CELLRFT: REINFORCEMENT FINE-TUNING FOR SINGLE-CELL PERTURBATION MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

Predicting cellular responses to perturbations supports the study of gene function, disease mechanisms, and therapeutic strategies. Despite advances in single-cell perturbation modeling, existing models typically optimize surrogate losses that do not directly reflect the biological criteria used for evaluation, so better data fitting need not yield better biological predictions. To address this mismatch, we introduce **CellRFT**, a reinforcement fine-tuning framework that uses biological evaluation as direct training feedback. CellRFT uses policy-gradient optimization to learn from non-differentiable evaluations of generated cell populations and integrates multiple biological rewards through hierarchical reward aggregation. Comprehensive experiments demonstrate CellRFT's applicability across different pretrained models and effectiveness in improving perturbation prediction, reveal that optimizing one biological criterion can help or hinder others, and show that complementary rewards can improve criteria beyond those directly optimized, offering a way to probe how biological metrics shape model behavior, with the potential to inform evaluation design. Code will be made available.

1 INTRODUCTION

Predicting cellular responses to genetic and chemical perturbations extends experimental coverage beyond feasible screens (Bunne et al., 2024; Adduri et al., 2026), supporting the study of gene function and disease mechanisms and the discovery of therapeutic strategies (Dixit et al., 2016; Bunne et al., 2024). Perturbation models are typically trained with reconstruction (Lotfollahi et al., 2019), denoising (Yuan et al., 2026), or distribution-matching (Adduri et al., 2026) losses, then evaluated on expression fidelity, affected-gene recovery, and perturbation discrimination using suites such as Cell-Eval (Adduri et al., 2026). These surrogate losses do not directly optimize the biological criteria, creating a *training–evaluation objective mismatch* in which better data fitting need not improve biological prediction (Zhu et al., 2025; Ahlmann-Eltze et al., 2025).

Learning directly from biological evaluation could reduce this mismatch, but many metrics involve non-differentiable thresholds or rankings. To use such feedback, reinforcement learning (RL) fine-tuning offers a potential route without differentiating through the evaluator, with successful applications in language models (Ouyang et al., 2022; Guo et al., 2025; Srivastava & Aggarwal, 2025) and diffusion-based image generation (Black et al., 2024; Liu et al., 2025; Zheng et al., 2026). In perturbation prediction, metrics such as affected-gene recovery assess responses across cells, so RL formulations requiring individual-output scores cannot be transferred directly. These metrics also assess distinct biological properties, so using them jointly as rewards requires understanding whether optimizing one helps or harms the others, a task-specific question that remains insufficiently explored.

To address these challenges, we propose **CellRFT**, a reinforcement fine-tuning framework that brings biological evaluation into single-cell perturbation model optimization and examines how reward choices shape predictive performance (Figure 1a). CellRFT uses policy-gradient RL to optimize biological rewards without differentiating through the evaluation metrics. To bridge the gap between individual generation trajectories and population-level evaluation, it generates multiple candidate sets of cells for each perturbation and shares each set's relative reward feedback across its constituent trajectories. To integrate multiple biological criteria, CellRFT uses hierarchical reward aggregation, averaging first within evaluation categories and then across categories to

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.

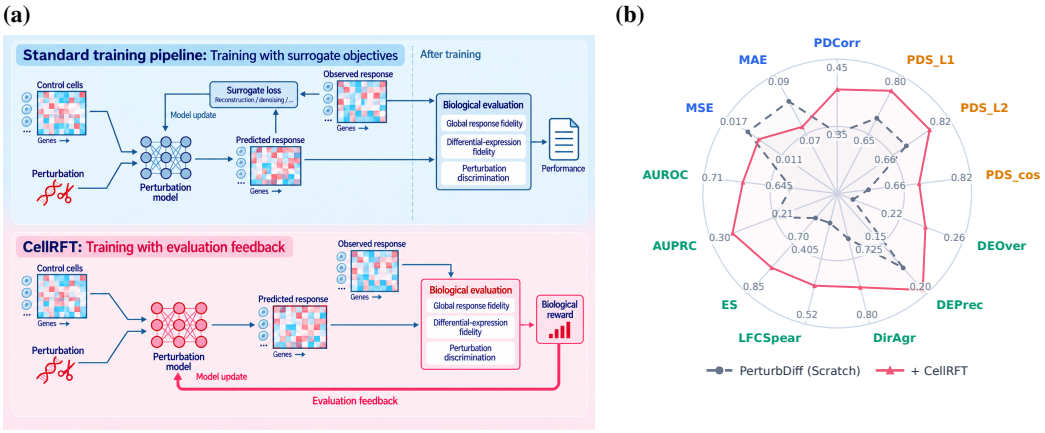


Figure 1: **Training–evaluation objective mismatch.** **(a)** Standard pipelines optimize surrogate objectives, such as reconstruction and denoising, whose improvement need not yield higher biological fidelity. CellRFT instead uses biological evaluation feedback to guide model updates through reinforcement learning. **(b)** CellRFT fine-tunes PerturbDiff (Scratch) (Yuan et al., 2026) on Replogle. The resulting model (+CellRFT) improves all 13 evaluated metrics, as defined in Table 2.

give each category equal total weight regardless of its number of metrics. We further investigate reward integration through systematic comparisons of individual rewards, category combinations, and aggregation rules, together with an analysis of how optimizing each reward affects other evaluation criteria. Comprehensive experiments demonstrate CellRFT's applicability across different pretrained models and effectiveness in improving perturbation prediction through direct biological feedback. Figure 1b illustrates these gains for PerturbDiff (Scratch), where CellRFT improves all 13 reported evaluation metrics. The experiments further reveal that optimizing one biological criterion can help or hinder others, while complementary rewards can jointly improve criteria beyond those directly optimized. Understanding these interactions can inform both the choice of training rewards and the selection of criteria for evaluating biological predictions.

In summary, our contributions are threefold:

- We introduce CellRFT, a reinforcement fine-tuning framework that turns biological evaluation into direct training feedback for single-cell perturbation models, addressing the mismatch between training objectives and biological evaluation criteria.
- We reveal how biological rewards reinforce or conflict with one another during optimization, showing that distinct evaluation criteria can benefit from shared training objectives and informing the design of complementary rewards.
- Comprehensive experiments demonstrate CellRFT's applicability across different pretrained models and effectiveness in improving perturbation prediction, and show how reward composition and aggregation can balance improvements across biological criteria.

2 Related Work

2.1 Single-Cell Perturbation Modeling

Single-cell perturbation prediction seeks to infer post-perturbation gene-expression profiles from control cells, a specified perturbation, and the cellular context. To represent the effect of a perturbation on cellular state, scGen (Lotfollahi et al., 2019) learns shifts in a latent expression space. CPA (Lotfollahi et al., 2023) further separates perturbation and cellular covariates in that space, allowing their effects to be recombined for prediction under new conditions. For generalization to unseen genetic perturbations, GEARS (Roohani et al., 2024) uses gene–gene relationships to share information between observed and unobserved perturbation targets. Beyond representing these conditional effects, learning from single-cell measurements requires handling unpaired observations: sequencing destroys the measured cells, so the same cell cannot be observed before and after per-

turbation. CellOT (Bunne et al., 2023) addresses this by learning an optimal transport map between control and perturbed populations, while STATE (Adduri et al., 2026) learns responses using a population-matching objective. Diffusion and flow models provide flexible generative formulations for these response distributions: Squidiff (He et al., 2026) learns a denoising process, and CellFlow (Klein et al., 2025) learns conditional flows. PerturbDiff (Yuan et al., 2026) extends diffusion to the space of cell distributions to capture variation among possible response distributions under the same observed conditions.

Despite their different modeling strategies, these methods typically optimize surrogate objectives, such as reconstruction, denoising, or distribution matching, rather than the biological criteria used for evaluation. This training–evaluation objective mismatch means that improvements in the training loss need not translate into better biological predictions, motivating direct use of biological evaluation as training feedback.

2.2 Reinforcement Learning and Biological Evaluation

Reinforcement learning (RL) has driven substantial progress across diverse fields by incorporating evaluation feedback into training and directing optimization toward task-level objectives. In language-based assistance and reasoning, likelihood-based training alone does not ensure that outputs follow user instructions, solve problems correctly, or execute successfully as code. RL introduces human preference judgments, answer verification, and execution results as rewards, improving instruction following, reasoning accuracy, and the completion of software engineering tasks (Ouyang et al., 2022; Guo et al., 2025; Cao et al., 2026). In visual generation, denoising and flow-matching losses do not directly measure whether images satisfy perceptual preferences or accurately depict the objects and relations specified in a prompt. Optimizing rewards derived from preference models and multimodal evaluators improves visual quality and compositional consistency, bringing training closer to the criteria used to judge generated images (Black et al., 2024; Liu et al., 2025; Zheng et al., 2026). In molecular design, reproducing known chemical structures does not ensure the activity and physicochemical properties needed for a useful compound. Property-based RL rewards guide optimization toward these requirements, enriching candidates for predicted target activity and, when combined with synthesis constraints, supporting the discovery of experimentally validated antibiotics (Olivecrona et al., 2017; Swanson et al., 2026).

In perturbation modeling, biologically meaningful predictions should capture overall transcriptional responses, recover the genes affected by an intervention, and distinguish the effects of different perturbations. To assess these properties, prior studies have developed and adopted complementary metrics of expression fidelity (Lotfollahi et al., 2019; Bunne et al., 2023), differential-expression recovery (Zhu et al., 2025), and perturbation discrimination (Wu et al., 2025; Roohani et al., 2025). Although Cell-Eval, released alongside STATE, consolidates these evaluation dimensions in a common suite (Adduri et al., 2026), how the corresponding metrics should jointly guide model optimization remains unresolved. Using them as biological rewards requires understanding whether optimizing one criterion reinforces or compromises performance on the others.

The most closely related works, developed concurrently with ours, are D^2R^2 (Fan et al., 2026) and PerturbCellRL (Wu et al., 2026). Both seek to improve the biological consistency of predicted cells, combining RL with regulatory relationships or pathway knowledge to guide prediction. In contrast, CellRFT offers a complementary perspective by studying two questions that remain insufficiently explored: how to integrate commonly used Cell-Eval metrics into RL training, and how their complementarity and redundancy should guide the construction of a unified biological objective.

3 Method

This section first introduces the problem formulation and biological evaluation criteria, then presents CellRFT and its training procedure.

3.1 Problem Formulation and Biological Evaluation

Problem Definition. For cellular context c (e.g., cell type or batch) and perturbation τ, consider unpaired control population $X_{\mathrm{ctrl}} \in \mathbb{R}^{N_{\mathrm{ctrl}} \times |\mathcal{G}|}$ and perturbed population $X_{\mathrm{pert}} \in \mathbb{R}^{N_{\mathrm{pert}} \times |\mathcal{G}|}$, with

cells in rows and genes $\mathcal{G}$ in columns. Perturbation modeling aims to learn a conditional generator $p_\theta(\cdot \mid X_{\text{ctrl}}, c, \tau)$ whose predicted cell distribution $\widehat{P}_{c,\tau}$ approximates the true perturbed-cell distribution $P_{c,\tau}$, i.e., $\widehat{P}_{c,\tau} \approx P_{c,\tau}$, while maximizing the expected biological reward of predictions $\widehat{X} \in \mathbb{R}^{N \times |\mathcal{G}|}$:

$$\max_\theta \ \mathbb{E}_{\widehat{X} \sim p_\theta(\cdot \mid X_{\text{ctrl}}, c, \tau)} \left[r\Big(\widehat{X}; X_{\text{ctrl}}, X_{\text{pert}}, c, \tau\Big) \right], \tag{1}$$

where r is a reward function given by the biological evaluation metric itself when larger values indicate better performance, and by its negative otherwise.

Biological Evaluation. Commonly used metrics are available in Cell-Eval, introduced alongside STATE (Adduri et al., 2026). These metrics assess agreement between predicted and ground-truth cell populations from three complementary perspectives: (i) *global response fidelity*, which measures how well predictions reproduce the overall expression response averaged across cells; (ii) *differential-expression fidelity*, which focuses on recovery of significantly affected genes and their response patterns; and (iii) *perturbation discrimination*, which evaluates whether a predicted response can be matched to the correct perturbation. Table 2 in Appendix A summarizes the 14 selected metrics; see Yuan et al. (2026) for further details. Scores are computed per perturbation τ and averaged over the evaluated perturbations $\mathcal{P}$ unless stated otherwise.

3.2 CellRFT: Reinforcement Fine-Tuning with Biological Evaluation

Perturbation modeling aims to maximize the expected biological reward defined in Eq. (1). In practice, models are typically trained with differentiable surrogate losses, but reducing these losses does not necessarily improve biological evaluation scores. Using biological evaluation directly as training feedback presents three challenges: (i) *non-differentiability*: significance thresholds, rankings, and gene-set operations in many metrics prevent direct backpropagation; (ii) *evaluation granularity*: biological metrics evaluate cell populations, while existing sample-level reinforcement learning (RL) formulations for generative models require a reward for each individual sample; and (iii) *multi-reward integration*: biological evaluation metrics remain largely unexplored as RL rewards, leaving unclear how each shapes model behavior and how to integrate these rewards into a unified training objective.

To handle these, we propose **CellRFT**, a reinforcement fine-tuning framework for single-cell perturbation modeling. It handles non-differentiability through policy-gradient RL, bridges the evaluation granularity gap through population-level sampling and shared feedback, and combines multiple rewards through hierarchical averaging. Figure 2 summarizes the sampling, reward and advantage computation, and policy optimization stages. The details are as follows.

Reinforcement Learning from Evaluation Feedback. Indeed, RL fine-tuning has driven substantial progress in large language models and text-to-image generation by incorporating evaluation into training. We use Flow-GRPO (Liu et al., 2025), a representative framework for image generation, to illustrate how RL bridges the training–evaluation gap.

Flow-GRPO alternates between sampling candidate images, evaluating them, and updating the generator using their relative performance. In the sampling stage, the model generates a group of candidate images $\{\widehat{\mathbf{x}}^{(g)}\}_{g=1}^{G}$ under the same text prompt. Each image is produced by a *trajectory* $\rho^{(g)}$, a sequence of stochastic generation steps, and receives a scalar evaluation *reward* $r(\widehat{\mathbf{x}}^{(g)})$. To account for differences in reward baselines and scales across prompts, Flow-GRPO normalizes rewards within each prompt group:

$$A^{(g)} = \frac{r(\widehat{\mathbf{x}}^{(g)}) - \text{mean}(\{r(\widehat{\mathbf{x}}^{(h)})\}_{h=1}^{G})}{\text{std}(\{r(\widehat{\mathbf{x}}^{(h)})\}_{h=1}^{G})}, \qquad g = 1, \ldots, G, \tag{2}$$

The resulting *advantage* $A^{(g)}$ measures relative performance within the group, reducing sensitivity of update weights to a prompt's absolute reward level and dispersion.

These advantages guide the model update by favoring trajectories that outperform others under the same prompt. The generator acts as a *policy*, with $p_\theta(\rho)$ denoting a trajectory's probability density. The reward-driven part of the local update follows the gradient of an advantage-weighted

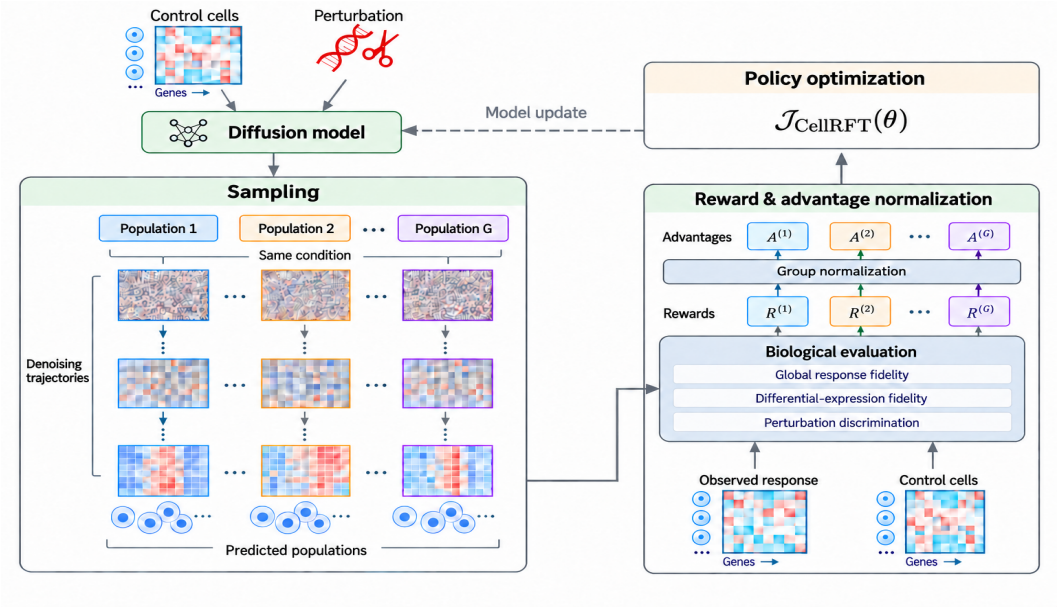


Figure 2: **Overview of CellRFT.** A diffusion model samples G candidate cell populations under the same condition. Biological evaluation compares each prediction with observed and control populations; metric rewards are averaged first within categories and then across categories to obtain $R^{(g)}$. Group normalization produces advantages $A^{(g)}$, each shared across all cells and denoising steps in its candidate population. These advantages guide policy optimization with $\mathcal{J}_{\text{CellRFT}}(\theta)$ (Eq. (7)) to update the model, completing the sampling–evaluation–optimization cycle.

log-likelihood surrogate:

$$\nabla_\theta \left[\frac{1}{G} \sum_{g=1}^{G} A^{(g)} \log p_\theta(\rho^{(g)}) \right] = \frac{1}{G} \sum_{g=1}^{G} A^{(g)} \nabla_\theta \log p_\theta(\rho^{(g)}). \tag{3}$$

Here the prompt, sampled trajectories, and advantages are held fixed. The gradient acts on the model's log likelihood, with $A^{(g)}$ supplying the evaluation feedback without requiring derivatives of the evaluator. Thus, non-differentiable evaluation can guide generation.

However, moving from image generation to perturbation modeling changes the granularity of evaluation: Flow-GRPO evaluates individual images, whereas biological evaluation assesses single-cell perturbation predictions collectively over sets of cells. Moreover, Flow-GRPO formulates optimization around a single scalar reward, while integrating multiple biological evaluation criteria into a unified reward remains an open research question. Together, these challenges make Flow-GRPO's formulation not directly applicable to perturbation modeling.

Population-Level Rewards and Hierarchical Aggregation. To incorporate population-level biological evaluation into training, CellRFT treats a sampled set of generated cells as one candidate, organizing sampling and reward comparison at the same granularity as biological evaluation.

For each perturbation τ, we first sample G candidate sets of B cells each:

$$\begin{aligned} \widehat{X}^{(g)} &= \{\widehat{\mathbf{x}}_n^{(g)}\}_{n=1}^{B}, \qquad g = 1, \ldots, G, \\ \widehat{\mathbf{x}}_n^{(g)} &\sim p_\theta(\cdot \mid X_{\text{ctrl}}, c, \tau), \qquad n = 1, \ldots, B. \end{aligned} \tag{4}$$

Then, we apply hierarchical aggregation to metric rewards, averaging first within categories and then across categories:

$$R^{(g)} = \frac{1}{|\mathcal{C}|} \sum_{k \in \mathcal{C}} \frac{1}{|\mathcal{J}_k|} \sum_{j \in \mathcal{J}_k} r_j \left(\widehat{X}^{(g)}; X_{\text{ctrl}}, X_{\text{pert}}, c, \tau \right). \tag{5}$$

Here $\mathcal{C}$ indexes the three evaluation categories in Table 2, and $\mathcal{J}_k$ contains the selected reward metrics in category k. By hierarchically aggregating rewards, we give each category equal total weight, avoiding optimization bias toward categories with more metrics.

Policy Optimization with Shared Population Advantages. We next use the aggregated rewards to iteratively update the generator, increasing the likelihood of candidate sets with higher biological rewards.

To account for differences in reward baselines and scales across perturbations, we normalize rewards within each perturbation group (Liu et al., 2025) to obtain a relative advantage for each candidate:

$$A^{(g)} = \frac{R^{(g)} - \text{mean}(\{R^{(h)}\}_{h=1}^{G})}{\text{std}(\{R^{(h)}\}_{h=1}^{G})}. \tag{6}$$

Each $A^{(g)}$ is shared across all denoising steps of the candidate's B cells.

Let $\mathbf{x}_{n,t}^{(g)}$ denote the state of cell n in candidate g at step t of a T-step denoising trajectory, with $\mathbf{x}_{n,0}^{(g)}$ the generated cell. Using the shared advantages, CellRFT maximizes the following clipped objective (Liu et al., 2025):

$$\mathcal{J}_{\text{CellRFT}}(\theta) = \mathbb{E}\left[\frac{1}{GBT}\sum_{g=1}^{G}\sum_{n=1}^{B}\sum_{t=1}^{T}\left\{\min\left(w_{n,t}^{(g)}A^{(g)}, \text{clip}\left(w_{n,t}^{(g)}, 1-\eta, 1+\eta\right)A^{(g)}\right)\right\}\right], \tag{7}$$

where the expectation is over training conditions and candidate trajectories sampled from the old policy $p_{\theta_{\text{old}}}$. The ratio $w_{n,t}^{(g)}$ compares the current denoising transition density with that of the sampling policy:

$$w_{n,t}^{(g)}(\theta) = \frac{p_\theta(\mathbf{x}_{n,t-1}^{(g)} \mid \mathbf{x}_{n,t}^{(g)}, X_{\text{ctrl}}, c, \tau)}{p_{\theta_{\text{old}}}(\mathbf{x}_{n,t-1}^{(g)} \mid \mathbf{x}_{n,t}^{(g)}, X_{\text{ctrl}}, c, \tau)}. \tag{8}$$

Clipping with threshold η limits the incentive to change this ratio in the direction favored by $A^{(g)}$.

The updated generator then samples new candidate sets for evaluation, and this sampling–evaluation–update cycle repeats throughout fine-tuning. Algorithm 1 in Appendix B summarizes the complete training procedure.

4 Experiments

This section evaluates CellRFT's applicability across different pretrained models and effectiveness in improving perturbation prediction, then examines how reward composition, aggregation, and complementarity shape improvements across biological criteria.

4.1 Experimental Setup

Dataset and Evaluation Protocol. We evaluate on Replogle (Nadig et al., 2025), following the experimental protocol of Yuan et al. (2026) with identical preprocessing and data splits. The benchmark covers 2,023 genetic perturbations across four cell lines (K562, RPE1, Jurkat, and HepG2), with predictions evaluated on 2,000 highly variable genes. HepG2 is the target cell line: training uses the other three cell lines together with 30% of HepG2 perturbations, while evaluation follows the original validation and test partitions. Preprocessing includes knockdown-efficacy filtering inherited from STATE (Adduri et al., 2026), library-size normalization, log transformation, and division of expression values by 10. Further details are provided in Yuan et al. (2026).

Baselines and Evaluation Metrics. We compare against simple statistical baselines, Mean (Kernfeld et al., 2025) and Linear (Ahlmann-Eltze et al., 2025). Mean averages expression within each perturbation; its three variants, Mean Variant (per Cell Type), Mean Variant (per Batch), and Mean Variant (Overall), instead average within each cell type, within each experimental batch, and across all cells, respectively. We also include representative perturbation models: CPA (Lotfollahi et al., 2023), STATE (Adduri et al., 2026), CellFlow (Klein et al., 2025), Squidiff (He et al., 2026), PerturbDiff (Scratch) (Yuan et al., 2026) and PerturbDiff (Finetuned) (Yuan et al., 2026). We evaluate

Table 1: Perturbation prediction performance on Replogle. Arrows indicate better performance, and bold highlights the best result for each metric within each block.

Differential-expression fidelity							
Model	DEOver ↑	DEPrec ↑	DirAgr ↑	LFCSpear ↑	AUROC ↑	AUPRC ↑	ES ↑
Mean	0.127	0.094	0.532	0.077	0.467	0.092	0.417
Mean Variant (per Cell Type)	0.178	0.087	0.743	0.492	0.404	0.078	0.417
Mean Variant (per Batch)	0.110	0.092	0.475	-0.024	0.450	0.088	0.421
Mean Variant (Overall)	0.111	0.093	0.474	-0.027	0.453	0.089	0.416
Linear	0.068	0.074	0.535	0.056	0.435	0.085	0.353
CPA	0.173	0.087	0.746	0.499	0.401	0.081	0.637
STATE	**0.196**	**0.193**	**0.778**	**0.506**	**0.633**	**0.239**	**0.818**
CellFlow	0.110	0.093	0.472	-0.033	0.434	0.086	0.442
Squidiff	0.039	0.091	0.432	0.000	0.366	0.079	0.419
PerturbDiff (Scratch)	0.190	0.174	0.702	0.342	0.625	0.210	0.623
CellRFT + PerturbDiff (Scratch)	**0.236**	**0.196**	**0.758**	**0.452**	**0.672**	**0.270**	**0.771**
PerturbDiff (Finetuned)	0.214	0.192	0.723	0.388	0.652	0.257	0.762
CellRFT + PerturbDiff (Finetuned)	**0.239**	**0.201**	**0.752**	**0.455**	**0.681**	**0.273**	**0.771**

Global response fidelity and perturbation discrimination						
Model	MSE ↓	MAE ↓	PDCorr ↑	PDS_{L1} ↑	PDS_{L2} ↑	$\text{PDS}_{\cos}$ ↑
Mean	0.0990	0.206	0.048	0.582	0.599	0.608
Mean Variant (per Cell Type)	0.0085	0.057	0.412	0.502	0.505	0.502
Mean Variant (per Batch)	0.0699	0.174	0.002	0.505	0.505	0.506
Mean Variant (Overall)	0.0711	0.175	-0.001	0.502	0.503	0.501
Linear	0.0130	0.074	0.058	0.519	0.517	0.667
CPA	0.0750	**0.054**	0.418	0.582	0.581	0.521
STATE	**0.0064**	0.055	**0.437**	**0.788**	**0.802**	**0.676**
CellFlow	0.0949	0.205	-0.003	0.507	0.507	0.495
Squidiff	4.6410	2.077	0.089	0.500	0.501	0.501
PerturbDiff (Scratch)	0.0147	0.081	0.340	0.690	0.700	0.575
CellRFT + PerturbDiff (Scratch)	**0.0135**	**0.072**	**0.405**	**0.759**	**0.768**	**0.697**
PerturbDiff (Finetuned)	**0.0146**	0.079	0.376	**0.758**	**0.762**	**0.639**
CellRFT + PerturbDiff (Finetuned)	0.0149	**0.076**	**0.409**	0.742	0.748	0.611

performance using 14 representative metrics from Cell-Eval (Adduri et al., 2026), covering global response fidelity, differential-expression fidelity, and perturbation discrimination (Table 2).

Implementation Details. We apply CellRFT to both PerturbDiff (Scratch) and PerturbDiff (Finetuned) (Yuan et al., 2026), corresponding to supervised training from scratch and supervised finetuning after pretraining, respectively. To maintain comparable reward scales and per-perturbation feedback, we select 11 metrics from Table 2, excluding the unbounded MSE and MAE and the cross-perturbation metric ES. Using these rewards, we fine-tune with a learning rate of 10^{-5} and $G = 6$ candidate sets per perturbation, each containing $B = 32$ cells.

4.2 Applicability and Effectiveness of CellRFT Across Pretrained Models

Table 1 compares statistical and learned baselines with the two PerturbDiff models before and after CellRFT on 13 Replogle metrics. The improvements in both PerturbDiff variants demonstrate CellRFT's applicability across different pretrained models and effectiveness in improving perturbation prediction. We refer to CellRFT applied to PerturbDiff (Scratch) as the scratch variant, and CellRFT applied to PerturbDiff (Finetuned) as the finetuned variant. The scratch variant improves all 13 re-

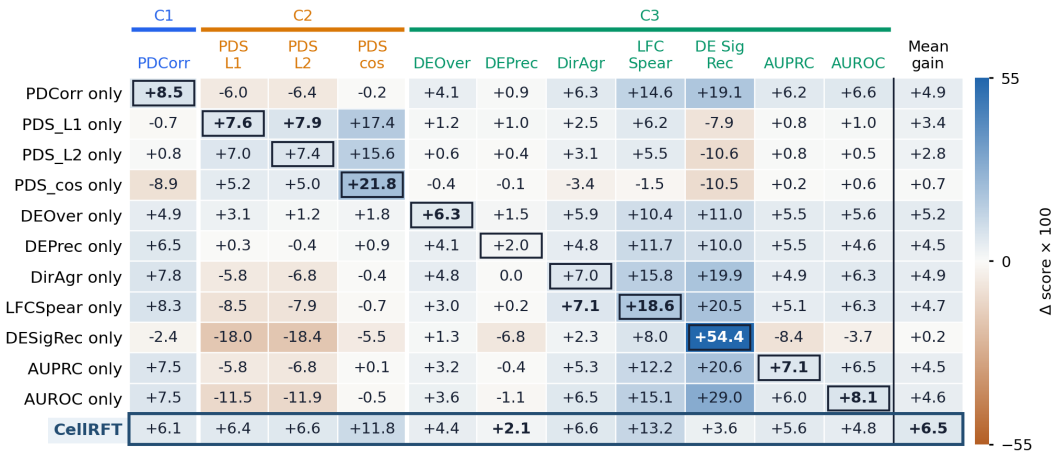


Figure 3: Cross-metric effects of single-reward fine-tuning. The ablation evaluates how using individual metrics as rewards affects performance across the evaluation suite, reporting gains over the pretrained model ($\times 100$). Single-reward variants use the corresponding evaluation metric as the sole reward, while CellRFT uses the hierarchical reward aggregation defined in Eq. (5). All single-reward variants improve their target metrics, but most incur cross-metric degradation; CellRFT achieves a more favorable overall trade-off, improving all 11 metrics with the highest mean gain.

ported metrics over PerturbDiff (Scratch). It also exceeds PerturbDiff (Finetuned) on every metric, although PerturbDiff (Finetuned) first uses additional perturbation data for pretraining before supervised fine-tuning. This result suggests that RL feedback can further extract useful capacity from a model trained without those extra pretraining data, making CellRFT useful for new cell types or perturbation panels where large-scale pretraining data are expensive to collect. The finetuned variant further improves nine metrics over PerturbDiff (Finetuned), showing that CellRFT also adds value after supervised fine-tuning.

The strongest gains concentrate on metrics that evaluate perturbation-induced gene-expression responses. In the scratch variant, DEOver increases from 0.190 to 0.236, LFCSpear from 0.342 to 0.452, AUROC from 0.625 to 0.672, and AUPRC from 0.210 to 0.270. The finetuned variant shows the same pattern, with consistent improvements across the differential-expression metrics. Consequently, both variants outperform all non-CellRFT baselines on DEOver, DEPrec, AUROC, and AUPRC. These metrics directly test whether the model identifies the affected genes and ranks their response magnitudes correctly, showing that CellRFT improves the perturbation-specific signal central to biological prediction rather than only matching average expression profiles.

The results also show that reward effects are coupled rather than metric-local. Some improvements transfer to metrics that are not used as rewards: MSE, MAE, and ES are excluded from training, yet all three improve for the scratch variant, and MAE and ES also improve for the finetuned variant. At the same time, including a metric in the joint reward does not guarantee improvement: the finetuned variant gains on differential-expression metrics but loses on all three PDS metrics. These heterogeneous responses motivate an ablation study of reward design: how individual metrics affect performance across the evaluation suite, and how combining rewards changes the resulting trade-offs.

4.3 Reward Composition and Aggregation

The main results show that gains in one biological criterion can coexist with losses in another. To test whether RL fine-tuning with a single evaluation metric as the reward reproduces these trade-offs and whether combining multiple metrics as rewards mitigates them, we design two ablations: *single-reward fine-tuning* to assess target and cross-metric effects, and *category-level combinations* to assess reward composition, including a comparison of aggregation rules using all 11 rewards. Both ablations use the 11 reward metrics from Table 2, grouped into global response fidelity (C1: PDCorr), perturbation discrimination (C2: three PDS variants), and differential-expression fidelity

	C1	C2			C3							
	PDCorr	PDS L1	PDS L2	PDS cos	DEOver	DEPrec	DirAgr	LFC Spear	DE Sig Rec	AUPRC	AUROC	Mean gain
C1 only	**+8.5**	-6.0	-6.4	-0.2	+4.1	+0.9	+6.3	+14.6	+19.1	+6.2	+6.6	+4.9
C2 only	-4.5	+7.2	+7.3	**+20.1**	+1.3	+1.0	-1.0	+0.2	-5.8	+1.3	+1.4	+2.6
C3 only	+6.2	-14.5	-14.5	-0.7	+3.6	-2.1	+6.4	**+15.6**	**+36.3**	+4.7	+7.7	+4.4
C1 + C2	+3.9	**+7.5**	**+7.7**	+11.8	+3.1	+1.8	+5.2	+10.4	-1.2	+2.6	+2.9	+5.1
C1 + C3	+8.1	-11.9	-12.4	-0.3	+3.8	-0.7	**+6.8**	+14.8	+28.8	**+6.3**	**+7.8**	+4.7
C2 + C3	+0.3	+6.9	+7.0	+16.8	+3.4	+0.5	+2.3	+4.1	+8.2	+5.0	+4.7	+5.4
Flat mean	+3.2	+5.1	+5.1	+11.7	+3.6	+0.4	+3.7	+7.1	+14.0	+5.3	+5.8	+5.9
CellRFT	+6.1	+6.4	+6.6	+11.8	**+4.4**	**+2.1**	+6.6	+13.2	+3.6	+5.6	+4.8	**+6.5**

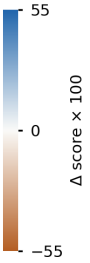


Figure 4: Cross-metric effects of reward composition and aggregation. The ablation evaluates how reward composition and aggregation affect performance across the evaluation suite, reporting gains over the pretrained model ($\times 100$). Category-level variants use all reward metrics from one evaluation category or a pair of categories. Flat mean takes the unweighted arithmetic mean of all 11 rewards, whereas CellRFT averages these rewards first within categories and then across categories (Eq. (5)). Combining complementary reward categories mitigates cross-metric trade-offs, and CellRFT achieves the highest mean gain while improving all 11 metrics.

(C3: seven metrics). Single-reward variants share the pretrained initialization, data split, and training budget. Category-level variants use all reward metrics from one category or a pair of categories; the full-reward comparison uses either an unweighted arithmetic mean (Flat mean) or hierarchical averaging (CellRFT; Eq. (5)). Evaluation covers all 11 metrics, reporting gains over the pretrained model ($\times 100$) and their arithmetic mean in the original metric scales.

As shown in Figures 3 and 4, (1) each single-reward variant improves its target metric, but most degrade other criteria, with DESigRec showing the clearest divergence between target and cross-metric performance and DEOver improving all 11 metrics; (2) combining complementary categories can mitigate these trade-offs: C3 alone reduces perturbation discrimination and C2 alone reduces DESigRec, whereas C2+C3 is the only pairwise combination that improves all 11 metrics; and (3) CellRFT improves all 11 metrics with the highest mean gain (6.5), exceeding Flat mean (5.9) and DEOver-only (5.2), despite not achieving the largest gain on every metric.

Overall, these experiments demonstrate CellRFT's applicability across different pretrained models and effectiveness in improving perturbation prediction, with particularly strong gains in differential-expression fidelity. Reward ablations show that improvements in individual metrics can incur losses elsewhere, and that reward composition and aggregation are important for balancing these effects. Appendix C further examines reward hierarchy and complementarity through response analysis and joint fine-tuning, showing that a few representative objectives can support broad improvements across biological criteria, while the full reward set achieves a more favorable overall trade-off.

5 Conclusion

We introduced CellRFT to address the mismatch between surrogate training objectives and biological evaluation by using evaluation feedback for reinforcement fine-tuning. Our analysis shows that biological rewards can reinforce or conflict with one another, and that complementary objectives can improve criteria beyond those directly optimized. Comprehensive experiments demonstrate CellRFT's applicability across different pretrained models and effectiveness in improving perturbation prediction.

The trade-offs exposed by reward optimization show why a gain in one metric need not mean a better biological prediction. Consensus on which metrics reliably establish such improvement remains incomplete, as illustrated by the revised evaluation criteria of the 2025 (Roohani et al., 2025) and 2026 (Arc Virtual Cell Initiative Team & Goodarzi, 2026) Virtual Cell Challenges. We hope CellRFT will inform metric selection by revealing what optimizing each criterion improves or compromises, helping the community rethink how progress in virtual cell modeling should be measured.

AI USE STATEMENT

Generative AI tools were used for language polishing.

REFERENCES

Abhinav K Adduri, Dhruv Gautam, Beatrice Bevilacqua, Mohsen Naghipourfar, Alishba Imran, Rohan Shah, Noam Teyssier, Rishi Verma, Christopher Carpenter, Basak Eraslan, et al. Predicting cellular responses to perturbation across diverse contexts with state. *Cell*, 2026. doi: 10.1016/j.cell.2026.07.052.

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.

Arc Virtual Cell Initiative Team and Hani Goodarzi. Virtual cell challenge 2026: Benchmarking zero-shot generalization across cellular contexts. *Cell*, 2026. doi: 10.1016/j.cell.2026.08.004.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, volume 2024, pp. 4965–4987, 2024.

Charlotte Bunne, Stefan G Stark, Gabriele Gut, Jacobo Sarabia Del Castillo, Mitch Levesque, Kjong-Van Lehmann, Lucas Pelkmans, Andreas Krause, and Gunnar Rätsch. Learning single-cell perturbation responses using neural optimal transport. *Nature methods*, 20(11):1759–1768, 2023.

Charlotte Bunne, Yusuf Roohani, Yanay Rosen, Ankit Gupta, Xikun Zhang, Marcel Roed, Theo Alexandrov, Mohammed AlQuraishi, Patricia Brennan, Daniel B Burkhardt, et al. How to build the virtual cell with artificial intelligence: Priorities and opportunities. *Cell*, 187(25):7045–7063, 2024.

Ruisheng Cao, Mouxiang Chen, Jiawei Chen, Zeyu Cui, Yunlong Feng, Binyuan Hui, Yuheng Jing, Kaixin Li, Mingze Li, Junyang Lin, et al. Qwen3-coder-next technical report. *arXiv preprint arXiv:2603.00729*, 2026.

Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P Fulco, Livnat Jerby-Arnon, Nemanja D Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, et al. Perturb-seq: dissecting molecular circuits with scalable single-cell rna profiling of pooled genetic screens. *cell*, 167(7): 1853–1866, 2016.

Ninghan Fan, Qi Liu, Xunuo Zhu, Yukai Sun, Luyuan Chen, Xuheng Zhou, Yuetian Du, Ming Kong, Xiaojun Zhu, Jie Liu, Zhan Zhou, and Qiang Zhu. D^2R^2: Discrete diffusion with regulation reinforcement for single-cell perturbation prediction. *arXiv preprint arXiv:2608.15288*, 2026. URL https://arxiv.org/abs/2608.15288.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Siyu He, Yuefei Zhu, Daniel Naveed Tavakol, Haotian Ye, Yeh-Hsing Lao, Zixian Zhu, Cong Xu, Shradha Chauhan, Guy Garty, Raju Tomer, et al. Squidiff: predicting cellular development and responses to perturbations using a diffusion model. *Nature methods*, 23(1):65–77, 2026.

Eric Kernfeld, Yunxiao Yang, Joshua S Weinstock, Alexis Battle, and Patrick Cahan. A comparison of computational methods for expression forecasting. *Genome biology*, 26(1):388, 2025.

Dominik Klein, Jonas Simon Fleck, Daniil Bobrovskiy, Lea Zimmermann, Sören Becker, Alessandro Palma, Leander Dony, Alejandro Tejada-Lapuerta, Guillaume Huguet, Hsiu-Chuan Lin, et al. Cellflow enables generative single-cell phenotype modeling with flow matching. *bioRxiv*, pp. 2025–04, 2025.

Jon M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *Advances in neural information processing systems*, 38:40783–40818, 2025.

Mohammad Lotfollahi, F Alexander Wolf, and Fabian J Theis. scgen predicts single-cell perturbation responses. *Nature methods*, 16(8):715–721, 2019.

Mohammad Lotfollahi, Anna Klimovskaia Susmelj, Carlo De Donno, Leon Hetzel, Yuge Ji, Ignacio L Ibarra, Sanjay R Srivatsan, Mohsen Naghipourfar, Riza M Daza, Beth Martin, et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular systems biology*, 19(6):MSB202211517, 2023.

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*, pp. 1–10, 2025.

Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics*, 9(1):48, 2017.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Yusuf Roohani, Kexin Huang, and Jure Leskovec. Predicting transcriptional outcomes of novel multigene perturbations with gears. *Nature biotechnology*, 42(6):927–935, 2024.

Yusuf H Roohani, Tony J Hua, Po-Yuan Tung, Lexi R Bounds, Feiqiao B Yu, Alexander Dobin, Noam Teyssier, Abhinav Adduri, Alden Woodrow, Brian S Plosky, et al. Virtual cell challenge: Toward a turing test for the virtual cell. *Cell*, 188(13):3370–3374, 2025.

Saksham Sahai Srivastava and Vaneet Aggarwal. A technical survey of reinforcement learning techniques for large language models. *ACM Transactions on Intelligent Systems and Technology*, 2025.

Kyle Swanson, Gary Liu, Denise B Catacutan, Stewart McLellan, Autumn Arnold, Megan M Tu, Eric D Brown, James Zou, and Jonathan M Stokes. Synthemol-rl: a flexible reinforcement learning framework for designing easily synthesizable antibiotics. *Molecular Systems Biology*, 22(6): 833–867, 2026.

Dongxia Wu, Mingyu Li, Yuhui Zhang, Anurendra Kumar, Emma Lundberg, Serena Yeung-Levy, and Emily B Fox. Perturbcellrl: Verifier-guided reinforcement learning for single-cell perturbation prediction. *arXiv preprint arXiv:2606.27752*, 2026.

Yan Wu, Esther Wershof, Sebastian Schmon, Marcel Nassar, Błażej Osiński, Ridvan Eksi, Zichao Yan, Rory Stark, Kun Zhang, and Thore Graepel. Perturbench: Benchmarking machine learning models for cellular perturbation analysis. *Advances in Neural Information Processing Systems*, 38, 2025.

Xinyu Yuan, Xixian Liu, Ya Shi Zhang, Zuobai Zhang, Hongyu Guo, and Jian Tang. PerturbDiff: Functional diffusion for single-cell perturbation modeling. In *International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=yBAqonxCdp.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. In *International Conference on Learning Representations*, volume 2026, pp. 134129–134150, 2026.

Hongxu Zhu, Amir Asiaee, Leila Azinfar, Jun Li, Han Liang, Ehsan Irajizad, Kim-Anh Do, and James P Long. Auprc: a metric for evaluating the performance of in-silico perturbation methods in identifying differentially expressed genes. *Briefings in bioinformatics*, 26(5):bbaf426, 2025.

A BIOLOGICAL EVALUATION METRICS

Table 2 expands the overview in Section 3.1 by summarizing the 14 biological evaluation metrics, their evaluation focus, abbreviations, ranges, and preferred directions.

Table 2: Biological evaluation metrics organized by evaluation perspective. Arrows indicate the preferred direction; ranges assume nondegenerate inputs.

Perspective	Metric name	Abbrev.	Evaluation focus	Range
Global response fidelity	Mean squared error	MSE	Squared error in average gene expression	$[0, \infty) \downarrow$
	Mean absolute error	MAE	Absolute error in average gene expression	$[0, \infty) \downarrow$
	Pearson delta correlation	PDCorr	Alignment of predicted and observed expression changes relative to controls	$[-1, 1] \uparrow$
Differential-expression fidelity	Differential-expression overlap	DEOver	Overlap between predicted and observed DE genes	$[0, 1] \uparrow$
	Differential-expression precision	DEPrec	Agreement of predicted DE genes with observed DE genes	$[0, 1] \uparrow$
	Significant-DE-gene recall	DESigRec	Recovery of observed significant DE genes	$[0, 1] \uparrow$
	Direction agreement	DirAgr	Agreement in up- and downregulation of observed DE genes	$[0, 1] \uparrow$
	Log-fold-change Spearman correlation	LFCSpear	Rank agreement of predicted and observed log-fold changes in DE genes	$[-1, 1] \uparrow$
	Area under the receiver operating characteristic curve	AUROC	Discrimination between DE and non-DE genes	$[0, 1] \uparrow$
	Area under the precision–recall curve	AUPRC	Precision–recall performance in identifying observed DE genes	$[0, 1] \uparrow$
	Effect size correlation	ES	Rank agreement of predicted and observed DE-gene counts across perturbations	$[-1, 1] \uparrow$
Perturbation discrimination	Perturbation discrimination score	PDS_{L1}	Matching predictions to the correct perturbation using L1 distance	$[1/L, 1] \uparrow$
	Perturbation discrimination score	PDS_{L2}	Matching predictions to the correct perturbation using L2 distance	$[1/L, 1] \uparrow$
	Perturbation discrimination score	$\text{PDS}_{\cos}$	Matching predictions to the correct perturbation using cosine distance	$[1/L, 1] \uparrow$

DE: differential expression; $L = |\mathcal{P}|$: number of reference perturbations. Cell-Eval's significant-DE-gene recall is $\text{DESigRec}_\tau = |\widehat{\mathcal{D}}_\tau \cap \mathcal{D}_\tau|/|\mathcal{D}_\tau|$, where $\mathcal{D}_\tau$ and $\widehat{\mathcal{D}}_\tau$ are the observed and predicted significant DE-gene sets, respectively, identified against controls (Adduri et al., 2026).

B CELLRFT TRAINING PROCEDURE

Algorithm 1 summarizes the training procedure described in Section 3.2. It combines population-level sampling, hierarchical reward aggregation, and policy updates using shared population advantages.

Algorithm 1 CellRFT Training

Input: Pretrained model p_θ; training set of $(X_{\text{ctrl}}, X_{\text{pert}}, c, \tau)$ tuples; reward functions r_j
Output: Fine-tuned model p_θ

```
for iteration e = 1, ..., E do
    θ_old ← θ
    Sample a minibatch of (X_ctrl, X_pert, c, τ) tuples
    for all (X_ctrl, X_pert, c, τ) in the minibatch do
        Sample G sets of B cells and their trajectories from p_θold (Eq. (4))
        Compute hierarchically aggregated rewards R^(g) (Eq. (5))
        Compute group-relative advantages A^(g) (Eq. (6))
    end for
    Update θ using shared population advantages by maximizing Eq. (7)
end for
return p_θ
```

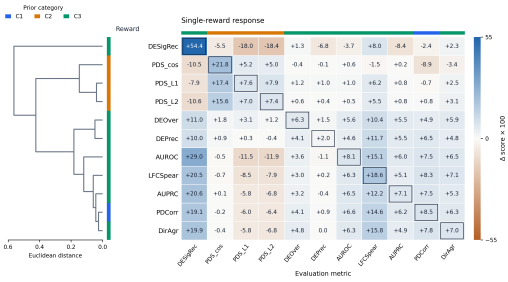
C Reward Hierarchy and Complementarity

Although the single-reward and category-combination ablations reveal cross-metric gains and trade-offs, they leave unresolved how these effects are organized within and across the predefined evaluation categories in Table 2. This structure matters for understanding how CellRFT jointly improves multiple aspects of biological fidelity: rewards from different categories may produce overlapping effects, whereas rewards within the same category may contribute differently to shared improvements.

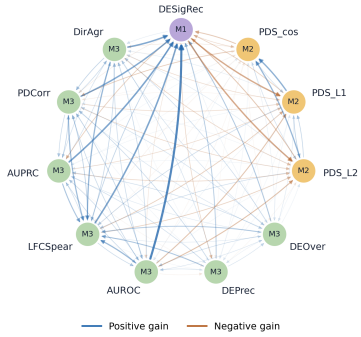
To this end, we first represent each single-reward variant by its gains over the pretrained model across all 11 metrics and cluster these raw response vectors using average linkage and Euclidean distances (Figure 5a). The resulting hierarchy partly recovers the predefined evaluation categories while revealing cross-category similarities in optimization effects. Specifically, the three PDS variants form a common branch, consistent with their shared perturbation-discrimination task, whereas DESigRec separates from the other differential-expression metrics. More notably, PDCorr and DirAgr form the closest pair despite belonging to different evaluation categories: both produce similar declines in perturbation-discrimination metrics and similar gains across the remaining metrics.

We then examine how rewards support one another through a directed network, where an edge from i to j records the gain in metric j after optimizing reward i (Figure 5b). For network visualization and within-group analysis, we use a three-group cut, supported by the highest mean silhouette score (0.624) among $k = 2, \ldots, 10$ and the largest jump in successive merge distances. We apply Hyperlink-Induced Topic Search (HITS) (Kleinberg, 1999), a network-ranking method, to positive, non-self gains within each non-singleton group. Hub scores characterize rewards providing support, while authority scores characterize metrics receiving it. This analysis reveals that similar response profiles can conceal distinct optimization roles. Specifically, PDS_L1 improves PDS_cos more strongly than the reverse, with a high hub score for PDS_L1 and a high authority score for PDS_cos (Figure 5c). Meanwhile, DEPrec receives limited within-group support despite clustering closely with DEOver: similarity in the effects a reward produces does not imply strong support for its own metric from other rewards. DirAgr also emerges as a leading hub candidate in the larger group. These roles suggest considering both rewards that improve other metrics and metrics that receive limited improvement from optimizing other rewards.

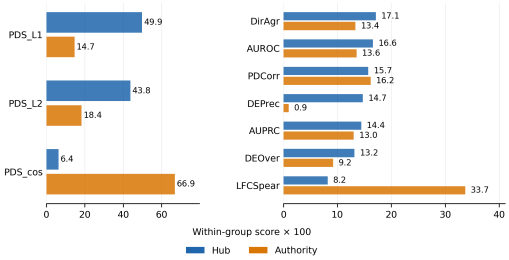
We select the metrics with the highest hub scores in the two non-singleton groups (PDS_L1 and DirAgr), retain DESigRec from the singleton group, and add DEPrec to supplement limited precision support. We jointly fine-tune using hierarchical averaging (Eq. (5)) over the selected response groups. Joint training reveals both broad indirect improvements and a recall-driven imbalance (Figure 5d). The four-metric subset exhibits a pattern consistent with reward hacking: recall rises from 0.382 to 0.833, accompanied by precision and PDCorr falling below the pretrained baseline. Removing DESigRec and reaveraging the retained groups yields a recall-excluded variant that improves 10 of 11 evaluation metrics using only three direct rewards. The representative subset demonstrates that a few direct objectives can drive improvements across multiple biological criteria. Full-reward CellRFT further achieves a more favorable balance across these criteria. These findings show that



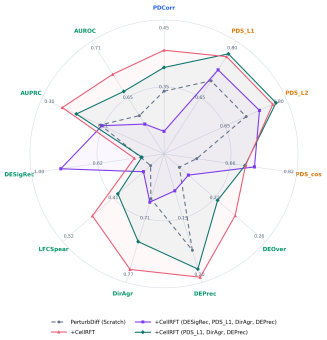
(a) Hierarchical structure of reward responses.



(b) Community structure of metric interactions.



(c) Influence and support within communities.



(d) Fine-tuning with reward subsets.

Figure 5: **Hierarchical structure and complementarity of rewards.** **(a)** Average-linkage clustering of single-reward response profiles using Euclidean distances reveals similarities across prior evaluation categories, with PDCorr and DirAgr forming the closest pair. **(b)** A directed network of cross-metric gains, colored by the three communities identified from the hierarchy, reveals asymmetric support and trade-offs within and across communities. **(c)** Within-community HITS analysis of positive gains distinguishes influential rewards such as PDS_L1 from supported metrics such as PDS_cos, while highlighting limited indirect support for DEPrec. **(d)** Joint fine-tuning with three core rewards improves 10 of 11 metrics over Scratch, showing that a compact subset can yield broad gains, while CellRFT with all 11 rewards provides a superior overall trade-off.

distinct evaluation criteria need not each require a separate training objective, providing an empirical basis for selecting rewards through their ability to support broader improvements.