BioCoLoop: Collaborative Agentic Research for Biological Model Improvement

Anonymous authors
Paper under double-blind review

Abstract

Artificial intelligence is increasingly used to automate biological model development. Recent self-improving agents advance this process by iteratively proposing model changes, training and evaluating them, and using the observed results to guide subsequent proposals. However, biological data are often distributed across laboratories, where data-sharing constraints can prevent centralized pooling and limit access to training and evaluation. We introduce BioCoLoop, a collaborative research framework that extends collaboration from parameter fitting to model development itself. BioCoLoop collaboratively trains and evaluates candidate model configurations across laboratories and uses laboratory-local development evidence, alongside model updates, as a shared research signal. An inner loop fits and evaluates candidates; an outer loop uses the accumulated evidence to guide subsequent proposals. A separate allocation policy uses early evaluation trajectories to determine which models receive further training. Across drug–target interaction prediction, observed-response proteomic efficacy prediction and cell perturbation identification, BioCoLoop improves most primary endpoints. Development-history feedback changes search trajectories and can reach useful configurations earlier, while evidence-guided allocation improves performance under the same training and evaluation budget.

1. Introduction

Artificial intelligence supports biological research through predictive modeling and the generation and evaluation of hypotheses (Cui et al., 2024; Theodoris et al., 2023; Roohani et al., 2024). Self-improving agents automate parts of model development (Jiang et al., 2025b; Lu et al., 2024; Zhou et al., 2026) through an iterative process of proposing, testing and revising (Madaan et al., 2023; Shinn et al., 2023; Huang et al., 2023). An agent proposes changes to a model design or its training hyperparameters, trains and evaluates the candidate, and uses the results to inform the next proposal.

In biology, useful data are often distributed across laboratories studying different cell types, perturbations or experimental assays. Data-sharing constraints can prevent these measurements from being pooled (Sheller et al., 2020; Bakhtiari et al., 2025; Dayan et al., 2021; Kaissis et al., 2020; Rieke et al., 2020). Collaborative learning enables laboratories to train a shared model through locally computed updates (McMahan et al., 2017; Wang et al., 2024; Li et al., 2020; Karimireddy et al., 2020; Kairouz et al., 2021). Conventional protocols commonly specify the model design and training hyperparameters before fitting begins, limiting how local biological and experimental differences inform subsequent design choices.

Local development evidence provides a complementary research signal: it records how candidate configurations perform, how their training trajectories evolve and which trials succeed or fail. Sharing these summaries allows laboratories to inform subsequent experiments while retaining their measurements locally. *Can an iterative research process use laboratory-local training and evaluation to guide model development without pooling the underlying data?*

BioCoLoop extends collaboration from parameter fitting to model development by sharing both model updates and laboratory-local development evidence. Figure 1 places this framework alongside centralized self-improving agents and conventional collaborative training. An inner loop fits and evaluates candidates across laboratories. An outer loop uses the resulting evidence to revise proposals. Collaborative participation supplies training data and evaluation panels; development-history

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

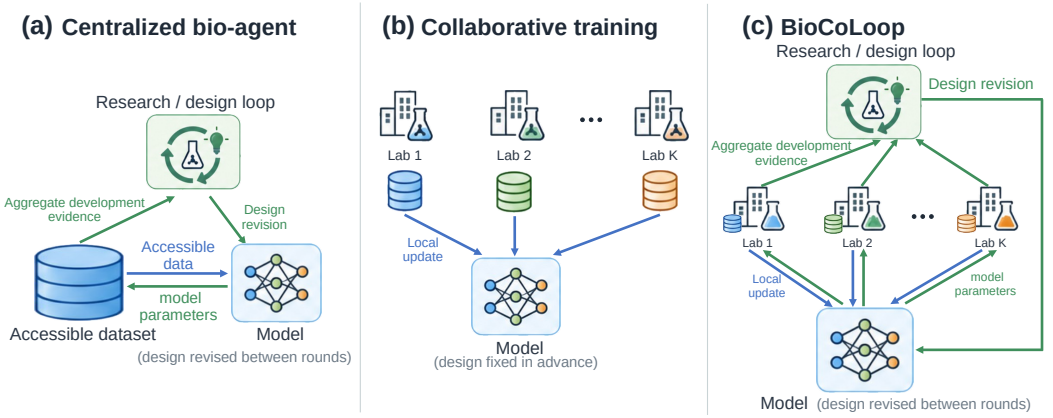

Figure 1: Research settings for biological AI. A centralized bio-agent revises a model using centrally accessible data. Collaborative training allows K laboratories to optimize the same model design and training hyperparameters by sharing model updates rather than raw data. BioCoLoop additionally uses results from local evaluation to guide subsequent changes to the model design and training hyperparameters.

feedback uses their measured outcomes to guide search. We also evaluate a separate policy that uses early development trajectories to allocate further training.

We evaluate BioCoLoop on drug–target interaction prediction, observed-response proteomic efficacy prediction and cell perturbation identification, with task-adapted AI-Scientist-v2 and AI-Researcher baselines and Qwen2.5-7B-Instruct and GPT-5.6 Luna as proposal generators. BioCoLoop improves most primary endpoints. Controlled studies examine laboratory participation, source compatibility, proposal-history feedback and training allocation. Our contributions are:

- We introduce a collaborative research framework that uses laboratory-local development evidence, alongside model updates, to guide iterative model development without centralizing raw data.
- We develop an inner loop for collaborative training and evaluation and an outer loop for evidence-guided proposal revision, with a common execution interface and persistent development history. A separate allocation policy uses early trajectories to direct training.
- Across drug–target interaction prediction, observed-response proteomic efficacy prediction and cell perturbation identification, we report gains on most primary endpoints, earlier discovery of useful configurations in selected searches, and improved training allocation at matched budgets.

2. Related work

Collaborative learning for biological data. Parameter aggregation enables institutions to train a shared model while retaining their measurements locally (McMahan et al., 2017; Sheller et al., 2020). In single-cell analysis, scFed addresses cell-type classification and FedscGen addresses distributed batch correction (Wang et al., 2024; Bakhtiari et al., 2025). Tabula and its Chiron implementation support collaborative single-cell foundation-model training through an agent-enabled platform (Ding et al., 2026; Mechtel et al., 2026). Automated design can interact with local training: FedEx tunes hyperparameters while training shared weights (Khodak et al., 2021), and Helmsman synthesizes collaborative-learning systems through planning, coding and simulation-based refinement (Li et al., 2025).

Self-improving agents in biological research. Self-improving agents turn experiment outcomes into subsequent computational proposals. AIDE searches candidate programs through drafting, debugging and improvement (Jiang et al., 2025b), while the AI Scientist and AI-Scientist-v2 connect ideas to executed experiments, replanning, tree search and writing (Lu et al., 2024; Yamada et al.,

2025). DrugEvolve combines research, implementation, evaluation and algorithm memory for drug-development tasks (Zhou et al., 2026; GAIR-NLP DrugEvolve contributors, 2026). BioCoLoop studies how multiple data-owning laboratories can supply development feedback through a shared research interface. TAPB, a ProteinTalks-derived efficacy head and scDEBART-based cell representations provide its task-specific prediction components (Lin et al., 2025; Sun et al., 2026; Sung & Kim, 2026). The proposal schema, fitting service and selection mechanism are common across tasks. Strong simple references on some perturbation benchmarks motivate evaluating the research procedure with fixed task predictors (Ahlmann-Eltze et al., 2025). Configuration search also connects to random search (Bergstra & Bengio, 2012), Bayesian optimization (Snoek et al., 2012) and language-model proposal distributions (Yang et al., 2024). We use feedback-free direct search to isolate the effect of development history under a shared proposal and fitting interface.

BioCoLoop connects collaborative fitting to a persistent development history. A shared coordinator executes candidate configurations across laboratories, aggregates their development measurements and returns structured evidence to a proposal model whose weights remain fixed. This interface supports history-guided proposal revision (Section 3.3) and, in a separate study, trajectory-guided training allocation (Section 3.4).

3. Method: collaborative biological model improvement

3.1 Problem setting and notation

Consider K laboratories working on the same biological prediction task. Laboratory i holds training data D_i^{train}, with $n_i = |D_i^{\text{train}}|$. Evaluation uses laboratory-local development panels $\{D_i^{\text{dev}}\}_{i\in\mathcal{P}}$, where $\mathcal{P}$ indexes the panels specified by the evaluation protocol. A task specification $\mathcal{T}$ defines the prediction inputs, outputs, training loss and evaluation metric. We write the task predictor as $f_{a,w}$, where $a \in \mathcal{A}$ specifies its design and w its trainable parameters. Given an initial design, a design space $\mathcal{A}$ and a research budget, BioCoLoop seeks a design and fitted predictor that perform well on the designated evaluation population. The three task families instantiate $f_{a,w}$ as task-specific predictors within this shared research procedure.

The *registered executable design space* $\mathcal{A}$ contains complete candidate configurations with four editable choices: learning rate, weight decay, coordinator momentum and an optional residual prediction module. Each `design_id` selects one configuration. The architecture choice enables or disables a zero-initialized, width-32 residual module; the task adapter supplies the backbone and prediction head.

A *shared coordinator* is the trusted execution service used by all participating laboratories. It resolves each design identifier, initializes and broadcasts the predictor, coordinates local fitting, aggregates model states and bounded development measurements, and applies the checkpoint and design-selection rules. Laboratories receive the same candidate state and schedule while retaining raw training and development measurements locally. The coordinator supplies the aggregate evidence to research model S (Figure 2), a language model whose weights θ remain fixed throughout a run. We use this fitting interface to evaluate history-guided proposal revision (Section 3.3) and trajectory-guided training allocation (Section 3.4) separately.

3.2 Collaborative fitting and shared evaluation

For each candidate a, the coordinator resets the predictor to its prescribed initialization w_0 and distributes it to all training laboratories. Let LocalEpoch_a denote one full local AdamW epoch with the candidate's learning rate and weight decay, batch size 64 and gradient-norm clipping at one. The optimizer state resets at each round. With sample weights $\pi_i = n_i / \sum_{j=1}^{K} n_j$, coordinator momentum coefficient β_a and initial velocity $v_0 = 0$, round r updates the trainable parameters by

$$u_{i,r} = \text{LocalEpoch}_a(w_{r-1}; D_i^{\text{train}}), \quad \bar{w}_r = \sum_{i=1}^{K} \pi_i u_{i,r}, \qquad (1)$$
$$v_r = \beta_a v_{r-1} + \bar{w}_r - w_{r-1}, \qquad w_r = w_{r-1} + v_r.$$

The coordinator carries momentum across aggregation rounds through v_r. With $\beta_a = 0$, aggregation reduces to sample-weighted parameter averaging (McMahan et al., 2017). In each round, the coor-

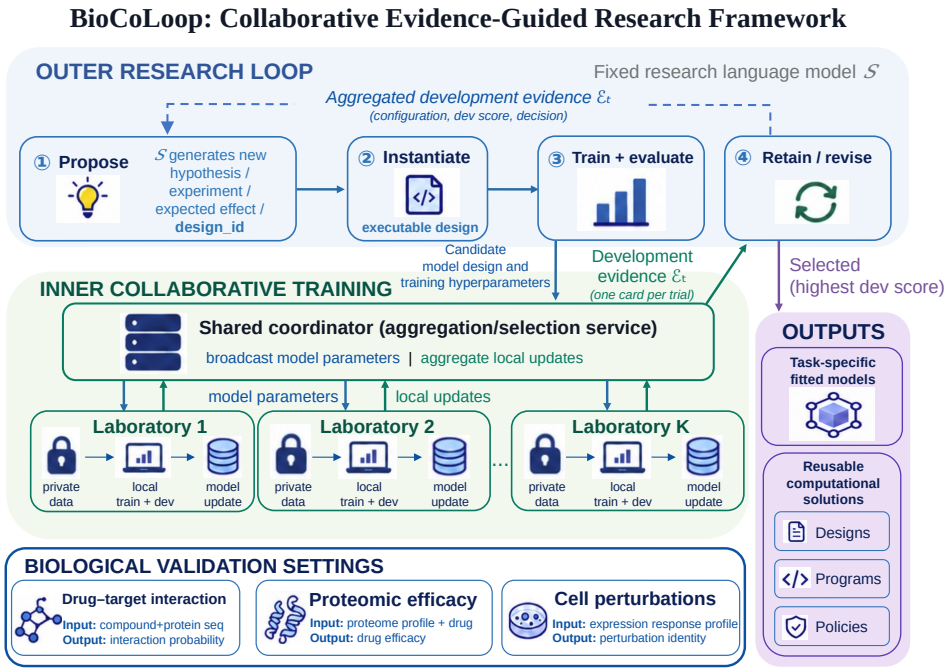

Figure 2: Data flow in BioCoLoop and the symbols used throughout. A candidate is an executable model configuration: a model design together with its training hyperparameters. The outer research loop is driven by the research model S with fixed weights θ, which proposes one candidate and receives the accumulated development evidence $\mathcal{E}_t$ after each evaluation. Every trial appends one development-evidence card to the persistent set $\mathcal{E}_t = \{e_1, \ldots, e_t\}$. A card records the executed configuration, the hypothesis, the best dev score and loss of the candidate and the acceptance decision. The shared coordinator of Section 3.1 is the logical service that resolves the candidate, distributes model parameters to K laboratories, fits them locally, aggregates the returned updates and summarizes each card. Section 3.2 defines the card and the dev score. The next proposal is conditioned on $\mathcal{E}_t$ through Equation 3. Task adapters implement drug–target interaction prediction, observed-response proteomic efficacy prediction and cell perturbation identification.

dinator sends the current trainable state and fixed candidate settings to every laboratory. Laboratory i returns its locally fitted state $u_{i,r}$ and training-example count n_i. Frozen feature extractors remain unchanged. Task adapters supply the biological inputs, trainable prediction components and local loss; raw examples, expression profiles, labels and per-example identifiers remain local.

Every five rounds and at the final round, laboratories evaluate the aggregated predictor f_{a,w_r} on the development panels indexed by $\mathcal{P}$. Each returns a scalar primary score $s_i(a, r)$, development loss $\ell_i^{\text{dev}}(a, r)$ and the bounded trajectory diagnostics required by the protocol. The main comparison uses participating laboratories' panels; fixed-pool repartitioning retains all ten original panels; source-transfer studies use the original target panels. The coordinator computes

$$\bar{s}(a, r) = \frac{1}{|\mathcal{P}|} \sum_{i \in \mathcal{P}} s_i(a, r), \qquad \bar{\ell}(a, r) = \frac{1}{|\mathcal{P}|} \sum_{i \in \mathcal{P}} \ell_i^{\text{dev}}(a, r). \tag{2}$$

The selected round $\widehat{r}_a$ maximizes $\bar{s}(a, r)$, with lower $\bar{\ell}(a, r)$ breaking ties and the earlier round retained on a complete tie. Its parameters are $\widehat{w}_a = w_{\widehat{r}_a}$. The pair $q(a) = (\bar{s}(a, \widehat{r}_a), -\bar{\ell}(a, \widehat{r}_a))$ ranks the design. We call $\bar{s}(a, \widehat{r}_a)$ the *dev score*, labelled "highest dev score" in Figure 2. Training weights examples, while selection weights the designated development panels equally. The evaluation schedule assigns each candidate 100 training rounds in the main proposal-search experiments. Each trial produces a *development-evidence card*. After t trials, the ordered set $\mathcal{E}_t = \{e_1, \ldots, e_t\}$ forms the accumulated *development evidence*. For $1 \le j \le t$, card e_j records the executed configuration a_j, hypothesis h_j, best development primary score and loss (s_j, ℓ_j), selected round $\widehat{r}_j$ and acceptance decision dec_j. Endpoint diagnostics Δ_j contain the first and last development-evaluation

rounds, their sample-weighted training losses, equal-laboratory development scores and losses, and the changes between those measurements, plus the optional worst-laboratory score and evaluating-laboratory count.

For auxiliary-source studies with different prediction targets, shared encoders follow the same update rule, while each source-specific head is aggregated only over laboratories using that prediction target. Checkpoint selection uses the original target's development panels. Each source retains its own prediction objective while contributing updates to the shared representation.

Training participation and development evaluation have explicit, separate roles in the interface. Training-example counts determine the contribution of each local update; the designated panels determine the population used to select checkpoints and designs. Recording the executed configuration alongside its trajectory and acceptance decision makes each trial available as a traceable research observation. The same card format carries evidence across task adapters even when their biological inputs and prediction losses differ.

3.3 Proposal revision from development evidence

The first proposal is generated once without development feedback and shared by direct and loop search. History feedback begins with the second proposal. For subsequent research steps t, let a_t be the retained design, $\mathcal{E}_t$ the development evidence accumulated so far and $\mathcal{U}_t$ the untried configurations. The research model S, with weights θ held fixed, generates a proposal p_t and the coordinator maps its `design_id`, denoted $\mathrm{id}(p_t)$, to the candidate:

$$p_t \sim P_\theta(\cdot \mid \mathcal{T}, a_t, \mathcal{E}_t, \mathcal{U}_t), \qquad \widetilde{a}_t = \mathrm{Compile}(\mathrm{id}(p_t)). \tag{3}$$

The proposal also contains a `hypothesis`, an `experiment` and an `expected_effect`, recording the motivation, intended comparison and anticipated outcome. The identifier determines execution; measured development performance determines selection.

After training $\widetilde{a}_t$, the coordinator appends the card e_t of Section 3.2 to the development evidence. Comparing q lexicographically by higher score and then lower loss gives

$$a_{t+1} = \begin{cases} \widetilde{a}_t, & q(\widetilde{a}_t) >_{\text{lex}} q(a_t), \\ a_t, & \text{otherwise}, \end{cases} \qquad \mathcal{E}_{t+1} = \mathcal{E}_t \parallel [e_t], \tag{4}$$

where $\parallel$ appends a card to the evidence set. Unsuccessful trials remain in the history. Invalid proposals and numerically failed fits consume a proposal slot and leave the incumbent unchanged. Updated context carries these outcomes into subsequent proposals. After the proposal budget is exhausted, the retained design and its selected checkpoint are evaluated on the held-out task.

A Norman trajectory illustrates the evidence returned to the proposer. With ten laboratories and seed 42, the first proposal, D06, is accepted at a tied development Top-1 of 40.00 because its loss decreases from 158.98 to 158.94. The second proposal, D08, retains the residual module and lowers the learning rate from 10^{-3} to 10^{-4}. Its development Top-1 rises from 10.20 at round 5 to 51.90 at round 100, and it is accepted. The next proposal, D07, reaches 45.00 and is rejected. The accumulated history records both the slower-improving accepted candidate and the unsuccessful alternative, with the exact configurations and checkpoint decisions retained in Appendix A.2.

3.4 Evidence-guided training allocation

We use trajectory-guided allocation in a controlled study (Section 4.4) to test whether early evaluation evidence improves the use of a fixed training budget. The screening–promotion policy retains both high-scoring and still-improving candidates. It screens ten predetermined designs for 20 rounds, evaluating every five rounds through Equation 2. Within each of three learning-rate groups, it promotes the best-scoring design and, among the remaining designs, the one with the largest score improvement from round 5 to round 20. Development loss breaks score ties. The six promoted designs restart from their initial parameters for 100-round training, and their best development checkpoints determine the final predictor.

The two promotion criteria reserve training capacity for candidates at different stages of convergence. Current score favors a strong screening checkpoint, while improvement over the shared screening

interval retains a candidate whose score is still rising. Grouping by learning rate preserves representation from each group in the full-training stage. All promotion decisions use development measurements.

The control trains all ten designs for 80 rounds. Both schedules use 800 aggregation rounds and 160 aggregate development evaluations. This study evaluates allocation over a predetermined candidate set independently of proposal generation. The main comparison and proposal-history experiments use 100 training rounds per candidate. Section 4.4 reports the allocation results; Appendix A provides the complete proposal schema, algorithm and allocation rules.

4. Experiments

4.1 Tasks, datasets and evaluation

We evaluate BioCoLoop on three biological prediction problems with task-specific base models and a common procedure for collaborative training, evaluation and proposal generation. The main setting partitions each dataset into ten disjoint groups. Complementary studies use distinct biological datasets or experimental scenarios as laboratories to evaluate source transfer.

Drug–target interactions. Given a compound and a protein sequence, the model predicts whether they interact. BindingDB provides 50,149 training pairs and 5,604 development pairs, with canonical compounds defining laboratory assignments. We use TAPB as the base model, with frozen ESM2 protein features and trainable drug, fusion and classification modules (Lin et al., 2025). The primary metric is AUROC on 5,505 random test pairs. Additional test sets containing 3,791 unseen-drug pairs and 2,507 unseen-protein pairs measure generalization to unseen entities. AP, MCC, accuracy and cross-entropy are reported as secondary metrics.

Observed-response proteomics. Given a drug representation and its measured 6h and 24h proteomic responses, the model predicts binary efficacy. A ProteinTalks PTPC subset provides 386 training, 105 development and 148 test conditions (Sun et al., 2026). Compound-group assignment gives each laboratory 49-50 training and development conditions. The base model is a ProteinTalks-derived efficacy model with protein and drug branches, trained from random initialization. Average precision (AP) is the primary metric. AUROC, accuracy and cross-entropy are secondary metrics. Unlike de novo response prediction, this task conditions on observed post-treatment proteomic measurements.

Cell perturbation identification. Given an expression response and five candidate interventions, the model identifies the intervention whose predicted response has the highest cosine similarity to the observation. We use VCC single-gene perturbations (Arc Institute, 2025), Norman double-gene perturbations (Norman et al., 2019) and a Tahoe subset of drug perturbations in A549 at 5 μM (Zhang et al., 2025). Intervention-disjoint training, development and test counts are 60/20/20, 70/20/27 and 180/60/67, respectively. The base model uses a frozen scDEBART backbone with a trainable response head (Sung & Kim, 2026). All methods use the same corrected attention masks and the same candidate sets. The primary metric is intervention-macro Top-1, which first averages accuracy within each intervention and then across interventions. Random selection has an expected Top-1 of 20%. Micro Top-1, MRR, Top-3 and cross-entropy are reported as secondary metrics. Appendix A provides the response construction, chemical conditioning, training losses, and secondary evaluation details.

4.2 Compared methods and shared protocol

We compare BioCoLoop with three types of baselines. The *fixed model* fits the task-specific base model once with a predefined configuration. *Direct optimization* uses BioCoLoop's proposal generator and registered executable design space, generating proposals without previous development outcomes. AI-Scientist-v2 (Yamada et al., 2025) and AI-Researcher (Tang et al., 2025) are adapted to the same biological tasks and fitting interface.

BioCoLoop combines collaborative fitting and evaluation with development-history feedback for subsequent proposals. We repeat the comparison using Qwen2.5-7B-Instruct and GPT-5.6 Luna as proposal generators (OpenAI, 2026). All methods share the task-specific base model, registered

Table 1: Main comparison by self-improving model (mean ± sample SD, seeds 42–44; scores ×100). Public self-improving agents use laboratory 0 and BioCoLoop ten under a shared design library, fits and held-out scorer. Bold marks the best complete value per endpoint and the lowest mean rank; underline marks the runner-up; † partial and – unscored cells are not imputed. Mean rank ranks the methods of a block from 1 (best) to 5 on every endpoint they completed and averages those ranks, so lower is better and the ↑/↓ offset is the gap to that block's fixed reference. The three cell endpoints are five-option identification tasks with a 20% chance level, so their headroom above chance is small. ⋆ marks a cell whose controller was admitted under an extended native-context budget; the cell it does not score is GPT-5.6 Luna AI-Researcher DTI, whose native trajectory did not finish inside that budget (Appendix C).

Model / method	DTI	Proteomics	Cell perturbation			Average
	BindingDB AUROC ↑	PTPC AP ↑	VCC Top-1 ↑	Norman Top-1 ↑	Tahoe Top-1 ↑	Mean rank ↓
Qwen2.5-7B-Instruct						
Fixed model	82.83 ± 1.22	25.05 ± 4.56	29.94 ± 0.00	**25.56** ± 0.00	19.40 ± 0.00	2.40
Direct	89.89 ± 1.02	20.13 ± 0.07	21.09 ± 4.38	14.15 ± 0.00	15.42 ± 6.21	3.40 ↓1.0
AI-Scientist-v2	89.68 ± 0.80	20.18 ± 0.43	18.61 ± 2.78	14.12 ± 0.04	15.42 ± 6.21	3.80 ↓1.4
AI-Researcher	86.09 ± 0.88	20.12 ± 0.05	19.70 ± 2.87†	21.73 ± 6.63	17.41 ± 6.21	3.75 ↓1.4
BioCoLoop	**93.76** ± 1.11	**36.99** ± 0.22	**31.54** ± 0.08	22.25 ± 0.45	**22.39** ± 2.59	**1.20** ↑1.2
GPT-5.6 Luna (low reasoning)						
Fixed model	82.83 ± 1.22	25.05 ± 4.56	29.94 ± 0.00	**25.56** ± 0.00	19.40 ± 0.00	2.20
Direct	89.89 ± 1.02	20.17 ± 0.42	18.61 ± 2.78	21.75 ± 6.59	15.92 ± 6.03	3.20 ↓1.0
AI-Scientist-v2	89.59 ± 0.95	23.37 ± 5.89	18.61 ± 2.78	14.12 ± 0.04	11.94 ± 2.11†	3.25 ↓1.0
AI-Researcher	–	20.29 ± 0.27⋆	18.61 ± 2.78⋆	14.12 ± 0.04⋆	15.42 ± 6.21⋆	3.75 ↓1.5
BioCoLoop	**93.67** ± 0.98	**37.11** ± 0.01	**31.50** ± 0.00	22.69 ± 0.73	**22.39** ± 2.59	**1.20** ↑1.0

executable design space and held-out evaluation. Table 1 specifies laboratory access. Appendix A (Table A.1) includes a same-access comparison and varies laboratory participation under a fixed research loop. Appendices A and C provide the implementation and training details.

The main comparison uses twelve configurations and at most six proposals beyond the initial design. Each valid candidate starts from the prescribed task initialization and receives 100 training rounds. Development score and loss determine the retained design and checkpoint before held-out scoring. We report means and sample standard deviations over seeds 42–44 on fixed data partitions. These summaries describe training and search variability; paired cluster-bootstrap intervals in the Statistical Supplement quantify held-out sample uncertainty. The allocation, source-transfer and extended-search studies use their separately specified budgets and seeds.

4.3 Main results

Table 1 compares BioCoLoop with the fixed, direct, AI-Scientist-v2 and AI-Researcher baselines using Qwen2.5-7B-Instruct and GPT-5.6 Luna as proposal generators. BioCoLoop improves most primary endpoints in both blocks, with the clearest gains on drug–target interaction prediction and observed-response proteomics. Cell perturbation results vary more across methods, and the fixed model retains the highest Norman Top-1. Appendix A reports the same-access control; Section 4.6 examines participation and partitioning effects.

With Qwen, BioCoLoop reaches 93.76 AUROC on BindingDB and 36.99 AP on proteomics; with Luna, it reaches 93.67 and 37.11, respectively. The corresponding fixed-model scores are 82.83 and 25.05. BioCoLoop also attains the highest complete VCC and Tahoe scores in both blocks, at 31.54/31.50 and 22.39 macro Top-1. On Norman, the fixed model scores 25.56, compared with 22.25 for Qwen BioCoLoop and 22.69 for Luna BioCoLoop.

4.4 Evidence-guided training allocation

Evidence-guided allocation improves held-out performance at a matched budget (Table 2). Uniform allocation trains all ten designs for 80 rounds. The evidence-guided policy screens them for 20 rounds and restarts six selected designs for 100-round training. Each schedule uses 800 aggregation rounds and 160 development evaluations.

Across the twelve task–seed comparisons, allocation yields four wins, eight ties and no losses. Norman Top-1 increases from 11.48% to 22.32%, with positive gains in all three seeds. A selected low-learning-rate model continues to improve beyond the uniform schedule's stopping point. Tahoe

Table 2: Evidence-guided training allocation at a matched budget. Held-out scores use the same ten designs, ten laboratories, 800 full-client aggregation rounds and 160 aggregate development evaluations per method. Uniform allocation gives every design 80 rounds; the evidence-guided scheduler screens all designs for 20 rounds and restarts six promoted designs for 100-round training. Gain is evidence-guided minus uniform, in percentage points; W/T/L counts seed-level wins, ties and losses (total 4/8/0). Norman 95% CI: [4.54, 17.73]; Tahoe: [−1.00, 7.46]. The separate DTI development replay is in Appendix A.

Benchmark	Uniform	Evidence-guided	Gain	W/T/L
Proteomics: efficacy	**37.47**	**37.47**	+0.00	0/3/0
Cell: VCC single-gene	**20.01**	**20.01**	+0.00	0/3/0
Cell: Norman double-gene	11.48	**22.32**	**+10.84**	3/0/0
Cell: Tahoe drug	20.90	**22.89**	**+1.99**	1/2/0

improves in one seed and ties in two; proteomics and VCC tie in all three seeds. The policy directs more training to candidates whose early trajectories show continued improvement.

Norman's mean gain is 10.84 percentage points, with a hierarchical paired-bootstrap 95% interval of [4.54, 17.73]. Screening retains candidates with high current scores and candidates with improving trajectories within each learning-rate group. Promoted candidates restart from initialization, and the reported budget includes both screening and full training.

4.5 Learning from different data sources

Source-transfer studies use laboratories from distinct biological datasets or experimental settings. We combine BindingDB with BioSNAP, Davis and Human; VCC with three external single-cell studies; and PTPC with Lin, Ruprecht and decryptE (Lin et al., 2026; Ruprecht et al., 2020; Eckert et al., 2025). Target development data select each model, and all configurations use the same target test set.

Transfer gains depend on the source. Adding all three proteomic studies improves target AP, and Davis and Human improve DTI AUROC. With laboratory and training-condition counts matched, external single-cell studies yield lower target Top-1 than additional data from VCC. Appendix B reports the source-specific effects and matched controls.

In the one-study-per-laboratory proteomic protocol, adding Lin, Ruprecht and decryptE raises target AP from 34.41 to 37.53 and AUROC from 72.33 to 74.39. Shared encoders receive updates from all participating sources, while each source-specific head retains its own prediction objective. The original target development panels select checkpoints throughout this comparison. The matched cell study uses four laboratories and 60 training interventions in each arm: the same-source VCC split reaches 30.70% Top-1, compared with 28.82% for VCC plus external studies. Both arms share the same target anchor; measured gene panels and underlying cell counts remain source-specific.

The separate proteomic protocol that retains ten target training partitions also improves AP when all three sources are added, from 34.98 to 35.68. Its target-only model retains the higher AUROC, 74.75 compared with 74.28. Each protocol uses its own target-only reference with the same adapter and initialization, preserving the interpretation of source effects under that training partition.

4.6 Effect of the number of laboratories

We vary $K \in \{1, 2, 5, 10\}$ by either adding laboratories and training data or repartitioning a fixed total data pool. Adding data through participation can improve performance, while fixed-pool repartitioning has task-dependent effects (Figure 3). Proteomics performs best at an intermediate K, and Norman degrades as the fixed pool is split across more laboratories. Performance depends on both data access and the partition used for local training.

Both settings use the default design and a fixed training schedule. Increasing participation adds the corresponding local development panels, while fixed-pool repartitioning preserves equal-weight evaluation over all ten original panels. For the cell tasks, local training groups also define the contrastive-negative pool, so partitioning affects the learning objective's available negatives as well as the sequence of updates. Appendix B.1 reports the two settings separately with their corresponding evaluation panels.

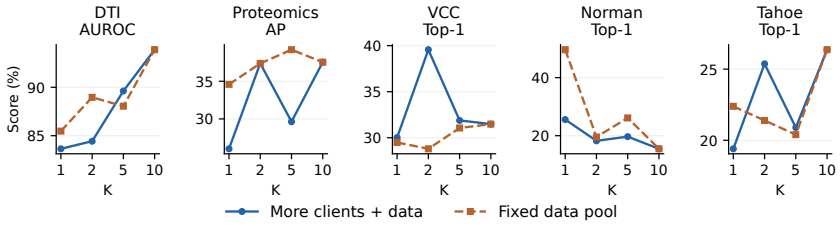

Figure 3: Effect of the number of laboratories. Solid lines add both laboratories and training data. Dashed lines divide the same training data across different numbers of laboratories. Full results are reported in Appendix B.

For DTI, AUROC increases from 83.65 to 93.88 as participation grows from one to ten laboratories, and from 85.47 to 93.88 in the fixed-pool study. With the proteomic data pool fixed, AP peaks at 39.19 with five laboratories and is 37.54 with ten. These curves retain the same default configuration at every laboratory count, providing a direct view of the training procedure's response to partitioning.

4.7 Development history and proposal trajectories

We compare direct optimization and history-guided search under the same laboratory access, registered executable design space, first proposal, training procedure and selection rule. History-guided search additionally receives the accumulated development evidence when generating subsequent proposals.

Development evidence changes search trajectories and can identify useful configurations earlier. In the extended Norman search, Luna reaches the common development target after four proposals with history, compared with 22 without it. Reducing the learning rate produces an early gain, and removing the residual head reaches the target. Both modes eventually attain the same held-out Top-1. Figures 4 and 5 report the complete prefix curves for the six-slot and extended protocols (Appendix B).

Search efficiency is measured by the first proposal attaining a common development target, defined as the lower terminal primary score within each direct/loop pair. Both trajectories can therefore reach that target. The common target is computed retrospectively from completed searches; candidate fitting, proposal budgets and development selection follow the registered protocol throughout execution. The curves retain every prespecified prefix, and loss-based tie-breaks can retain a new configuration after the primary score has plateaued. These diagnostics distinguish when a useful configuration is first reached from the final predictor selected at the full budget.

Final held-out effects are limited and endpoint-specific. Among the ten six-slot pairs, Norman/Luna improves by 0.81 macro Top-1 points (22.30 to 23.11), DTI/Luna decreases by 0.33 AUROC points and eight pairs tie. The extended 36-design protocol yields seven ties and one decrease. Appendix B.4 details the trajectories, including divergence between development selection and held-out performance.

5. Conclusion

BioCoLoop extends collaborative learning from parameter fitting to iterative model development. Laboratories contribute model updates for shared fitting and development evidence for research decisions through a common execution interface. Across the evaluated biological tasks, BioCoLoop improves most primary endpoints. Controlled studies show that development history changes proposal trajectories and can reach useful configurations earlier, while a separate matched-budget study demonstrates improved training allocation. Collaborative participation and evidence-guided decisions provide complementary components of a research loop in which laboratories jointly inform both how a model is fitted and what is tested next.

Reproducibility statement

Section 3 specifies the research and training interfaces; Section 4 defines task inputs, partitions, reference models and metrics. Appendix A gives the proposal schema, registered executable design space, optimization settings and matched-budget promotion rule. Appendices B and C document the sensitivity studies and public-agent adaptations. The Statistical Supplement reports complete per-seed contrasts and confidence intervals, including non-positive differences. An anonymized release accompanies the submission with the implementation, task adapters, configurations, proposal log and prediction-level evaluation records. These records retain code and data hashes, random seeds, development trajectories, selection decisions and saved predictions. The proposal log covers all 504 attempted slots (20×6 in the six-slot protocol and 16×24 in the extended protocol) and their configuration changes. Aggregate tables are generated from completed records, with independent score recomputation for the main comparison and follow-up evaluations. Qwen generation is seed-reproducible; Luna reproducibility and the public agents' interface constraints are specified in Appendices A and C.

Ethics statement

This computational study uses previously released biological resources under their source access conditions and collects no new measurements or participant data. Raw data remain local in the simulated laboratory protocol. Multi-institutional deployment would require a privacy and security policy for parameter updates and aggregate diagnostics. Predictions are evaluated for research use and have no biological or therapeutic validation.

AI use statement

The main comparison retains matched proposal records for local Qwen2.5-7B-Instruct and the service identified as `gpt-5.6-luna`. Luna is run at low reasoning effort through a text-only bridge with tools disabled. Its service does not expose a generation seed, so the seed labels identify the training/search repetitions rather than deterministic language-model sampling. AI assistants contributed to hypothesis refinement, experimental protocol and software development, data processing and checks, result analysis and interpretation, literature retrieval, figures, and manuscript drafting and revision. The research artifacts retain the generated proposals and their measured outcomes. The authors are responsible for verifying the final methods, results, citations and manuscript, including AI-assisted material.

References

Constantin Ahlmann-Eltze, Wolfgang Huber, and Simon Anders. Deep-learning-based gene perturbation effect prediction does not yet outperform simple linear baselines. *Nature Methods*, 22(8): 1657–1661, 2025. doi: 10.1038/s41592-025-02772-6. URL https://doi.org/10.1038/s41592-025-02772-6.

Arc Institute. Virtual cell challenge. Arc Virtual Cell Atlas dataset, 2025. URL https://github.com/ArcInstitute/arc-virtual-cell-atlas/tree/4f7bb170c61d2fafce70f67db6e52c7d9af1be4b/virtual-cell-challenge. 2025 data release; training file adata_Training.h5ad. Documentation accessed September 22, 2026.

Mohammad Bakhtiari, Stefan Bonn, Fabian Theis, Olga Zolotareva, and Jan Baumbach. FedscGen: privacy-preserving federated batch effect correction of single-cell RNA sequencing data. *Genome Biology*, 26:216, 2025. doi: 10.1186/s13059-025-03684-6. URL https://doi.org/10.1186/s13059-025-03684-6.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. In *Journal of Machine Learning Research*, volume 13, pp. 281–305, 2012. URL https://jmlr.org/papers/v13/bergstra12a.html.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scGPT: toward building a foundation model for single-cell multi-omics using generative AI. *Nature*

Methods, 21(8):1470–1480, 2024. doi: 10.1038/s41592-024-02201-0. URL https://doi.org/10.1038/s41592-024-02201-0.

Ittai Dayan, Holger R. Roth, Aoxiao Zhong, Ahmed Harouni, Amilcare Gentili, Anas Z. Abidin, Andrew Liu, Anthony Beardsworth Costa, et al. Federated learning for predicting clinical outcomes in patients with COVID-19. *Nature Medicine*, 27(10):1735–1743, 2021. doi: 10.1038/s41591-021-01506-3. URL https://doi.org/10.1038/s41591-021-01506-3.

Jiayuan Ding, Jianhui Lin, Ziyang Miao, Nils Mechtel, Shiyu Jiang, Yixin Wang, Zhaoyu Fang, Jorge D. Martin-Rufino, Chen Weng, Reuben Saunders, Weize Xu, Jonathan S. Weissman, Min Li, Jiliang Tang, Wei Ouyang, Yuancheng Ryan Lu, and Xiaojie Qiu. Predictive single cell foundation model for gene regulation and aging with privacy-preserving tabular learning, 2026. URL https://arxiv.org/abs/2607.19400v1. Version 1.

Stephan Eckert, Nicola Berner, Karl Kramer, Annika Schneider, Julian Müller, Severin Lechner, Sarah Brajkovic, Amirhossein Sakhteman, Christian Graetz, Jonas Fackler, Michael Dudek, Michael W. Pfaffl, Percy Knolle, Stephanie Wilhelm, and Bernhard Kuster. Decrypting the molecular basis of cellular drug phenotypes by dose-resolved expression proteomics. *Nature Biotechnology*, 43:406–415, 2025. doi: 10.1038/s41587-024-02218-y. URL https://doi.org/10.1038/s41587-024-02218-y. First published online 7 May 2024.

GAIR-NLP DrugEvolve contributors. DrugEvolve: Official source and applications, 2026. URL https://github.com/GAIR-NLP/DrugEvolve/tree/e945c7d0da28ddce73904419116a51e03921e6e6. Commit e945c7d0da28ddce73904419116a51e03921e6e6.

HKUDS. AI-Researcher: Official source, 2025. URL https://github.com/HKUDS/AI-Researcher/tree/f9a6f8480860c193afff600eeffe3defcee8a978. Commit f9a6f8480860c193afff600eeffe3defcee8a978.

Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. MLAgentBench: Evaluating language agents on machine learning experimentation, 2023. URL https://arxiv.org/abs/2310.03302v1. Version 1, 5 October 2023.

Longda Jiang, Carol Dalgarno, Efthymia Papalexi, Isabella Mascio, Hans-Hermann Wessels, Huiyoung Yun, Nika Iremadze, Gila Lithwick-Yanai, Doron Lipson, and Rahul Satija. Systematic reconstruction of molecular pathway signatures using scalable single-cell perturbation screens. *Nature Cell Biology*, 27:505–517, 2025a. doi: 10.1038/s41556-025-01622-z. URL https://doi.org/10.1038/s41556-025-01622-z.

Zhengyao Jiang, Dominik Schmidt, Dhruv Srikanth, Dixing Xu, Ian Kaplan, Deniss Jacenko, and Yuxiang Wu. AIDE: AI-driven exploration in the space of code, 2025b. URL https://arxiv.org/abs/2502.13138v1. Version 1, 18 February 2025.

Peter Kairouz, H. Brendan McMahan, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021. doi: 10.1561/2200000083. URL https://doi.org/10.1561/2200000083.

Georgios A. Kaissis, Marcus R. Makowski, Daniel Rückert, and Rickmer F. Braren. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*, 2(6):305–311, 2020. doi: 10.1038/s42256-020-0186-1. URL https://doi.org/10.1038/s42256-020-0186-1.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U. Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning, 2020. URL https://arxiv.org/abs/1910.06378v1. Version 1, 14 October 2019; published at ICML 2020.

Mikhail Khodak, Renbo Tu, Tian Li, Liam Li, Maria-Florina Balcan, Virginia Smith, and Ameet Talwalkar. Federated hyperparameter tuning: Challenges, baselines, and connections to weight-sharing. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://papers.neurips.cc/paper_files/paper/2021/hash/a0205b87490c847182672e8d371e9948-Abstract.html.

Haoyuan Li, Mathias Funk, and Aaqib Saeed. Helmsman: Autonomous synthesis of federated learning systems via collaborative LLM agents, 2025. URL https://arxiv.org/abs/2510.14512v2. Version 2, 19 December 2025.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks, 2020. URL https://arxiv.org/abs/1812.06127v1. Version 1, 14 December 2018; published at MLSys 2020.

Chuwei Lin, Catherine M. Sniezek, Christopher D. McGann, Rashmi Karki, Ross M. Giglio, Benjamin A. Garcia, José L. McFaline-Figueroa, and Devin K. Schweppe. Defining the heterogeneous molecular landscape of lung cancer cell responses to epigenetic inhibition. *Communications Biology*, 9:176, 2026. doi: 10.1038/s42003-025-09455-0. URL https://doi.org/10.1038/s42003-025-09455-0. Published 28 January 2026.

Gaoming Lin, Xin Zhang, Zhonghao Ren, Quan Zou, Prayag Tiwari, Changjun Zhou, and Yijie Ding. TAPB: an interventional debiasing framework for alleviating target prior bias in drug-target interaction prediction. *Nature Communications*, 16:10867, 2025. doi: 10.1038/s41467-025-66915-1. URL https://doi.org/10.1038/s41467-025-66915-1.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The AI scientist: Towards fully automated open-ended scientific discovery, 2024. URL https://arxiv.org/abs/2408.06292v1. Version 1, 12 August 2024.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative refinement with self-feedback, 2023. URL https://arxiv.org/abs/2303.17651v1. Version 1, 30 March 2023.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282, 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Nils Mechtel, Wei Ouyang, and Chiron Platform contributors. Chiron Platform: Federated single-cell training and agent interfaces, 2026. URL https://github.com/aristoteleo/chiron/tree/3241b35c8b516c126aa68da450708de3da717ee0. Commit 3241b35c8b516c126aa68da450708de3da717ee0.

Ajay Nadig, Joseph M. Replogle, Angela N. Pogson, Mukundh Murthy, Steven A. McCarroll, Jonathan S. Weissman, Elise B. Robinson, and Luke J. O'Connor. Transcriptome-wide analysis of differential expression in perturbation atlases. *Nature Genetics*, 57:1228–1237, 2025. doi: 10.1038/s41588-025-02169-3. URL https://doi.org/10.1038/s41588-025-02169-3.

Thomas M. Norman, Max A. Horlbeck, Joseph M. Replogle, Alex Y. Ge, Albert Xu, Marco Jost, Luke A. Gilbert, and Jonathan S. Weissman. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 365(6455):786–793, 2019. doi: 10.1126/science.aax4438. URL https://doi.org/10.1126/science.aax4438.

OpenAI. The GPT-5.6 model family: the `gpt-5.6-luna` service, 2026. Proprietary reasoning service accessed through the OpenAI Codex CLI at low reasoning effort with tools disabled; the service exposes no generation seed. Accessed September 2026.

Joseph M. Replogle, Reuben A. Saunders, Angela N. Pogson, Jeffrey A. Hussmann, Alexander Lenail, Alina Guna, Lauren Mascibroda, Eric J. Wagner, Karen Adelman, Gila Lithwick-Yanai, Nika Iremadze, Florian Oberstrass, Doron Lipson, Jessica L. Bonnar, Marco Jost, Thomas M. Norman, and Jonathan S. Weissman. Mapping information-rich genotype-phenotype landscapes with genome-scale Perturb-seq. *Cell*, 185(14):2559–2575.e28, 2022. doi: 10.1016/j.cell.2022.05.013. URL https://doi.org/10.1016/j.cell.2022.05.013.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletarì, Holger R. Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N. Galtier, Bennett A. Landman, Klaus Maier-Hein, Sébastien Ourselin, Micah Sheller, Ronald M. Summers, Andrew Trask, Daguang Xu, Maximilian Baust, and M. Jorge Cardoso. The future of digital health with federated learning. *npj Digital Medicine*, 3:119, 2020. doi: 10.1038/s41746-020-00323-1. URL https://doi.org/10.1038/s41746-020-00323-1.

Yusuf Roohani, Andrew Lee, Qian Huang, Jian Vora, Zachary Steinhart, Kexin Huang, Alexander Marson, Percy Liang, and Jure Leskovec. BioDiscoveryAgent: An AI agent for designing genetic perturbation experiments, 2024. URL https://arxiv.org/abs/2405.17631v1. Version 1, 27 May 2024.

Benjamin Ruprecht, Julie Di Bernardo, Zhao Wang, Xuan Mo, Oleg Ursu, Matthew Christopher, Rafael B. Fernandez, Li Zheng, Brian D. Dill, Huijun Wang, Yuting Xu, Andy Liaw, Jonathan D. Mortison, Nirodhini Siriwardana, Brian Andresen, Meir Glick, James R. Tata, Victoria Kutilek, Ivan Cornella-Taracido, and An Chi. A mass spectrometry-based proteome map of drug action in lung cancer cell lines. *Nature Chemical Biology*, 16:1111–1119, 2020. doi: 10.1038/s41589-020-0572-3. URL https://doi.org/10.1038/s41589-020-0572-3.

SakanaAI. The AI scientist-v2: Official source, 2025. URL https://github.com/SakanaAI/AI-Scientist-v2/tree/96bd51617cfdbb494a9fc283af00fe090edfae48. Commit 96bd51617cfdbb494a9fc283af00fe090edfae48.

Micah J. Sheller, Brandon Edwards, G. Anthony Reina, Jason Martin, Sarthak Pati, Aikaterini Kotrotsou, Mikhail Milchenko, Weilin Xu, Daniel Marcus, Rivka R. Colen, and Spyridon Bakas. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports*, 10:12598, 2020. doi: 10.1038/s41598-020-69250-1. URL https://doi.org/10.1038/s41598-020-69250-1.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning, 2023. URL https://arxiv.org/abs/2303.11366v1. Version 1, 20 March 2023.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, 2012. URL https://arxiv.org/abs/1206.2944.

Rui Sun, Liujia Qian, Yongge Li, Tong Liu, Honghan Cheng, Xuedong Zhang, Xueya Zhou, Yuecheng Zhan, Guangmei Zhang, Zhengchao Luo, Kunpeng Ma, Chunlong Wu, Dongchen Ji, Zhangzhi Xue, Hongxue Meng, Yuhang Xiang, Dingwei Lei, Qianhe Zhou, Wenbin Hu, Yuhan Deng, Lingling Tan, Qi Xiao, Zhiwei Liu, Lei Zeng, Liqin Qian, Xuan Zheng, Qiong Hu, Nianzi Luo, Weinan E, Peijie Zhou, Han Wen, Yi Zhu, and Tiannan Guo. An operational perturbation proteomics-based virtual cell model. *Nature*, 2026. doi: 10.1038/s41586-026-11001-9. URL https://doi.org/10.1038/s41586-026-11001-9. Published online 9 September 2026.

Jieun Sung and Wankyu Kim. scDEBART: Predicting in silico single-cell perturbation responses via large-scale differential expression learning. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=pJyidZg93y.

Jiabin Tang, Lianghao Xia, Zhonghang Li, and Chao Huang. AI-Researcher: Autonomous scientific innovation, 2025. URL https://arxiv.org/abs/2505.18705v1. Version 1, 24 May 2025.

Christina V. Theodoris, Ling Xiao, Anant Chopra, Mark D. Chaffin, Zeina R. Al Sayed, Matthew C. Hill, Helene Mantineo, Elizabeth M. Brydon, Zexian Zeng, X. Shirley Liu, and Patrick T. Ellinor. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023. doi: 10.1038/s41586-023-06139-9. URL https://doi.org/10.1038/s41586-023-06139-9.

Shuang Wang, Bochen Shen, Lanting Guo, Mengqi Shang, Jinze Liu, Qi Sun, and Bairong Shen. scFed: federated learning for cell type classification with scRNA-seq. *Briefings in Bioinformatics*, 25(1):bbad507, 2024. doi: 10.1093/bib/bbad507. URL https://doi.org/10.1093/bib/bbad507.

Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob Foerster, Jeff Clune, and David Ha. The AI scientist-v2: Workshop-level automated scientific discovery via agentic tree search, 2025. URL https://arxiv.org/abs/2504.08066v1. Version 1, 10 April 2025.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2309.03409.

Jesse Zhang, Airol A. Ubas, Richard de Borja, Valentine Svensson, Nicole Thomas, Neha Thakar, Aidan Winters, Umair Khan, Matthew G. Jones, John D. Thompson, Vuong Tran, Joseph Pangallo, Efthymia Papalexi, Ajay Sapre, Hoai Nguyen, Oliver Sanderson, Maria Nigos, Olivia Kaplan, Sarah Schroeder, Bryan Hariadi, Simone Marrujo, Crina Curca Alec Salvino, Guillermo Gallareta Olivares, Ryan Koehler, Gary Geiss, Alexander Rosenberg, Charles Roco, Daniele Merico, Nima Alidoust, Hani Goodarzi, and Johnny Yu. Tahoe-100M: A giga-scale single-cell perturbation atlas for context-dependent gene function and cellular modeling. *bioRxiv*, 2025. doi: 10.1101/2025.02.20.639398. URL https://www.biorxiv.org/content/10.1101/2025.02.20.639398v3. Version 3, May 10, 2025.

Zhimeng Zhou, Yang Nan, Minjie Mou, Yuntao Qian, Yixiu Liu, Zhengyu Zuo, Hao Yang, Weixian Xu, Bo Li, Wanghao Jiang, Yanlin Ren, Yang Liao, Yimeng Wang, Yinghong Li, Qingxia Yang, Zhiheng Xi, Tiantian Mi, Huaicheng Sun, Pengfei Liu, and Feng Zhu. An AI system for autonomous algorithm evolution in drug development. *bioRxiv*, 2026. doi: 10.64898/2026.08.16.745117. URL https://doi.org/10.64898/2026.08.16.745117. Version 1, posted 20 August 2026.

Appendix A. Algorithm and task protocols

A.1 Research algorithm and optimization

All three task families use the parameter update, development-selection rule and proposal interface of Section 3. A task adapter defines the input representation, prediction components, training loss and development scorer. The research process below starts from the task's reference design. The fitted reference also supplies its first evidence card.

	BioCoLoop research procedure
1	Fit the reference design from the task initialization and seed, then retain its best development checkpoint and initialize the research history.
2	Generate the shared first proposal without development feedback. From the second slot onward, supply the task, untried configurations, retained design and prior evidence cards.
3	Parse the proposal and map its `design_id` to an executable configuration. Allow at most two generation attempts in the slot.
4	Fit a valid candidate from the same initialization procedure and seed for the assigned number of aggregation rounds.
5	Select its best development checkpoint. Replace the incumbent only if its score/loss pair is better.
6	Append the executed configuration, hypothesis, diagnostics and decision to the history, including rejected or failed trials.
7	After the final slot, return the retained design, its selected checkpoint and the trial history, and score that checkpoint on the held-out task.

In the controlled direct/loop comparison, the first proposal is generated once without development feedback and shared. History enters loop search from the second proposal. Direct search subsequently receives the default configuration and untried identifiers without development outcomes; history-guided search receives the retained configuration and accumulated cards. Both use the same registered executable design space, proposal count and candidate-training schedule. Appendix B reports every prespecified proposal prefix. A configuration evaluated previously under the same task, data access and seed can reuse its completed fit while retaining its logical position and budget charge in each search.

Each aggregation round contains one local AdamW epoch per participating laboratory. The batch size is 64, gradient-norm clipping is one, and the local optimizer state resets each round. The learning rate and weight decay follow the selected design. The proximal penalty is zero throughout the reported design libraries. The coordinator weights local parameter states by their number of training examples. Coordinator momentum follows Equation 1 and applies only to trainable parameters. Floating-point buffers use sample-weighted averaging, while integer and Boolean buffers use an elementwise maximum. These buffers store non-parameter model state. Every candidate starts with the same task-specific public or random initialization procedure and study seed. The optional residual module has width 32 and a zero-initialized output, while the original prediction head remains trainable. The implementation calls each laboratory a *client* or *worker*, and its configuration field `server_momentum` stores the coordinator momentum coefficient. These names refer to the same laboratory and coordinator roles used in Section 3.

Full-training candidates receive 100 aggregation rounds, with development evaluation every five rounds. The selection key is mean primary score followed by negative mean loss over the development panels designated by the protocol (Section 3.2). In the main experiment these panels are the participating laboratories. A complete tie retains the earlier checkpoint or incumbent. Training-example counts refer to the adapter's training rows: for the cell tasks these are intervention-level training profiles, rather than the number of cells underlying a profile.

A.2 Registered executable design space, proposals and evidence

The prefix `D` stands for *design*; its two-digit suffix identifies a configuration. D00 is the starting configuration. The main study selects six further proposals from D01–D11 without repeating a design within a search. The table maps each identifier to its four-factor configuration, with a zero-initialized width-32 residual module when enabled. Appendix B specifies the extended design space and its identifiers.

Design	Learning rate	Weight decay	Momentum	Residual module
D00	0.001	0.001	0	no
D01	0.0001	0.001	0	no
D02	0.01	0.001	0	no
D03	0.001	0	0	no
D04	0.001	0.01	0	no
D05	0.001	0.001	0.9	no
D06	0.001	0.001	0	yes
D07	0.001	0.001	0.9	yes
D08	0.0001	0.001	0	yes
D09	0.0001	0.001	0.9	yes
D10	0.01	0.01	0	no
D11	0.001	0.01	0	yes

The main comparison has two proposal-model blocks. Qwen2.5-7B-Instruct uses bfloat16 inference, temperature 0.7, top-p 0.9 and at most 512 new tokens. For zero-indexed proposal slot t and generation attempt $b \in \{0, 1\}$, its sampling seed is $z + t + 10{,}000b$, where z is the study seed. GPT-5.6 Luna uses a text-only bridge at low reasoning effort with tools disabled and the same bounded proposal protocol. Its service exposes no generation seed; the seed labels identify training and search repetitions. An invalid identifier, repeated configuration or malformed response triggers the second attempt. A second failure consumes the slot without fitting a candidate. A numerically failed fit leaves the incumbent unchanged and remains in the evidence history.

A proposal has exactly four fields:

```
{
  "hypothesis": "why the proposed change may help",
  "experiment": "the intended controlled comparison",
  "expected_effect": "the anticipated metric or diagnostic change",
  "design_id": "D08"
}
```

The three descriptive fields must be nonempty strings and record the agent's rationale. The identifier must belong to the registered executable design space and determines the executed configuration.

An evidence card contains the executed configuration, its hypothesis, the best development score and loss, the selected round, and the acceptance decision. Its trajectory summary includes the first and last development-evaluation rounds, their sample-weighted training losses, equal-laboratory development scores and losses, and the last-minus-first changes in development score and loss. For a 100-round fit these endpoints are rounds 5 and 100. In the main comparison, the development summary also records the worst laboratory score and the number of evaluating laboratories. Training participation and the designated evaluation panels are specified separately for source-transfer and repartitioning studies. Ordered cards are supplied as context, and unsuccessful trials are not removed.

An observed proposal sequence. For Norman with ten laboratories and seed 42, D06 is accepted in the first slot: it ties D00's development Top-1 of 40.00 but reduces development loss from 158.98 to 158.94. In the second slot, D08 retains the residual module and changes the learning rate from 10^{-3} to 10^{-4}. Its development Top-1 rises from 10.20 at round 5 to 51.90 at round 100. Its development loss falls from 168.12 to 153.85. The candidate is accepted. The next proposed design, D07, reaches a best Top-1 of 45.00 and is rejected. Thus the history records both the slower-improving accepted candidate and the later unsuccessful alternative.

Three representative trajectories are summarized below. Scores and losses are multiplied by 100 and proposal slots are one-indexed.

Study	Selected design	Primary	Loss	First direct / loop slot
PTPC, seed 42	D04	66.58	52.12	5 / 5
VCC, seed 43	D00	24.04	160.78	initial baseline
Norman, seed 42	D08	51.90	153.85	3 / 2

PTPC selects stronger weight decay through the loss tie-break at equal AP, while VCC retains its initial design. Section 4.7 compares search trajectories under the shared design space, proposal budget and selection rule.

A.3 Independent training-allocation control

This study fixes the candidate set and changes only the allocation of training rounds. It uses seeds 53–55 on proteomics and the three cell settings. The ten configurations cover the following factor combinations. Every configuration uses the same ten laboratories.

Learning rate	Weight decay	Momentum	Residual module
0.001	0.001	0	no
0.0001	0.001	0	no
0.0001	0.001	0	yes
0.0001	0.01	0.9	no
0.0001	0.01	0.9	yes
0.001	0.001	0.9	yes
0.01	0.001	0	yes
0.01	0.001	0.9	yes
0.01	0.01	0.9	no
0.001	0.01	0	yes

Uniform allocation trains every design for 80 rounds. Trajectory-guided allocation first trains each for 20 rounds. Within each of the three learning-rate groups, it promotes the design with the highest primary score at its best screening checkpoint, using lower development loss to break ties. From the remaining designs in that group, it also promotes the largest development-score improvement from round 5 to round 20, using lower round-20 development loss to break ties. Complete ties follow the order of the configurations in the table. The six promoted designs restart from their initial parameters and train for 100 rounds. Final selection considers their full-training checkpoints, while the four unpromoted designs stop after screening.

Both schedules use 800 aggregation rounds: $10 \times 80 = 10 \times 20 + 6 \times 100$. With ten participating laboratories, this is 8,000 local epochs. Evaluation every five rounds gives 160 aggregate development evaluations, each formed from ten laboratory evaluations. These are matched training and evaluation counts. The scheduler itself generates no language-model proposals.

Across the 12 task–seed held-out comparisons, allocation gives four wins, eight ties and no losses. Norman improves by 10.84 Top-1 points on average, with seed-level effects of +10.00, +11.70 and +10.81 and a hierarchical paired-bootstrap 95% interval of [4.54, 17.73]. Tahoe gives one +5.97-point seed and two ties, and VCC and PTPC tie in all three seeds. A separate TAPB development replay uses the seven-design intersection of existing trajectories. Uniform allocation receives 525 candidate-rounds and trajectory-guided allocation receives 520, with a mean development AUROC difference of +0.39 points (two wins and one tie). It is not included in the twelve held-out comparisons.

A.4 Data construction and task adapters

Drug–target interaction. The BindingDB partition contains 50,149 fitting and 5,604 development pairs. Canonical compounds are assigned to ten laboratories, giving 4,932–5,058 training and 517–643 development pairs per laboratory. The fixed test sets contain 5,505 random, 3,791 unseen-drug and 2,507 unseen-protein pairs. TAPB uses frozen public protein features and trainable drug, fusion and classification components. An optional residual adapter precedes its native confounder-averaged prediction. Classification NLL and masked-language loss form its local training objective.

Observed-response proteomics. The PTPC source pool contains 333 compound groups and 491 conditions, partitioned into 386 fitting and 105 development conditions. A laboratory owns 35–42 fitting conditions. The test set contains 148 conditions. For each 6h and 24h profile, preprocessing applies log1p, within-profile min–max normalization over finite observations, zero filling and unit L2 normalization. The 2,048-bit Morgan vector is separately L2-normalized and scaled by 0.5. The two 5,171-dimensional proteomic blocks and drug vector form a 12,390-dimensional float32 input. The ProteinTalks-derived efficacy head has protein and drug convolutional branches followed by a

64–32–1 classifier, totaling 464,193 parameters under D00. It is trained from random initialization using binary cross-entropy. All comparison methods use these same inputs and architecture. AP is the selection metric.

Cell inputs and response units. VCC, Norman and Tahoe use whole-intervention fitting, development and test splits of 60/20/20, 70/20/27 and 180/60/67. Training uses one mean response profile per intervention. VCC development and test queries average disjoint groups of up to 64 cells, omitting a final group smaller than 16 cells. Norman queries are the released scVI-derived differential-expression profiles for each gene pair. Tahoe uses one aggregate response profile per compound at the selected cell context and dose. These are the units scored by the identification task.

The scDEBART backbone receives the shared untreated reference and candidate perturbation encoding. Its features are fixed while the response head is trained. The corrected feature extractor implements three attention masks as additive blocking masks. Its correction preserves the pretrained weights, gene panels, response targets, intervention partitions and candidate options. Genetic-task scoring excludes the union of directly perturbed genes. Tahoe concatenates a fixed 128-dimensional projection of a 2,048-bit Morgan fingerprint to the 768-dimensional backbone features, with the additional head-input weights initialized to zero. Its drug projection uses seed 1234. The training loss is response MSE plus 0.1 times a contrastive loss over interventions in the local training minibatch. The cosine-logit temperature is 0.1.

Five-option queries and scoring. For every development or test intervention, four distinct distractors are sampled uniformly without replacement from the *same* development or test intervention pool, excluding the true intervention. The true intervention is added and the five positions are shuffled. A single roster is fixed per intervention and reused for all of its response queries, all training seeds and all methods. Split construction and roster sampling use the preparation generator initialized with seed 20260918. Training does not use these five-option rosters, and its contrastive negatives come from the local training minibatch.

For each query, predicted candidate responses and the observed response are L2-normalized over the scoring genes. Candidates are ranked by their cosine similarity. Equal scores retain the fixed shuffled option order. Let G be the number of evaluated interventions, m_g the number of response queries for intervention g, and $r_{g,j}$ the rank of the true option for its jth query. The primary score is

$$\text{MacroTop1} = \frac{1}{G}\sum_{g=1}^{G}\frac{1}{m_g}\sum_{j=1}^{m_g}\mathbf{1}\{r_{g,j}=1\}.$$

Thus each intervention receives equal weight even when it contributes a different number of query profiles. Micro Top-1, mean reciprocal rank and Top-3 average over individual queries. The development loss is cross-entropy over the five cosine scores divided by temperature 0.1. Development selection first computes these scores within each laboratory and then averages laboratories equally.

Identification diagnostics. The reported frozen-scDEBART identification diagnostics span 7.5–31.6% macro Top-1 against a 20% chance level and 36.00–57.00 MRR against a 46.00 chance level. The untrained public head scores 7.5% on Norman. True-option positions are near-uniform across the fixed rosters, and a prototype oracle built from observed held-out responses retrieves the true option at 94.5% on VCC and 100% on Norman.

A.5 Complete comparisons and uncertainty

The following tables report the control comparison, secondary metrics and all three drug–target test sets. Bold marks the best displayed value within a task and metric, including ties. Higher is better except for binary cross-entropy and cell cross-entropy.

Table A.1: Control comparison for the task-reference models. The fixed and feedback-free direct references train or search on a single site, so they are reported on one laboratory; BioCoLoop is reported at one and ten laboratories so that the effect of laboratory participation can be read under the same research loop. Entries are held-out mean ± sample SD across completed training/search seeds, multiplied by 100. The data partitions remain fixed; this SD measures run-to-run variation, not dataset uncertainty. DTI uses the random held-out endpoint. Bold marks all displayed maxima.

		DTI	Proteomics	Cell perturbation		
Labs	Method	AUROC ↑	AP ↑	VCC Top-1 ↑	Norman Top-1 ↑	Tahoe Top-1 ↑
1	Fixed model	82.83±1.22	25.05±4.56	29.94±0.00	**25.56**±0.00	19.40±0.00
1	Direct	89.89±1.02	20.13±0.07	21.09±4.38	14.15±0.00	15.42±6.21
1	Our loop	89.89±1.02	20.13±0.07	20.22±2.22	14.15±0.00	15.42±6.21
10	Our loop	**93.76**±1.11	**36.99**±0.22	**31.54**±0.08	22.25±0.45	**22.39**±2.59
Completed seeds ($n/3$)		3/3	3/3	3/3	3/3	3/3

Table A.2: Held-out secondary metrics for the task-reference models, averaged over seeds 42–44 and multiplied by 100. Bold indicates all tied maxima, or minima for loss.

Method	Labs	Metric 1	Metric 2	Metric 3	Metric 4
DTI ($n = 3$)		AUROC	AP	MCC	BCE
Fixed model	1	82.83	81.57	49.95	52.37
Direct	1	89.89	87.77	65.71	61.02
Our loop	1	89.89	87.77	65.71	61.02
Our loop	10	**93.76**	**92.73**	**73.71**	**36.59**
Proteomics ($n = 3$)		AP	AUROC	BCE	Acc.
Fixed model	1	25.05	65.49	42.65	**85.14**
Direct	1	20.13	61.80	44.34	81.76
Our loop	1	20.13	61.80	44.34	81.76
Our loop	10	**36.99**	**73.87**	**39.83**	**85.14**
VCC ($n = 3$)		Micro Top-1	MRR	Top-3	CE
Fixed model	1	33.61	**56.53**	**73.89**	**154.51**
Direct	1	21.48	47.10	62.78	160.82
Our loop	1	20.74	46.94	64.26	160.86
Our loop	10	**33.98**	56.34	69.44	158.14
Norman ($n = 3$)		Micro Top-1	MRR	Top-3	CE
Fixed model	1	**25.56**	48.50	63.93	**158.74**
Direct	1	14.15	42.12	58.35	160.39
Our loop	1	14.15	42.12	58.35	160.39
Our loop	10	22.25	**50.07**	**70.96**	158.83
Tahoe ($n = 3$)		Micro Top-1	MRR	Top-3	CE
Fixed model	1	19.40	45.00	58.21	163.08
Direct	1	15.42	42.60	56.72	161.90
Our loop	1	15.42	42.60	56.72	161.90
Our loop	10	**22.39**	**47.95**	**61.19**	**160.75**

Table A.3: All three frozen TAPB held-out endpoints under the committed common protocol. Scores are completed-seed means multiplied by 100; N/A means no complete independently verified evaluation. The main DTI endpoint is random held-out.

Endpoint	Method	Labs	AUROC	AP
Random	Fixed model	1	82.83	81.57
Random	Direct	1	89.89	87.77
Random	Our loop	1	89.89	87.77
Random	Our loop	10	**93.76**	**92.73**
Unseen drug	Fixed model	1	81.96	85.78
Unseen drug	Direct	1	88.39	89.35
Unseen drug	Our loop	1	88.39	89.35
Unseen drug	Our loop	10	**91.65**	**92.58**
Unseen protein	Fixed model	1	76.40	78.75
Unseen protein	Direct	1	83.26	80.53
Unseen protein	Our loop	1	83.26	80.53
Unseen protein	Our loop	10	**87.63**	**85.07**

Three-seed means and sample standard deviations summarize training/search variability on fixed partitions. Sample-level intervals use paired cluster bootstrap with 1,000 fixed-seed resamples of

retained predictions. Drug identity defines clusters for the random and unseen-drug interaction sets, and protein sequence defines clusters for the unseen-protein set. Proteomic and cell intervals use compound and intervention clusters, respectively. Both methods in a contrast receive the same resampled clusters. The training-allocation study instead uses 5,000 hierarchical paired resamples: training seeds are drawn first, followed by compound or intervention groups within each sampled seed.

The accompanying *BioCoLoop Statistical Supplement*, Tables S.1–S.2, reports all per-seed differences and confidence intervals for the main contrasts and three drug–target test sets, including zero and negative differences. These model-conditional intervals quantify test-sample uncertainty; several cell comparisons include zero.

The accompanying research records contain the executable configurations, proposals, development histories and saved predictions needed to reconstruct these comparisons. The reproducibility statement describes their availability and verification.

A.6 Scope and limitations

The experiments simulate multiple laboratories on shared compute using public data. They evaluate the research protocol without measuring communication or straggler costs or providing formal privacy guarantees. Model development is confined to the finite registered executable design space, with four editable factors and one optional residual module. Qwen generation is seed-reproducible; the proprietary Luna service exposes no generation seed. Public self-improving agents use task-adapted interfaces with the component restrictions and incomplete executions documented in Appendix C. The study includes no centralized pooled-data reference using the same complete data pool.

Appendix B. Laboratory, source and proposer sensitivity

Sensitivity study design. We examine collaboration scale, source diversity, proposer choice and research budget using the same target scorers as the main study. Laboratory-count experiments cover all five endpoints at $K \in \{1, 2, 5, 10\}$ under both increasing participation and a fixed total data pool. DTI uses seed 61, while the lighter proteomics and cell experiments use seeds 61–63. Independent-source DTI and proteomics use seed 61, and external-source VCC uses seeds 61–63. The two-proposer study has two separate protocols: a 12-design, six-slot study on all five endpoints (DTI seed 42; other endpoints seed 61), and a 36-design, 24-slot study on the four lighter endpoints at seed 61. Each candidate receives 100 training rounds. The latter changes the design space as well as the search budget, so budget comparisons are made within a protocol. Main-table results retain their original seeds 42–44 and protocol.

B.1 Number of laboratories

The participation experiment uses a fixed nested set of the first K original laboratories, increasing both training and development access. The fixed-pool experiment merges the complete ten-laboratory pool into K training groups while retaining equal-weight evaluation over the original ten development panels. Thus its selection population stays fixed as training partitions change. The identical $K = 10$ fit is reused by both comparisons. Every unique fit uses the default design and 100 rounds. The cell objective forms contrastive negatives within each local training group, so repartitioning changes the negative pool as well as optimization steps. These comparisons characterize the implemented training pipeline at different collaboration scales. Zero between-seed SD can occur when a fixed pretrained head receives an identical single-batch update, which describes training variability on these fixed partitions.

Table B.1: Increasing participation. A nested set of the original ten client partitions participates, so increasing K also adds training data. The default design and 100-round trainer are fixed. DTI uses seed 61; the four light endpoints report mean ± sample SD over seeds 61–63. Scores are metric $\times 100$; bold denotes the best K within each row, including displayed ties.

Task	Metric	$K = 1$	$K = 2$	$K = 5$	$K = 10$
DTI	AUROC	83.65	84.43	89.62	**93.88**
PTPC	AP	26.03 ± 5.57	37.38 ± 0.60	29.63 ± 2.50	**37.54 ± 0.27**
VCC	Macro Top-1	30.02 ± 0.78	**39.56 ± 0.11**	31.88 ± 0.00	31.48 ± 0.16
Norman	Macro Top-1	**25.56 ± 0.00**	18.22 ± 0.00	19.68 ± 1.11	15.48 ± 0.00
Tahoe	Macro Top-1	19.40 ± 0.00	25.37 ± 0.00	20.90 ± 0.00	**26.37 ± 0.86**

Table B.2: Fixed data pool. All target data are retained and merged into K clients; only the partition count changes. The default design and 100-round trainer are fixed. DTI uses seed 61; the four light endpoints report mean ± sample SD over seeds 61–63. Scores are metric $\times 100$; bold denotes the best K within each row, including displayed ties.

Task	Metric	$K = 1$	$K = 2$	$K = 5$	$K = 10$
DTI	AUROC	85.47	88.96	88.05	**93.88**
PTPC	AP	34.59 ± 5.66	37.36 ± 1.89	**39.19 ± 0.43**	37.54 ± 0.27
VCC	Macro Top-1	29.49 ± 0.00	28.80 ± 0.20	31.06 ± 0.22	**31.48 ± 0.16**
Norman	Macro Top-1	**49.63 ± 4.20**	19.56 ± 0.20	26.12 ± 0.11	15.48 ± 0.00
Tahoe	Macro Top-1	22.39 ± 0.00	21.39 ± 0.86	20.40 ± 0.86	**26.37 ± 0.86**

B.2 Independent sources and target evaluation

For DTI and cell transfer, the entire target training pool forms one laboratory, and each auxiliary collection contributes a separate local update through sample-weighted aggregation. Table B.3 distinguishes these source-defined laboratories from the within-corpus partitions used in the laboratory-count study. Only the original target development panels select the checkpoint, and every configuration is evaluated on the same target test set. Target-only, individual-source and combined-source configurations start from the same task initialization.

DTI auxiliaries are BioSNAP, Davis and Human from the pinned TAPB release, with 19,092, 1,721 and 2,929 retained training pairs after drug–protein-pair de-duplication against target splits and pre-

Table B.3: Source and context mapping in the cross-source experiments. The DTI units are database collections, the cell units preserve their study and biological scenario, and each proteomic auxiliary study owns one client and its response head. The ten PTPC clients remain partitions of one target corpus. Biological contexts within a study are not counted as additional independent studies.

Task	Client unit	Sources and biological contexts
DTI	Four data collections	Target BindingDB; BioSNAP; Davis; Human. Each collection contributes one client after pair-level de-duplication.
Cell	Four study/scenario clients	Target VCC; Replogle K562; Nadig HepG2; Jiang IFNB/K562. Each auxiliary source contributes one client.
Proteomics	Ten target partitions plus three source clients	Clients 0–9: PTPC partitions; client 10: Lin (A549, H292, PC9, PSC1); client 11: Ruprecht (A549, Calu6, Calu1, 2030, 2122); client 12: decryptE (Jurkat, five doses).

ceding sources. All five DTI source configurations complete the same 100-round schedule. VCC auxiliary data comprise Replogle K562, Nadig HepG2 and Jiang IFNB/K562, with respectively 60, 60 and 58 retained training interventions (Replogle et al., 2022; Nadig et al., 2025; Jiang et al., 2025a). Source feature panels preserve measured genes, and target development/test intervention identities are excluded from source training. The source manifest records retained subsets, feature intersections and exclusions.

Table B.4: Independent-source transfer on the same target heldout set, including every prespecified source and their combination. DTI (seed 61) adds BioSNAP, Davis or Human; VCC (seeds 61–63, mean ± SD) adds Replogle, Nadig or Jiang. The PTPC auxiliary multi-task model (seed 61) adds Lin or Ruprecht and is compared only with its matched auxiliary target-only model. The two-source PTPC protocol shown here uses Lin and Ruprecht; the three-source extension is reported separately in Table B.7. Changes are paired against target-only training; bold marks the best score within a target/model block. Scores are multiplied by 100.

Target	Additional source	Metric	Score ↑	Δ (pp)
DTI	Target only	AUROC	85.47	+0.00
DTI	+ BioSNAP	AUROC	82.44	-3.03
DTI	+ Davis	AUROC	87.86	+2.39
DTI	+ Human	AUROC	**87.95**	+2.48
DTI	+ All three	AUROC	84.30	-1.18
VCC	Target only	Macro Top-1	29.49 ± 0.00	+0.00 ± 0.00
VCC	+ Replogle	Macro Top-1	27.12 ± 0.08	-2.37 ± 0.08
VCC	+ Nadig	Macro Top-1	26.94 ± 0.18	-2.55 ± 0.18
VCC	+ Jiang	Macro Top-1	**33.07 ± 0.00**	+3.58 ± 0.00
VCC	+ All three	Macro Top-1	27.33 ± 0.00	-2.15 ± 0.00
PTPC auxiliary	Target only	AP	34.98	+0.00
PTPC auxiliary	+ Lin	AP	35.01	+0.03
PTPC auxiliary	+ Ruprecht	AP	**35.50**	+0.52
PTPC auxiliary	+ Both	AP	35.31	+0.34

The cell study reports mean and SD across three seeds with a paired intervention-bootstrap analysis. Jiang improves target Top-1 by 3.58 points, with a 95% bootstrap interval of $[-2.11, 9.51]$. Source-specific gains and decreases are reported in Table B.4.

Matched source–scenario laboratories. Table B.5 compares four fixed-design settings: 15 VCC training interventions in one laboratory, all 60 VCC interventions in one laboratory, the same 60 partitioned into four laboratories, and four source-defined laboratories with 15 interventions each. The latter uses VCC, Replogle/K562, Nadig/HepG2 and Jiang/IFNB/K562. Both four-laboratory arms share the same 15-intervention VCC anchor. Hash-based sampling fixes the auxiliary identities before fitting, excluding target training, development and test intervention identities as well as duplicates across auxiliary laboratories. This experiment matches training intervention counts, not raw-cell counts: the same-source arm represents 77,771 cells, versus 33,469 in the cross-scenario arm. Each source retains its measured gene panel and corresponding loss mask.

All four settings use identical head initialization, the default design, 100 rounds and seeds 61–63. The original ten target-development panels select checkpoints, and the common target test contains 20

Table B.5: Fixed-design source–scenario ablation on VCC. K: training clients; N: training intervention conditions. The two $K = 4, N = 60$ arms share a 15-condition VCC anchor and differ in the sources of the other three clients. All arms use 100 rounds and the same target evaluation. Mean $\pm$ sample SD over three seeds; best means are bold. Cell counts and gene panels remain source-specific; protocol details are in Appendix B.

Training sources	K	N	Top-1 (%) ↑	MRR (%) ↑
VCC, limited data	1	15	29.36 ± 0.26	$\mathbf{56.97 \pm 0.19}$
VCC, all target data	1	60	29.43 ± 0.09	56.00 ± 0.09
VCC, same-source split	4	60	$\mathbf{30.70 \pm 0.42}$	56.24 ± 0.04
VCC + external studies	4	60	28.82 ± 0.00	56.30 ± 0.00

interventions and 360 five-option queries. No proposal search is performed. All twelve fits complete, and independent evaluation of saved predictions reproduces the metrics. Cross-scenario Top-1 is lower in each seed, with a mean difference of -1.88 points relative to the same-source four-laboratory setting; the MRR difference is near zero. The protocol and source identities remain fixed throughout the comparison.

Two-source proteomic transfer and proposal search. Two independent lung-cancer studies provide auxiliary measurements: Lin et al. (Lin et al., 2026) and Ruprecht et al. (Ruprecht et al., 2020). Their continuous response labels differ from the binary target endpoint. A shared masked-protein encoder of width 64 and drug encoder of width 32 therefore feed source-specific prediction heads. Target fitting uses its binary efficacy objective, Lin predicts continuous CellTiter-Glo response, and Ruprecht predicts an interval-valued potency outcome. Target data retain the original ten laboratories. Lin and Ruprecht each add one study-level laboratory. Shared encoders use sample-weighted aggregation, whereas each private head is updated only by its own source laboratories. The common laboratory and coordinator interfaces are reused. This adapter is compared against its own target-only control, separately from the main-table PTPC predictor.

The Lin subset contains 28 drug conditions (20 source-training, eight source-development), spanning five training and two development drugs. Inputs are 24-hour, 10-μM protein log-fold changes, and the response is the mean of the three released CellTiter-Glo ratio measurements. The HCT16/HCT116 alias is excluded from auxiliary training. The Ruprecht subset contains 109 matched drug conditions (85 training, 24 development; 17 and five drugs respectively). Protein log-fold changes at 24 hours are paired with CellTiter-Glo EC50 values in $\log_{10}$ nM: 49 exact, 18 approximate and 42 right-censored outcomes. Squared distance to the encoded potency interval is the loss. The adapter treats explicit lower bounds and values at or above 10,000 nM as right-censored and encodes approximate values with a factor-of-two interval. These are analysis conventions. Profiling dose is not a model input, although source acquisition uses dose conditions chosen in relation to drug response. The target retains 386 training and 105 development examples, with 148 held-out observations across 92 compounds and 22 positives.

All four source settings receive the same 100-round procedure. A two-proposal Luna loop is executed independently for each setting. These 12 fits and eight valid proposal calls produce eight fixed/loop evaluations, with identical held-out ranking scores within each source arm (Table B.6).

Table B.6: Independent-source proteomics auxiliary pilot, seed 61. Fixed design and a two-proposal Luna loop use the same shared-encoder/source-specific-head model and 100-round training allocation. Every source arm and both selection modes are retained. The heldout AP gain from this short loop is zero in all four source conditions; source-transfer gains are a different comparison.

Training sources	Selection	AP ↑	Δ vs target only (pp)
Target only	Fixed	34.98	+0.00
Target only	Loop	34.98	+0.00
+ Lin	Fixed	35.01	+0.03
+ Lin	Loop	35.01	+0.03
+ Ruprecht	Fixed	**35.50**	+0.52
+ Ruprecht	Loop	**35.50**	+0.52
+ Both	Fixed	35.31	+0.34
+ Both	Loop	35.31	+0.34

Target-only AP/AUROC are 34.98/74.75. Adding Lin yields 35.01/74.78, Ruprecht 35.50/74.28 and both 35.31/74.17. Paired compound bootstrap (1,000 resamples) gives 95% intervals for the AP changes of $[-0.20, 0.28]$, $[-1.46, 2.64]$ and $[-1.62, 2.49]$ points, respectively. These results measure transfer from two independent external studies and provide the source-matched controls for the two-proposal comparison.

Three-source proteomic transfer. The third auxiliary study, decryptE (Eckert et al., 2025), measures Jurkat proteomes after 18 hours of drug exposure at 1, 10, 100, 1,000 and 10,000 nM. Its source objective predicts the released continuous metabolic-activity measurement relative to DMSO. We align compounds using the published names, explicit alternative names and structures, and match protein features by UniProt identifier. Inputs comprise log-transformed measured normalized protein intensities, an observation mask, drug fingerprints and dose indicators. Splits group all doses of the same chemical graph together, and normalization is fitted on source-training measurements. Target development and test compounds are excluded by canonical identity and stereo-agnostic chemical graph.

The repository's 145 main processed matrices contain one additional manually classified Vincristine copy. The uniform standard release therefore provides 144 compounds. Pre-training identity checks exclude 12 compounds overlapping the target development/test sets, six records with unsupported internal names and two records with conflicting released chemical-identity/structure annotations. The resulting source contains 124 distinct compounds and 620 dose conditions: 495 training observations from 99 compounds and 125 development observations from 25 compounds. The source manifest records every exclusion and preserves the original annotations.

We append a private continuous-response head to the shared adapter and instantiate this same four-head model in every control. All original encoder and head tensors retain identical seed-61 initialization. The four prespecified arms are target only, target plus decryptE, target plus Lin and Ruprecht, and target plus all three studies. Every arm completes 100 rounds with the same local optimizer and sample-weighted shared-parameter aggregation. Checkpoint selection uses equal-laboratory target-development AP with binary cross-entropy as the tiebreaker, and auxiliary development scores are diagnostic. These fixed-design comparisons use zero proposal slots and are separate from the preceding two-source research-loop study.

Table B.7: Three-source proteomic transfer on PTPC. All four arms use the same four-head adapter, seed 61, 100-round fitting schedule and target-development checkpoint rule. AP and AUROC are multiplied by 100; bold identifies the best value in each metric. The target test set contains 148 observations from 92 compounds. These are fixed-design source comparisons; the separate two-source proposal study appears in Table B.6.

Additional source	AP ↑	AUROC ↑	Δ AP (pp)
Target only	34.98	**74.75**	+0.00
+ decryptE	35.03	74.21	+0.05
+ Lin + Ruprecht	35.31	74.17	+0.34
+ Lin + Ruprecht + decryptE	**35.68**	74.28	+0.70

Combining all three sources raises target AP from 34.98 to 35.68, a 0.70-point gain. Relative to Lin and Ruprecht together, adding decryptE increases AP by 0.37 points and AUROC by 0.11 points. Target-only training retains the highest AUROC across the four arms (74.75 versus 74.28 for all three). These source-transfer estimates use one seed. The four complete CPU fits take 101.21 seconds in total, and sealed checkpoints and predictions are independently rescored with exact agreement.

One study per training laboratory. To assign every training laboratory to a distinct source, we additionally merge all 386 target training examples into one laboratory. Lin, Ruprecht and decryptE each remain a separate source laboratory. The four arms use $K = 1, 2, 3, 4$ laboratories and respectively 386, 881, 491 and 986 training conditions. Their four-head architecture, seed-61 initialization, local loss, sample-weighted shared-parameter aggregation and source-private head ownership are unchanged. The merged target laboratory performs seven minibatches per local epoch, instead of aggregating updates from ten smaller target partitions. This is therefore a separately trained protocol, with a matching target-only control.

Table B.8: One worker per proteomic study. The target training pool is merged into one client; Lin, Ruprecht and decryptE each form one additional client. K counts training clients and N counts training conditions. All four fixed-design arms use 100 rounds, the same initialization (seed 61), and equal-weight evaluation over the original ten target development panels. AP and AUROC are percentages on the common target held-out set; best values are bold.

Training sources	K	N	AP ↑	AUROC ↑
Target study	1	386	34.41	72.33
+ decryptE	2	881	36.58	73.81
+ Lin + Ruprecht	3	491	36.09	72.69
+ Lin + Ruprecht + decryptE	4	986	**37.53**	**74.39**

The original ten target-development panels still select checkpoints with equal panel weights, independently of the number of training laboratories. All four configurations complete 100 rounds and use zero proposal slots. Held-out AP and AUROC are independently recomputed from saved predictions, with maximum absolute discrepancy 1.11×10^{-16}. The full CPU run takes 76.18 seconds.

With one laboratory per study, all three auxiliary sources improve target AP from 34.41 to 37.53 and AUROC from 72.33 to 74.39. Adding decryptE to Lin and Ruprecht increases AP by 1.44 points. These single-seed comparisons measure source transfer under a fixed design, varying available training data and participating studies.

Within-study proteomic contexts. HCC1143, HCC1806 and MDA-MB-453-1 are drawn from the mtPTDS supplementary collection (Sun et al., 2026), each with 47 retained drug conditions and 3,501 proteins shared with the target panel. They are three biological contexts within one study. Target development/test compounds are excluded from source training. These experiments use the original proteomic adapter and seeds 61–63.

Table B.9: Within-study PTPC context transfer, reported separately from independent-source transfer. HCC1143, HCC1806 and MDA-MB-453-1 belong to the same mtPTDS study. Values are mean ± sample SD over seeds 61–63, multiplied by 100. Confidence intervals resample target compounds while retaining paired seed predictions. These contexts are not additional independent studies; the original model and results are unchanged.

Additional context	AP ↑	Δ AP (pp)	Paired 95% CI
Target only	34.59 ± 5.66	+0.00 ± 0.00	[0.00, 0.00]
+ HCC1143	34.94 ± 5.15	+0.35 ± 0.56	[-1.22, 1.60]
+ HCC1806	35.23 ± 5.35	+0.64 ± 0.32	[-0.78, 1.77]
+ MDA-MB-453-1	35.20 ± 5.31	+0.61 ± 0.44	[-0.85, 2.04]
+ All three	**35.47** ± **6.36**	+0.87 ± 8.89	[-2.44, 3.99]

B.3 Research-model and six-slot proposal comparison

The short study uses the 12 configurations in the registered executable design space of Appendix A. Direct search receives untried identifiers without measured scores, while the loop additionally receives accumulated development evidence. Both modes share initialization and the first proposal. Each slot allows two attempts to return the four-field proposal schema, and failed slots consume budget and remain in the trace. Every valid candidate restarts from the common task/seed initialization and is fitted for 100 rounds. Prefixes at 0, 1, 2, 4 and 6 slots are evaluated after the trajectory is complete. An exact fit cache can reuse an identical configuration while retaining its logical training-budget charge.

Qwen uses the local sampling settings in Appendix A. The second proposer is the `gpt-5.6-luna` service, accessed in a text-only session with tool use disabled, with request metadata and responses retained. The comparison holds task messages, editable designs, training procedure and proposal budget fixed across backends. Service-level decoding controls and parameter count are not exposed by the Luna interface.

All five endpoints have both backends and both search modes. Among the ten direct/loop pairs, Norman/Luna improves by 0.81 Top-1 points, DTI/Luna decreases by 0.33 AUROC points and eight pairs tie. The twenty DTI prefix scores were independently recomputed from saved predictions for all

Table B.10: Original 12-design/six-slot heldout results. DTI seed 42; four light endpoints seed 61. All displayed budgets are prefixes of this single protocol; they are not selected by test performance. Every candidate uses 100 rounds. Scores use each original task metric $\times 100$; bold marks the best displayed score within a row, including ties.

Task	Proposer	Fixed	Direct@6	Loop@6
DTI	Luna	94.27	**94.60**	94.27
DTI	Qwen	**94.27**	**94.27**	**94.27**
PTPC	Luna	**37.27**	**37.27**	**37.27**
PTPC	Qwen	**37.27**	**37.27**	**37.27**
VCC	Luna	**31.50**	**31.50**	**31.50**
VCC	Qwen	**31.50**	**31.50**	**31.50**
Norman	Luna	15.48	22.30	**23.11**
Norman	Qwen	15.48	**22.30**	**22.30**
Tahoe	Luna	25.37	**26.87**	**26.87**
Tahoe	Qwen	**25.37**	**25.37**	**25.37**

Table B.11: Original 12-design/six-slot search dynamics. DTI seed 42; four light endpoints seed 61. D/L means direct/loop. The common target is the lower of the two terminal development primary scores, so it is attainable by both runs; first records its earliest attainment. Last gain is the last strictly improving primary-score slot; last retain also includes loss-tiebreak selections. These are retrospective diagnostics, not stopping rules. Slot 0 is initialization.

Task	Proposer	Dev. target	First D/L	Last gain D/L	Last retain D/L	Valid D/L
DTI	Luna	94.71	0/0	6/0	6/0	6/6
DTI	Qwen	94.71	0/0	0/0	0/0	6/6
PTPC	Luna	66.72	0/0	0/0	0/0	6/6
PTPC	Qwen	66.72	0/0	0/0	0/0	6/5
VCC	Luna	24.04	3/6	3/6	3/6	6/6
VCC	Qwen	23.65	0/0	0/0	0/0	6/6
Norman	Luna	51.70	5/5	5/5	5/5	6/6
Norman	Qwen	51.70	3/4	3/4	3/4	6/6
Tahoe	Luna	30.00	0/0	0/0	6/3	6/6
Tahoe	Qwen	30.00	0/0	0/0	0/0	4/4

5,505 test pairs, with checkpoint and output hashes verified. Figure 4 shows the complete prespecified prefix sequence for every endpoint, alongside its development selection trajectory.

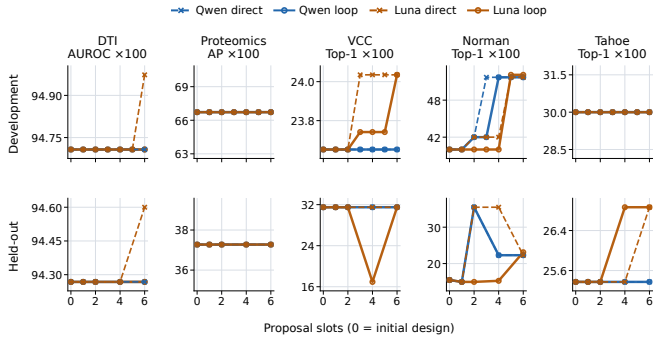

Figure 4: Research trajectories with and without evidence history in the six-slot protocol. Curves report development selection and held-out scores at each proposal prefix, where each candidate receives 100 rounds and slot 0 is the initial design.

B.4 Extended search and task-dependent saturation

The extended study expands the registered executable design space to 36 configurations: three learning rates, three weight decays, two coordinator-momentum values and two residual-head settings. It runs 24 proposal slots for both backends and modes on PTPC, VCC, Norman and Tahoe. Prefixes are 0, 1, 2, 4, 6, 8, 12, 16, 20 and 24. Initialization, full candidate fitting, development selection and invalid-slot accounting are unchanged. DTI's backend comparison is reported in the six-slot study in B.3.

Table B.12: Extended 36-design/24-slot heldout results. Four light endpoints, seed 61; no native DTI 24-slot result is implied. All displayed budgets are prefixes of this single protocol; they are not selected by test performance. Every candidate uses 100 rounds. Scores use each original task metric $\times 100$; bold marks the best displayed score within a row, including ties.

Task	Proposer	Fixed	Direct@6	Loop@6	Direct@24	Loop@24
PTPC	Luna	**37.27**	**37.27**	**37.27**	**37.27**	**37.27**
PTPC	Qwen	**37.27**	**37.27**	**37.27**	**37.27**	**37.27**
VCC	Luna	**31.50**	16.94	16.94	20.23	20.01
VCC	Qwen	**31.50**	**31.50**	**31.50**	19.61	19.61
Norman	Luna	15.48	**35.48**	23.11	23.11	23.11
Norman	Qwen	15.48	**22.30**	**22.30**	**22.30**	**22.30**
Tahoe	Luna	25.37	13.43	**26.87**	13.43	13.43
Tahoe	Qwen	25.37	**26.87**	**26.87**	13.43	13.43

Table B.13: Extended 36-design/24-slot search dynamics. Four light endpoints, seed 61; no native DTI 24-slot result is implied. D/L means direct/loop. The common target is the lower of the two terminal development primary scores, so it is attainable by both runs; first records its earliest attainment. Last gain is the last strictly improving primary-score slot; last retain also includes loss-tiebreak selections. These are retrospective diagnostics, not stopping rules. Slot 0 is initialization.

Task	Proposer	Dev. target	First D/L	Last gain D/L	Last retain D/L	Valid D/L
PTPC	Luna	66.72	0/0	0/0	0/0	24/24
PTPC	Qwen	66.72	0/0	0/0	0/0	19/14
VCC	Luna	27.68	16/9	16/10	16/10	24/24
VCC	Qwen	24.83	18/23	18/23	18/23	11/16
Norman	Luna	52.10	22/4	22/4	22/7	24/24
Norman	Qwen	51.70	2/5	2/5	22/19	15/12
Tahoe	Luna	35.00	5/13	5/13	5/13	24/24
Tahoe	Qwen	35.00	18/14	18/14	18/14	14/15

Luna produces valid designs in all 192 attempted slots, and Qwen in 116 of 192. Across 16 mode trajectories, the primary development score is unchanged in the final eight slots in 12 cases, and the

full score/loss selection key is unchanged in ten. Each trajectory retains 11–24 unvisited designs, so these plateaus occur before the registered executable design space is exhausted.

Table B.13 measures the first slot attaining the smaller terminal development score within each direct/loop pair, a common target reached by both modes. Luna reaches Norman's common target earlier with feedback (4 versus 22 slots), whereas direct search reaches Tahoe's target earlier (5 versus 13). Proteomics retains the initial design. Table B.15 connects Norman's trajectory to concrete configuration changes: reducing the learning rate to 10^{-4} gives an early development gain, and removing the residual head reaches the final primary-score target. Subsequent loss-tiebreak selections can change the retained design without increasing that primary score.

Within this extended protocol, Qwen's VCC held-out Top-1 decreases from 31.50 at slot 6 to 19.61 at slot 24 while development selection improves. The development and held-out curves report this divergence alongside proposal validity and the remaining unvisited configurations.

B.5 Executed designs and proposal traces

The short protocol uses the D00–D11 configurations in Appendix A.2. The extended registered executable design space contains all $3 \times 3 \times 2 \times 2 = 36$ combinations in Table B.14. An identifier LiWjMkRl selects the indexed value of each factor, with $i, j \in \{0, 1, 2\}$ and $k, l \in \{0, 1\}$. For example, L0W1M0R1 specifies learning rate 10^{-4}, weight decay 10^{-3}, zero coordinator momentum and an enabled residual module. Every configuration has proximal coefficient zero, and the task-specific prediction interface remains fixed.

Table B.14: Exact factor-to-index mapping for the extended 36-design library. Identifiers concatenate the four prefix–index pairs in L, W, M, R order, and all combinations of the listed values are included.

Factor	Identifier prefix	Index 0	Index 1	Index 2
Learning rate	L	10^{-4}	10^{-3}	10^{-2}
Weight decay	W	0	10^{-3}	10^{-2}
Coordinator momentum	M	0	0.9	–
Residual module	R	disabled	enabled	–

The reproducibility archive retains all 504 attempted proposal slots and their acceptance decisions: 120 slots in the short protocol and 384 in the extended protocol. Its machine-readable trace and expanded tables record each proposal's configuration, changes from the incumbent and measured development score.

Table B.15: All retained proposals for the Norman/Luna extended 36-design, 24-slot study, seed 61. The two modes share initialization and first proposal. lr, wd, m and R denote learning rate, weight decay, server momentum and residual-head indicator. Development selection is followed by evaluation at every prespecified heldout prefix; the full attempted-slot trace is retained separately.

Mode	Slot	Executed configuration	Dev. Top-1
Direct	0	Common initial recipe	40.00
Direct	1	lr=0.001, wd=0.001, m=0, R=1	40.00
Direct	2	lr=0.001, wd=0.001, m=0.9, R=1	42.00
Direct	7	lr=0.001, wd=0.01, m=0.9, R=1	44.60
Direct	10	lr=0.0001, wd=0, m=0, R=1	51.70
Direct	11	lr=0.0001, wd=0.001, m=0, R=1	51.70
Direct	22	lr=0.0001, wd=0.001, m=0, R=0	52.10
Loop	0	Common initial recipe	40.00
Loop	1	lr=0.001, wd=0.001, m=0, R=1	40.00
Loop	2	lr=0.0001, wd=0.001, m=0, R=1	51.70
Loop	4	lr=0.0001, wd=0.001, m=0, R=0	52.10
Loop	7	lr=0.0001, wd=0, m=0, R=0	52.10

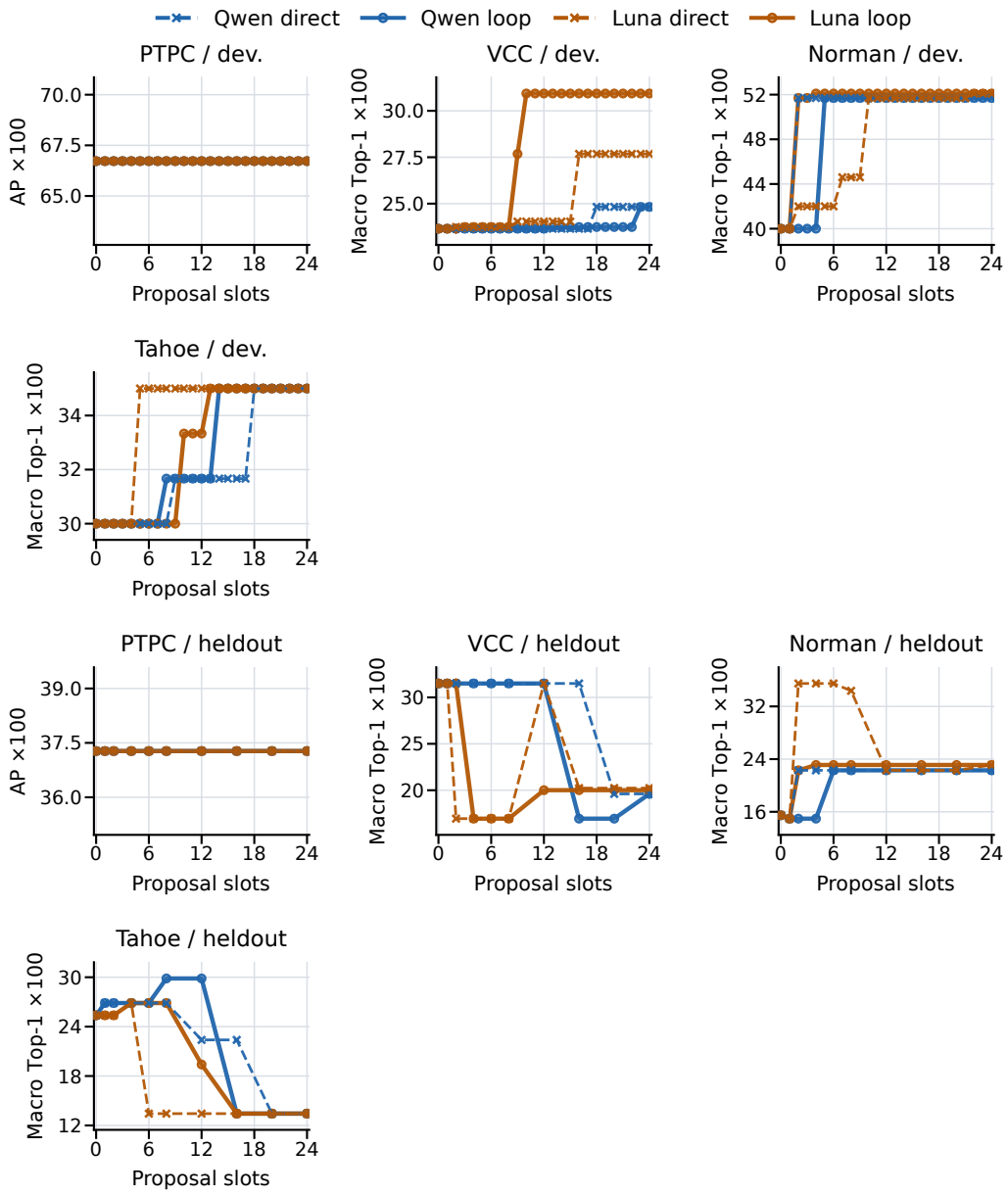

Figure 5: Development selection (upper panels) and held-out performance (lower panels) in the separate 36-design, 24-slot protocol on four endpoints. Curves are paired within the same backend, initialization and design space. Score plateaus may still include loss-tiebreak selections. All prespecified prefixes are displayed. Six-slot points here belong to this 36-design protocol, not the separate 12-design experiment.

Appendix C. Public self-improving-agent comparison

C.1 Shared biological task interface

AI-Scientist-v2 (Yamada et al., 2025) and AI-Researcher (Tang et al., 2025) use BioCoLoop's fitting and evaluation service, at source versions `96bd51617cfd` and `f9a6f8480860`, respectively (SakanaAI, 2025; HKUDS, 2025). Both use the registered executable design space of twelve configurations, at most six proposals, 100 rounds per valid candidate and the shared held-out scorer. The service fits D00 afresh, then executes submitted `design_id` values; the hypothesis, experiment and expected-effect fields record rationale. Repeated or invalid requests consume slots, and agents may stop early. The original development score/loss rule selects checkpoints.

AI-Scientist-v2 retains its experiment manager, tree-search paths, reviews and journal. AI-Researcher retains native idea selection, planning, implementation, review handoffs and iterative analysis. Prompts use the registered design interface; arbitrary generated-code execution, external retrieval, figure review and manuscript generation are disabled. The matched blocks use local Qwen2.5-7B-Instruct (temperature 0.7, top-p 0.9, at most 512 generated tokens per call) and GPT-5.6 Luna (low reasoning effort, text-only bridge, tools disabled). Adapters allow at most 160 model calls and 28,000 input tokens per call, raised to 65,536 for the four starred Luna AI-Researcher retries (C.2). Qwen uses generation seeds; Luna labels identify training/search repetitions. Per-run call counts, token usage and transport receipts document the variable language-model costs.

C.2 Prediction quality and execution outcomes

A scored run completes its agent's research process within the registered cap, seals development selection and passes independent held-out rescoring. Table 1 summarizes seeds 42–44; fixed, direct and BioCoLoop rows are complete. Daggers mark partial cells and dashes mark unscored cells, without imputation. Partial cells are Qwen2.5-7B-Instruct AI-Researcher on VCC ($2/3$ seeds, F_{tool}) and GPT-5.6 Luna AI-Scientist-v2 on Tahoe ($2/3$, F_{svc}).

Four Luna AI-Researcher cells exceeded the 28,000-token admission (F_{ctx}) and were re-run at 65,536 tokens with the same tasks, seeds, design space, six-proposal cap, training and scorer. Their completed cells carry stars, with per-seed records retained. BindingDB (DTI), the longest native refinement trajectory, remains unscored ($0/3$): one frozen seed reached 29,167 input tokens (F_{ctx}), and two terminated in service transport (F_{svc}). Extended-admission retries also ended in transport before held-out selection was sealed. The final table column gives mean rank across completed endpoints.

Receipts distinguish native pipeline (F_{pipe}), tool-transport (F_{tool}), registered context-limit (F_{ctx}) and Luna service-transport (F_{svc}) terminations before sealed selection. The seed-42 Qwen AI-Scientist-v2 DTI example trains three candidates before substage schema validation fails, leaving its incumbent unsealed and unscored.

Held-out scoring follows sealed development selection. An independent verifier recomputes each scored cell's metric from saved predictions and checks checkpoint and artifact hashes. DTI discrepancies are at most 10^{-15}. Records retain raw calls, native stage history, candidate requests, selections and verification receipts.

C.3 Task-transport repair

The AI-Researcher adaptation removes repeated task packets from stage prompts, preserving the complete task and evidence in system prompts, native reasoning and tool results. The bridge accepts an unambiguous `parameters` alias for `arguments` and standard single-tool-call envelopes without inferring actions or values. This revision preserves model weights, context allowance, training and scoring. It applies across all five endpoints and the Luna sensitivity; failures remain recorded.