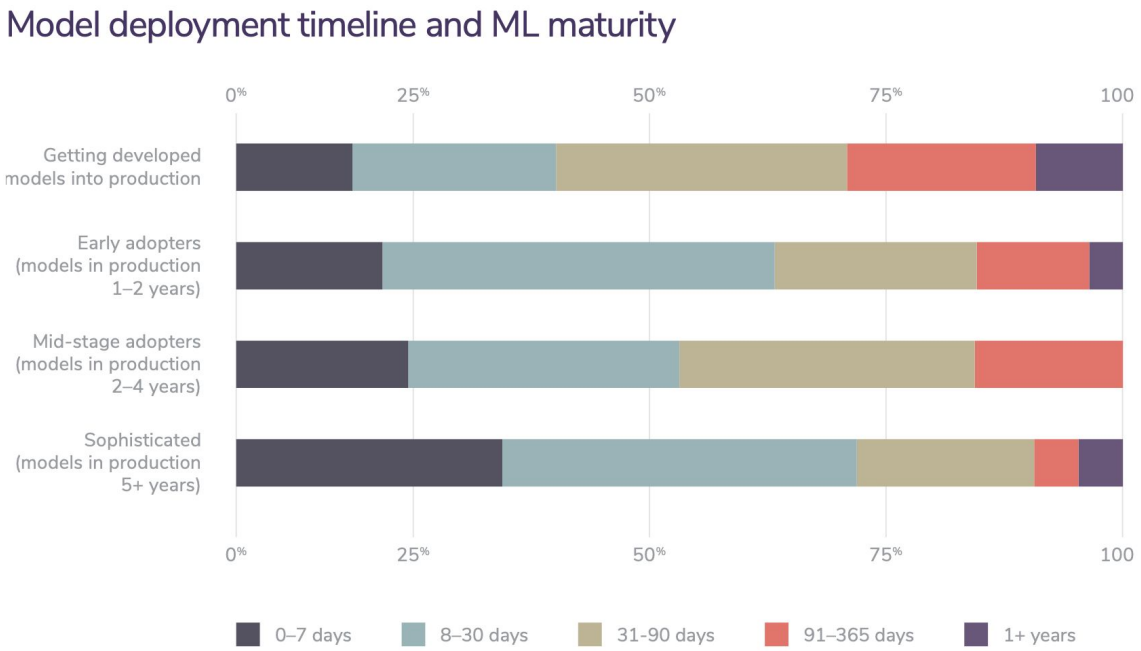
[StackOverflow Developer Survey 2019]

Myth #6: ML can magically transform your business overnight

Myth #6: ML can magically your business overnight

Magically: possible
Overnight: no

Efficiency improves with maturity



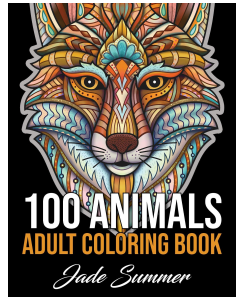
2020 state of enterprise machine learning (Algorithmia, 2020)


STAR WARS
THE MANDALORIAN
500
PUZZLE
ROMPECABEZAS


FERRERO
COLLECTION
24


Premium Scalp Massager


SMACKER


100 ANIMALS
ADULT COLORING BOOK
Jade Summer


Bear


COMO SI TE A Llama


Bob Ross

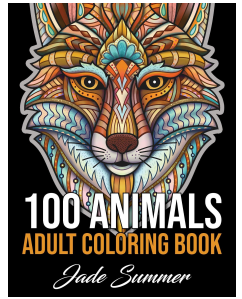
9

6

3

Ishan got a head scratcher!

2



7

5

1

8

10

4

Machine Learning Systems Design

Next class: Designing an ML system