

AI Scaling Laws and Engineering Programs

AI4SE & SE4AI Workshop
September 23, 2026

Tyler Cody, Ph.D.
Peter Beling, Ph.D.

Systems Engineering for Artificial Intelligence (SE4AI)

minimize:

$$f(\mathbf{x})$$

$$\mathbf{x} \in \mathbb{R}^n$$

subject to:

$$\mathbf{g}_L \leq \mathbf{g}(\mathbf{x}) \leq \mathbf{g}_U$$

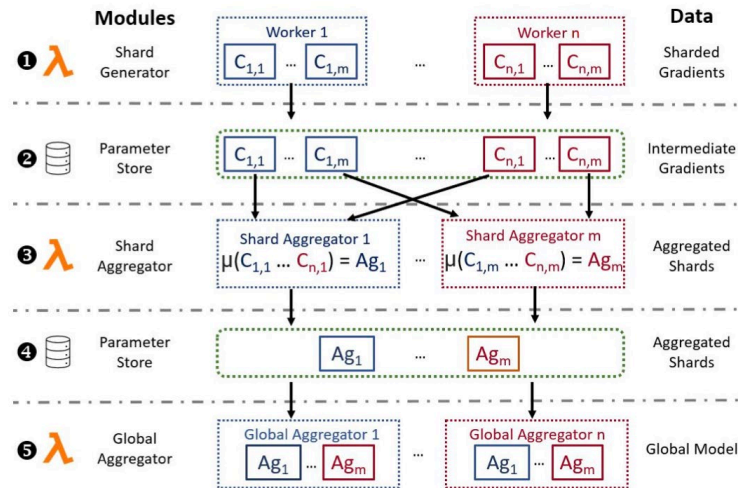
$$\mathbf{h}(\mathbf{x}) = \mathbf{h}_t$$

$$\mathbf{a}_L \leq \mathbf{A}_i \mathbf{x} \leq \mathbf{a}_U$$

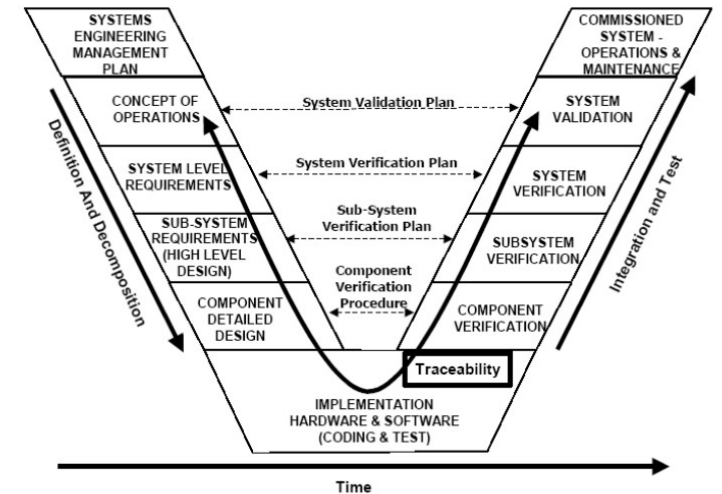
$$\mathbf{A}_e \mathbf{x} = \mathbf{a}_t$$

$$\mathbf{x}_L \leq \mathbf{x} \leq \mathbf{x}_U$$

AI Research



AI Engineering
Research



AI Systems Engineering
Research

Executive Summary

There is a progressive expansion of what needs to be modeled:

- 1. The learned function.** We put data and compute into a learning process and obtain functionality.
- 2. The system that uses and sustains that function.** Its operating environment, physical dependencies, feedback, maintenance, and update mechanisms matter.
- 3. The program that develops that function over time.** Its decisions depend on expectations about what further data, compute, and development effort will produce.

From Models to Systems

2020: AI as model/function inside a system

- Model behavior
- Data dependence
- Probabilistic failure
- Operating envelope



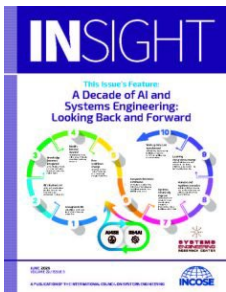
2025: AI-enabled system as socio-technical system

- Human-AI teaming
- Digital twins
- Trust
- Lifecycle adaptation
- Resilience



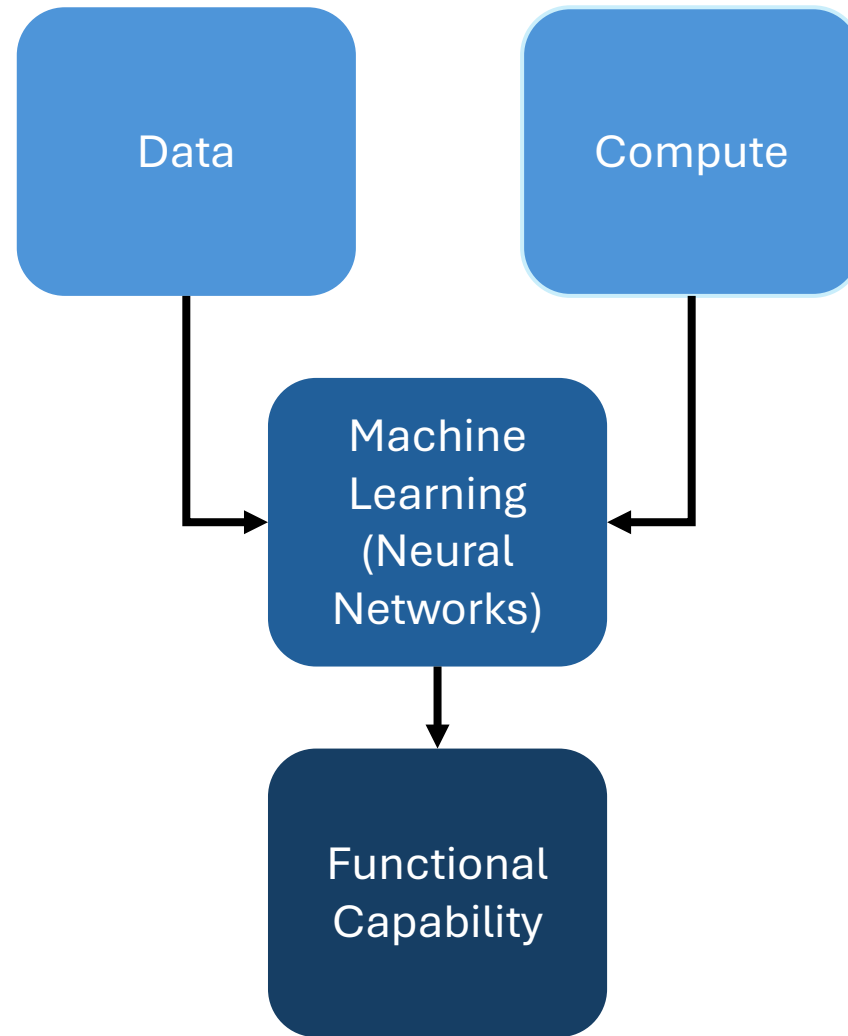
2030: AI-generated system

- Generated requirements
- Generated code
- Generated tests
- Digital thread
- Governance of generation pipeline



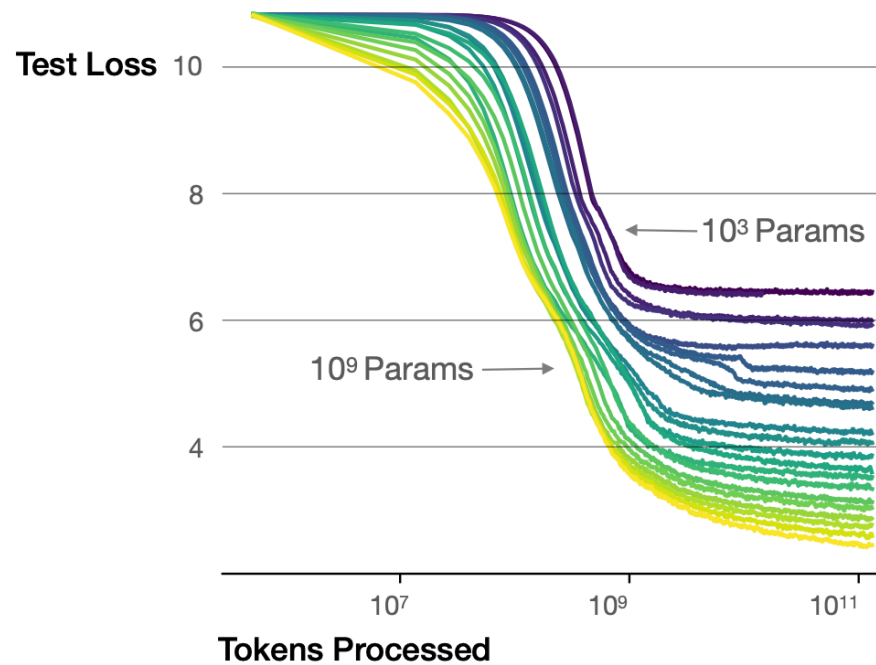
Cody, Tyler, and Peter Beling. "From Models to Systems: The Evolution of T&E for Artificial Intelligence from 2020 to 2030." INSIGHT 29.3 (2026): 90-94.

From Resources to Useful Functions

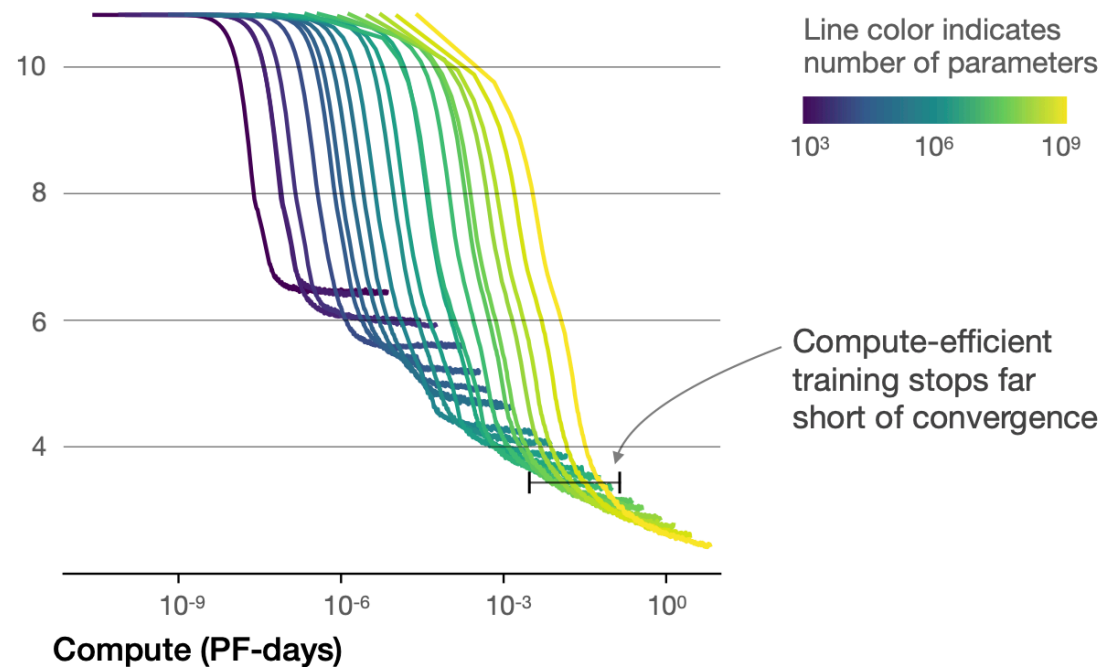


(Foundation Model) Scaling Laws

Larger models require **fewer samples** to reach the same performance



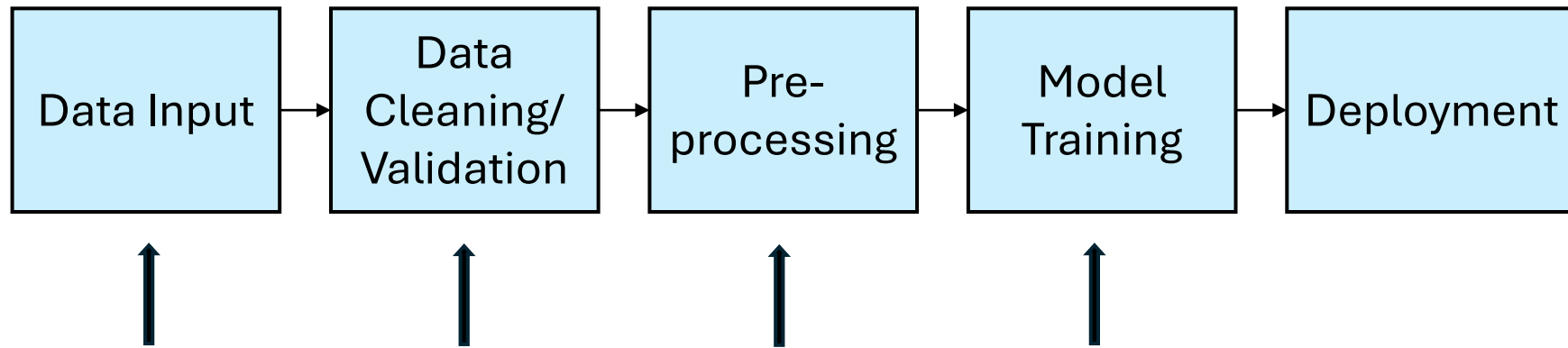
The optimal model size grows smoothly with the loss target and compute budget



Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., ... & Amodei, D. (2020). Scaling laws for neural language models. arXiv preprint arXiv:2001.08361.

Typical “Model” of Machine Learning (Pipeline)

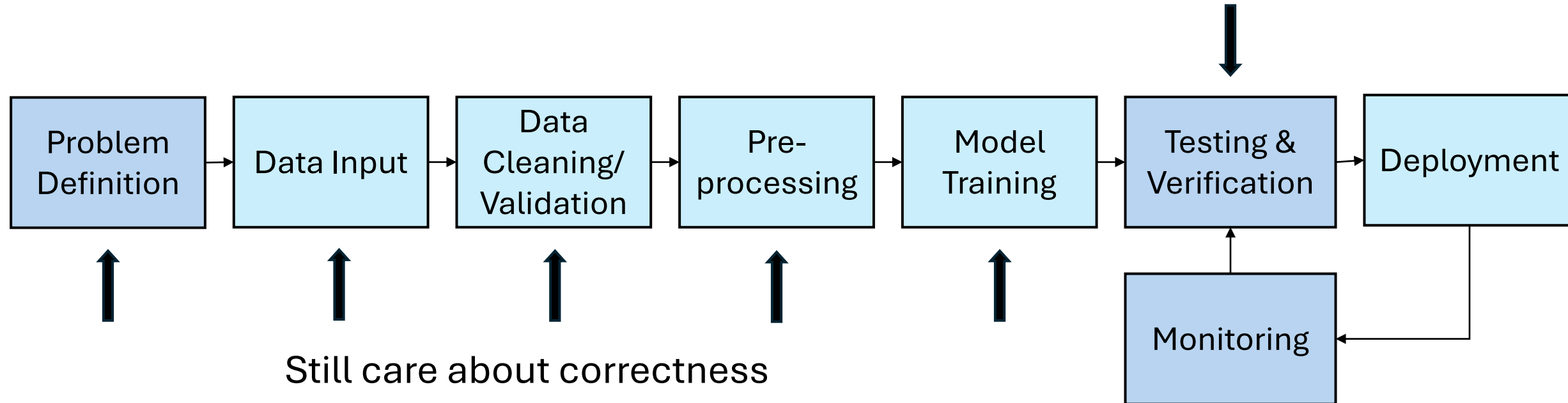
Typical representation of AI/ML pipeline:



Focus of quality testing is on doing these steps correctly, and without bias

Practical “Model” of Machine Learning (Pipeline)

Real-world focused AI researchers and practitioners often add:

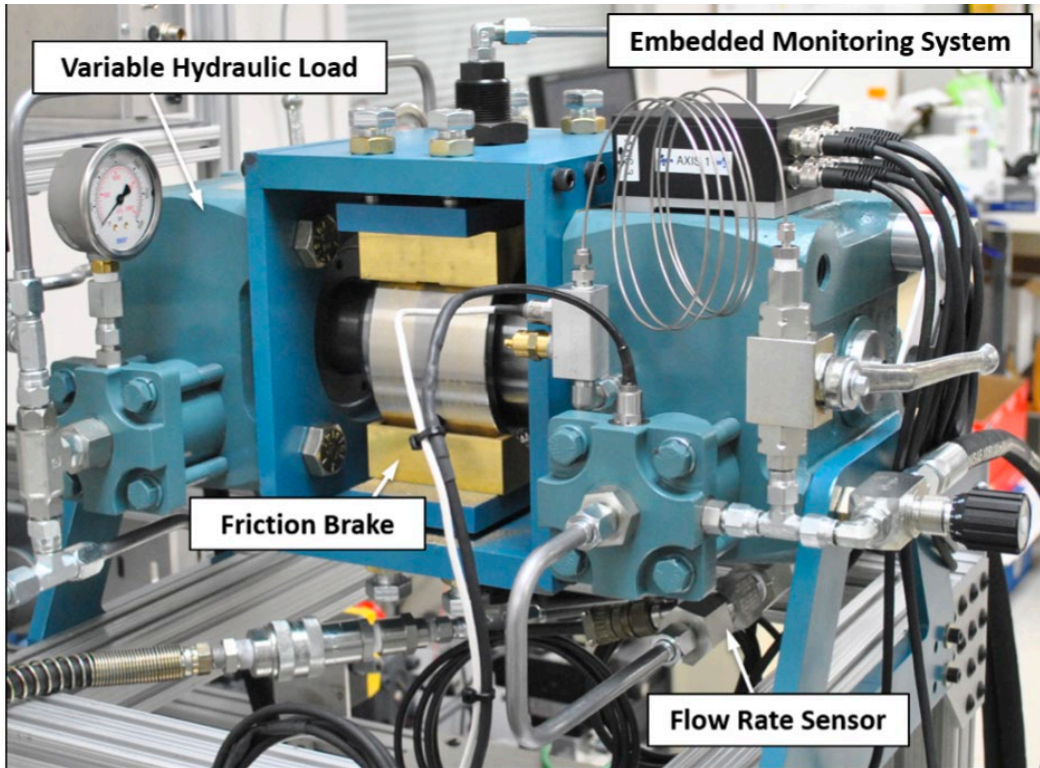


Still care about correctness

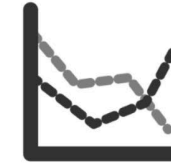
and also, if you solved the right problem and
model works as intended

... but this is still focused on the model as the system.

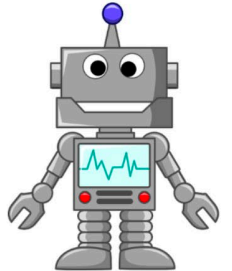
Machine Learning in the Wild



Data Extracted From
Production System



Data Sent to ML



Maintenance Changes
Production System



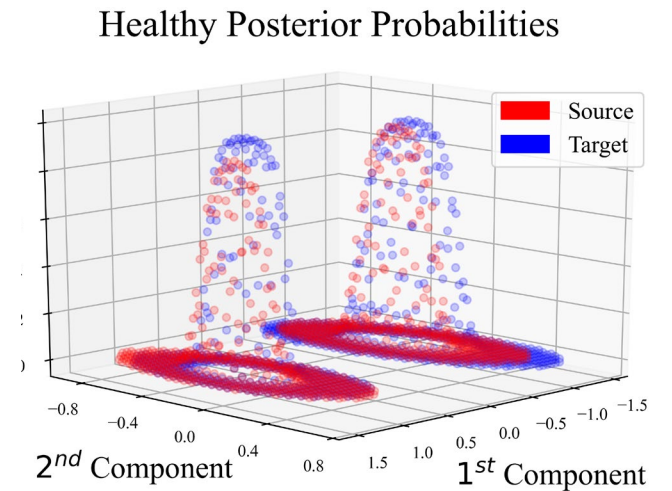
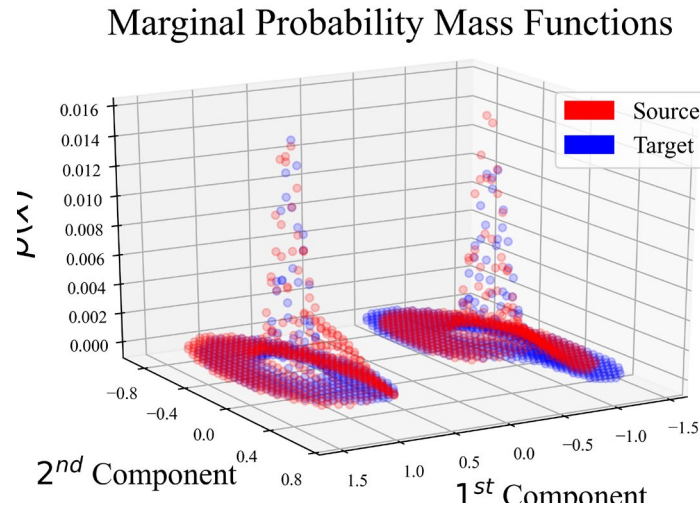
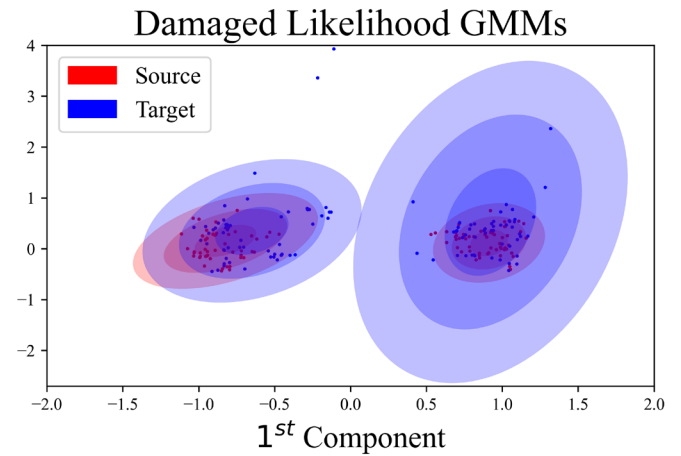
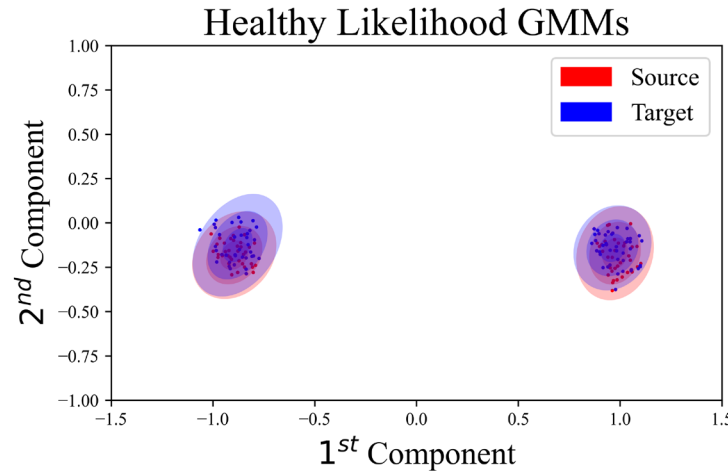
ML Informs
Maintenance

Probabilistic Models of Behavior → Difficulty for AIES

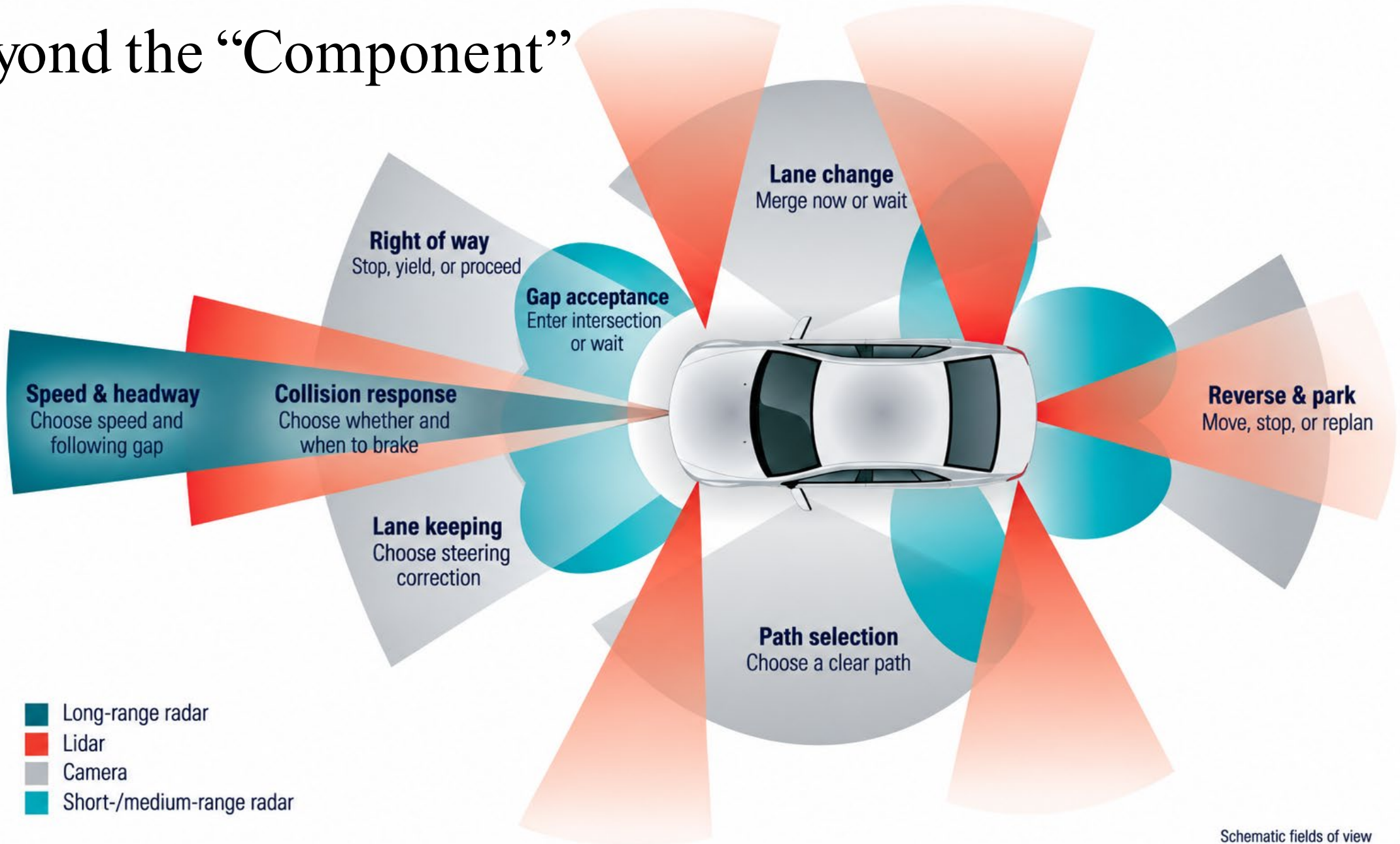
We take PCA of sensor data and fit the likelihoods with Gaussian mixture models (GMMs). We use a prior to combine them and Bayes Theorem to find the posterior.

Transfer Distance	$P(Y = 0)$			
	0.40	0.90	0.99	0.999
$\delta_{X Y=0}$	0.22	-	-	-
$\delta_{X Y=1}$	0.54	-	-	-
δ_X	0.41	0.25	0.22	0.22
$\delta_{Y=0 X}$	0.24	0.23	0.23	0.23

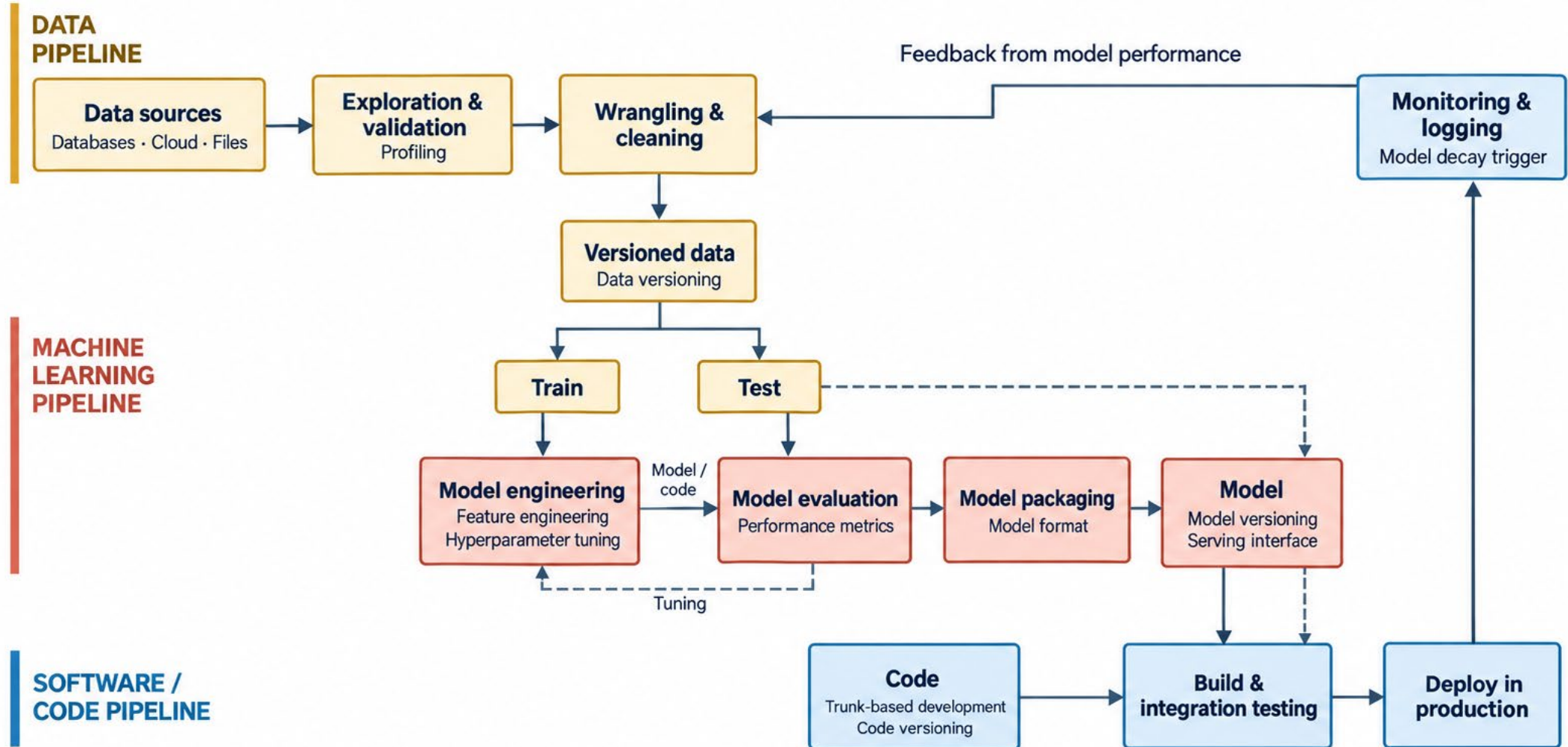
Hellinger Distances



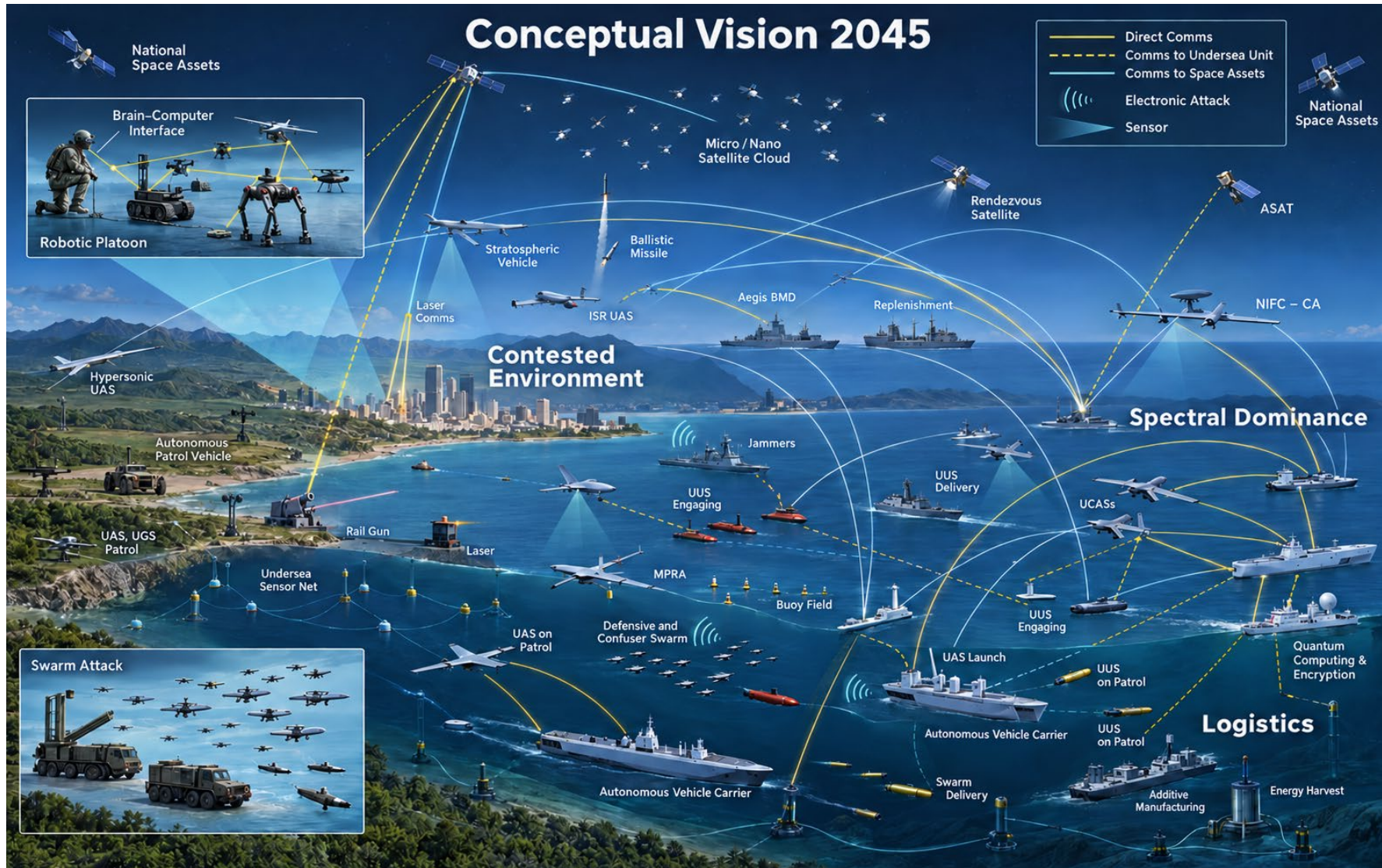
Beyond the “Component”



“Machine Learning Operations”



The Systems “Wall”



A Difference in Worldviews: Resilience

Categories from Survey of ICML/NeurIPS

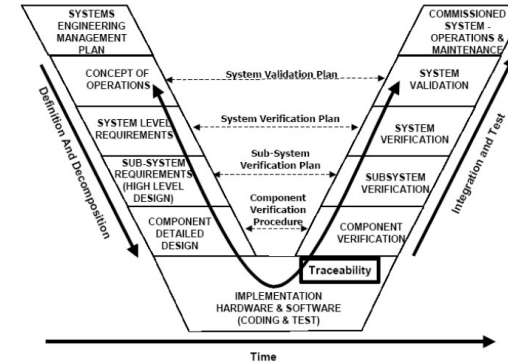
- Robustness against adversarial attack
- Fault Tolerance in Distributed Learning
- Data Diversity and Representation
- Resilience to Concept Drift

NIST's Definition of Operational Resilience

- “the ability of systems to resist, absorb, and recover from or adapt to an adverse occurrence during operation that may cause harm, destruction, or loss of ability to perform mission-related functions”

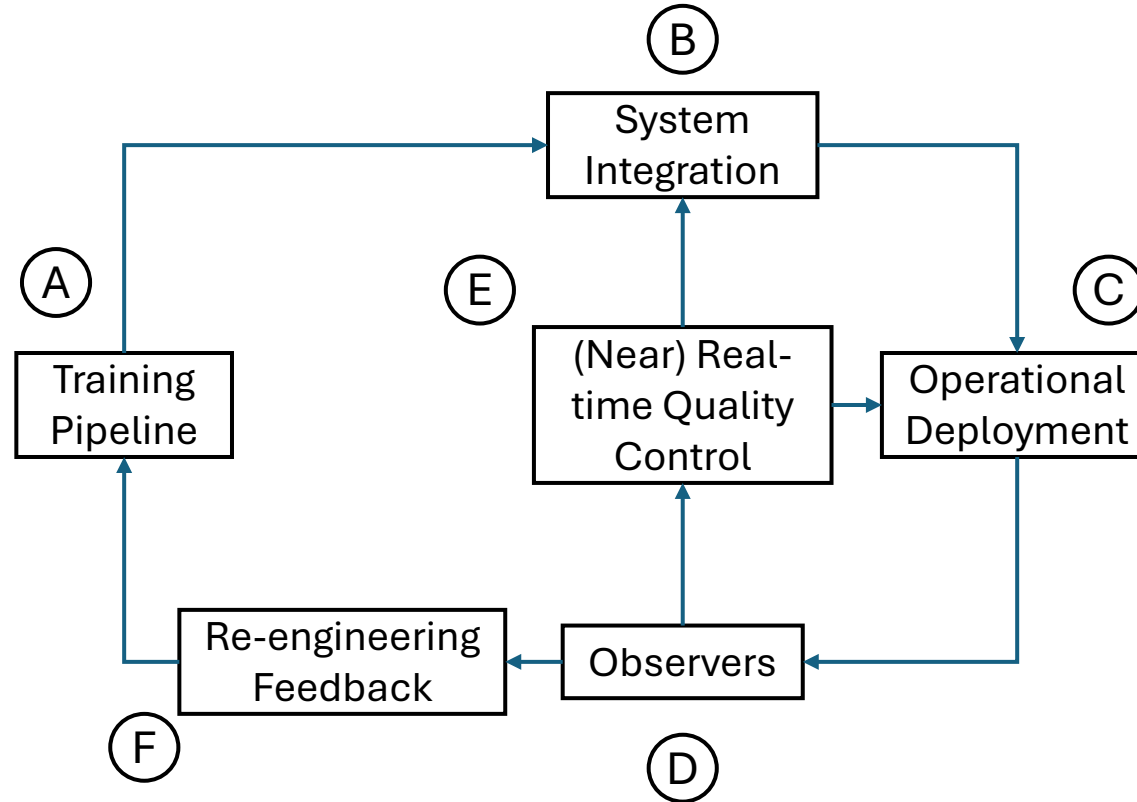
minimize: $f(\mathbf{x})$
 $\mathbf{x} \in \mathbb{R}^n$
subject to: $\mathbf{g}_L \leq \mathbf{g}(\mathbf{x}) \leq \mathbf{g}_U$
 $\mathbf{h}(\mathbf{x}) = \mathbf{h}_t$
 $\mathbf{a}_L \leq \mathbf{A}_t \mathbf{x} \leq \mathbf{a}_U$
 $\mathbf{A}_e \mathbf{x} = \mathbf{a}_t$
 $\mathbf{x}_L \leq \mathbf{x} \leq \mathbf{x}_U$

AI
Research

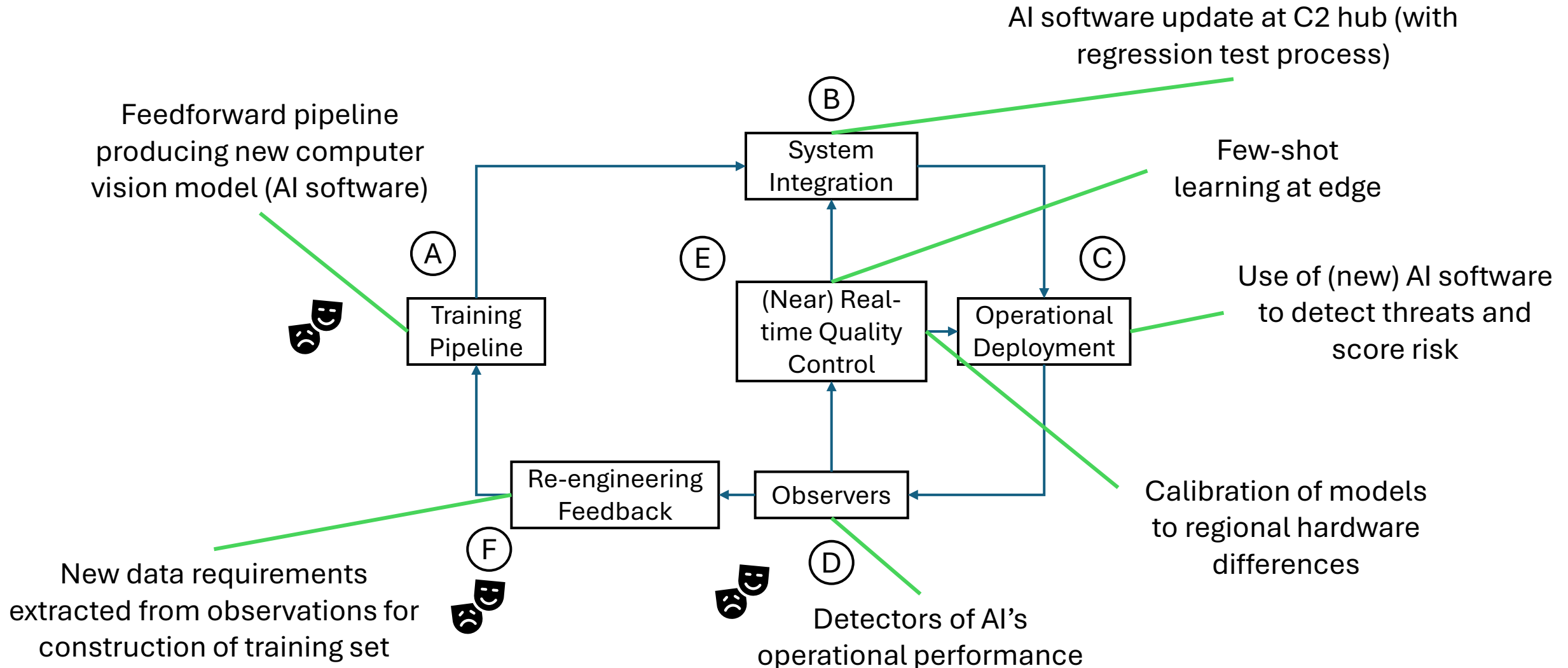


AI Systems Engineering
Research

AI Resilience from a Systems Worldview



AI Resilience from a Systems Worldview



What are foundation models (FMs)?



Key Characteristics:

- Massive pre-training (trillions of tokens)
- Task versatility via prompt- or fine-tuning
- Emergent abilities & scaling laws
- Compute-heavy training once, reuse everywhere

“AI is undergoing a paradigm shift with the rise of models (e.g., BERT, DALL-E, GPT-3) that are trained on broad data at scale and are adaptable to a wide range of downstream tasks. We call these models foundation models to underscore their critically central yet incomplete character.”

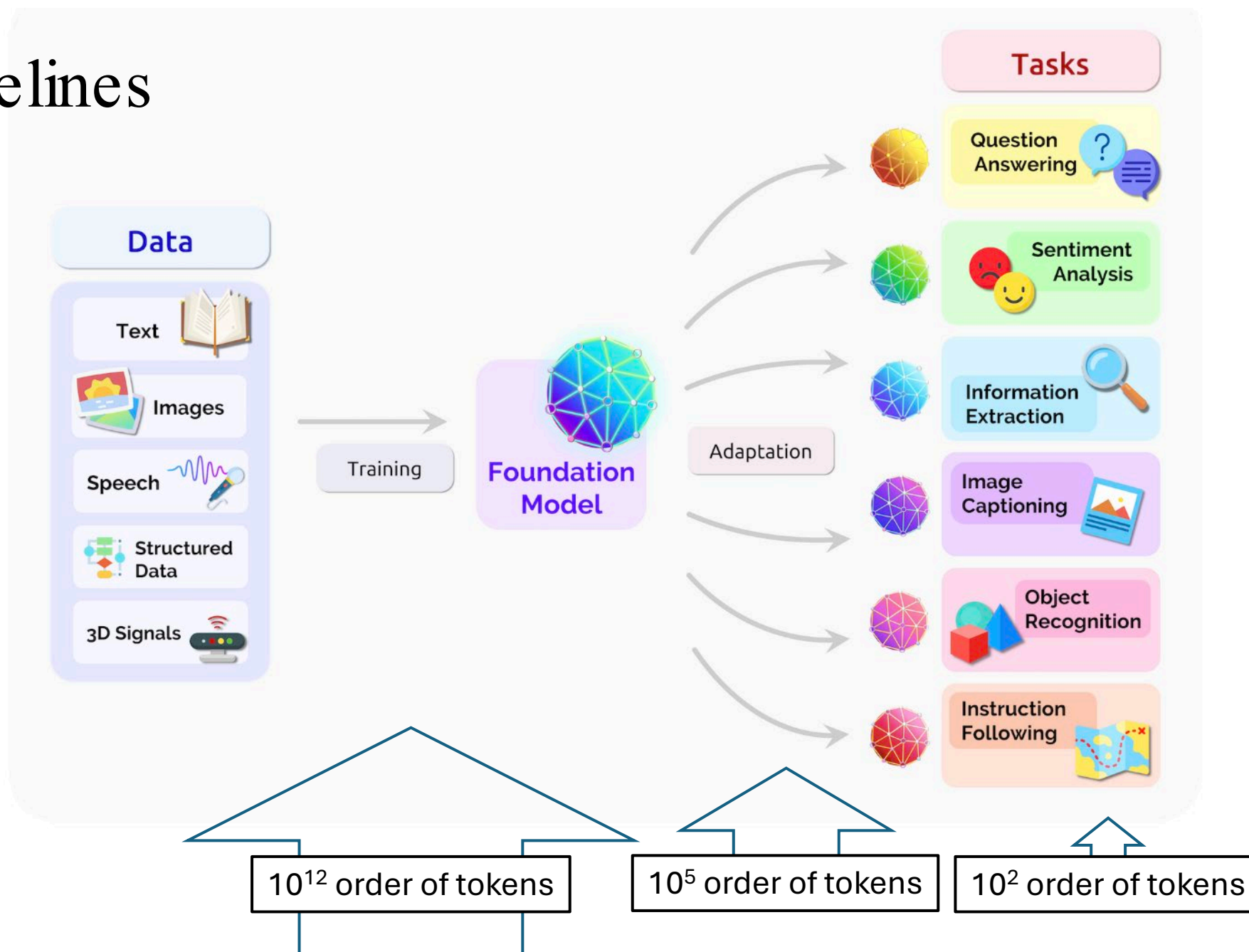
2022, Authors of “On the opportunities and risks of foundation models”



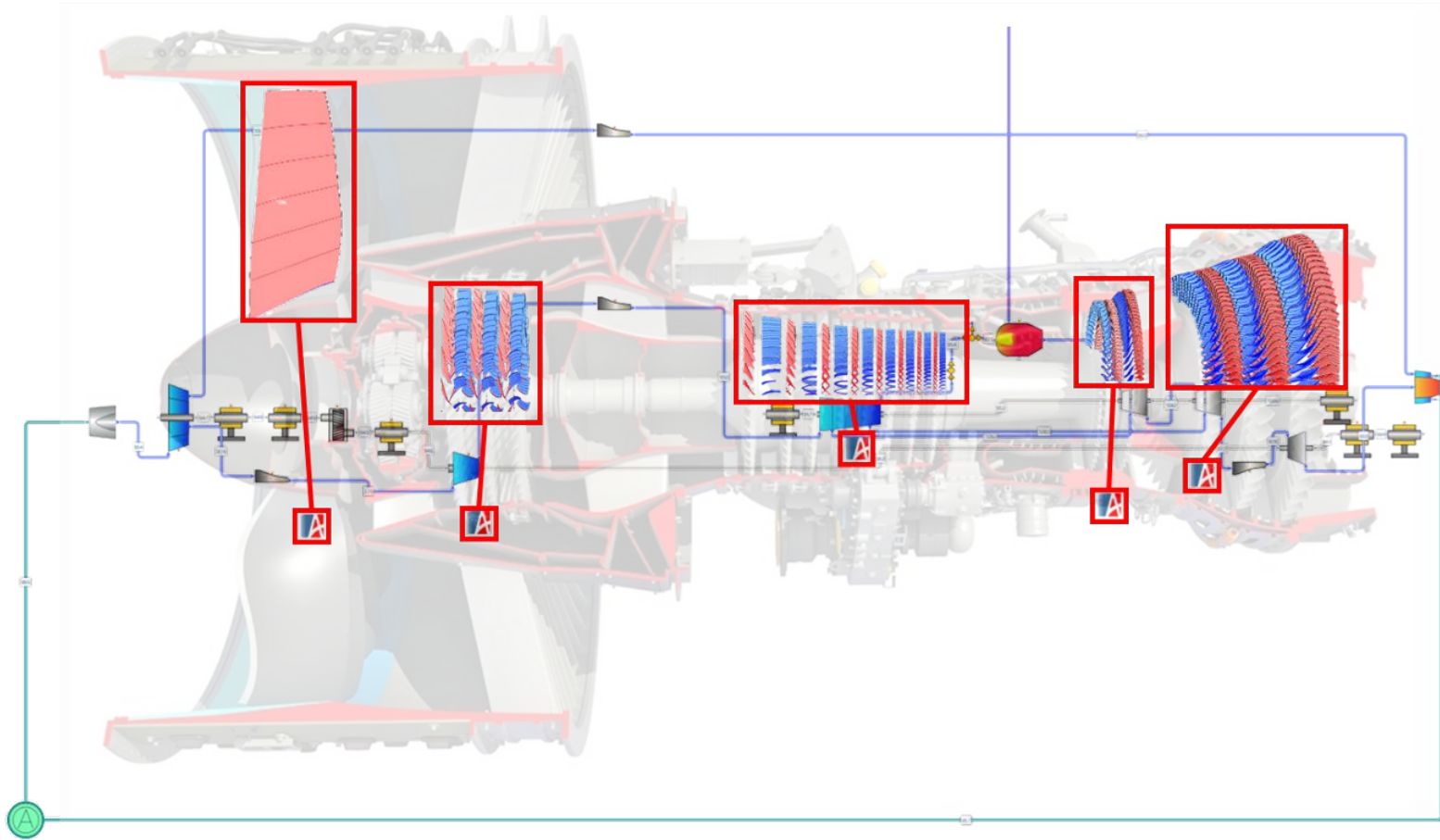
Center for
Research on
Foundation
Models

Bommasani, Rishi, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein et al. "On the opportunities and risks of foundation models." arXiv preprint arXiv:2108.07258 (2021).

FM Pipelines



Scaling Laws as Performance Modeling



Retrieved from <https://www.softinway.com/the-evolution-of-jet-engine-design-methods/>

Scaling Laws as Performance Modeling

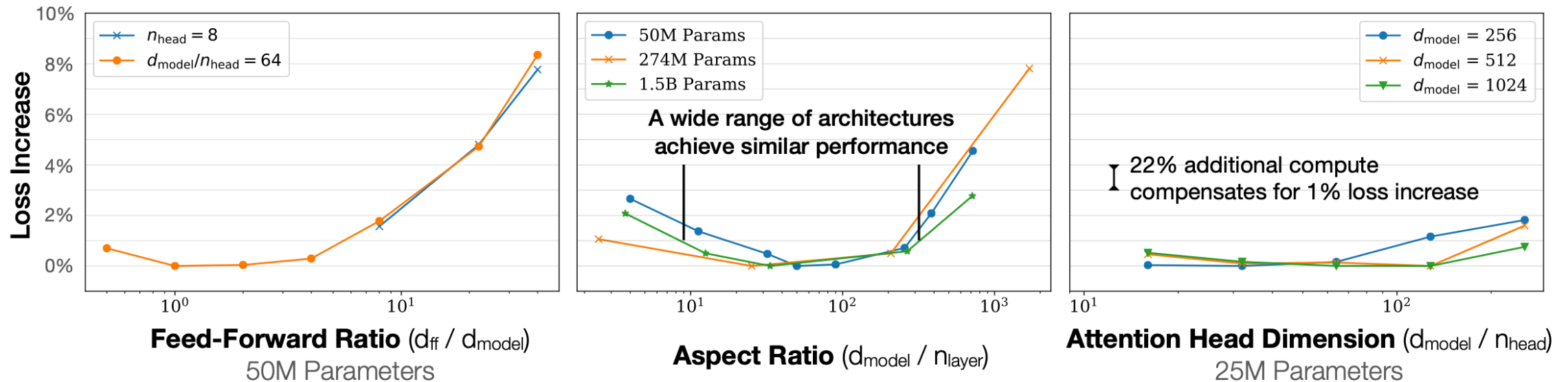


Figure 5 Performance depends very mildly on model shape when the total number of non-embedding parameters N is held fixed. The loss varies only a few percent over a wide range of shapes. Small differences in parameter counts are compensated for by using the fit to $L(N)$ as a baseline. Aspect ratio in particular can vary by a factor of 40 while only slightly impacting performance; an $(n_{layer}, d_{model}) = (6, 4288)$ reaches a loss within 3% of the $(48, 1600)$ model used in [RWC⁺19].

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., ... & Amodei, D. (2020). Scaling laws for neural language models. arXiv preprint arXiv:2001.08361.

Modeling AI(-Heavy) Programs

1. We inherit capabilities and uncertainty—not just datasets

The program must establish evidence for its intended use of an inherited model.

2. Expected scaling gains become dependencies of the engineering program

Resource allocation, architecture, integration, and schedule may depend on performance not yet demonstrated.

3. Test and evaluation must examine the capability-growth claim, not only the current model

A scaling relationship needs enough predictive credibility and enough connection to mission performance to support decisions being made from it.