

Scaling Laws and Resource Trade-offs

What does a resource curve predict—and what does it leave out?

Complexity, Information, and Cross-Domain Transfer

Why make allocation—not model size—the question?

Larger language models had often been trained on roughly similar token counts. Under fixed compute, more parameters leave fewer tokens: apparent capacity gains can conceal undertraining.

Hoffmann et al. motivate varying both. The intellectual move is to compare allocations on one budget before treating size as the independent cause of progress.

Generate candidates by conserving the scarce resource

Candidate allocation	Parameters $N \times$	Tokens $D \times$
More capacity; fewer tokens	2	$\frac{1}{2}$
Keep current balance	1	1
Less capacity; more tokens	$\frac{1}{2}$	2

Assume compute is proportional to $N \times D$; keep architecture and data quality comparable. Which allocation wins requires evidence.

Model size and data compete for training compute

$$L(N,D)=E_0+A N^{-\alpha}+B D^{-\beta}; \quad C=\kappa ND$$

A simplified loss model separates parameter and data effects. Assume $A, B, \alpha, \beta, \kappa > 0$ and compute $C = \kappa ND$.

N is parameter count; D is training tokens; E_0 is a fitted loss floor. This model concerns language-model loss, not validated discoveries.

Deriving the optimal allocation

$$\alpha A N^{-\alpha} = \beta B D^{-\beta}$$

Substitute $D=C/(\kappa N)$. Differentiating the reducible loss gives a balance between marginal parameter and data gains.

For positive constants and exponents, N^* scales as $C^{\beta/(\alpha+\beta)}$ and D^* as $C^{\alpha/(\alpha+\beta)}$.

Kaplan and Hoffmann fit different regimes

Fit	N exponent	D exponent	Method
Kaplan §6	.73	.27	Batch-adjusted frontier
Hoffmann §3.1	.50	.50	Training-curve envelope
Hoffmann §3.2	.49	.51	IsoFLOP minima
Hoffmann §3.3	.46	.54	Parametric loss fit

Both optimize language-model loss under training compute, but their data, schedules and fitted frontiers differ. These exponents describe allocation, not loss decay.

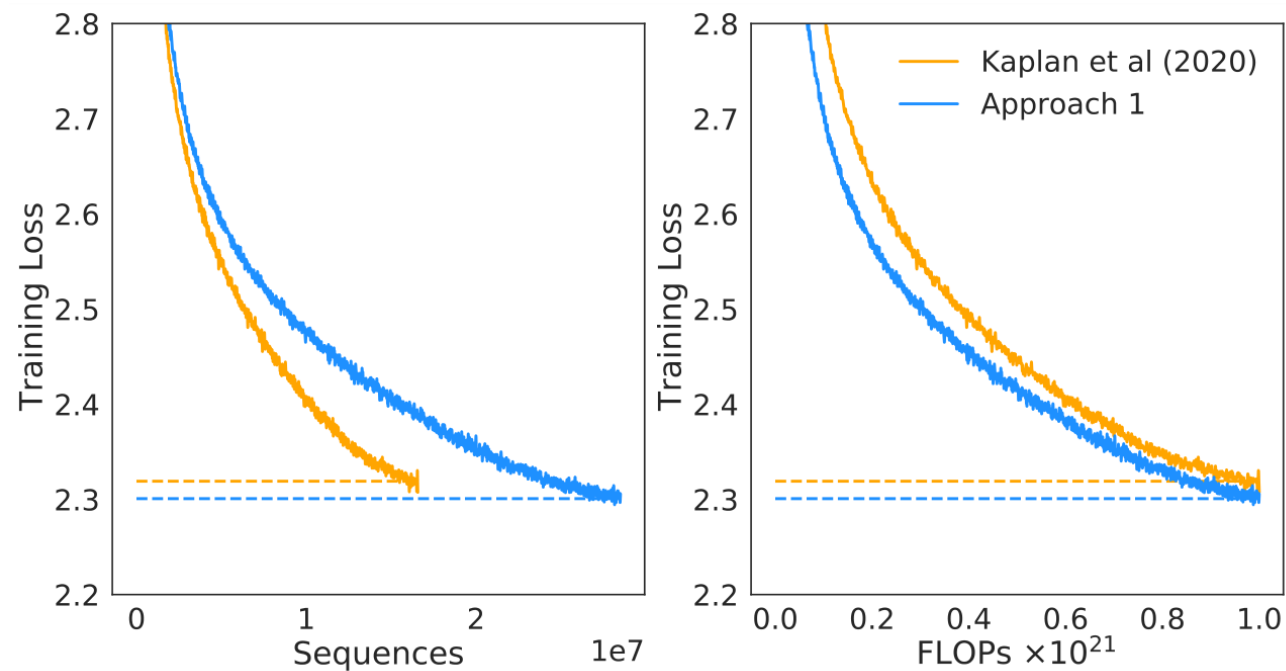
What was varied, measured and extrapolated?

Hoffmann's study trained over 400 models: roughly 70M–16B parameters and 5B–500B tokens. It matched learning-rate decay to training duration.

Besiroglu et al. reconstructed plotted data and refitted the parametric model. Their estimates differ from the reported fit and agree better with the other two allocation approaches.

This checks estimation from existing runs. The next comparison tests allocation at a matched training budget.

A head-to-head test at the same training budget



At 10^{21} FLOPs, the 2.80B-parameter model reaches lower training loss than the 4.74B model. Matched compute reverses the advantage suggested by the sequences axis.

What a sixteenfold compute increase implies

$$N \text{ multiplier} = 16^a; \quad D \text{ multiplier} = 16^{(1-a)}$$

Using the fitted allocation powers, $16\times$ compute predicts about $7.57\times$ parameters and $2.11\times$ tokens under $(0.73, 0.27)$.

Using $(0.49, 0.51)$ predicts about $3.89\times$ parameters and $4.11\times$ tokens. The product remains 16.

Hoffmann fitted hundreds of runs spanning roughly 70M–16B parameters and billions of tokens, then tested a larger allocation with Chinchilla.

How does measurement error scale with evidence?

$$E[(\bar{X} - \mu)^2] = \sigma^2/n; \quad \text{RMSE} = \sigma/\sqrt{n}$$

Independent measurements with common mean μ and variance σ^2 give an unbiased sample mean $\bar{X}$. Four times as many measurements halve its root mean squared error (RMSE).

The exponent is 1 for squared error and $\frac{1}{2}$ for RMSE. The outcome and assumptions determine the scaling claim before we choose a resource budget.

A resource vector precedes a scalar budget

$$\mathbf{b} = (C, n, E, R, T)$$

Compute C , data n , experimental count E , adaptive rounds R and time T can have different constraints.

A scalar budget requires prices or weights. A laboratory may have a binding time constraint even when compute is inexpensive.

A power law with an irreducible floor

$$L(C) = L_{\infty} + a C^{-\alpha}$$

A common empirical model is loss $L(C)=L_{\infty}+aC^{-\alpha}$, where $a>0$ and $\alpha>0$.

Doubling compute multiplies the reducible component by $2^{-\alpha}$. It does not multiply total loss by that factor when L_{∞} is positive.

Small exponents imply expensive improvements

$$C_{\text{new}} / C_{\text{old}} = 2^{1/\alpha}$$

With $\alpha=0.1$, halving reducible loss requires compute multiplied by $2^{10}=1,024$. With $\alpha=0.5$, the multiplier is four.

The exponent matters, but so do the floor and the cost of changing the method rather than only adding compute.

Search success can saturate without a power law

$$Q(n+1) - Q(n) = p(1-p)^n$$

For independent attempts with success probability p , $Q(n) = 1 - (1-p)^n$. The expected gain from one more attempt is $p(1-p)^n$.

For a fixed task and generator this gain decreases with n , even when computation per attempt stays constant.

Heterogeneous tasks change the aggregate curve

$$E[(1-p)^n] \geq (1-E[p])^n$$

For task-specific success probability p , average failure is $E[(1-p)^n]$. This is generally not $(1-E[p])^n$.

At $n \geq 2$, convexity implies heterogeneity raises average failure relative to replacing p with its mean.

A worked heterogeneity example

$$Q_{\text{mixture}}(2)=0.59 \text{ versus } Q_{\text{mean}}(2)=0.75$$

Half the tasks have $p=0.9$ and half have $p=0.1$. Two attempts yield average success $0.5 \times 0.99 + 0.5 \times 0.19 = 0.59$.

Using the average $p=0.5$ would predict 0.75. The same mean one-attempt score hides different scaling.

Best observed score can rise through noise

More samples increase the maximum observed score even when candidate quality is unchanged.

To infer real improvement, evaluate the selected candidate independently or use an acceptance rule that accounts for selection.

Discovery yield is a pipeline quantity

$$E[Y_{\text{useful}}] = n \times p_s \times p_v \times p_u$$

Let n be proposals, p_s screening probability, p_v validation conditional on screening and p_u usefulness conditional on validation.

Expected useful yield is $n p_s p_v p_u$. Each factor can depend on the resource allocation and the selected distribution.

Better average prediction need not improve tail selection

Model A may be more accurate over common easy examples while ranking the rare valuable region worse than model B.

Discovery requires the conditional performance of selected candidates, not only average error on a broad test distribution.

2025–2026 perturbation evidence sharpens the distinction

Ahlmann-Eltze and colleagues compare deep perturbation predictors with simple baselines. Systema examines how systematic variation can inflate apparent prediction performance.

These studies caution against inferring biological understanding from aggregate expression-prediction metrics.

A budget allocation between proposal and validation

$$\text{tested candidates} \leq \min[m(c), e/ka]$$

Suppose compute spending c gives $m(c)$ candidates and assay spending e tests at most e/ka candidates, where ka is cost per assay.

If candidate quality stays fixed, yield is constrained by $\min[m(c), e/ka]$. Excess proposals can be worthless when assay capacity binds.

Improving the screen can move the bottleneck

A more selective screen can increase validation per assay while reducing coverage of unusual candidates.

After screening improves, synthesis capacity, confirmatory testing or interpretation may become the limiting stage.

A resource frontier is a set of feasible outcomes

$$Q^*(b) = \sup_{\pi \text{ feasible}} Q(\pi, b)$$

For each resource vector b , define $Q^*(b)$ as the best attainable success probability over an explicitly allowed policy class.

At the same accepted outcome, A costs 100 and takes 10 days; B costs 150 and takes 6; C costs 180 and takes 12. C is dominated; choosing A or B requires a budget or a value of time.

Observed performance can be nonmonotonic

More search can overfit a surrogate, select noise or consume resources needed for confirmation. More data can shift the training distribution.

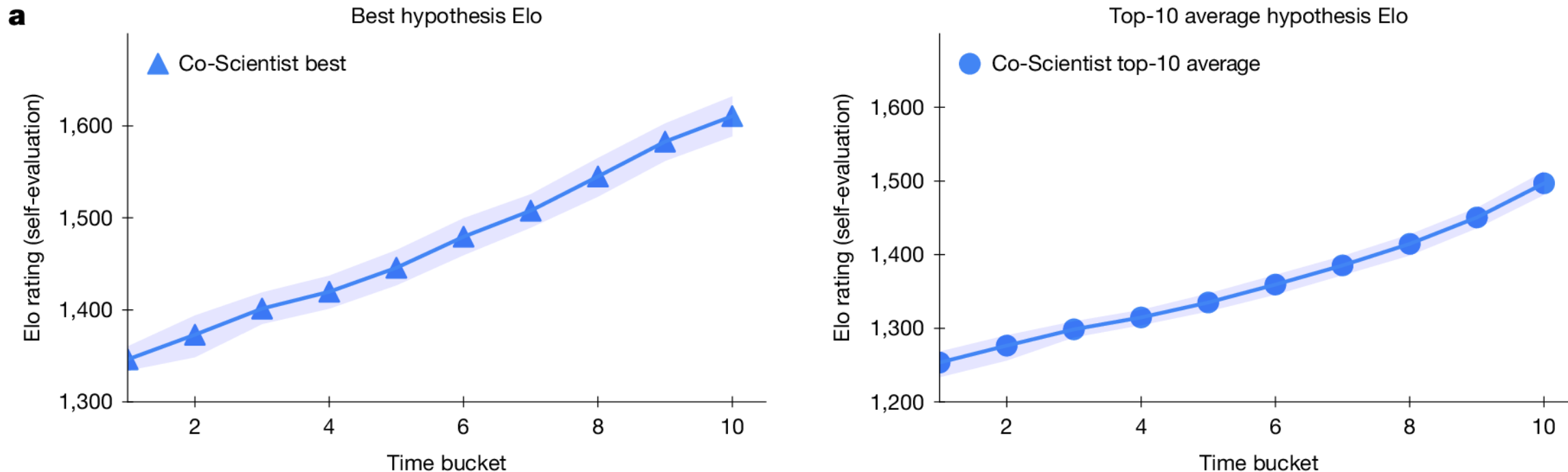
If every extra resource is optional and the true optimum is defined correctly, $Q^*(b)$ is nondecreasing; a particular procedure need not be.

Co-Scientist's curve measures hypothesis ratings

Figure 2 in the 2026 paper tracks automatic Elo ratings across generated hypotheses and separate expert evaluations.

Those curves support claims about measured hypothesis quality under that evaluation. They do not estimate the number of independently validated discoveries per unit compute.

Co-Scientist: what the scaling curves measure



Across 203 research goals, later time buckets have higher internal hypothesis Elo ratings. These curves measure the system's comparative auto-evaluation, not the number of independently validated discoveries.

A 2026 result with an explicit search budget

$$N = C L K = 32 \times 100 \times 16 = 51,200$$

SimpleTES's pruning study uses $C=32$ chains, $L=100$ rounds and $K=16$ candidates: 51,200 nominal evaluator queries per unpruned run.

Across six mathematical tasks and three runs each, retaining one chain at round 25 preserved the original best score in 10 of 18 runs.

Pruning makes the resource–retention trade-off visible

Chains kept / cutoff	Nominal queries	Best chain retained
Keep 8 after round 12	17,408	11 / 18 runs
Keep 8 after round 25	22,400	14 / 18 runs
Keep 16 after round 12	28,672	12 / 18 runs
Keep 16 after round 25	32,000	16 / 18 runs

The full run uses 51,200 queries. More nominal queries do not guarantee more retained winners across these policies.

A 2026 result with its resource denominator

TTT-Discover, TriMul ablation: best H100 kernel runtime is 1,203.10 μ s with the full method and 2,060.70 μ s with reuse but no training.

The experiment matches rollout counts, not training FLOPs, token lengths or verification cost. It reports a best kernel per run.

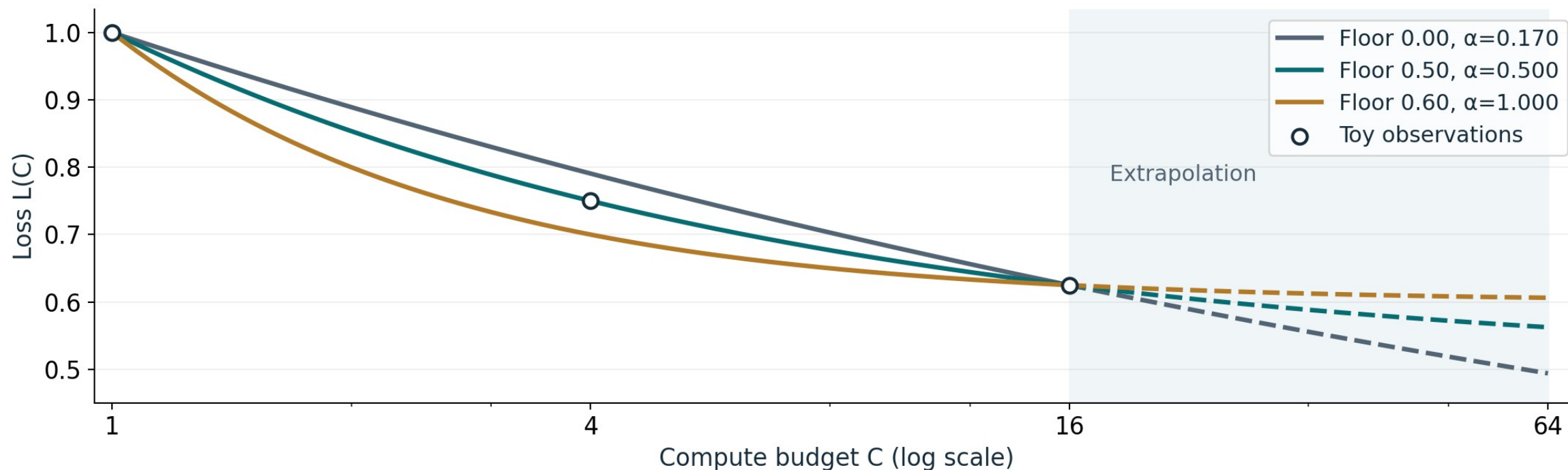
Table 9 specifies $50 \text{ steps} \times 512 \text{ rollouts} = 25,600$. Figure 4 prints 256,000 for the same product, an internal caption inconsistency.

The same endpoints permit different fitted exponents

Assumed floor E_0	Fitted α	Predicted $L(4)$	Predicted $L(64)$
0	.1695	.7906	.4941
.50	.5000	.7500	.5625
.60	1.0000	.7000	.6063

All three curves match $L(1)=1$ and $L(16)=.625$. The held-out middle observation $L(4)=.750$ discriminates among these exact toy fits.

Floor assumptions change the extrapolation



Dots are the three toy observations. Curves fit only $C=1$ and $C=16$; dashed portions extrapolate beyond the observed range. The .50 floor fits the middle point here. Three points still provide little evidence for a long-range law.

What would establish a discovery scaling regularity?

Measure a stable scientific endpoint over a representative task family while varying resources and preserving the validation rule.

Replicate across regimes, explain exceptions, and test whether a proposed mechanism predicts the response to a changed allocation.

Conclusion: scaling is conditional on a process

Resources improve outcomes through particular mechanisms: coverage, learning, information, validation and parallel execution.

Next: when an improvement in one problem can be transferred to another, and what each kind of transfer must preserve.

References

Jared Kaplan et al. (2020). Scaling Laws for Neural Language Models. arXiv:2001.08361v1; preprint

Haotian Ye et al. (2026). Structured Scaling of AI Discovery Across Diverse Scientific Domains. arXiv:2604.19341v2; preprint, 27 July 2026

Jordan Hoffmann et al. (2022). Training Compute-Optimal Large Language Models. NeurIPS 2022; conference paper; arXiv v1

Tamay Besiroglu et al. (2024). Chinchilla Scaling: A replication attempt. arXiv:2404.10102v1; preprint, 15 April 2024

References · continued

[Tatsunori Hashimoto \(2025\). CS336 Lecture 9: Scaling Laws — Basics. Stanford University, Spring 2025; lecture slides](#)

[Juraj Gottweis et al. \(2026\). Accelerating scientific discovery with Co-Scientist. Nature 655, 487–496; journal article](#)

[Ramon Viñas Torné et al. \(2025/2026\). Systema: a framework for evaluating genetic perturbation response prediction beyond systematic variation. Nature Biotechnology 44, 1050–1059 \(2026\); online 2025](#)

[Constantin Ahlmann-Eltze, Wolfgang Huber and Simon Anders \(2025\). Deep-learning-based gene perturbation effect prediction does not yet outperform simple linear baselines. Nature Methods 22, 1657–1661; journal article](#)

References · continued

Martin Pacesa et al. (2025). One-shot design of functional protein binders with BindCraft. Nature 646, 483–492; online 27 August 2025

Joseph L. Watson et al. (2023). De novo design of protein structure and function with RFdiffusion. Nature 620, 1089–1100; journal article

Mert Yuksekgonul et al. (2026). Learning to Discover at Test Time. arXiv:2601.16175v2, 5 February 2026; preprint