

CSE 234

Data Systems for Machine Learning

Winter 2024

Arun Kumar

About Myself



Summers: 110F!

2009: Bachelors in CSE from IIT Madras, India



Winters: -40F!

2009–16: MS and PhD in CS from UW-Madison
PhD thesis area: Data systems for ML workloads



2016-: UC San Diego CSE
2019-: + UC San Diego HDSI

Ahh! :)

My Current Research

New abstractions, algorithms, and software systems
to “**democratize**” ML-based data analytics from
a data management/systems standpoint

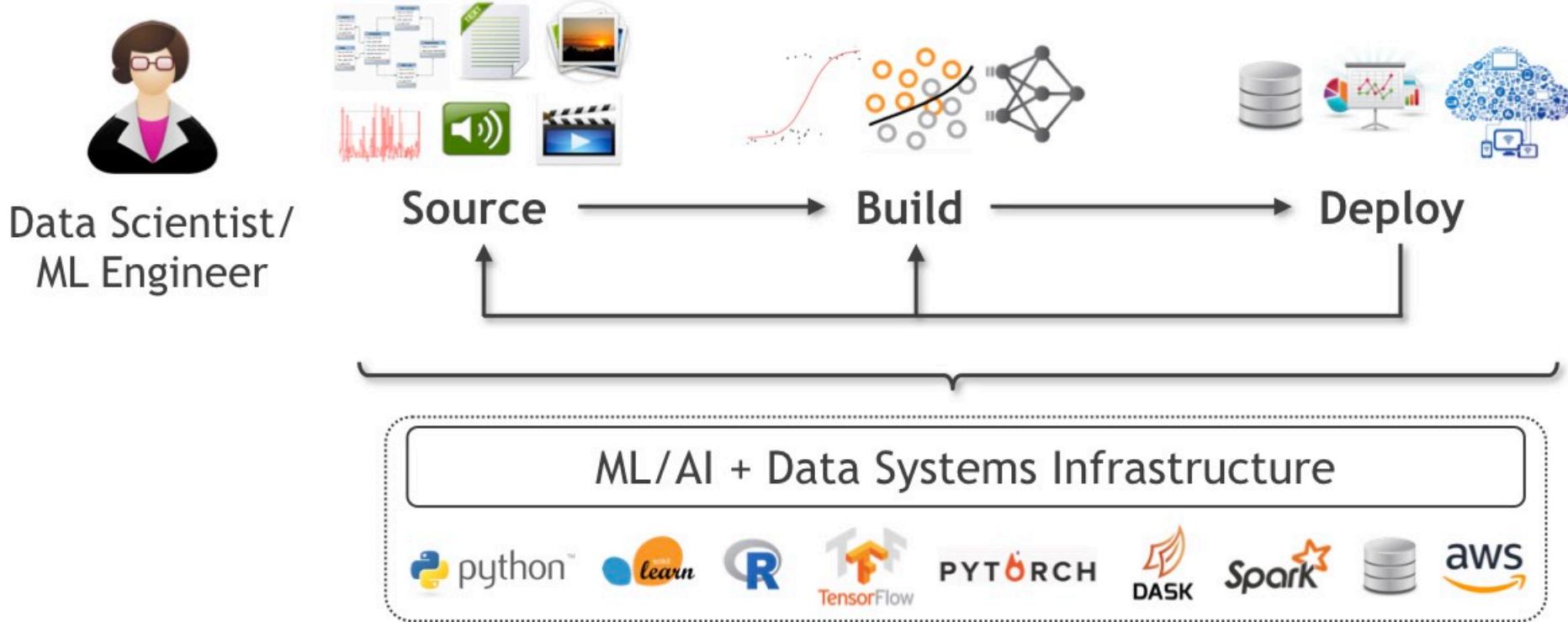
$$\text{Democratization} = \text{System Efficiency (Lower time/costs)} + \text{Human Efficiency (Higher productivity)}$$

Practical and scalable data systems for ML analytics

Inspired by relational database systems principles

Exploit insights from *learning theory* and *optimization theory*

My Current Research



Research Approach : *Abstract* key steps + *Formalize* computation + *Automate* grunt work + *Optimize* execution

What is this course about? Why take it?

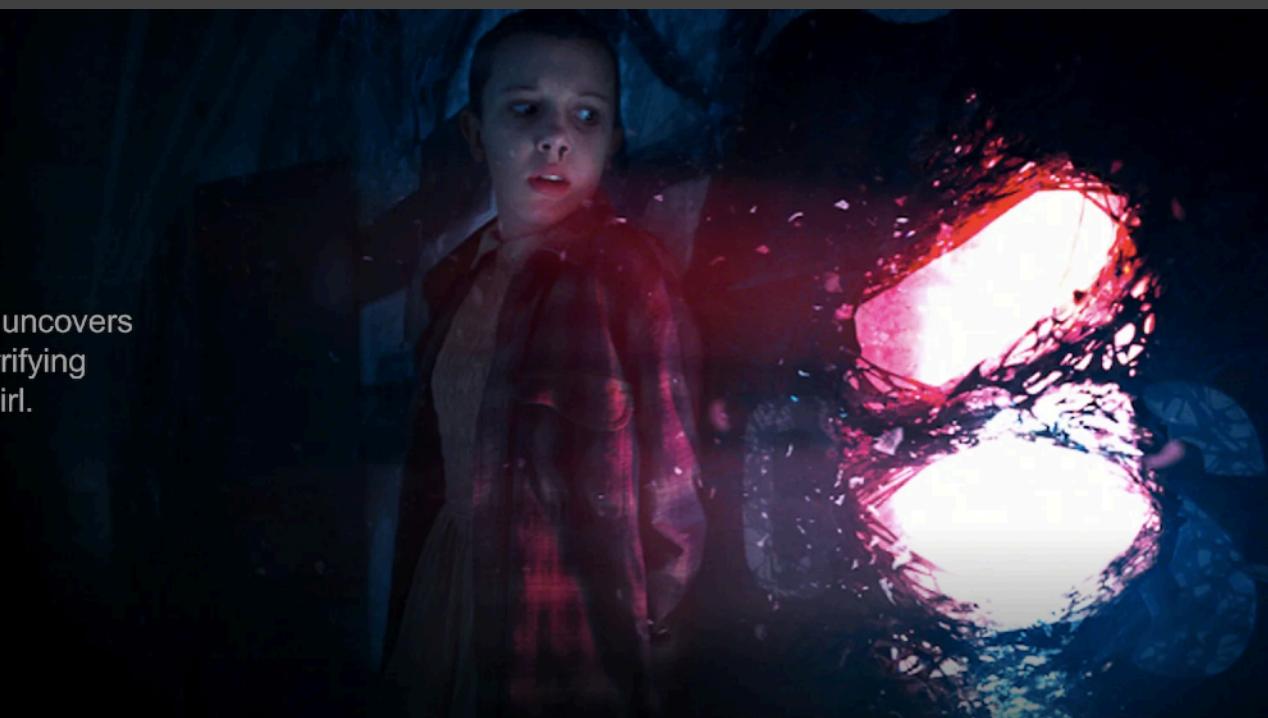
1. Netflix's “spot-on” recommendations

NETFLIX ORIGINAL **STRANGER THINGS**

95% Match 2017 2 Seasons 4K Ultra HD 5.1

When a young boy vanishes, a small town uncovers a mystery involving secret experiments, terrifying supernatural forces and one strange little girl.

Winona Ryder, David Harbour, Matthew Modine
TV Shows, TV Sci-Fi & Fantasy, Teen TV Shows



Popular on Netflix



Recently Watched



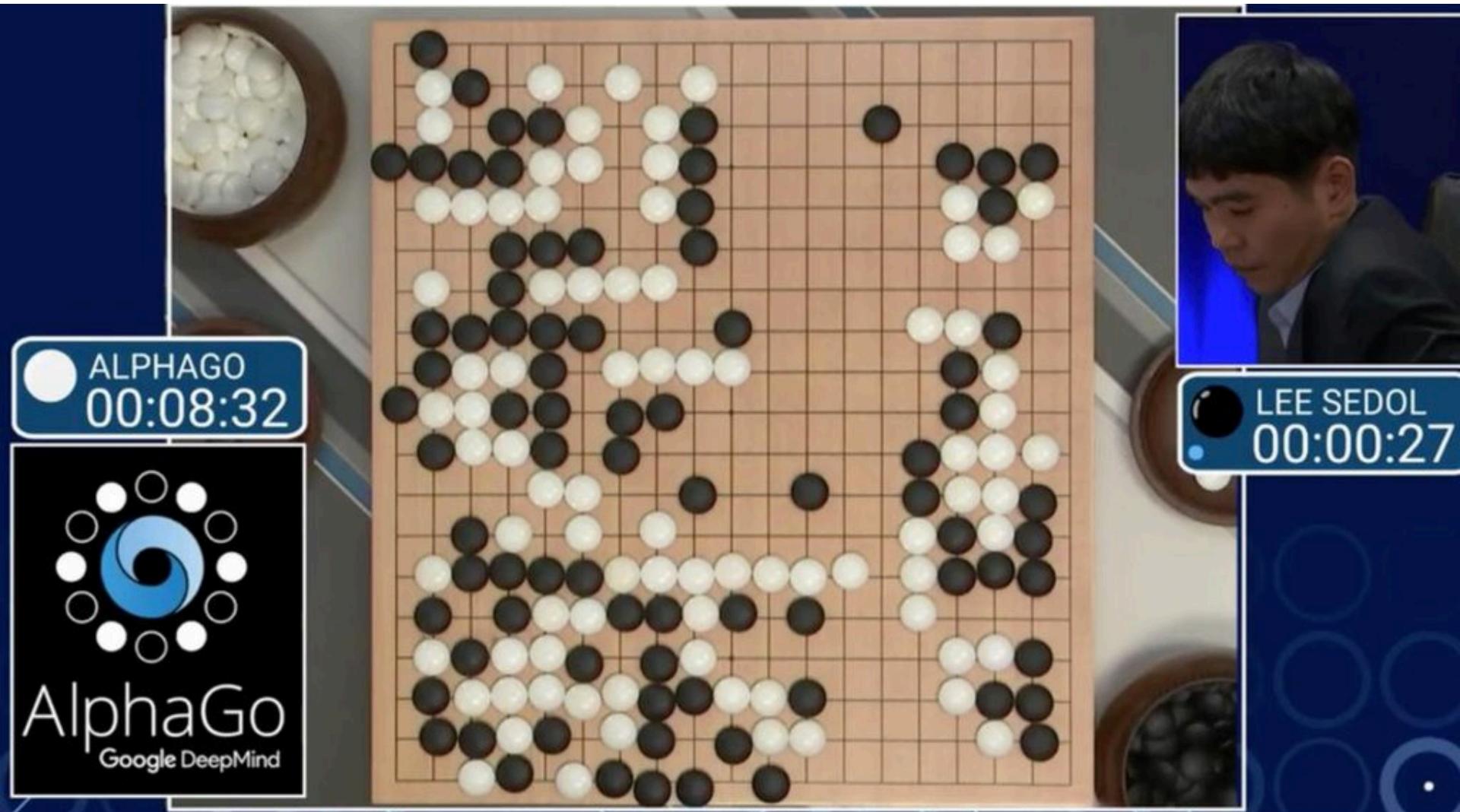
How does Netflix know that?

Large datasets + Machine learning!



Log all user behavior (views, clicks, pauses, searches, etc.)
Recommender systems apply ML to TBs of data from all users and movies to deliver a tailored experience

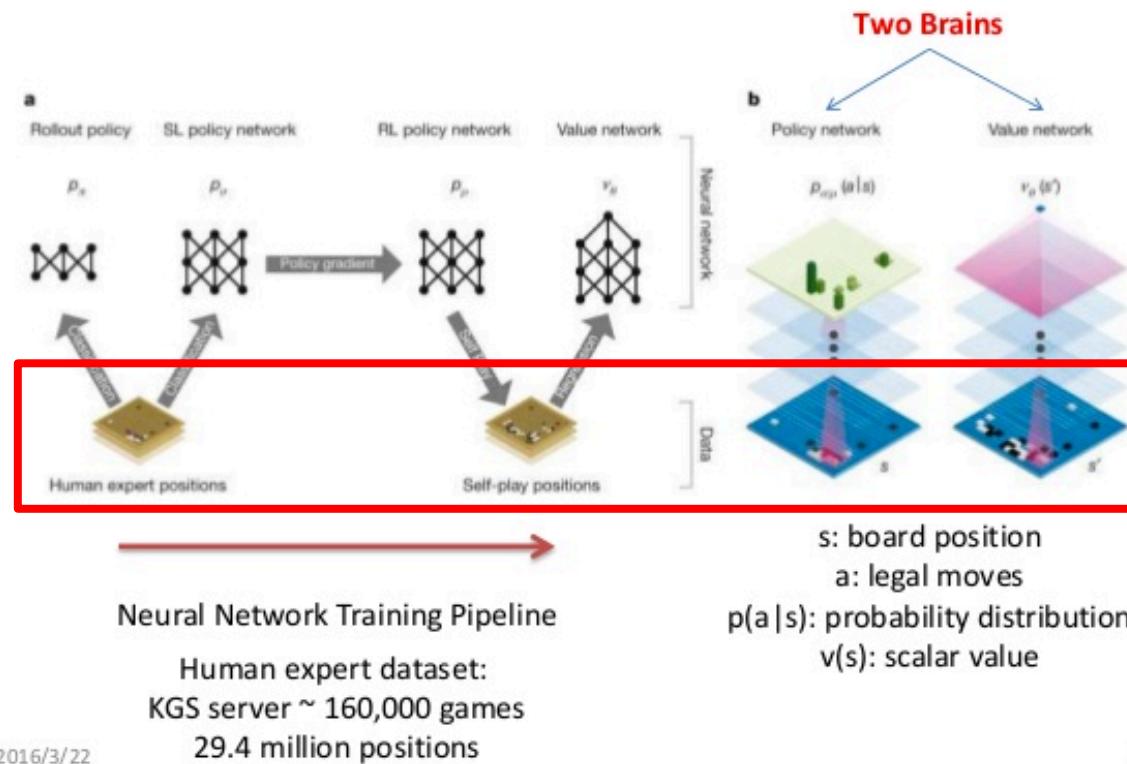
2. AlphaGo defeats human champion!



How did AlphaGo achieve that?

Breakthrough powered by deep learning

Architecture of AlphaGo



Deep CNNs to visually process board status in plays

3. The brave new era of LLMs!



I'd like to visualize the dataset to understand the class distribution and identify class imbalance. Can you help me with the visualization?



Sure, I can help you with that. One way to visualize the class distribution in the sklearn20newsgroups dataset is to create a histogram or a bar chart showing the number of samples for each class.



Here's an example code to visualize the class distribution using matplotlib:

```
python
import matplotlib.pyplot as plt

# Load the dataset
newsgroups_train = fetch_20newsgroups(subset='train', shuffle=True)

# Count the number of samples for each class
class_counts = {}
for label in newsgroups_train['target']:
    class_name = newsgroups_train['newsgroup_names'][label]
    if class_name not in class_counts:
        class_counts[class_name] = 1
    else:
        class_counts[class_name] += 1

# Create a bar chart
plt.bar(class_counts.keys(), class_counts.values())
plt.xlabel('Newsgroup')
plt.ylabel('Number of Samples')
plt.title('Class Distribution')

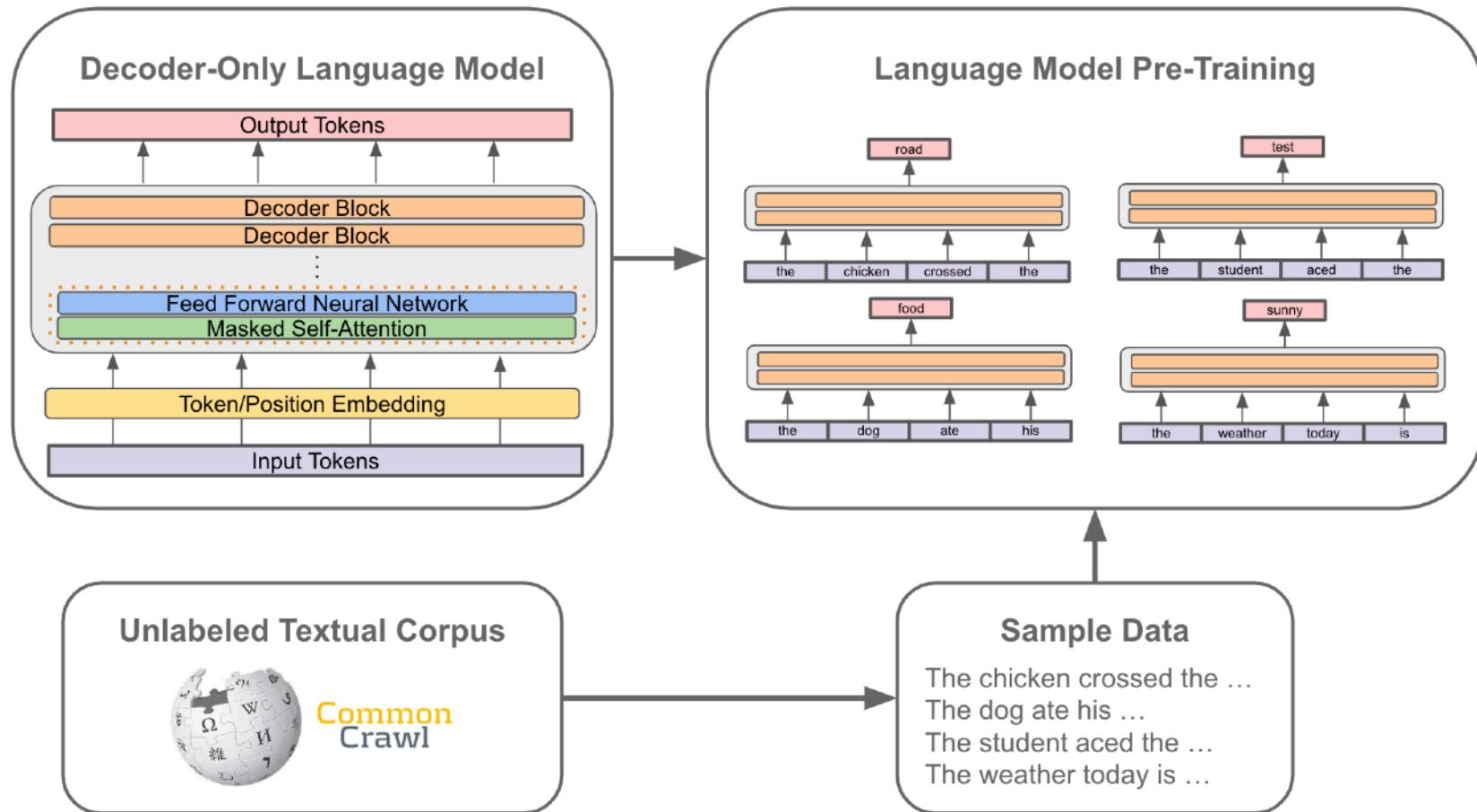
# Show the plot
plt.show()
```

Copy code

Regenerate response

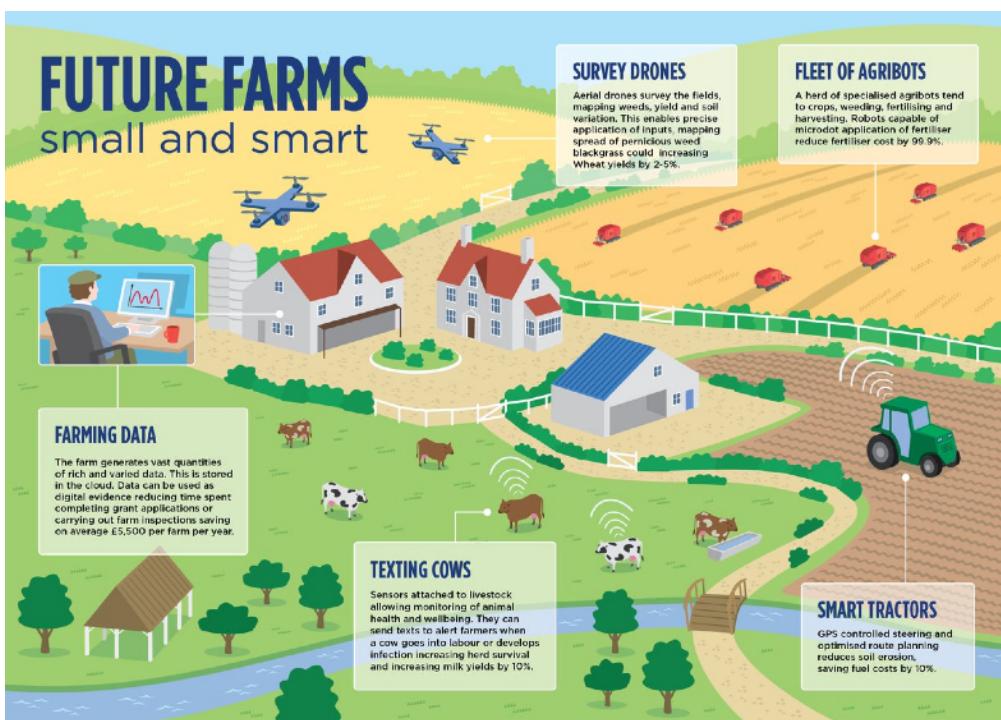
How did ChatGPT learn to write that?

“Pre-training” with mass self-supervision

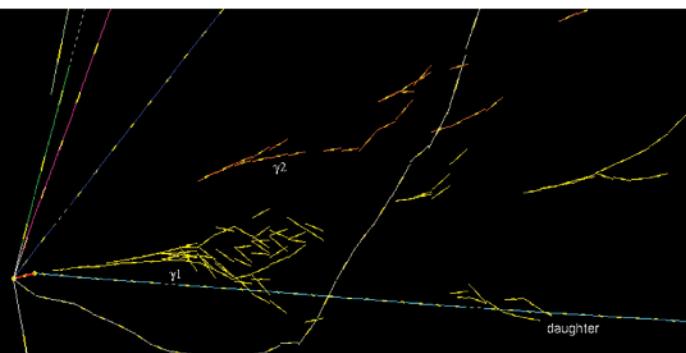
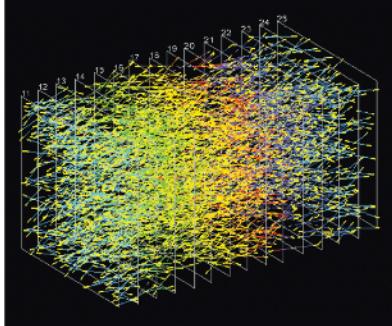
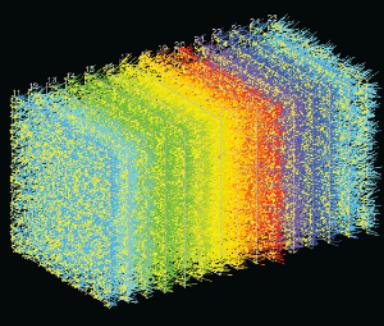


100s of billions of parameters + PBs of data

Innumerable “enterprise” applications



“Domain sciences” and healthcare tech
are also becoming data+ML intensive

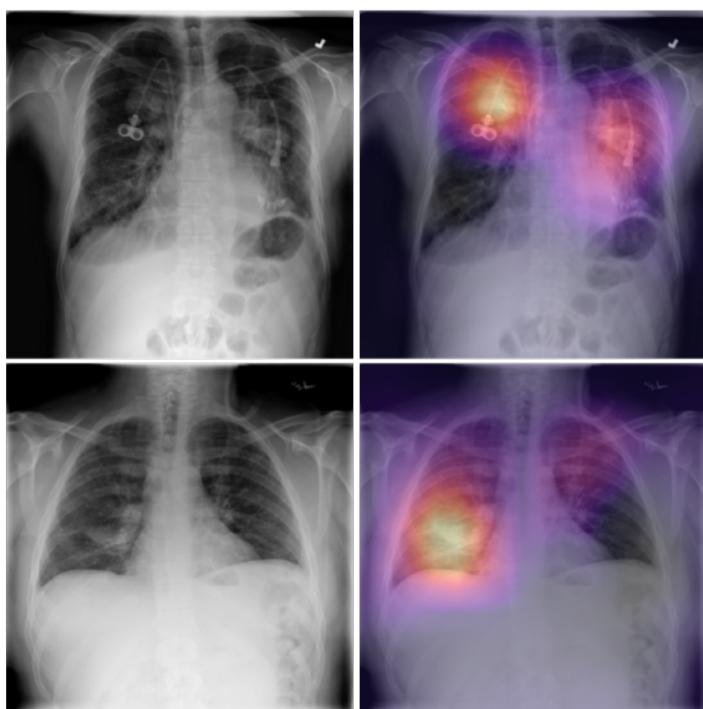


This is Data Release 16.

Data Surveys Instruments



The Sloan Digital Sky Survey: Mapping the Universe



Software systems for ML over large and complex datasets are now critical for digital applications in many domains

The Age of “Big Data”/“Data Science”

The New York Times

SundayReview | NEWS ANALYSIS

The Age **Forbes** / Entrepreneurs

By STEVE LOHR F

MAR 25, 2015 @ 7:33 PM 4,407 VIEWS



Email



Share



Tweet



Save

Josh Steimle, CON

For roughly a decade, information about Big Data. The IDC industry will experience by 2018. What this

Forbes

Drowning In Big Data - Finding Insight In A Digital DATA Josh Steimle, CON by Thomas H. Davenport and D.J. Patil FROM THE OCTOBER 2012 ISSUE

SUMMARY SAVE SHARE COMMENT 5 TEXT SIZE PRINT BUY COPIES \$8.95



Harvard Business Review

When Jonathan Goldman arrived for work in June 2006 at LinkedIn, the business networking site, the place still felt like a start-up. The company had just under 8 million accounts, and the number was growing quickly as existing members invited their friends and colleagues to join. But users weren't seeking out connections with the people who were already on the site at the rate executives had expected. Something was apparently missing in the social experience. As one LinkedIn manager put it, "It was like arriving at a conference reception and realizing you don't know anyone. So you just stand in the corner sipping your drink—

But what more is there to it than just taking a bunch of ML/AI courses?

Academic ML 101

Generalized Linear Models (GLMs); from statistics

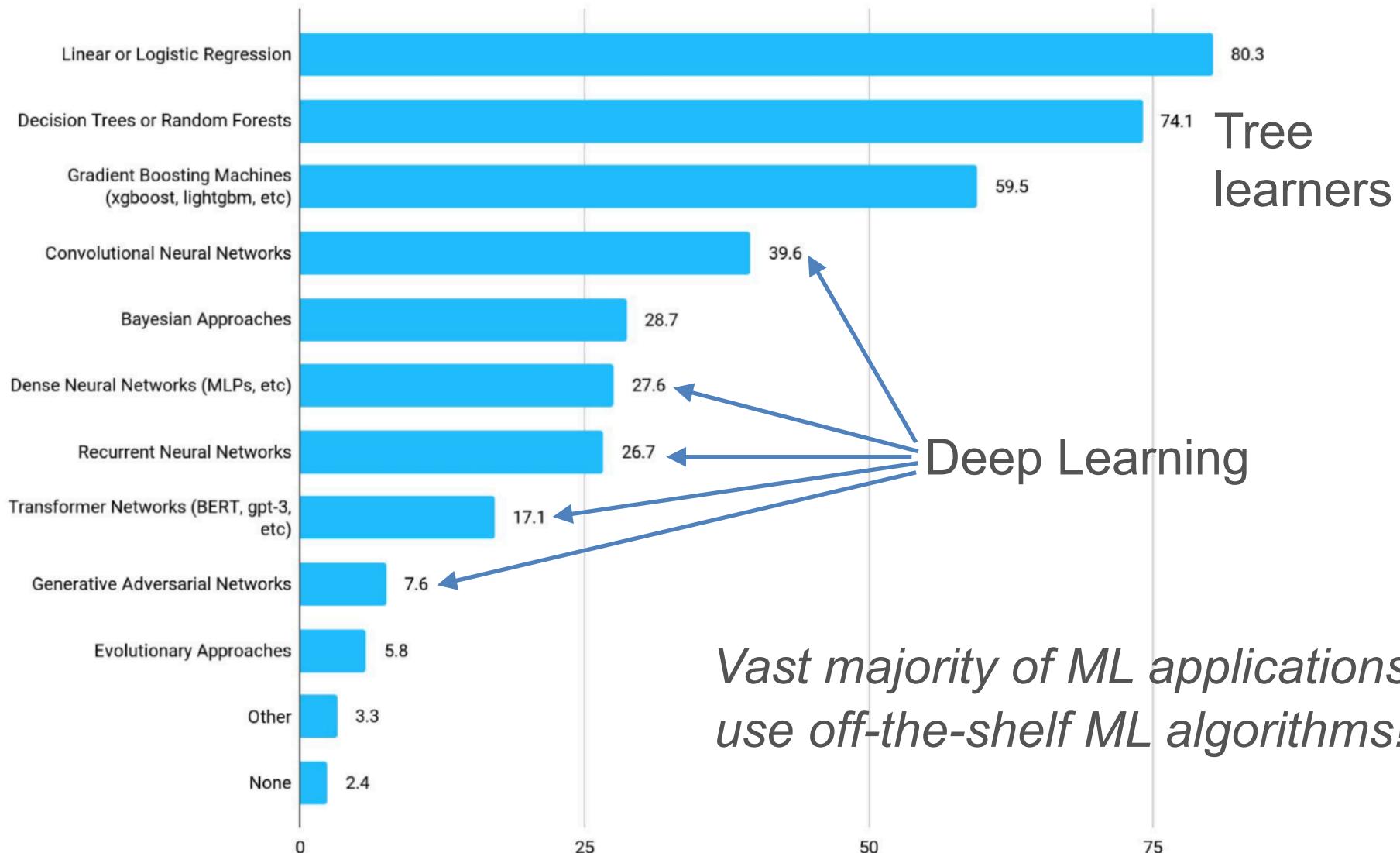
Bayesian Networks; inspired by causal reasoning

Decision Tree-based: CART, Random Forest, Gradient-Boosted Trees (GBT), etc.; inspired by symbolic logic

Support Vector Machines (SVMs); inspired by psychology

Artificial Neural Networks (ANNs): Multi-Layer Perceptrons (MLPs), Convolutional NNs (CNNs), Recurrent NNs (RNNs), Transformers, etc.; inspired by brain neuroscience

Real-World ML 101



The rise of “foundational” LLMs is causing another massive wave of disruption in NLP/AI applications!

Real-World ML 101

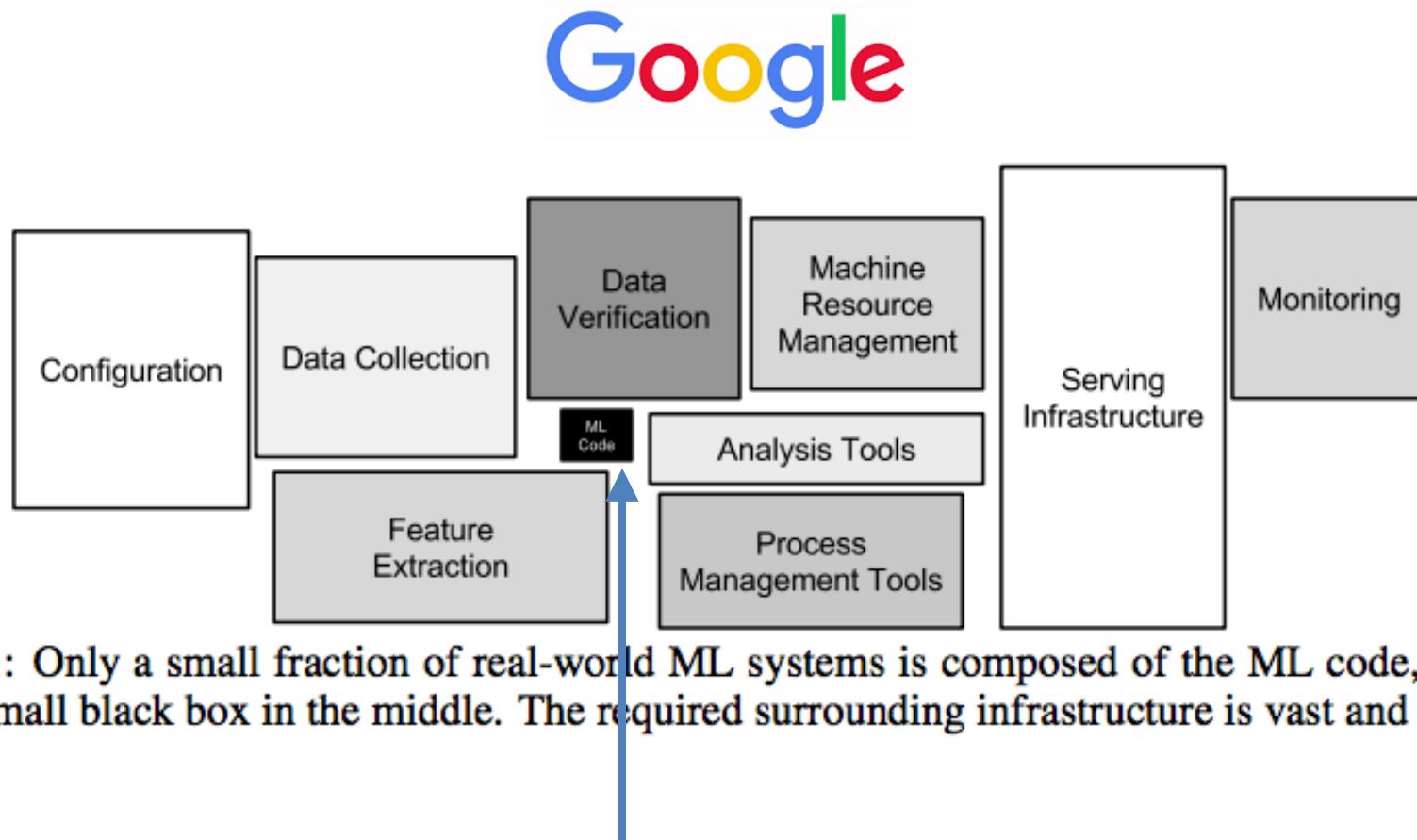
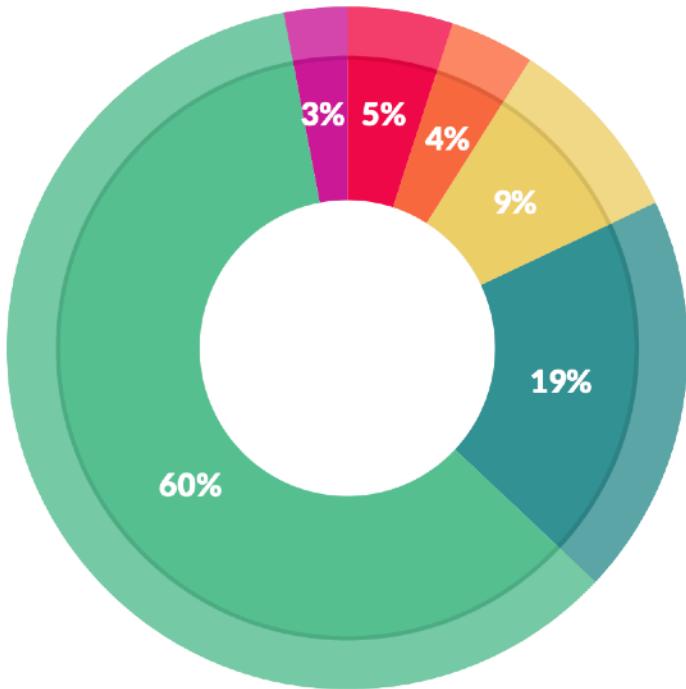


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

Almost all of your ML / AI courses put together! :)

Real-World ML 101



What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

80% of ML users' time/effort (often more) spent on data issues!

Real-World ML 101

Uber

Meet Michelangelo: Uber's Machine Learning Platform

Jeremy Hermann and Mike Del Balso

September 5, 2017

“Building and managing data pipelines is typically one of the most costly pieces of a complete machine learning solution.”

Rules of Machine Learning:
Best Practices for ML Engineering

“Do machine learning like the great engineer you are, not like the great machine learning expert you aren’t.”

Google

Real-World ML 101

Transitioning from Academic Machine Learning to AI in Industry



Jeremy Karnowski [Follow](#)
Aug 9, 2017 · 6 min read



[Jeremy Karnowski](#) and [Emmanuel Ameisen](#), Insight AI

1. System design
2. Structured ML modules
3. Software testing
4. Integrating with data infrastructure
5. Model serving

Real-World ML 101

The Rise of “Data-Centric” AI



Andrew Ng  @AndrewYNg

Announcing the Data-Centric AI competition! I'm excited to invite you to participate in this new competition format, and see how you can improve an AI system only by refining the data it depends on!

bit.ly/3vwE56i

DeepLearning.AI | LANDING AI

Data-Centric AI Competition

June 17 - Sept 4, 2021

LANDING AI and DeepLearning.AI

The Data-Centric AI “Movement”



Tools & methodologies for accelerating open-source dataset iteration:

- Tools that quantify and accelerate time to source and prepare high quality data
- Tools that ensure that the data is labeled consistently, such as label consensus
- Tools that make improving data quality more systematic
- Tools that automate the creation of high quality supervised learning training data from low quality resources, such as forced alignment in speech recognition
- Tools that produce consistent and low noise data samples, or remove labeling noise or inconsistencies from existing data
- Tools for controlling what goes into the dataset and for making high level edits efficiently to very large datasets, e.g. adding new words, languages, or accents to speech datasets with thousands of hours
- Search methods for finding suitably licensed datasets based on public resources
- Tools for creating training datasets for small data problems, or for rare classes in the long tail of big data problems
- Tools for timely incorporation of feedback from production systems into datasets
- Tools for understanding dataset coverage of important classes, and editing them to cover newly identified important cases
- Dataset importers that allow easy combination and composition of existing datasets
- Dataset exporters that make the data consumable for models and interface with model training and inference systems such as webdataset.
- System architectures and interfaces that enable composition of dataset tools such as, MLCube, Docker, Airflow

New Venues on Data/Systems for ML



[About](#) | [Schedule](#) | [Important Dates](#) | [CfP](#) | [Topics](#) | [Submission](#) | [Accepted Papers](#) | [Invited Speakers](#) | [People](#)

The DEEM workshop will be held on Sunday, June 18th, in conjunction with [SIGMOD/PODS 2023](#). The workshop will be held in hybrid (in-person and virtual) form. DEEM brings together researchers and practitioners at the intersection of applied machine learning, data management and systems research, with the goal to discuss the arising data management issues in ML application scenarios.



SIGMOD 2022 CALL FOR RESEARCH PAPERS

Data Science Track

We invite the submission of original research in data science targeting the entire data life cycle of real applications. This data life cycle encompasses databases/data management/data systems/data engineering often leveraging statistical, Machine Learning and Artificial Intelligence methods and using massive and heterogeneous collections of potentially messy datasets. Data science papers study phenomena at scales and granularities never before possible. Such papers are expected to focus on data-intensive components of data science pipelines; and solve problems in areas of interest to the

Fifth Conference on Machine Learning and Systems

- Year (2022) ▾
 - [Help](#) ▾
 - [My Registrations](#)
 - [Profile](#) ▾
 - [Contact Us](#)
 - [Sponsor Info](#)
 - [Conflicts of Interest](#)
 - [Code of Conduct](#)
 - [Proceedings](#)
- ## MLSys



Alon Halevy, Arun Kumar and Nesime Tatbul

FEBRUARY 10,
2020

Dates Calls Organizers

Santa Clara Convention Center

Mon Apr 11th through Thu the 14th, 2022

Registration

[Pricing](#) Register starting Feb 06 01 PM PST

[Registration Cancellation Policy](#)

Schedule

[Video Library 2021](#)

[Tweet](#)

Conference Overview

The Conference on Machine Learning and Systems targets research at the intersection of machine learning and systems. The conference aims to elicit new connections amongst these fields, including identifying best practices and design principles for learning systems, as well as developing novel learning methods and theory tailored to practical machine learning workflows. Topics include:

- Efficient model training, inference, and serving
- Distributed and parallel learning algorithms
- Privacy and security for ML applications
- Testing, debugging, and monitoring of ML applications
- Fairness, interpretability and explainability for ML applications
- Data preparation, feature selection, and feature extraction

SCALABLE DATA SCIENCE: A NEW RESEARCH TRACK CATEGORY AT PVLDB VOL 14 / VLDB 2021

≡ Uncategorized

This post introduces and explains the newly created category of “Scalable Data Science” within the Research Track of PVLDB. This category comes into effect for volume 14, i.e., submissions starting April 1, 2020, which will be evaluated by the Review Board of PVLDB vol 14 for presentation at VLDB 2021.

The Growth of Data Science



HOME

PROGRAM

ATTENDING

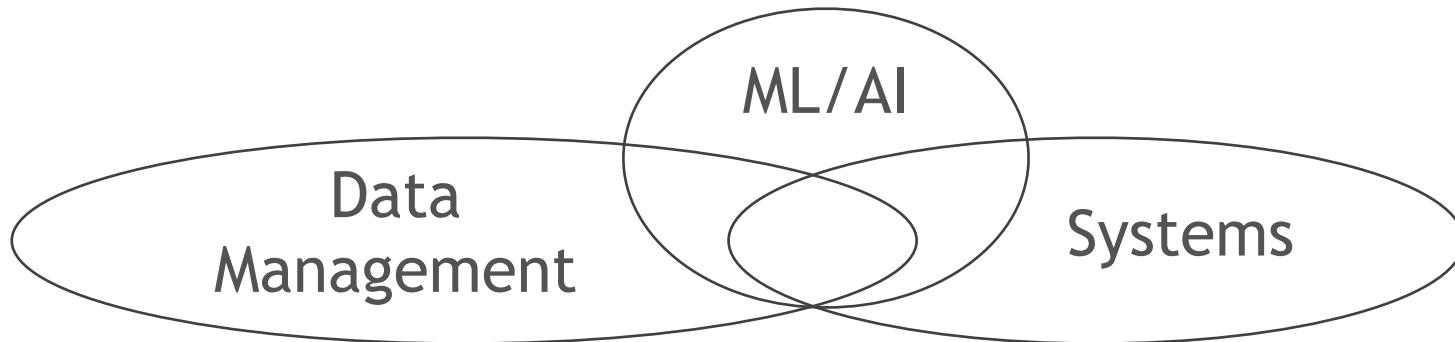
CALLS

KDD

SPONSORS

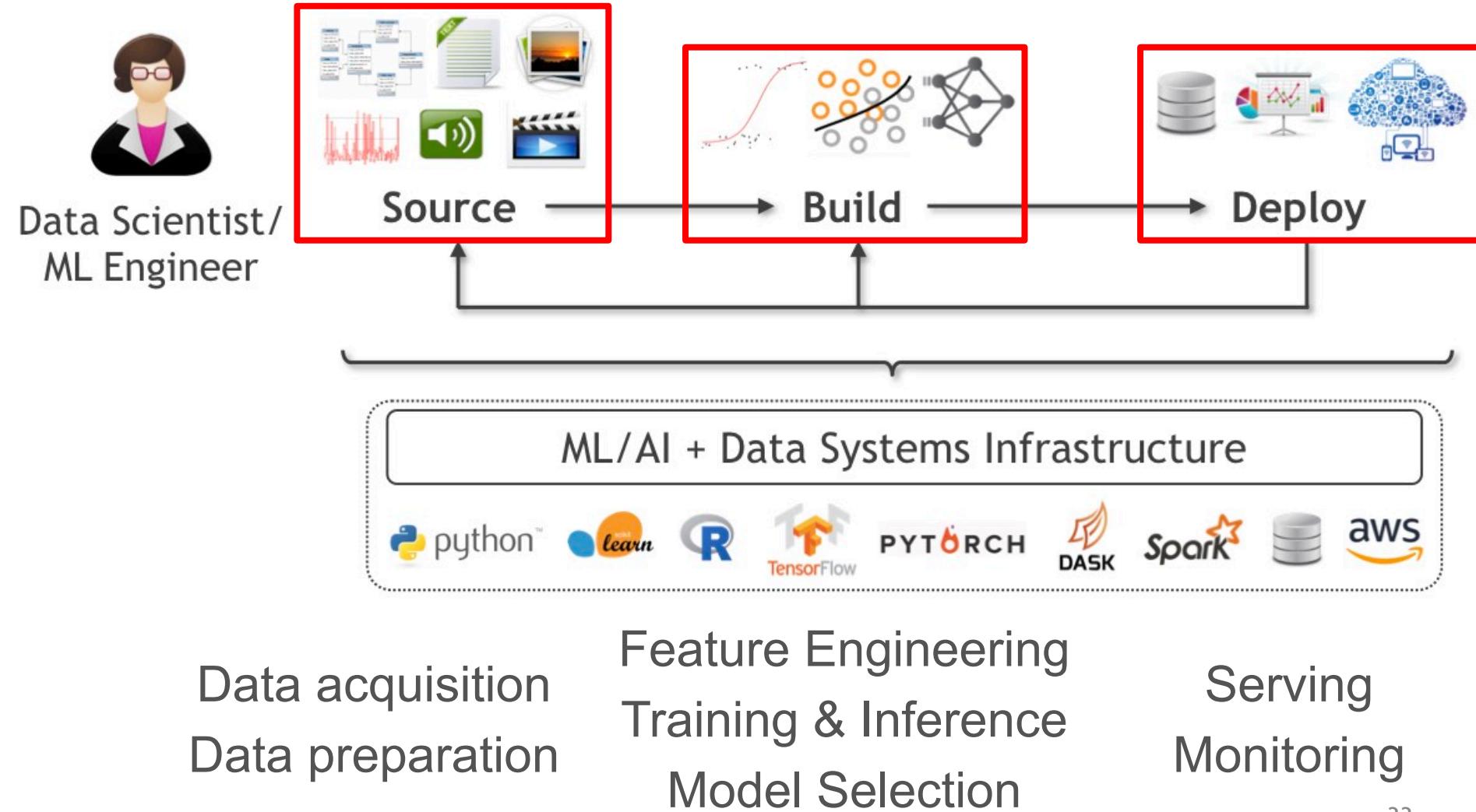
ORGANIZERS

CSE 234 will get you to think about the data systems that power this new boom of ML/AI



1. “**Data ...**”: How to organize, query, scale, and manage the analysis of large and complex datasets?
2. “... **Systems** ...”: How to make the most effective use of all machine resources?
3. “... **for ML**”:
 - 3.1. **Source**: Application’s raw data -> “ML-ready” data
 - 3.2. **Build**: “ML-ready” data -> Prediction pipelines
 - 3.3. **Deploy**: Productionize prediction pipelines

The Lifecycle of ML-based Analytics



Now for the logistics ...

Prerequisites

- ❖ A course on **ML algorithms**, e.g., CSE 151.
- ❖ A course on either **DBMS internals** (e.g., CSE 132C or CSE 232A) or **operating systems** (e.g., CSE 120).
- ❖ The above courses could have been taken at UCSD or elsewhere.
- ❖ Industrial or substantial project experiences on these topics might suffice in place of such courses. Email me if you are not sure if you satisfy the prerequisites.

<https://cseweb.ucsd.edu/classes/wi24/cse234-a/>

Components and Grading

- ❖ **Paper Reviews:** **16%** ($8 \times 2\%$); best 8 of 9
- ❖ **Quiz:** **9%**
- ❖ **Midterm Exam:** **20%**
- ❖ **Cumulative Final Exam:** **50%**
- ❖ **Peer Instruction Activities:** **5%** ($5 \times 1\%$); best 5 of 6

<https://cseweb.ucsd.edu/classes/wi24/cse234-a/>

Grading Scheme

Hybrid of relative and absolute; grade is the **better** of the two

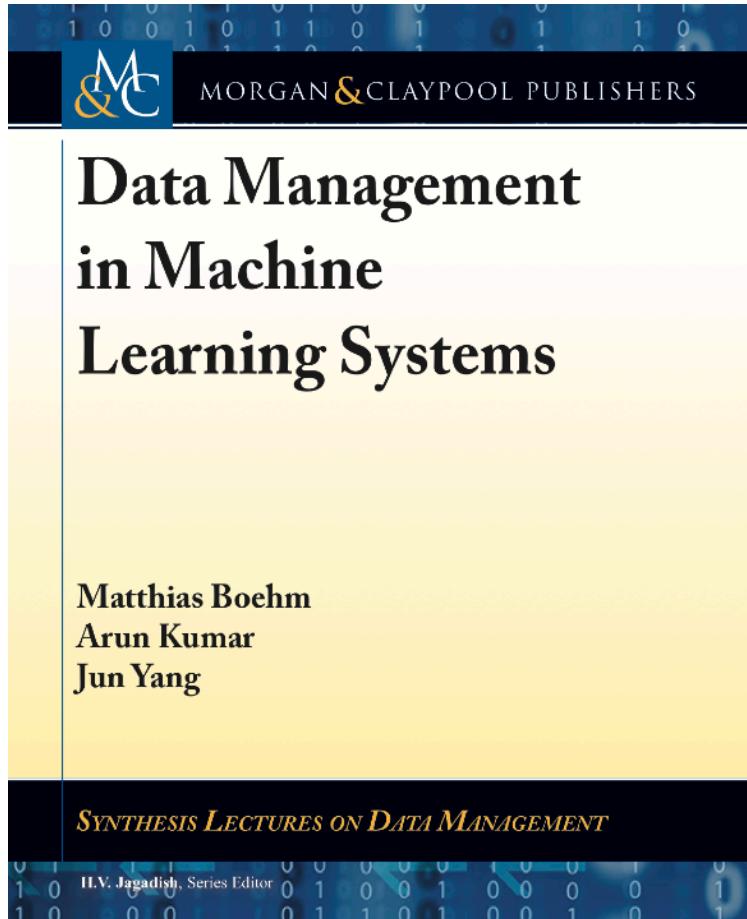
Grade	Relative Bin (Use strictest)	Absolute Cutoff (>=)
A+	Highest 5%	95
A	Next 15% (5-20)	90
A-	Next 15% (20-35)	85
B+	Next 15% (35-50)	80
B	Next 15% (50-65)	75
B-	Next 5% (65-75)	70
C+	Next 5% (75-80)	65
C	Next 5% (80-85)	60
C-	Next 5% (85-90)	55
D	Next 5% (90-95)	50
F	Lowest 5%	< 50

Example: Score 86 and 60%ile; Rel: B+; Abs: A-; so, A-

Tentative Course Schedule

Week	Topic
1	Introduction, ML Lifecycle Overview, and Basics
1-3	Topic 1: Classical ML Training at Scale
Build	Topic 2: DL Systems (incl. LLM & DLRM Systems)
5	Review Discussion 1 on Fri, Feb 16
6	Midterm Exam on Tue, Feb 20
7-8	Topic 3: Model Selection Systems
Source	Topic 4: Data-Centric Issues in ML/AI
9-10	Topic 5: ML Platforms and Feature Stores
Deploy	Topic 6: MLOps and LLMOps
10	Review Discussion 2 on Fri, Mar 15
11	Final Exam on Thu, Mar 21

Suggested Textbook



Aka “MLSys Book”
PDF is free via UCSD VPN
Also check out our library

Day-to-day Logistics

- ❖ Lectures, quiz, exams, peer instruction activities, and my office hours are all **in-person only**
 - ❖ Standard policy on makeup for quiz/exams
 - ❖ Attending lectures/discussions live is **optional**
 - ❖ Automatically **podcast** to podcast.ucsd.edu
 - ❖ Discussion slot used only **twice**; before each exam
- ❖ **Piazza** for all announcements and asynchronous Q&A
 - ❖ Do help out your peers by discussing questions
- ❖ **Google Forms** for Peer Instruction Activities

<https://cseweb.ucsd.edu/classes/wi24/cse234-a/>

Course Administrivia

- ❖ **Lectures:** TuTh 5-6:20pm PT @ WLH 2005
 - ❖ Podcast always available for async. viewing
- ❖ **Instructor:** Arun Kumar; akk018 [at] ucsd.edu
 - ❖ Office hours: Fri 1:30-2:30pm PT @ 3218 CSE
- ❖ **Piazza:** <https://piazza.com/ucsd/winter2024/cse234>
- ❖ **Teaching Assistants:**
 - ❖ Kabir Nagrecha and Aditya Gulati
- ❖ See course website for TA contact info. and OHs

<https://cseweb.ucsd.edu/classes/wi24/cse234-a/>

General Dos and Do NOTs

Do:

- ❖ Come to lectures in person as much as possible, ask questions, and participate actively in class discussions
- ❖ Review slides and podcast videos by yourself
- ❖ Follow all announcements and participate in discussions on Piazza; pitch in to answer peers' questions
- ❖ Use "CSE234:" as subject prefix for all emails to me/TAs

Do NOT:

- ❖ Record anything in class without explicit permission from me and all other participants
- ❖ Harass, intimidate, or intentionally talk over other students
- ❖ Violate academic integrity on any graded components

So, what is an “ML system” anyway?

ML Systems

Q: What is a Machine Learning (ML) System?

- ❖ A data processing system (aka *data system*) for mathematically advanced data analysis operations, i.e., inferential, predictive, or generative:
 - ❖ Statistical analysis; ML, deep learning (DL); data mining (domain-specific applied ML + feature eng.)
 - ❖ *High-level APIs* to express ML computations over (large) datasets
 - ❖ *Execution engine* to run ML computations efficiently

Categorizing ML Systems

❖ Orthogonal Dimensions of Categorization:

1. **Scalability:** In-memory libraries vs. Scalable ML system (works on larger-than-memory datasets)
2. **Target Workloads:** General ML library vs. Decision tree-oriented vs. Deep learning-oriented, etc.
3. **Implementation Reuse:** Layered on top of scalable data system vs. Custom from-scratch framework

Major Existing ML Systems

General ML Libraries:

In-memory:



Disk-based files:



Layered on RDBMS/Spark:



Cloud-native:



“AutoML” platforms:



Tree Learning Systems:



Deep Learning Systems:



LLM Systems:



Data Systems Concerns in ML

Key concerns in ML:

Q: How do “ML Systems” relate to ML?

Runtime efficiency (sometimes)

Additional key *practical* concerns in ML Systems:

ML Systems : ML :: Computer Systems : TCS
Scalability (and efficiency at scale)

Usability

Manageability

Developability

*Long-standing
concerns in the
DB systems
world!*

Q: ~~Q: What are the difficulties in building large-scale machine learning systems?~~

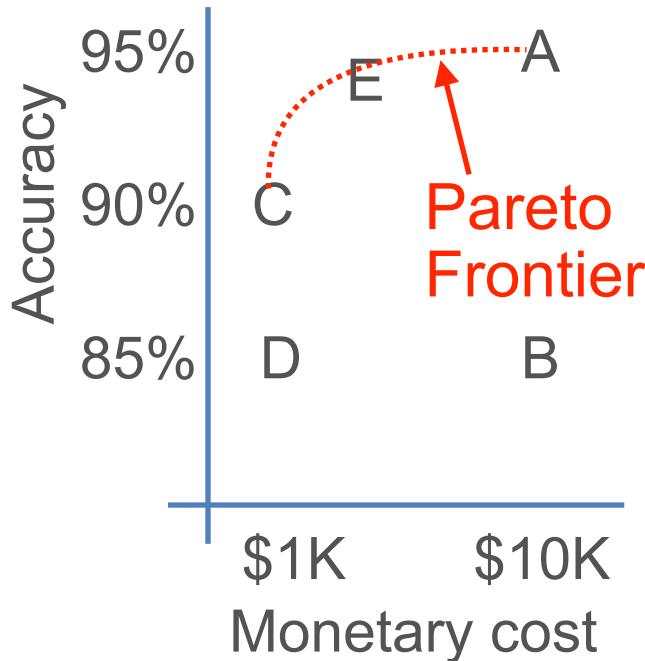
Conceptual System Stack Analogy

	Relational DB Systems	ML Systems
Theory	First-Order Logic Complexity Theory	Learning Theory Optimization Theory
Program Formalism	Relational Algebra	Tensor Algebra Gradient Descent
Program Specification	SQL	PyTorch? Scikit-learn?
Program Modification	Query Optimization	???
Execution Primitives	Parallel Relational Operator Dataflows	Depends on ML Algorithm
Hardware	CPU, GPU, FPGA, NVM, RDMA, etc.	

Real-World ML: Pareto Surfaces

Q: Suppose you are given ad click-through prediction models A, B, C, and D with accuracies of 95%, 85%, 90%, and 85%, respectively. Which one will you pick?

Q: What about now?



- ❖ Real-world ML users must grapple with multi-dimensional *Pareto surfaces*: accuracy, monetary cost, training time, scalability, inference latency, tool availability, interpretability, fairness, etc.
- ❖ *Multi-objective optimization* criteria set by application needs and business policies.

Learning Outcomes of this Course

- ❖ View ML/AI algorithms as data-intensive programs and employ systems techniques to make them scalable and fast.
- ❖ Understand the myriad data management issues in the end-to-end ML lifecycle and how to handle them in practice.
- ❖ Reason about practical tradeoffs between accuracy, scalability, efficiency, usability, cost, etc. in ML applications.
- ❖ Think critically and objectively about research in this intersectional area and maybe identify gaps in the literature.

What this course is NOT about

- ❖ NOT a course on basics of ML, databases, or systems
 - ❖ Sanity check! You must know what all these terms mean: gradient descent, decision tree, neural network, schema, query optimization, memory hierarchy, and GPU
- ❖ NOT a course on ML *algorithmics*; we focus on ML *systems*
- ❖ NOT a course on how to use/apply ML algorithms or tools

On the paper reviewing component ...

Goal of Peer Review in Research

- ❖ “Gatekeeping” for quality of publication venue
- ❖ Collation of scientific/technical knowledge of the field
- ❖ Provide constructively critical feedback to authors
- ❖ Appreciate strong efforts of authors
- ❖ Recognize/reward research novelty, creativity, depth
- ❖ Identify/support emerging research problems/areas

Goal of Paper Reviews in CSE 234

- ❖ Data+systems+ML area is still nascent, growing, and fast-evolving. Staying on top of latest major research matters for both researchers and practitioners
- ❖ Teach how to read cutting-edge research papers with a “critical thinking” mindset
- ❖ Teach how to appreciate/evaluate emerging ideas in an objective, honest, and balanced manner
- ❖ Make you take the paper readings seriously! :)
- ❖ Perhaps try to identify research gaps or extensions?

Paper Reviews in CSE 234

- ❖ 9 papers for reviewing via Google Forms
 - ❖ Only best 8 scores will be used; 16% of total score
- ❖ The review form asks for 3 main things (with length limits):
 - ❖ Summary of the problem and key ideas
 - ❖ 3 major strong points
 - ❖ 3 major limitations
- ❖ Discussion with your peers about the papers is acceptable
 - ❖ But final submitted review must be **entirely your own**
 - ❖ Otherwise it will be an Academic Integrity violation
 - ❖ OK to use ChatGPT/Bard/AI writing assistance tools

<https://cseweb.ucsd.edu/classes/wi24/cse234-a/>

Paper Reviews in CSE 234

- ❖ TA will evaluate your reviews; 2-point criteria:
 - ❖ **Thoroughness:** Does it cover key strong/weak points?
 - ❖ **Exposition:** Is it constructive and well written?
- ❖ Scores will be posted on Canvas Gradebook
- ❖ If you lose any points, TA will post individual feedback
- ❖ Helpful tips on how to read and evaluate research papers:
 - ❖ Keshav's writeup: [PDF link](#)
 - ❖ Mitzenmacher's writeup: [PDF link](#)

<https://cseweb.ucsd.edu/classes/wi24/cse234-a/>

Sample paper to review from past

Towards a Unified Architecture for in-RDBMS Analytics

Xixuan Feng

Arun Kumar

Benjamin Recht

Christopher Ré

Department of Computer Sciences
University of Wisconsin-Madison

{xfeng, arun, brecht, chrisre}@cs.wisc.edu

ABSTRACT

The increasing use of statistical data analysis in enterprise applications has created an arms race among database vendors to offer ever more sophisticated in-database analytics. One challenge in this race is that each new statistical technique must be implemented from scratch in the RDBMS,

late 1990s and early 2000s, this brought a wave of data mining toolkits into the RDBMS. Several major vendors are again making an effort toward sophisticated in-database analytics with both open source efforts, e.g., the MADlib platform [17], and several projects at major database vendors. In our conversations with engineers from Oracle [38] and EMC Greenplum [91], we learned that a key bottleneck in

ACM SIGMOD 2012

Project Bismarck

(Topic: Scaling ML to data stored in RDBMSs)

My 3-line summary

- ❖ (Setting) Integration of ML procedures with RDBMSs is often used for large-scale analytics over RDBMS-resident data without needing to move/copy data.
- ❖ (Problem) Redesigning and reimplementing every individual ML procedures for in-RDBMS execution from scratch is a long, tedious, and wasteful development process.
- ❖ (Approach) This paper proposes a unified abstraction and software architecture for a large class of ML procedures based on incremental gradient descent (IGD) that is implementable using the existing common RDBMS abstraction of user-defined aggregate (UDA).

Sample good summary from S1

- ❖ Each RDBMS has its own tools for ML problems. Usually they have different tools for different ML algorithms, which makes them difficult to maintain and cause tons of development overhead.
- ❖ Since most of the ML techniques can be represented as algorithms solving convex programming problem, i.e. minimizing some convex cost function, it is possible to use one single architecture to unify all of them.
- ❖ The authors proposed a unified architecture based on IGD and UDA allowing developer to adapt it to different ML problems with little development overhead.
- ❖ The authors also proposed a modified reservoir sampling technique called MRS. What is more, the authors studied the influence of data ordering and parallelized BISMARCK.

Sample good strong points from S1

- ❖ Generality. A single framework solves multiple problems, making maintaining and development easier. The reuse of codes are drastically improved. One optimization for BISMARCK means optimizations for ALL.
- ❖ UDA-based. It is very easy to re-implement for different RDBMSes.
- ❖ Efficiency. It is only a little slower than an common aggregation and it out-performs many of the built-in tools provided by RDBMSes.

Sample good weak points from S1

- ❖ Generality means loss of speciality. Using IGD for all convex problems may cause a consequence that some of the ML techniques can be more efficiently solved by some other specific techniques. This is the tradeoff.
- ❖ The limitations of IGD are also limitations of BISMARCK: internally sequential, hard to tune. Problems [that] cannot be solved by IGD cannot use BISMARCK.
- ❖ RDBMSes have more to offer. BISMARCK only utilizes the UDA of databases. There is still space for optimization. Especially for distributed databases.

Other good strong points from class

- ❖ New optimization strategies can be tested using Bismarck without having to make changes for all analytic techniques.
- ❖ The paper honestly studies its overhead as well as thoroughly compares the result of integrations in three different RDBMS.
- ❖ The experiments are compelling. Their use of wall-clock-time measurements, and benchmarks against native UDA speeds, presents a strong case.
- ❖ The organization of the paper is very helpful such that reader who has little knowledge in this area can read and grasp the main concepts. The authors take real-world examples where necessary to explain the concepts which is helpful.

Other good weak points from class

- ❖ The theoretical justification of why IGD is essentially commutative and algebraic lacked depth. The claim that averaging models trained on different segments of the data would lead to convergence seemed dubious.
- ❖ Another limitation is that this architecture is designed for single node RDBMS. Currently, more applications move to cloud services and use distributed database or frameworks such as Hadoop and Spark.
- ❖ The assumption that the data is static might not be upheld in a production environment and Bismarck has no provision to support online learning.
- ❖ Strong assumption that the state (model parameters) fit in RAM.