Machine Learning Systems Design

What's machine learning systems design?

The process of defining the

- **interface,**
- **algorithms,**
- **data**,
- **infrastructure**,
- **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**,
- **hardware**

for a machine learning system to satisfy **specified requirements**.

$$\underbrace{\text{reliable, scalable, maintainable}}_{\text{general software systems}},\ \underbrace{\text{adaptable}}_{\text{systems that learn from data}}$$

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

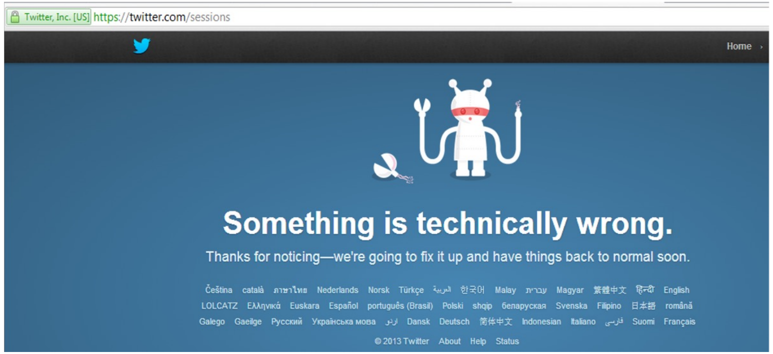
Reliability

The system should continue to perform the **correct function** at the **desired level of performance** even in the face of adversity (hardware or software faults, and even human error).

What does "correct" mean for ML systems when there are no ground truth labels?

ML systems fail silently

Normal software fails



ML systems fails

Scalability

As the system grows (in data volume, traffic volume, or complexity), there should be reasonable ways of dealing with that growth.

We'll focus on systems at scale!

Designing Data-Intensive Applications (Martin Kleppmann, O'Reilly 2017)

Scalability

As the system grows (in data volume, traffic volume, or complexity), there should be reasonable ways of dealing with that growth.

Autoscaling: the number of machines can go up or down depending on usage

Scalability: cautionary tale

Amazon's one hour of downtime on Prime Day may have cost it up to $100 million in lost sales

Sean Wolfe
Jul 19, 2018, 10:53 AM

> "If their auto-scaling was working, things would have scaled automatically and they wouldn't have had this level of outage," Caesar said. "There was probably an implementation or configuration error in their automatic scaling systems."

Internal documents show how Amazon scrambled to fix Prime Day glitches (Eugene Kim, CNBC 2018)

Maintainability

Over time, many people (ML engineers, DevOps, **subject matter experts**) will work on the system, and they should all be able to work on it productively.

The importance of SMEs in ML systems

- **Subject matter experts** (doctors, lawyers, bankers, farmers, stylists, etc.) are not only users but also developers of ML systems.
- Domain expertise is needed for:
  - problem formulation
  - data labeling
  - feature engineering
  - error analysis
  - model evaluation
  - reranking predictions

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

Maintainability: cross-team collaboration

- How to help engineers and SMEs communicate effectively?

New hot keywords: no-code / low-code ML

Adaptability

To adapt to **changing data distributions** and **business requirements**, the system should have some capacity for both **discovering aspects for performance improvement** and **allowing updates without service interruption**.

Linked to maintainability

We'll cover more about this later!

ML in research vs. in production

	Research	Production
Objectives	Model performance	Different stakeholders have different objectives

However, this is being debated: [Utility is in the Eye of the User: A Critique of NLP Leaderboards] (Ethayarajh and Jurafsky, EMNLP 2020)

"... historical focus on performance-based evaluation has been at the expense of other qualities that the NLP community values in models, such as compactness, fairness, and energy efficiency ....

For example, a highly inefficient model would provide less utility to practitioners but not to a leaderboard!

We advocate for more transparency on leaderboards, such as the reporting of statistics that are of practical concern (e.g., model size, energy efficiency, and inference latency)."

Computational priority

	Research	Production
Objectives	Model performance	Different stakeholders have different objectives
Computational priority	Fast training, high throughput	Fast $\underbrace{\text{inference}}_{\text{generating predictions}}$, low latency

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

Latency: time to move a leaf

Throughput: how many leaves in 1 sec

Real-time: low latency = high throughput

Batched: high latency, high throughput

Latency matters

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

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

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

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

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

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

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

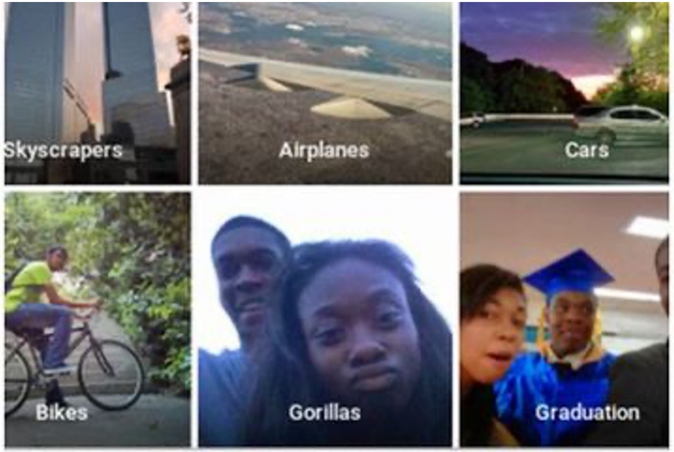
THE COGNITIVE CODER

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

Fairness



Google Shows Men Ads for Better Jobs

by Krista Bradford | Last updated Dec 1, 2019

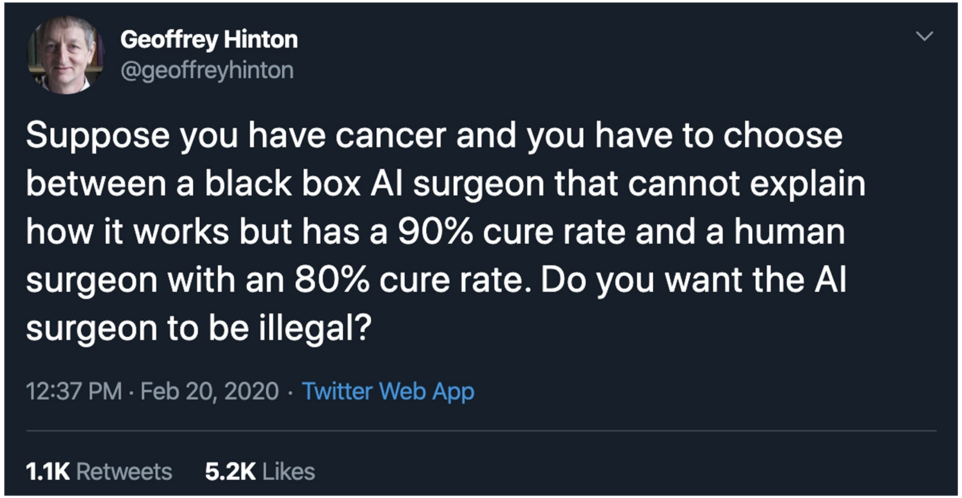
The Berkeley study found that both face-to-face and online lenders rejected a total of 1.3 million creditworthy black and Latino applicants between 2008 and 2015. Researchers said they believe the applicants "would have been accepted had the applicant not been in these minority groups." That's because 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

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

Interpretability



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

Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

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

“Distribution shifts -- where the training distribution differs from the test distribution -- can substantially degrade the accuracy of machine learning (ML) systems deployed in the wild.

Despite their ubiquity in the real-world deployments, these distribution shifts are under-represented in the datasets widely used in the ML community today.

To address this gap, we present WILDS, a curated benchmark of 10 datasets reflecting a diverse range of distribution shifts that naturally arise in real-world applications, such as shifts across hospitals for tumor identification; across camera traps for wildlife monitoring; and across time and location in satellite imaging and poverty mapping.”

WILDS (Koh and Sagawa et al., 2020):

Dynamic datasets

“On each dataset, we show that standard training yields substantially lower out-of-distribution than in-distribution performance.

This gap remains even with models trained by existing methods for tackling distribution shifts, underscoring the need for new methods for training models that are more robust to the types of distribution shifts that arise in practice. ”

WILDS (Koh and Sagawa et al., 2020)

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	✓

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

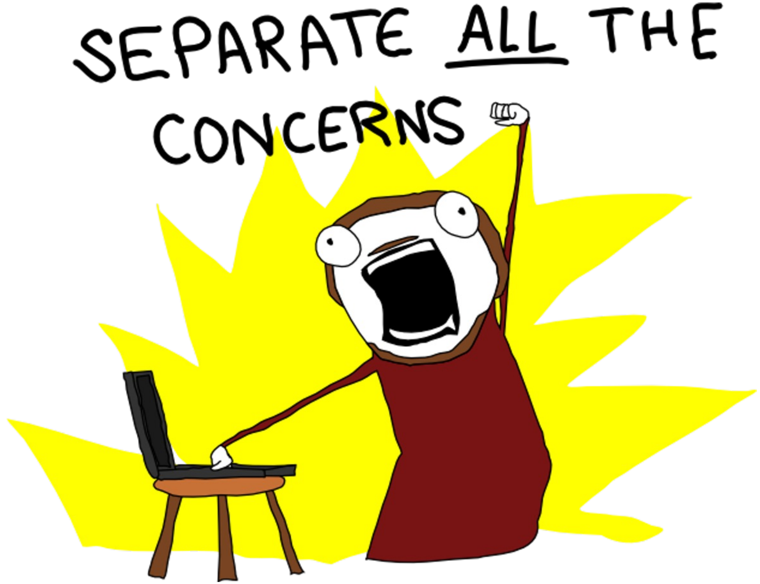


Image by Arda Cetinkaya

ML systems

- Code and data are tightly coupled
  - ML systems are part code, part data
- Not only test and version code, need to $\underbrace{\text{test and version data}}_{\text{the hard part}}$ too

ML System: version data

- Line-by-line diffs like Git does not work with datasets
- Cannot 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

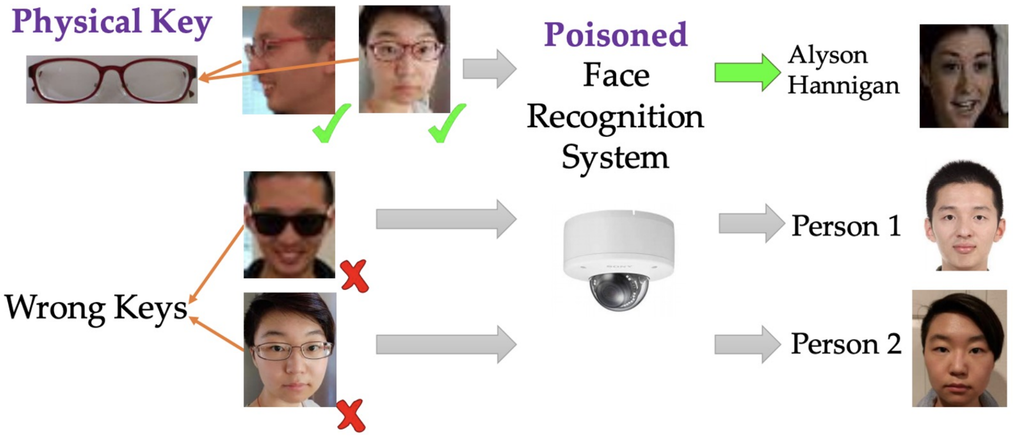


Fig. 1: An illustrating example of backdoor attacks. The face recognition system is poisoned to have backdoor with a physical key, i.e., a pair of commodity reading glasses. Different people wearing the glasses in front of the camera from different angles can trigger the backdoor to be recognized as 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?

ML production myths

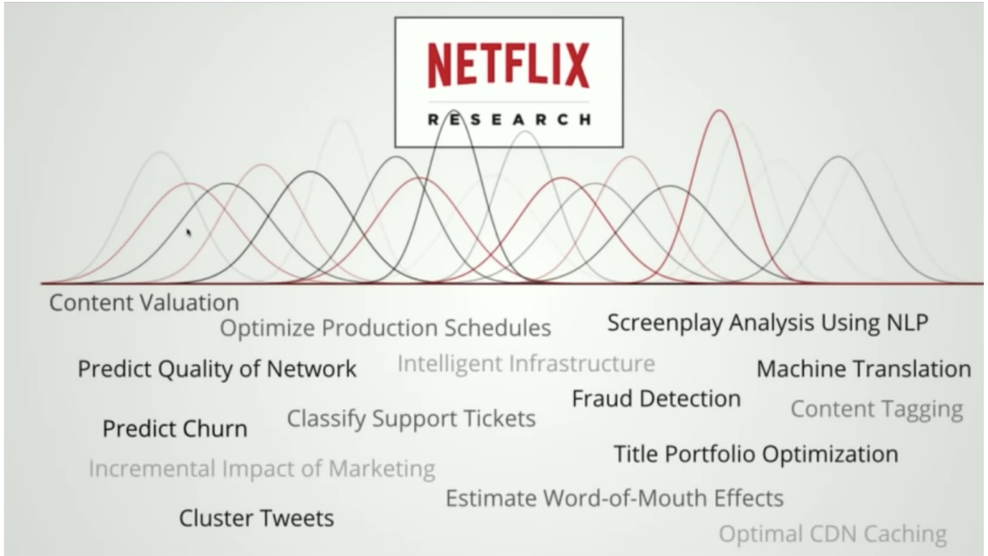
Myth #1: Deploying is hard

Deploying is not hard.
But, deploying reliably is hard!

ML production myths

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

Booking.com: 150+ models, Uber: thousands



Slide from: CS 329 | Chip Huyen | cs329s.stanford.edu

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 got!

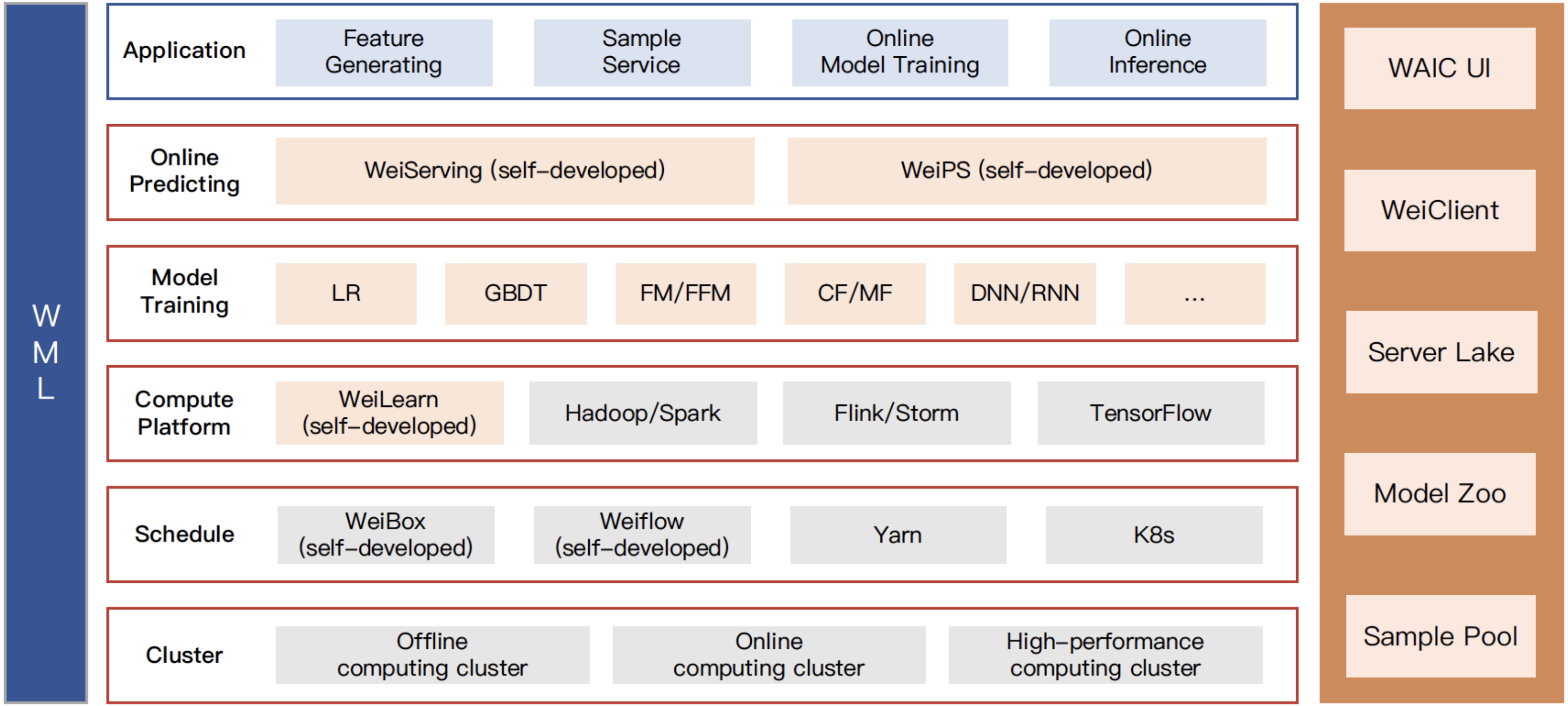
Myth #4: You won't need to update your models as much

DevOps standard

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

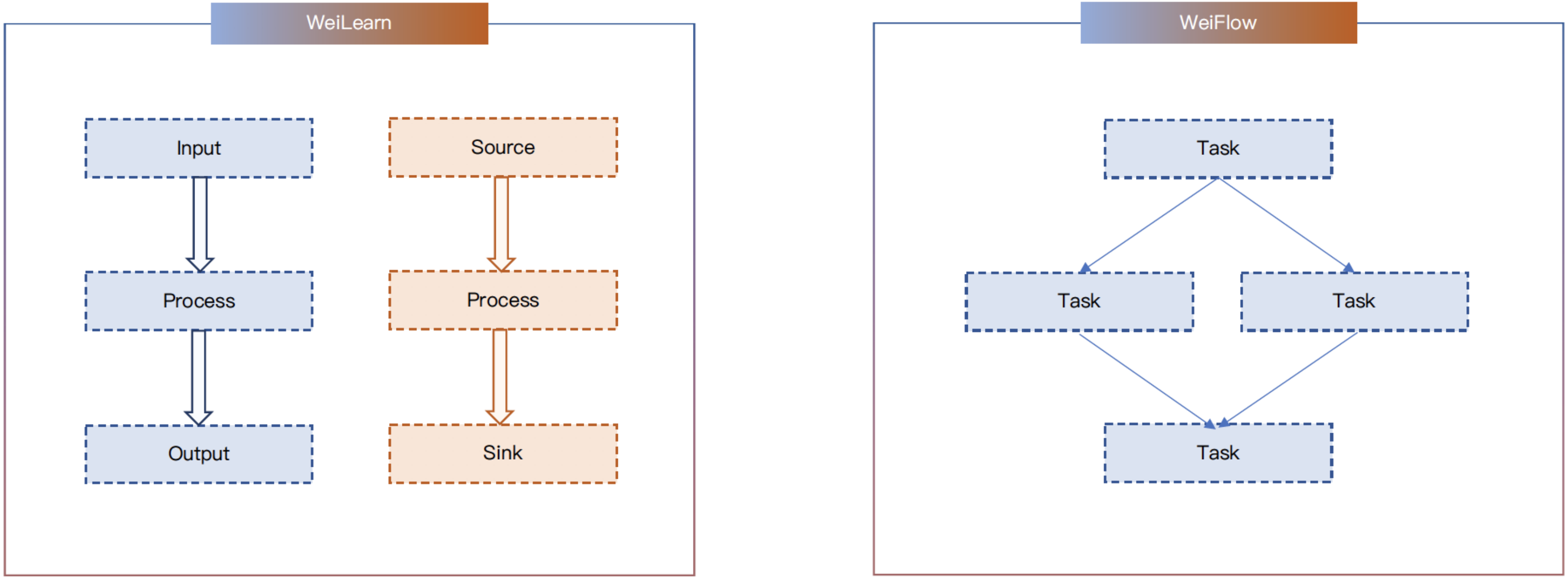
Weibo's ML iteration cycles: 10 minutes

Weibo's iteration cycle: 10 mins



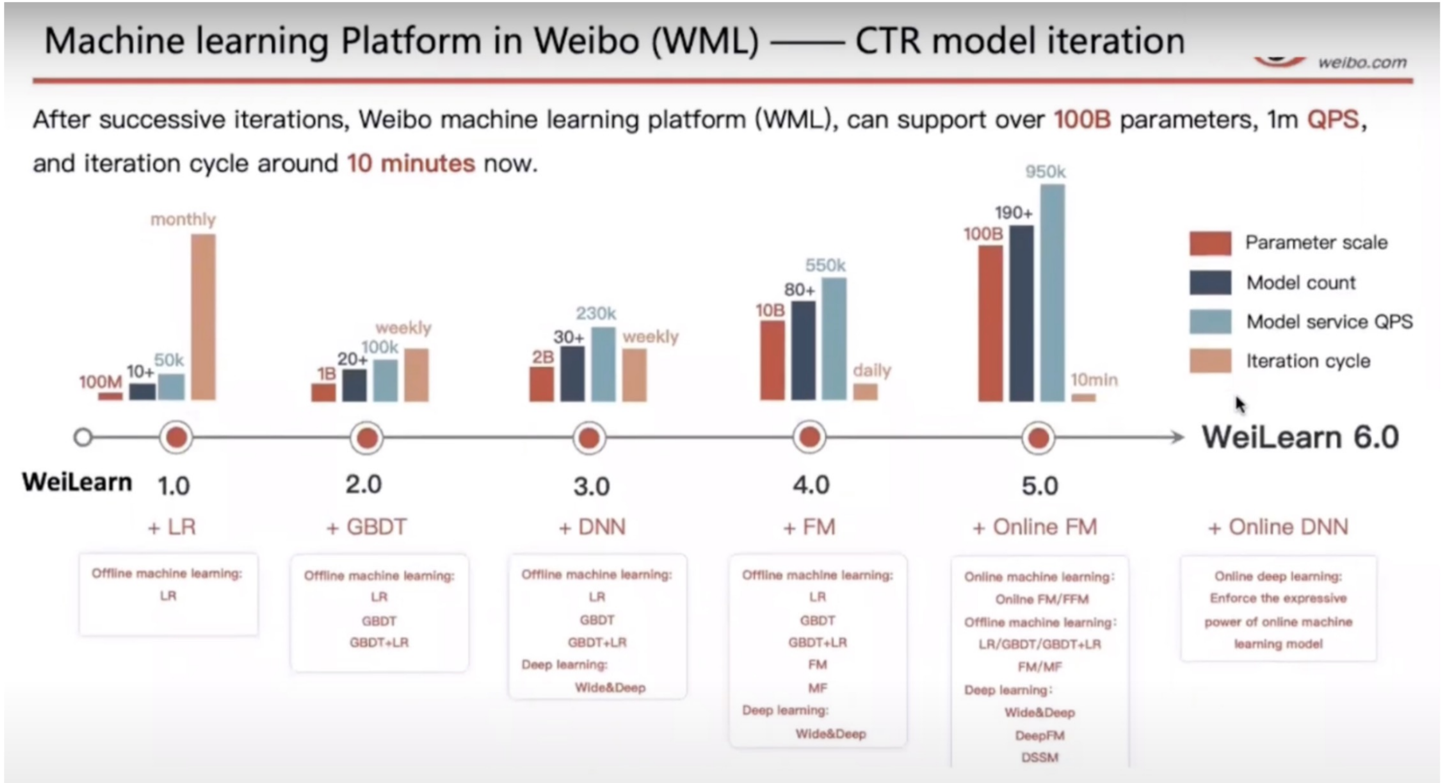
Machine learning with Flink in Weibo (Qian Yu, QCon 2019)

Weibo's iteration cycle: 10 mins



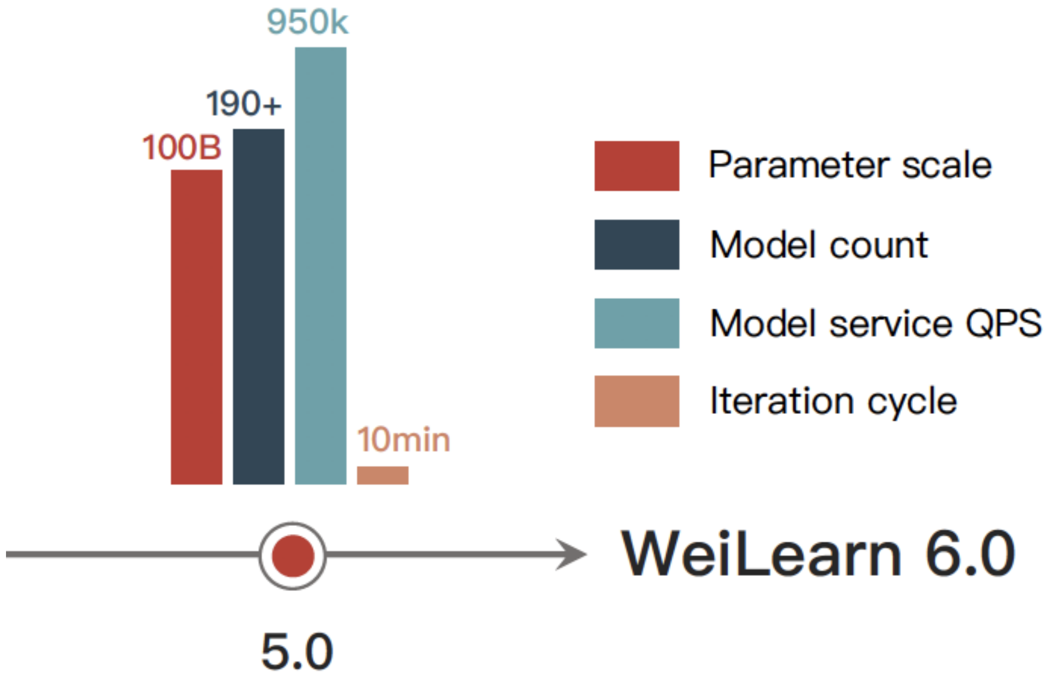
Machine learning with Flink in Weibo (Qian Yu, QCon 2019)

Weibo's iteration cycle: 10 mins



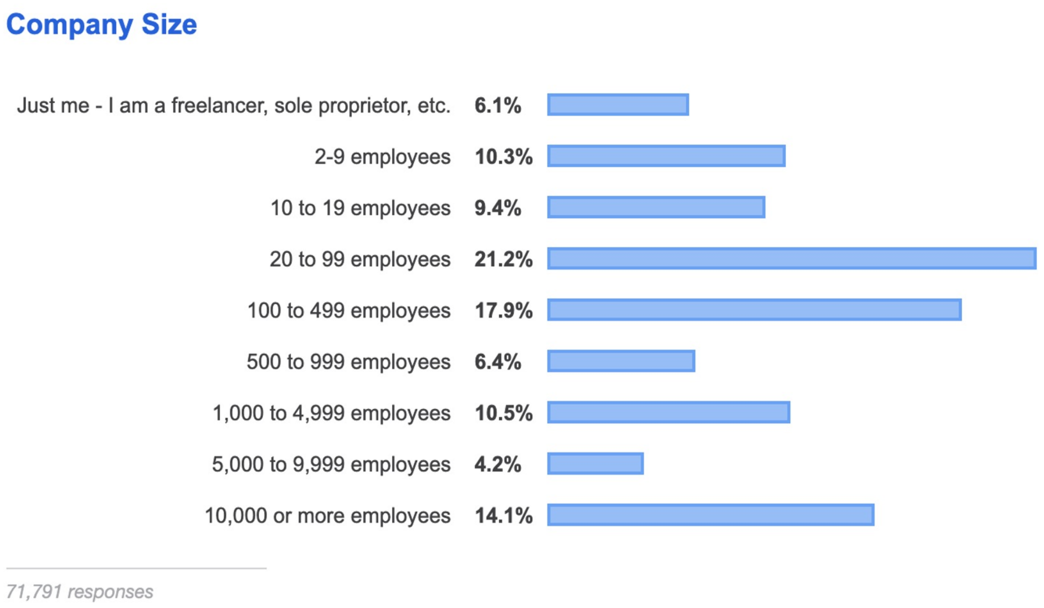
Machine learning with Flink in Weibo (Qian Yu, QCon 2019)

Weibo's iteration cycle: 10 mins



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

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

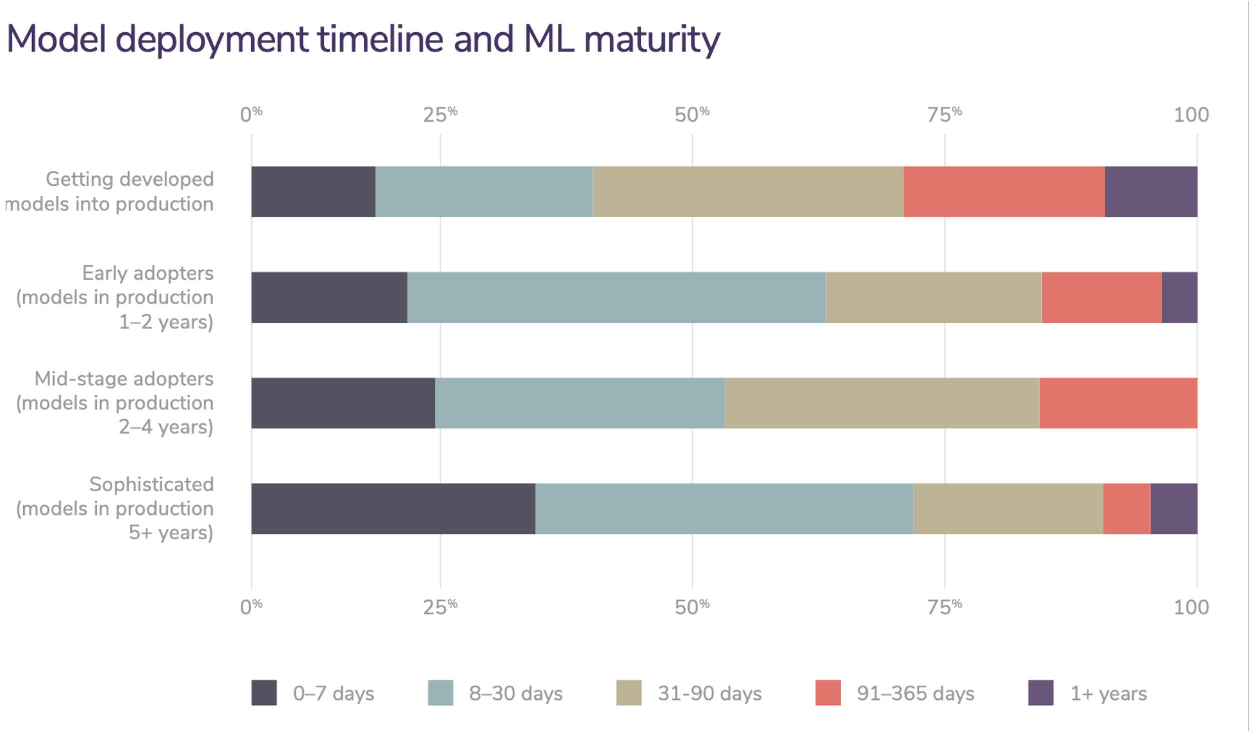


StackOverflow Developer Survey 2019

Myth #6: ML can change your business overnight

Magically: possible, you just need one of these lamps

Efficiency improves with maturity



2020 state of enterprise machine learning (Algorithmia, 2020)