Calibrated Reinforcement Learning for LLM-Guided Perturbation Screen Design

Anton Thieme · thiemea@stanford.edu
Stanford University, CS224R Spring 2026

Extended Abstract

Predicting whether knocking down a gene will differentially affect the expression of a target gene is a key reasoning task in functional genomics. We frame this as a natural-language prediction problem over the PerturbQA benchmark and train a calibrated predictor via reinforcement learning on a distillation-initialized 8B-parameter language model.

Our pipeline proceeds in three stages. First, we distill 952 successful reasoning traces from Claude Sonnet into a Qwen3-8B LoRA via supervised fine-tuning (SFT), which brings format compliance from 0% to 97% and provides an initial policy for RL. Second, we apply Group Relative Policy Optimization (GRPO) with an RLCR-style reward that combines a correctness term with a strictly proper Brier score, explicitly training for both accuracy and calibration. We diagnose and resolve the flat-reward problem in naïve GRPO—where $\geq$82% of training steps produce zero intra-group variance—via a learnability filter and appropriate loss configuration. Third, we implement calibration-guided test-time compute (TTC) that selectively allocates extra inference budget to low-confidence predictions using budget forcing and self-consistency.

On the PerturbQA validation set (three cell types, n=1,146 items), our best RLCR checkpoint improves binary F1 from 0.498 (Claude Sonnet 4.5) and 0.580 (SFT) to **0.749**, while reducing expected calibration error from 0.097 to **0.054**. A controlled six-arm hyperparameter sweep isolates the contributions of KL regularization (prevents reward hacking and F1 collapse below SFT), moderate off-policy reuse (`num_iterations`=2 outperforms on-policy by +0.07 F1), and data composition (40% positive rate balances learning signal). Calibration-guided selective TTC matches uniform self-consistency (K=8) at 6$\times$ lower token cost. On RPE1, a held-out epithelial cell type unseen during training, the policy maintains 0.745 F1, demonstrating cross-cell-type generalization from hematopoietic/hepatic training lineages.

Key contributions: (1) A complete SFT$\rightarrow$RLCR-GRPO pipeline for calibrated scientific prediction with an 8B LM; (2) diagnosis and resolution of the zero-variance gradient problem in GRPO for classification; (3) demonstration that calibration-aware rewards improve both accuracy and calibration; (4) calibration-guided selective test-time compute that is Pareto-efficient in accuracy vs. cost.

1 Introduction

CRISPR interference (CRISPRi) perturbation screens are a central tool in functional genomics: by knocking down individual genes and measuring transcriptomic changes, biologists map causal gene regulatory networks [Wu et al., 2025]. While doing these screens across the whole genome and reading out every RNA molecule in single cells is possible, it is extremely expensive and limits the potential biological insights due to the small number of readout technologies combinable with screens at this scale. Computational predictors that can accurately estimate whether a perturbation will differentially affect a target's expression, and how confident that estimate is, could dramatically reduce experimental cost by prioritizing the most informative experiments.

Recent work has shown that large language models (LLMs) can reason about biological perturbation effects when provided with appropriate context [Wu et al., 2025]. However, using an LLM as a scientific predictor introduces two challenges that standard next-token training does not address. First, the model must produce well-calibrated confidence scores: in an iterative screening setting, overconfident wrong predictions waste expensive experimental budget, while underconfident correct predictions leave discoveries on the table. Second, the model must learn from a training signal that is structurally different from language modeling—the reward is a sparse binary correctness signal (differentially expressed or not).

A foundational experiment motivates our approach: we find that enriching prompts with knowledge-graph-extracted biological context (pathway membership, expression correlations, disease annotations) improves Claude Sonnet's F1 from 0.40 to 0.48—but only when the model is allowed to reason step-by-step before answering (Appendix, Figure 6). Without reasoning, the enriched context provides no benefit over simple prompts. This suggests that learning *how* to reason over structured biological evidence is the key bottleneck, motivating an RL approach that directly rewards correct, calibrated predictions.

We address both challenges through reinforcement learning with calibration rewards (RLCR). Starting from a supervised fine-tuning (SFT) warm-start that distills reasoning traces from Claude Sonnet, we train with GRPO using a composite reward that combines binary correctness with a strictly proper Brier scoring rule. This explicitly incentivizes the model to output calibrated confidence estimates alongside its predictions.

Our main contributions are:

1. A complete SFT→RLCR-GRPO training pipeline for calibrated LLM-based scientific prediction, achieving 0.749 F1 and 0.054 ECE on PerturbQA (vs. 0.498 F1, 0.097 ECE for Claude Sonnet 4.5).
2. Diagnosis and resolution of the zero-variance gradient problem in GRPO for classification tasks, where decisive policies produce degenerate advantage estimates.
3. A controlled hyperparameter sweep isolating the effects of KL regularization, off-policy reuse, and data composition on training stability and final performance.
4. Calibration-guided selective test-time compute that matches uniform sampling at 6× lower cost.
5. Cross-cell-type generalization to a held-out epithelial lineage (RPE1).

2 Related Work

Perturbation effect prediction. GEARS [Roohani et al., 2024] predicts post-perturbation expression profiles using graph neural networks over gene regulatory networks. PerturbQA [Wu et al.,

2025] reframes perturbation prediction as a natural-language QA task and shows that LLM-based predictors can outperform models trained directly on biological perturbation data.

LLM-based perturbation prediction. Recent work has explored several axes for improving LLM performance on perturbation tasks. SynthPert [Phillips et al., 2025] demonstrates that supervised fine-tuning on synthetic reasoning traces of hypothetical perturbation mechanisms improves prediction quality. AROMA [Wang et al., 2026] shows that augmenting LLM reasoning with biological priors from knowledge graphs improves perturbation effect prediction. Our work combines both directions, distilling reasoning traces via SFT and leveraging knowledge-graph-extracted context in prompts, and adds RL training with calibration rewards to further improve accuracy and uncertainty quantification.

Verbalized confidence and calibration. Lin et al. [2022] show that LLMs can be fine-tuned to express calibrated uncertainty in natural language. LACIE [Stengel-Eskin et al., 2024] demonstrates that RL fine-tuning improves confidence calibration in LLMs by training with listener-aware feedback. RLCR [Damani et al., 2025] extends this line by using rewards based on strictly proper scoring rules (Brier scores) to jointly optimize accuracy and calibrated confidence estimation. We adapt the RLCR framework to a biological prediction task and identify failure modes specific to classification (the zero-variance gradient problem) that require task-specific solutions.

RL for LLM reasoning. DeepSeek-R1 [DeepSeek-AI, 2025] demonstrated that GRPO can train reasoning capabilities in LLMs without supervised data. Yao et al. [2026] analyze GRPO's off-policy dynamics, showing that the algorithm is implicitly off-policy and that wider clipping can accelerate learning—consistent with our finding that moderate off-policy reuse (`num_iterations`=2) outperforms strictly on-policy training.

Test-time compute scaling. Budget forcing and self-consistency are standard approaches for scaling inference compute [Wang et al., 2023]. Muennighoff et al. [2025] show that appending a continuation cue ("Wait,") to force extended reasoning improves accuracy on hard problems. We show that calibrated confidence scores enable selective allocation of such test-time compute that is substantially more cost-efficient than uniform scaling.

3 Background

3.1 Problem setting

We consider the PerturbQA binary prediction task [Wu et al., 2025]: given a perturber gene X knocked down via CRISPRi in cell type C, predict whether a target gene Y is differentially expressed (DE) and provide a confidence score. Each input prompt contains the gene pair (X, Y), the cell type description, known functional annotations, and lists of previously observed hits and non-hits for target Y (simulating information available in an iterative screen). The model outputs a structured JSON prediction with a binary DE status and a scalar confidence $c \in [0, 1]$.

The training set spans three cell types (HepG2, Jurkat, K562) with 208,432 (C, X, Y) items of which less than 10,000 were used here. Evaluation uses two held-out sets stratified by perturber (no perturber appears in both train and val): a validation set (n=1,146) for the main comparisons, and a separate TTC validation set (n=2,100) for test-time compute experiments.

3.2 Group Relative Policy Optimization

GRPO [DeepSeek-AI, 2025] generates K rollouts per prompt and computes advantages as within-group reward differences:

$$\hat{A}_i = \frac{r_i - \mu_{\text{group}}}{\sigma_{\text{group}} + \epsilon} \tag{1}$$

where r_i is the reward for rollout i, and $\mu_{\text{group}}, \sigma_{\text{group}}$ are the mean and standard deviation of rewards within the group. The policy is updated with a clipped surrogate objective analogous to PPO. A KL penalty $\beta \cdot D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})$ anchors the policy to the SFT reference.

3.3 RLCR reward

Following Damani et al. [2025], we define a composite reward:

$$r = r_{\text{correct}} + r_{\text{Brier}} \tag{2}$$

where $r_{\text{correct}} \in \{0, 1\}$ indicates whether the binary DE prediction matches ground truth, and $r_{\text{Brier}} = 1 - (p - y)^2$ is one minus the binary Brier score between the predicted DE probability p and the binary label $y \in \{0, 1\}$. The Brier score is a strictly proper scoring rule: it is uniquely minimized when p equals the true probability of DE, so optimizing it incentivizes calibration rather than overconfidence.

4 Method

4.1 Stage A: SFT warm-start via distillation

Base Qwen3-8B produces 0% parseable predictions on our structured output format (Figure 1a), making direct RL training infeasible. We first distill successful reasoning traces from Claude Sonnet 4.5 into the base model via LoRA SFT.

We collect 952 traces where Claude correctly predicts the DE status with parseable structured output. A 10-variant LoRA hyperparameter sweep (varying rank, learning rate, and epochs) selects the checkpoint maximizing validation F1. The resulting SFT model achieves 97% format compliance and F1 = 0.580, providing a viable starting policy for RL (Figure 1b).

4.2 Stage B: RLCR-GRPO training

The zero-variance gradient problem. Naïve GRPO on our task produces flat training reward because the advantage estimate (Eq. 1) requires within-group reward variance. When the SFT policy is decisive, all K rollouts predict the same class, the group has identical rewards and $\sigma_{\text{group}} = 0$, yielding zero gradient. On our initial 800-prompt, class-imbalanced training set, ≥82% of steps had zero correctness variance.

We resolve this with two interventions:

1. **Learnability filter.** We pre-filter the training set to retain only prompts where an earlier policy checkpoint produces mixed outcomes (1–3 of K=4 rollouts correct). This ensures every training prompt contributes nonzero gradient. The learnable set is refreshed periodically as the policy improves.
2. **Training configuration.** We use the BNPO loss variant (`loss_type=bnpo`) with unscaled rewards (`scale_rewards=False`) and a constant-with-warmup learning rate schedule, following the RLCR recipe [Damani et al., 2025].

Together, these reduce the zero-variance rate from 82% to 0% and produce steady reward improvement during training.

KL regularization. A KL penalty (β=0.04) anchoring to the SFT reference is critical for training stability. Without it (β=0), F1 degrades below the SFT baseline within 150 steps as the policy reward-hacks by exploiting shallow correlations (Section 5.3).

Off-policy reuse. We use moderate off-policy reuse (`num_iterations`=2), performing two gradient steps per batch of rollouts with a clip ratio of 0.5. This improves sample efficiency: the policy sees each expensive rollout twice, and the wider clip accommodates the resulting distribution shift (Section 5.3).

Data composition. The natural training distribution (∼15% DE-positive) provides insufficient positive-class signal. We upsample the learnable prompt pool to ∼40% positive rate, balancing the gradient between classes without introducing synthetic data.

These adjustments together result in a policy with both high F1 score and low brier score (Figure 1b,c).

4.3 Stage C: Calibration-guided test-time compute

The calibrated confidence scores from RLCR enable selective allocation of inference budget. We implement two TTC strategies:

Self-consistency [Wang et al., 2023]: sample K=8 rollouts and aggregate via confidence-weighted majority vote. This is the uniform baseline that applies extra compute to all items.

Budget forcing [Muennighoff et al., 2025]: on the initial greedy pass, flag items where predicted confidence falls below a threshold τ. For flagged items, the submitted answer is discarded but the reasoning trace is retained, a continuation cue ("Wait,") is appended, and the model resumes thinking from where it left off. This cycle repeats until the model submits a high-confidence answer or a round cap is reached. This selective approach spends extra tokens only where the model is uncertain.

Selective TTC exploits the observation that a calibrated model's confidence is informative about where extra compute will help: low-confidence items are disproportionately likely to benefit from additional reasoning.

5 Experiments

5.1 Setup

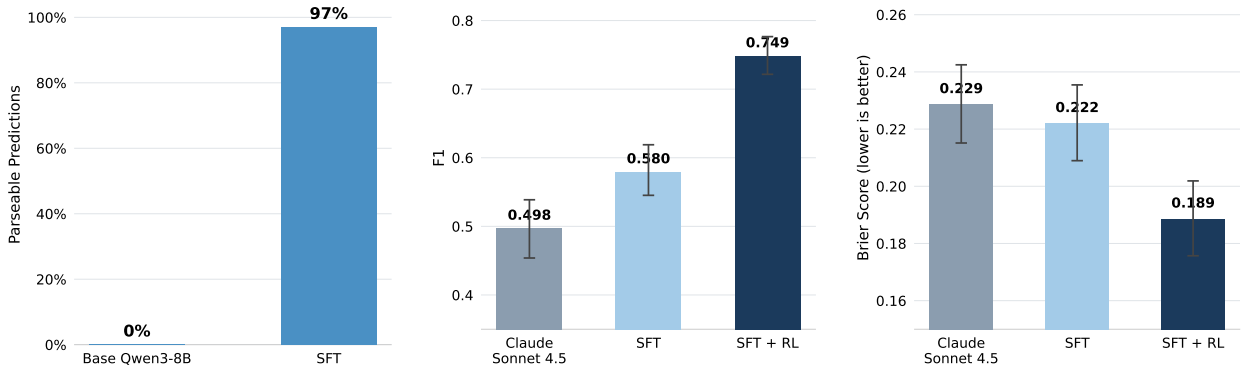
Dataset. We use PerturbQA [Wu et al., 2025] with CRISPRi perturbations across three cell types (HepG2, Jurkat, K562). Training prompts are filtered via the learnability criterion (Section 4.2) and upsampled to ∼40% positive rate. The validation set (n=1,146) is used for the main comparisons; a separate TTC validation set (n=2,100) is used for test-time compute evaluation.

Baselines. (1) *Claude Sonnet 4.5*: the teacher model used for distillation, evaluated zero-shot with the same prompt format. (2) *SFT*: the distillation-initialized Qwen3-8B LoRA before RL. (3) *RLVR*: RL with correctness-only reward (no Brier term), to isolate the calibration reward's contribution.

Metrics. Binary F1 for the DE class (F1), Brier score, and expected calibration error (ECE, 10-bin). For error bars, we report 95% bootstrap confidence intervals over test items (1,000 resamples).

Training. All RL runs use Qwen3-8B with LoRA (rank 16), K=8 rollouts per prompt, effective batch size 16 (per-device batch 1, 2 GPUs, gradient accumulation 8), and learning rate 5×10^{-6} with constant-with-warmup schedule. The training pool is pre-filtered via the learnability criterion (Section 4.2) and upsampled to ∼40% positive rate. Training runs for ∼800 gradient steps (∼12 GPU-hours on two H100 80GB GPUs).

5.2 Main results



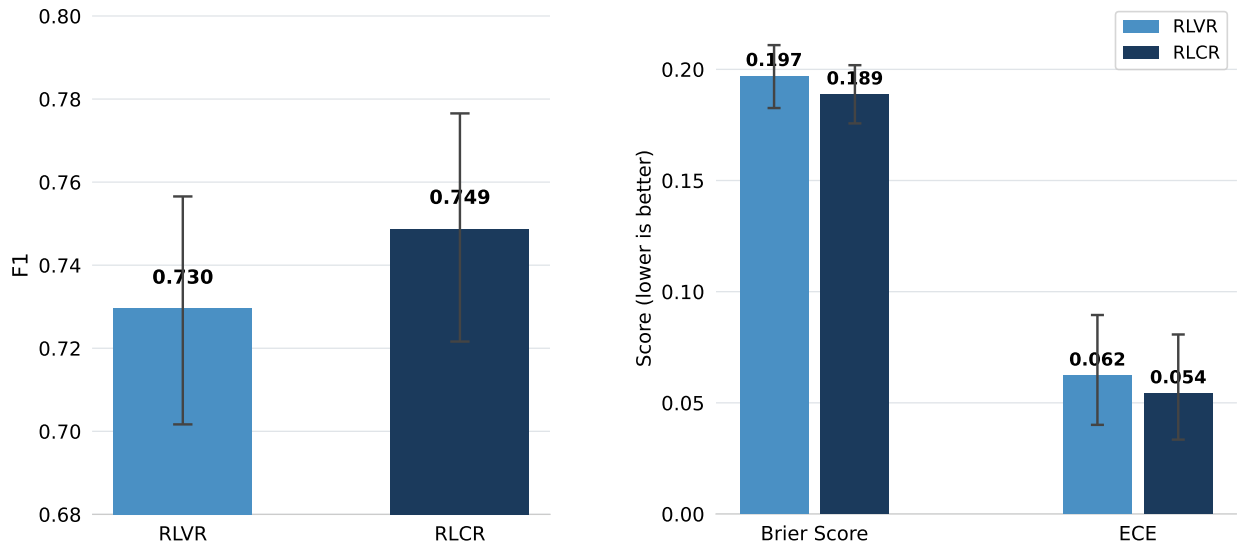
(a) SFT is necessary: base Qwen3-8B produces 0% parseable outputs.

(b) RL substantially improves F1 over both Claude and SFT baselines.

(c) RL also improves calibration: Brier score drops monotonically across the pipeline.

Figure 1: **Pipeline progression** on the validation set (n=1,146). (a) Supervised fine-tuning is a prerequisite: base Qwen3-8B cannot produce structured outputs. (b) RLCR-GRPO lifts F1 from 0.498 (Claude) and 0.580 (SFT) to 0.749. (c) Brier score improves at each stage, confirming that RL training improves calibration as well as accuracy. Error bars: 95% bootstrap CIs.

Figure 1 summarizes the pipeline progression. SFT is a strict prerequisite for RL: base Qwen3-8B produces zero parseable predictions despite identical prompting (Figure 1a). The SFT warm-start brings F1 to 0.580, showing that fine-tuning on only successful traces already results in better performance. RLCR-GRPO training then lifts F1 to 0.749 (+29% relative over SFT, +50% over Claude Sonnet 4.5's 0.498) (Figure 1b), while Brier score drops from 0.229 to 0.189 (Figure 1c).



(a) RLCR slightly outperforms correctness-only RLVR on F1.

(b) RLCR improves both Brier score and ECE over RLVR.

Figure 2: **Effect of the calibration reward** (validation set, n=1,146). Comparing RLVR (correctness-only reward) against RLCR (correctness + Brier). The Brier term does not hurt accuracy (a) while substantially improving calibration (b). Error bars: 95% bootstrap CIs.

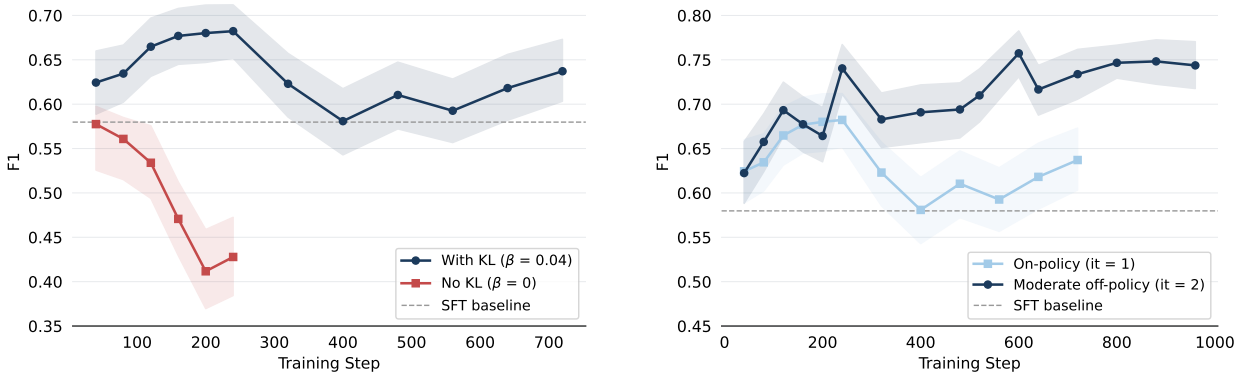
The calibration reward does not trade off accuracy for calibration. Comparing RLCR (correctness + Brier) against RLVR (correctness only) in Figure 2, RLCR achieves 0.749 F1 vs. RLVR's 0.730. The CIs overlap, but the Brier term at minimum does not hurt accuracy while slightly improving calibration (ECE: 0.062 → 0.054, Brier: 0.197 → 0.189).

Table 1: **Summary of main results** on the validation set (n=1,146). Best values in bold; ↓ = lower is better. RLVR uses correctness-only reward (no Brier term); RLCR adds the Brier calibration reward. 95% bootstrap CIs: F1 ±0.03, Brier ±0.01, ECE ±0.02.

Model	F1 ↑	Brier ↓	ECE ↓
Claude Sonnet 4.5	0.498	0.229	0.097
SFT (Qwen3-8B LoRA)	0.580	0.222	0.083
RLVR (correctness only)	0.730	0.197	0.062
RLCR (ours)	**0.749**	**0.189**	**0.054**

5.3 Training ablations

We conduct a controlled hyperparameter sweep from a common SFT initialization to isolate the contributions of individual design choices.



(a) KL regularization (β=0.04) prevents F1 collapse. Without KL, the policy reward-hacks and drops below SFT within 150 steps.

(b) Moderate off-policy reuse (`it`=2) outperforms strictly on-policy (`it`=1), reaching higher peak F1 with less variance.

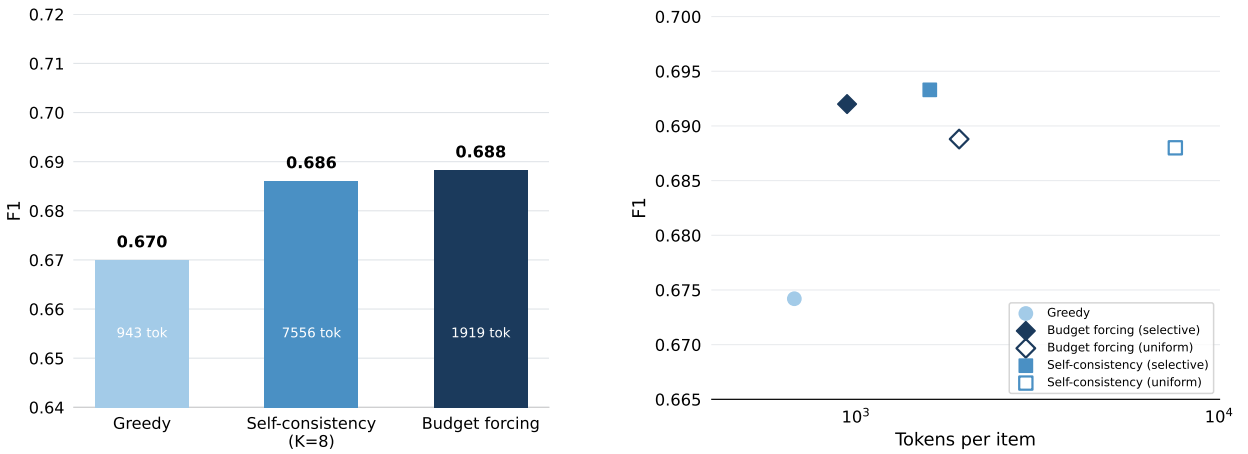
Figure 3: **Training stability ablations** on the validation set (n=1,146). Both panels show F1 across training steps. Shaded bands: 95% bootstrap CIs. Dashed line: SFT baseline (F1 = 0.580).

KL regularization is necessary. Figure 3a compares training with β=0.04 against β=0 (no KL penalty). Without KL anchoring, F1 peaks briefly at step ∼60 before collapsing to 0.43 by step 240—well below the SFT baseline of 0.58. The KL-regularized run maintains F1 above SFT throughout and reaches its peak of 0.68 around step 200. Inspection of the β=0 policy reveals reward hacking: the model learns to always predict the majority class (no-DE) with high confidence, which achieves high Brier reward on the imbalanced test set but destroys F1.

Off-policy reuse improves efficiency. Figure 3b compares on-policy (`num_iterations`=1) with moderate off-policy (`num_iterations`=2, clip ratio 0.5). The off-policy variant reaches higher peak F1 (0.757 vs. 0.682) and shows less training instability at later checkpoints. Reusing each batch of rollouts for two gradient steps doubles the effective data seen per vLLM generation pass, improving sample efficiency. Yao et al. [2026] frame GRPO's off-policy reuse as implicit regularization, which is

consistent with the improved stability we observe.

5.4 Test-time compute



(a) Both TTC methods improve F1 over greedy decoding. Token cost shown inside bars.

(b) Selective TTC (filled markers) achieves comparable F1 to uniform (open markers) at much lower token cost.

Figure 4: **Calibration-guided test-time compute** on the TTC validation set (n=2,100). (a) F1 and token cost for greedy (normal), self-consistency (K=8), and budget forcing. (b) Pareto frontier: selective allocation via calibrated confidence matches uniform TTC at ∼6× lower cost.

Figure 4 evaluates TTC strategies. Both self-consistency (K=8) and budget forcing improve F1 from 0.675 (greedy) to 0.688 (Figure 4a). The key result is in Figure 4b: selective allocation, which spends extra tokens only on items where the calibrated confidence falls below a threshold, matches uniform self-consistency at ∼1,000 tokens/item vs. ∼7,500 tokens/item. Budget forcing (selective) achieves 0.692 F1 at 943 tokens/item, a 6× cost reduction relative to uniform K=8.

This efficiency gain is a direct consequence of calibrated confidence: the model's uncertainty is informative about where extra compute will help. An uncalibrated model would misallocate TTC budget to items it is confidently wrong about, gaining nothing.

5.5 Cross-cell-type generalization

Alongside the aforementioned information about functional associations, the prompt also contains a paragraph briefly naming and describing the cell type. This essentially frames the training of a single policy to predict perturbation effects in different cell types as a multi-task RL problem. To test whether the policy generalizes beyond its training cell types (HepG2, Jurkat, K562: two hematopoietic and one hepatic lineage), we evaluate on RPE1, an hTERT-immortalized retinal pigment epithelium line that is biologically distinct: non-transformed, diploid, and epithelial rather than hematopoietic or hepatic. This cell line can biologically be expected to have significantly different responses to the same perturbations.

RPE1 was fully held out during training, no RPE1 items appear in any training or validation split. We evaluate the best RLCR checkpoint, the SFT baseline, and Claude Sonnet on n=2,100 RPE1 test items at natural class rates (31.6% DE-positive).

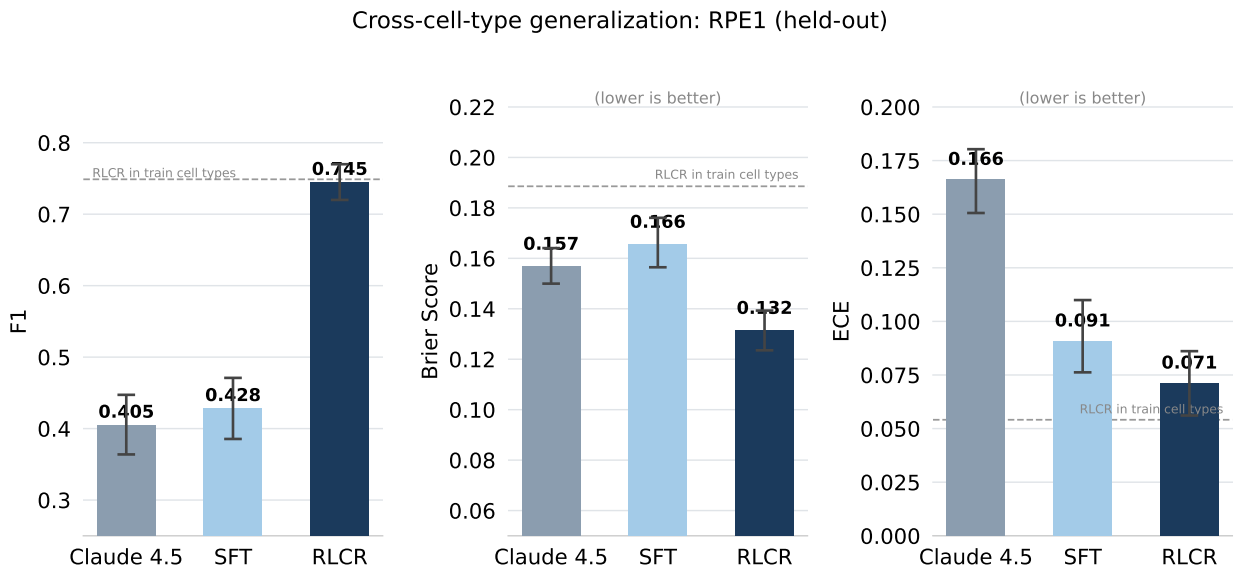


Figure 5: **Cross-cell-type generalization to RPE1** (n=2,100, 31.6% DE-positive). RLCR maintains strong F1 on a biologically distinct held-out cell type, while both SFT and Claude suffer from low DE-positive recall. Dashed lines: RLCR performance on the training-cell-type validation set for reference. Error bars: 95% bootstrap CIs.

Figure 5 shows the results. RLCR achieves 0.745 F1 on RPE1, compared to 0.428 for SFT and 0.405 for Claude Sonnet. The low performance of Claude on this cell type (F1 = 0.405, recall = 0.262) underscores RPE1's difficulty: a strong frontier model fails to identify most DE-positive pairs. RLCR restores DE-positive recall to 0.762, suggesting that RL training enables the model to learn generalizable reasoning over the knowledge-graph-extracted biological context, pathway membership, expression correlation, shared disease annotations, and to learn to combine that with the cell type information in a way that facilitates transfer of performance across cell-type lineages. Calibration also transfers: RLCR achieves 0.132 Brier and 0.071 ECE on RPE1, substantially outperforming SFT (0.166 Brier, 0.091 ECE) on the same held-out cell type.

6 Discussion

The zero-variance problem is structural, not hyperparametric. The flat-reward failure in naïve GRPO is not a matter of learning rate tuning. It stems from the interaction between GRPO's relative advantage formulation and the discrete nature of classification. When a policy is decisive (all rollouts agree), the reward variance is zero regardless of whether the prediction is correct. This contrasts with generation tasks where token-level variation naturally produces reward diversity. Our learnability filter resolves this by ensuring every training prompt has nonzero gradient, but the fundamental tension between GRPO's design and classification tasks warrants further study.

Calibration as an enabler. The Brier reward does not hurt accuracy (F1: 0.730 → 0.749, within bootstrap CI) while slightly improving calibration (ECE: 0.062 → 0.054). We hypothesize this is because the Brier gradient provides learning signal even when the binary prediction is correct, the model still improves by sharpening its confidence on correct predictions. Beyond accuracy, calibration enables downstream applications: selective TTC is 6× cheaper with calibrated confidence. It could also become useful when experiments are actually based on the policies predictions and could be an interesting direction to explore to facilitate iterative perturbation screens by providing a notion of

uncertainty in the model and with that a way to follow an exploration vs. exploitation approach.

Cross-cell-type transfer. The RLCR policy maintains 0.745 F1 on RPE1, a held-out epithelial cell type biologically distant from the training lineages (hematopoietic and hepatic), on which even Claude Sonnet achieves only 0.405 F1. This suggests that RL training enables generalizable reasoning over knowledge-graph-extracted biological context, not cell-type-specific shortcuts, and that the skills learned during training transfer to genuinely novel biological settings. This is particularly useful for applications in a wet-lab setting.

Limitations. Our evaluation is limited to four cell types (three training, one held-out). The model's reasoning quality depends on the biological context assembled in the prompt; prompts for poorly annotated genes may produce less reliable predictions.

7 Conclusion

We presented a complete pipeline for training a calibrated LLM-based predictor of perturbation effects via RLCR-GRPO. Starting from distillation-initialized SFT, reinforcement learning with a Brier-augmented reward lifts F1 by 50% over Claude Sonnet while reducing calibration error by 44%. The calibrated confidence scores enable efficient test-time compute allocation. A controlled hyperparameter sweep identifies KL regularization, moderate off-policy reuse, and balanced data composition as key ingredients for stable RLCR training. The policy generalizes to RPE1, a held-out epithelial cell type biologically distinct from the training lineages. These results suggest that calibration-aware RL is a promising approach for scientific prediction tasks where both accuracy and uncertainty quantification matter.

AI Tools Disclosure

Claude Code was used in a highly supervised manner for boilerplate implementations, code review, debugging, and formatting throughout the project. Claude Sonnet 4.5 served as the teacher model for SFT distillation.

References

Mehul Damani, Daniel Shenfeld, Yash Deshpande, and Dhruv Mahajan. Beyond binary rewards: Training LMs to reason about their uncertainty, 2025. URL `https://arxiv.org/abs/2507.16806`.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025. URL `https://arxiv.org/abs/2501.12948`.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research (TMLR)*, 2022. URL `https://openreview.net/forum?id=8s8K2UZGTZ`.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xian Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling, 2025. URL `https://arxiv.org/abs/2501.19393`.

Ryan Phillips, Menghua Wu, Tommaso Biancalani, David Richmond, and Jan-Christian Huetter. SynthPert: Enhancing LLM biological reasoning via synthetic reasoning traces for cellular perturbation prediction, 2025.

Yusuf Roohani, Kexin Huang, and Jure Leskovec. Predicting transcriptional outcomes of novel multigene perturbations with GEARS. *Nature Biotechnology*, 2024. doi: 10.1038/s41587-023-01905-6. URL https://www.nature.com/articles/s41587-023-01905-6.

Elias Stengel-Eskin, Jianing Geng, Benjamin Quah, Lingpeng Li, and Benjamin Van Durme. LACIE: Listener-aware finetuning for calibration in large language models. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024.

Haonan Wang et al. AROMA: Augmented reasoning over a multimodal architecture for virtual cell genetic perturbation modeling. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2026.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Menghua Wu, Russell Littman, Jacob Levine, Lin Qiu, Tommaso Biancalani, David Richmond, and Jan-Christian Huetter. Contextualizing biological perturbation experiments through language. In *Proceedings of the Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2502.21290.

Zhihong Yao et al. Group-relative REINFORCE is secretly an off-policy algorithm: Demystifying some myths about GRPO and its friends. In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*, 2026.

A Foundational Experiment

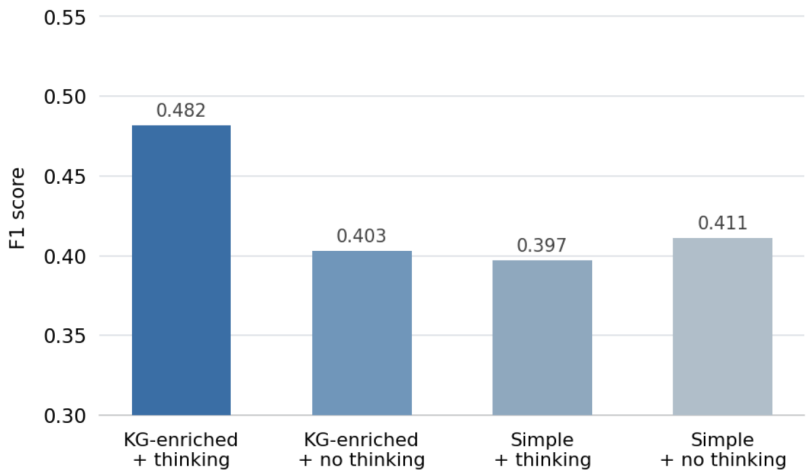


Figure 6: **Knowledge-graph context helps only when the model reasons about it.** Claude Sonnet 4.5 evaluated on a PerturbQA development set under four conditions: simple vs. KG-enriched prompts, with and without step-by-step reasoning. KG-enriched prompts improve F1 from 0.40 to 0.48, but only when reasoning is enabled; without reasoning, enriched and simple prompts perform identically. This interaction motivates training a model to reason effectively over structured biological context via RL.