# AI-Systems
# Big Ideas

Joseph E. Gonzalez
Co-director of the RISE Lab
jegonzal@cs.berkeley.edu

# Logistics

- Go to website: https://ucbrise.github.io/cs294-ai-sys-fa19/
- Make sure you are on the course Piazza
  - Needed for announcements
- Signup for **3 discussion** slots as **different roles here**:

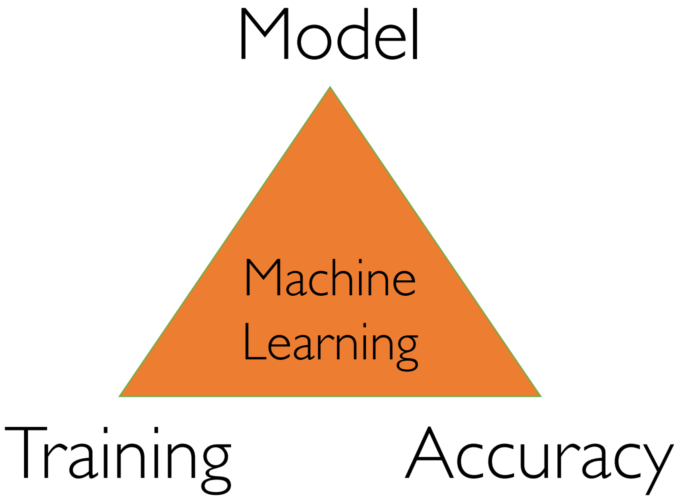
https://tinyurl.com/aisysfa19signup

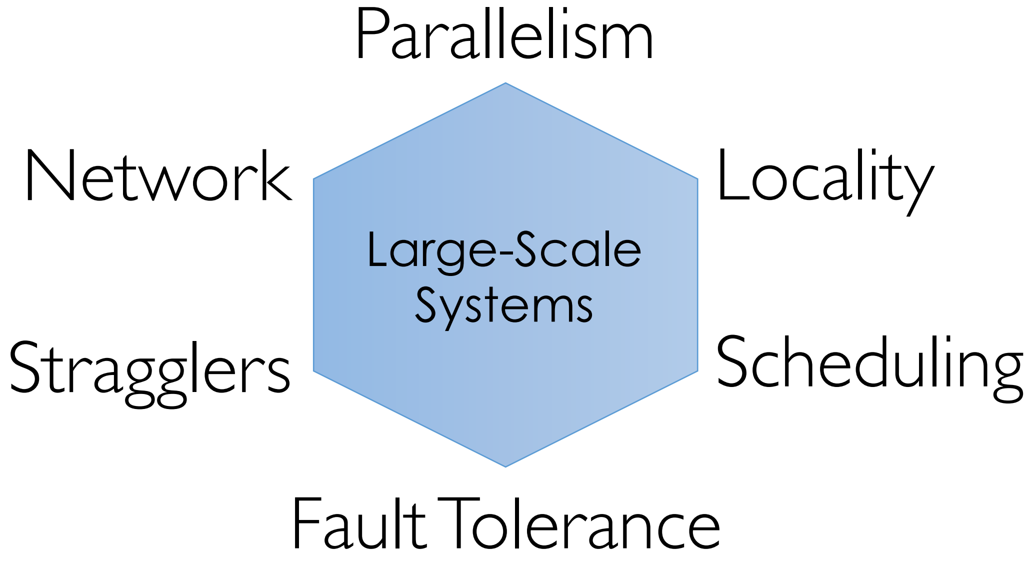
# Logistics

- Limited to 45 spots due to maximum room occupancy
  - Unable to change lecture room
- Please do the fist few assignments
  - I will ask people to drop the class if they do not
- If you are planning to drop the class do it soon
- I plan to teach the class again next Fall

# Recap

# Design Complexity



# Managing Complexity Through Abstraction

Identify common patterns

**Learning Algorithm**

Common Patterns

Define a narrow interface

**Abstraction (API)**

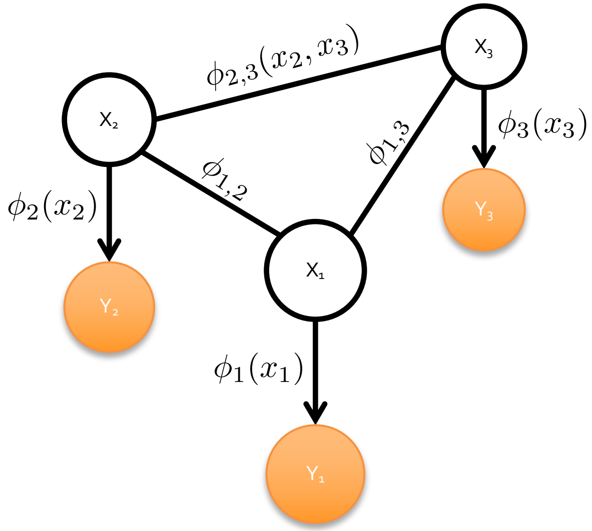
Exploit limited abstraction to address system design challenges

**System**

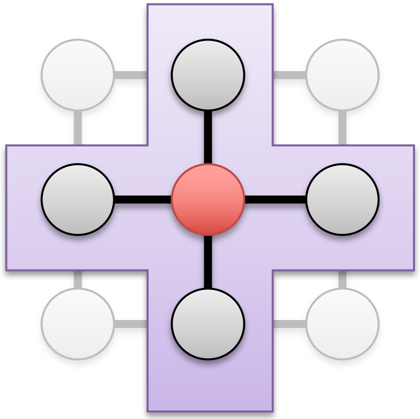
1. Parallelism
2. Data Locality
3. Network
4. Scheduling
5. Fault-tolerance
6. Stragglers

# PhD in Machine Learning from CMU 2013

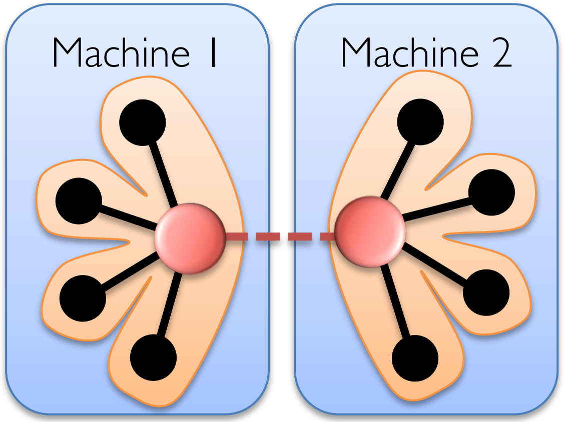
Machine Learning | Abstractions | Scalable Systems




Graphical Model Inference



Vertex Program




GraphLab/GraphX System

# Machine learning community has had an evolving focus on AI Systems

Fast Algorithms

Distributed Algorithms

Deep Learning Frameworks

2009 → 2019

ML for Systems

Machine Learning Frameworks

RL for Systems

→ Integration of Communities

# Waves of AI Research & Connection to Systems

- **1950 to 1974:** *Birth of AI*
  - 1951 Marvin Minsky builds first neural network machine (SNARC)
- **1974 to 1980:** *First AI Winter*
  - Limited processing power and data
- **1980 to 1987:** *Second Wave of AI*
  - XCON (AI for Systems) for DEC → saves $40M annually
- **1987 to 1993:** *Second AI Winter*
  - Collapse of the AI Hardware Market
- **1993 to 2011:** *AI Goes Stealth Mode (aka Machine Learning)*
  - *Confluence of ideas + compute + data → AI starts to work but we call it ML*
- **2011 to 2019:** *Third Wave (AI Goes Deep)*
  - *Compute + data + abstractions → Emergence of AI developers*

# New Forces Driving AI Revolution

## Data


Benchmarks

## Compute


## Abstractions

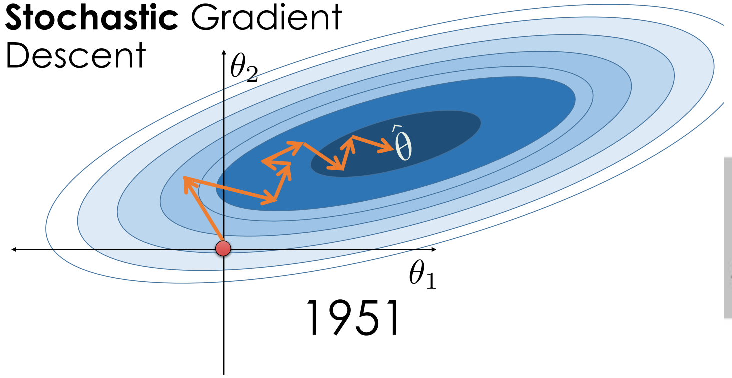
PyTorch

Caffe

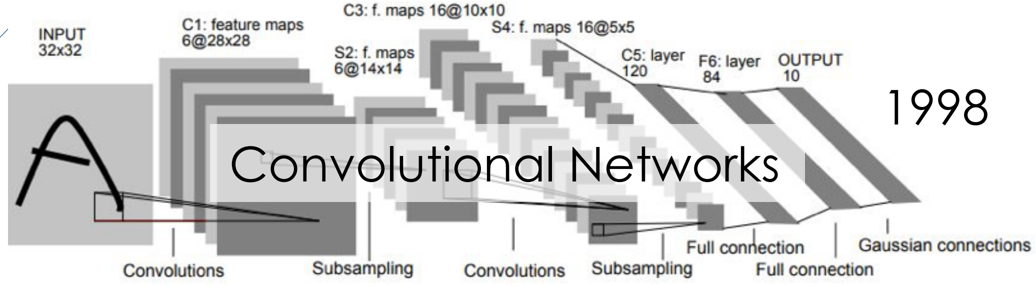
TensorFlow

Advances in Algorithms and Models

**Stochastic** Gradient Descent




1951




Convolutional Networks

1998

# Overview of the Reading

# Required Reading

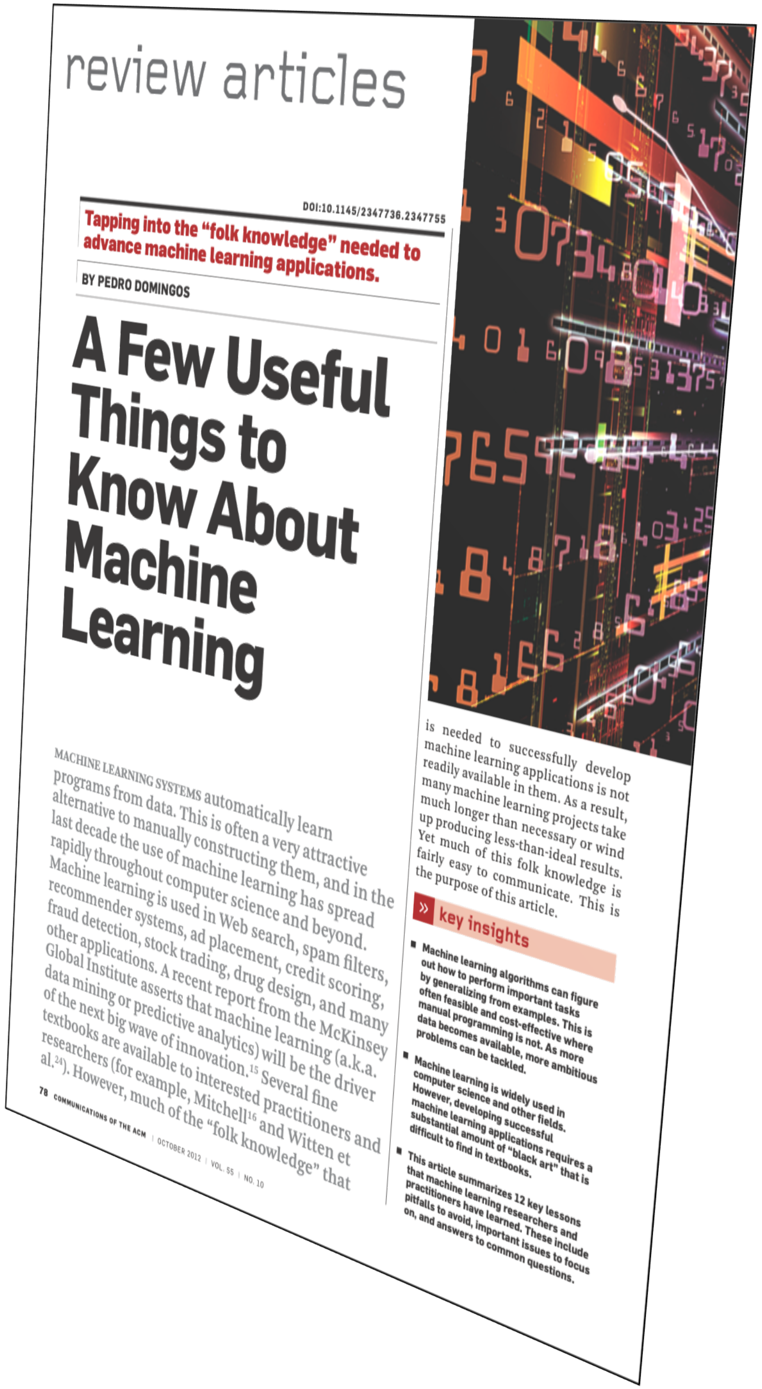
- Perspective on Machine Learning
  - A Few Useful Things to Know About Machine Learning
  - **Goal:** Provide some context on high-level ideas in ML
- Perspective on Systems
  - *Principles of Computer System Design*
  - **Goal:** Provide some context on high-level ideas in Systems
- Views on the field of AI-Systems
  - SysML: The New Frontier of Machine Learning Systems
  - A Berkeley View of Systems Challenges for AI (Mini PC)
  - **Goal:** Observe two recent framings of AI-Systems Research

# A Few Useful Things to Know About Machine Learning

Pedro Domingos (CACM'12)

# Context

- When: 2012
  - Right before explosion in deep learning
- Why?
  - Provides an overview of several of the **big ideas in ML**
  - Describes **essential ingredients** of machine learning
  - Outlines **key trade-offs**
- Issues
  - Pretty focused on classic problems



review articles

DOI:10.1145/2347736.2347755

Tapping into the "folk knowledge" needed to advance machine learning applications.

BY PEDRO DOMINGOS

# A Few Useful Things to Know About Machine Learning

MACHINE LEARNING SYSTEMS automatically learn programs from data. This is often a very attractive alternative to manually constructing them, and in the last decade the use of machine learning has spread rapidly throughout computer science and beyond. Machine learning is used in Web search, spam filters, recommender systems, ad placement, credit scoring, fraud detection, stock trading, drug design, and many other applications. A recent report from the McKinsey Global Institute asserts that machine learning (a.k.a. data mining or predictive analytics) will be the driver of the next big wave of innovation.[15] Several fine textbooks are available to interested practitioners and researchers (for example, Mitchell[16] and Witten et al.[24]). However, much of the "folk knowledge" that is needed to successfully develop machine learning applications is not readily available in them. As a result, many machine learning projects take much longer than necessary or wind up producing less-than-ideal results. Yet much of this folk knowledge is fairly easy to communicate. This is the purpose of this article.

» key insights

- Machine learning algorithms can figure out how to perform important tasks by generalizing from examples. This is often feasible and cost-effective where manual programming is not. As more data becomes available, more ambitious problems can be tackled.
- Machine learning is widely used in computer science and other fields. However, developing successful machine learning applications requires a substantial amount of "black art" that is difficult to find in textbooks.
- This article summarizes 12 key lessons that machine learning researchers and practitioners have learned. These include pitfalls to avoid, important issues to focus on, and answers to common questions.

# Big Ideas in ML Research

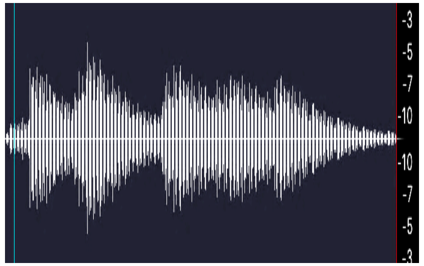
- Generalization (Underfitting/Overfitting)
  - What is being “learned”?
- Inductive Biases and Representations
  - What assumptions about domain enable efficient learning?
- Efficiency (Data and Computation)
  - How much data and time are needed to learn?
- Details: Objectives/Models/Algorithms

# Machine Learning ≈ Function Approximation

## Object Recognition

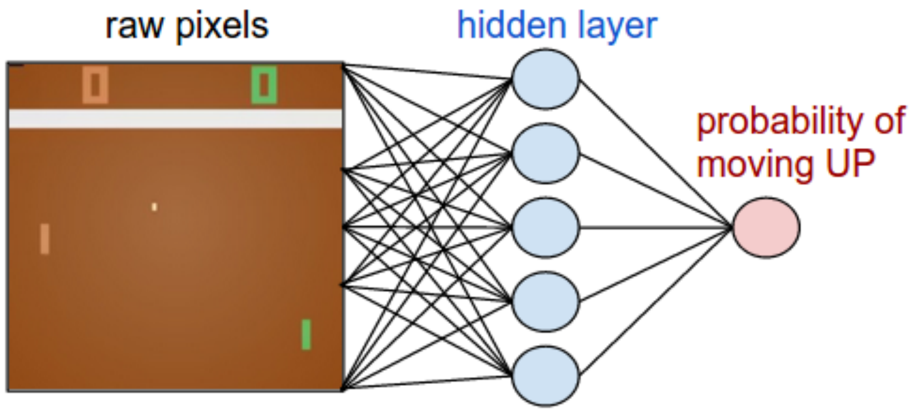
**Label:***Cat*

## Speech Recognition

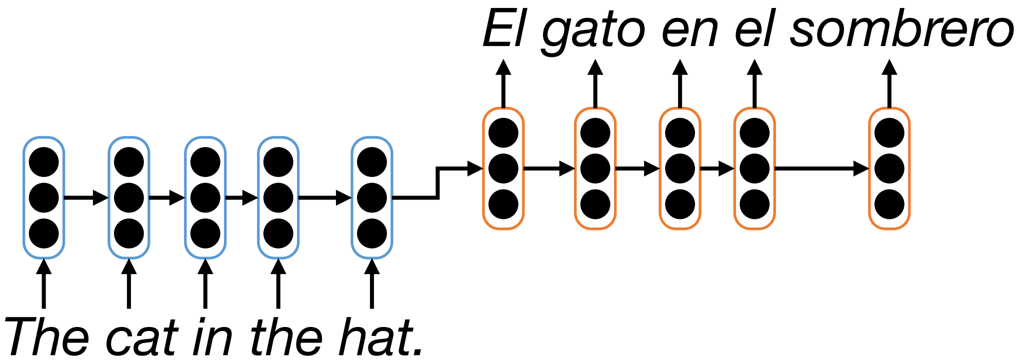


*"The cat in the hat"*

## Robotic Control




## Machine Translation



# Supervised Machine Learning

- Given data containing the function **inputs** and **outputs**

Data

| **Input** | | **Output** | |
|---|---|---|---|
| $X_1$ |  | $Y_1$ | cat |
| $X_2$ |  | $Y_2$ | *baby* |
| … | | … | |
| $X_n$ |  | $Y_n$ | *baby* |

Model

$$f_\theta(x) \to y$$

Parameters

Goal

Loss

$$\theta^* = \arg\min_\theta \mathbb{E}_D \left[ L\left(f_\theta(x), y\right) \right]$$

Over future data

Training (approximates the goal over training data):

$$\hat{\theta} = \arg\min_\theta \frac{1}{n} \sum_{i=1}^{n} L\left(f_\theta(x_i), y_i\right)$$

# Much of the research focus

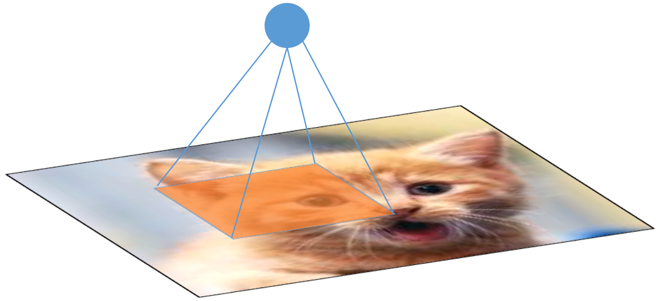
$$f_\theta(x) \rightarrow y$$

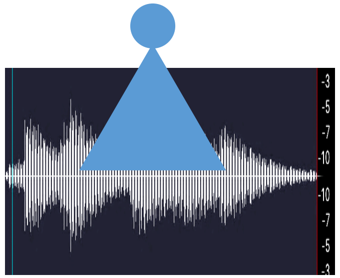
- How do we make our functions sufficiently expressive
- **Inductive Bias:** capture domain knowledge and assumptions
- Easy to train → differentiable and

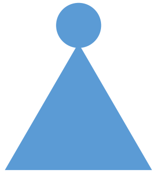
# Architectures for Different kinds of inputs

## Convolutional Networks

spatial reasoning tasks





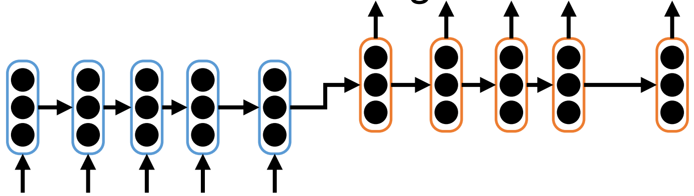


The quick brown fox…

## Recurrent Networks

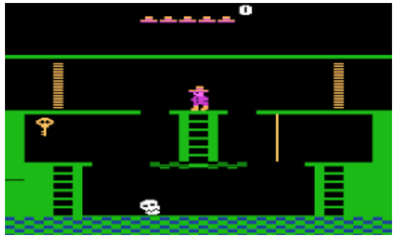
Sequential reasoning tasks

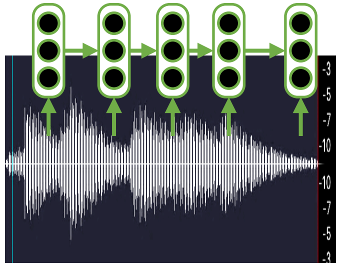
*El gato en el sombrero*



*The cat in the hat.*

Reinforcement
Learning



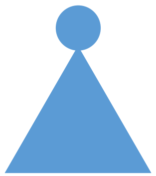


Speech
recognition

# Architectures for Different kinds of inputs

## al Networks

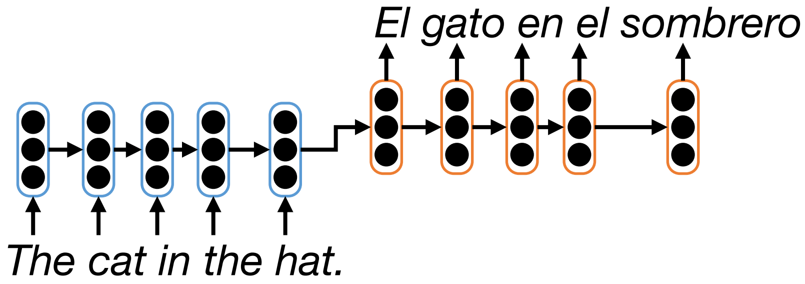
oning tasks



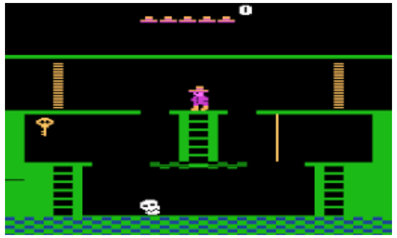
The quick brown fox…

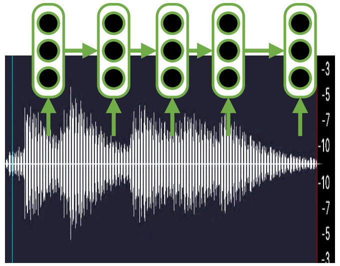
## Recurrent Networks

Sequential reasoning tasks




Reinforcement Learning

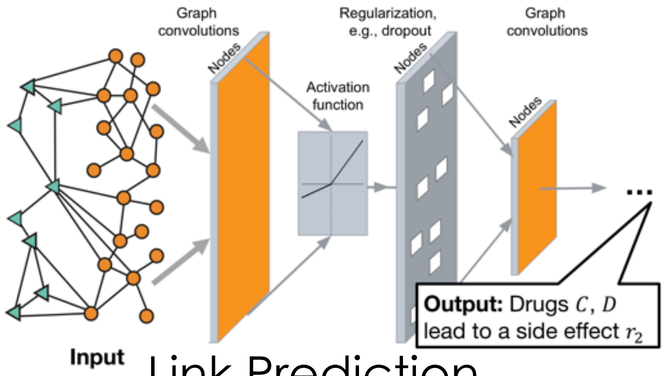




Speech recognition

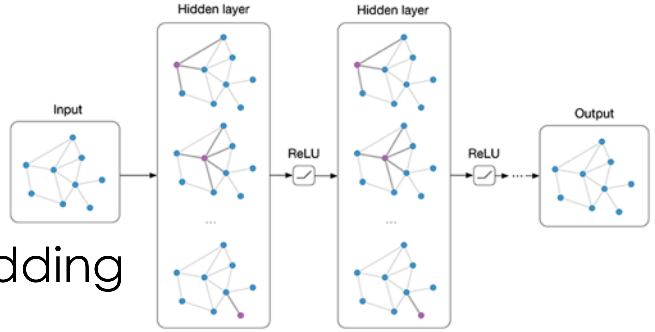
## Graph Networks

Operating on graph data



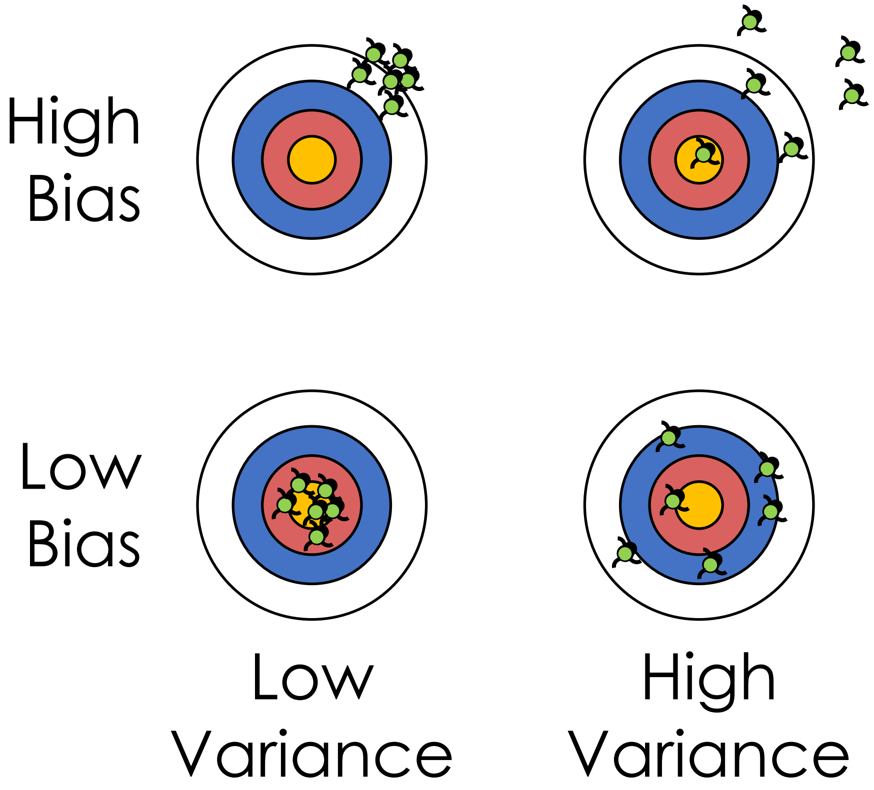

Link Prediction

Graph Embedding



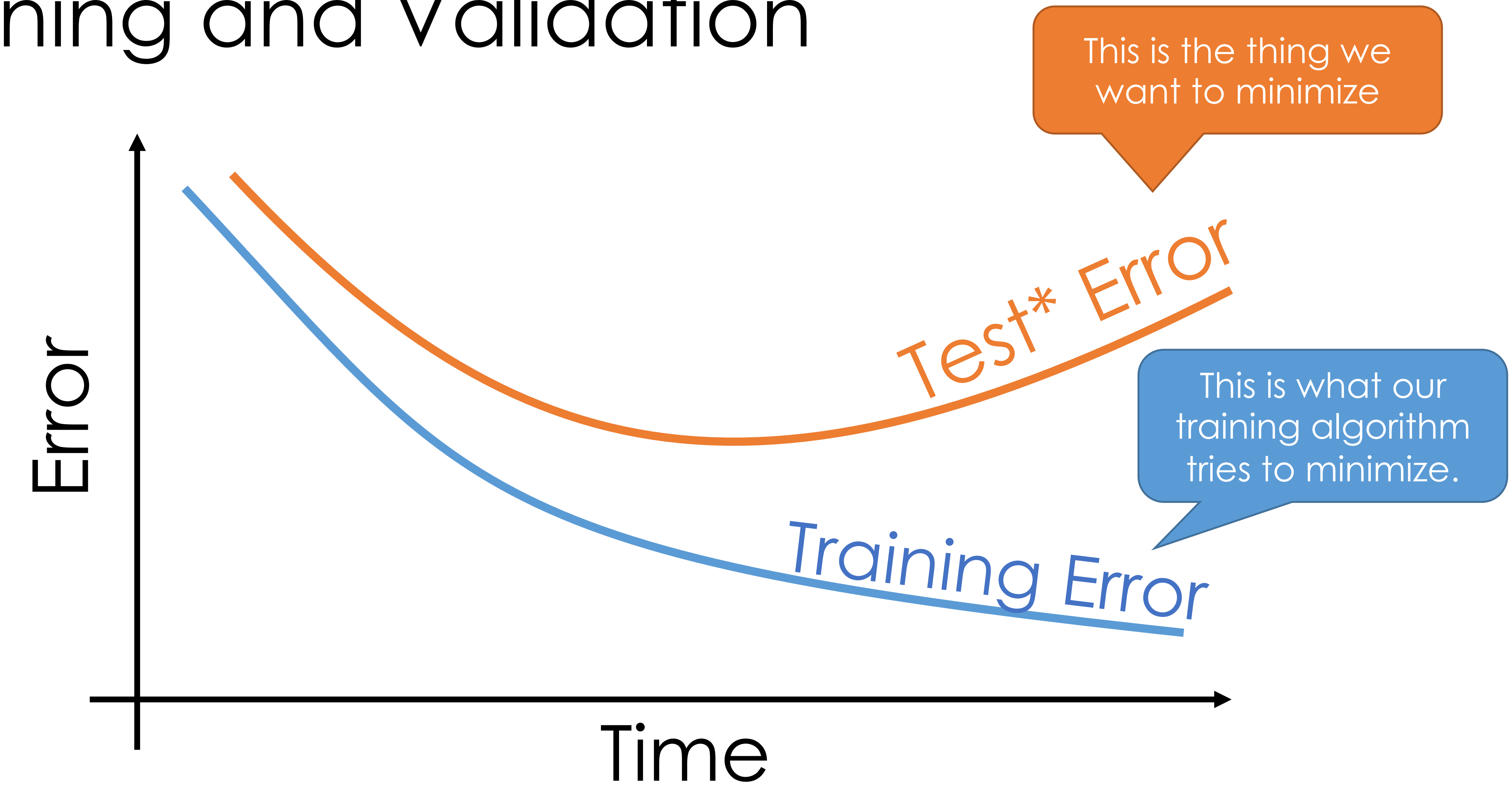
# The Bias Variance Tradeoff

- Fundamental trade-off in ML (classically)




- Low bias learning techniques
  - Typically higher variance …
- Increasing data supports
  - Higher variance techniques
- Deep neural networks?
  - Focus on **training procedure** not models to control tradeoff
    - Initialization, SGD, Dropout, learning rates, early stopping, …

# Training and Validation




*If you are making modeling decisions based on this then it should be called validation error.

# On Dataset Size and Learning

- Data is a a resource! (e.g., like processors and memory)
  - Is having lots of processors a problem?
- You don’t have to use all the data!
  - Though using more data can often help
- More data *often** dominates models and algorithms

EXPERT OPINION

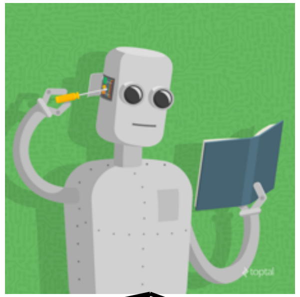
Contact Editor: **Brian Brannon,** bbrannon@computer.org

**The Unreasonable Effectiveness of Data**

Alon Halevy, Peter Norvig, and Fernando Pereira, *Google*

*More data also enables more sophisticated.

# Taxonomy of Machine Learning



Labeled Data

Reward

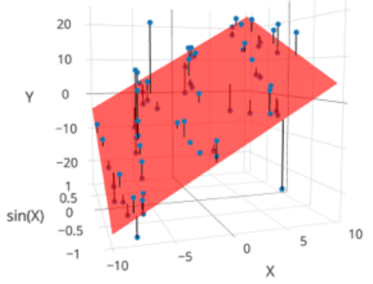
Unlabeled Data

## Supervised Learning

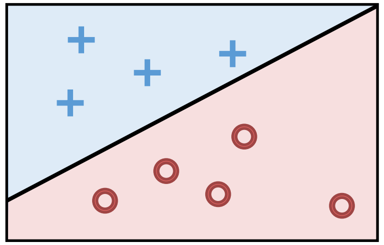
Quantitative Response

Categorical Response

### Regression




### Classification

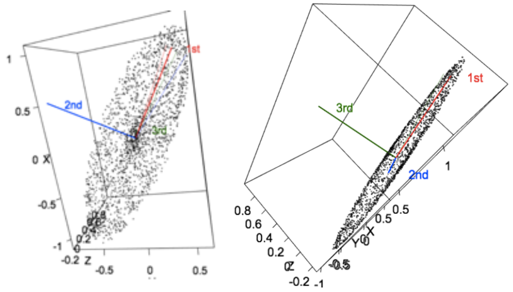


## Reinforcement Learning

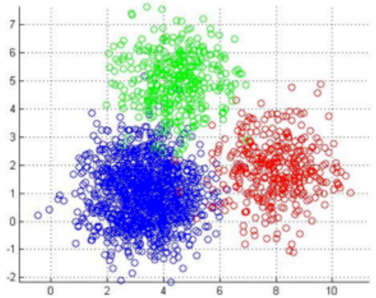

Alpha Go

## Unsupervised Learning

### Dimensionality Reduction



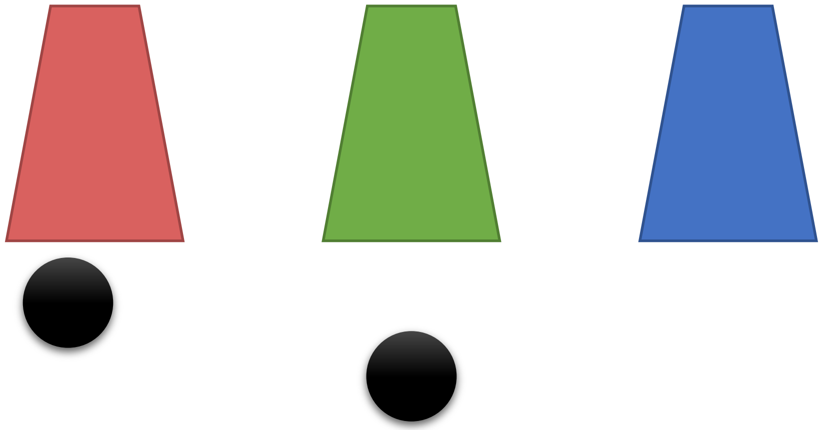
### Clustering



# Machine Learning is not Magic

- Requires data/interaction with signal


- Requires some assumptions about the learned process



Place ball in one of the cups

# Required Reading

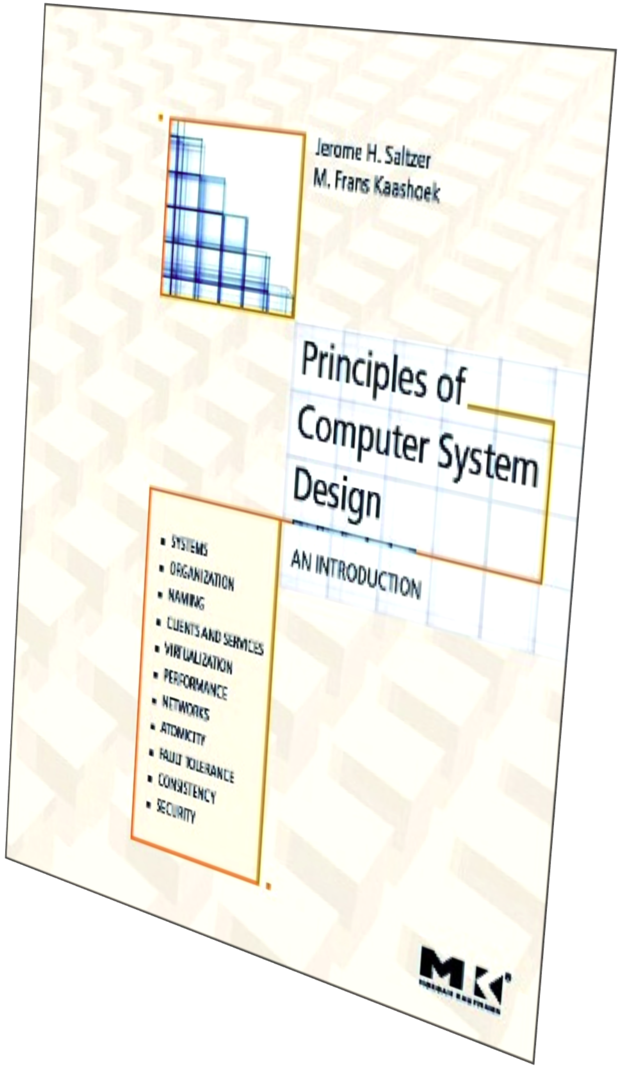
- Perspective on Machine Learning
  - A Few Useful Things to Know About Machine Learning
  - **Goal:** Provide some context on high-level ideas in ML
- Perspective on Systems
  - *Principles of Computer System Design*
  - **Goal:** Provide some context on high-level ideas in Systems
- Views on the field of AI-Systems
  - SysML: The New Frontier of Machine Learning Systems
  - A Berkeley View of Systems Challenges for AI (Mini PC)
  - **Goal:** Observe two recent framings of AI-Systems Research

# Principles of Computer System Design (Chapter 1)

Jerome H. Saltzer and M. Frans Kaashoek (MIT Press 2009)

# Context

- What?
  - MIT Systems (6.033) Course Textbook
- Why?
  - Really well written book
  - Provides an overview of several of the **big ideas in systems**
  - Discusses the **fundamental challenges** addressed in systems research




- Related Reading
  - "Hints for Computer System Design" Butler Lampson (Berkeley PhD, Turing Award Winner)

# Big Ideas in Systems Research

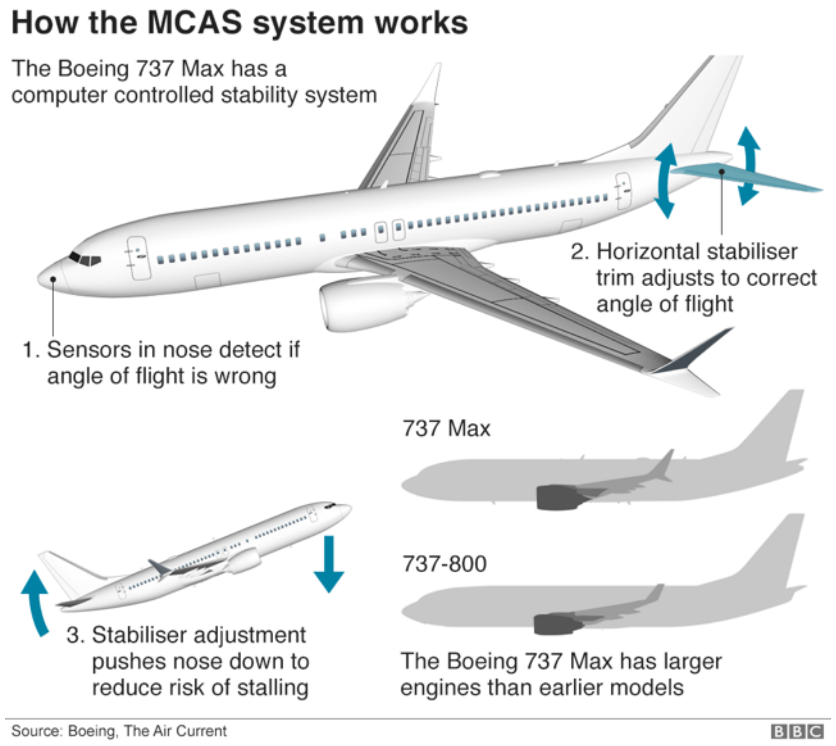
- Managing Complexity
  - Abstraction, modularity, layering, and hierarchy
- Tradeoffs
  - What are the fundamental constraints?
  - How can you reach new points in the trade-off space?
- Problem Formulation
  - What are the requirements and assumptions?

# Sources of Complexity in Systems

- **Emergent Properties:** properties of a system that are not evident in the individual components
  - Difficult to anticipate system behavior based on the behavior of the individual parts
  - A system is often greater than the sum of its parts.

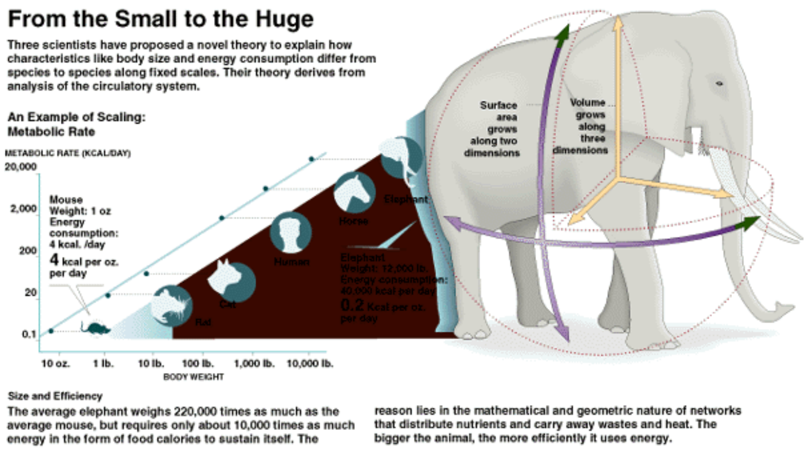
# Sources of Complexity in Systems


- **Propagation of Effects:** a small change in one part of the system can affect many other parts of the system.
  - "There are no small changes in a large system"
- Implications
  - Difficult to reason about affects of changes
  - Slows down innovation



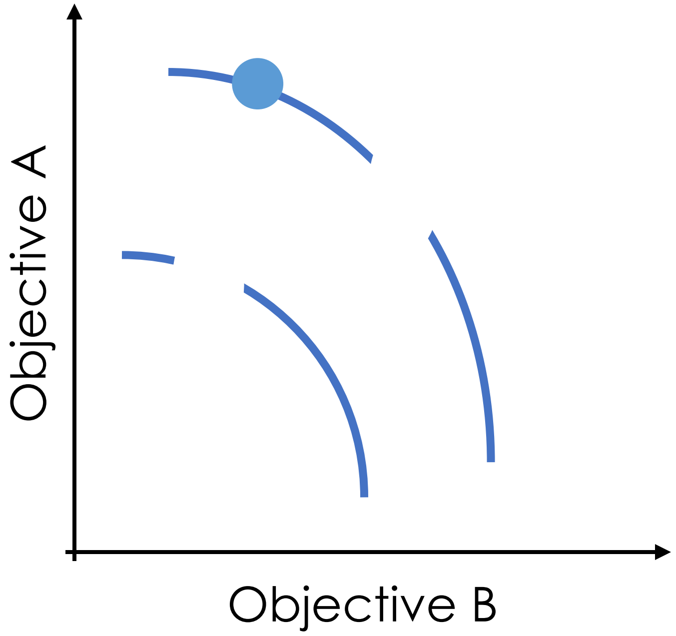
# Sources of Complexity in Systems

- **Incommensurate Scaling:** not all parts of a system scale at the same rate.
  - A 10x change in performance or scale → changes in design
  - Solving a bigger problem can often require new designs
- Examples?
  - CPU speeds and memory bandwidth



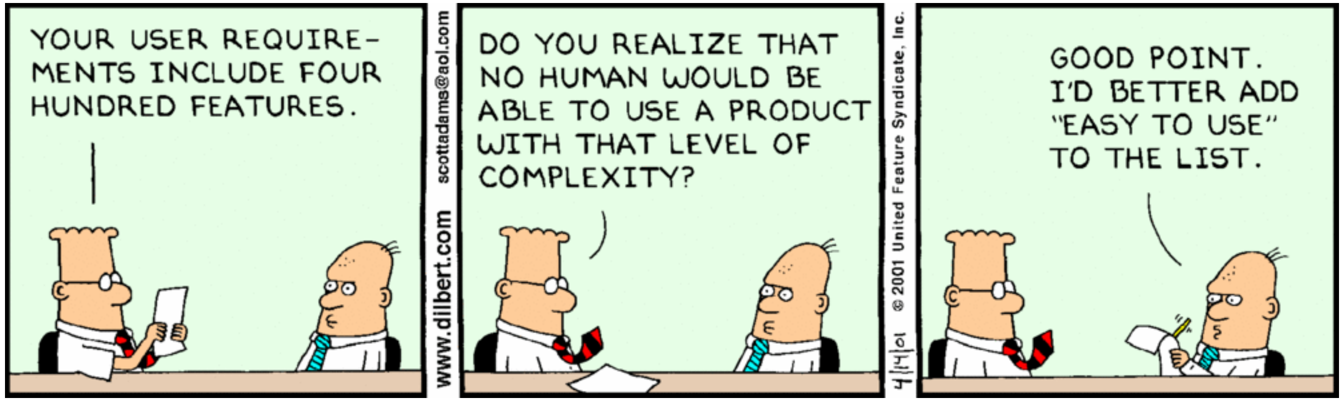

# Sources of Complexity in Systems

- **Tradeoffs:** Finding the right balance of competing objectives or requirements.
- Examples?
  - Bias and Variance from earlier.
- Issue?
  - Pushing the frontier
  - Moving through the tradeoff space
  - Finding the right balance



# Sources of Complexity in Systems

- **Excessive Generality**
  - If it is good for everything, it is good for nothing.
- Examples?

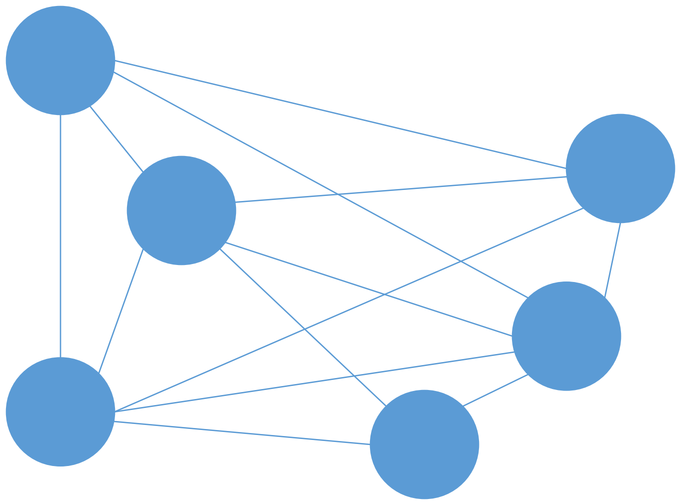
# Sources of Complexity in Systems

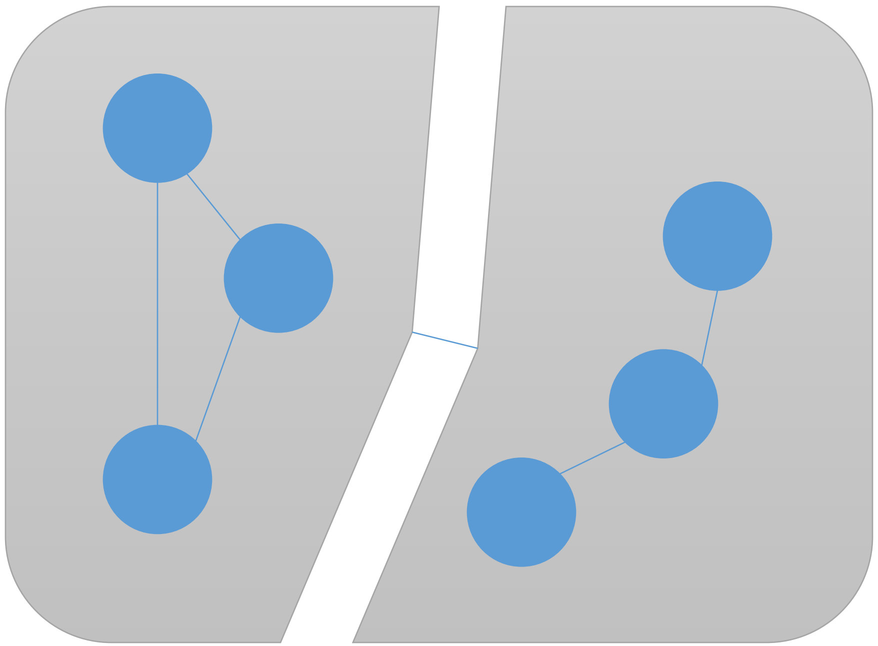
- **Cascading and interacting requirements:**
  - the quantity and interaction between requirements can disproportionally complicate system design.
- **Principle of Escalating Complexity**
  - Adding a requirement increases complexity out of proportion

- Identifying the **right (minimal) requirements** is often a key contribution in systems research

# Coping with Complexity

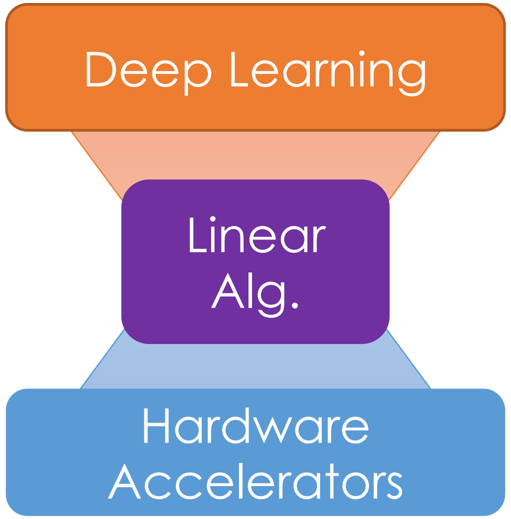
- **Modularity and Abstraction:** dividing the system into smaller parts with well defined boundaries





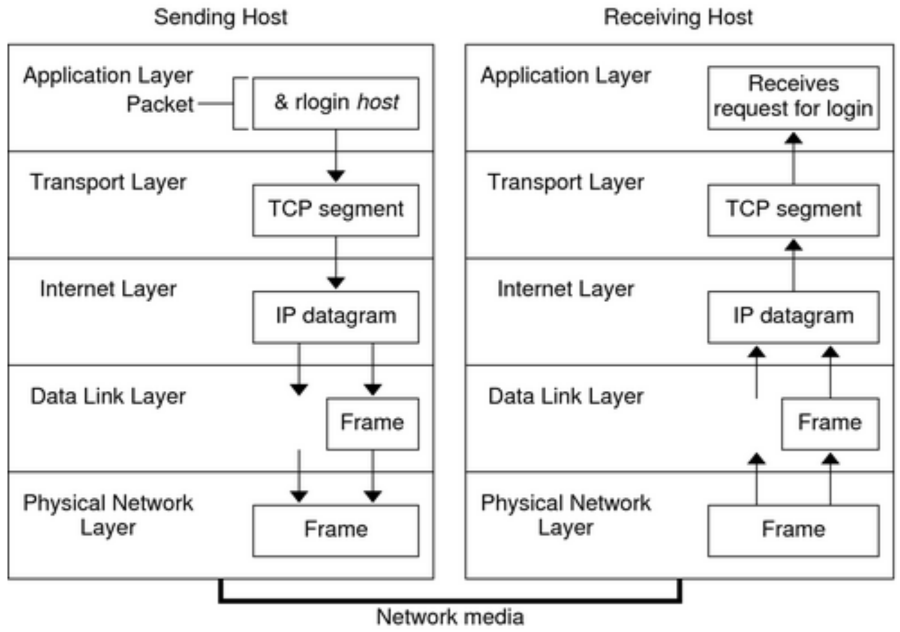
# Abstraction

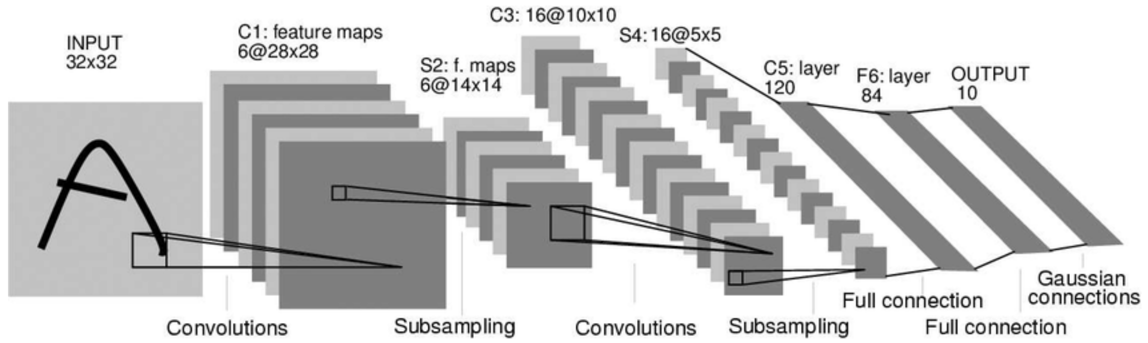
- Good abstraction design can have substantial impact
  - Often a key contribution in systems research
- Examples?
  - Theano → Caffe → TensorFlow → PyTorch
- What makes a good abstraction?
  - **Simplicity** → matches user's expectations
  - **Expressiveness** → captures user's intent
- What makes a bad abstraction?
  - **Leaky Abstractions:** requires understanding design decision of underlying system



# Coping with Complexity

- **Layering and Hierarchy:** mechanisms for composing modules
- Examples?

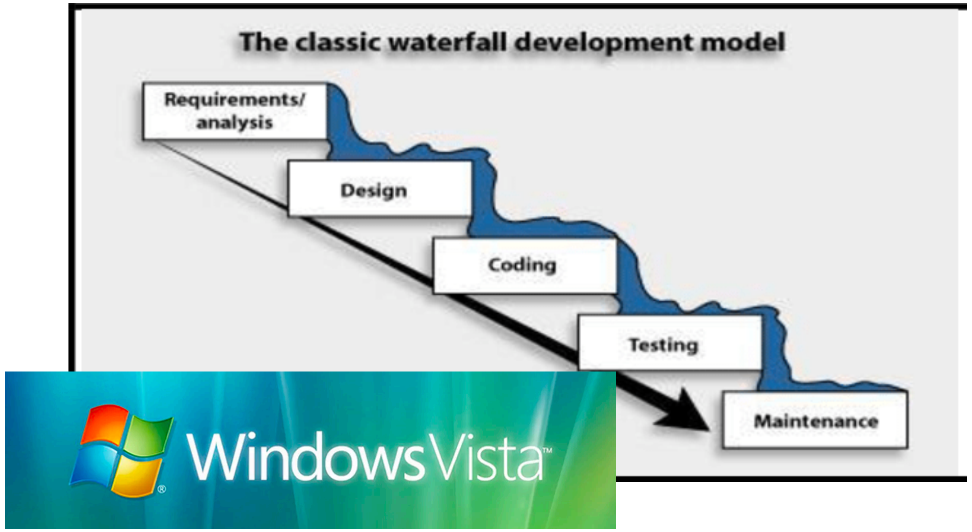


# Coping with Complexity

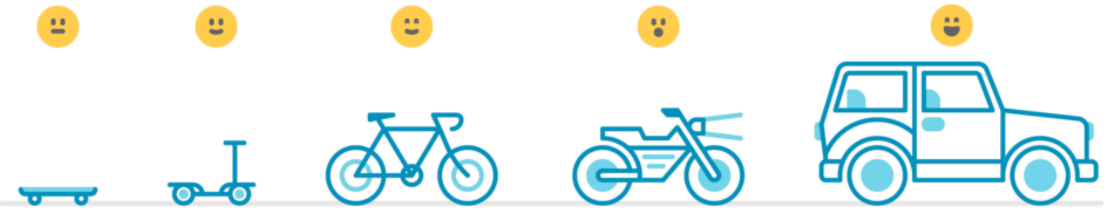
- **Iteration:** start simple and evaluate design decisions incrementally
  - Take small steps, measure often, be prepared to abandon designs, study failures

Waterfall (old looking graphic)



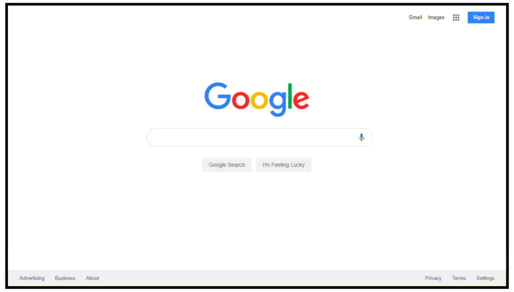

Incremental

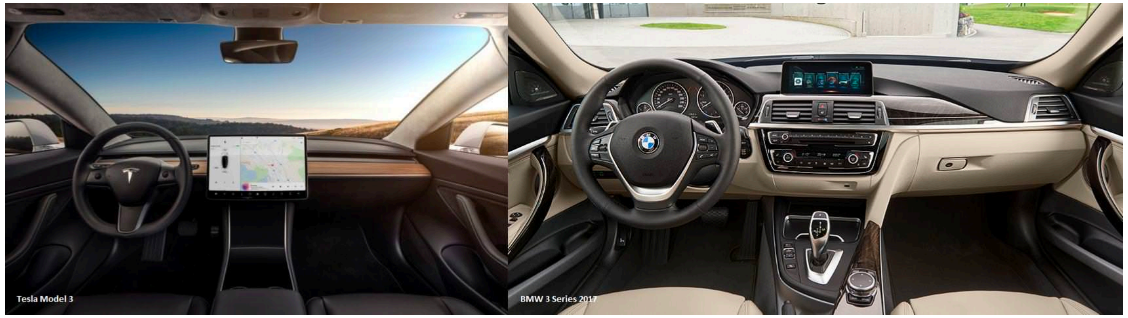
Iterative



# Coping with Complexity

- **Adopt Sweeping Simplifications:** seek the **minimal design** and leverage **simplifying assumptions.**
- **Minimal Design:** when in doubt throw it out.
  - Good systems are defined by what they leave out


- **Simplifying Assumptions:**
  - The choice of **assumptions can be a contribution**
  - **State** and **justify** your assumptions

# Required Reading

- Perspective on Machine Learning
  - A Few Useful Things to Know About Machine Learning
  - **Goal:** Provide some context on high-level ideas in ML
- Perspective on Systems
  - *Principles of Computer System Design*
  - **Goal:** Provide some context on high-level ideas in Systems
- Views on the field of AI-Systems
  - SysML: The New Frontier of Machine Learning Systems
  - A Berkeley View of Systems Challenges for AI (Mini PC)
  - **Goal:** Observe two recent framings of AI-Systems Research

# PC Meeting for

*“A Berkeley View of Systems Challenges for AI”**

*Important Disclaimer: I am a co-author on this paper.

# In Class PC Meeting Format (V.0)

Each paper has allocated ~30 Minutes for discussion

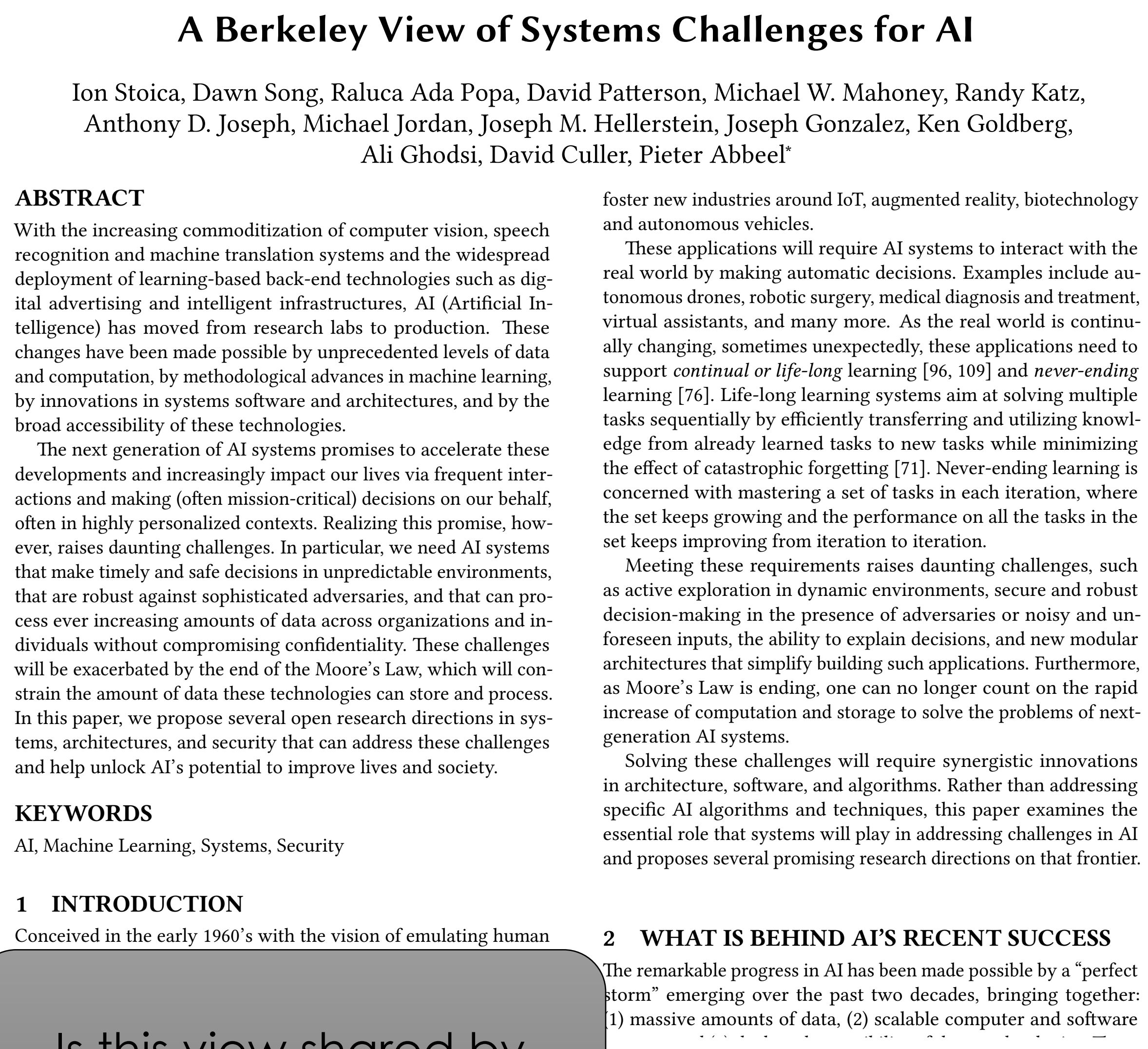
- **Neutral:** recap of the paper (neutral opinion) [5 Minutes]
- **Advocate:** Strengths of the paper [5 Minutes]
- **Critic:** Weaknesses of the paper [5 Minutes]
- Class will discuss **rebuttal** and **improvements** [10 Minutes]
- Brief in-class vote for acceptance into the AI-Sys prelim

# Neutral Presenter

# Paper Overview

## Context:

- **Published:** *TR'2017*
- **From:** *UC Berkeley Faculty*
- **Format:** *Vision Paper*
- **Details:** *Part of a series of Berkeley Views …*
  - *Berkeley View on Serverless …*
  - *Berkeley View on Cloud …*
  - *Berkeley View on Parallel …*



A Berkeley View of Systems Challenges for AI

Ion Stoica, Dawn Song, Raluca Ada Popa, David Patterson, Michael W. Mahoney, Randy Katz, Anthony D. Joseph, Michael Jordan, Joseph M. Hellerstein, Joseph Gonzalez, Ken Goldberg, Ali Ghodsi, David Culler, Pieter Abbeel*

ABSTRACT

With the increasing commoditization of computer vision, speech recognition and machine translation systems and the widespread deployment of learning-based back-end technologies such as digital advertising and intelligent infrastructures, AI (Artificial Intelligence) has moved from research labs to production. These changes have been made possible by unprecedented levels of data and computation, by methodological advances in machine learning, by innovations in systems software and architectures, and by the broad accessibility of these technologies.

The next generation of AI systems promises to accelerate these developments and increasingly impact our lives via frequent interactions and making (often mission-critical) decisions on our behalf, often in highly personalized contexts. Realizing this promise, however, raises daunting challenges. In particular, we need AI systems that make timely and safe decisions in unpredictable environments, that are robust against sophisticated adversaries, and that can process ever increasing amounts of data across organizations and individuals without compromising confidentiality. These challenges will be exacerbated by the end of the Moore's Law, which will constrain the amount of data these technologies can store and process. In this paper, we propose several open research directions in systems, architectures, and security that can address these challenges and help unlock AI's potential to improve lives and society.

KEYWORDS

AI, Machine Learning, Systems, Security

1 INTRODUCTION

Conceived in the early 1960's with the vision of emulating human

foster new industries around IoT, augmented reality, biotechnology and autonomous vehicles.

These applications will require AI systems to interact with the real world by making automatic decisions. Examples include autonomous drones, robotic surgery, medical diagnosis and treatment, virtual assistants, and many more. As the real world is continually changing, sometimes unexpectedly, these applications need to support *continual or life-long* learning [96, 109] and *never-ending* learning [76]. Life-long learning systems aim at solving multiple tasks sequentially by efficiently transferring and utilizing knowledge from already learned tasks to new tasks while minimizing the effect of catastrophic forgetting [71]. Never-ending learning is concerned with mastering a set of tasks in each iteration, where the set keeps growing and the performance on all the tasks in the set keeps improving from iteration to iteration.

Meeting these requirements raises daunting challenges, such as active exploration in dynamic environments, secure and robust decision-making in the presence of adversaries or noisy and unforeseen inputs, the ability to explain decisions, and new modular architectures that simplify building such applications. Furthermore, as Moore's Law is ending, one can no longer count on the rapid increase of computation and storage to solve the problems of next-generation AI systems.

Solving these challenges will require synergistic innovations in architecture, software, and algorithms. Rather than addressing specific AI algorithms and techniques, this paper examines the essential role that systems will play in addressing challenges in AI and proposes several promising research directions on that frontier.

2 WHAT IS BEHIND AI'S RECENT SUCCESS

The remarkable progress in AI has been made possible by a "perfect storm" emerging over the past two decades, bringing together: (1) massive amounts of data, (2) scalable computer and software

Is this view shared by everyone at Berkeley?

# The **Neutral** Presenter will Summarize

- What is the **problem** being solved?
- **Related work**
- What was the **solution**? (Summary!)
- What **metrics** did they use to evaluate their solution?
  - What were the **Baselines** of comparison?
- What was the **key insight** or **enabling idea**?
- What are the **claimed technical contributions**?

Unfortunately, this is a **View Paper** so this guidance won't quite work.

# What is the Problem?

- **View Paper →** Frames but doesn't solve problems
- *Provides context to the problem domain*
  - Credits recent success of AI on advances in systems, large datasets, and accessibility (open-source + cloud)
  - Future advances in AI require systems innovations
  - Discusses trends in technology and their implications on AI
- **Summary Description:** *This paper describes the key research directions at the intersection of AI and Systems.*

# Summary of Problems

Which problems do you remember?

- Systems that learn and act continuously in dynamic env.
- Preserving privacy and security in AI systems
  - Learning across competing entities
  - Addressing corrupted or fraudulent data and queries
- End of Moore's law and implications on AI hardware
  - AI hardware's role in security
- Ensuring actions taken by AI systems can be explained
- Managing the compositions of models and software in complex systems
- Provisioning AI across the cloud and edge boundaries

# Related Work

- SysML: The New Frontier of Machine Learning Systems
  - Required reading …
- "Who will Control the Swarm"
  - Focuses more on real-time AI/Control in the cloud
  - Slightly more provocative position (cloud is central to swarms)
- "Infrastructure for Usable Machine Learning: The Stanford DAWN Project" (DAWN Project is like the RISE Lab)
  - Stronger emphasis on "usability"
  - Slightly sharper description of specific projects
  - Missed security …

# **Example Problem:** *Systems that learn and act continuously in a dynamic env.*

- **Potential Requirements**
  - Need to update model or latent state in response to observations (**state management**)
  - Need render predictions and learn interactively (**latency**)
  - Need to reason about environment (**modeling/simulation**)
- **Proposed Solutions**
  - Focus on reinforcement learning
  - Leverage dynamic parallelization and simulation
- **Metrics**
  - **Learning:** *Accuracy/Reward* + delay in responding to *concept drift*
  - **System:** *Action Latency, consistency, resource efficiency, …*

# Key Insights and/or Enabling Ideas

- Emphasis on **whole system** and not just **training**
  - **Continuous** training and inference
  - Focus on **composition** of models and traditional software
- Interaction between **security** and **AI**
  - **Hardware:** neural network accelerator → security accelerator
  - **Incentives:** enabling competing parties to learn together
  - **Provenance:** use system to track relationship between data and models → use for explanation

# Technical Contributions

- Algorithms: none
- Theoretical Results: none
- Experimental Results: none

This is common with view/survey papers.

→ Doesn't mean it won't have impact.

# The Advocate

This was an amazing paper …

# **Advocate** and **Critic** Will Discuss

- **Novelty** and **Impact**
  - Are the problem and solution novel and how will the solution affect future research?
- **Technical Qualities**
  - Are the problem **framing** and **assumptions** reasonable
  - Discuss merits of the technical **contributions**
  - Does the **evaluation** support claims and reveal limitations of the proposed approach?
- **Presentation**
  - Discuss the **writing clarity** and **presentation of results**
  - Positioning of **related work**

# Novelty and Impact

What was novelty?

- **Context:** Framing the role of systems in AI today
  - Discussion around open-source and cloud
- **Proposed Problems:**
  - Emphasis on whole system support and composition of models
  - Interaction between security, hardware, and AI
  - Learning across competing organizations
- **Proposed Solutions:** (not the focus)
  - Interesting ideas around role of simulation, enclaves, and use of provenance

Impact:

- Defined research agenda for the **RISE Lab**
- Helped position **NSF expedition** proposal (successful)

# Technical Qualities

- **Assumptions** and **Framing**
  - AI is the future and demands system innovation
  - Emphasis on RL and parallel computing
    - Already an explosion in Deep RL work and parallel systems for AI
  - Need to address training and inference
    - Early evidence of this need in content recommendation systems
- **Contributions**
  - Articulates significance of systems in AI and open challenges
  - Clearly frames a set of interesting well motivated research directions
  - Proposes first steps towards studying some of these problems
- **Evaluation:** None

# Presentation

- Great use of **summary text** to highlight key points
- Attempts to **separate challenges** from **solutions**

# The Critic

This was an amazing paper …

# Novelty and Impact

What was novelty?

- **Context:** A fair amount of the context is well established.
- **Problems:**
  - Many of the problems (e.g., lifelong learning, robust learning, adversarial inputs, secure data, online learning) are well established
  - Problems around RL were not well grounded in applications
- **Solutions:** (not the focus)
  - Many of the problems didn't have clear directions for solutions or were sufficiently established to already have a large body of solutions

Impact:

- Has not yet been well cited.

# Technical Qualities

- **Assumptions** and **Framing**
  - Some assumptions about requirements for AI systems (e.g., need for real-time simulation and online learning) are not justified
  - Many of the assumptions/framing statement are not particularly novel
- **Contributions**
  - While there are a few interesting directions outlined, much of this paper is a summary of several active research agendas
- **Evaluation:** None

# Summary of Problems

Which problems are novel?

- Systems that learn and act continuously in dynamic env.
- Preserving privacy and security in AI systems
  - Learning across competing entities
  - Addressing corrupted or fraudulent data and queries
- End of Moore's law and implications on AI hardware
  - AI hardware's role in security
- Ensuring actions taken by AI systems can be explained
- Managing the compositions of models and software in complex systems
- Provisioning AI across the cloud and edge boundaries

# Presentation

- This reads like a paper written by committee
  - (… it was)
- **Lack of focus:** Too many research directions and ideas
- **Not enough "view":**
  - Doesn't really take a controversial stance
  - it looks more to the present than the future…
- **Writing is a bit disorganized**
  - Several sections repeat standard motivations about importance of ML
  - Lifelong learning is injected in a few random places without much discussion

# Class Discussion

# Rebuttal

- How could the authors address the critic’s concerns?
- How might the authors have improved the paper?
  - Technically
  - Presentation

# Voting

- Would you recommend this paper to a colleague?
- Would you recommend this be part of future reading assignments?
- Should this be part of the AI-Systems Prelim Exam
  - Taken by graduate students to begin research in the field

**https://tinyurl.com/y66fxtyc**