

Research at AI speed

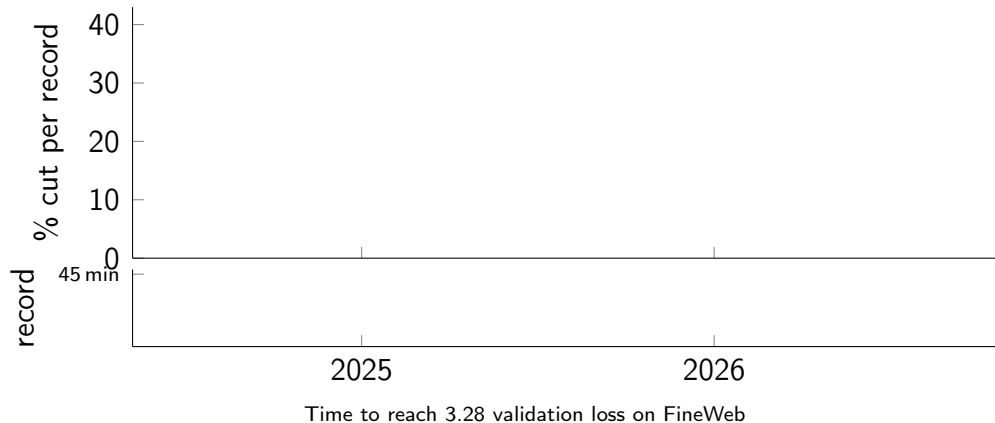
John Langford
Microsoft

With help from *many* folks!

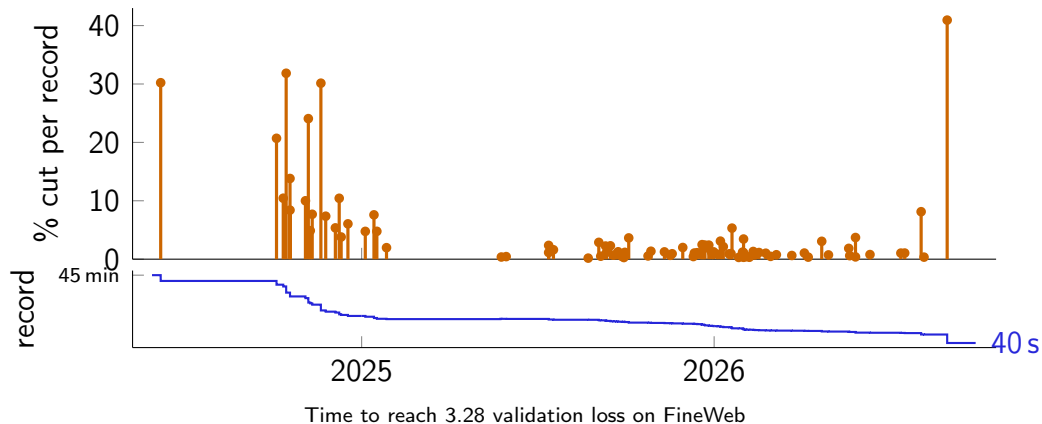


COLM 2026

Modded Nanogpt performance improvements over time



Modded Nanogpt performance improvements over time



Old vs. New research styles

Idea

Old vs. New research styles



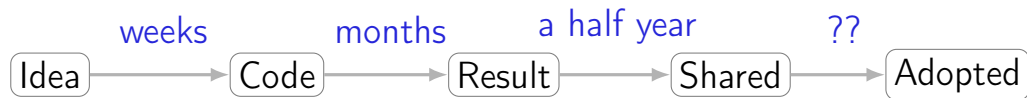
Old vs. New research styles



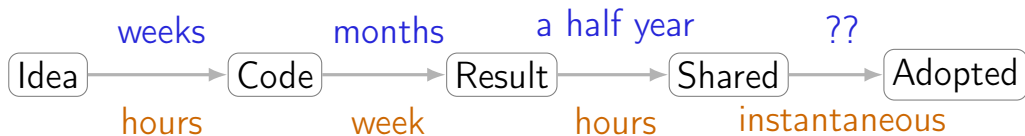
Old vs. New research styles



Old vs. New research styles



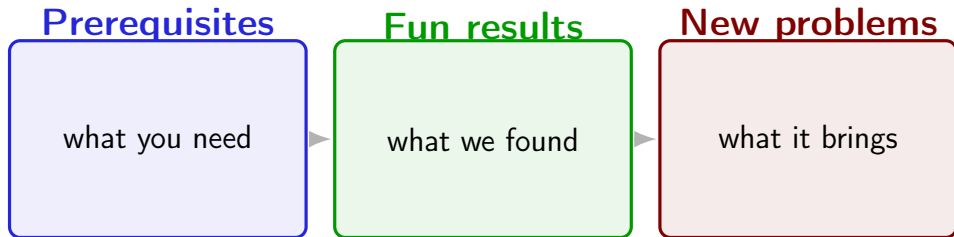
Old vs. New research styles



Speed is a quality.

How can I do something useful?

Three parts



Prerequisite 1: level up on agents

then

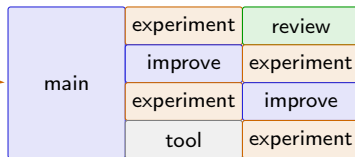
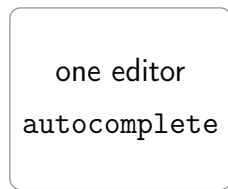
one editor
autocomplete

you

Prerequisite 1: level up on agents

then

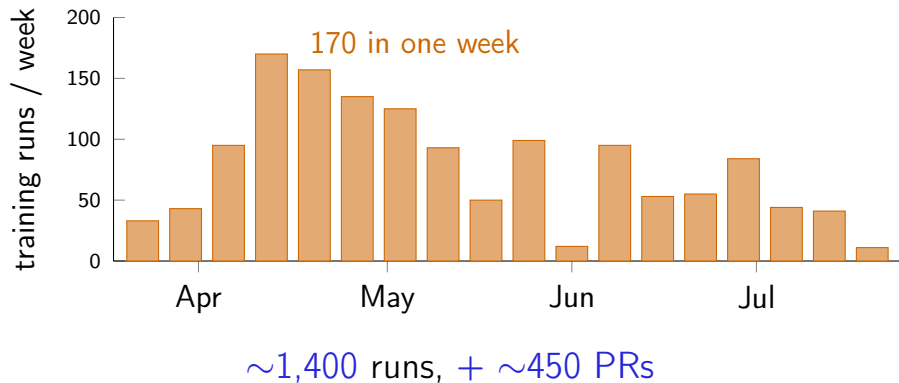
now (see <https://hunch.net/?p=13763837>)



decide, design tests, review

Creative testing
Deinterrupt workflow
Human → agent → tools

Example: a pretraining sprint



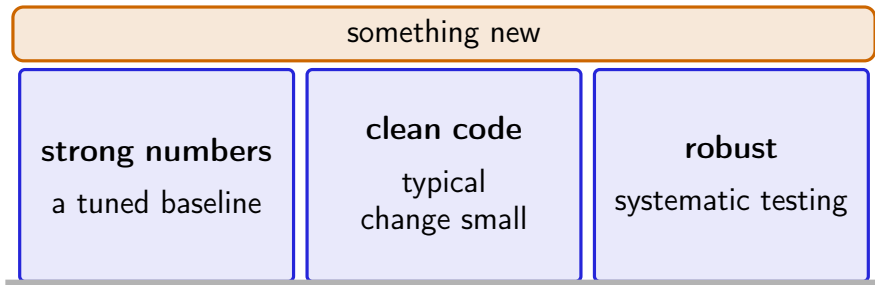
Prerequisite 2: a strong pipeline

strong numbers
a tuned baseline

clean code
typical
change small

robust
systematic testing

Prerequisite 2: a strong pipeline



Where is your strong base?

Pretraining	Sizes	Built by
modded-nanogpt	124M-, 350M-quality	Keller Jordan et al.
SmolLM	135M, 360M, 1.7B, 3B	Hugging Face
Marin	8B, 32B	Stanford, Open Athena

All: public data, exact mixtures, logs.

Should you build your own?

Pros

- Fits your needs best
- Learn the details
- No waiting on others

Should you build your own?

Pros

- Fits your needs best
- Learn the details
- No waiting on others

Cons

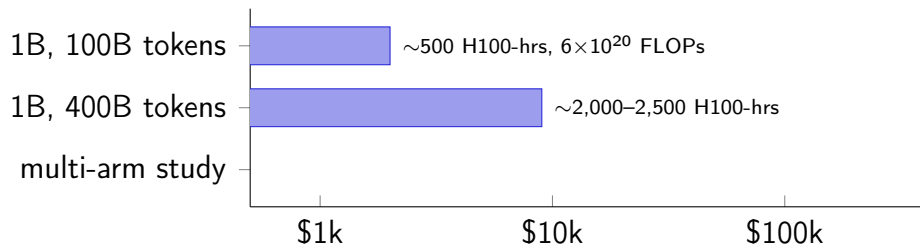
- x10 effort
- x10 expertise
- keeping current

Probably no by default.

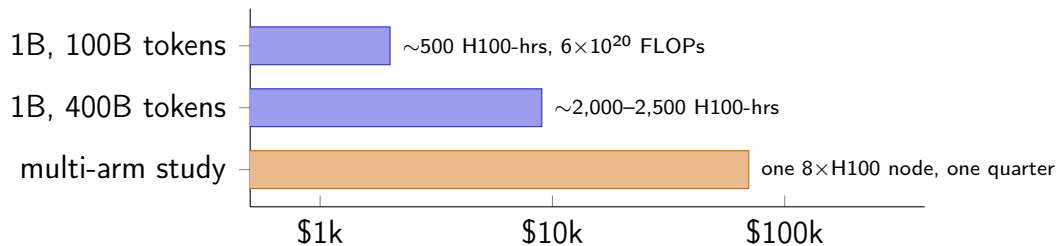
We built one: a 4B on 5T tokens

	KV@32k (GB) ↓	bits per byte ↓			DCLM CORE ↑
		climbmix	web	code	
Ours (unreleased)	0.65	0.657	0.780	0.32	0.57
phi4-4B (unreleased)	12.9	0.790	0.880	0.96	0.55
Qwen3-4B base	4.8	0.641	0.779	0.31	0.57
Qwen3.5-4B base	1.1	0.651	0.789	0.33	0.57
Nemotron-3-Nano-4B	0.5	0.729	0.895	0.39	0.50
Ministral-3-3B	3.5	1.094	1.300	0.51	0.39
OLMo3-7B	5.9	0.633	0.764	0.36	0.54

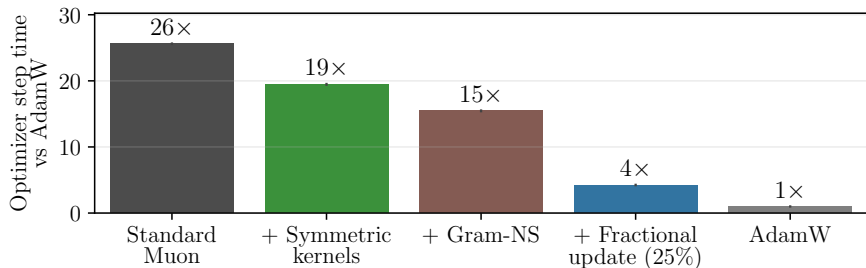
Prerequisite 3: capacity for experiments



Prerequisite 3: capacity for experiments

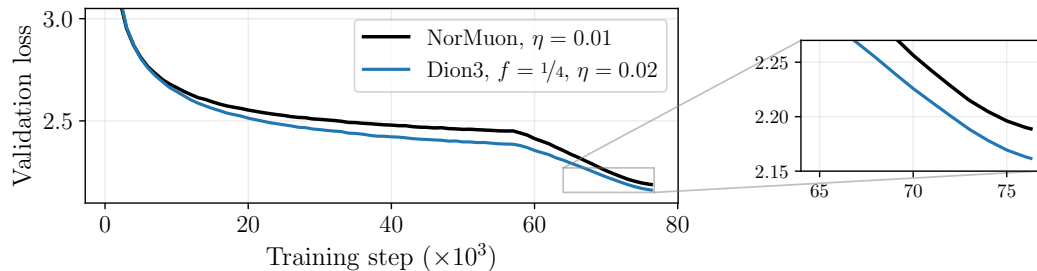


Fun result 1: Dion3



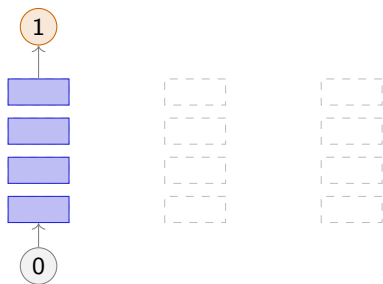
N. Amsel, J. Zhang, K. Ahn, A. Naeimi, A. Feng, B. Chen, T. Dao, J. Langford. "Dion3: Full-stack orthogonal updates." arXiv 2608.11612

A Surprise! lower loss at 14B

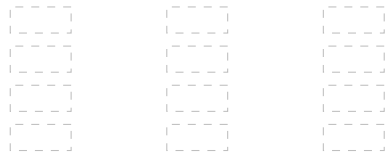


Fun result 2: Full-Bandwidth Transformer

standard decoding



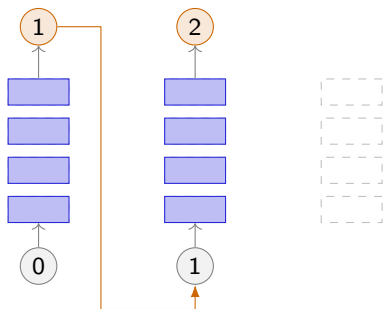
full-bandwidth decoding



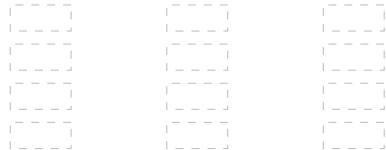
X. Wang, Z. Cai, Z. Zhan, H. Dong, Y. Fan, G. de Rosa, T. Pearce, J. Langford. "Full-bandwidth transformer."
arXiv 2608.08888.

Fun result 2: Full-Bandwidth Transformer

standard decoding



full-bandwidth decoding

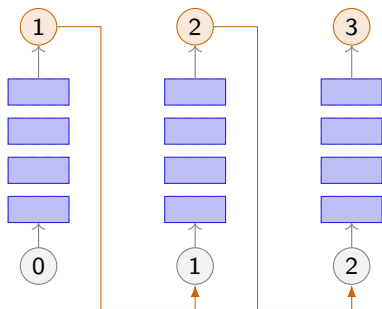


Standard: the next step sees only a **sampled token**.

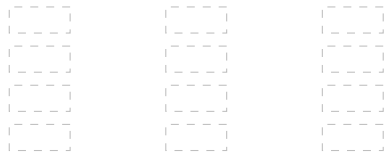
X. Wang, Z. Cai, Z. Zhan, H. Dong, Y. Fan, G. de Rosa, T. Pearce, J. Langford. "Full-bandwidth transformer."
arXiv 2608.08888.

Fun result 2: Full-Bandwidth Transformer

standard decoding



full-bandwidth decoding

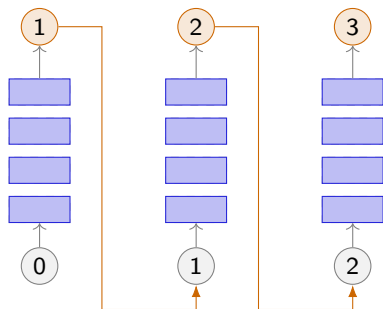


Standard: the next step sees only a **sampled token**.

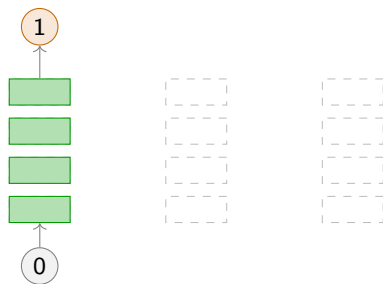
X. Wang, Z. Cai, Z. Zhan, H. Dong, Y. Fan, G. de Rosa, T. Pearce, J. Langford. "Full-bandwidth transformer."
arXiv 2608.08888.

Fun result 2: Full-Bandwidth Transformer

standard decoding



full-bandwidth decoding

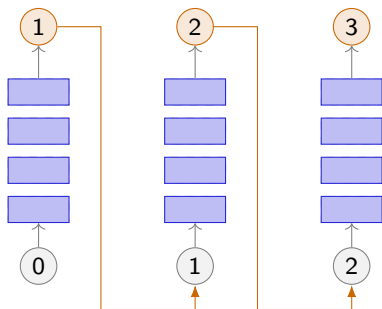


Standard: the next step sees only a **sampled token**.

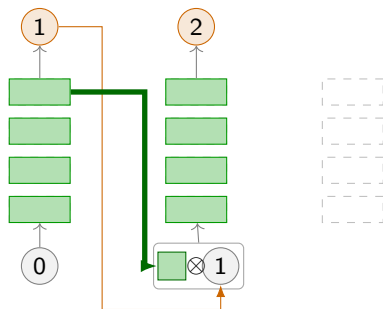
X. Wang, Z. Cai, Z. Zhan, H. Dong, Y. Fan, G. de Rosa, T. Pearce, J. Langford. "Full-bandwidth transformer."
arXiv 2608.08888.

Fun result 2: Full-Bandwidth Transformer

standard decoding



full-bandwidth decoding

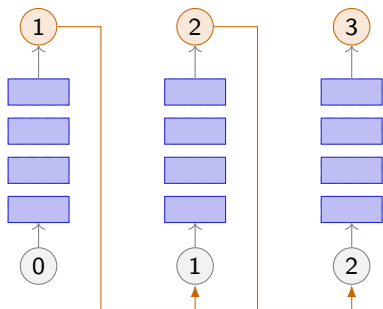


Standard: the next step sees only a **sampled token**.

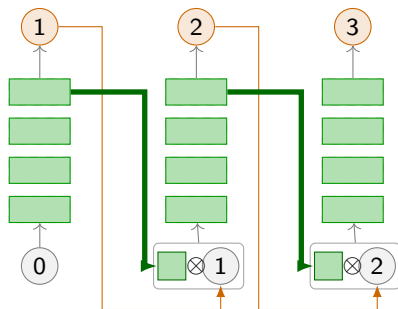
X. Wang, Z. Cai, Z. Zhan, H. Dong, Y. Fan, G. de Rosa, T. Pearce, J. Langford. "Full-bandwidth transformer."
arXiv 2608.08888.

Fun result 2: Full-Bandwidth Transformer

standard decoding



full-bandwidth decoding

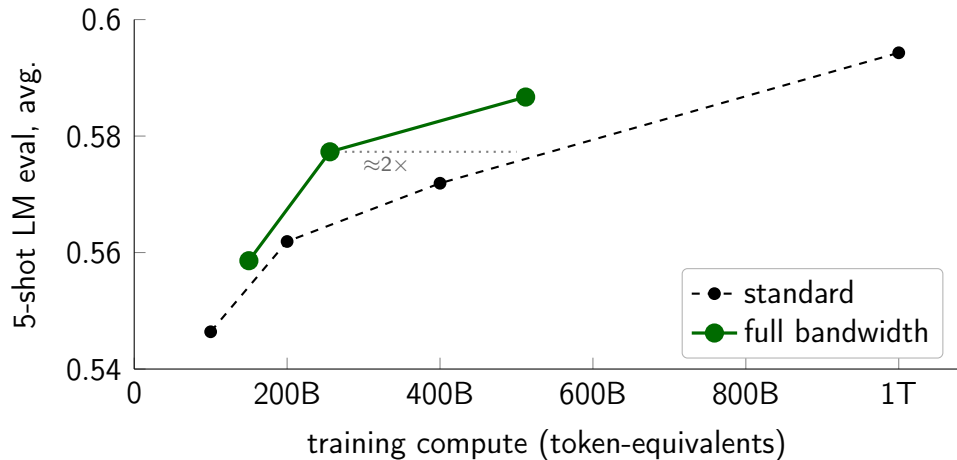


Standard: the next step sees only a **sampled token**.

Full bandwidth: it also sees the **whole top-layer state**.

X. Wang, Z. Cai, Z. Zhan, H. Dong, Y. Fan, G. de Rosa, T. Pearce, J. Langford. "Full-bandwidth transformer."
arXiv 2608.08888.

1B model performance



More concise reasoning

Problem

What is the last nonzero digit to the right of the decimal point in the decimal expansion of $\frac{137}{500}$?

More concise reasoning

Problem

What is the last nonzero digit to the right of the decimal point in the decimal expansion of $\frac{137}{500}$?

Standard decoding (200B model)

To determine the last nonzero digit to the right of the decimal point in the decimal expansion of $\frac{137}{500}$, we start by performing the division of 137 by 500.

First, we set up the division:

$$137 \div 500$$

Since 137 is less than 500, the decimal expansion will start with a zero. [...326 tokens omitted...]

Since the remainder is now 0, the decimal expansion of $\frac{137}{500}$ is:

$$0.274$$

The last nonzero digit to the right of the decimal point is 4.

More concise reasoning

Problem

What is the last nonzero digit to the right of the decimal point in the decimal expansion of $\frac{137}{500}$?

Standard decoding (200B model)

To determine the last nonzero digit to the right of the decimal point in the decimal expansion of $\frac{137}{500}$, we start by performing the division of 137 by 500.

First, we set up the division:

$$137 \div 500$$

Since 137 is less than 500, the decimal expansion will start with a zero. [...326 tokens omitted...]

Since the remainder is now 0, the decimal expansion of $\frac{137}{500}$ is:

$$0.274$$

The last nonzero digit to the right of the decimal point is 4.

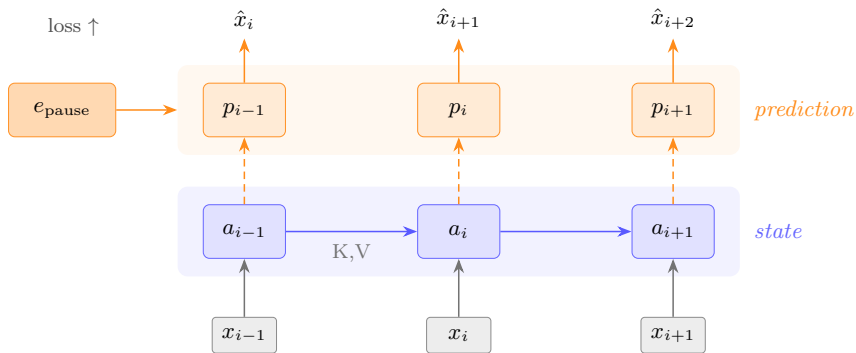
FBT decoding (200B model)

To find the last nonzero digit to the right of the decimal point in the decimal expansion of $\frac{137}{500}$, we can perform the division and observe the pattern of the decimal expansion.

$$\frac{137}{500} = 0.274$$

The last nonzero digit to the right of the decimal point in the decimal expansion of $\frac{137}{500}$ is 4.

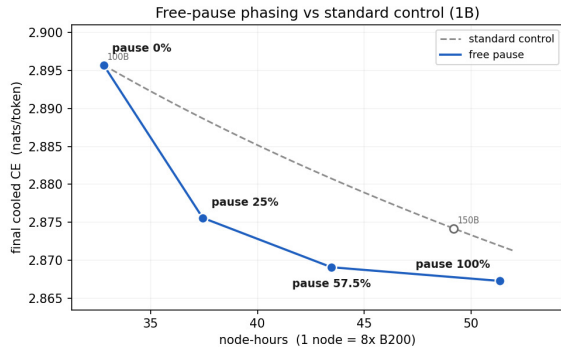
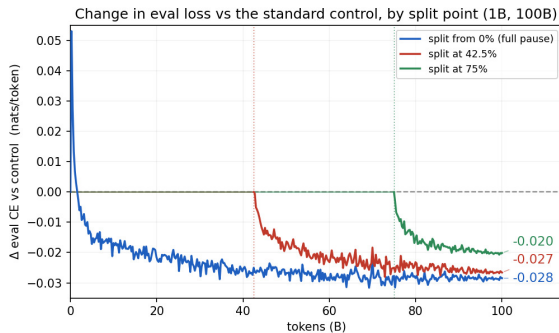
Fun result 3: Almost Free State Prediction Separation



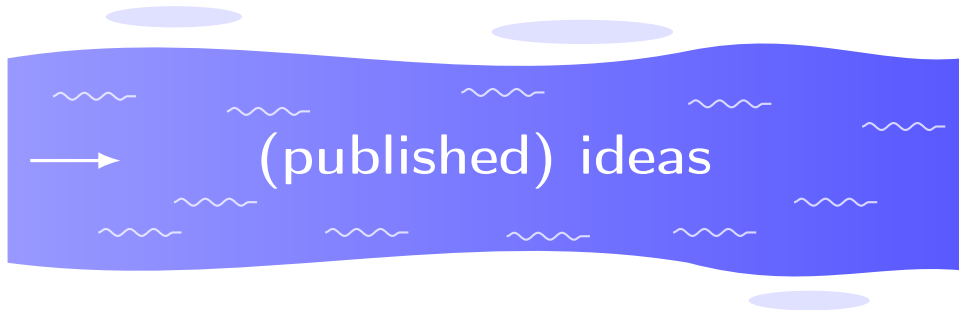
solid = writes/uses K/V dashed = query only, no K/V written p_i attends $a_{\leq i}$ weights shared, $\times L$ layers

J. Langford, N. Godey, G. Monea, Y. Artzi, H. Dong, Y. Fan, G. de Rosa, Z. Zhan. "Almost free state prediction separation." arXiv 2609.03807

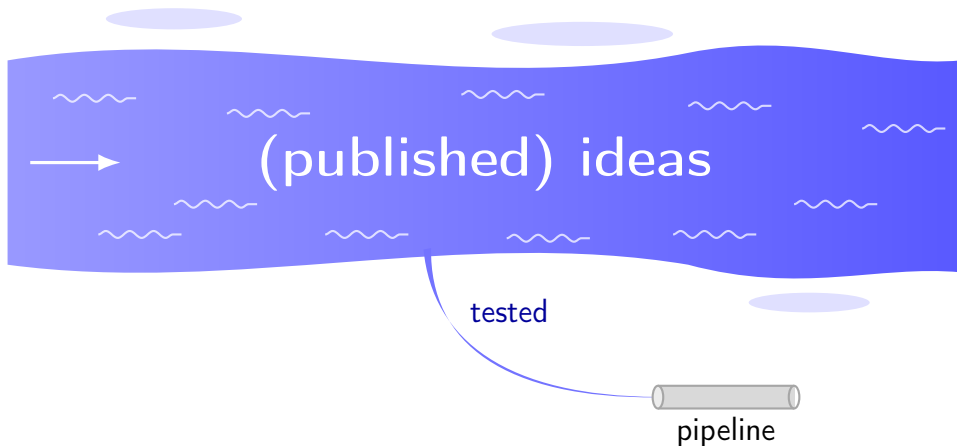
Free pause tokens gain $>$ cost



New problem 1: change stress

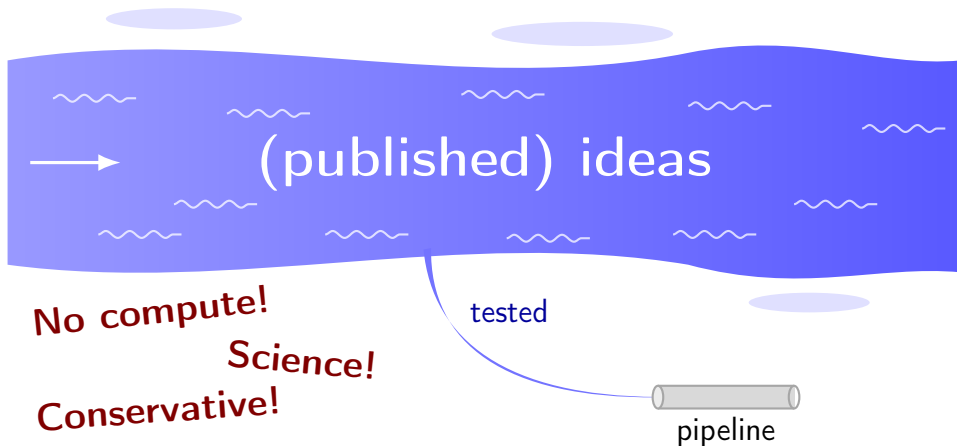


New problem 1: change stress



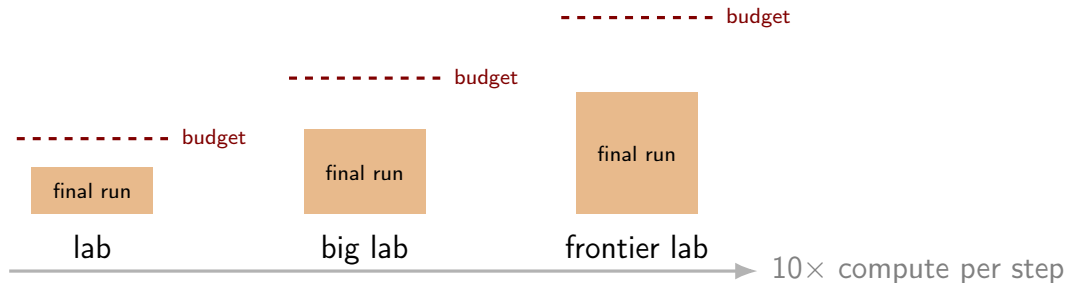
First reaction: **reject change** at the rate research produces it.

New problem 1: change stress

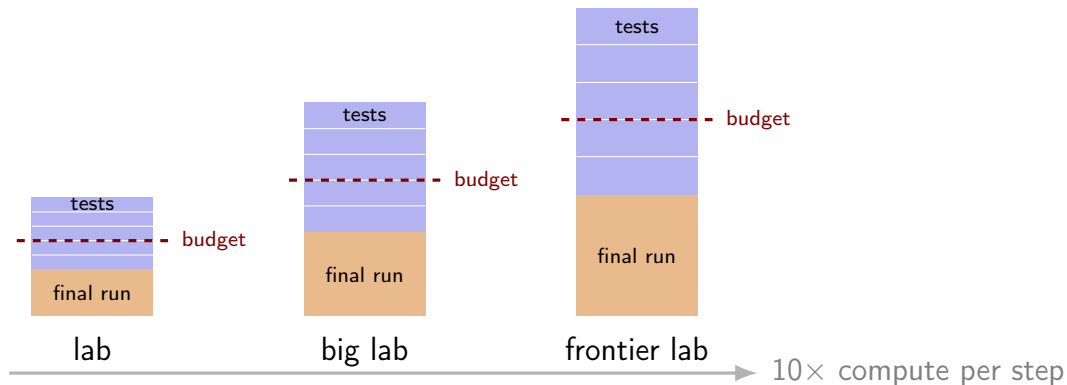


First reaction: **reject change** at the rate research produces it.

"No compute!"

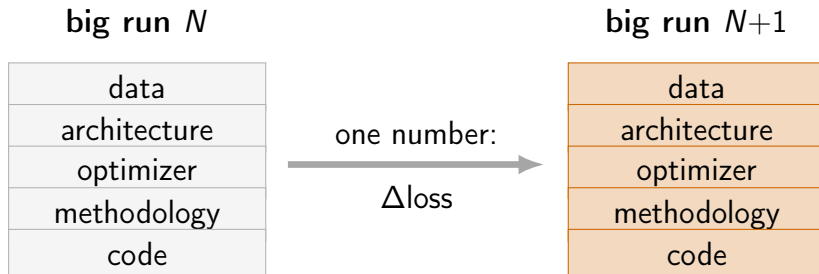


“No compute!”

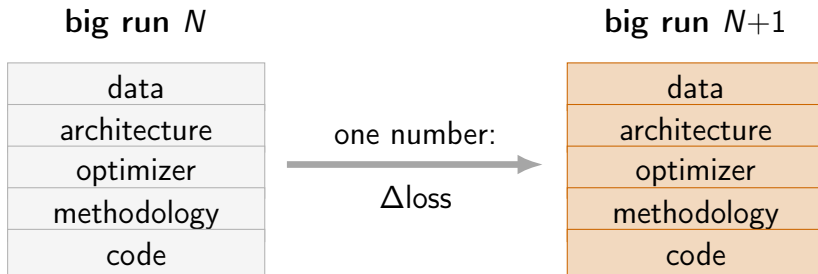


Ambitions scale with compute, and tests far outnumber the final run:
compute is always short.

“Science!”



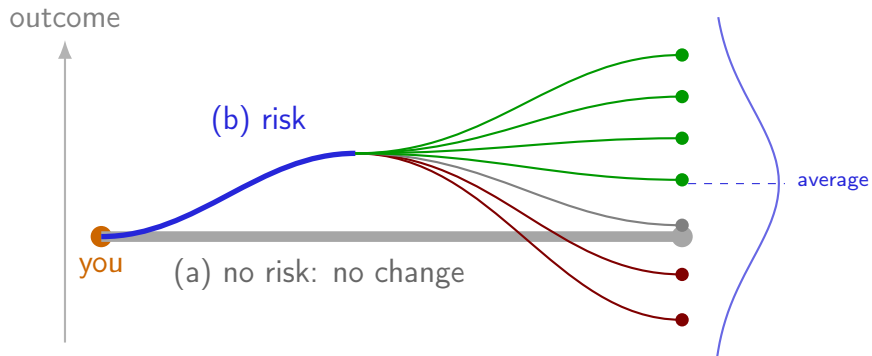
“Science!”



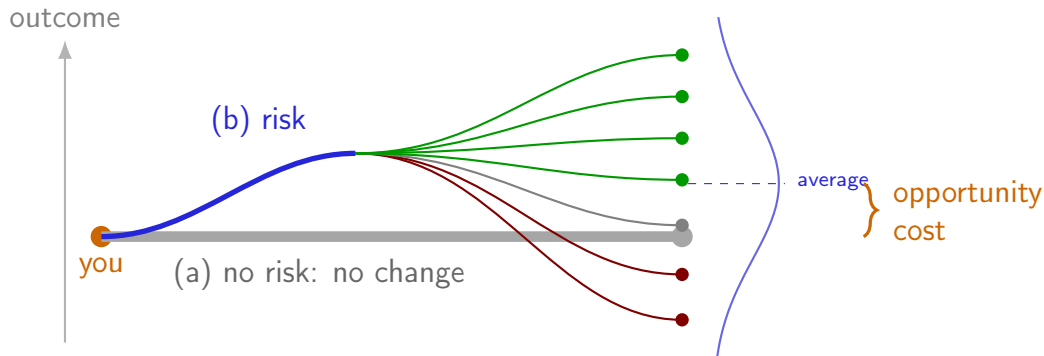
$$\Delta_{\text{loss}} = \Delta_{\text{data}} + \Delta_{\text{arch}} + \Delta_{\text{opt}} + \Delta_{\text{method}} + \Delta_{\text{code}}$$

One measurement, five unknowns.

"Conservative!"

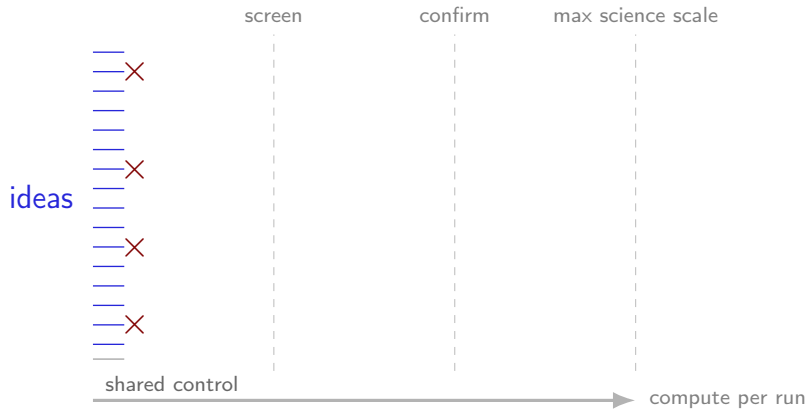


“Conservative!”

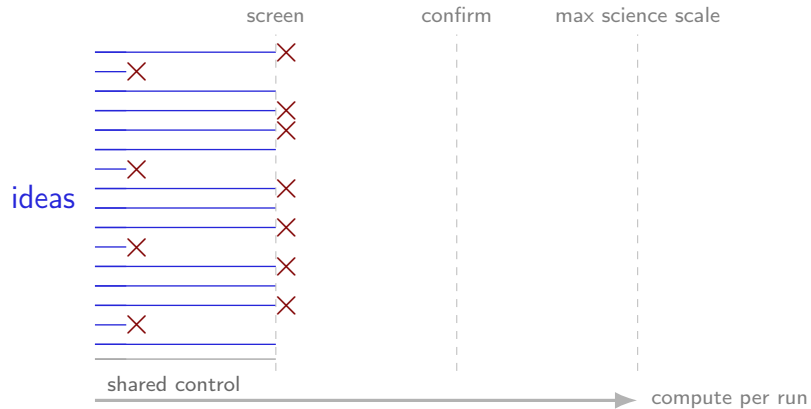


“Conservative” counts the cost of change, **not of standing still.**

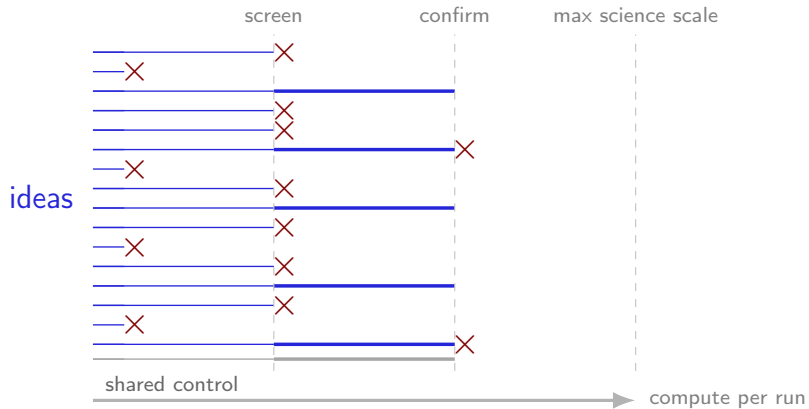
Handling change stress



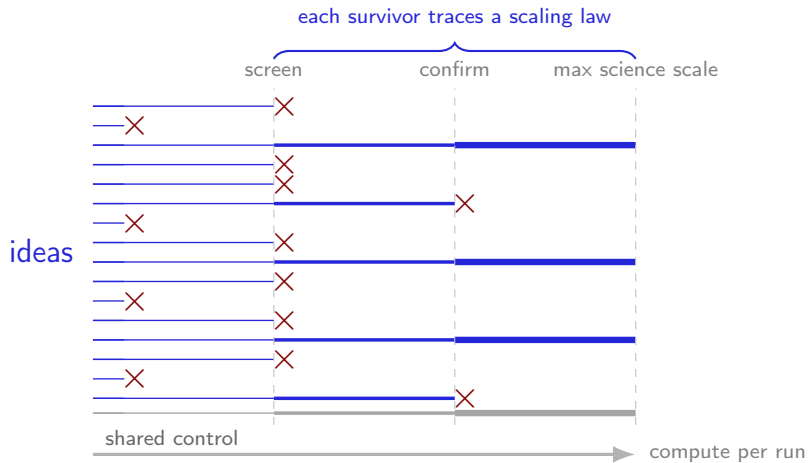
Handling change stress



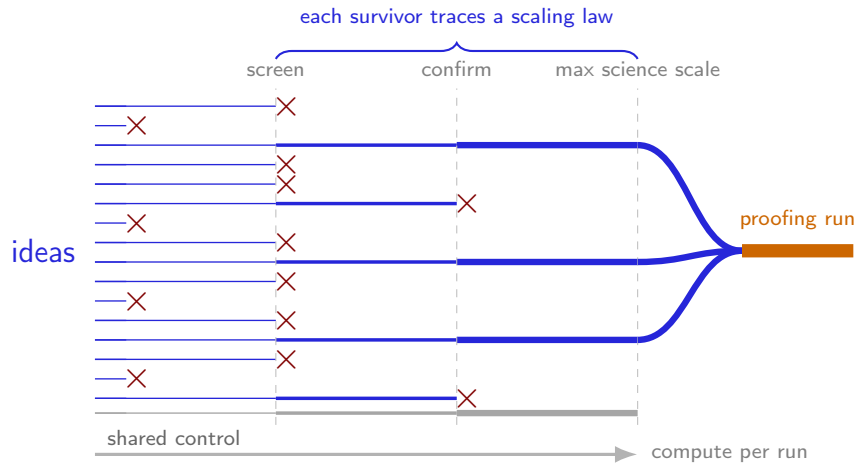
Handling change stress



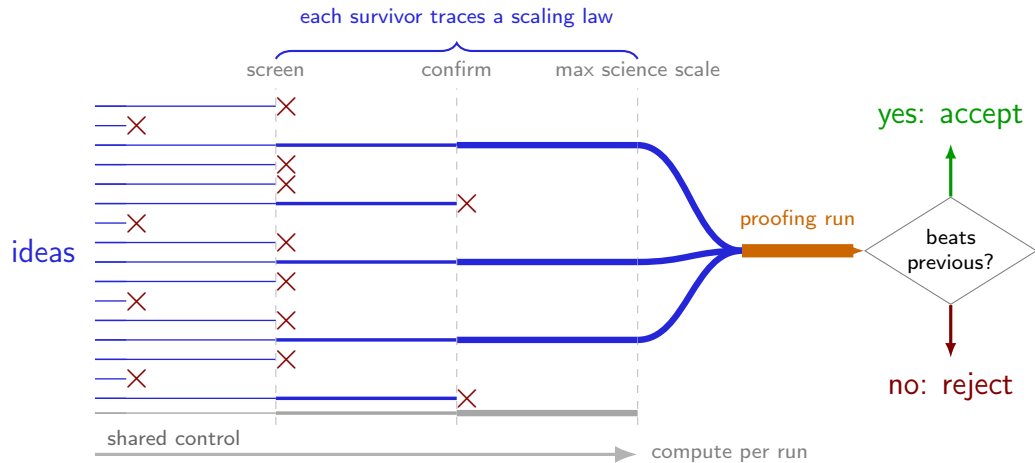
Handling change stress



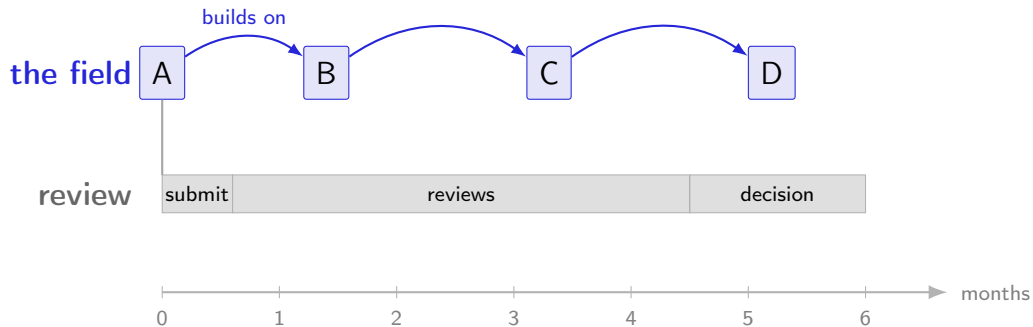
Handling change stress



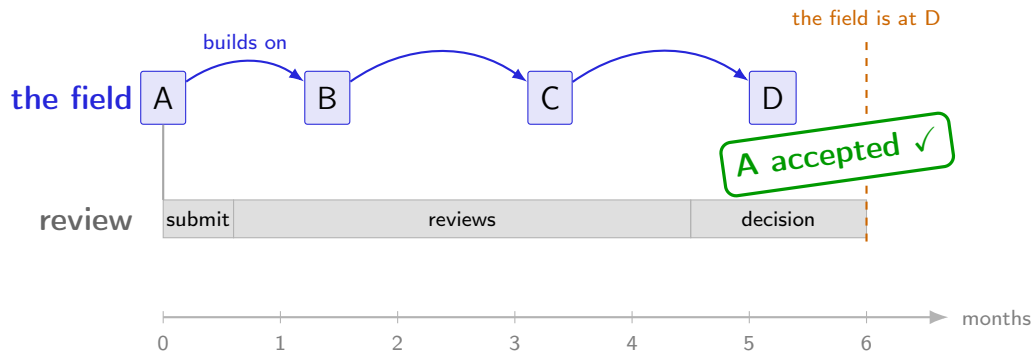
Handling change stress



New problem 2: review stress

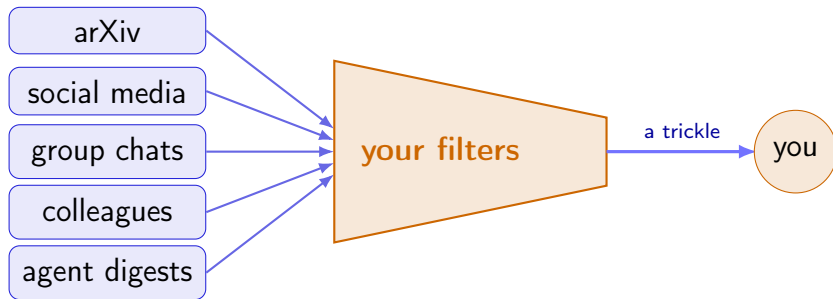


New problem 2: review stress

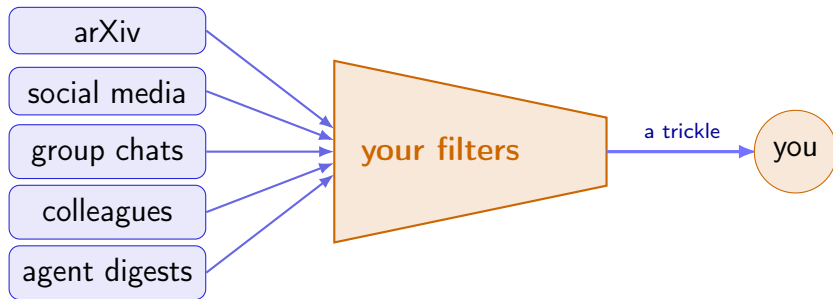


Acceptance **certifies** rather than **directs**.

The real review system

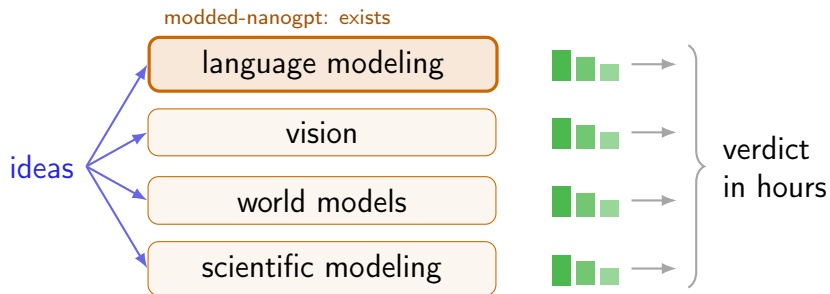


The real review system

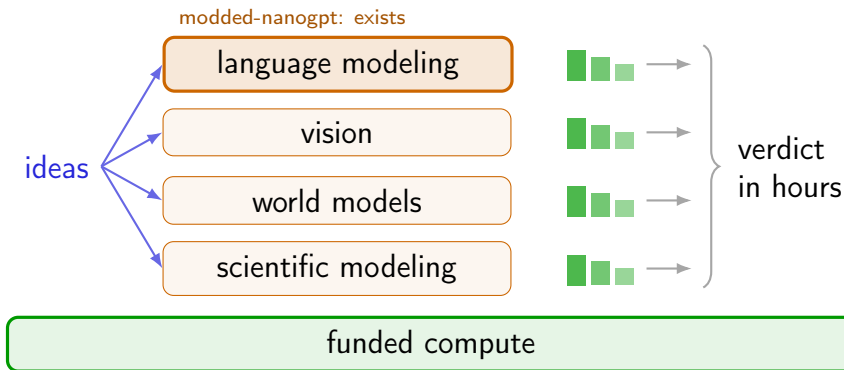


Everyone runs an **impromptu review system**.

An alternative: open benchmark tracks



An alternative: open benchmark tracks



Centralized fast **review for empirical improvements**.
Maybe phase into conferences with a reserved benchmark track?

New problem 3: meeting the prerequisites

Level up on agents?

A strong pipeline?

Capacity for
experiments?

New problem 3: meeting the prerequisites

Level up on agents? Yes.

A strong pipeline?

Capacity for
experiments?

New problem 3: meeting the prerequisites

Level up on agents?

Yes.

A strong pipeline?

Use and support it.

Capacity for
experiments?

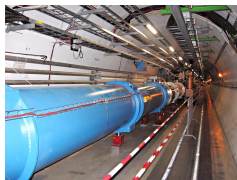


Photo: Julian Herzog, CC BY-SA 3.0, via Wikimedia Commons.

New problem 3: meeting the prerequisites

Level up on agents?

Yes.

A strong pipeline?

Use and support it.

Capacity for experiments?

Balance compute against *salary*, not travel.

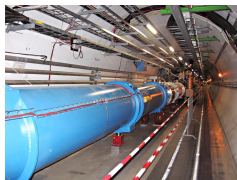


Photo: Julian Herzog, CC BY-SA 3.0, via Wikimedia Commons.

How to do something useful



Thanks!



Kwangjun Ahn



Noah Amsel



Yoav Artzi



Ziyang Cai



Berlin Chen



Tri Dao



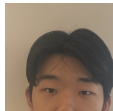
Gustavo de Rosa



Harry Dong



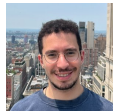
Ying Fan



Austin Feng



Nathan Godey



Giovanni Monea



Ali Naeimi



Tim Pearce



Jayden Teoh



Xi Wang



Zheng Zhan



Jack Zhang