FROM MEASURED BIOLOGY TO ANALYSIS DECISIONS: HOW FAR DOES POST-TRAINED KNOWLEDGE TRAVEL?

Anonymous authors
Paper under double-blind review

ABSTRACT

Closed-loop biological discovery requires models to do more than acquire domain knowledge: they must reuse prior experimental knowledge when analyzing new evidence. We study whether biological knowledge introduced through post-training can serve this role. We introduce MorphKnow, containing 15,004 experimentally measured phenotypic knowledge items derived from perturbational morphology, and MorphAnalysis, a paired suite of 967 data-analysis tasks that require combining prior biological knowledge with task-specific experimental evidence. We evaluate knowledge reuse across accessibility, inferential extension, and analytical application, and conduct controlled post-training experiments with supervised fine-tuning, distillation-based objectives, and reinforcement learning. We find a sharp separation between knowledge acquisition and downstream usability. Token-level post-training makes MorphKnow highly accessible and supports some unseen comparisons, but portability depends strongly on the operation required: it weakens for conditional lookups and held-out biological entities, and does not translate into improved scientific analysis. Supplying the correct facts does not restore analytical performance: models follow the biological knowledge but make less effective use of the current task evidence. Preserving analysis behavior during training, or explicitly prompting the model to recall the relevant fact before analysis, still does not produce reliable gains on the tasks that depend on that fact. These results suggest that parametric knowledge acquisition alone does not make biological knowledge a usable scientific prior, motivating post-training and evaluation that explicitly target knowledge–evidence integration.

1 INTRODUCTION

Modern drug discovery is increasingly envisioned as a closed-loop process in which experimental measurements do not merely provide final readouts, but continuously inform the next round of analysis, prioritization, and experimentation (Koscher et al., 2023; Boiko et al., 2023). Emerging autonomous and agentic discovery systems seek to connect stages of the design–make–test–analyze cycle, allowing evidence from one experiment to influence what is measured or tested next (Ghareeb et al., 2026; Fang et al., 2026; Huang et al., 2025). In such settings, experimental data analysis depends not only on evidence in the current dataset, but also on biological knowledge accumulated from prior experiments. Established biological knowledge from prior experiments may indicate which phenotypic relationships are likely to persist across cell types or conditions, or which readouts are biologically informative, while the current measurements determine their statistical reliability. For large language models (LLMs) intended to participate in closed-loop biological discovery, a central question is therefore not only whether they can acquire new biological knowledge, but whether that knowledge remains usable when subsequent analysis decisions depend on it.

Existing research has approached this challenge from two complementary directions. In biomedical and biological language modeling, the dominant goal has been specialization: adapting general-purpose models through domain-specific pre-training or instruction tuning to acquire specialized biological knowledge and capabilities, including biomedical question answering, biological reasoning, and cellular response prediction or virtual-cell modeling (Singhal et al., 2023; Wu et al., 2026; De Brouwer et al., 2026). More recent work has begun to ask how well these gains generalize be-

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.

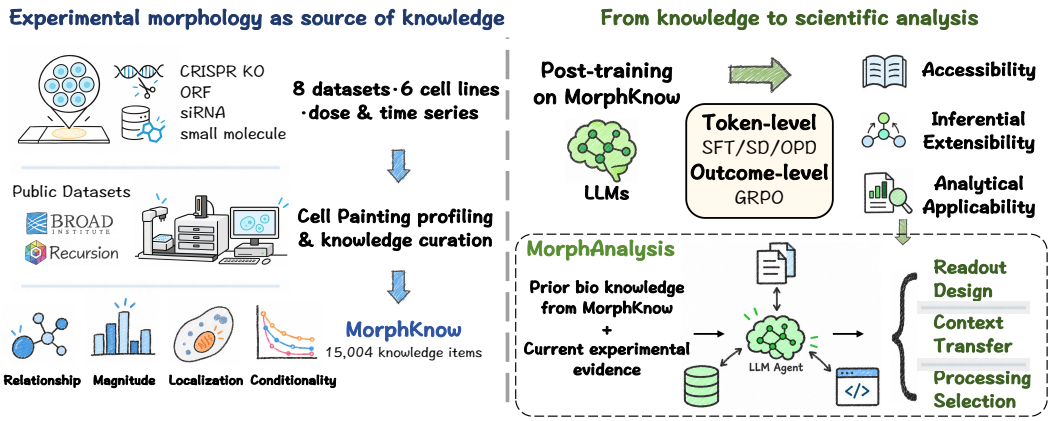

Figure 1: Overview of our framework for studying the portability of experimentally measured phenotypic knowledge, from MorphKnow post-training to knowledge portability and scientific analysis with MorphAnalysis.

yond the biological distributions seen during post-training (Fesser et al., 2026). In general-purpose language modeling, meanwhile, a growing body of work has shifted from improving reasoning benchmarks to studying the transferability of the acquired reasoning capability itself—asking, for example, whether post-training on mathematics or other reasoning-intensive domains improves performance on qualitatively different tasks (Huan et al., 2025; Hu et al., 2026; Cheng et al., 2026). These two lines emphasize different aspects of adaptation: one asks how effectively models can acquire and use specialized biological knowledge, while the other asks how far post-trained reasoning capabilities transfer across tasks and domains. Closed-loop biological discovery brings the two questions together: when a model acquires new biological knowledge through post-training, can that knowledge itself travel far enough to become useful in subsequent experimental data analysis?

To answer this question, we introduce a controlled benchmark suite for testing how far newly acquired biological knowledge can be reused beyond its original training setting. **MorphKnow** contains 15,004 experimentally measured phenotypic knowledge items, while **MorphAnalysis** provides 967 scientific analysis tasks paired with supporting biological knowledge in which previously learned knowledge must be combined with current experimental evidence. We evaluate this reuse across three progressively broader settings: **Accessibility** asks whether the same knowledge remains recoverable under unseen formulations; **Inferential Extensibility** asks whether it supports conclusions beyond those directly supervised, over both familiar and held-out entities; and **Analytical Applicability** asks whether it can cross a capability boundary into experimental data analysis. We use perturbational morphology because it provides rich, experimentally measured relationships across genes, compounds, cellular contexts, doses, and time points. Unlike existing morphology benchmarks that primarily treat images or derived profiles as model inputs or representations, MorphKnow uses morphology as a source of knowledge, converting image-derived phenotypic measurements into empirical biological relations for language-model post-training and closed-book evaluation.

We study this question through controlled post-training experiments on Qwen3.5-9B, comparing supervised fine-tuning (SFT) (Ouyang et al., 2022), self-distillation (SD) (Shenfeld et al., 2026), on-policy distillation (OPD)(Agarwal et al., 2024), and GRPO (Shao et al., 2024), with selected replication on GLM-4-9B. Beyond the three portability evaluations, we explicitly supply relevant facts, inspect intermediate training checkpoints, preserve analysis behavior through trajectory replay, and prompt models to recall facts before analysis, allowing us to separate knowledge acquisition from its downstream use. The results reveal a sharp separation between the two. Token-level supervision makes the measured biological knowledge highly accessible and supports some unseen comparisons, but reuse depends strongly on the operation required: it weakens for conditional lookups and held-out biological entities, and does not translate into improved analysis when prior knowledge must be combined with current experimental evidence. Even when the correct facts are supplied, the post-trained models remain fact-consistent while making less effective use of the task data, and this shift appears early during training.

This distinction is especially consequential for closed-loop biological discovery. Accumulated biological knowledge should act as a prior that guides analysis without overriding new experimental

Resource / Evaluation	Coverage	Size
Experimental Foundation	CRISPR, ORF, siRNA, compound and soluble-factor perturbations across 6 cell lines, with dose and time series	8 datasets, 7,100 perturbations
MorphKnow	Relationship, Magnitude, Localization, Conditionality	15,004 questions
Eval: Accessibility	Held-out formulations	1,000 questions
Eval: Inferential Extensibility	Held-out biological entities & Composed comparisons.	1,815 & 882 questions
Eval: Analytical Applicability	Processing selection, Readout design, Context transfer	967 tasks

Table 1: Benchmark overview: from perturbational morphology to knowledge portability evaluation.

evidence. Our experiments show that parametric acquisition alone does not produce this behavior: even when a fact is verifiably learned, preserving analysis behavior during training or explicitly prompting the model to recall that fact before analysis does not yield reliable gains on the tasks that depend on it. Explicit recall can make the learned fact more influential, but can also reduce reliance on the current data. The central challenge is therefore not simply to place more biological knowledge into model parameters, but to make that knowledge function as a usable scientific prior: accessible when relevant, reusable under the required operation, and appropriately integrated with new measurements. This suggests that post-training for scientific agents should be evaluated, and ultimately optimized, for knowledge–evidence integration rather than factual acquisition alone.

Our contributions are threefold.

- We study whether experimentally measured biological knowledge acquired through post-training remains usable beyond direct recall, from reformulation and inference to data-analysis decisions that combine prior knowledge with new evidence.
- We introduce MorphKnow and MorphAnalysis, a paired benchmark suite that turns perturbational morphology measurements into post-training knowledge and tests its downstream analytical use.
- Through controlled post-training and intervention studies, we show that knowledge portability is operation-dependent and that factual acquisition alone does not make biological knowledge a usable prior for scientific analysis.

2 Benchmarks and Task Design

We study knowledge portability using experimentally measured phenotypic relations derived from perturbational morphology. Across eight experimental datasets, we construct **MorphKnow**, which captures measured biological knowledge, and **MorphAnalysis**, which tests whether that knowledge can inform scientific data-analysis decisions. Together, they support three progressively broader evaluations: **Accessibility** tests whether learned knowledge remains recoverable under new formulations; **Inferential Extensibility** tests whether it supports conclusions that were not directly supervised; and **Analytical Applicability** tests whether it remains useful when the task changes from producing a fact to performing scientific analysis. Table 1 summarizes the overall design.

2.1 Experimentally Measured Phenotypic Knowledge

We build on eight perturbational morphology datasets covering CRISPR knockout, ORF overexpression, siRNA, small-molecule, and soluble-factor perturbations across six cell lines, with dose and time series in several datasets. Every MorphKnow fact is derived from these measurements rather than from literature or curated annotations. Morphology therefore serves as the source of knowledge rather than as model input: normalized Cell Painting profiles are aggregated across replicates and converted into perturbation-level relations based on profile similarity, control-referenced effect magnitude, compartment-specific effects, and their changes across experimental contexts.

The facts available from each dataset depend on its experimental design: dose- and time-dependent knowledge requires the corresponding factorial measurements, while effect magnitude requires a control-referenced scale. We establish ground truth before item construction and retain only relations that reproduce across experimental batches and separate from measurement noise. Ground-truth qualification uses per-plate normalization and feature selection without joint batch-integration across the evaluation data. Appendix B.2 gives the full qualification procedure.

2.2 MorphKnow: Four Classes of Phenotypic Knowledge

From these qualified relations, we construct MorphKnow, containing 15,004 closed-book items across four classes of phenotypic knowledge. At evaluation time, models see neither the morphological profiles nor the statistics from which the answers were derived.

We write $s_c(p, q)$ for the phenotypic similarity between perturbations p and q measured in experimental context c (cell line, dose, and time point), $m_c(p)$ for the magnitude of p's effect relative to controls, and $m_c^{(k)}(p)$ for that magnitude restricted to stain channel k (DNA, ER, RNA, AGP, or Mito). The four classes ask whether $s_c(A, B) > s_c(A, C)$ (**Relationship**), whether $m_c(A) > m_c(B)$ (**Magnitude**), whether $m_c^{(k_1)}(A) > m_c^{(k_2)}(A)$ (**Localization**), and whether a relation established in one context still holds in another, for instance across cell lines or between two doses of the same compound (**Conditionality**). These forms are instantiated by more than ten task families as pairwise comparisons, multi-way choices, rankings, and detection or potency questions, spanning genes, compounds, and soluble factors across the contexts mentioned before. Complete task definitions and class distributions are in Appendix B.3. MorphKnow provides the post-training supervision and the substrate for the two evaluations that follow.

2.3 MorphKnow Evaluations: Accessibility and Inferential Extensibility

We derive two evaluations that test reuse beyond direct supervision while varying the formulation, required inference, or biological entities in controlled ways.

Accessibility tests whether a learned fact remains accessible under an unseen formulation. We hold the biological entities, experimental context, underlying relation, and correct answer fixed, and render each query using a formulation that never appears during post-training. A drop at this level therefore indicates dependence on how the fact is expressed rather than on the underlying fact itself.

Inferential Extensibility asks whether acquired knowledge supports conclusions that were not directly supervised. We consider two generalization regimes. In the *transductive* setting, the biological entities are familiar from post-training, but the target conclusion is new. **Composed Comparisons** instantiate this setting with two task types. In *Scalar Chains*, post-training contains $q_c(A) > q_c(Z)$ and $q_c(Z) > q_c(B)$, while evaluation asks for the previously unseen comparison between A and B. In *Selector–Property Lookups*, the model first identifies $\arg\max_{p\in\{A,B\}} m_c(p)$ and then reports which of k_1 or k_2 carries more of the selected perturbation's effect; neither component alone determines the answer. In the *inductive* setting, we test entity-level OOD generalization. **Entity-Held-Out Generalization** withholds perturbation identities from MorphKnow post-training while preserving the relation type. Each evaluation item contains at least one held-out identity, testing whether relational knowledge learned on seen entities transfers to unseen biological entities.

2.4 MorphAnalysis: Analytical Applicability

The preceding evaluations remain within the activity used during knowledge training: producing a biological fact. MorphAnalysis changes the task by asking whether the same knowledge can inform a scientific data-analysis decision. It contains 967 agentic tasks, each linked to a specific MorphKnow fact, allowing us to test whether training on that particular fact influences the corresponding analysis. Each task provides a scientific scenario and a working directory; the model inspects files, runs numerical analyses, and returns a structured answer that is graded deterministically. Real experiments determine the biological entities, contexts, and knowledge labels, while the numerical evidence is generated within empirically observed ranges to control the information available from the files alone.

The benchmark contains three recurring analysis decisions. **Processing Selection** asks which processed representation best preserves a known biological relationship. **Readout Design** asks for a readout that captures an informative phenotype while satisfying task-specific quality-control constraints. **Context Transfer** asks the model to transfer an estimate from a measured source context to an unmeasured target context. In Readout Design and Context Transfer, the MorphKnow fact determines one component of the decision and the provided files determine the other, so neither source alone is sufficient. Processing Selection is intentionally simpler: the supporting fact alone determines the correct choice, providing a minimal test of whether learned biological knowledge can

enter an analysis decision. In the default evaluation, the required MorphKnow fact is not provided in the prompt and must be supplied from post-training.

3 Post-Training and Evaluation

3.1 Post-Training Methods and Configurations

To test whether knowledge portability depends on the post-training mechanism, we compare four methods that differ in how supervision is delivered: SFT, self-distillation (SD), on-policy distillation (OPD), and GRPO. Our main backbone is Qwen3.5-9B Qwen Team (2026); we additionally use GLM-4-9B Glm et al. (2024) for selected replication experiments.

They vary along two design dimensions relevant to our study: whether supervision is provided at the token or sequence level, and whether training follows reference responses or the model's own samples. SFT, SD, and OPD provide *token-level target supervision*, whereas GRPO provides *sequence-level outcome feedback*. **SFT** maximizes the likelihood of the reference answer. **SD** matches the next-token distribution of a frozen answer-conditioned teacher along the reference response, while **OPD** applies the same forward-KL objective along responses sampled from the current student. In both cases, the teacher receives the question and known answer as privileged context, while the student receives only the question. **GRPO** samples multiple responses from the current policy and optimizes group-relative advantages based on final-answer correctness, with a KL penalty to the initial policy.

Within each backbone, all methods start independently from the same initial checkpoint and draw training questions from the same MorphKnow training pool, but use method-specific training budgets. All configurations use rank-64 LoRA. SFT, SD, and OPD train for 10 epochs, whereas GRPO uses a fixed 40-step schedule. We report the final checkpoint of each pre-specified schedule without selecting checkpoints on downstream evaluations. Full optimization settings, sampling parameters, and compute are provided in Appendix C.

3.2 Training Splits and Evaluation

We use two MorphKnow post-training splits. Models evaluated on Accessibility (AX), the transductive Inferential Extensibility tasks (IE-SC and IE-SP), and Analytical Applicability (AA) are trained from the full 15,004-item MorphKnow pool. Entity-Held-Out Generalization (IE-EH) instead uses the identity-held-out split, with 13,189 post-training items and 1,815 evaluation items covering 537 held-out identities. Models trained on the full pool are not evaluated on IE-EH because its evaluation items occur in that pool; conversely, models trained on the entity-held-out split are evaluated only on IE-EH. The evaluation sets contain 1,000 AX-HF questions, 722 IE-SC questions, 160 IE-SP questions, 1,815 IE-EH questions, and 967 MorphAnalysis tasks (480 Processing Selection (AA-PS), 247 Readout Design (AA-RD), 240 Context Transfer tasks (AA-CT)). MorphAnalysis is never used for post-training, and the composed targets in IE-SC and IE-SP are absent from the post-training supervision.

For each model and question, we draw three responses and average correctness across samples; missing or invalid answers are scored as incorrect. We report accuracy for all evaluations. Because AX and IE have different answer-independent baselines, we additionally report baseline-normalized accuracy, $\mathrm{BNA} = \frac{\mathrm{Accuracy}-c}{1-c}$, where c is the best constant-answer baseline for the corresponding evaluation. For AA-RD and AA-CT, we additionally report whether the selected option is consistent with the relevant biological fact and with the task evidence. All main Qwen evaluations use the same non-thinking decoding protocol. More details are provided in Appendix C.

4 Main Results

4.1 Knowledge is acquired, but its portability is selective

Figure 2 places all evaluations on a common scale, ordered from the smallest to the largest departure from post-training supervision. It shows that token-level post-training reliably internalizes

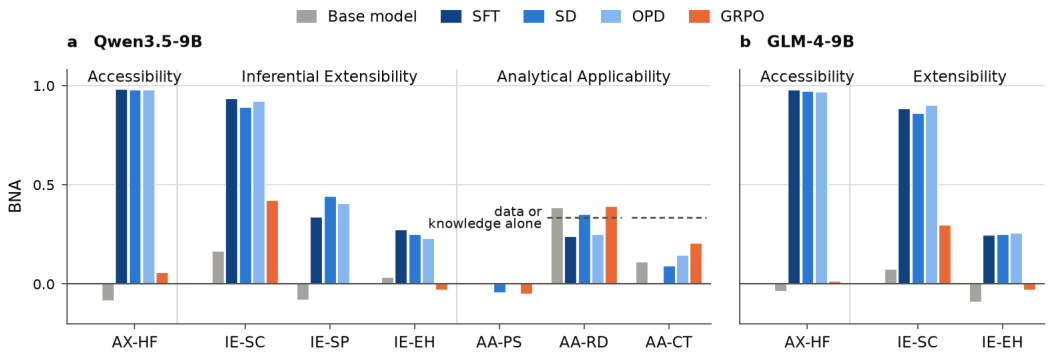

Figure 2: **Post-training makes knowledge highly accessible, but transfer to analytical application remains limited.** Baseline-normalized accuracy (BNA; 0 = best constant-answer baseline, 1 = perfect accuracy), averaged over three responses per question. Evaluations are grouped by increasing conceptual distance from post-training supervision: Accessibility, Inferential Extensibility, and Analytical Applicability. (a) Qwen3.5-9B across all seven evaluations. (b) GLM-4-9B replication on AX-HF, IE-SC, and IE-EH. IE-EH models are post-trained on the entity-held-out split. Dashed lines for AA-RD and AA-CT indicate the BNA corresponding to the 50% raw-accuracy baseline achievable using either the biological knowledge or the task data alone.

MorphKnow. On held-out formulations (AX-HF), SFT, SD, and OPD reach near-perfect performance on both backbones, indicating that the learned knowledge is not tied to the surface form used during training. This knowledge also supports new comparisons over seen entities: on Scalar Chains (IE-SC), the three methods retain high performance on both Qwen and GLM.

Portability weakens, however, when the target departs more substantially from the supervised facts. Selector–Property Lookups (IE-SP) reduce BNA to 0.34–0.44 on Qwen, while Entity-Held-Out Generalization (IE-EH) reaches only 0.23–0.27 despite remaining above the base model. Thus, post-training supports recombination of learned facts substantially better than transfer to unseen biological entities. GRPO exhibits the opposite pattern: it improves some composed reasoning, but acquires little of the underlying MorphKnow knowledge itself.

4.2 Acquiring knowledge does not make it analytically usable

The strongest discontinuity appears at Analytical Applicability. Despite near-perfect MorphKnow accessibility, none of the token-level methods improves over the base model on any MorphAnalysis family. The failure is already visible on AA-PS, where the relevant MorphKnow fact alone determines the correct choice: all models remain near chance. On AA-RD and AA-CT, which additionally require combining biological knowledge with task-specific evidence, token-level post-training likewise provides no gain and in several cases reduces performance relative to the base model. In contrast, GRPO largely preserves the base model's analytical performance despite acquiring much less MorphKnow knowledge. This sharp disconnect between knowledge acquisition and analytical reuse motivates the diagnostic analyses in next section. Full per-task results, backbone replications, and additional robustness analyses are provided in Appendix D.

5 Understanding Knowledge Portability

The main results establish a gap between knowledge acquisition and downstream reuse. We now ask what determines whether acquired knowledge remains usable beyond direct recall. Additional analyses, held-out-entity breakdowns, and training-dynamics results are provided in Appendix E.

5.1 Knowledge Portability Depends on the Required Operation

This dependence is already visible within Inferential Extensibility. Token-level methods remain strong on IE-SC (BNA 0.89–0.93 on Qwen3.5-9B), but fall substantially on IE-SP (0.34–0.44) and

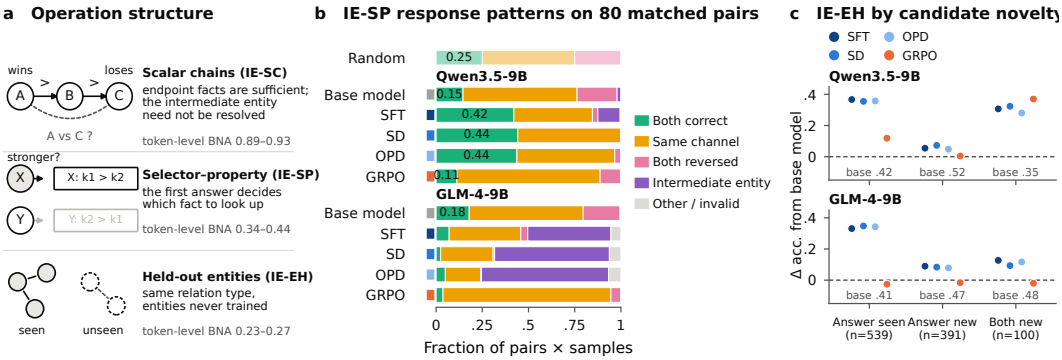

Figure 3: **(a)** Structure of the three Inferential Extensibility evaluations. IE-SC asks for an unseen endpoint comparison; because each endpoint's supporting fact already identifies it as a winner or loser, the intermediate entity need not be resolved. IE-SP instead requires a selector-dependent lookup, in which the first answer determines which property fact must be accessed. IE-EH applies familiar relation types to entities absent from post-training. Gray text reports BNA ranges for SFT, SD, and OPD on Qwen3.5-9B. **(b)** Mutually exclusive response patterns over 80 matched IE-SP pairs (80 pairs $\times$ 3 samples). Paired items reverse the selector while keeping entities and channels fixed; "same channel" indicates selector-insensitive responding, whereas "intermediate entity" indicates returning the selected perturbation instead of its requested property. The top row shows the expectation under independent random answering. **(c)** Change in IE-EH accuracy relative to the corresponding base model, stratified by candidate novelty. Labels at the zero line give raw base-model accuracy. Positive gains when both candidates are unseen show that preference for previously seen entities does not explain all improvement. IE-EH models are trained on the entity-held-out split.

IE-EH (0.23–0.27). IE-SC and IE-SP both rely on two learned facts and use short two-choice outputs; what differs is the operation required to reuse those facts (Figure 3a).

Generalization to unseen comparisons is strong. IE-SC (Inferential Extensibility–Unseen Comparison) asks the model to compare two endpoints that never appear together during post-training, so the test cannot be solved by recalling a previously seen comparison. However, each endpoint already appears in a learned fact that indicates whether it is the stronger or weaker member of that relation. The model can therefore answer the new comparison by reusing these endpoint-level signals, without explicitly resolving the intermediate entity. IE-SC thus demonstrates generalization to a new comparison, but should not be interpreted as evidence of explicit multi-hop reasoning. A simple answer-frequency heuristic is also insufficient to explain the result (Appendix E.1.1).

Conditional lookup is substantially harder. IE-SP (Inferential Extensibility–Selector-Dependent Property Lookup) requires the result of one learned relation to determine which second fact must be retrieved. Matched pairs isolate this dependency (Figure 3b): Qwen token-level methods solve both paired items in 42–44% of cases, compared with 25% under random answering, but remain far from reliable. Thus, facts that are individually accessible become substantially harder to use when one retrieval must condition the next.

Unseen entities create a second boundary. IE-EH (Inferential Extensibility–Entity Holdout) tests entity-level OOD generalization: the relation type is familiar, but the biological entities are held out from post-training. Gains are largest when the correct candidate was seen during training, revealing a strong seen-entity preference (Figure 3c). Positive gains nevertheless persist when both candidates are unseen, so this preference does not explain the full effect; because the base model is itself biased on this subset, we treat the residual as mixed evidence rather than direct proof of new knowledge.

Together, these results show that **knowledge portability depends on how learned facts must be used, rather than simply on whether they have been acquired**. The same knowledge can support an unseen comparison, yet become much less reliable when reuse requires conditional lookup or transfer to unseen entities. This motivates a stronger test in scientific analysis: what happens when the relevant fact is already available, but must be combined with new experimental evidence?

5.2 Supplying the Facts Does Not Restore Analytical Performance

Analytical Applicability provides a stronger test: the relevant biological fact must be combined with evidence from the current task. One possible explanation for the gap in Figure 2 is that the learned fact is not reliably brought into the analytical decision. We therefore supply the correct fact directly in the task context. Figure 4 shows that this is insufficient: SFT, SD, and OPD remain substantially worse than the base model despite having the fact explicitly available. Their choices are usually consistent with the supplied fact, but much less often with the task data. At the same time, these models also exhibit substantial changes in response length and tool use after post-training (Appendix E), indicating that the oracle-fact result is accompanied by a broader shift in downstream task execution. We therefore do not interpret fact availability alone as isolating a knowledge–evidence integration failure.

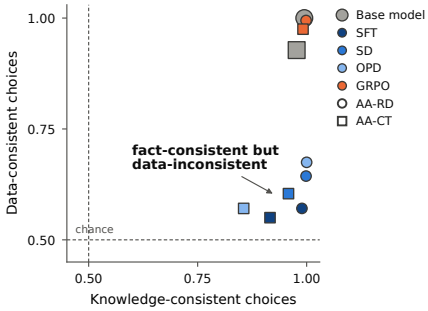

Figure 4: **Supplying the correct facts does not restore analytical performance.** Under the oracle-fact condition on AA-RD and AA-CT, SFT, SD, and OPD remain strongly fact-consistent but substantially less data-consistent than the base model; GRPO remains close to the base model.

This shift appears early during post-training. Along SFT and SD training, MorphKnow accuracy improves gradually, while data-consistent decisions on AA-CT and AA-RD change much earlier (Figure 5). By 0.5 epoch, only a small fraction of the eventual knowledge gain has been acquired, yet analytical behavior has already shifted substantially. Similar patterns across additional methods and backbones are reported in Appendix E.2.3. Thus, changes in downstream task execution emerge while knowledge acquisition is still underway, rather than only after the new facts have been strongly fit.

Together, these results show that **knowledge post-training changes not only what the model knows, but also its downstream analytical behavior**. This behavioral shift makes the raw Analytical Applicability degradation difficult to attribute to knowledge use alone. GRPO largely preserves the base model's analytical behavior, but also acquires much less MorphKnow knowledge, so we do not attribute this difference to reinforcement learning itself. This motivates a stricter test: whether newly learned facts provide a specific analytical benefit when the analysis behavior itself is preserved.

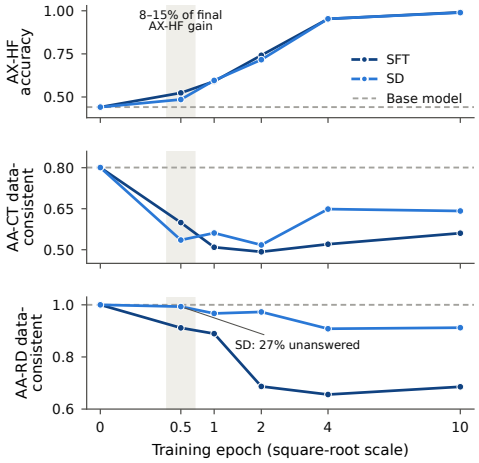

Figure 5: **Analytical behavior shifts early during knowledge post-training.** Qwen3.5-9B along SFT and SD training. MorphKnow accuracy and data-consistent analytical decisions are tracked across training checkpoints.

6 Can Learned Facts Become Usable Scientific Priors?

Section 5.2 shows that knowledge post-training can alter downstream analytical behavior, making it difficult to determine whether the lack of analytical gain reflects knowledge use itself or broader execution interference. We therefore ask a stricter question: if analytical behavior is preserved during training, does learning a particular biological fact actually improve the downstream analyses that depend on that fact?

To test this directly, we construct paired training runs. Supporting MorphKnow facts are split into two halves; within each pair, the analysis replay data are identical, while the two models are trained on complementary sets of facts. For any downstream item supported by fact f, we can therefore compare the same item under a model trained on f and a matched model for which f was held out. Their difference measures the *fact-specific downstream gain*. To reduce the execution shift identified in Section 5.2, we interleave knowledge training with analysis trajectories. We consider *generic replay*, using multi-turn data-analysis trajectories unrelated to MorphAnalysis, and *task replay*, using base-model trajectories from the MorphAnalysis task family. Full replay construction and training details are provided in Appendix F. These experiments use Qwen3.5-9B with SFT.

6.1 Preserving Analysis Behavior Does Not Restore Fact-Specific Gains

Replay substantially restores the base model's analytical execution and data-consistent behavior (Appendix F), reducing the major execution shift observed in Section 5.2. Both replay settings successfully separate trained from held-out facts: performance on trained supporting facts improves by 35–37 percentage points relative to their held-out counterparts. However, this advantage does not carry over to the corresponding analyses. Without supplying the facts at inference time, the fact-specific gain in analytical accuracy is near zero, with one-sided 95% upper bounds of +3.2 points under generic replay and +2.2 points under task replay (Figure 6, left). Task replay does improve overall AA-CT performance, but this improvement is similar whether or not the supporting fact for a given item was included in post-training. These results show that **even when analytical behavior is largely preserved, learning a specific fact does not produce a measurable advantage on the analyses that depend on it**. A model can acquire the relevant knowledge and improve at the analysis task family, while the two improvements remain largely independent.

6.2 Explicit Recall Increases Fact Influence, Not Analytical Benefit

One remaining possibility is that the facts are learned but are not naturally brought into the analytical decision. We therefore probe recall under progressively less explicit prompts. The trained-versus-held-out advantage is 24 points when the fact is directly requested, decreases to 16 points with task context and 13 points with the full task prompt, and disappears under an open-ended request (Figure 6, middle). Thus, the learned facts remain accessible under targeted prompting, but are less reliably surfaced when the model must identify the relevant knowledge on its own.

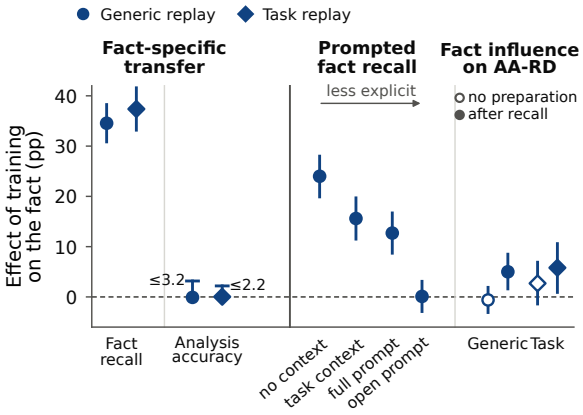

Figure 6: **Learned facts can be recalled and made influential without producing reliable fact-specific analytical gains.** Qwen3.5-9B with SFT and analysis replay. Left: training on the relevant supporting fact produces large gains in factual recall but near-zero gains in the corresponding analytical accuracy; bars denote one-sided 95% upper bounds. Right: the trained-fact advantage weakens as recall prompts become less explicit. Explicitly recalling the fact before AA-RD shifts decisions toward that fact, but does not establish a corresponding accuracy improvement.

Explicitly recalling the fact before AA-RD also shifts decisions toward fact-consistent choices, but does not produce a reliable gain in accuracy (Figure 6, right). Data-consistent choices additionally decrease, including for the base model. Thus, **making learned knowledge more salient can increase its influence without producing a reliable analytical benefit**.

Together, these experiments separate factual acquisition from downstream usefulness. The trained facts can be learned, directly recalled, and made influential in the final decision, yet none of these guarantees improved analysis. **A learned fact therefore does not automatically become a usable scientific prior: it must also be brought into the decision and appropriately balanced against the current evidence.**

7 Discussion and Conclusion

Our results show that factual acquisition does not guarantee scientific use: post-training can make new biological knowledge highly accessible without yielding reliable fact-specific gains in analyses that combine prior knowledge with new evidence. This motivates post-training and evaluation that target the downstream use of learned knowledge, rather than factual acquisition alone. Our study is limited to perturbational morphology, controlled analysis tasks, two 9B-scale backbones, and a small set of post-training recipes; some diagnostics use a single training seed, and our evidence for altered task execution is behavioral rather than mechanistic. Overall, learning, recalling, and appropriately using a scientific fact under new evidence should be treated as distinct capabilities.

AI USE STATEMENT

In this work, we used generative AI tools to propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, clean and reformat datasets, and support qualitative and thematic data analysis. We did not use generative AI tools to help develop theoretical models or conceptual frameworks, assist with translation, or interpret results. Generating synthetic datasets, formulating mathematical claims, providing critical ingredients for proving mathematical claims, and assisting in the writing of proofs were not applicable to this work. Additionally, we used generative AI tools for literature search and for editing the paper for readability. We reviewed all AI-assisted work: research hypotheses and methodological suggestions were evaluated by the authors before use; AI-assisted code and data-processing steps were checked and tested for correctness; literature-search results were verified against the original sources; and all AI-assisted edits were reviewed by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All experiments are built on publicly available model checkpoints, including Qwen3.5-9B and GLM-4-9B-0414. Complete post-training and evaluation details are provided in the appendix, including model and LoRA configurations, optimization hyperparameters, sampling and decoding settings, training schedules, random seeds, compute resources, data splits, and evaluation procedures. All reported models use the final checkpoint of a pre-specified training schedule, without checkpoint selection on downstream evaluations. MorphKnow, MorphAnalysis, and the corresponding benchmark metadata are released anonymously at `https://huggingface.co/datasets/anonymousdataset/MorphKnow`. The cell painting data used to construct the benchmarks are derived from publicly available datasets released by the Broad Institute and Recursion.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Vidit Agrawal, John Peters, Tyler Thompson, Mohammad Sanian, Chau Pham, Nikita Moshkov, Arshad Kazi, Aditya Pillai, Jack Freeman, Byunguk Kang, et al. Chammi-75: Pre-training multi-channel models with heterogeneous microscopy images. In *International Conference on Learning Representations*, volume 2026, pp. 81574–81622, 2026.

Daniil A Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023.

Srinivas Niranj Chandrasekaran, Jeanelle Ackerman, Eric Alix, D Michael Ando, John Arevalo, Melissa Bennion, Nicolas Boisseau, Adriana Borowa, Justin D Boyd, Laurent Brino, et al. Jump cell painting dataset: morphological impact of 136,000 chemical and genetic perturbations. *BioRxiv*, pp. 2023–03, 2023.

Srinivas Niranj Chandrasekaran, Beth A Cimini, Amy Goodale, Lisa Miller, Maria Kost-Alimova, Nasim Jamali, John G Doench, Briana Fritchman, Adam Skepner, Michelle Melanson, et al. Three million images and morphological profiles of cells treated with matched chemical and genetic perturbations. *Nature Methods*, 21(6):1114–1121, 2024.

Jorge Zhoujun Cheng, Shibo Hao, Tianyang Liu, Fan Zhou, Yutao Xie, Feng Yao, Yuexin Bian, Nilabjo Dey, Yonghao Zhuang, Yuheng Zha, et al. Revisiting reinforcement learning for llm reasoning from a cross-domain perspective. *Advances in Neural Information Processing Systems*, 38, 2026.

Michael F Cuccarese, Berton A Earnshaw, Katie Heiser, Ben Fogelson, Chadwick T Davis, Peter F McLean, Hannah B Gordon, Kathleen-Rose Skelly, Fiona L Weathersby, Vlad Rodic, et al. Functional immune mapping with deep-learning enabled phenomics applied to immunomodulatory and covid-19 drug discovery. *Biorxiv*, pp. 2020–08, 2020.

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.

Edward De Brouwer, Carl Edwards, Alexander Wu, Jenna Collier, Graham Heimberg, Xiner Li, Meena Subramaniam, Ehsan Hajiramezanali, David Richmond, Jan-Christian Hütter, et al. Assaybench: An assay-level virtual cell benchmark for llms and agents. *arXiv preprint arXiv:2605.10876*, 2026.

Ada Fang, Kevin Li, Ayush Noori, Lukas Fesser, and Marinka Zitnik. Closing the loop in ai-driven biomedical discovery. 2026.

Lukas Fesser, Hanlin Zhang, Michelle M Li, Eric Wang, Bryan Perozzi, Shekoofeh Azizi, Sham M Kakade, and Marinka Zitnik. How post-training shapes biological reasoning models. *arXiv preprint arXiv:2606.16517*, 2026.

Ali Essam Ghareeb, Benjamin Chang, Ludovico Mitchener, Angela Yiu, Caralyn J Szostkiewicz, Dmytro Shved, Gavin J Gyimesi, Jon M Laurent, Samantha M Wright, Muhammed T Razzak, et al. A multi-agent system for automating scientific discovery. *Nature*, pp. 1–3, 2026.

Team Glm, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*, 2024.

Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Petar Sirkovic, Artiom Myaskovsky, Grzegorz Glowaty, Felix Weissenberger, Alessio Orlandi, Dan Popovici, et al. Accelerating scientific discovery with co-scientist. *Nature*, pp. 1–3, 2026.

Emre Hayir, Lorin Crawford, and Alex X Lu. Morphohelm: A comprehensive benchmark for evaluating representations for microscopy-based morphology assays. *arXiv preprint arXiv:2605.15383*, 2026.

Chuxuan Hu, Yuxuan Zhu, Antony Kellermann, Caleb Biddulph, Suppakit Waiwitlikhit, Jason Benn, and Daniel Kang. Breaking barriers: Do reinforcement post training gains transfer to unseen domains? In *International Conference on Learning Representations*, volume 2026, pp. 23100–23123, 2026.

Maggie Huan, Yuetai Li, Tuney Zheng, Xiaoyu Xu, Seungone Kim, Minxin Du, Radha Poovendran, Graham Neubig, and Xiang Yue. Does math reasoning improve general llm capabilities? understanding transferability of llm reasoning. *arXiv preprint arXiv:2507.00432*, 2025.

Kexin Huang, Serena Zhang, Hanchen Wang, Yuanhao Qu, Yingzhou Lu, Yusuf Roohani, Ryan Li, Lin Qiu, Gavin Li, Junze Zhang, et al. Biomni: A general-purpose biomedical ai agent. *biorxiv*, 2025.

QHJAJ Huangfu and JA Julian Hall. Parallelizing the dual revised simplex method. *Mathematical programming computation*, 10(1):119–142, 2018.

Brent A Koscher, Richard B Canty, Matthew A McDonald, Kevin P Greenman, Charles J McGill, Camille L Bilodeau, Wengong Jin, Haoyang Wu, Florence H Vermeire, Brooke Jin, et al. Autonomous, multiproperty-driven molecular discovery: From predictions to measurements and back. *Science*, 382(6677):eadi1407, 2023.

Siqi Kou, Qingyuan Tian, Hanwen Xu, Zihao Zeng, and Zhijie Deng. Which data attributes stimulate math and code reasoning? an investigation via influence functions. *Advances in Neural Information Processing Systems*, 38:12302–12324, 2026.

Oren Kraus, Federico Comitani, John Urbanik, Kian Kenyon-Dean, Lakshmanan Arumugam, Saber Saberian, Cas Wognum, Safiye Celik, and Imran S Haque. Rxrx3-core: benchmarking drug-target interactions in high-content microscopy. *arXiv preprint arXiv:2503.20158*, 2025.

Ludovico Mitchener, Jon M Laurent, Alex Andonian, Benjamin Tenmann, Siddharth Narayanan, Geemi P Wellawatte, Andrew White, Lorenzo Sani, and Samuel G Rodriques. Bixbench: a comprehensive benchmark for llm-based agents in computational biology. *arXiv preprint arXiv:2503.00096*, 2025.

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.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.

Elana Simon, Kyle Swanson, and James Zou. Language models for biological research: a primer. *Nature Methods*, 21(8):1422–1429, 2024.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Mohamed Amin, Le Hou, Kevin Clark, Stephen R Pfohl, Heather Cole-Lewis, et al. Toward expert-level medical question answering with large language models. *Nature medicine*, 31(3):943–950, 2025.

Maciej Sypetkowski, Morteza Rezanejad, Saber Saberian, Oren Kraus, John Urbanik, James Taylor, Ben Mabey, Mason Victors, Jason Yosinski, Alborz Rezazadeh Sereshkeh, et al. Rxrx1: A dataset for evaluating experimental batch correction methods. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 4285–4294. IEEE, 2023.

Gregory P Way, Ted Natoli, Adeniyi Adeboye, Lev Litichevskiy, Andrew Yang, Xiaodong Lu, Juan C Caicedo, Beth A Cimini, Kyle Karhohs, David J Logan, et al. Morphology and gene expression profiling provide complementary information for mapping cell state. *Cell systems*, 13 (11):911–923, 2022.

Zhiting Wei, Yiheng Wang, Yicheng Gao, Shuguang Wang, Ping Li, Duanmiao Si, Yuli Gao, Siqi Wu, Danlu Li, Kejing Dong, et al. Benchmarking algorithms for generalizable single-cell perturbation response prediction. *Nature Methods*, 23(2):451–464, 2026.

Kenny Workman, Zhen Yang, Harihara Muralidharan, Aidan Abdulali, and Hannah Le. scbench: Evaluating ai agents on single-cell rna-seq analysis. *arXiv preprint arXiv:2602.09063*, 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, 2026.

Yihang Xiao, Jinyi Liu, Yan Zheng, Shaoqing Jiao, Jianye Hao, Xiaohan Xie, Mingzhi Li, Ruitao Wang, Fei Ni, Yuxiao Li, et al. Cellagent: Llm-driven multi-agent framework for natural language-based single-cell analysis. *BioRxiv*, pp. 2024–05, 2024.

APPENDIX

This appendix provides supplementary material to complement the main paper. It includes:

- **Section A** provides an extended discussion of related work on biological post-training, knowledge transfer, phenotypic benchmarks, and scientific agents.
- **Section B** describes benchmark construction, ground-truth qualification, and task generation for MorphKnow and MorphAnalysis. It also gives representative examples of MorphKnow and MorphAnalysis tasks, prompts, and analysis traces.
- **Section C** gives the complete post-training, optimization, and evaluation configurations for SFT, SD, OPD, and GRPO.
- **Section D** reports additional main results and robustness analyses across tasks and model backbones.
- **Section E** provides supplementary analyses of knowledge portability, including composed comparisons, entity-level OOD generalization, and training dynamics.
- **Section F** presents additional experiments on preserving analytical behavior, fact-specific downstream transfer, and explicit fact recall.

A RELATED WORKS

A.1 PHENOTYPIC BENCHMARKS AND SCIENTIFIC AGENTS IN BIOLOGY

Recent benchmarks in perturbational biology increasingly evaluate whether models can predict cellular responses under unseen perturbations or experimental contexts. Single-cell benchmarks such as PerturBench Wu et al. (2026) and scPerturBench (Wei et al., 2026) focus on perturbation-response prediction, while virtual-cell efforts such as AssayBench (De Brouwer et al., 2026) extend this paradigm toward phenotypic screening by predicting assay outcomes from experimental context. Across these settings, experimental measurements primarily serve as prediction targets: models are evaluated on how well they anticipate an unobserved cellular or assay-level response. Image-based phenotypic profiling provides a complementary benchmark setting. Existing Cell Painting benchmarks such as CHAMMI-75 (Agrawal et al., 2026), MorphoHELM (Hayir et al., 2026) and rxrx3-core (Kraus et al., 2025) primarily evaluate whether image-derived representations preserve biologically meaningful variation across perturbations while remaining robust to technical effects. Here, morphology functions as a model input or representation space. MorphKnow instead assigns morphology a different role: experimentally measured phenotypic profiles are used as a source of biological knowledge, from which empirical relationships are curated and subsequently learned and evaluated in a closed-book language-model setting.

A parallel line of work develops scientific agents that directly interact with biological data and computational tools. Systems and benchmarks such as CellAgent (Xiao et al., 2024), BixBench (Mitchener et al., 2025), and scBench (Workman et al., 2026) evaluate agents on realistic biological-analysis workflows involving data inspection, computation, and multi-step reasoning, while broader platforms such as Biomni (Huang et al., 2025) extend agentic workflows across diverse biomedical research tasks. More recent systems move toward increasingly complete scientific discovery loops: Co-Scientist (Gottweis et al., 2026) uses multi-agent scientific reasoning to generate and refine biomedical hypotheses that are subsequently validated experimentally, while Robin (Ghareeb et al., 2026) connects literature-based hypothesis generation, experimental data analysis, and hypothesis refinement in an iterative therapeutics-discovery workflow. These efforts ask whether AI systems can perform or coordinate increasingly large portions of the scientific workflow using existing parametric knowledge, retrieved information, tools, and experimental observations. Our setting asks a complementary question: if specific empirical biological knowledge is introduced through post-training, does that newly acquired knowledge subsequently become usable in scientific analysis? MorphKnow and MorphAnalysis therefore connect phenotypic benchmarking and scientific agents through an explicit path from knowledge acquisition to downstream application.

Table 2: Dataset-level coverage of MorphKnow. Knowledge classes are included only when supported by the experimental design and ground-truth qualification procedure of the corresponding dataset.

Dataset	Cells	Perturbations	Knowledge classes	Questions
CPJUMP1	A549	compound × 2 time points	Conditionality	314
CPG004 (batch 1)	A549	compound, 6-point dose	Relationship, Conditionality	1,800
CPG004 (batch 2)	A549, MCF7, U2OS	compound × 3 time points	Magnitude, Localization, Conditionality	530
CPG0016	U2OS	CRISPR, ORF, compound	Relationship, Magnitude, Localization	6,983
rxrx1	HEPG2, HUVEC, RPE, U2OS	siRNA	Relationship, Conditionality	2,681
rxrx2	HUVEC	soluble factor	Relationship, Magnitude	412
rxrx3-core	HUVEC	gene, compound	Relationship, Magnitude, Localization	1,280
rxrx19b	HUVEC (cytokine storm)	compound	Magnitude, Localization	1,004

A.2 Transfer of Post-Trained Knowledge and Capabilities

Post-training is widely used to specialize general-purpose language models by incorporating domain-specific knowledge and supervision. In biology and biomedicine, continued pre-training, instruction tuning, and task-specific fine-tuning have been used to strengthen biomedical knowledge, biological reasoning, and increasingly multi-omics and sequence-level understanding (Singhal et al., 2023; Cui et al., 2024). More recent work has begun to examine whether such specialization remains robust beyond the biological entities and contexts represented during training, and how different post-training stages shape this generalization. This goal is complementary to retrieval-augmented approaches (Singhal et al., 2025; Simon et al., 2024), which make external knowledge available at inference time: post-training instead studies whether knowledge can become internalized and remain available for subsequent reasoning and decision making (Fesser et al., 2026). Recent methods also combine knowledge-oriented training with retrieval, suggesting that the two mechanisms need not be mutually exclusive.

In parallel, general-purpose post-training research has increasingly moved from improving individual reasoning benchmarks toward studying the transferability of learned reasoning capabilities. Work (Huan et al., 2025; Hu et al., 2026; Kou et al., 2026) across mathematics, rule-based reasoning, coding, science, and planning asks whether reasoning learned in one setting transfers to qualitatively different tasks, and how this transfer depends on the choice of supervised or reinforcement-based post-training. Our work studies a complementary transfer object: rather than asking how far a learned reasoning capability travels, we ask whether new experimentally measured biological knowledge acquired through post-training can remain usable as the model moves from knowledge recall to inference and ultimately to experimental data analysis.

B Benchmark Construction

B.1 Experimental Datasets and Knowledge Coverage

MorphKnow draws on eight perturbational morphology datasets (Table 2) Chandrasekaran et al. (2024); Way et al. (2022); Kraus et al. (2025); Sypetkowski et al. (2023); Chandrasekaran et al. (2023); Cuccarese et al. (2020). Each dataset supports only the knowledge types justified by its experimental design: a dataset measured in a single condition cannot support statements about how a relation changes across conditions, while a dataset with insufficient negative-control replication cannot reliably support control-referenced effect-magnitude comparisons. We therefore extract different knowledge classes from different datasets, and knowledge coverage is intentionally sparse across datasets.

The uneven coverage across datasets follows from four experimental constraints.

1. **cpg0016 supports no Conditionality.** cpg0016 is measured in a single cellular and temporal context, so there is no matched experimental axis over which a relation can be compared. In contrast, CPJUMP1 contributes only Conditionality items because its repeated measurements across time support cross-context comparisons. cpg0004 batch 2 also supports Conditionality through its factorial design, while its control-referenced measurements additionally support Magnitude and Localization.

2. **rxrx1 supports neither Magnitude nor Localization.** Each plate contains only a single negative-control well, which is insufficient to reliably estimate the within-plate control dispersion required by our control-referenced magnitude definition. We therefore do not construct control-referenced Magnitude or Localization items from rxrx1; its MorphKnow families are instead based on correlation-derived relationships, which do not require the same control-referenced scale.
3. **rxrx19b supports no Conditionality.** Although rxrx19b provides reliable perturbation-level measurements, it does not provide a matched experimental axis over which the relevant relationships can be compared across contexts. We therefore use it for Magnitude and Localization, but not for Conditionality.
4. **rxrx2 is not used for Localization.** rxrx2 supports reproducible perturbation-level relationships and effect magnitudes, but its available measurements do not satisfy the qualification criteria used for compartment-level comparisons. We therefore restrict this dataset to Relationship and Magnitude knowledge.

B.2 Ground-Truth Qualification

Ground truth is established before benchmark items are selected or rendered. Qualification follows a shared sequence of preprocessing, perturbation-level aggregation, entity filtering, relation qualification, and comparison-level reliability checks, with family-specific modifications for detection and potency tasks. Benchmark construction in Appendix B.1 operates only on measurements that pass the corresponding qualification procedure.

Preprocessing and partition independence. Ground-truth computation uses well-position correction, per-plate MAD normalization, an inverse normal transform, and feature selection Chandrasekaran et al. (2024). We restrict qualification to preprocessing operations that can be applied without jointly fitting across the experimental partitions used to assess reproducibility. In particular, we exclude batch-integration procedures such as Harmony, because jointly aligning the batches would couple the same partitions whose agreement is subsequently used as a reliability signal.

Batch-specific perturbation consensus. Replicate wells are aggregated by the median to obtain one consensus profile for each perturbation within each experimental batch. We retain batch-specific consensus profiles at this stage rather than immediately pooling across batches, so that perturbation and relation reliability can be evaluated from their agreement across independently measured batches.

Entity reliability filtering. For relation-based families, we first require the perturbation itself to exhibit reproducible morphology across batches. For perturbation g, we define:

$$R_g = \mathrm{mean}_{i<j} \cos(\mathbf{z}_{g,i}, \mathbf{z}_{g,j}) \,,$$

where $\mathbf{z}_{g,i}$ is the consensus profile of perturbation g in batch i. We calibrate the scale of chance cross-batch agreement using a pseudo-group permutation null that preserves the number and batch composition of observations while assigning a different perturbation identity in each batch. The upper tail of this null differs across experimental arms; for example, the 99.9-th percentile is 0.450 for the CPG0016 CRISPR arm and 0.316 for the corresponding ORF arm. We use a single operating point, $R_g \geq 0.5$, across arms. This threshold is therefore not interpreted as a significance test; rather, it is a conservative reliability criterion chosen to lie above the upper null reference observed across arms while trading retained coverage against cross-batch reproducibility. The effect of this operating point is summarized in Table 3 for the largest arm:

Detection and potency families require a different treatment. These tasks explicitly distinguish active from inactive perturbations, and inactive perturbations need not exhibit high cross-batch profile consistency. Applying the same R_g criterion would therefore preferentially remove one of the target classes. For these families, admission instead relies on measurement-quality requirements and comparison against matched on-plate negative controls.

Relation qualification. For similarity-based families, candidate related pairs are defined relative to the empirical similarity distribution among admitted entities. We retain pairs above the family-specific 99th percentile rather than applying a universal cosine-similarity threshold, because the

Admission threshold	Genes retained	Qualified pairs	Audit-batch agreement
None	7,137	254,647	47.2%
$R_g > 0.4$	1,756	15,409	81.9%
$R_g \geq 0.5$	**999**	**4,984**	**91.6%**

Table 3: Coverage–reliability trade-off for entity admission in the CPG0016 CRISPR arm. The audit batch is excluded from the computation of R_g, but is not excluded from the final ground-truth computation. The final column therefore measures split reliability of the admission criterion rather than independent validation of the benchmark ground truth.

background similarity distribution varies across datasets and perturbation modalities. This threshold defines the candidate relation pool; subsequent benchmark items must additionally satisfy comparison-level reliability criteria.

Comparison reliability. Many MorphKnow items compare two relations sharing an anchor. Reliability must therefore be assessed on the comparison itself rather than on either relation independently. Let $s_c(A, B) - s_c(A, C)$ denote the difference between the two relevant measurements under context c. We require the sign of Δ to agree across the consensus batches used for qualification and require its magnitude to exceed two estimated noise widths. The noise width is used only as an empirical scale for separating clearly ordered comparisons from measurements near the experimental noise level; it is not interpreted as a calibrated standard error. In particular, cosine similarity is nonlinear and the two terms in Δ share an anchor, so a simple independent homoscedastic error model does not strictly apply. As an audit of the resulting criterion, we also examine agreement on a batch excluded from the computation of the qualification statistic:

Margin	n	Sign agreement across qualification batches	Audit-batch agreement
1–2 noise widths	257	56.0%	74.3%
2–3 noise widths	130	89.2%	89.2%
≥ 3 noise widths	110	**96.4%**	**97.3%**

Table 4: Comparison reliability as a function of empirical margin. The audit batch is excluded from the qualification statistic but contributes to the final ground-truth estimate; these values therefore quantify split reliability rather than independent held-out validation.

B.3 MorphKnow Construction and Task Design

This section describes how the qualified empirical relations in Appendix B.1 are rendered and selected to form MorphKnow. The released benchmark contains 15,004 items spanning four knowledge classes and 16 task families.

Release Format and Input Boundary. Each item contains a task-family identifier, response format, experimental context, question, candidate answers when applicable, reference answer, and output specification. Definitions required by a particular family–for example, the operational definition of a detectable phenotype–are provided in the system field. The experimental context specifies only information intended to be available to the model: cell line, perturbation modality, time point, dose, and feature representation. MorphKnow does not expose the underlying Cell Painting profiles, replicate-level measurements, batch identifiers, or statistics used to construct the ground truth. Source identifiers and replicate depth are also withheld to prevent technical metadata from serving as shortcuts to measurement reliability or detectability. The required output is structured, and the reference answer uses the same surface form as the corresponding candidate when a choice set is provided. For compound-based families, canonical SMILES are additionally provided after the question to make compound identity unambiguous. These descriptors expose molecular structure but do not contain morphology-derived measurements; 4,375 MorphKnow items include SMILES.

Task Families. The four knowledge classes are instantiated through 16 task families with different response formats and experimental settings. Table 5 summarizes the released composition.

Class	ID	n	Format	Family
Relationship	A	5,273	binary	Phenotypic neighbor
	B	622	ternary	Neighbor with `neither`
	C	440	binary	Cross-modality neighbor
Magnitude	D	1,038	binary	Effect magnitude
	E	2,066	ternary	Effect magnitude with `neither_detected`
	J	76	binary	Pairwise potency
	K	54	band	First detectable dose band
Localization	F	1,482	binary	Pairwise compartment effect
	G	480	3-choice	Largest of three compartments
	H	319	ranking	Rank three compartments
Conditionality	I	1,233	binary	Selectivity across cell lines
	L	648	binary	Neighbor relation across cell lines
	M	296	binary	Magnitude relation across doses
	N	433	binary	Compartment composition across doses
	O	230	binary	Magnitude relation across time
	P	314	binary	Neighbor relation across time

Table 5: Composition of MorphKnow by knowledge class, task family, and response format.

B.3.1 Representative Items

The following examples illustrate the canonical question forms. Exact wording varies across templates within each family.

Relationship (A; binary).

> *Context.* In a Cell Painting experiment, U2OS cells were treated with a small molecule at 5 µM for 48 h. Morphological features were extracted with CellProfiler and normalized per plate.
>
> *Question.* Three perturbations are compared in this assay: an anchor, *buparlisib*, and two candidates, *ganetespib* and *benzamil*. Which candidate produces a morphological phenotype more similar to that of the anchor? (canonical SMILES for all three compounds follow)
>
> *Output.* {`"choice": "<compound identifier as written>"`}

Magnitude (E; ternary).

> *System.* A perturbation is considered to have a detectable morphological phenotype when its control-referenced effect exceeds the predefined detection criterion consistently across the qualification batches.
>
> *Question.* Which of `GTF2E1` and `GSTA5` produced the stronger reliably detected morphological deviation from its matched controls? Select `neither_detected` if neither perturbation produced a reliably detected response.
>
> *Output.* {`"choice": "<gene symbol or neither_detected>"`}

Localization (H; ranking).

> *Context.* Cell Painting features are grouped by stain channel: DNA, ER, RNA, AGP, and Mito.
>
> *Question.* For the `NKX3-1` knockout, rank Mito, DNA, and ER by morphological effect magnitude, from largest to smallest.
>
> *Output.* {`"ranking": ["<largest>", "<middle>", "<smallest>"]`}

Conditionality (M; binary).

> *Context.* A549 cells were treated with a small molecule for 48 h. At 3.333 μM, flumazenil produces a stronger morphological effect than cilostazol.

Question. At 1.111 μM, which compound produces the stronger effect: flumazenil or cilostazol?

Output. {"choice": "<compound name>"}

Conditionality items may therefore provide a measured relation in one experimental context and ask how that relation changes under another. The remaining response formats include three-way compartment selection (family G) and selection of the first detectable dose band (family K).

Content-Valued Third Options. Families B and E include a third response, `neither` or `neither_detected`. These responses are semantic ground-truth categories rather than abstention options. They indicate that neither candidate satisfies the corresponding relation or detection criterion and must satisfy the same qualification and split-reliability requirements as the other answers. Because the frequency of this category is determined by the qualified empirical pool, these families are not forced into equal answer frequencies. Their constant-strategy baselines are therefore higher than those of position-balanced binary families, and are reported separately in the benchmark diagnostics below.

Constrained Item Selection. Ground-truth qualification produces substantially more admissible comparisons than are included in MorphKnow. We therefore select items under exposure constraints designed to limit repeated use of the same entities and entity pairs. An entity may appear as an anchor at most twice and as a candidate at most three times, with a cap of four in the ORF arm for feasibility. The same candidate pair may appear at most twice; when it recurs, its measured winner must differ across the two items rather than being created by label inversion. Candidate construction additionally matches activity, measurement reliability, and annotation density to reduce shortcuts based on perturbation strength or entity familiarity. Selection is formulated as a two-stage mixed-integer program over a restricted candidate pool. The first stage maximizes the number of selected items subject to the exposure constraints; conditional on this count, the second stage maximizes the total ground-truth margin. We solve each restricted instance with HiGHS Huangfu & Hall (2018) and iteratively expand the candidate pool using targeted candidate generation. Expansion stops when two consecutive rounds increase the selected item count by less than 1%. This procedure does not certify global optimality over the complete admissible pool. We therefore compare the achieved count with the structural upper bound:

$$\min(\text{quota ceiling}, b\text{-matching bound}, \text{slot bound}) ,$$

which is itself not necessarily tight. Ratios to this bound should not be compared across different quota policies, because changing the exposure caps also changes the corresponding bound.

Surface-Form and Answer-Bias Controls. We analysis two potential benchmark shortcuts: sensitivity to question formulation and systematic answer bias. Figure 7 summarizes both analyses.

For answer bias, we compare the random-answer baseline with the strongest constant strategy within each task family. The latter is defined as the better of always emitting a fixed answer string and always selecting a fixed output position, where the response format permits such a strategy. Binary families are approximately position-balanced around 0.5. Families with semantic third options or non-binary outputs naturally have different constant baselines; these differences are treated as properties of the response format rather than forced away through relabeling.

Across the complete 15,004-item corpus, the item-count-weighted constant-strategy baseline is 0.467, compared with a random-answer baseline of 0.457. Evaluation subsets use the corresponding baseline computed on that subset.

We separately evaluate formulation sensitivity by rendering the same underlying items under alternative phrasings while holding the entities, context, and reference answer fixed. Figure 7 reports base-model accuracy for each formulation within a task family. Most binary families remain close to their format-specific random baseline, while several non-binary families exhibit different baseline levels because of their response spaces. The held-out formulation protocol used for the Accessibility evaluation is described separately in Appendix B.4.

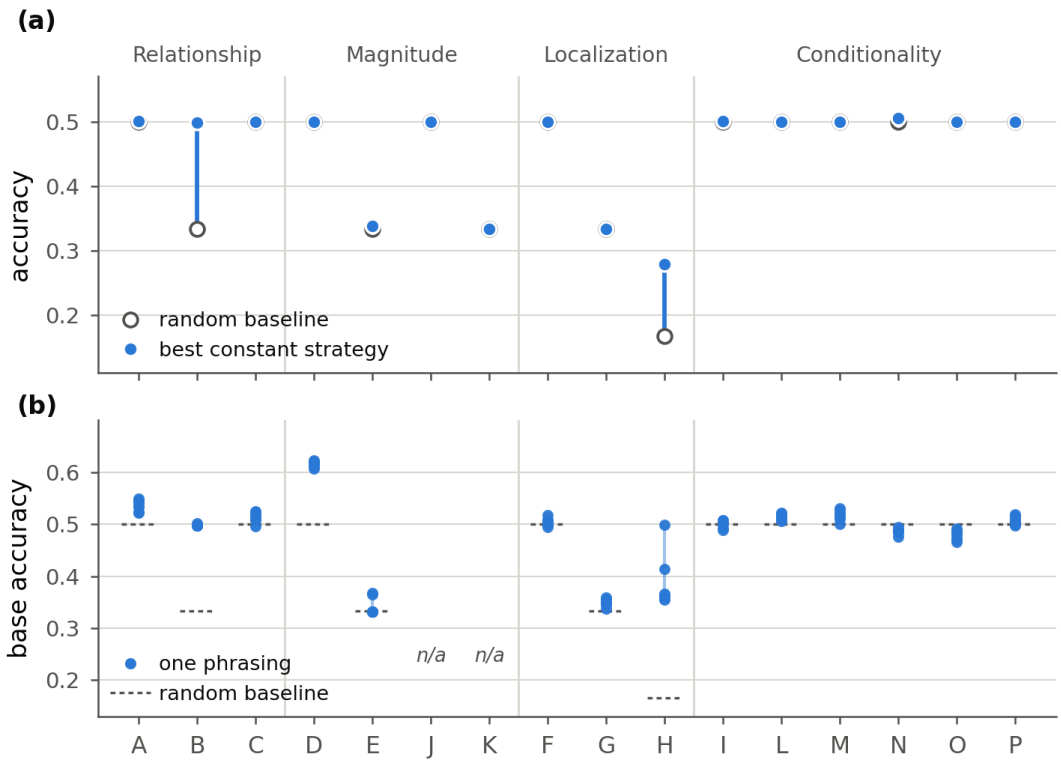

Figure 7: Audits of answer bias and formulation sensitivity in MorphKnow. **(a)** Random-answer accuracy and the strongest constant strategy for each task family. The constant strategy is the better of a fixed answer string and, where defined, a fixed output position. Binary families are approximately balanced at 0.5, whereas ternary, ranking, and other structured-output families have format-specific baselines. **(b)** Base-model (Qwen-3.5-9B) accuracy across alternative question formulations of the same task families; each point corresponds to one phrasing and dashed horizontal segments indicate the random baseline for that response format. Families J and K have only one released formulation and are therefore not included in the cross-formulation audit. Vertical separators and headings group task families into Relationship, Magnitude, Localization, and Conditionality.

B.4 Accessibility: Held-Out Formulations

Accessibility evaluates whether a learned MorphKnow fact remains recoverable when the same underlying relation is expressed in a formulation not seen during post-training. The biological entities, experimental context, target relation, and reference answer are held fixed; only the linguistic formulation of the query changes.

Held-Out Formulation Construction. For each eligible task family, we construct three held-out formulations that are disjoint from the formulations used during post-training. We use GPT-5.6 to rewrite the task at the template level while preserving the experimental setting, queried relation, candidate structure, and required output format. A domain expert reviews every generated formulation for semantic equivalence, answer preservation, and absence of additional biological or experimental information. Formulations are rejected if they remove information required by the original task, introduce new cues, alter the queried relation, or otherwise change the correct answer. Template construction is performed independently of model performance. Families J and K contain only a single canonical formulation and are excluded from the Accessibility evaluation. The resulting evaluation therefore covers 14 of the 16 MorphKnow task families.

Evaluation Set and Paired Rendering. We construct a 1,000-item Accessibility set with 250 items from each of the four MorphKnow knowledge classes. Within each class, items are distributed approximately evenly across the eligible task families:

A	B	C	D	E	F	G	H	I	L	M	N	O	P
126	62	62	125	125	84	83	83	42	42	41	42	41	42

We use within-class rather than corpus-proportional sampling so that each eligible task family contributes sufficient examples to diagnose formulation sensitivity. The resulting set is therefore a balanced portability evaluation rather than an estimate of the task-family distribution of the full MorphKnow corpus.

For each selected item, one of the three held-out formulations defined for its task family is sampled at random and used to render the evaluation query. The biological entities, experimental context, target relation, response format, and reference answer are preserved from the corresponding MorphKnow item. Thus, the evaluation changes the linguistic realization of the query while keeping the underlying knowledge target fixed. All evaluated model conditions use the same rendered Accessibility set.

Semantic Preservation Audit. We additionally audit held-out formulations that alter the surface polarity of the question. For example, a similarity query may ask which candidate "departs least" from an anchor rather than which candidate is "most similar." Such changes modify the surface form but should preserve the underlying relation and reference answer. Across the 1,000-item Accessibility set, 124 items are rendered with such reversed-polarity formulations. The reference answer agrees with that of the corresponding original MorphKnow item in all 124 cases. Together with the domain-expert review above, this verifies that the Accessibility evaluation changes formulation rather than the underlying biological target. Figure 7 provides a complementary audit of formulation sensitivity by comparing model accuracy across alternative task formulations. This analysis is conducted independently of held-out-template construction and is not used to select Accessibility formulations.

B.5 Inferential Extensibility

Inferential Extensibility evaluates whether knowledge acquired from MorphKnow can support conclusions that were not themselves directly supervised. We consider two settings. *Compositional inference* combines learned facts to derive a new relation among previously supervised entities, whereas *entity-held-out transfer* evaluates facts involving biological identities excluded from MorphKnow post-training supervision.

B.5.1 Compositional Inference

A compositional item targets a relation that is absent from the post-training supervision but can be derived from relations that are present in it. We construct two forms of composition: scalar-chains and selector–property lookups.

Scalar-chain Composition. For a scalar phenotypic quantity $q_c(\cdot)$ measured under a fixed experimental context c, post-training provides two comparisons,

$$q_c(A) > q_c(Z), \qquad q_c(Z) > q_c(B),$$

while the evaluation query asks for the previously unsupervised comparison between A and B. Because all three quantities refer to the same measured scalar under the same dataset, cell line, dose, time point, and estimator, the conclusion follows from the ordering

$$q_c(A) > q_c(B).$$

We deliberately restrict this construction to scalar quantities for which ordering is transitive. In particular, we do not compose phenotypic-similarity relations: high similarity between A and Z, and between Z and B, does not imply high similarity between A and B. The released set contains 722 scalar-chain items, including 554 derived from Magnitude family D and 168 from family I.

Selector–property Lookups. The second setting composes knowledge across two stages. The model must first use a learned comparison to identify which of two perturbations satisfies a selector criterion, and then retrieve a property associated with the selected perturbation.

Concretely, the first step identifies

$$p^* = \text{argmax}_{p \in \{A,B\}} m_c(p),$$

and the second asks which of two morphological compartments carries the larger effect for p^*. Each underlying pair is instantiated in matched form for the stronger and weaker perturbation. The compartment answers are required to differ within a matched pair, preventing the property answer from being recovered without first resolving the selector step. This construction yields 160 selector–property items.

Supervision and Leakage Controls. For every compositional item, we retain the MorphKnow identifiers of the prerequisite facts and verify that all prerequisite facts are present in the post-training split. We then search across all MorphKnow training families and response formats to confirm that the composed target relation is not directly supervised elsewhere. For scalar-chain composition, the two endpoint entities therefore never appear in a training item that directly determines the target ordering. For selector–property composition, the selector relation and the property relation are each supervised separately, but no single training item provides the final composed answer. Matched selector–property pairs are additionally retained only when the two property answers differ, preventing the selector step from being bypassed. All referring expressions are automatically checked to resolve to entities explicitly named in the item, and premises are checked not to state the target answer directly. Answer-position frequencies and constant-strategy baselines are checked separately from compositional correctness.

B.5.2 Entity-Held-Out Generalization

Entity-held-out generalization tests whether post-training on MorphKnow supports predictions about relations involving biological identities absent from the MorphKnow supervision set. "Held out" therefore refers specifically to MorphKnow post-training supervision; it does not imply that the underlying base model has never encountered the entity during pretraining.

Canonical Entity Identity. Genes and soluble factors are resolved by their current NCBI symbols. Compounds require additional alias resolution because the same molecule may appear under generic names, development codes, catalogue identifiers, salt forms, or alternative SMILES strings. We therefore group compound aliases at the connectivity level using the first block of the InChIKey. This procedure maps 7,304 observed names to 7,147 canonical identities. Connectivity-level grouping may merge stereoisomers that are chemically distinct but share the same first InChIKey block; this is conservative for split construction because such variants are placed on the same side of the split. None of the affected aliases appears in the released evaluation set.

Coverage-controlled Split Construction. We construct the held-out split at the canonical-identity level. Once an identity is assigned to the evaluation side, every MorphKnow item containing that identity is excluded from post-training supervision. Because identities differ substantially in how many MorphKnow items they participate in, the split is coverage-controlled rather than a random sample of biological identities. Identities are considered in increasing order of the number of items they touch. An identity is withheld when doing so contributes to a knowledge class that has not yet reached its target evaluation size and does not increase that class beyond 1.25 times its target. Prioritizing lower-degree identities prevents a single highly connected entity from overshooting a class target in one step. The released split withholds 537 canonical identities and contains 13,189 post-training items and 1,815 entity-held-out evaluation items. The evaluation set contains 500 Relationship, 497 Magnitude, 400 Localization, and 418 Conditionality items.

Split Verification. No post-training item contains a withheld identity, and every evaluation item contains at least one withheld identity. Thus, an entity-held-out item need not consist exclusively of unseen identities; rather, each evaluated relation includes at least one entity for which no MorphKnow supervision is provided. For compound items, the standard MorphKnow input still includes canonical SMILES. Thus, entity holdout removes all MorphKnow phenotypic supervision involving the held-out compound, while retaining the molecular descriptors normally available in the task. The post-training and evaluation partitions are disjoint and together recover all 15,004 MorphKnow items. Template assignments are preserved when the split is written, so split construction changes entity exposure without introducing an additional formulation shift.

B.6 MorphAnalysis Construction and Evaluation

MorphAnalysis evaluates whether experimentally measured phenotypic knowledge acquired during post-training can be used in downstream scientific analysis. We instantiate three types of decisions that arise naturally in perturbational profiling: selecting among alternative processing outputs, designing a reduced experimental readout, and transferring an estimate across experimental contexts.

The tasks are grounded in real perturbational experiments but use controlled numerical instantiation where necessary. Biological entities, experimental contexts, MorphKnow relations, and the empirical scales used to calibrate the tasks are derived from real datasets. The controlled component is used to isolate the contribution of biological knowledge, balance competing decision factors, and remove knowledge-free shortcuts. It therefore modifies the numerical realization of a realistic analysis problem rather than replacing the scientific decision with an unrelated synthetic task.

B.6.1 Harness, Evidence Conditions, and Grading

Each task is instantiated as a working directory. The model receives a scenario description and access to two tools: `list_files`, which exposes files available for analysis, and `run_python`, which executes Python with NumPy and pandas inside the task directory and returns standard output and error. The final response is a single JSON object, for example {`"choice": "plan_2"`}, graded by exact match.

The primary MorphAnalysis evaluation uses the **no-fact** setting: no biological fact is provided explicitly, and the model must rely on knowledge acquired during post-training together with the evidence available in the task environment.

We additionally construct two auxiliary diagnostic conditions for ablation analysis. These are not separate benchmark tasks; they use the same scenarios, files, candidate actions, and graders as the primary evaluation, differing only in whether a `Reference facts` block is inserted:

- **Oracle-fact.** The relevant MorphKnow fact is supplied explicitly. This removes knowledge acquisition or retrieval as a bottleneck and isolates whether the model can correctly integrate the biological fact with the task evidence.
- **Counterfactual-fact.** The corresponding directional fact is replaced by its conflicting counterpart. This serves as an intervention on the biological knowledge signal, testing whether the downstream decision changes in the expected direction rather than remaining fixed under the same task evidence.

Thus, benchmark performance is reported on the no-fact condition, while the oracle-fact and counterfactual-fact variants are used as controlled diagnostics of knowledge utilization.

B.6.2 Design Principles and Shortcut Controls

MorphAnalysis is designed to preserve the structure of realistic scientific decisions while making the contribution of prior biological knowledge identifiable.

Knowledge and task evidence are complementary. A correct decision requires both biological knowledge and evidence contained in the task environment. In Processing Selection, MorphKnow identifies relationships that a useful processed representation should preserve, whereas the candidate matrices reveal which processing output actually preserves them. In Readout Design, MorphKnow identifies where the biological effect is concentrated, whereas the QC file identifies which readout is technically compromised. In Context Transfer, MorphKnow determines whether a relation should transfer with the same or reversed direction, whereas the source-context measurements determine the quantity to transfer and which measurements should be excluded.

The scientific setting is experiment-grounded. Entity identities, cell lines, perturbation modalities, experimental axes, and knowledge labels are derived from the underlying perturbational datasets. When numerical values must be generated to control the information available to the model, their ranges and structure are calibrated to the corresponding experimental measurements. Controlled generation is therefore used to disentangle information sources and suppress unintended shortcuts, not to abstract away the application.

Solvability is verified programmatically. For every released task, an oracle procedure given the relevant MorphKnow fact and the task files must recover the reference answer. This verifies that the intended information is sufficient and that the required computation can be reproduced from the released task environment.

Shortcut resistance is evaluated directly. We do not assume that a shortcut available before and after post-training cancels in their performance difference. Instead, we test knowledge-free and evidence-free strategies directly. These include fixed-answer strategies, strategies using only one information source, option-structure heuristics, and family-specific statistical shortcuts. Semantically meaningful terminology is retained when it reflects realistic analysis practice, but candidate task variants are screened for cases in which terminology itself dominates evidence use. Construction factors such as answer position, relation direction, entity ordering, nuisance assignment, and swap status are balanced exactly where the available task size permits and otherwise as closely as possible. Family-specific balance statistics are reported below.

B.6.3 A1: Processing Selection

Decision. A1 represents a common analysis choice in which the same perturbational screen can yield alternative processed representations depending on choices such as replicate aggregation or feature selection. The model receives two candidate perturbation-by-perturbation correlation matrices and must choose which one should be retained for downstream module detection.

MorphKnow provides relationships that should be preserved by a biologically useful representation. The matrices themselves are still required: knowing that two perturbations are related does not determine which processing choice better preserves that relationship.

Representative Task. We show a example constructed from Cell Painting CRISPR screen containing 29 knockouts with five wells per perturbation. Two candidate datasets differ only in how replicate wells are aggregated:

> `agg_median/`: per-perturbation profile = median of its wells `agg_mean/`: per-perturbation profile = mean of its wells

The corresponding correlation matrices are stored at

> `runs/run_51e8/agg_median/correlation_matrix.csv`
> `runs/run_51e8/agg_mean/correlation_matrix.csv`.

The model is asked which representation should be retained for downstream module detection. In the oracle-fact condition it additionally receives the experimentally derived relation

> *APOA4 and MSH5 are related in this assay.*

The model must then inspect the candidate matrices to determine which processing output better preserves this relation. The reference answer for this instance is `agg_mean`.

Controlled Numerical Instantiation. The released A1 set contains 480 items. Each task embeds one or more MorphKnow relationships into a larger perturbation matrix. The numerical construction is designed to instantiate the effect that a processing decision can have on downstream biological structure while preventing the correct matrix from being recognized through unrelated global statistics.

Profiles are generated in an 80-dimensional latent representation with six factors. Related perturbations share latent structure plus noise, whereas unrelated pairs are approximately orthogonal. Six batch offsets introduce batch-structured variation. Candidate processing outputs are constructed so that one preserves the selected MorphKnow relationships while the alternative attenuates their relative placement.

In the preserving matrix, known related pairs occupy approximately the 80th-95th percentile of the matrix's correlation distribution. In the alternative matrix, those pairs are moved toward the 40th-60th percentile while selected unrelated pairs are elevated. The intended relationships are not placed

at the extreme tail, because doing so would allow them to be identified as generic outliers without knowing which biological pairs matter.

Each manipulated relation is paired with a structural counterpart receiving the opposite assignment, reducing differences in matrix-level spectral statistics. Nuisance content is symmetrized across the two candidate matrices, and correctness is balanced across processing labels. Filler genes are drawn from outside the MorphKnow entity set used to define the planted relations.

Processing-label Analysis. The processing names are intentionally kept semantically meaningful rather than replaced by anonymous labels, because such terminology is part of the real analysis decision. We therefore screen candidate processing contrasts for the possibility that the names themselves induce a strong answer preference. For each contrast, we measure the base model's dominant-label preference, defined as $\max(p, 1-p)$ where p is the probability of choosing the first processing alternative, together with the average number of tool calls made before answering.

Processing contrast	Dominant-label preference	Tool calls / run	Retained
`agg_mean` / `agg_median`	0.63	6.0	Yes
`all_features` / `feature_select`	0.82	5.8	Yes
Batch-correction contrast	0.95	4.7	No
Normalization-reference contrast	0.99	3.4	No

The excluded contrasts show both near-deterministic name preferences and substantially less tool use, consistent with the model committing to an answer from the terminology rather than inspecting the numerical evidence. We therefore exclude these variants from A1. The retained contrasts may still induce a directional preference–most notably the `all_features`/`feature_select` pair–but this preference is not predictive of the reference answer because correctness is balanced across the two processing choices. A fixed processing-name strategy therefore remains at chance. The screening criterion is instead intended to avoid task variants in which semantic priors substantially suppress evidence-seeking behavior.

Difficulty Strata and Shortcut Controls. A1 contains three distributed settings with 1, 3, or 8 planted relations, with 120 items at each level, together with 120 hub-structured items containing eight relations. Selecting a fixed processing alternative scores 0.50 by construction. We additionally test strategies based on global matrix magnitude, variance, spectral properties, and local difference structure. The controlled construction prevents these statistics from identifying the correct processing output independently of the relevant MorphKnow relationships. Each task retains the identifiers of the MorphKnow entries used in its construction. Across the 480 A1 items, 1,110 MorphKnow entries are referenced.

B.6.4 A2: Readout Design

Decision. A2 represents a follow-up experimental-design decision. A perturbation identified in a primary Cell Painting screen advances to a reduced-plex confirmation assay in which only three of six readout blocks can be retained. The model chooses among four candidate three-block configurations.

The decision combines two independent information sources. Let k denote the block carrying the perturbation's strongest morphological effect and b the block identified as technically compromised by the QC report. Task construction requires $k \neq b$. We refer to all blocks other than k and b as the *remaining blocks*. The four candidate configurations instantiate the full 2×2 structure:

Option	Composition	Contains k	Contains b
A	k + two remaining blocks	Yes	No
B	k, b + one remaining block	Yes	Yes
C	b + two remaining blocks	No	Yes
D	three remaining blocks	No	No

Thus, neither the biological knowledge nor the QC report alone determines the correct option.

Representative Task. We use *(S)-crizotinib* advanced from a HUVEC Cell Painting screen under a cytokine-storm condition to a confirmation experiment as a representative task. Only three of six readout blocks can be retained:

> A. keep {Brightfield, ER, Mito} B. keep {AGP, DNA, RNA} C. keep {DNA, ER, RNA}
> D. keep {AGP, Brightfield, Mito}

The primary batch's QC report is available at `runs/qc_report.csv`. In the oracle-fact condition, the model is told that the strongest morphological effect is in **ER**. Inspection of the QC file shows that **DNA** is compromised, with relative control dispersion 1.43 and 27.6% of wells flagged, compared with approximately 0.95–1.03 dispersion and 5–7% flagged wells for the remaining blocks. The correct choice is therefore [A], which retains ER while excluding DNA.

Controlled Numerical Instantiation. The entity pool consists of perturbations whose strongest morphological block is unambiguous, yielding 247 items: 175 compounds from RxRx19b and 72 gene perturbations from RxRx3-core. The real experiment determines the perturbation identity, the strongest-effect block k, and the relevant sampling structure. The QC layer is then instantiated within ranges calibrated to observed channel-level variation. The compromised block has relative control dispersion sampled from $U(1.35, 1.60)$ and flagged-well fraction from $U(0.18, 0.32)$, compared with $U(0.94, 1.06)$ and $U(0.03, 0.08)$ for the remaining blocks. These ranges are anchored to observed variation in the underlying perturbational data rather than chosen as arbitrary numerical scales. QC values appear only in the task file and are not repeated in the scenario.

Option Symmetry and Shortcut Controls. Six blocks are used so that the twelve block slots across the four three-block configurations can be balanced symmetrically: each block appears exactly twice. This prevents option overlap alone from revealing which configuration simultaneously contains the biologically informative block and excludes the compromised block. By construction $\mathrm{Acc}(\text{random}) = 0.25, \mathrm{Acc}(\text{knowledge only}) = 0.50, \mathrm{Acc}(\text{QC only}) = 0.50$, whereas an oracle procedure combining both information sources reaches 1.00.

We additionally evaluate pre-specified knowledge-free heuristics, including fixed answer position, fixed block or block triple, maximal or minimal overlap with the QC-excluded option, average overlap with the other options, option size, and the number of QC-clean blocks. Each such strategy is required to remain within five percentage points of the 0.25 random baseline before release.

B.6.5 A3: Context Transfer

Decision. A3 represents the transfer of an experimentally measured quantity from a well-characterized source context to a target context in which the perturbation itself has not been measured. The source context contains full perturbation measurements, whereas only controls are available in the target context. The target quantity therefore cannot be estimated directly from target-context perturbation data.

Each task combines two decisions. First, the model must determine whether the relevant biological relation is conserved or reversed between source and target contexts. This direction is determined by MorphKnow. Second, the model must identify which of two flagged source plates contains a genuine plate-level artifact and should be excluded. This second decision must be made from the task files. The four candidate actions form the cross product

$$\{\text{exclude flagged plate 1},\ \text{exclude flagged plate 2}\} \times \{\text{positive transfer},\ \text{negative transfer}\}.$$

Biological knowledge therefore determines transfer direction, whereas statistical analysis of the source measurements determines which observations should enter the estimate.

Representative Task. In `cpjump1_time_C_0004`, the anchor is *anandamide*, with *simvastatin* and *BAM7* as the two candidates in A549 cells. The analysis transfers from 24 h to 48 h. The target quantity is the 48 h retrieval margin

$$\mathrm{corr}(\text{anandamide}, \text{simvastatin}) - \mathrm{corr}(\text{anandamide}, \text{BAM7}).$$

The perturbations were measured at full depth at 24 h using five plates with 20 wells per plate. At 48 h, only control wells are available, so the target retrieval margin is not directly observable. QC

flags `SRC_P05` and `SRC_P04`, exactly one of which contains a genuine plate-level offset. The four candidate plans combine the plate to exclude with multiplication of the source estimate by either $+0.80$ or -0.80. In the oracle-fact condition, the model additionally receives the information that the relative ordering of *simvastatin* and *BAM7* with respect to *anandamide* is **conserved** between 24 h and 48 h. It must combine this relation with the statistical evidence identifying the affected source plate. The reference answer for this instance is `plan_2`.

Experimental Axes and Estimators. The 240 released A3 tasks span cell-line, time, and dose shifts across five context-axis configurations. Cell-line transfer uses rxrx1 and CPG004, time transfer uses CPJUMP1 and CPG004, and dose transfer uses the multi-dose CPG004 screen. Two estimator families reflect the corresponding experimental quantities. A neighbor-margin estimator is used in 144 tasks and compares correlations with two candidate perturbations. An intensity-margin estimator is used in 96 tasks and compares control-referenced effects between two measurement groups.

Relation Labels and Rendering Verification. Conserved and reversed labels are stored in a canonical structured relation table after accounting for entity ordering. These canonical records, rather than the rendered question text, determine the ground-truth direction. As an end-to-end consistency check, the rendered prompts are independently parsed and the implied relation is compared with the structured record; all 1,488 checked instances agree with the canonical labels.

Controlled Numerical Instantiation. The *sign* of the transfer is determined by the experimentally measured MorphKnow conserved/reversed relation. The transfer magnitude, 0.80, is a controlled benchmark parameter used to give the two possible directions a common numerical scale; it is not an experimentally estimated biological transfer coefficient.

The remaining numerical ranges are calibrated to the source experiments. Source margins use $|m| \sim U(0.12, 0.40)$, and well-level noise uses $\sigma \sim U(0.12, 0.26)$, corresponding to the lower portion of the observed empirical ranges. Each task contains five or six source plates with 12–20 wells per plate and 32 control wells in the target context. Exactly one of the two flagged source plates receives a genuine plate-level offset. Construction requires that the affected plate be statistically identifiable while remaining on the same scale as the biological signal: its deviation must satisfy $z \geq 3.5$, remain below 0.32 in absolute magnitude, and exceed twice the deviation of the flagged decoy plate. These constraints act on the nuisance realization and do not alter the MorphKnow relation determining transfer direction.

Balancing and Baselines. The 240 released tasks are balanced across the five context-axis configurations, two relation directions, two flagged-plate positions, and four answer positions. Each of the four plans is correct in 60 tasks. Source-margin sign and candidate ordering are also balanced, and all released entity combinations are distinct. By construction, $\mathrm{Acc}(\text{random}) = 0.25$, $\mathrm{Acc}(\text{knowledge only}) = 0.50$, $\mathrm{Acc}(\text{statistics only}) = 0.50$, whereas an oracle procedure combining the correct MorphKnow relation with the source-context measurements reaches 1.00.

C Post-Training and Evaluation Details

C.1 Models and Adapters

Backbones. Our main backbone is Qwen3.5-9B using its instruction-tuned checkpoint and non-thinking chat template (`enable_thinking=False`); the same template is used for training and evaluation. Unless otherwise noted, the main Qwen evaluations use the same non-thinking decoding protocol. We additionally evaluate a GRPO configuration with thinking enabled to examine whether explicit reasoning changes knowledge acquisition and portability. Selected replication experiments use GLM-4-9B-0414 with its default chat template.

LoRA configuration. All methods update rank-64 LoRA adapters while keeping the backbone frozen. We use scaling factor $\alpha = 128$ and apply adapters to the attention and MLP projection layers supported by each backbone. Qwen3.5-9B has 160.4M trainable adapter parameters and GLM-4-

Table 6: Backbone and adapter configuration.

	Qwen3.5-9B	GLM-4-9B-0414
Chat template	non-thinking	default
LoRA rank / α	64 / 128	64 / 128
Trainable adapter parameters	160.4M	190.4M
LoRA dropout (SFT/SD/OPD; GRPO)	0.05; 0	0.05; 0
Training precision	bf16	bf16
Maximum training length	1,024	768

9B-0414 has 190.4M. LoRA dropout is 0.05 for SFT, SD, and OPD and 0 for GRPO. Training uses bf16; SD and OPD compute the distillation loss in fp32.

No training example is truncated at these sequence lengths. GRPO uses prompts of at most 768 tokens and generations of at most 64 tokens. MorphKnow-based evaluations use a 2,048-token context, while multi-turn MorphAnalysis tasks use a 65,536-token context.

C.2 Training Objectives

Let x denote a MorphKnow question, $a = (a_1, \ldots, a_{|a|})$ its reference answer, p_θ the student model with trainable adapters, and p_{θ_0} the frozen initial model. Token-level objectives are applied only to response positions.

Supervised fine-tuning. SFT minimizes the negative log-likelihood of the reference response:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\sum_{t=1}^{|a|} \log p_\theta(a_t \mid x, a_{<t}). \tag{1}$$

Answer-conditioned teacher. SD and OPD use the same frozen teacher, p_{θ_0}, but provide it with privileged access to the known answer. Specifically, the teacher receives the original question together with the reference answer and then the original prompt, whereas the student receives only the original prompt. For a response prefix $h_{<t}$, we define

$$q_t(\cdot) = p_{\theta_0}(\cdot \mid c(x, a), h_{<t}), \qquad p_t(\cdot) = p_\theta(\cdot \mid x, h_{<t}), \tag{2}$$

where $c(x, a)$ denotes the answer-conditioned teacher context.

Self-distillation. SD evaluates the forward-KL distillation objective along the reference response, i.e., $h_{<t} = a_{<t}$:

$$\mathcal{L}_{\mathrm{SD}}(\theta) = \sum_{t=1}^{|a|} \mathrm{KL}(q_t \,\|\, p_t)\,. \tag{3}$$

On-policy distillation. OPD uses the same teacher and forward-KL objective, but evaluates it along a response z sampled from the current student. Sampling is treated as stop-gradient, so gradients flow only through the student token distributions:

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{z \sim \mathrm{sg}[p_\theta(\cdot|x)]} \left[\sum_{t=1}^{|z|} \mathrm{KL}(q_t \,\|\, p_t) \right], \tag{4}$$

with $h_{<t} = z_{<t}$. During training, OPD samples with temperature 0.7, top-p 0.8, and at most 32 new tokens. A reverse-KL variant is treated as an auxiliary ablation rather than part of the main comparison.

Loss normalization. For Qwen3.5-9B, token-level losses are normalized by the total number of response tokens in each optimizer step (sampled response tokens for OPD). For GLM-4-9B-0414, they are normalized by the number of reference-answer tokens in the step.

Table 7: Main Qwen3.5-9B training hyperparameters.

	SFT	SD / OPD	GRPO
Optimizer	AdamW	AdamW	AdamW
Weight decay	0.01	0.01	0.01
Peak learning rate	1×10^{-4}	1×10^{-4}	1×10^{-5}
Schedule	cosine to 0	cosine to 0	constant
Warm-up	3%	3%	3 steps
Gradient clipping	1.0	1.0	1.0
Effective batch	64 responses	64 responses	128 questions $\times$ 8 rollouts
Training budget	10 epochs	10 epochs	40 steps
Maximum length	1,024	1,024	768 prompt + 64 response (12,288 for thinking)
Training sampling	–	OPD: $T = 0.7$, top-$p = 0.8$	$T = 1.0$, top-$p = 1.0$
KL regularization	–	forward KL	$\beta = 10^{-3}$
GRPO clip / policy epochs	–	–	0.2 / 1

GRPO. For each question, GRPO samples $G = 8$ responses from the rollout policy. A response receives reward $r_i = 1$ only when it parses as the required JSON object and contains the correct answer; otherwise $r_i = 0$. There is no separate format reward. Group-relative advantages are

$$\hat{A}_i = \frac{r_i - \bar{r}}{s_r + \delta}, \tag{5}$$

where $\bar{r}$ and s_r are the mean and standard deviation of rewards within the group. The policy is optimized with the clipped GRPO objective

$$\mathcal{J}_{\mathrm{GRPO}}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\min\Big(\rho_{i,t}\hat{A}_i, \mathrm{clip}(\rho_{i,t}, 1-\epsilon, 1+\epsilon)\hat{A}_i\Big)\right] - \beta\,\widehat{\mathrm{KL}}(\pi_\theta \| \pi_{\theta_0}), \tag{6}$$

where

$$\rho_{i,t} = \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\mathrm{roll}}(y_{i,t} \mid x, y_{i,<t})}, \tag{7}$$

π_{roll} is the policy that generated the rollout, $\epsilon = 0.2$, and $\beta = 10^{-3}$. We use the low-variance per-token KL estimator $\exp(u) - u - 1$, with $u = \log \pi_{\theta_0} - \log \pi_\theta$. The KL term regularizes the policy toward the initial model and is not included in the reward.

C.3 Optimization and Training Budgets

SFT, SD, and OPD share the optimizer, learning-rate schedule, effective batch size, and 10-epoch training schedule. GRPO uses a method-specific 40-step schedule. Thus, all methods train from the same initial checkpoint and MorphKnow split within each comparison. Methods use their pre-specified, method-specific training schedules; we do not match total compute or question exposure across objectives. All primary runs use seed 23 and report the final checkpoint of the pre-specified schedule; no downstream evaluation is used for checkpoint selection.

The full-split SFT, SD, and OPD runs contain approximately 2,350 optimizer steps. GRPO processes 128 questions per step for 40 steps (5,120 question instances). GLM-4-9B-0414 uses the same settings except for a maximum training length of 768 tokens; token-level methods use two GPUs with gradient accumulation of 2, and GRPO uses two GPUs without warm-up. Models evaluated on IE-EH are trained on the 13,189-item identity-held-out split, using the same method-specific settings (10 epochs for SFT/SD/OPD and 40 steps for GRPO).

C.4 Decoding and Evaluation Protocol

MorphKnow-based evaluations. Qwen3.5-9B uses the non-thinking template with temperature 0.7, top-p 0.8, and top-k 20. GLM-4-9B-0414 uses temperature 0.75 and top-p 0.9. Each AX-HF, IE-SC, IE-SP, and IE-EH item is sampled three times with at most 64 new tokens; Qwen samples use seeds 0, 1, and 2. Missing, unparseable, or out-of-range answers are scored as incorrect. Requests that fail for infrastructure reasons are rerun.

Table 8: Training compute in H100 GPU-hours.

	SFT	SD	OPD	GRPO
Qwen3.5-9B	4.3	5.3	8.3	8.3
GLM-4-9B-0414	4.6	6.0	10.6	6.1

MorphAnalysis evaluations. Each analytical task is run three times independently. The model interacts with the task sandbox through `list_files` and `run_python`; Python execution has a 60-second timeout and tool output is truncated to 8,000 characters. The model may generate up to 2,000 tokens per turn. AA-PS and AA-RD allow up to 15 turns, while AA-CT allows up to 18 turns and 12 tool calls. The main score reads the final valid `"choice"` field, with one final reminder if no choice is produced; a strict-format sensitivity analysis requires a well-formed final JSON response. Invalid final answers are scored as incorrect.

For AA-RD and AA-CT, we additionally record whether each final decision is consistent with the relevant biological fact and with the task-specific numerical evidence. This decomposition is used for the oracle-fact and knowledge–evidence integration analyses in Section 5.2.

C.5 Compute and Implementation

All training runs and Evaluation use NVIDIA H100 GPUs. Table C.4 reports training cost in H100 GPU-hours. SFT, SD, and OPD are implemented with PyTorch 2.13, Transformers 5.15, TRL 1.13, PEFT 0.20, and Accelerate 1.14. GRPO uses verl 0.9.0 with an FSDP actor and vLLM 0.27.1 for rollouts. Evaluation uses vLLM 0.27.1. Unless otherwise noted for targeted intervention studies, each post-training configuration is trained once; evaluation metrics average three independently sampled responses or episodes per item.

D Additional Main Results and Robustness

This appendix provides the complete numerical results underlying Figure 2 and evaluates two additional post-training configurations: reverse-KL OPD and thinking-enabled GRPO. Unless otherwise noted each evaluation item is sampled three times. Reported 95% confidence intervals quantify variation across evaluation items.

D.1 Full Results

Table 9 reports raw accuracy, while Table 10 reports the baseline-normalized accuracy (BNA) used in Figure 2. IE-EH uses models trained on the entity-held-out split. The reverse-KL OPD and thinking-enabled GRPO rows are additional variants and are not part of the main comparison in Figure 2.

The complete results support the pattern summarized in the main text: token-level methods strongly improve Accessibility and IE-SC, while gains are smaller on IE-SP and IE-EH and do not translate into consistent improvements across the three MorphAnalysis families. GRPO shows much weaker knowledge acquisition while remaining closer to the base model on the analytical tasks. The unusual GLM behavior on IE-SP is analyzed further in Appendix E.

D.2 Additional Post-Training Variants

Reverse-KL OPD. We additionally replace the forward-KL objective in OPD with reverse KL while keeping the remaining configuration fixed. Reverse-KL OPD acquires less MorphKnow knowledge than the main forward-KL configuration and does not improve analytical transfer. On Qwen3.5-9B, AX-HF accuracy decreases from 0.990 under forward-KL OPD to 0.920 under reverse-KL OPD, while AA-RD and AA-CT remain below the base model (Table 9). The same qualitative reduction in MorphKnow acquisition is observed on GLM-4-9B-0414.

Table 9: Raw evaluation accuracy. Each value averages three sampled responses per evaluation item. Dashes indicate evaluations not reported for that backbone. Thinking-enabled GRPO is compared with the corresponding thinking-mode base model.

		AX-HF	IE-SC	IE-SP	IE-EH	AA-PS	AA-RD	AA-CT
Qwen3.5-9B	Base model	0.441	0.598	0.460	0.517	0.499	0.538	0.333
	SFT	0.991	0.968	0.669	0.637	0.500	0.429	0.256
	SD	0.990	0.947	0.721	0.625	0.478	0.514	0.319
	OPD	0.990	0.963	0.702	0.616	0.499	0.437	0.360
	GRPO	0.515	0.722	0.500	0.487	0.476	0.545	0.404
	OPD (reverse KL)	0.920	0.907	0.531	0.602	0.497	0.340	0.249
Qwen3.5-9B (thinking)	Base model	0.450	0.608	0.512	0.487	0.490	0.547	0.203
	GRPO	0.472	0.651	0.540	0.479	0.497	0.534	0.386
GLM-4-9B-0414	Base model	0.467	0.554	0.487	0.456	—	—	—
	SFT	0.990	0.945	0.319	0.624	—	—	—
	SD	0.986	0.934	0.223	0.626	—	—	—
	OPD	0.984	0.953	0.181	0.629	—	—	—
	GRPO	0.493	0.661	0.492	0.487	—	—	—
	OPD (reverse KL)	0.873	0.855	0.344	0.606	—	—	—

Table 10: Baseline-normalized accuracy with 95% confidence intervals over evaluation items. The non-thinking Base, SFT, SD, OPD, and GRPO rows are the values summarized in Figure 2.

		AX-HF	IE-SC	IE-SP	IE-EH	AA-PS	AA-RD	AA-CT
Qwen3.5-9B	Base model	-0.08 [-0.14,-0.03]	0.17 [0.11,0.23]	-0.08 [-0.22,0.06]	0.03 [-0.01,0.07]	-0.00 [-0.08,0.07]	0.38 [0.31,0.46]	0.11 [0.06,0.17]
	SFT	0.98 [0.97,0.99]	0.93 [0.91,0.96]	0.34 [0.20,0.48]	0.27 [0.23,0.32]	0.00 [-0.09,0.09]	0.24 [0.17,0.30]	0.01 [-0.04,0.06]
	SD	0.98 [0.97,0.99]	0.89 [0.86,0.92]	0.44 [0.31,0.58]	0.25 [0.21,0.29]	-0.04 [-0.12,0.04]	0.35 [0.29,0.42]	0.09 [0.04,0.15]
	OPD	0.98 [0.97,0.99]	0.92 [0.90,0.95]	0.40 [0.27,0.54]	0.23 [0.19,0.27]	-0.00 [-0.09,0.08]	0.25 [0.19,0.31]	0.15 [0.09,0.20]
	GRPO	0.06 [-0.00,0.12]	0.42 [0.35,0.49]	0.00 [-0.15,0.15]	-0.03 [-0.07,0.02]	-0.05 [-0.12,0.03]	0.39 [0.32,0.47]	0.21 [0.15,0.27]
	OPD (reverse KL)	0.84 [0.81,0.88]	0.81 [0.76,0.85]	0.06 [-0.09,0.21]	0.20 [0.16,0.25]	-0.01 [-0.09,0.08]	0.12 [0.05,0.19]	-0.00 [-0.06,0.06]
Qwen3.5-9B (thinking)	Base model	-0.07 [-0.12,-0.02]	0.19 [0.13,0.24]	0.02 [-0.09,0.14]	-0.03 [-0.06,0.01]	-0.02 [-0.10,0.05]	0.40 [0.32,0.47]	-0.06 [-0.11,-0.02]
	GRPO	-0.03 [-0.08,0.02]	0.28 [0.22,0.33]	0.08 [-0.04,0.19]	-0.04 [-0.08,-0.01]	-0.01 [-0.08,0.07]	0.38 [0.30,0.45]	0.18 [0.12,0.25]
GLM-4-9B-0414	Base model	-0.03 [-0.09,0.02]	0.07 [0.00,0.14]	-0.03 [-0.16,0.11]	-0.09 [-0.13,-0.05]	—	—	—
	SFT	0.98 [0.97,0.99]	0.89 [0.85,0.92]	-0.36 [-0.50,-0.23]	0.25 [0.20,0.29]	—	—	—
	SD	0.97 [0.96,0.99]	0.86 [0.83,0.90]	-0.55 [-0.67,-0.44]	0.25 [0.21,0.29]	—	—	—
	OPD	0.97 [0.96,0.98]	0.90 [0.87,0.93]	-0.64 [-0.75,-0.53]	0.26 [0.22,0.30]	—	—	—
	GRPO	0.01 [-0.04,0.07]	0.30 [0.23,0.37]	-0.02 [-0.16,0.13]	-0.03 [-0.07,0.02]	—	—	—
	OPD (reverse KL)	0.75 [0.71,0.79]	0.70 [0.65,0.75]	-0.31 [-0.45,-0.17]	0.21 [0.17,0.26]	—	—	—

Thinking-enabled GRPO. We also train GRPO with thinking enabled and compare it with the corresponding thinking-mode base model under the same decoding configuration. The resulting changes on MorphKnow evaluations are small, and there is no clear improvement on AA-PS or AA-RD. The apparent gain on AA-CT is largely explained by task completion: the thinking-mode base model produces a final answer for 40% of episodes, compared with 79% for GRPO, while accuracy conditional on producing an answer remains similar (0.50 versus 0.49). We therefore interpret the AA-CT difference primarily as improved completion rather than improved analytical reasoning. Full thinking-mode decoding settings are given in Appendix C.

D.3 Completion Rates on MorphAnalysis

Completion rates are generally high in non-thinking mode, although AA-CT varies somewhat across configurations. In thinking mode, the much larger completion difference on AA-CT explains most of the apparent accuracy gain discussed above.

E Analysis of Knowledge Portability

This appendix provides the analyses supporting Section 5.1 and Section 5.2. We first examine the three Inferential Extensibility settings in more detail, then report the oracle-fact decomposition, training dynamics, behavioral diagnostics, and backbone-validity checks used to interpret Analytical Applicability.

Table 11: Completion rates on MorphAnalysis for Qwen3.5-9B. Values report the fraction of episodes that produce a valid final answer; incomplete episodes are scored as incorrect in the main accuracy results.

		AA-PS	AA-RD	AA-CT
Non-thinking	Base model	0.983	1.000	0.858
	SFT	1.000	0.995	0.974
	SD	0.940	0.946	1.000
	OPD	0.998	0.920	1.000
	GRPO	0.985	0.999	0.946
	OPD (reverse KL)	1.000	1.000	1.000
Thinking	Base model	0.987	0.999	0.404
	GRPO	0.997	0.999	0.789

Table 12: IE-SC accuracy relative to a global answer-frequency heuristic. The heuristic chooses the candidate with the higher answer rate in post-training and decides 662 of 722 items. Accuracy first averages the three samples of each item and then averages across items. Δ is the paired difference from the corresponding base model on conflict items, with a 90% cluster-robust interval.

		Agree	Tie	Conflict	Δ on conflict
	Items	563	60	99	
	Best constant	0.529	0.517	0.545	
Qwen3.5-9B	Base model	0.628	0.478	0.505	—
	SFT	0.978	0.878	0.966	$+0.461$ [$+0.395$, $+0.527$]
	SD	0.967	0.811	0.912	$+0.407$ [$+0.332$, $+0.483$]
	OPD	0.984	0.839	0.916	$+0.411$ [$+0.342$, $+0.479$]
	GRPO	0.729	0.606	0.747	$+0.242$ [$+0.177$, $+0.308$]
GLM-4-9B	Base model	0.568	0.500	0.505	—
	SFT	0.964	0.828	0.906	$+0.401$ [$+0.241$, $+0.560$]
	SD	0.950	0.828	0.906	$+0.401$ [$+0.245$, $+0.557$]
	OPD	0.975	0.856	0.889	$+0.384$ [$+0.194$, $+0.573$]
	GRPO	0.677	0.539	0.643	$+0.138$ [$+0.068$, $+0.208$]

E.1 Inferential Extensibility

E.1.1 IE-SC: construction and answer-frequency controls

Each IE-SC item is constructed from a three-term ordering $A > Z > B$ for one measured quantity in a fixed experimental context and asks for the unseen comparison between A and B. The endpoints never co-occur in a post-training item. However, A appears in a supporting comparison that it wins and B in one that it loses, so the endpoint identities themselves carry sufficient directional information. Thus, IE-SC tests generalization to a previously unseen comparison but does not require explicit resolution of the intermediate entity.

We additionally test whether a simpler answer-frequency heuristic can explain the high IE-SC accuracy. The heuristic chooses the candidate with the higher empirical answer rate in the post-training set, using empirical-Bayes shrinkage toward 0.5. Table 12 separates items on which this heuristic agrees with, ties, or contradicts the gold answer. On the 99 conflict items, the token-level methods remain far above both the base model and the best constant answer. Removing the item's own supporting facts from the frequency counts produces the same qualitative conclusion. These controls rule out a simple global answer-frequency explanation, while not identifying the internal strategy used by the model.

E.1.2 IE-SP: matched-pair response patterns

The 160 IE-SP items form 80 matched pairs. The two items in a pair contain the same perturbations and stain-channel options but reverse the selector (stronger versus weaker overall response), so the correct channels also reverse. Under independent random answering, a pair is answered correctly on

Table 13: IE-SP response patterns over 80 matched pairs. Each pair is classified separately for each of the three samples. "Intermediate" means that at least one response returns the selected perturbation rather than the requested property.

		Both correct	Same channel	Both reversed	Intermediate	Other
Random answering		0.250	0.500	0.250	0	0
Qwen3.5-9B	Base model	0.146	0.617	0.217	0.021	0.000
	SFT	0.421	0.425	0.029	0.121	0.004
	SD	0.442	0.558	0.000	0.000	0.000
	OPD	0.438	0.529	0.033	0.000	0.000
	GRPO	0.113	0.775	0.113	0.000	0.000
GLM-4-9B	Base model	0.179	0.617	0.200	0.000	0.004
	SFT	0.071	0.388	0.037	0.450	0.054
	SD	0.025	0.283	0.008	0.621	0.062
	OPD	0.050	0.192	0.004	0.688	0.067
	GRPO	0.037	0.908	0.054	0.000	0.000

both items with probability 0.25, reversed on both with probability 0.25, and answered with the same channel with probability 0.50. We classify each of the three sampled response pairs accordingly.

Table 13 shows that Qwen token-level methods solve both items in 42–44% of matched pairs, while the base model most often returns the same channel for both selectors. GLM token-level methods exhibit a different failure mode: they frequently return the selected perturbation rather than the requested channel. This format error accounts for much of their poor strict IE-SP score; among legal channel answers, their accuracy is approximately 0.53–0.55 compared with 0.49 for the base model.

E.1.3 IE-EH: CANDIDATE NOVELTY

To examine entity-level OOD behavior, we focus on the 1,030 two-candidate IE-EH items for which at least one candidate is absent from the entity-held-out post-training split. Table 14 separates cases in which the correct answer is the seen candidate, the unseen candidate, or both candidates are unseen. Gains are largest when the correct candidate was seen during post-training, consistent with a learned preference for previously seen entities. However, positive gains remain when both candidates are unseen for the token-level methods on both backbones. Because the Qwen base model is substantially below chance on the both-unseen subset, these residual gains should not be interpreted as direct evidence that the model acquired new entity-specific knowledge.

As a complementary check, on items with one seen and one unseen candidate we model the probability of selecting the seen candidate as a function of its training answer rate while controlling for whether the seen candidate is correct, exposure, answer position, and dataset. For token-level methods, higher training answer rate is associated with a 0.10–0.14 increase in selecting the seen entity per standard deviation, while the coefficient indicating that the seen entity is actually correct also increases. The two effects therefore coexist: post-training introduces a seen-entity preference while also improving discrimination on some items.

E.2 ANALYTICAL APPLICABILITY

E.2.1 ORACLE-FACT DECOMPOSITION

AA-RD and AA-CT permit each answer to be decomposed into agreement with the relevant MorphKnow fact and agreement with the task evidence. In AA-RD, a choice is knowledge-consistent when it retains the channel identified by the fact and data-consistent when it excludes the block that fails quality control. In AA-CT, a choice is knowledge-consistent when its predicted direction agrees with the fact and data-consistent when it excludes the failed source plate. Random answering gives 0.5 for each component.

Table 14: IE-EH by candidate novelty. Entries are accuracy, with the difference from the corresponding base model in parentheses. All models are trained on the entity-held-out split.

		Answer seen ($n = 539$)	Answer new ($n = 391$)	Both new ($n = 100$)
Qwen3.5-9B	Base model	0.416	0.517	0.350
	SFT	0.783 (+0.367)	0.572 (+0.055)	0.657 (+0.307)
	SD	0.771 (+0.355)	0.590 (+0.072)	0.673 (+0.323)
	OPD	0.774 (+0.357)	0.565 (+0.048)	0.630 (+0.280)
	GRPO	0.535 (+0.119)	0.522 (+0.004)	0.720 (+0.370)
GLM-4-9B	Base model	0.415	0.473	0.483
	SFT	0.746 (+0.331)	0.563 (+0.090)	0.610 (+0.127)
	SD	0.763 (+0.348)	0.557 (+0.084)	0.577 (+0.093)
	OPD	0.758 (+0.343)	0.552 (+0.078)	0.600 (+0.117)
	GRPO	0.389 (−0.026)	0.457 (−0.016)	0.463 (−0.020)

Table 15: Qwen3.5-9B under the oracle-fact condition. Knowledge- and data-consistent choice rates are computed over answered episodes. AA-PS has no corresponding decomposition.

	AA-PS	AA-RD			AA-CT		
	Acc.	Acc.	Knowledge	Data	Acc.	Knowledge	Data
Base model	0.982	0.995	0.995	1.000	0.908	0.977	0.928
SFT	0.537	0.563	0.989	0.571	0.489	0.916	0.550
SD	0.494	0.605	0.999	0.644	0.582	0.958	0.604
OPD	0.557	0.646	1.000	0.675	0.490	0.856	0.571
GRPO	0.983	0.993	0.999	0.995	0.971	0.992	0.975
OPD (reverse KL)	0.513	0.514	0.953	0.552	0.431	0.842	0.511

Table 15 reports this decomposition when the correct MorphKnow fact is supplied directly in the prompt. SFT, SD, and OPD remain strongly knowledge-consistent but are substantially less data-consistent than the base model. GRPO remains close to the base model on both components.

E.2.2 Response behavior and evidence access

Knowledge post-training also changes the observable execution pattern on MorphAnalysis.vTable 16 shows that the token-level methods generally produce shorter responses and fewer tool calls than the base model, especially on AA-PS and AA-RD.

The oracle-fact AA-RD transcripts provide a more direct evidence-access check (Table 17). The QC table appears in tool outputs for 94–100% of episodes across the token-level methods, yet their final choices are data-consistent in only 57–69% of answered episodes, compared with 100% for the base model. Thus, their lower data consistency cannot be explained simply by failure to open or read the relevant file.

E.2.3 Training dynamics

Figure 8 and Table 18 extend the training-trajectory analysis in Figure 5. SFT, SD, and OPD acquire MorphKnow gradually, but their analytical behavior changes much earlier. On AA-CT, data-consistent choices drop within the first half epoch for all three forward-supervised methods, when only a small fraction of the eventual AX-HF gain has been acquired. On AA-RD, SFT shows an early decline in data consistency, while SD and OPD more often change their completion behavior before later recovering some data use. GRPO acquires much less MorphKnow knowledge and shows no analogous decline over its shorter training trajectory.

Because data consistency is defined over answered episodes in the main analysis, we separately audit the effect of unanswered episodes. Completion and choice quality can move in different directions early in training; Table 19 therefore reports alternative scoring rules for SD and OPD at 0.5 epoch. Our main claim concerns the quality of decisions conditional on the model producing an answer, rather than completion rate itself.

Table 16: Response behavior without supplied facts on Qwen3.5-9B. Reported values are median completion tokens per episode and mean tool calls per episode.

	AA-PS		AA-RD		AA-CT	
	Tokens	Tool calls	Tokens	Tool calls	Tokens	Tool calls
Base model	1,824	4.78	1,137	3.47	7,052	7.56
SFT	8	0.05	121	2.09	958	3.67
SD	8	1.67	546	3.44	3,564	6.47
OPD	8	0.05	636	3.79	5,742	7.00
GRPO	1,570	4.33	1,135	3.65	6,504	7.16

Table 17: Evidence access in oracle-fact AA-RD transcripts (Qwen3.5-9B). Statistics use the first sample of each of the 247 tasks. "Read QC" indicates that the QC fields appear in a tool output. "Data-consistent" is computed over answered episodes.

	Read QC	Wrote analysis	Median assistant chars.	Data-consistent
Base model	1.000	0.968	1,353	1.000
SFT	0.992	0.000	15	0.575
SD	0.996	0.445	15	0.613
OPD	0.943	0.607	1,049	0.694
GRPO	1.000	0.939	1,396	0.988

E.2.4 QC-DECISION PROBES

We additionally run single-turn AA-RD probes that remove tool-use and multi-turn execution as possible confounds. With the correct fact supplied, token-level models more often conclude that no block fails quality control, despite largely reporting the underlying QC values correctly. Amplifying the anomaly reduces this behavior, and when the supplied fact conflicts with the data, the token-level models more often follow the fact than the base model. The effect is also sensitive to wording: on GLM-4-9B-0414, presenting the same information as a factual statement reduces data-consistent choices more than presenting it as a requirement for the current run.

These probes cover one task family and one protocol. We therefore treat them as behavioral evidence that post-training changes how prior biological information is weighted against current evidence, rather than as identification of an internal mechanism.

E.3 MORPHANALYSIS EVALUATION ON GLM-4-9B

We report the main multi-turn MorphAnalysis results on Qwen3.5-9B. Under the same agent harness, GLM-4-9B-0414 shows limited tool engagement on AA-PS and AA-CT, while tool use is more consistent on AA-RD (Table 20). In contrast, Qwen3.5-9B consistently accesses the task data across all three families and reaches near-ceiling performance when the correct facts are supplied. We therefore restrict the primary multi-turn Analytical Applicability analysis to Qwen3.5-9B, and use GLM only for the single-turn diagnostic probes.

F SUPPORTING EXPERIMENTS FOR USABLE SCIENTIFIC PRIORS

This appendix provides the full design and supporting results for Section 6. All experiments use Qwen3.5-9B with LoRA SFT and one training seed per replay recipe. We first describe the paired fact-split design and the two replay settings, then report the manipulation check, fact-specific downstream effects, prompted recall probes, and the intervention that explicitly recalls the relevant facts before analysis.

F.1 PAIRED FACT-SPLIT DESIGN AND REPLAY CONSTRUCTION

Paired fact allocation. Each AA-RD and AA-CT item is linked to a specific set of supporting MorphKnow facts. To prevent direct restatements of a held-out fact from remaining in training,

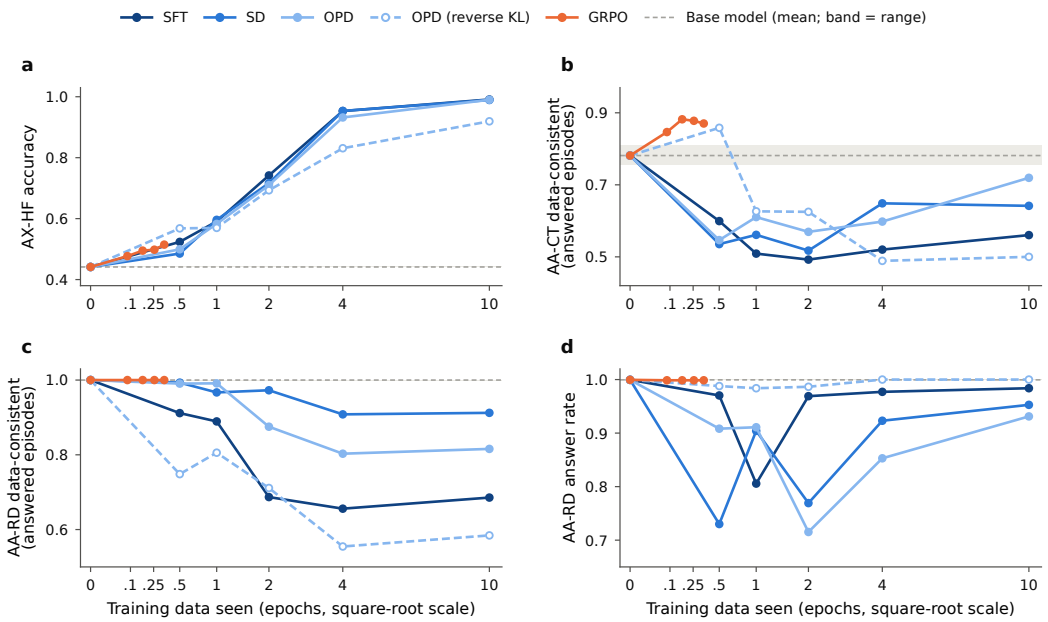

Figure 8: Training trajectories of all post-training methods (Qwen3.5-9B, no facts supplied). The horizontal axis shows MorphKnow data seen in epochs on a square-root scale; a GRPO step processes 128 questions. Epoch 0 is the base model. **(a)** AX-HF accuracy. **(b,c)** Data-consistent choices on AA-CT and AA-RD over answered episodes. **(d)** Fraction of AA-RD episodes that end with a valid answer.

Table 18: Training trajectories on Qwen3.5-9B without supplied facts. Data-consistent rates are computed over answered episodes. Epoch 0 gives the range across the repeated base-model evaluations used for the trajectory batches.

	Epochs	AX-HF	AA-PS acc.	AA-RD acc.	AA-RD answered	AA-RD data	AA-CT acc.	AA-CT data
Base model	0	0.441	0.478–0.503	0.537–0.570	0.999–1.000	1.000	0.300–0.336	0.756–0.810
SFT	0.5	0.524	0.488	0.475	0.970	0.912	0.275	0.599
	1	0.591	0.508	0.371	0.806	0.889	0.239	0.509
	2	0.742	0.497	0.321	0.969	0.687	0.251	0.492
	4	0.953	0.508	0.363	0.977	0.656	0.249	0.520
	10	0.991	0.501	0.410	0.984	0.686	0.281	0.560
SD	0.5	0.485	0.083	0.424	0.730	0.993	0.253	0.535
	1	0.596	0.358	0.436	0.904	0.967	0.275	0.561
	2	0.716	0.326	0.387	0.769	0.973	0.274	0.517
	4	0.953	0.468	0.516	0.923	0.908	0.306	0.649
	10	0.989	0.478	0.493	0.953	0.912	0.322	0.642
OPD	0.5	0.499	0.224	0.491	0.908	0.991	0.261	0.547
	1	0.583	0.343	0.486	0.911	0.991	0.294	0.610
	2	0.710	0.402	0.347	0.715	0.875	0.290	0.569
	4	0.932	0.451	0.416	0.853	0.803	0.314	0.598
	10	0.990	0.501	0.460	0.931	0.816	0.357	0.719
OPD (reverse KL)	0.5	0.568	0.501	0.366	0.988	0.748	0.426	0.858
	1	0.570	0.495	0.391	0.984	0.806	0.318	0.626
	2	0.693	0.460	0.332	0.987	0.711	0.315	0.625
	4	0.831	0.501	0.305	1.000	0.555	0.250	0.489
	10	0.919	0.497	0.333	1.000	0.584	0.261	0.500
GRPO	0.09	0.477	0.485	0.552	0.999	1.000	0.381	0.847
	0.17	0.495	0.501	0.556	0.999	1.000	0.426	0.882
	0.26	0.498	0.503	0.534	0.999	1.000	0.404	0.878
	0.34	0.515	0.492	0.543	0.999	1.000	0.383	0.870

we construct a *relation package* for each item containing its supporting facts, MorphKnow items equivalent to or entailed by them, and training items that restate those facts in their premise or answer. Overlapping packages are grouped so that dependent facts are assigned together.

Table 19: Sensitivity of the early AA-CT result to scoring unanswered episodes. Values are percentage-point differences from the same-batch base model at 0.5 epoch. The main analysis uses the "answered episodes" column.

	Answered episodes	All episodes, lenient	All episodes, strict
SD	-25.5	-17.2	$+19.9$
OPD	-22.5	-11.3	$+21.8$

Table 20: Validity audit of the multi-turn MorphAnalysis tasks for the base models. Entries report the fraction of episodes with at least one tool call / accuracy, without and with the correct facts supplied.

	AA-PS		AA-RD		AA-CT	
	No facts	Facts	No facts	Facts	No facts	Facts
GLM-4-9B-0414	0.41 / 0.494	0.33 / 0.507	0.98 / 0.291	0.74 / 0.509	0.00 / 0.278	0.00 / 0.329
Qwen3.5-9B	1.00 / 0.507	1.00 / 0.983	1.00 / 0.538	1.00 / 0.995	1.00 / 0.388	1.00 / 0.915

The resulting packages are divided into two complementary halves. We then train two knowledge models with identical replay data: model 1 is trained on the facts assigned to half 1 while holding out the packages assigned to half 2, and model 2 uses the opposite allocation. For a downstream item supported by fact f, this creates a matched comparison between a model trained on f and a model for which f was held out.

Let $Y_{m,i}$ denote the performance of model $m \in \{1, 2\}$ on item i, averaged over three sampled responses, and define

$$d_i = Y_{1,i} - Y_{2,i}.$$

For task family f, the fact-specific effect is

$$\tau_f = \frac{1}{2}\left(\bar{d}_{f,\text{half 1}} - \bar{d}_{f,\text{half 2}}\right).$$

A model-level performance offset affects both terms equally and therefore cancels. The aggregate effect reported in the main text is the average of the AA-RD and AA-CT effects. Confidence intervals use CR2 cluster-robust standard errors with Satterthwaite degrees of freedom over the final relation-package clusters.

Generic replay. To preserve general multi-turn analysis behavior without rehearsing MorphAnalysis, we generate 3,000 non-biological data-analysis problems over synthetic tables. The problems require operations such as aggregation, joining multiple files, filtering, unit conversion, and threshold comparisons, and are solved by the base model using the same `list_files` and `run_python` tools as MorphAnalysis. Trajectories are retained when they complete with a valid structured answer, regardless of correctness, because replay is intended to preserve the base model's analysis behavior rather than provide additional task supervision. This yields 2,845 trajectories. The prompts and trajectories are screened for MorphKnow entity names and Cell Painting terminology.

Task replay. For task replay, the base model solves a replay subset of MorphAnalysis four times under the standard evaluation protocol. Replay/evaluation overlap is removed at the item and visible-entity level, leaving 1,257 valid replay trajectories from AA-PS and AA-RD. The corresponding evaluation uses 118 held-out AA-RD items and all 240 AA-CT items.

Training. Every assistant turn in a replay trajectory is supervised, including reasoning, tool calls, and the final answer; system messages, user messages, and tool outputs are not. Knowledge and replay receive equal total loss weight,

$$\mathcal{L} = \frac{1}{2}\mathcal{L}_{\text{know}} + \frac{1}{2}\mathcal{L}_{\text{replay}}, \qquad \mathcal{L}_s = \frac{1}{N_s}\sum_{t \in s} \ell_t.$$

All remaining optimization settings follow Appendix C.

Table 21: Manipulation check on the supporting facts. Each knowledge model is evaluated on facts included in its post-training set and on matched facts held out through relation packages. Differences are averaged over the two fact halves and the two task families; intervals are 90% cluster-robust intervals.

		AA-RD facts		AA-CT facts	
Replay	Model	Trained	Held out	Trained	Held out
Generic replay	Knowledge half 1	1.000	0.645	1.000	0.591
	Knowledge half 2	1.000	0.758	1.000	0.583
	Trained − held out	+34.6 [+30.8, +38.3] points			
	Held out − base model	+16.2 [+11.1, +21.3] points			
Task replay	Knowledge half 1	1.000	0.540	1.000	0.652
	Knowledge half 2	1.000	0.653	1.000	0.556
	Trained − held out	+37.4 [+33.2, +41.6] points			
	Held out − base model	+12.9 [+7.4, +18.4] points			

Table 22: Fact-specific downstream accuracy effects without supplied facts. The estimate compares performance on the same downstream items when their supporting facts were trained versus held out. Upper bounds are one-sided 95% bounds from the cluster-robust analysis.

Replay	Scoring	Effect (pp)	One-sided 95% upper bound (pp)
Generic replay	Lenient	−0.09	+3.17
	Strict	−0.49	+2.05
Task replay	Lenient	+0.03	+2.19
	Strict	+0.83	+2.85

F.2 Manipulation Check and Fact-Specific Transfer

The paired design first requires that the trained and held-out fact sets are actually separated by post-training. Table 21 confirms this manipulation: the knowledge models answer their trained supporting facts correctly under candidate scoring, while performance on matched held-out facts is substantially lower. Averaged over halves and task families, the trained-versus-held-out difference is 34.6 percentage points under generic replay and 37.4 points under task replay. Held-out facts remain somewhat above the base model because relation packages remove only equivalent or entailed restatements, while related facts about the same entities under other conditions remain available.

Despite this factual separation, the corresponding downstream accuracy difference is near zero. Table 22 reports the prespecified fact-specific contrast without supplying the facts at inference time. The one-sided 95% upper bounds are only +3.17 points under generic replay and +2.19 points under task replay, matching the values summarized in Section 6. Requiring a well-formed final answer produces the same qualitative conclusion.

Figure 9 gives the same contrast separately for AA-RD and AA-CT, for both analytical accuracy and fact-consistent choices. The wider AA-CT intervals reflect the smaller effective number of relation-package clusters.

F.3 Prompted Recall of Learned Facts

We next test whether the trained-versus-held-out fact advantage remains accessible under prompts that progressively reduce how directly the relevant fact is requested. The probes use the four generic-replay models without tools and the standard evaluation decoding settings. Direct fact queries ask the original MorphKnow question; subsequent conditions place the same query after increasing amounts of task context. The open condition instead asks what the model knows about the compound and extracts the channel named as most affected from the generated text.

Table 23 shows a clear decline in the trained-fact advantage as the prompt becomes less explicit. For AA-RD supporting facts, the advantage is 24.0 points when the fact is directly requested, 15.6 points with task context, 12.7 points with the full task prompt, and approximately zero under the

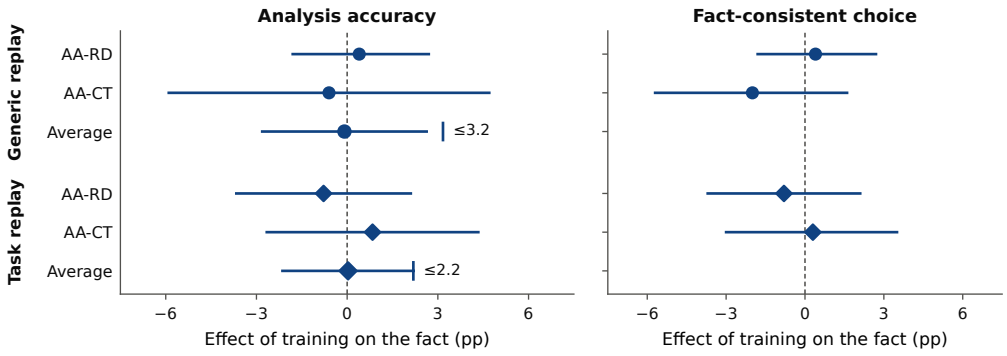

Figure 9: Effect of training the relevant fact, by task family (Qwen3.5-9B; no facts supplied). Left: analytical accuracy. Right: fact-consistent choices, counting unanswered episodes as not fact-consistent. Circles denote generic replay and diamonds task replay; error bars show 90% cluster-robust intervals.

Table 23: Prompted recall of supporting facts under progressively less explicit prompts. "Trained" is the accuracy of the knowledge models on facts included in post-training; the difference compares those facts with matched held-out facts. Intervals are 90% cluster-robust intervals.

	AA-RD facts		AA-CT facts	
Prompt	Trained	Trained − held out	Trained	Trained − held out
Direct fact query	0.918	+24.0 [+19.9, +28.0]	0.760	+19.6 [+10.3, +29.0]
+ task context	0.815	+15.6 [+11.5, +19.7]	0.662	+8.4 [+2.8, +14.0]
Full task prompt	0.789	+12.7 [+8.7, +16.7]	0.658	+9.7 [+2.7, +16.7]
Open prompt	—	+0.1 [−2.9, +3.1]	—	—

open prompt. The same directional pattern appears for the AA-CT facts in the three structured conditions. A blinded audit of the open-prompt extraction found the automatic scoring broadly consistent with the generated text.

F.4 Explicit Recall Before Analysis

Finally, we test whether making the supporting facts explicit immediately before analysis increases their downstream influence. Each AA-RD item is solved under three conditions with the same analysis budget. *No preparation* uses the standard protocol. *Generic preparation* asks the model to describe a general strategy for choosing a configuration from the QC report without discussing the compound. *Explicit fact recall* asks the model, without tools, to answer the MorphKnow questions for the supporting facts before proceeding to the analysis. The generic-preparation condition controls for the presence of an additional pre-analysis turn without explicitly cueing the biological knowledge.

Figure 10 shows the model-level behavior across the three conditions, while Table 24 reports the paired trained-versus-held-out effects for the prespecified no-preparation versus explicit-recall contrast. Under generic replay, explicit recall increases the fact-specific shift toward fact-consistent choices from -0.6 to $+5.0$ percentage points, a change of $+5.6$ points, but the corresponding change in accuracy is not established. Under task replay, the direction is similar but less precise.

For the task-replay preparation diagnostic, tool definitions are omitted during the recall-only preparation turn because tool execution is disabled in that turn. We report the resulting protocol in Figure 10 and Table 24. Taken together, these experiments show that the trained facts are genuinely acquired, remain accessible under targeted prompting, and can be made more influential in the final decision. None of these manipulations, however, establishes a corresponding fact-specific improvement in analytical accuracy.

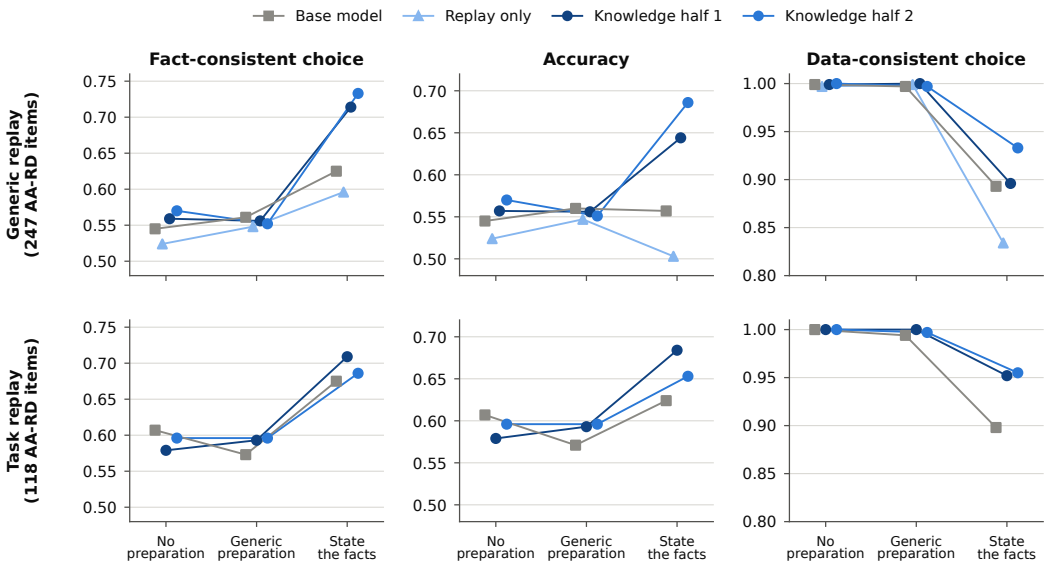

Figure 10: AA-RD behavior under three preparation conditions (Qwen3.5-9B). Top: generic replay (247 items). Bottom: task replay with the repaired preparation protocol (118 items). The knowledge-model levels average over items whose supporting facts were trained and held out; Table 24 reports their paired difference.

Table 24: Effect of explicit fact recall before AA-RD. Recall is the fraction of supporting facts stated correctly in the preparation turn. τ_{none} and τ_{recall} are the fact-specific effects without preparation and after explicit recall, respectively; D is their difference. Intervals are 90% cluster-robust intervals.

		Recall in preparation turn		Fact-consistent choice		
Replay	Protocol	Trained facts	Held-out facts	τ_{none}	τ_{recall}	D
Generic replay	Main	0.807 / 0.763	0.662 / 0.687	−0.6 [−3.1, +1.9]	+5.0 [+1.6, +8.5]	+5.6 [+1.7, +9.6]
Task replay	Repaired	0.786 / 0.726	0.596 / 0.685	+2.7 [−1.4, +6.9]	+5.8 [+0.9, +10.6]	+3.0 [−3.0, +9.1]
				Accuracy		
Generic replay	Main			−0.5 [−3.0, +2.0]	+2.8 [−0.8, +6.4]	+3.3 [−0.8, +7.4]
Task replay	Repaired			+2.7 [−1.4, +6.9]	+5.5 [+0.3, +10.8]	+2.8 [−3.7, +9.2]