# RUBRICOPT: RECURSIVE REFINEMENT OF SEMANTIC RUBRICS FOR PREFERENCE OPTIMIZATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Pairwise preference labels specify which response is preferred but leave the behavioral basis of that preference implicit. We introduce RubricOpt, a framework that makes a semantic rubric an explicit, editable specification for preference optimization. RubricOpt measures semantic demand—how behaviors separate task success from failure—and semantic supply—how often preference pairs contrast those behaviors—in a shared eight-factor space. Semantic Rubric Steering uses this diagnosis to condition training on a short natural-language rubric while keeping the response pairs, labels, and preference objective fixed. The rubric is gradually withdrawn before training ends, and all evaluation is rubric-free. A gated refinement loop revises clauses when recurring failures reveal requirements missing from the current wording. Our analysis gives sufficient conditions for criterion-free behavioral improvement and for retaining success-probability gains after withdrawal. Experiments use two preference corpora and three models from two families, each compared with a matched ordinary-DPO control. On Tulu3, steering improves BBH by 3.80–8.35 percentage points and the seven-benchmark average by 1.06–2.96 points across Qwen3-4B, Qwen3-8B, and Mistral-7B. On HelpSteer3, four rubric compositions improve Qwen3-4B Arena-Hard win rates by 4.1–7.1 points, without a material increase in response length. Matched HelpSteer3 continuation on Qwen3-4B adds 8.15 BBH points under the unchanged rubric, while the tested wording edit yields no detectable additional BBH gain, separating the benefit of continued training from the effect of refinement.

## 1 INTRODUCTION

Preference optimization learns from pairs of candidate responses: a label says which response is preferred, but does not decompose how reasoning, factual grounding, completeness, and presentation contribute to that preference. Direct Preference Optimization (DPO) turns this comparison directly into a policy update (Rafailov et al., 2023). Later objectives refine the loss and its reference: IPO replaces the pairwise objective with a squared loss on the margin (Azar et al., 2024), KTO drops the pairing requirement altogether (Ethayarajh et al., 2024), and SimPO removes the reference model and normalises by length (Meng et al., 2024). A complementary line localizes credit inside a response, reweighting tokens by a forward-KL penalty (Zeng et al., 2024), by importance sampling against an estimated weight (Liu et al., 2025), by a token-level reward used to select tokens (Yang et al., 2025b), by a token-level reward used as a regulariser (Zhou et al., 2025), or by token importance (Yang et al., 2026). The pair-level behavioral distinction that training should emphasize, however, remains implicit.

That compression matters because a preferred response can win for several reasons at once, while the optimizer receives one binary ordering. Preference optimization can overfit an imperfect proxy: reward-model scores degrade as optimisation pressure grows, both with a separate reward model (Gao et al., 2023) and inside direct alignment algorithms (Rafailov et al., 2024), and a proxy can be satisfied without the intended behaviour (Skalse et al., 2022). It can instead reward length, which correlates with RLHF reward (Singhal et al., 2024) and with the scores automatic judges assign (Zheng et al., 2023), or improve agreement with an annotator without improving correctness (Sharma et al., 2024). Human labels are not a gold standard either (Hosking et al., 2024), and the collection pipeline has structural limitations (Casper et al., 2023). These failures have differ-



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.

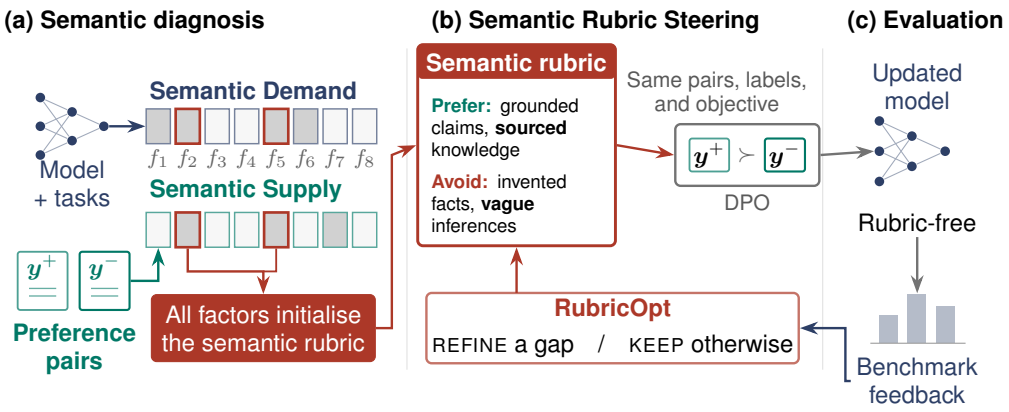



Figure 1: The RubricOpt loop. (a) A model and its preference data are placed in one shared factor space, which measures *semantic demand* (how strongly a factor separates success from failure on the task) and *semantic supply* (how often the preference pairs contrast that factor). The two highlighted factors are the ones the diagnosis flags as in demand but rarely contrasted. (b) All eight factors initialise the semantic rubric, which then conditions DPO while the pairs, labels, and objective stay fixed. (c) The rubric is withdrawn before evaluation, so every reported score is rubric-free; item-level benchmark failures feed a refinement operator that either restates one clause (REFINE) or leaves it as it stands (KEEP), which is how the same ontology ends up weighing factors differently across models and benchmarks.

ent behavioral causes, but the pair label does not say which distinction a particular update should represent.

Fine-grained evaluation begins to expose those distinctions. FLASK scores responses along interpretable skill dimensions (Ye et al., 2024), EvalTree profiles weaknesses as a hierarchical capability tree (Zeng et al., 2025), and ErrorMap and ErrorAtlas chart failures into error categories (Ashury-Tahan et al., 2026). Diagnosis-driven post-training then uses those failures to change data or supervision: DiagEvo feeds diagnosed errors back into self-evolution (Wei et al., 2026), and APTER turns expert-grounded rubrics into post-training supervision (Wang et al., 2026). Each of these lines stops at the same boundary, and the boundary has three faces. A behavioural diagnosis describes the model, not what its preference data actually contrasts; a criterion that improves a model does not say which behavioural distinction carried the improvement, or whether that distinction survives once the criterion is removed.

RubricOpt addresses all three. It places a model and its preference data in one eight-factor coordinate system, so that task demand and preference-data supply are measured on the same axes. It conditions the preference pair on a short natural-language rubric during training and withdraws that rubric before evaluation, leaving the pairs, labels, and objective untouched. And it refines the rubric recursively from item-level failures, admitting an edit only when a deterministic gate passes it.

The measurements follow the same three-part structure. Demand and supply disagree on exactly the factors the downstream benchmarks need most. Conditioning is worth 8.35 BBH points and 1.82 average points on the comprehensive-capability suite when the Tulu3 corpus supplies the pairs, and 4.1–7.1 Arena-Hard points on the open-ended generative suite when HelpSteer3 does, without a material increase in response length; a factor-set ablation moves all four compositions within 0.40 average points, so no single group carries that effect. And the loop's own control separates its two outputs: another epoch under the unchanged rubric adds 8.15 BBH points, while the tested wording edit adds none, which is why an edit has to be evaluated apart from continued training. The pattern holds across different preference corpora, model families, and model scales.

The contributions are (i) a semantic account of what preference optimisation is asked to learn, by placing a model and its preference data in one eight-factor coordinate system; (ii) training-only semantic rubric steering, which leaves every reported score rubric-free; and (iii) a gated refinement loop with a matched continuation control separating an edit from further training.

## 2 Related Work and Problem Setup

**Preference optimization and credit assignment.** DPO fits chosen–rejected pairs with a classification objective (Rafailov et al., 2023). Later objectives change one component of that recipe: IPO replaces it with a squared loss on the pairwise margin (Azar et al., 2024), KTO drops the pairing requirement by scoring each response against a reference point (Ethayarajh et al., 2024), and SimPO removes the reference model and normalises the margin by length (Meng et al., 2024). Two analyses place the family in a wider frame: one connects these objectives to imitation learning (Xiao et al., 2025), and one disentangles how their update suppresses the rejected response and preserves the chosen one (Chen et al., 2026). A complementary line assigns credit inside a response rather than to the response as a whole: TDPO reweights tokens by a forward-KL penalty (Zeng et al., 2024), TIS-DPO reweights them by importance sampling against an estimated weight (Liu et al., 2025), SEPO selects tokens using a token-level reward estimate (Yang et al., 2025b), T-REG uses a token-level reward as a regulariser (Zhou et al., 2025), and TIDPO guides the update by token importance (Yang et al., 2026). RubricOpt operates at a different level: a semantic factor names a behavioral distinction that can span several tokens or reasoning steps, while the pairwise loss itself is left unchanged.

**Criteria, diagnosis, and recursive feedback.** Written criteria already enter learning through several interfaces. As reward signals, Rubrics as Rewards scores on-policy responses against a rubric (Gunjal et al., 2025); as data, Configurable Preference Tuning builds rubric-conditioned synthetic preference pairs (Gallego, 2025); and as weights, CoRT contrasts rubric-conditioned with criterion-free replays to form token-level weights (Zhang et al., 2026). Criterion-conditioned training also has close relatives outside preference optimisation: learning using privileged information trains on a signal that is absent at test time (Vapnik & Vashist, 2009); context distillation trains on a context and then evaluates without it, introduced as an alignment technique (Askell et al., 2021) and developed as a standalone method (Snell et al., 2022); and classifier-free guidance interpolates between conditioned and unconditioned predictions (Ho & Salimans, 2022). On the diagnostic side, fine-grained evaluation describes response quality through interpretable skill sets and error taxonomies—FLASK by grading along skill dimensions (Ye et al., 2024), EvalTree by profiling weaknesses as a hierarchical capability tree (Zeng et al., 2025), and ErrorMap and ErrorAtlas by charting the failure landscape into error categories (Ashury-Tahan et al., 2026)—and diagnosis-driven post-training then acts on what those tools find, with DiagEvo feeding diagnosed errors back into self-evolution through a hierarchical error memory (Wei et al., 2026) and APTER turning expert-grounded rubrics into post-training supervision (Wang et al., 2026). On the recursive side, Self-Refine edits outputs from self-feedback (Madaan et al., 2023), Self-Rewarding has the model generate its own reward signals (Yuan et al., 2024), Meta-Rewarding adds a meta-judge layer (Wu et al., 2025a), self-play preference optimisation regenerates the preferences themselves (Wu et al., 2025b), and automatic prompt optimisation edits inference-time text from feedback (Pryzant et al., 2023). RubricOpt differs from each of these in the same way: it edits a training-only specification rather than the responses, the rewards, or the pairs; it measures the preference data on the same axes as the task; and it accepts an edit only through a deterministic gate, then compares that edit with a matched unchanged-rubric continuation rather than assuming the edit caused the change. Because LLM judges exhibit position, verbosity, and self-enhancement biases (Zheng et al., 2023), the gate checks evidence references and textual invariants; independent judge checks are in appendix B.3.

### 2.1 Problem Setup

We study preference optimization on a fixed preference corpus $\mathcal{D} = \{(\boldsymbol{x}, \boldsymbol{y}^+, \boldsymbol{y}^-)\}$, where $\boldsymbol{x}$ is a prompt and $\boldsymbol{y}^+$ and $\boldsymbol{y}^-$ are the chosen and rejected responses. A model $m$ with a trainable policy $\pi_\theta$ and a frozen reference $\pi_{\text{ref}}$ is trained on $\mathcal{D}$ with Direct Preference Optimization (Rafailov et al., 2023), which fits each pair with

$$\mathcal{L} = -\mathbb{E}\left[\log \sigma\left(\beta\left[\log \frac{\pi_\theta(\boldsymbol{y}^+ \mid \boldsymbol{x})}{\pi_{\text{ref}}(\boldsymbol{y}^+ \mid \boldsymbol{x})} - \log \frac{\pi_\theta(\boldsymbol{y}^- \mid \boldsymbol{x})}{\pi_{\text{ref}}(\boldsymbol{y}^- \mid \boldsymbol{x})}\right]\right)\right], \tag{1}$$

where $\beta$ is the DPO temperature and the expectation runs over the corpus. The pairs, their labels, and this objective are held fixed throughout the paper; only the text that conditions a pair may change, which is the degree of freedom RubricOpt introduces.

Behavior is described in a fixed ontology of eight factors, $\mathcal{F} = \{f_1, \ldots, f_8\}$, covering task comprehension, reasoning validity, verification and calibration, factual and evidential grounding, com-

pleteness, instruction adherence, clarity and organization, and efficiency and conciseness (Tab. 5). A blind judge reads a prompt and a single response and returns one ordinal score per factor, $z_k(\boldsymbol{x}, \boldsymbol{y}) \in \{0, \ldots, 4\}$ for factor $f_k$; the anchors of that scale and the judge configuration are in appendix B.3.

The rubric that conditions training covers the whole ontology, so different models and benchmarks end up weighing these factors differently only through how the refinement loop restates them, never by dropping one. Every behavioral quantity in this paper is a contrast between two groups of these scores, and only two contrasts are used. On the task side, responses to a benchmark $B$ are grouped by the task outcome $o \in \{0, 1\}$ marking whether the task check passed, and *semantic demand* is the mean score gap between the two groups:

$$D_{m,B,k} = \mathbb{E}[z_k \mid o = 1, m, B] - \mathbb{E}[z_k \mid o = 0, m, B]. \tag{2}$$

Demand is an association measured for one model on one task, not a causal effect of training. On the data side, the two responses inside one preference pair are contrasted, $\Delta z_k = z_k(\boldsymbol{x}, \boldsymbol{y}^+) - z_k(\boldsymbol{x}, \boldsymbol{y}^-)$, and three statistics summarize its distribution over the corpus:

$$\mu_k = \mathbb{E}[\Delta z_k], \qquad C_k = \Pr(\Delta z_k \neq 0), \qquad P_k = \Pr(\Delta z_k > 0). \tag{3}$$

$\mu_k$ is the mean signed contrast, $C_k$ is the contrast coverage (the share of pairs that separate the two responses on $f_k$ at all), and $P_k$ is the label-agreement rate (the share of contrasts whose sign matches the preference label). Demand is a property of the model–task pair and supply a property of the corpus, but both live on the same eight axes, so the two can be compared directly.

The artifact the method edits is a *rubric* $R_t$: a short natural-language specification whose clauses are keyed to factors. The factors a rubric covers form a group $\mathcal{G} \subseteq \mathcal{F}$, a frozen bank compiles that group into one criterion text $\mathrm{crit}(\mathcal{G})$, and $\varnothing$ denotes the absent criterion. In this paper $\mathcal{G} = \mathcal{F}$ everywhere except in the factor-set ablation of Sec. 5.2, which asks whether the composition matters at all. A conditioning context is therefore $u \in \{\mathrm{crit}(\mathcal{G}), \varnothing\}$, and $\pi_\theta(\boldsymbol{y} \mid \boldsymbol{x}, u)$ is the policy's distribution over responses under that context. Because the rubric is present only during training, every reported score is $\pi_\theta(\boldsymbol{y} \mid \boldsymbol{x})$.

Tab. 9 collects the complete symbol table, including the quantities introduced by the transfer analysis and the mechanism probes in the appendix.

## 3 From Behavioral Diagnosis to Recursive Refinement

### 3.1 Shared Behavioral Coordinates

With the coordinates of Sec. 2.1 fixed, the diagnosis is a measurement on one model and one corpus: we score responses from Qwen3-4B on each benchmark and each HelpSteer3 pair, then read out demand $D_{m,B,k}$ (Eq. (2)) and supply $\mu_k$, $C_k$, $P_k$ (Eq. (3)). Fig. 4 reports the complete Qwen3-4B demand matrix; intervals, judge checks, and length controls appear in appendices B.3 and D.1.

The Qwen3-4B diagnosis separates the factors into Common ($f_2$ reasoning validity, $f_5$ completeness), Specific ($f_4$ factual grounding, $f_7$ clarity, $f_8$ efficiency), and Low-demand ($f_1$ comprehension, $f_3$ verification, $f_6$ instruction adherence). Fig. 3 shows why demand and supply are both needed. Factor $f_2$ is meaningful on five of seven benchmarks but is contrasted in only $14.6\%$ of HelpSteer3 pairs; factor $f_5$ is meaningful on four and contrasted in $47.4\%$. The coordinates connect task needs, rubric clauses, and later failure feedback without treating either axis as a prediction of training gain.

### 3.2 Training-Only Semantic Rubric Steering

A frozen bank holds one Prefer clause and one matched Avoid clause per factor, so it composes any factor group into a single criterion text. Every rubric in this paper covers all eight factors, $\mathcal{G} = \mathcal{F}$: composition is mechanical, in the fixed order $f_1$–$f_8$, and adds no learned weights, auxiliary loss, or altered labels. At training step $t$ of $N$, one context is drawn for the whole pair:

$$u_t = \begin{cases} \mathrm{crit}(\mathcal{G}), & \text{w.p. } q(\rho_t), \\ \varnothing, & \text{otherwise}, \end{cases} \qquad \rho_t = t/N, \tag{4}$$

where $q = 1$ for the first $10\%$ of training, follows a cosine withdrawal over the middle $80\%$, and is zero for the final $10\%$:

$$q(\rho) = \begin{cases} 1, & \rho \le .1, \\ \frac{1}{2}\{1 + \cos[\pi(\rho - .1)/.8]\}, & .1 < \rho \le .9, \\ 0, & \rho > .9. \end{cases} \tag{5}$$

Policy and frozen reference receive the same context. Rubric tokens are masked from the response loss, and the realized exposure is $0.5009$.

For context $u \in \{\mathrm{crit}(\mathcal{G}), \varnothing\}$, training uses the original DPO margin and objective

$$M_u = \log \frac{\pi_\theta(\boldsymbol{y}^+ \mid \boldsymbol{x}, u)}{\pi_{\mathrm{ref}}(\boldsymbol{y}^+ \mid \boldsymbol{x}, u)} - \log \frac{\pi_\theta(\boldsymbol{y}^- \mid \boldsymbol{x}, u)}{\pi_{\mathrm{ref}}(\boldsymbol{y}^- \mid \boldsymbol{x}, u)}, \qquad \mathcal{L} = -\mathbb{E}[\log \sigma(\beta M_{u_t})]. \tag{6}$$

Evaluation uses $\pi_\theta(\boldsymbol{y} \mid \boldsymbol{x})$ without a rubric. Conditioning can change the update applied to the same pair; formal conditions for criterion-free transfer are given in Sec. 3.4.

### 3.3 Criterion-Free Transfer

The rubric is withdrawn before evaluation, so the question that decides whether the method can work at all is whether conditioning during training moves the criterion-free policy. Two steps answer it: what update conditioning applies, and which direction that update moves the behaviour the benchmark measures.

Hold the dataset and reference fixed. For context $u \in \{\mathrm{crit}(\mathcal{G}), \varnothing\}$, let $s_u(\boldsymbol{y}) = \nabla_\theta \log \pi_\theta(\boldsymbol{y} \mid \boldsymbol{x}, u)$. Differentiating the Semantic Rubric Steering loss in Eq. (6) yields

$$g_u = \nabla_\theta[-\log \sigma(\beta M_u)] = -\kappa_u[s_u(\boldsymbol{y}^+) - s_u(\boldsymbol{y}^-)], \qquad \kappa_u = \beta\sigma(-\beta M_u). \tag{7}$$

The response score gradient sums token log-probability gradients. Conditioning can change both these gradients and the scalar multiplier. It does not introduce token weights or a criterion-specific latent reward. With the step-dependent conditioning rate of Eq. (5),

$$\mathbb{E}_{u_t}[g_{u_t}] = q(\rho_t) g_{\mathrm{crit}(\mathcal{G})} + (1 - q(\rho_t)) g_{\varnothing}. \tag{8}$$

This is the same preference target applied through a different training context, and the mixture changes over training as the criterion is withdrawn.

What remains is to characterize that direction.

### 3.4 Formal Statement for Criterion-Free Transfer

For a criterion-free evaluation distribution $\nu$, define the behavioural readout

$$A_k(\theta) = \mathbb{E}_{\boldsymbol{x} \sim \nu,\, \boldsymbol{y} \sim \pi_\theta(\cdot \mid \boldsymbol{x})}[z_k(\boldsymbol{x}, \boldsymbol{y})], \tag{9}$$

and let $\boldsymbol{G}_c$ and $\boldsymbol{G}_0$ be mean preference-loss gradients with and without the criterion, with $\boldsymbol{G}_q = (1 - q)\boldsymbol{G}_0 + q\boldsymbol{G}_c$.

**Proposition 1** *Suppose $A_k$ has an $H_k$-Lipschitz gradient on a neighborhood containing the two update segments. For $\theta_q = \theta - \eta \boldsymbol{G}_q$ and $\theta_0 = \theta - \eta \boldsymbol{G}_0$,*

$$A_k(\theta_q) - A_k(\theta_0) = -\eta q \langle \nabla_\theta A_k, \boldsymbol{G}_c - \boldsymbol{G}_0 \rangle + r_k, \tag{10}$$

$$|r_k| \le \frac{H_k \eta^2}{2} \left( \|\boldsymbol{G}_q\|^2 + \|\boldsymbol{G}_0\|^2 \right). \tag{11}$$

**The behavioral direction.** Assume differentiation and expectation can be interchanged and define $\bar{z}_k(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{y} \sim \pi_\theta(\cdot \mid \boldsymbol{x})}[z_k(\boldsymbol{x}, \boldsymbol{y})]$ at the current parameter state. The score-function identity gives

$$\nabla_\theta A_k = \mathbb{E}_{\boldsymbol{x} \sim \nu,\, \boldsymbol{y} \sim \pi_\theta(\cdot \mid \boldsymbol{x})}\left[ (z_k(\boldsymbol{x}, \boldsymbol{y}) - \bar{z}_k(\boldsymbol{x})) \nabla_\theta \log \pi_\theta(\boldsymbol{y} \mid \boldsymbol{x}) \right]. \tag{12}$$

The baseline term vanishes because the conditional mean score gradient is zero. A finite response space with a smooth, positive policy provides a sufficient setting for this identity; the factor score itself need not be differentiable in the response. The scoring function must be defined on that space, and a randomized judge can be represented by its fixed conditional mean score. Eq. (12) concerns generated responses under the criterion-free policy. The training-pair contrast $\Delta z_k$ and the gradient cosine in the existing probes are distinct quantities.

**Actual optimizer steps.** For two parameter increments $\boldsymbol{\delta}_c$ and $\boldsymbol{\delta}_0$ from the same parameter state, the same Taylor argument yields

$$A_k(\theta+\boldsymbol{\delta}_c) - A_k(\theta+\boldsymbol{\delta}_0) = \langle \nabla_\theta A_k, \boldsymbol{\delta}_c - \boldsymbol{\delta}_0 \rangle + r_k,$$
$$|r_k| \le \tfrac{1}{2} H_k \big( \|\boldsymbol{\delta}_c\|^2 + \|\boldsymbol{\delta}_0\|^2 \big). \tag{13}$$

Here the same smoothness assumption holds along both update segments. This form accommodates adaptive updates and momentum. A paired test isolating the context would start from the same parameters, optimizer state, and minibatch and evaluate the two resulting increments. The full-gradient mixture in Prop. 1 instead describes an idealized local step; averaging raw gradients does not generally equal averaging adaptive optimizer steps. Both formulations describe smooth expectations over sampled responses, whereas greedy-decoded outputs can change discontinuously. The final pilot shifts are therefore evidence about the observed behavior, not direct estimates of the proposition's inner product.

A fixed alignment does not by itself settle the sign, which is why the proposition is stated as a projected direction rather than as a magnitude: for $\boldsymbol{G}_0 = (1, 0)$ and $\boldsymbol{G}_c = (0.77, \sqrt{1-0.77^2})$, whose cosine is $0.77$, the incremental first-order effect is negative for a behavioural gradient $(0, 1)$, positive for $(0, -1)$, and zero for one orthogonal to $\boldsymbol{G}_c - \boldsymbol{G}_0$. The probes in appendix C.2 measure exactly that alignment, so they constrain the projection rather than the sign.

### 3.5 Evidence-Grounded Recursive Refinement

Given rubric $R_t$, the policy trained with it, and item-level benchmark feedback $\mathcal{E}_{t+1}$, one iteration is

$$R_t \longrightarrow \pi_{t+1} \longrightarrow \mathcal{E}_{t+1} \longrightarrow R_{t+1}. \tag{14}$$

RubricOpt (1) extracts failures, (2) aggregates patterns that recur across benchmarks, (3) attributes each pattern to $f_1$–$f_8$ and checks the current clause, (4) proposes a local clarification when the failure exposes a missing constraint or ambiguity, and (5) applies deterministic checks for evidence references and preservation of the factor anchor. A covered requirement or a failed check returns KEEP. Full prompts, invariants, and audit traces are in appendices F.1 and G.

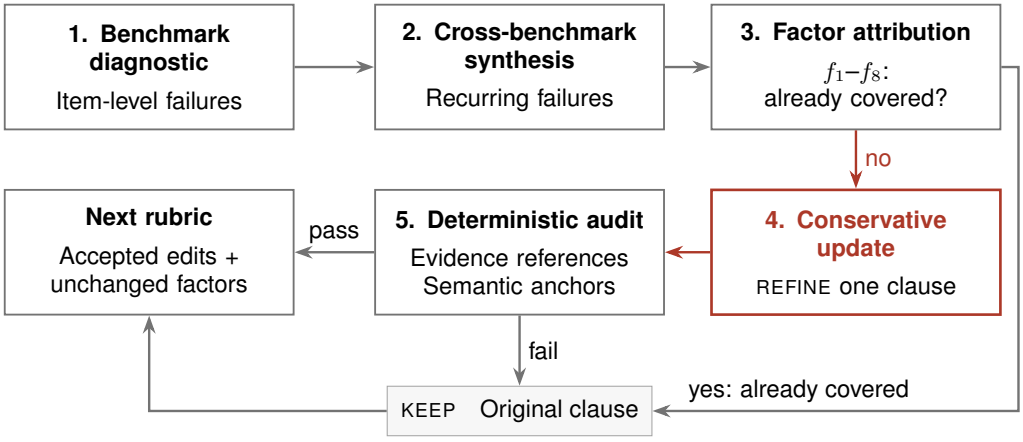



Figure 2: Recursive refinement. Item-level failures are synthesized across benchmarks and attributed to the fixed factors. Covered requirements lead to KEEP; an uncovered recurring constraint permits a local REFINE proposal, which must pass deterministic evidence and semantic-anchor checks.

## 4 Experimental Setup

The design crosses two benchmark classes, two preference corpora, two model families, and three model sizes, so that each claim can be read against the axis it varies on.

**Models.** We cover two families and three sizes: Qwen3-4B and Qwen3-8B from the Qwen3 family (Yang et al., 2025a), and Mistral-7B-Instruct-v0.3 from the Mistral family (Jiang et al., 2023). No arm touches the base weights. Every run trains a rank-64 LoRA adapter ($\alpha = 128$, dropout $0.05$) at learning rate $10^{-6}$, global batch $64$, DPO $\beta = 1$, and seed $42$, for one epoch.

**Data.** We use two public preference corpora, chosen because they differ in what they can teach. HelpSteer3 (Wang et al., 2025) supplies 36,264 pairs over 566 steps, and its contrasts are mostly open-ended and stylistic; Tulu3 (Lambert et al., 2025) supplies 116,355 pairs over 1,818 steps from a broad instruction mixture. Within one corpus, the ordinary-DPO and rubric arms share the model, pairs, labels, objective, optimizer, seed, and epoch count, so budgets are matched within a corpus rather than across the two.

**Benchmark classes.** Evaluation is always rubric-free and falls into two classes, which we keep separate in every table and figure. The *comprehensive-capability suite* is seven accuracy benchmarks—GSM8K (Cobbe et al., 2021), MATH-Hard (Hendrycks et al., 2021), BBH (Suzgun et al., 2023), MuSR (Sprague et al., 2024), ARC (Clark et al., 2018), TruthfulQA (Lin et al., 2022), and MMLU (Wang et al., 2024)—each with a task check and therefore a binary outcome; this is the class the Tulu3 substrate moves. The *open-ended generative suite* is three protocols—MT-Bench (Zheng et al., 2023), Arena-Hard (Li et al., 2024), and WildBench (Lin et al., 2024)—on which a judge scores free-form responses and there is no task check; this is the class the HelpSteer3 substrate moves. Generative cells use 2,000 matched items and Arena-Hard uses 500.

Every score is rubric-free, and no arm sees the rubric at evaluation time. Generation, judging, extraction, paired tests, and all absolute scores are in appendices E.3, H.1 and H.12.

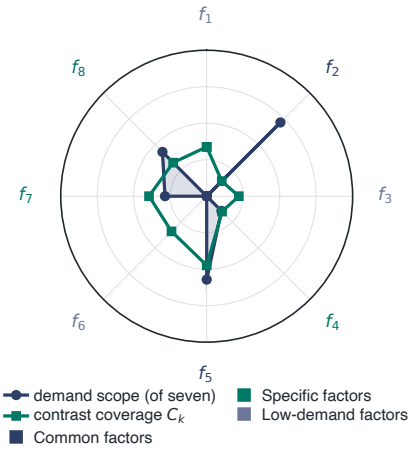



Figure 3: The Qwen3-4B diagnostic. Blue: how many of the seven benchmarks separate success from failure on each factor. Green: the share of HelpSteer3 pairs that contrast it.

## 5 Results

### 5.1 RQ1: Does Training-Only Conditioning Improve Rubric-Free Evaluation?

**Comprehensive-capability suite (Tulu3).** We first change the preference corpus while retaining the base model, objective, and full rubric. Tab. 1 gives rubric-free accuracy on both Qwen3 sizes. At four billion, conditioning raises BBH from 33.70 to 42.05 (+8.35) and the seven-task average from 58.52 to 60.34; MATH-Hard is unchanged, and the other five benchmarks move by at most 2.28 points. At eight billion the same direction appears on a much higher base (+3.80 BBH, +1.06 average), and the third family reproduces it: Mistral-7B gains +8.05 BBH points over its own ordinary-DPO control and +2.96 on the seven-task average, which puts it above its own untrained base as well. BBH is the largest cell in all three models.

Table 1: Tulu3 substrate, comprehensive-capability suite. Rubric-free accuracy after one epoch, in percent; Avg. is the unweighted seven-task mean. Each model is compared with its own ordinary-DPO control, so rows are read within a model block: Qwen3-4B, Qwen3-8B, and Mistral-7B-Instruct-v0.3.

| Model | Arm | GSM8K | MATH-Hard | BBH | MuSR | ARC | TQA | MMLU-Pro | Avg. |
|---|---|---|---|---|---|---|---|---|---|
| 4B | Base | 85.90 | 93.70 | 30.95 | 43.39 | 62.80 | 52.27 | 41.98 | 58.71 |
| 4B | Ordinary DPO | 84.53 | 93.15 | 33.70 | 42.59 | 61.09 | 52.71 | 41.88 | 58.52 |
| 4B | RubricOpt-R1 | 86.81 | 93.20 | **42.05** | 43.12 | 61.35 | 53.74 | 42.10 | **60.34** |
| 4B | RubricOpt-R2 (refine) | 80.74 | 92.40 | 38.85 | 42.06 | 59.47 | 55.16 | 40.43 | 58.44 |
| 8B | Ordinary DPO | 91.80 | 93.45 | 70.00 | 55.20 | 94.65 | 76.80 | 67.35 | 78.46 |
| 8B | RubricOpt-R1 | 91.40 | **95.15** | **73.80** | 55.15 | **95.00** | **77.85** | **68.30** | **79.52** |
| Mistral | Base | 45.75 | 19.65 | 36.95 | 52.45 | 79.45 | 58.75 | 30.05 | 46.15 |
| Mistral | Ordinary DPO | 44.45 | 20.90 | 32.40 | 50.95 | 79.50 | 53.90 | 30.00 | 44.59 |
| Mistral | RubricOpt-R1 | **47.90** | **21.50** | **40.45** | **53.85** | **79.60** | **57.40** | **32.15** | **47.55** |

**Open-ended generative suite (HelpSteer3).** On the HelpSteer3 substrate the effect lands on the open-ended protocols instead (Tab. 2). All four four-billion compositions raise Arena-Hard by 4.1–7.1 points against ordinary DPO on the same 500 items ($p \leq 0.008$), without a material increase in mean response length, while MT-Bench and WildBench move by less than a third of a point. Read against the Tulu3 result, the two corpora behave as complementary substrates: each moves the benchmark class it is suited to.

Table 2: HelpSteer3 substrate, open-ended generative suite. All values are rubric-free and each model is compared with its own ordinary-DPO control; Arena-Hard is the fixed-anchor win rate against a text-only reference answer.

| Model | Arm | MT-Bench | Arena-Hard | WildBench |
|---|---|---|---|---|
| Qwen3-4B | Base | 6.32 | 31.7% | 2.93 |
| Qwen3-4B | Ordinary DPO | 6.89 | 33.1% | 2.92 |
| Qwen3-4B | RUBRICOPT-R1 | 6.93 | 37.2% | 2.78 |
| Qwen3-8B | Ordinary DPO | 7.219 | 59.9% | 3.531 |
| Qwen3-8B | RUBRICOPT-R1 | 7.416 | 56.4% | 3.565 |
| Mistral-7B | Base | 5.09 | 18.2% | 2.32 |
| Mistral-7B | Ordinary DPO | 4.85 | 23.0% | 2.27 |
| Mistral-7B | RUBRICOPT-R1 | 5.14 | 22.3% | 2.29 |

**Model coverage.** Both tables carry the larger Qwen3 size and the Mistral family as well, each against its own matched ordinary-DPO control. The third family reproduces the capability result on the Tulu3 substrate (Tab. 1) and the same correction of ordinary-DPO drift on the open-ended protocols, where MT-Bench is the protocol that moves for both larger models (+0.20 and +0.29).

### 5.2 RQ2: WHAT DOES THE FACTOR-SET ABLATION SHOW?

All four HelpSteer3 compositions improve Arena-Hard, but their effects overlap and the added factors show no clear ordering. On the seven accuracy benchmarks, the four compositions span only 0.40 average points and are indistinguishable on that suite. We therefore carry the full ontology forward: the refinement operator may reach a factor that a smaller first-round set would have excluded, and the ablation tests composition rather than selecting the numerically best arm.

Table 3: Factor-set ablation, Qwen3-4B on HelpSteer3. Rubric-free scores. The factors column names what each arm conditions on; the last row is the full ontology the paper carries forward. The same ablation on the comprehensive-capability suite is Tab. 6.

| Arm | Factors | MT-Bench | Arena-Hard | WildBench |
|---|---|---|---|---|
| Ordinary DPO | — | 6.89 | 33.1% | 2.92 |
| Common | $f_2$, $f_5$ | 6.72 | 38.2% | 2.80 |
| Common + Specific | $f_2$, $f_4$, $f_5$, $f_7$, $f_8$ | 6.61 | 39.8% | 2.75 |
| Common + Low | $f_1$, $f_2$, $f_3$, $f_5$, $f_6$ | 6.90 | **40.2%** | 2.75 |
| All eight (RUBRICOPT-R1) | $\mathcal{F}$ | **6.93** | 37.2% | 2.78 |

BBH is the only benchmark on this suite where a composition separates from ordinary DPO: Common exceeds it by 2.30 points ($p = 0.0053$), and the other six columns move by less. The absence of an ordering among compositions is what motivates the full ontology as an editing space, rather than a claim that all eight clauses individually improve accuracy.

**Why the composition shows no ordering.** The diagnosis provides a data-side reading of the result shape. Reasoning validity is task-relevant on five of seven benchmarks, yet 85.4% of HelpSteer3 pairs tie on $f_2$. This sparse observable support is consistent with the limited spread of gains across the seven tasks, while the stronger open-ended Arena-Hard effect occurs on the substrate from which the rubric was diagnosed.

The same statistic on Tulu3 sharpens that reading. Tulu3 pairs separate the two responses more often than HelpSteer3 pairs on seven of the eight factors, and its lowest coverage is 0.294 where

HelpSteer3 has three sparse factors at $0.146$, $0.149$, and $0.220$ (Fig. 5). The largest Tulu3 advantages fall on $f_2$ ($+0.268$), $f_8$ ($+0.199$), $f_4$ ($+0.161$), and $f_7$ ($+0.135$). A corpus that already contrasts nearly every factor leaves the criterion less to add, which is one reading of why the Tulu3 substrate's gains stay concentrated in BBH instead of spreading across the seven tasks. This is a hypothesis about two corpora rather than a measurement of the mechanism, and it does not downgrade the HelpSteer3-side effects: an effect measured under one corpus remains an effect under that corpus.

### 5.3 RQ3: What Does Recursive Refinement Contribute?

The first Qwen3-4B pass finds one recurring $f_4$ gap. The frozen clause prohibited unsupported assertions but left the evidence requirement for "common knowledge" ambiguous. RubricOpt adds seven words:

```
Prefer:  factual claims grounded in the given information or
explicitly sourced common knowledge.
Avoid:  invented facts, unsupported assertions, or
inferences from vague common knowledge.
```

The edit clarifies the specification; its behavioral contribution is tested by two branches initialized from the same policy and differing only in $R_0$ versus $R_1$.

Table 4: Qwen3-4B round-1 decisions. REFINE requires a recurring missing constraint; malformed or already-covered proposals return KEEP.

| Factor | Decision | Basis |
|---|---|---|
| $f_4$ factual grounding | REFINE | unsourced "common knowledge" constraint missing |
| $f_3$ verification | KEEP | requirement covered; proposed rewrite fails gate |
| $f_1$, $f_2$, $f_5$–$f_8$ | KEEP | no recurring uncovered requirement |

Round 1 also shows why refinement needs both a coverage decision and a deterministic gate. Factor $f_3$ receives the strongest failure signal, but its existing Avoid clause already names the concrete requirement (unchecked steps). The engine nevertheless proposes an $f_3$ rewrite; the gate rejects it because it drops part of the frozen semantic anchor and imports another factor's vocabulary. Factor $f_3$ therefore returns to KEEP. The coverage rule prevents redundant editing, while the gate prevents a malformed edit from changing factor identity.

## 6 Conclusion

RubricOpt makes a natural-language behavioral specification an explicit, editable part of preference optimization. Shared behavioral coordinates connect task demand, preference-data support, rubric clauses, and failure feedback; training-only steering yields rubric-free gains; and recursive refinement turns specific failures into auditable local edits. Across the measured settings the method improves BBH and Arena-Hard, while matched continuation shows why an edit has to be evaluated separately from additional training. The diagnosis also says where to intervene next: demand and supply distinguish a task requirement from the contrast support the corpus actually provides, so once a recurring failure is already specified, the next experiment is to change the data—for instance by adding targeted contrasts—rather than to expand the rubric further. The resulting design principle is simple: diagnose behavior and data in one space, expose the specification during training, and update it only when feedback identifies a concrete requirement the current text does not state.

**Scope and future work.** Demand is associative, the ontology is manually specified, and the current study uses three model sizes from two families, two corpora, one seed, and in-feedback refinement items; the demand diagnostic and the refinement pass have not yet been run for the second corpus or the third family. The next steps are to add a content-matched context placebo and to test whether targeted contrast augmentation turns diagnosed supply gaps into behavioral gains.

## REFERENCES

Shir Ashury-Tahan, Yifan Mai, Elron Bandel, Michal Shmueli-Scheuer, and Leshem Choshen. ErrorMap and ErrorAtlas: Charting the failure landscape of large language models. *arXiv preprint arXiv:2601.15812*, 2026. doi: 10.48550/arXiv.2601.15812. URL https://arxiv.org/abs/2601.15812.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021. doi: 10.48550/arXiv.2112.00861. URL https://arxiv.org/abs/2112.00861.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Rémi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4447–4455. PMLR, 2024. URL https://proceedings.mlr.press/v238/gheshlaghi-azar24a.html.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Wang, Samuel Marks, Charbel-Raphaël Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J. Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Bıyık, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*, 2023. URL https://arxiv.org/abs/2307.15217. arXiv:2307.15217.

Wei Chen, Yubing Wu, Junmei Yang, Delu Zeng, Qibin Zhao, John Paisley, Min Chen, and Zhou Wang. Towards disentangled preference optimization dynamics: Suppress the loser, preserve the winner. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=TaNH4XiQ6P.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? Try ARC, the AI2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12634–12651. PMLR, 2024. URL https://proceedings.mlr.press/v235/ethayarajh24a.html.

Víctor Gallego. Configurable preference tuning with rubric-guided synthetic data. *arXiv preprint arXiv:2506.11702*, 2025. doi: 10.48550/arXiv.2506.11702. URL https://arxiv.org/abs/2506.11702. ICML 2025 Workshop on Models of Human Feedback for AI Alignment.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*. PMLR, 2023. URL https://arxiv.org/abs/2210.10760.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint*

*arXiv:2507.17746*, 2025. doi: 10.48550/arXiv.2507.17746. URL https://arxiv.org/abs/2507.17746.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset, 2021. URL https://arxiv.org/abs/2103.03874.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. doi: 10.48550/arXiv.2207.12598. URL https://arxiv.org/abs/2207.12598. NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications.

Tom Hosking, Phil Blunsom, and Max Bartolo. Human feedback is not gold standard. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2309.16349. arXiv:2309.16349.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2025. URL https://arxiv.org/abs/2411.15124.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline, 2024. URL https://arxiv.org/abs/2406.11939.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. WildBench: Benchmarking LLMs with challenging tasks from real users in the wild, 2024. URL https://arxiv.org/abs/2406.04770.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods, 2022. URL https://arxiv.org/abs/2109.07958.

Aiwei Liu, Haoping Bai, Zhiyun Lu, Yanchao Sun, Xiang Kong, Xiaoming Wang, Jiulong Shan, Albin Madappally Jose, Xiaojiang Liu, Lijie Wen, Philip S. Yu, and Meng Cao. TIS-DPO: Token-level importance sampling for direct preference optimization with estimated weights. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/7fb9f39075a5202472676a7531568212-Abstract-Conference.html.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2019. URL https://papers.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

Yu Meng, Mengzhou Xia, and Danqi Chen. SimPO: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3946. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e099c1c9699814af0be873a175361713-Abstract-Conference.html.

Reid Pryzant, Dan Iter, Jerry Li, Yin Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with "gradient descent" and beam search. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7957–7968, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.494. URL https://aclanthology.org/2023.emnlp-main.494/.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2338. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html.

Rafael Rafailov, Yaswanth Chittepu, Ryan Park, Harshit Sikchi, Joey Hejna, Bradley Knox, Chelsea Finn, and Scott Niekum. Scaling laws for reward model overoptimization in direct alignment algorithms. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2406.02900.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards understanding sycophancy in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.13548. arXiv:2310.13548.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in RLHF. In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2310.03716. arXiv:2310.03716.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2209.13085.

Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context. *arXiv preprint arXiv:2209.15189*, 2022. doi: 10.48550/arXiv.2209.15189. URL https://arxiv.org/abs/2209.15189.

Zayne Sprague, Xi Ye, Kaj Bostrom, Swarat Chaudhuri, and Greg Durrett. MuSR: Testing the limits of chain-of-thought with multistep soft reasoning, 2024. URL https://arxiv.org/abs/2310.16049.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.09261. arXiv:2210.09261.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5–6):544–557, 2009. doi: 10.1016/j.neunet.2009.06.042. URL https://doi.org/10.1016/j.neunet.2009.06.042.

Xukai Wang, Liangqi Li, Zhiyue Xu, Jingang Zhou, Xiaoyu Shi, Jiansheng Cai, Bo Zhang, Zhe Li, and Xu-Yao Zhang. APTER: Adaptive post-training with expert-grounded rubrics, 2026. URL https://arxiv.org/abs/2608.14212.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark, 2024. URL https://arxiv.org/abs/2406.01574.

Zhilin Wang, Jiaqi Zeng, Olivier Delalleau, Hoo-Chang Shin, Felipe Soares, Alexander Bukharin, Ellie Evans, Yi Dong, and Oleksii Kuchaiev. Helpsteer3-preference: Open human-annotated preference data across diverse tasks and languages, 2025. URL https://arxiv.org/abs/2505.11475.

Xincheng Wei, Yifan Ding, Yoshua Li, Dongsheng Ma, Rongxiang Weng, Xunliang Cai, Wenjian Ding, and Yao Zhang. DiagEvo: Diagnosis-guided self-evolution via hierarchical error memory, 2026. URL https://arxiv.org/abs/2609.00768.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason E. Weston, and Sainbayar Sukhbaatar. Meta-rewarding language models: Self-improving alignment with LLM-as-a-meta-judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Suzhou, China, 2025a. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.583. URL https://aclanthology.org/2025.emnlp-main.583/.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://arxiv.org/abs/2405.00675. arXiv:2405.00675.

Teng Xiao, Yige Yuan, Mingxiao Li, Zhengyu Chen, and Vasant G. Honavar. On a connection between imitation learning and RLHF. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2503.05079. arXiv:2503.05079.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Erxue Min, and Sophia Ananiadou. Selective preference optimization via token-level reward function estimation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7032–7056, Suzhou, China, 2025b. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.359. URL https://aclanthology.org/2025.emnlp-main.359/.

Ning Yang, Hai Lin, Yibo Liu, Baoliang Tian, Guoqing Liu, and Haijun Zhang. Token-importance guided direct preference optimization. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/172fe9f8cc55953bad5c24774bf0142b-Abstract-Conference.html. ICLR 2026 Poster.

Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. FLASK: Fine-grained language model evaluation based on alignment skill sets. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2307.10928. ICLR 2024 Spotlight; arXiv:2307.10928.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E. Weston. Self-rewarding language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57905–57923. PMLR, 2024. URL https://proceedings.mlr.press/v235/yuan24d.html.

Yongcheng Zeng, Guoqing Liu, Weiyu Ma, Ning Yang, Haifeng Zhang, and Jun Wang. Token-level direct preference optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 58348–58365. PMLR, 2024. URL https://proceedings.mlr.press/v235/zeng24c.html.

Zhiyuan Zeng, Yizhong Wang, Hannaneh Hajishirzi, and Pang Wei Koh. EvalTree: Profiling language model weaknesses via hierarchical capability trees. In *Second Conference on Language Modeling*, 2025. URL https://arxiv.org/abs/2503.08893. arXiv:2503.08893.

Bo-Wen Zhang, Junwei He, Wen Wang, Song-Lin Lv, Wentao Ma, Rongyi Lin, Shuhan Zhong, and Lan-Zhe Guo. CoRT: Counterfactual replay for token-level rubric-guided policy optimization. *arXiv preprint arXiv:2607.25659*, 2026. doi: 10.48550/arXiv.2607.25659. URL https://arxiv.org/abs/2607.25659.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623. Curran Associates, Inc., 2023. doi: 10.52202/075280-2020. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/91f18a1287b398d378ef22505bf41832-Paper-Datasets_and_Benchmarks.pdf.

Wenxuan Zhou, Shujian Zhang, Lingxiao Zhao, and Tao Meng. T-REG: Preference optimization with token-level reward regularization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27876–27889, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1353. URL https://aclanthology.org/2025.acl-long.1353/.

**LLM usage disclosure.** Large language models were used in this work in three ways. *Aiding and polishing the writing*: yes. *Retrieval and discovery, such as finding related work*: yes. *Drafting sections of the paper*: yes. Details of each use are described in the paper.

# A Complete Diagnostic and Main-Result Tables

This appendix preserves the complete specifications and result grids summarized in the main text.

## A.1 Factor ontology and complete demand profile

Table 5: The fixed semantic factor space and frozen rubric bank, keyed by factor symbol (the names are in Sec. 2.1). Each factor contributes one Prefer clause and one matched Avoid clause.

| Factor | Prefer | Avoid |
|---|---|---|
| $f_1$ | understand what was asked | answer a different question or ignore conditions |
| $f_2$ | use grounded, consistent reasoning | make unsupported inferences or invalid deductions |
| $f_3$ | verify key steps and calibrate uncertainty | leave steps unchecked or express unsupported confidence |
| $f_4$ | ground claims in given information or common knowledge | invent facts or make unsupported assertions |
| $f_5$ | include the reasoning and information needed | omit necessary reasoning, information, or task parts |
| $f_6$ | follow instructions and stay on task | drift off-topic or violate requested formats |
| $f_7$ | express and organize the response clearly | jumble steps or hide the answer |
| $f_8$ | keep only necessary content | repeat reasoning or add filler |

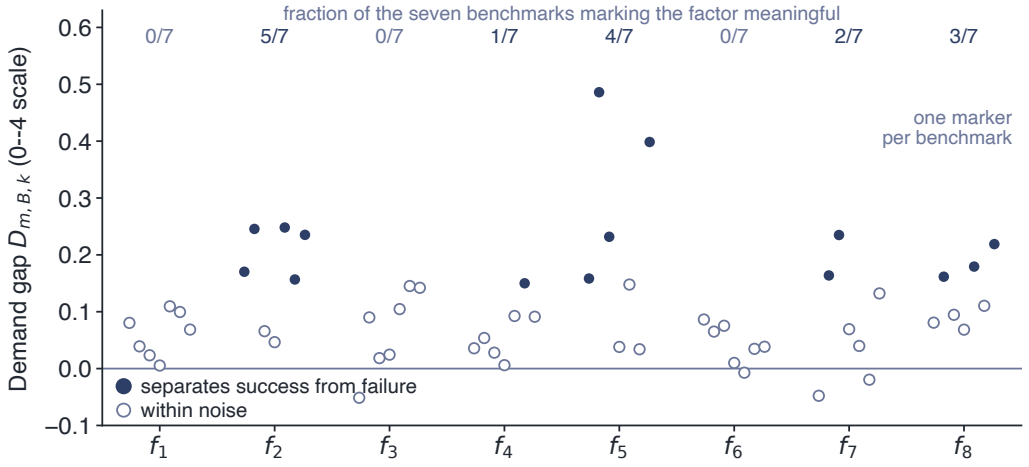



Figure 4: Qwen3-4B semantic demand over the seven benchmarks. Each marker is one benchmark's factor-score gap between successful and failed responses; filled markers are the cells the frozen diagnostic rule marks meaningful, and the figure above each factor gives that count out of seven. Fig. 3 reads the same model against preference-data supply; intervals and effect sizes are in appendix D.1.

## A.2 Formal statement for criterion-free transfer

The proposition and its proof are in Sec. 3.4.

### A.3 Complete fixed-rubric results

Tabs. 1 and 2 are retained in the main text because they directly support the effectiveness claim, and the factor-set ablation that supports the composition claim is plotted in **??**b; its exact per-benchmark values are below. This appendix supplies the paired deltas, confidence intervals, and protocol details behind them.

Table 6: HelpSteer3 factor-set ablation, Qwen3-4B. Rubric-free accuracy in percent; Avg. is the unweighted mean.

| Condition | GSM8K | MATH | BBH | MuSR | ARC | TQA | MMLU | Avg. |
|---|---|---|---|---|---|---|---|---|
| Base | 85.90 | 93.70 | 30.95 | 43.39 | 62.80 | 52.27 | 41.98 | 58.71 |
| Ordinary DPO | 87.57 | 93.05 | 34.95 | 43.12 | 62.63 | 53.91 | 42.30 | 59.65 |
| Common | 87.04 | 93.40 | 37.25 | 43.12 | 63.23 | 53.72 | 42.23 | 60.00 |
| Common + Specific | 87.04 | 93.25 | 38.45 | 42.99 | 63.40 | 53.71 | 42.25 | 60.16 |
| Common + Specific + Low | 87.49 | 93.80 | 36.05 | 43.25 | 63.31 | 53.74 | 42.25 | 59.98 |
| Common + Low | 86.88 | 93.40 | 35.80 | 43.12 | 63.14 | 53.69 | 42.26 | 59.76 |

### A.4 Complete matched-refinement and Mistral results

Table 7: Matched wording-edit increment RubricOpt-R2 (refine) − RubricOpt-R2 (keep) on the open-ended protocols. Intervals are 95% bootstrap intervals over items.

| Benchmark | Difference | 95% CI | Paired test |
|---|---|---|---|
| MT-Bench | +0.4062 | [+0.0813, +0.7562] | $p = 0.061$ |
| Arena-Hard | +0.0030 | [−0.035, +0.042] | $p = 0.918$ |
| WildBench | +0.0977 | [−0.031, +0.231] | $p = 0.164$ |

Table 8: Mistral-7B-Instruct-v0.3 × HelpSteer3 open-ended results; the same model's comprehensive-capability cells are the Mistral block of Tab. 1. DPO is ordinary DPO and Δ the paired difference RubricOpt-R1−DPO on identical items.

| Protocol | Base | DPO | RubricOpt-R1 | Δ |
|---|---|---|---|---|
| MT-Bench (0–10) | 5.09 | 4.85 | 5.14 | +0.32 |
| Arena-Hard (win rate) | 0.182 | 0.230 | 0.223 | −0.007 |
| WildBench (1–10) | 2.32 | 2.27 | 2.29 | +0.024 |

## B Notation and Measurement Protocol

### B.1 Notation

**Proof of Prop. 1.** For either $\boldsymbol{G} = \boldsymbol{G}_q$ or $\boldsymbol{G} = \boldsymbol{G}_0$, the Lipschitz-gradient assumption gives the Taylor expansion

$$A_k(\theta - \eta\boldsymbol{G}) = A_k(\theta) - \eta\langle\nabla_\theta A_k, \boldsymbol{G}\rangle + e(\boldsymbol{G}), \qquad |e(\boldsymbol{G})| \le \tfrac{1}{2}H_k\eta^2\|\boldsymbol{G}\|^2. \tag{15}$$

Subtract the two expansions and use $\boldsymbol{G}_q - \boldsymbol{G}_0 = q(\boldsymbol{G}_c - \boldsymbol{G}_0)$. The remainder is $r_k = e(\boldsymbol{G}_q) - e(\boldsymbol{G}_0)$, whose absolute value is bounded by Eq. (11) by the triangle inequality. In particular, a sufficient condition for strict improvement over the ordinary step is

$$-q\langle\nabla_\theta A_k, \boldsymbol{G}_c - \boldsymbol{G}_0\rangle > \frac{H_k\eta}{2}\big(\|\boldsymbol{G}_q\|^2 + \|\boldsymbol{G}_0\|^2\big). \tag{16}$$

A nonzero first-order term dominates the quadratic remainder for sufficiently small positive $\eta$, proving the sign statements. A zero inner product leaves the sign undetermined. Improvement over the initial policy is a different comparison: its first-order term is $-\eta\langle\nabla_\theta A_k, \boldsymbol{G}_q\rangle$. □

Table 9: Notation used in the method and appendix.

| Symbol | Meaning |
|---|---|
| $m, \pi_\theta, \pi_{\text{ref}}$ | Current model, its trainable policy, and the frozen reference. |
| $B, o$ | Diagnostic task and its binary outcome. |
| $\mathcal{D}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}^+, \boldsymbol{y}^-$ | Preference data, prompt, response, and chosen/rejected responses. |
| $\mathcal{F} = \{f_1, \ldots, f_8\}, K$ | Fixed factor ontology, its eight factors, and their count. |
| $z_k(\boldsymbol{x}, \boldsymbol{y}) \in \{0, \ldots, 4\}$ | Ordinal score a blind judge assigns to factor $f_k$. |
| $D_{m,B,k}$ | Model- and task-specific semantic demand (Eq. (2)). |
| $\Delta z_k, \mu_k, C_k, P_k$ | Pair contrast, its mean, contrast coverage, and label-agreement rate (Eq. (3)). |
| $\mathcal{G} \subseteq \mathcal{F}, \text{crit}(\mathcal{G}), \varnothing$ | Factor group a rubric covers (all eight in this paper), the criterion compiled from its clauses, and the absent criterion. |
| $u, u_t, q(\rho_t)$ | Conditioning context $u \in \{\text{crit}(\mathcal{G}), \varnothing\}$, the context sampled at training step $t$, and the conditioning rate at progress $\rho_t$. |
| $\beta, \sigma, \mathcal{L}$ | DPO scale, logistic function, and Semantic Rubric Steering loss. |
| $M_u, s_u, g_u, \kappa_u$ | Chosen–rejected log-ratio margin, response score gradient, loss gradient, and scalar multiplier, all in context $u$. |
| $\nu, A_k, H_k$ | Evaluation prompt distribution, criterion-free expected factor score $\mathbb{E}_\nu[\bar{z}_k(\boldsymbol{x})]$, and its gradient Lipschitz constant. |
| $\boldsymbol{G}_c, \boldsymbol{G}_0, \boldsymbol{G}_q$ | Mean loss gradients with and without the criterion, and their mixture at rate $q$. |
| $\eta, \theta_q, \theta_0, r_k$ | Step size, two local updated parameter states, and the transfer remainder. |
| $\bar{z}_k(\boldsymbol{x}), \boldsymbol{\delta}_c, \boldsymbol{\delta}_0$ | Prompt-specific expected factor score and two actual optimizer increments from a shared state. |
| $S_k$ | Semantic shift, the change in mean factor score after an update. |
| $R_t, \pi_t, \mathcal{E}_t$ | Rubric, policy, and behavioral feedback evidence at round $t$ (Eq. (14)). |
| RUBRICOPT-R2 (KEEP) | Matched re-optimization branch that keeps the criterion $R_0$. |
| RUBRICOPT-R2 (REFINE) | Matched re-optimization branch that refines the criterion to $R_1$; the two branches differ only in the criterion. |

### B.2 From Response Scores to Semantic Contrasts

Every quantity in this paper comes from one measurement and one operation. The measurement is a factor score. The operation is a contrast between two groups of scores. The quantities differ only in which groups are contrasted, so a reader who follows this section can reconstruct the rest.

**Response scores.** The judge reads a prompt and a single response and returns one integer per factor, $z_k(\boldsymbol{x}, \boldsymbol{y}) \in \{0, \ldots, 4\}$. The judge does not see the task outcome, the preference role, or the gold answer (appendix B.3). This is the only primitive.

**Demand.** Group responses to a benchmark by whether the task check marked them correct, and contrast the two groups: $D_{m,B,k} = \mathbb{E}[z_k \mid \text{success}] - \mathbb{E}[z_k \mid \text{failure}]$ (Eq. (2)). The contrast is between populations of responses to the same benchmark for the same model.

**Supply.** For one preference pair, contrast the two responses within the pair: $\Delta z_k = z_k(\boldsymbol{x}, \boldsymbol{y}^+) - z_k(\boldsymbol{x}, \boldsymbol{y}^-)$. Summarizing over pairs gives $\mu_k$, $C_k$, and $P_k$ (Eq. (3)). The contrast is within a pair, and the pair supplies the label.

**Semantic shift.** After a criterion has been trained, the same scores can be computed on outputs of the criterion-free policy before and after the update, giving $S_k = \mathbb{E}[z_k^{\text{after}}] - \mathbb{E}[z_k^{\text{before}}]$. This asks where training moved the model rather than whether a task score improved. **??** reports this measurement for the pilot conditions, over the 32,000 criterion-free pilot responses, and the aligned shift it was designed to detect was not observed.

**Worked demand example.** One GSM8K prompt asks for the cost of six pounds of butter when a promotion makes the second pound half price at $4.00 per pound. One sampled response applies the promotion correctly, pays $6 for each pair of pounds, and answers $18. Another misreads it as $4 per one and a half pounds and answers $16. Both are confident and both are well written; only one is right. The judge scores the successful response $[2, 3, 1, 3, 2, 3, 3, 3]$ and the failed one $[2, 1, 1, 3, 2, 3, 3, 3]$. The difference is $[0, +2, 0, 0, 0, 0, 0, 0]$. The two responses are identical on seven factors and differ only on reasoning validity. The failed response is not worse at everything, which is why a per-factor profile carries more than the outcome label. This pair does not estimate demand. Demand is the aggregate success–failure expectation over the benchmark population, and the example only illustrates the scoring operation that the aggregate is built from.

**Worked supply example.** The same operation applies to a preference pair. In one HelpSteer3 pair—a joke request—the chosen and rejected responses differ by $[0, 0, 0, 0, +1, 0, 0, 0]$. The pair supplies evidence about completeness and no evidence at all about reasoning validity. Aggregated over the dataset, this is the pattern behind the $f_2$ tie rate of 0.854 (Tab. 15): most pairs say nothing about that factor, even though the target tasks separate success from failure on it.

A second pair shows that supply is not always single-factor. In a second pair, the prompt asks whether there has ever been a female US president. The chosen response answers correctly and notes that several women have run. The rejected response invents eleven female presidents and names Geraldine Ferraro and Barack Obama. Here the difference is $[+2, +2, 0, +2, +1, +2, 0, +1]$: the pair carries evidence about task comprehension, reasoning validity, factual grounding, completeness, and instruction adherence at once. One pair can therefore contribute to several factors, which is why supply is summarized as a distribution over factors rather than assigned to one.

### B.3 Judge Configuration and Reproducibility

**Production configuration.** All scores reported in this paper come from a single judge, Qwen3-8B, running one fixed, version-tagged judge prompt. Decoding is greedy (temperature 0, top-$p$ 1, top-$k$ 1) with at most 1100 new tokens, at batch size 64. The judge is blind to the task outcome, the gold answer, the preference role, and the identity of the generating model. One call returns all eight factor scores. Scores are raw integers and are not normalized across factors, responses, or runs.

**Output schema and parsing.** One call returns a single JSON object with one entry per factor, each holding an integer score and a supporting sentence. Parsing proceeds in stages: an invalid-

escape repair, then a regular-expression salvage of the eight scores, then up to two re-decodes. A field that survives all stages is logged as a parse error and excluded from the canonical set rather than imputed; 23 of 108,708 responses in the first scoring pass took that path.

**Score anchors.** The shared scale is 0 = clearly poor or absent, 1 = weak, 2 = mixed or partial, 3 = good, and 4 = strong. The production prompt defines the top of the scale as an absolute standard rather than a comparative one: a 4 is earned by concrete extra rigor, such as re-checking a computation, stating and handling an assumption or edge case, noticing an ambiguity, or flagging its own uncertainty, and it is explicitly not awarded for being correct and fluent. A 3 is competent, complete, routine work with no extra rigor; a 2 does the minimum with visible gaps; a 1 is missing or merely asserts an important behavior. The prompt also asks the judge to list two strengths and two weaknesses before scoring, and gives per-factor rules that cap a score at 1; for reasoning validity, the cap applies when any step is a non-sequitur or an equation does not follow.

**Applicability.** The prompt permits `NA` only when a behavior cannot occur for that response kind, and states that a required behavior which is missing still scores 0 or 1 rather than `NA`. In practice the rule never fired: across the 108,685 canonical scored responses, every factor was assessable on every response.

**Secondary agreement audit.** A different judge model rescored 1,200 responses. On items where the audit judge gave a decisive score, mean agreement with the production judge was 0.741. Per factor, agreement was highest for task comprehension (0.909), factual and evidential grounding (0.857), and instruction adherence (0.800), and lowest for clarity and organization (0.633), verification and calibration (0.616), and efficiency and conciseness (0.559). Agreement is moderate overall and weakest for the style-adjacent factors, which is why results involving clarity and organization are read cautiously in the main text rather than as clean contrasts.

**Factor separability.** Across 108,685 complete responses, the production judge yields mean absolute inter-factor correlation 0.195 and maximum 0.362. The older judge prompt, used as a collapse control, reaches maximum correlation 1.000 and mean 0.378, and produces only four distinct score vectors across 288 responses, whereas the production prompt produces 822. The eight factors are therefore not collapsed onto one quality axis in this analysis, though low pairwise correlation is a weaker statement than independence or causal separation.

# C Semantic Rubric Steering Detail

## C.1 Conditioning and the Preference Update

The gradient decomposition, the transfer proposition and its proof are in Sec. 3.4.

## C.2 Mechanistic Diagnostics in Detail

This appendix reports the probes summarized in **??**. Tab. 10 collects the three readouts; "not supported" in the last column refers to the hypothesis that criterion-free behavior moves along the factor a criterion names.

**Semantic shift.** The 32,000 criterion-free responses of the four pilot conditions (base, ordinary DPO, $f_2$ criterion, $f_7$ criterion) were rescored with the production judge. Scores are paired by prompt and sample, and shifts are reported on the raw 0–4 scale with bootstrap intervals. Tab. 11 summarizes the outcome. The comparison that the mechanism hypothesis predicts is the target-factor row for $f_2$ and $f_7$ relative to ordinary DPO; neither is positive and aligned. The largest shift anywhere in the comparison is 0.014, and every criterion-specific shift is at most 0.025, against demand gaps of roughly 0.15 to 0.49 for the same factors on the target tasks.

**Likelihood-margin probe.** The probe uses a strict normalized likelihood margin: the criterion is present as context only and is excluded from both the response log-probability numerator and

Table 10: Mechanism readouts, measured at development scale rather than on the formal arms.

| Readout | What it measures | Result | Interpretation |
|---|---|---|---|
| Semantic shift | Factor scores on 32,000 criterion-free responses before and after training | Criterion-specific shifts $\leq 0.025$, against demand gaps of 0.15–0.49 on the same factors; the $f_2$ and $f_7$ arms do not move their own factor upward | No criterion-aligned criterion-free shift (*not supported*) |
| Likelihood sensitivity | Normalized margin change by sign of the pair's semantic delta, criterion tokens excluded | $f_5$ is weakly and directionally correct, with a matched-factor slope about twice the cross-factor slope; $f_2$ is non-monotone with a rank correlation of 0.003 | Weak factor-specific sensitivity for one factor, none detectable for the other |
| Gradient probe | Cosine and magnitude change between conditioned and unconditioned parameter gradients on 7,253 pairs, no optimizer step | Cosine 0.74–0.79, relative change 0.65–0.72; signed reinforcement uniform at $-0.17$ to $-0.25$; matched-delta rank correlations $+0.007$ and $-0.019$ | The criterion changes the update; the pair's semantic delta does not predict the change in this sample |

Table 11: Criterion-free semantic shift on the 32,000 pilot responses. Each row contrasts two conditions after the criterion has been removed. Shifts are on the raw 0–4 factor scale. "Target" is the factor the criterion names.

| Comparison | Target | Observed shift |
|---|---|---|
| Ordinary − Base | — | $f_5$ $+0.011$ and $f_7$ $+0.014$, both significant; all other factors near zero. |
| $f_2$ − Ordinary | $f_2$ | Target shift $\approx +0.001$, interval includes zero; $f_7$ shifts significantly downward; no target-aligned pattern. |
| $f_7$ − Ordinary | $f_7$ | Target shift significantly negative; no aligned transfer. |

the response-length denominator, so the quantity can be compared across conditioned and unconditioned contexts. Pairs are bucketed by the sign of the semantic delta $\Delta z_k$ for the probed factor, and the margin change is reported in nats per response token. Tab. 12 gives the buckets. For $f_5$ the three buckets are ordered correctly and the matched-factor slope, $+0.0226$ per point of semantic contrast, is about twice the cross-factor slope; for $f_2$ the buckets are not ordered, the rank correlation is $0.003$, and the slope is indistinguishable from the cross-condition baseline.

Table 12: Strict normalized likelihood-margin response to criterion conditioning, in nats per response token, bucketed by the sign of the semantic delta of the probed factor. Criterion tokens are context only and are excluded from the numerator and the length denominator.

| Probed factor | $\Delta > 0$ | $\Delta = 0$ | $\Delta < 0$ | Spearman | Slope |
|---|---|---|---|---|---|
| $f_5$ Completeness | $+0.0252$ | $-0.0132$ | $-0.0265$ | $0.076$ | $+0.0226$ |
| $f_2$ Reasoning validity | $+0.0034$ | $-0.0089$ | $-0.0016$ | $0.003$ | $\approx$ baseline |

**Gradient probe.** The probe uses 7,253 pairs, a $20\%$ sample of the full training set, and compares the parameter gradient induced by a conditioned context with the gradient induced by the original prompt on the same pair. The two gradients differ, with a cosine of about $0.74$ to $0.79$ and a relative gradient change of about $0.65$ to $0.72$ across the sampled pairs. The change is not, however, organized by the semantic delta of the factor. Signed reinforcement is uniformly negative across the sample, between $-0.17$ and $-0.25$, which means the conditioned update does not preferentially reinforce pairs that contrast the probed factor in the preferred direction. Matched-factor rank correlations are $+0.007$ for an $f_2$ criterion and $-0.019$ for an $f_5$ criterion, and a bivariate fit including both factors gives coefficients of about $-0.0026$ and $-0.0024$. The Stage-1 $f_5$ likelihood signal of Tab. 12 therefore does not clearly propagate into a semantic gradient-response signal.

### C.3 Criteria and the Frozen Criterion Bank

**Bank structure.** The formal study uses a frozen composable criterion bank. The bank holds one entry per factor, and each entry has exactly three fields: a canonical factor name, one positive behavior, and one matched failure mode. Tab. 5 in the main text quotes all eight entries verbatim; this appendix gives the composition rule, a worked example, and the token accounting. The bank was frozen before the formal runs and each composition's suffix was fingerprinted, so a reported run can be traced to the exact criterion text it used. A composed single-factor instruction is 21 to 23 tokens under the Qwen3-4B tokenizer, and a four-factor set adds about 43 to 64 masked tokens to the training context. The $f_2$ and $f_5$ clauses follow the wording frozen for the progressive runs rather than the pilot: the $f_5$ entry states the positive behavior as "responses that include the reasoning and information needed to fully address the task", not the shorter pilot phrasing.

**Composition.** Composition is mechanical: for a factor group $\mathcal{G}$, each factor contributes one clause to the prefer list and one clause to the avoid list, in the fixed factor order $f_1$–$f_8$, and no factor is dropped or weighted. The formal runs use $\mathcal{G} = \mathcal{F}$, so the criterion carries all eight clause pairs. The template is

```
Capability priorities: <factor A>, <factor B>
Prefer:
- <prefer clause of factor A>;
- <prefer clause of factor B>.
Avoid:
- <avoid clause of factor A>;
- <avoid clause of factor B>.
```

For the Common group $\mathcal{G} = \{f_2, f_5\}$, the composed criterion is:

```
Capability priorities: reasoning validity, completeness
Prefer:
- reasoning steps that are logically grounded and internally
consistent;
```

```
- responses that include the reasoning and information
needed to fully address the task.
Avoid:
- unsupported intermediate inferences or invalid deductions;
- missing necessary reasoning, information, or important
parts of the task.
```

Larger sets extend the same two lists in the order the factors are listed, so a criterion for the full eight-factor set contains eight prefer clauses and eight avoid clauses. Colour marks the artifact header, the preferred behaviors, and the matched failure modes; it is a reading aid and is not part of the criterion text.

**The $f_7$ pilot criterion.** The $f_7$ criterion used in the pilot predates the bank and differs from the $f_7$ bank entry: the pilot text states only a positive behavior, "organize the answer so that its reasoning and conclusions are easy to follow", and is 21 tokens long, whereas the bank entry pairs a positive behavior with a matched failure mode like every other entry. The pilot comparison therefore varies factor identity and criterion specificity together, as appendix E.2 notes. The formal study of Sec. 5.2 uses the symmetric bank entries, which pair every factor with a matched failure mode.

# D Diagnosis Detail

## D.1 Demand Measurement Detail

Tab. 14 reports the full Qwen3-4B demand matrix with confidence intervals and effect sizes. Every cell of the seven-benchmark matrix is present; none is interpolated. Bold marks the cells the source matrix flags as meaningful. All values are model-specific and are not pooled across models. The matrix is stable across splits: under a rule that flags a cell when its diagnostic or held-out value moves by more than 0.06 or reverses sign while the full-data value is at least 0.10 in magnitude, no cell is flagged, and the largest cross-scope movement is 0.045. The Qwen3-8B comparison covers all seven benchmarks and is summarized in Tab. 16.

Tab. 13 gives the task-family summary used in Sec. 3.1. Verification and calibration separates success from failure about six times more strongly on the knowledge benchmarks than on the reasoning benchmarks (0.131 against 0.020), and factual and evidential grounding follows the same direction (0.111 against 0.031), while clarity and organization runs the other way (0.051 against 0.105). Within the knowledge group, ARC is driven by reasoning validity, TruthfulQA-MC by verification and factual grounding, and MMLU-Pro by completeness; MuSR is close to flat on every factor. A deployment target usually contains several benchmarks with different semantic needs, and how they are combined changes which factor appears most important: on the knowledge group the equal-per-benchmark average puts reasoning validity first (0.213) and completeness second (0.193), whereas pooling by response count reverses the order, because MMLU-Pro supplies about 86% of the responses in the group, raising completeness to 0.356. Neither summary is wrong, but a count-weighted average silently lets the largest benchmark decide the priority, so we keep the per-benchmark profiles and use the equal-per-benchmark average only as a descriptive summary.

**Model-dependent capability signatures.** Demand changes with the evaluated model as well as with the task. The Qwen3-4B and Qwen3-8B profiles correlate strongly on TruthfulQA-MC ($r = 0.953$), GSM8K ($r = 0.906$), and MMLU-Pro ($r = 0.868$), moderately on MuSR ($r = 0.522$), MATH-Hard ($r = 0.461$), and ARC ($r = 0.457$), and negatively on BBH ($r = -0.751$). The disagreement is not confined to one factor: on BBH, $f_5$, $f_7$, $f_6$, and $f_2$ all change sign, with $f_5$ separating success on Qwen3-4B ($+0.232$) but not on Qwen3-8B ($-0.100$), and $f_7$ behaving the same way ($+0.235$ against $-0.171$). The demand scope changes as well, and that matters more for a diagnosis than any single gap: reasoning validity is meaningful on five of the seven benchmarks for Qwen3-4B but on only one for Qwen3-8B, verification and calibration rises from none to two, and clarity and organization from two to three (Tab. 16). A priority read off the weaker model would not describe the stronger one, which supports a fixed ontology with a changing capability signature rather than one universal factor ranking, and it makes the grouping of Sec. 3.1 specific to Qwen3-4B. The two profiles share coordinates but have different distributions and numerical scales; we inspect them jointly without defining a difference or scalar mismatch score.

Table 13: Task-family summary of Fig. 4, by factor symbol (the factor names are in Sec. 2.1). Each entry is the unweighted mean of the benchmark gaps in that group, so every benchmark counts once regardless of how many responses it contributes. The last column is the knowledge mean minus the reasoning mean.

| Factor | Reasoning | Knowledge | Difference |
|---|---|---|---|
| $f_1$ | $+0.037$ | $+0.093$ | $+0.056$ |
| $f_2$ | $+0.132$ | $+0.213$ | $+0.081$ |
| $f_3$ | $+0.020$ | $+0.131$ | $+0.111$ |
| $f_4$ | $+0.031$ | $+0.111$ | $+0.080$ |
| $f_5$ | $+0.229$ | $+0.193$ | $-0.036$ |
| $f_6$ | $+0.059$ | $+0.022$ | $-0.037$ |
| $f_7$ | $+0.105$ | $+0.051$ | $-0.054$ |
| $f_8$ | $+0.101$ | $+0.170$ | $+0.069$ |

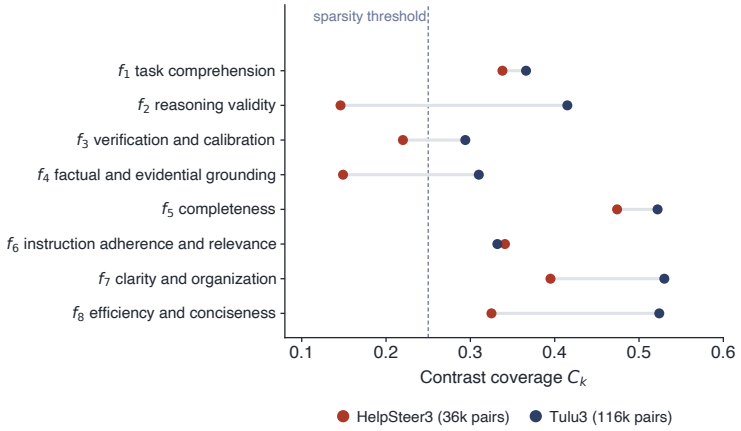



Figure 5: Contrast coverage in the two corpora, measured with the same frozen factor judge. $C_k$ is the fraction of pairs whose two responses differ on factor $f_k$; a dashed rule marks the $0.25$ sparsity threshold. HelpSteer3 leaves three factors sparse ($f_2$, $f_4$, $f_3$), while Tulu3 contrasts seven of the eight factors more often and has no factor below the threshold. The HelpSteer3 column is the same $38{,}173$ assessable pairs as Tab. 15; the Tulu3 column is a deterministic 1-in-6 sample ($19{,}376$ assessable pairs of $116{,}355$), so the columns are comparable but are sample statistics.

Table 14: Qwen3-4B demand detail over seven benchmarks (full diagnostic matrix). Gaps are on the 0–4 score scale, intervals are 95%, and $d$ is Cohen's $d$. Bold marks the meaningful flag in the source matrix.

| Benchmark | Factor | Gap | 95% CI | Cohen's $d$ |
|---|---|---|---|---|
| GSM8K | $f_1$ | $+0.080$ | $[0.057, 0.107]$ | $0.578$ |
| | $f_2$ | $+0.170$ | $[0.130, 0.214]$ | $0.756$ |
| | $f_3$ | $-0.051$ | $[-0.089, -0.013]$ | $-0.112$ |
| | $f_4$ | $+0.036$ | $[0.016, 0.059]$ | $0.319$ |
| | $f_5$ | $+0.158$ | $[0.123, 0.195]$ | $0.588$ |
| | $f_6$ | $+0.086$ | $[0.049, 0.126]$ | $0.270$ |
| | $f_7$ | $-0.048$ | $[-0.091, -0.006]$ | $-0.093$ |
| | $f_8$ | $+0.081$ | $[0.053, 0.109]$ | $0.222$ |
| MATH-Hard | $f_1$ | $+0.039$ | $[0.033, 0.046]$ | $0.326$ |
| | $f_2$ | $+0.246$ | $[0.228, 0.264]$ | $0.709$ |
| | $f_3$ | $+0.090$ | $[0.074, 0.108]$ | $0.167$ |
| | $f_4$ | $+0.054$ | $[0.044, 0.064]$ | $0.313$ |
| | $f_5$ | $+0.486$ | $[0.465, 0.506]$ | $1.105$ |
| | $f_6$ | $+0.065$ | $[0.054, 0.077]$ | $0.267$ |
| | $f_7$ | $+0.164$ | $[0.145, 0.181]$ | $0.315$ |
| | $f_8$ | $+0.162$ | $[0.145, 0.178]$ | $0.366$ |
| BBH | $f_1$ | $+0.023$ | $[0.011, 0.034]$ | $0.103$ |
| | $f_2$ | $+0.066$ | $[0.048, 0.083]$ | $0.160$ |
| | $f_3$ | $+0.018$ | $[-0.016, 0.052]$ | $0.035$ |
| | $f_4$ | $+0.028$ | $[0.024, 0.031]$ | $0.124$ |
| | $f_5$ | $+0.232$ | $[0.203, 0.259]$ | $0.408$ |
| | $f_6$ | $+0.075$ | $[0.060, 0.090]$ | $0.200$ |
| | $f_7$ | $+0.235$ | $[0.198, 0.271]$ | $0.447$ |
| | $f_8$ | $+0.095$ | $[0.068, 0.121]$ | $0.196$ |
| MuSR | $f_1$ | $+0.005$ | $[-0.013, 0.023]$ | $0.021$ |
| | $f_2$ | $+0.047$ | $[0.019, 0.074]$ | $0.123$ |
| | $f_3$ | $+0.025$ | $[-0.013, 0.062]$ | $0.050$ |
| | $f_4$ | $+0.006$ | $[-0.003, 0.015]$ | $0.049$ |
| | $f_5$ | $+0.038$ | $[-0.004, 0.077]$ | $0.069$ |
| | $f_6$ | $+0.010$ | $[-0.010, 0.028]$ | $0.038$ |
| | $f_7$ | $+0.069$ | $[0.036, 0.103]$ | $0.147$ |
| | $f_8$ | $+0.069$ | $[0.031, 0.104]$ | $0.136$ |
| ARC | $f_1$ | $+0.110$ | $[0.076, 0.145]$ | $0.680$ |
| | $f_2$ | $+0.248$ | $[0.190, 0.310]$ | $0.988$ |
| | $f_3$ | $+0.105$ | $[0.058, 0.151]$ | $0.210$ |
| | $f_4$ | $+0.093$ | $[0.057, 0.132]$ | $0.570$ |
| | $f_5$ | $+0.148$ | $[0.096, 0.196]$ | $0.297$ |
| | $f_6$ | $-0.007$ | $[-0.042, 0.028]$ | $-0.023$ |
| | $f_7$ | $+0.040$ | $[-0.007, 0.086]$ | $0.090$ |
| | $f_8$ | $+0.179$ | $[0.128, 0.230]$ | $0.317$ |
| TruthfulQA-MC | $f_1$ | $+0.100$ | $[0.070, 0.129]$ | $0.280$ |
| | $f_2$ | $+0.157$ | $[0.118, 0.193]$ | $0.362$ |
| | $f_3$ | $+0.145$ | $[0.109, 0.183]$ | $0.288$ |
| | $f_4$ | $+0.150$ | $[0.113, 0.189]$ | $0.327$ |
| | $f_5$ | $+0.034$ | $[-0.005, 0.072]$ | $0.065$ |
| | $f_6$ | $+0.035$ | $[0.000, 0.068]$ | $0.081$ |
| | $f_7$ | $-0.020$ | $[-0.054, 0.016]$ | $-0.043$ |
| | $f_8$ | $+0.110$ | $[0.076, 0.144]$ | $0.232$ |
| MMLU-Pro | $f_1$ | $+0.069$ | $[0.064, 0.074]$ | $0.275$ |
| | $f_2$ | $+0.235$ | $[0.225, 0.245]$ | $0.485$ |
| | $f_3$ | $+0.142$ | $[0.133, 0.151]$ | $0.283$ |
| | $f_4$ | $+0.091$ | $[0.084, 0.098]$ | $0.256$ |
| | $f_5$ | $+0.399$ | $[0.388, 0.410]$ | $0.700$ |
| | $f_6$ | $+0.038$ | $[0.032, 0.044]$ | $0.122$ |
| | $f_7$ | $+0.132$ | $[0.124, 0.141]$ | $0.278$ |
| | $f_8$ | $+0.219$ | $[0.209, 0.230]$ | $0.387$ |

Table 15: Semantic supply in the final 38,173 assessable HelpSteer3 pairs. $C_k$ is contrast coverage and $P_k$ is positive contrast coverage; the tie fraction is $1-C_k$, reported to three decimals, so rounding may leave a row summing to 0.999 or 1.001. The last column reports the reduction in mean contrast after length matching, available only for $f_2$, $f_5$, and $f_7$.

| Factor | $\mu_k$ | $C_k$ | $P_k$ | Tie | Reduction |
|---|---|---|---|---|---|
| $f_1$ Task comprehension | +0.104 | 0.338 | 0.212 | 0.662 | — |
| $f_2$ Reasoning validity | +0.091 | 0.146 | 0.103 | 0.854 | 13% |
| $f_3$ Verification and calibration | +0.082 | 0.220 | 0.151 | 0.780 | — |
| $f_4$ Factual and evidential grounding | +0.087 | 0.149 | 0.099 | 0.851 | — |
| $f_5$ Completeness | +0.156 | 0.474 | 0.304 | 0.526 | 38% |
| $f_6$ Instruction adherence and relevance | +0.099 | 0.341 | 0.205 | 0.660 | — |
| $f_7$ Clarity and organization | +0.143 | 0.395 | 0.261 | 0.605 | 47% |
| $f_8$ Efficiency and conciseness | +0.066 | 0.325 | 0.185 | 0.675 | — |

Table 16: Model dependence of demand. Left block: correlation between the Qwen3-4B and Qwen3-8B demand profiles within each benchmark (eight factors each). Right block: the number of the seven benchmarks on which each factor is flagged meaningful, for the two models. A scope change is a change in which tasks demand a behavior, not just in how strongly they do.

| Benchmark | $r$ | Factor | 4B / 8B |
|---|---|---|---|
| TruthfulQA-MC | +0.953 | $f_2$ Reasoning validity | 5/7 / 1/7 |
| GSM8K | +0.906 | $f_5$ Completeness | 4/7 / 2/7 |
| MMLU-Pro | +0.868 | $f_8$ Efficiency and conciseness | 3/7 / 1/7 |
| MuSR | +0.522 | $f_7$ Clarity and organization | 2/7 / 3/7 |
| MATH-Hard | +0.461 | $f_3$ Verification and calibration | 0/7 / 2/7 |
| ARC | +0.457 | $f_4$ Factual and evidential grounding | 1/7 / 0/7 |
| BBH | −0.751 | $f_1$, $f_6$ | 0/7 / 0/7 |

**Demand–supply regimes.** Three of the four cells of Fig. 6 contain a factor. Reasoning validity ($f_2$) combines broad demand with the lowest coverage of the eight (0.146) and the highest tie rate (0.854), and its contrast is comparatively length-mild, falling about 13% after length matching: the clearest case where the target tasks ask for a behavior the current data rarely demonstrates. Completeness ($f_5$) combines broad demand with the broadest coverage (0.474), though that contrast is strongly length-sensitive and falls about 38% after length matching, so the supply is abundant but not clean. Factual and evidential grounding ($f_4$) is demanded on one benchmark and sparsely contrasted ($C_k = 0.149$); verification and calibration ($f_3$) is the neighboring case with a similar family contrast and broader coverage (0.220). The fourth cell, specific demand with abundant supply, is empty: no factor occupies it in the current model–task–data state, and we leave it empty rather than relabel a factor to fill it. Efficiency and conciseness ($f_8$) is the boundary case, drawn with an open marker: its scope of three clears the broad cut by one benchmark, and its coverage of 0.325 sits between the two supply cuts. Clarity and organization ($f_7$) is meaningful on two benchmarks, is driven mainly by BBH, changes sign between models, and is among the most length-sensitive factors we measured (appendix D.2). The placement is specific to Qwen3-4B: the same figure drawn for the stronger model would move factors between cells, since the regimes describe a model–task–data state rather than a property of the factor or of the benchmark.

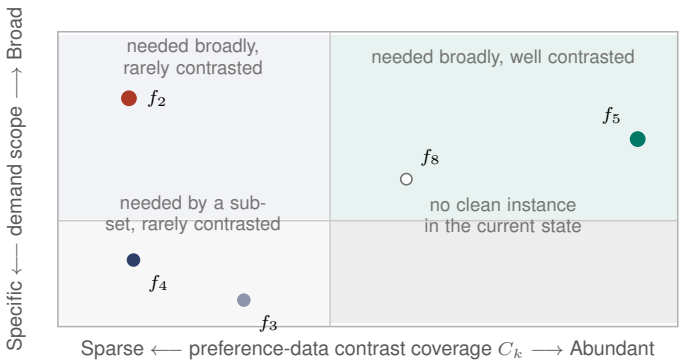



Figure 6: Demand scope against supply availability for Qwen3-4B. The horizontal axis is HelpSteer3 contrast coverage $C_k$; the vertical axis is the number of the seven benchmarks on which the factor is flagged meaningful. Divisors are placed in the gaps left by the two cut rules (coverage 0.25 and 0.30, scope one and three), not fitted to the points. Three cells contain a factor: reasoning validity ($f_2$), completeness ($f_5$), and the factual/verification pair ($f_4$, $f_3$). The fourth cell is empty in the current state. The open marker is efficiency and conciseness ($f_8$), which clears the broad cut by one benchmark and the sparse cut by 0.075. The two axes are not subtracted, and the placement is specific to this model, these tasks, and this dataset.

### D.2 Why Clarity and Organization Is Read Cautiously

Clarity and organization ($f_7$) sits in the specific group by the coverage rule of Sec. 3.1, and it was also the control criterion in the pilot. Four measured properties are why we read any $f_7$-specific result, in either role, with more caution than the others. Its demand scope is mixed: it is meaningful on two of seven benchmarks, and BBH supplies most of that signal. Its sign changes between models, separating success on Qwen3-4B ($+0.235$ on BBH) and not on Qwen3-8B ($-0.171$). Its contrast is the most length-sensitive we measured, falling about $47\%$ after length matching. And its supply is broad ($C = 0.395$), so it does not illustrate the sparse-supply case either. The judge-agreement audit is also weakest for the style-adjacent factors, with clarity at 0.633. It remains useful as a pilot control precisely because it is broad in the data and weakly demanded by the tasks: if criterion identity had no behavioral consequence, an $f_7$ criterion should have looked like ordinary DPO. It did not (appendix E.2). In the formal study, $f_7$ enters the specific set by the same coverage rule that places the other narrow-scope factors there, and we do not single it out as an exemplar.

### D.3 RQ4: How Large Are the Paired Effects?

Every effect reported above is an item-paired comparison, computed within one model–substrate cell on identical items. Tab. 19 gives the difference, its $95\%$ confidence interval and the exact paired $p$-value for each contrast the paper reads as a result, alongside the contrasts that do not separate.

Four rows carry a positive result and all four are on BBH, except the single MATH-Hard cell in the eight-billion second-substrate block. The largest contrast is not the wording edit but the continuation: one further epoch under the unchanged rubric adds $+8.15$ BBH points over the first-round policy ($p < 0.0001$), which is why the edit is read against that continuation and not against the starting policy.

## E Fixed-Rubric Results Detail

### E.1 Formal Per-Benchmark Differences

Tab. 17 states the comparisons fixed in advance for the diagnosis-guided multi-factor study, and the question each one answers.

Table 17: Comparisons fixed in advance for the diagnosis-guided multi-factor study, and the question each one answers. Every comparison is between two conditions trained on the same pairs, labels, objective, budget, and schedule, so the criterion set is the only difference.

| Comparison | Question |
|---|---|
| A: Common − Ordinary | Do the broadly demanded capabilities help broadly? |
| B: Common + Specific − Common | Do the specific capabilities add value beyond the common ones? |
| C: Common + Low − Common | Do the low-demand capabilities provide the same benefit when added? |
| D: RUBRICOPT-R1− Common + Specific | Does adding the low-demand set help, or does it dilute, once the specific set is present? |
| E: Common + Specific vs. Common + Low | Is any benefit attributable to the diagnosis, or would any additional semantic instruction do? |

Tab. 18 gives the per-benchmark differences behind Sec. 5.2: the spread across conditions is concentrated on BBH, and the other six columns move together.

Table 18: Per-benchmark differences between the trained conditions, in percentage points. The full six-condition table is Tab. 6.

| Comparison | GSM8K | MATH | BBH | MuSR | ARC | TQA | MMLU | Avg. |
|---|---|---|---|---|---|---|---|---|
| A Common − Ordinary | −0.53 | +0.97 | −0.03 | 0.00 | +0.60 | −0.19 | −0.07 | +0.11 |
| B +Specific − Common | 0.00 | −0.91 | −0.17 | −0.13 | +0.17 | −0.01 | +0.02 | −0.15 |
| C +Low − Common | −0.16 | −0.41 | −0.19 | 0.00 | −0.09 | −0.03 | +0.03 | −0.12 |
| D (+Spec+Low) − (+Spec) | +0.45 | +0.46 | +0.02 | +0.26 | −0.09 | +0.03 | 0.00 | +0.17 |
| E +Spec vs. +Low | +0.16 | −0.50 | +0.02 | −0.13 | +0.26 | +0.02 | −0.01 | −0.03 |

To isolate the $f_4$ wording change, both continuation branches start from the same frozen RUBRICOPT-R1, train for one epoch on the same 36,264 pairs, share the optimizer configuration, seed, exposure mask, and reference, and differ only in whether conditioning uses $R_0$ or $R_1$. The comparison between the two branches measures the edit; comparison with the starting policy measures the effect of continuing preference optimization.

One further epoch under unchanged $R_0$ adds 8.15 BBH points over the starting policy ($p < 0.0001$) and is flat on MATH-Hard ($-0.15$, $p = 0.82$). Relative to that matched continuation, the edited $R_1$ branch changes BBH by $-0.50$ points ($p = 0.56$, 95% CI $[-2.1, +1.1]$) and MATH-Hard by $+0.05$ ($p = 1.00$, CI $[-0.7, +0.8]$). On the open-ended protocols, the edit increment is $+0.003$ on Arena-Hard and $+0.098$ on WildBench, with intervals spanning zero; MT-Bench is $+0.406$, with a bootstrap interval above zero but paired rank-test $p = 0.061$. These results attribute the BBH gain to continued training under the rubric rather than to this wording edit. Full paired statistics are in Tab. 7.

The loop also exhibits both recursive outputs. A second 4B pass processes feedback from the refined branch and proposes changes to five factors. The gate rejects $f_3$ for insufficient anchor retention and $f_6$ for missing diagnostic support, leaving accepted $f_4$, $f_5$, and $f_7$ clauses. This $R_2$ candidate has not yet been trained, so it demonstrates recursive editing rather than a performance gain. On Qwen3-8B, the same operator processes the seven official benchmark cells. Although synthesis identifies improvement directions, attribution finds that the failures' concrete requirements are already stated by $R_0$; unsupported proposals fail the gate and the operator returns KEEP for every factor under that feedback set. KEEP is therefore a feedback-specific decision, not a claim that the specification is globally complete.

Table 19: Item-level paired comparisons, each computed within one model×substrate cell. Effects whose interval excludes zero are bold. The three rows that carry a positive result all sit on BBH; MATH-Hard contributes exactly one, and it is in the eight-billion second-substrate cell.

| Cell | Benchmark | Comparison | $\Delta$ | 95% CI | $p$ | Reading |
|---|---|---|---|---|---|---|
| 4B × HS3 | MATH-Hard | Common − ordinary | $+0.0035$ | $[-0.0065, +0.0140]$ | $0.51$ | null |
| 4B × HS3 | MATH-Hard | RUBRICOPT-R1 − DPO | $+0.0075$ | $[-0.0010, +0.0165]$ | $0.11$ | null |
| 4B × HS3 | MATH-Hard | $R_1 - R_0$ | $+0.0005$ | $[-0.0070, +0.0080]$ | $1.00$ | null |
| 4B × Tulu3 | MATH-Hard | rubric − ordinary | $+0.0005$ | $[-0.0085, +0.0095]$ | $1.00$ | null |
| 8B × HS3 | MATH-Hard | RUBRICOPT-R1 − ordinary | $-0.0030$ | $[-0.0120, +0.0060]$ | $0.59$ | null |
| 8B × Tulu3 | MATH-Hard | **rubric − ordinary** | $\mathbf{+0.0170}$ | $[+0.0070, +0.0270]$ | $\mathbf{0.0003}$ | positive |
| 4B × HS3 | BBH | **Common − ordinary** | $\mathbf{+0.0230}$ | $[+0.0065, +0.0395]$ | $\mathbf{0.0053}$ | positive |
| 4B × HS3 | BBH | RUBRICOPT-R1 − DPO | $+0.0110$ | $[-0.0045, +0.0265]$ | $0.18$ | null |
| 4B × HS3 | BBH | RUBRICOPT-R1 − Common | $-0.0120$ | $[-0.0275, +0.0030]$ | $0.14$ | null |
| 4B × HS3 | BBH | $R_1 - R_0$ | $-0.0050$ | $[-0.0210, +0.0105]$ | $0.56$ | null |
| 4B × HS3 | BBH | $R_0$ − **RUBRICOPT-R1** | $\mathbf{+0.0815}$ | $[+0.0605, +0.1025]$ | $< \mathbf{0.0001}$ | positive |
| 4B × Tulu3 | BBH | **rubric − ordinary** | $\mathbf{+0.0835}$ | $[+0.0555, +0.1120]$ | $< \mathbf{0.0001}$ | positive |
| 8B × HS3 | BBH | RUBRICOPT-R1 − ordinary | $+0.0130$ | $[-0.0035, +0.0300]$ | $0.13$ | null |
| 8B × Tulu3 | BBH | **rubric − ordinary** | $\mathbf{+0.0380}$ | $[+0.0190, +0.0565]$ | $< \mathbf{0.0001}$ | positive |

### E.2 DEVELOPMENT PILOT: CRITERION IDENTITY CHANGES LEARNED BEHAVIOR

This is development evidence, not a main result. It establishes whether criterion identity has behavioral consequences before the full-epoch study was run, at a training scale that cannot support a performance claim, and it is reported here so that the formal study's design choices have a visible motivation.

The pilot compares ordinary DPO, a single-factor $f_2$ criterion, and a single-factor $f_7$ criterion while holding the pairs, labels, DPO objective, data order, and LoRA configuration fixed. It trains Qwen3-4B for $150$ steps at a fixed conditioning rate $q = 0.2$, so the $f_2$ and $f_7$ conditions condition exactly the same training rows and the comparison varies the criterion text alone. The three conditions reach comparable optimization magnitude: final losses are $0.6797$ for ordinary DPO, $0.6836$ for the $f_2$ criterion, and $0.6836$ for the $f_7$ criterion, with final margins of $0.0254$, $0.0215$, and $0.0195$.

Under the paired pilot protocol, the $f_2$ criterion is favored over the $f_7$ criterion on all four pass@4 comparisons (Tab. 20). Paired tests detect a difference on GSM8K (permutation $p = 0.002$) and on MuSR pass@4 (McNemar $p = 0.035$, with 17 passes unique to $f_2$ against 6 unique to $f_7$). MATH-Hard and BBH show the same direction without a significant difference. The $f_7$ criterion also shifts negatively relative to ordinary DPO on MuSR pass@4 (McNemar $p = 0.0127$, 3 passes unique to $f_7$ against 14 unique to ordinary DPO). The $f_2$ criterion and ordinary DPO do not differ significantly on any of the four benchmarks, and ordinary DPO does not differ significantly from the base model on any of them (Tab. 20).

The supported observation is a feasibility result about the training interface: changing the criterion text alone produced direction-consistent downstream differences: the $f_2$ criterion separates from ordinary DPO on the benchmark the diagnosis marks as its strongest, and the $f_7$ criterion moves where the diagnosis does not support it.

This pilot tests directional feasibility at limited training scale, covering about $0.20$ epoch. Multiple paired tests make the nominal findings exploratory, and no multiple-testing correction is applied.

Table 20: Exploratory paired evidence from the 150-step development pilot, using the paired pilot protocol rather than the official protocol. Reported $p$-values are interpreted at the nominal level across multiple comparisons, not as a family-wise confirmatory result. No cell here may be read as a performance claim for the pilot conditions.

| Comparison | Paired evidence |
|---|---|
| $f_2$ vs. $f_7$ | $f_2$ higher on all four pass@4 comparisons; GSM8K permutation $p = 0.002$; MuSR pass@4 McNemar $p = 0.035$; MATH-Hard and BBH not significant. |
| $f_7$ vs. Ordinary DPO | $f_7$ lower on MuSR pass@4; McNemar $p = 0.0127$. |
| $f_2$ vs. Ordinary DPO | No significant difference on any of the four. |
| Ordinary DPO vs. base | No significant difference on any of the four. |

The $f_2$ and $f_7$ pilot criteria also differ in specificity: the $f_2$ criterion names both a positive behavior and a matched failure mode, while the $f_7$ pilot criterion names only the positive behavior. Factor identity and criterion specificity are therefore not fully disentangled in this pilot, which is one reason the formal study uses the symmetric two-field criterion bank of appendix C.3. The pilot measures task scores, not the factor profile of the trained policies; **??** reports that measurement directly.

### E.3 Experimental Details

**Judge.** The diagnostic models are Qwen3-4B and Qwen3-8B, scored by the Qwen3-8B judge described in appendix B.3.

**Pilot raw scores.** Tab. 21 reports the scores behind the paired tests of appendix E.2.

Table 21: Pilot raw scores, reported as accuracy and pass@4. The base row is the untrained starting model; the three trained conditions share pairs, labels, objective, data order, and LoRA configuration, and differ only in the training criterion. These small differences are paired evidence about criterion identity, not a standalone performance ranking, and they belong to the paired pilot protocol.

| Benchmark | Base | Ordinary DPO | $f_2$ criterion | $f_7$ criterion |
|---|---|---|---|---|
| GSM8K accuracy | 0.8945 | 0.8955 | 0.9015 | 0.8910 |
| GSM8K pass@4 | 0.928 | 0.928 | 0.932 | 0.926 |
| MATH-Hard accuracy | 0.9070 | 0.9105 | 0.9125 | 0.9095 |
| MATH-Hard pass@4 | 0.964 | 0.964 | 0.966 | 0.962 |
| MuSR accuracy | 0.5870 | 0.5825 | 0.5795 | 0.5840 |
| MuSR pass@4 | 0.706 | 0.708 | 0.708 | 0.686 |
| BBH accuracy | 0.0520 | 0.0535 | 0.0560 | 0.0495 |
| BBH pass@4 | 0.142 | 0.126 | 0.142 | 0.130 |

**Supply data and length matching.** The supply analysis uses the 38,173 assessable pairs from 38,181 raw HelpSteer3 pairs, a 99.98% assessable rate. Chosen responses are longer, so we also report a length-matched subset: pairs whose chosen and rejected responses differ by at most 164 characters, the lower-quartile bound of the length-difference distribution, giving $n = 9{,}551$. For each factor we additionally fit $\Delta z_k = \alpha + \gamma\, z(\Delta L)$ over the full set and report $\gamma$ per standard deviation of the length difference; across the factors we examined, $\gamma$ is 0.040 for reasoning validity, 0.021 for factual grounding, 0.175 for completeness, 0.105 for clarity, and $-0.177$ for efficiency.

**Pilot training.** The pilot trains Qwen3-4B with LoRA (rank 64, $\alpha = 128$, dropout 0.05) and DPO $\beta = 1.0$ at learning rate $10^{-6}$ for 150 steps with seed 42, at a fixed conditioning rate $q = 0.2$, using `torchrun` DDP on 6× Ascend 910B1. The effective batch is $2 \times 4 \times 6 = 48$ pairs per optimizer step, so a run draws about 7,200 pairs, roughly 0.20 epoch of the 36,264 rows. Of the 36,264 rows, 7,173 are criterion-conditioned and 29,091 use the original prompt; the $f_2$ and $f_7$ conditions condition exactly the same row indices, verified row by row. Maximum prompt length

is 1,024 tokens and total sequence length is 2,048 in bf16 with FlashAttention-2. When a prompt is truncated, the final 1,024 tokens are kept, so the criterion at the end of the prompt is not the part that is dropped. The reference model is the base model with the LoRA adapter disabled, evaluated without gradient tracking on the same batch tensors, and recomputed at every step with no log-probability cache. Run logs imply about 47 minutes per run. No separate cost record was kept, so this is a wall-clock reading inferred from the logs rather than a formal accounting.

**Formal training.** Each diagnosis-guided condition and the ordinary-DPO baseline train for one full epoch over 36,264 preference pairs, which is 566 optimizer steps at a global batch of 64 pairs, on eight Ascend NPUs under distributed data parallel. LoRA uses rank 64, $\alpha = 128$, and dropout 0.05; DPO uses $\beta = 1.0$; the learning rate is $10^{-6}$ under a frozen optimizer schedule; the seed is 42. Conditioning follows Eq. (5) with phase fractions 0.10/0.80/0.10, giving a realized cumulative criterion exposure of 0.5009. Evaluation is criterion-free.

**Splits.** Diagnostic and held-out halves are assigned by an `md5` hash of the prompt identifier, so the two sets are disjoint by construction. The pilot evaluation prompts come from a different source than the HelpSteer3 training pairs, so there is no overlap between them. The two demand matrices agree on every cross-scope cell we checked.

**Pilot evaluation and paired tests.** Pilot evaluation draws 500 prompts per benchmark with 4 samples per prompt at temperature 0.7, top-$p$ 0.8, and top-$k$ 20, a schedule used for this pilot rather than the formal protocol. The unit of analysis is the prompt. McNemar tests are exact and use the pass@4 indicator; the sign-flip permutation test uses 10,000 resamples with seed 20260904, is two-sided, and acts on the paired per-prompt mean-correctness differences. Tab. 22 lists the discordant counts behind the McNemar results. No formal multiple-testing correction is applied across the 24 tests, so the nominal $p$-values in Tab. 20 are exploratory; the pilot reading rests on the direction consistency and on the significant negative control.

Table 22: Pilot paired pass@4 discordant counts, by benchmark. An entry $a/b$ means $a$ prompts passed only under the first condition and $b$ only under the second. These counts underlie the exact McNemar tests reported in the main text.

| Benchmark | $f_2$ vs $f_7$ | $f_7$ vs ordinary |
|---|---|---|
| GSM8K | 4/1 | 4/5 |
| MATH-Hard | 4/2 | 2/3 |
| MuSR | 17/6 | 3/14 |
| BBH | 23/17 | 15/13 |

**WildBench tag decomposition.** On the full eight-factor arm, coding and debugging falls by 0.14 ($n = 86$), math by 0.32 ($n = 57$), data analysis by 0.16, and information seeking by 0.10, while editing gains 0.13 and reasoning is flat. This is the signature of a checklist-style exactness that WildBench's tags penalize and that other protocols reward; it is the only decomposition we extracted from the evaluation artifacts, and it is a description of the direction rather than a causal attribution to a factor.

**Judge history of the anchor files.** The base and ordinary-DPO anchor files of Tab. 2 were scored partly by an earlier judge version, which produced 11 to 13 ties where the formal arms produced none. The paired tests therefore use same-judge scores only, and no cross-judge comparison is made anywhere in this paper.

**Official evaluation.** The formal study uses the released benchmarks through the standard evaluation harness, with adapter evaluation enabled and the base model merged. The reported baselines use this protocol (Tab. 6). An earlier cross-check ran the three pilot conditions under a slightly different harness configuration and found no large regression, but its resolution was too small to adjudicate the pilot-scale differences and its baselines are not comparable to the formal protocol, so we do not tabulate it.

The demand values in Fig. 4 and Tab. 14 are for Qwen3-4B unless a row states otherwise. They are not pooled estimates.

# F RUBRICOPT DETAIL

## F.1 EVIDENCE SCALE, AUDIT TRACE, AND ENGINE RELIABILITY

Fig. 2 in the main text gives the five-stage workflow; this section records its evidence scale, engine behavior, and complete audit trace.

**Evidence scale.** The diagnostic stage reads item-level judge rationales for the all-eight-factor arm against ordinary DPO and the base model on Arena-Hard (500 items), MT-Bench (80 items, including a dedicated second-turn batch), and WildBench (256 items), together with generated correctness items carrying a gold answer (GSM8K 2,000, MATH-Hard 2,000, MMLU-Pro 2,000). The engine was called 20 times over 445 paired examples in the diagnostic stage, and the deterministic stages operated over more than 6,000 paired items.

**Engine.** The analysis and proposal engine is the current policy itself: the all-eight-factor merged checkpoint, served in bf16 with eager attention at temperature 0, so the model that reads the benchmark feedback is the model the feedback is about. The external anchors are the gold answers and benchmark outcomes, the fixed judge's rationales and checklists, and the frozen $R_0$ text with its recorded fingerprint.

**Audit trace.** The first candidate failed the gate. One violation was real: an $f_3$ prefer clause had dropped the phrase "in a calibrated way" from the frozen anchor. Two inspector queries were repaired, and one warning flagged $f_3$ importing vocabulary belonging to $f_2$. The violation-repair call returned the candidate unchanged, so the orchestrator resolved it mechanically, reverting the violating factor to KEEP and leaving the $f_4$ refinement in place, since that proposal was gate-compliant and its author remains the model. The resulting candidate passed the gate with zero problems and zero warnings, and an auxiliary model audit agreed with all eight decisions. This trace is the empirical basis for the design claim in Sec. 3.5: the deterministic gate caught a defect the model's own audit had passed, and the model could not repair it afterwards.

**Engine reliability.** One of the twenty diagnostic calls returned output without the required structured block and recovered on retry; the cross-benchmark synthesis required a completeness retry before it produced a usable signature. We record these because they bound how much of the procedure can be delegated to a 4B engine: diagnosis and proposal were usable, autonomous auditing and self-repair were not.

## F.2 MATCHED RE-OPTIMIZATION PROTOCOL AND SECOND FEEDBACK ROUND

Tab. 23 lists the frozen settings of the matched re-optimization described in Sec. 5.3. The two branches share a single exposure mask, so the criterion is drawn on exactly the same pairs in the same order in both; the only difference between them is the rubric text, whose seven other clauses are identical.

The second 4B feedback pass is the same five-stage procedure as round 1, applied to RUBRICOPT-R2 (REFINE) rather than to RUBRICOPT-R1. Its diagnostic stage reads the same protocols for the refined branch, its attribution stage compares each recurring failure against the $R_1$ wording rather than against $R_0$, and its acceptance gate applies the same eight eligibility and semantic-preservation checks as round 1, plus two round-2 procedural integrity checks (only gate-accepted factors may differ, and the decision ledger must be consistent with the rendered rubric; appendix G). The one difference in the input is the policy whose behavior is being diagnosed and the rubric the attribution is checked against; no eligibility rule or threshold was altered between rounds. The pass returned REFINE for five factors and, after the gate rejected two proposals, accepted three edits (Sec. E.1); its output $R_2$ differs from $R_1$ in exactly those clauses and has not been trained. An earlier pass over the same input returned KEEP on all eight factors (appendix H.3 records it), and the Qwen3-8B all-KEEP pass is a separate experiment (Sec. E.1).

Table 23: Frozen settings of the matched re-optimization. Both branches differ only in the rubric used for conditioning.

| Item | Setting |
|---|---|
| Initial policy | the all-eight-factor merged checkpoint of Sec. 5.2, shared by both branches |
| Data | 36,264 HelpSteer3 pairs, one epoch, 566 optimizer steps |
| Batch | global 64 pairs per step, identical in both branches |
| Optimizer | fresh AdamW, learning rate $10^{-6}$, cosine decay, 5% warmup |
| Preference objective | DPO with $\beta = 1.0$, mean-per-token log-probabilities |
| Adapters | fresh LoRA, rank 64, $\alpha = 128$, dropout 0.05, FP32 |
| Schedule | 10/80/10 progressive withdrawal, pair-level all-or-nothing |
| Exposure | one shared exposure mask, so both branches condition the same pairs |
| Reference | frozen initial policy with the adapter disabled, not the base model |
| Seed | 42 |
| Primary comparison | RUBRICOPT-R2 (REFINE) − RUBRICOPT-R2 (KEEP) |
| Secondary comparison | RUBRICOPT-R2 (REFINE) − RUBRICOPT-R1, the effect of continuing preference optimization rather than of rubric refinement; it is not read as evidence about the refinement |

### F.3 SUBSTRATE CONTROL

The HelpSteer3 experiments were designed with the risk that their findings are properties of one corpus. The substrate control of Sec. 5.1 addresses this: the same base model, objective, and optimizer configuration, and the same full eight-factor rubric $R_0$, on Tulu3 (Lambert et al., 2025), an independently assembled corpus of different size and composition, whose semantic supply profile is measured in Fig. 5. Budgets are matched within each corpus (one epoch: 566 optimizer steps over 36,264 pairs on HelpSteer3, 1,818 steps over 116,355 pairs on Tulu3), so the two substrates share the epoch count, not the step count. The control reproduces a conditioning gain on the second corpus (BBH +8.35 item-paired and a +1.8-point seven-benchmark average; Sec. 5.1) while its open-ended protocols move flat to nominally negative (appendix H.4), which is the source of the substrate-dependence reading in Sec. 5.3. Both the demand diagnosis of Sec. 3.1 and the specification and refinement machinery of Secs. 3.2 and 3.5 are defined for any such corpus, and the two supply profiles of Fig. 5 are what the diagnosis compares when it explains why the two substrates move different benchmark classes.

## G THE RUBRICOPT WORKFLOW IN FULL

Sec. 3.5 states what RubricOpt does and why each constraint exists. This appendix gives the operational specification: the five prompts that are actually executed, their output schemas, the deterministic gate that decides acceptance, and three executed rounds shown end to end so that the reader can see what a decision looks like in practice. The prompts are quoted as the contracts they are; the examples are drawn from the artefacts, not reconstructed.

### G.1 STAGES AND CONTRACTS

Five stages run in sequence against a policy $\pi_t$ and its frozen criterion $R_t$. Each is a separate model call with its own output schema, and no stage sees the next stage's output; the numbers come from the prompt files and follow execution order.

The contracts are narrow on purpose. Stage 1 sees the factor schema, the criterion currently in force, and one benchmark's evidence, and returns a four-key YAML document:

```
preserve:
  - behavior:  / factor:  / evidence:
improve:
```

Table 24: Execution order of the five stages. Each stage is a separate model call with its own schema; no stage sees the next stage's output.

| # | Stage | Consumed | Produced |
|---|---|---|---|
| 1 | benchmark diagnostic | one benchmark's evidence | behaviors to preserve / to improve / candidate gaps |
| 2 | cross-benchmark synthesis | all stage-1 reports | recurrence with an evidence-strength level |
| G | factor attribution | stage-2 report | a coverage judgment per factor |
| 3 | conservative update | stage-2 + G | a candidate $R_{t+1}$ |
| 4 | rubric audit | $R_t$, candidate, evidence | nine audit keys |
| E | deterministic gate | candidate + ledger | accept / revert |

```
    - failure_mode:  / factor:  / evidence:
  criterion_gaps:
    - factor:  / current_problem:  / suggested_direction:
  overall_diagnosis:
```

Five rules travel with it: use only $f_1$–$f_8$ and invent no factors; separate the behaviours to preserve from the behaviours to improve; do not infer a problem from a single isolated example unless the evidence is strong; do not attribute a benchmark-score change to the criterion without behavioural evidence; and keep the analysis concrete and concise. The third rule does the most work, because it is what stops a vivid single example from becoming a rubric edit.

Stage 2 aggregates every stage-1 report for one policy. It may not rewrite the criterion or propose wording; it organises evidence and records disagreement, adding the benchmarks that support each pattern, an evidence-strength level (strong, moderate, weak), and an explicit record of contradictions. A benchmark where the policy did not change still counts as evidence about what is stable.

Stage G attributes each recurring behaviour onto the ontology and decides whether the current wording already covers it. This is what separates RubricOpt from a procedure that appends an instruction whenever something goes wrong, and the rule is stated as a two-way implication:

> If the current Prefer/Avoid wording states this failure mode in full, the model simply did not learn it: the decision is KEEP. If the wording does not state the failure mode, or states only part of it, *that is the only case in which* REFINE *may be considered.*

Attribution must also disambiguate the four factors a failure can plausibly sit in—reasoning validity, verification and calibration, completeness, and instruction adherence—and name the primary one when several are involved. It attributes only what the evidence supports.

Stage 3 is described to the engine as a *small optimisation step* under ten numbered rules. Three are conservative: preserve the factor names and the $f_1$–$f_8$ ontology, preserve the behaviours marked for preservation, and address only high-confidence recurring problems. Six bound the edit: prefer small edits that clarify wording, narrow an overly broad reading, make an implicit requirement explicit, or refine the matched Avoid clause; do not redesign a factor; introduce no benchmark names, examples, scores, datasets or task-specific tricks; add no numerical weights or intensifiers such as MUST or CRITICAL; keep one Prefer clause and one matched Avoid clause per factor; and modify as few factors and words as possible. The tenth rule is the stopping condition: *if the evidence does not justify a change, keep the original wording*. The output is a candidate $R_{t+1}$ plus a per-factor ledger carrying the action, the reason, and the old and new Prefer and Avoid clauses.

Stage 4 audits the candidate against the original, against itself, and against the evidence, on eight axes—semantic identity preserved, benchmark leakage, dataset leakage, unsupported edit, over-specificity, matched Prefer/Avoid pair, preserve constraints respected, and edit magnitude—emitting one line of justification per axis and a single `PASS` or `FAIL`. The audit is advisory rather than decisive: in one executed round it returned a clean eight-of-eight `PASS` on a candidate that the deterministic gate then rejected.

### G.2 THE DETERMINISTIC GATE

Acceptance is decided by eight checks in round 1 and ten in round 2; round 2 adds that only gate-accepted factors may differ, and that the decision ledger is consistent with the rendered rubric.

Table 25: The deterministic acceptance checks. The byte-match and recall checks are what make the edit a small step rather than a rewrite: the input clauses must match the frozen wording byte-for-byte, and the output must retain at least $60\%$ of the input's content tokens while staying within a ten-word budget.

| Check | Failure behaviour |
|---|---|
| all eight factors present exactly once with a KEEP/REFINE decision | missing factor ⇒ reject |
| refinement input clauses match the frozen canonical wording byte-for-byte | mismatch ⇒ reject |
| content-token recall $\geq 0.60$ and clause length within $[-5, +10]$ words | identity lost ⇒ reject |
| one matched Prefer/Avoid pair per factor | unpaired ⇒ reject |
| no benchmark or dataset names, digits, intensifiers, example markers | forbidden token ⇒ reject |
| every refinement traced to an attribution judging the failure not fully covered | untraceable ⇒ reject |
| edit magnitude small | over budget ⇒ reject |
| cross-factor vocabulary imports | warning only |
| (round 2) only gate-accepted factors may differ | unaccepted edit ⇒ reject |
| (round 2) the decision ledger is consistent with the rendered rubric | mismatch ⇒ reject |

A factor that violates a check is reverted to KEEP, and the repair attempt is a separate model call. When the repair fails, the orchestrator reverts the factor mechanically and records the revert; it never re-issues a failing edit as if the gate had passed it.

### G.3 THREE EXECUTED ROUNDS

**The accepted $f_4$ edit.** Round 1 diagnosed the four-billion-parameter eight-factor policy from nineteen engine calls over twenty batches drawn from eight paired pools. Stage 2 marked $f_3$ and $f_4$ as recurring with strong evidence, and judged $f_4$ not fully covered at moderate confidence: its canonical wording named invented facts and unsupported assertions, but not the failure the feedback kept producing—claims justified by an appeal to "common knowledge" that was never checked. Stage 3's first candidate also rewrote $f_3$; the gate rejected that clause because the new Prefer did not match the frozen wording, and the orchestrator reverted $f_3$ and kept $f_4$. The accepted edit is the whole of $R_1$:

```
f4 factual and evidential grounding
Prefer: factual claims grounded in the given information or
explicitly sourced common knowledge
Avoid: invented facts, unsupported assertions, or
inferences from vague common knowledge
```

Seven words were added across two clauses, and content recall against the input was $1.00$. The $f_3$ revert carries the design: $f_3$ had the strongest failure signal of any factor, and it was kept because its Avoid clause already named unchecked steps—the policy had not learned a behaviour the rubric already specified, and the gate refused to restate it.

**A round that edits three factors.** The second four-billion round ran against RUBRICOPT-R2 (REFINE). Attribution returned REFINE for $f_4$, $f_5$, $f_3$, $f_6$ and $f_7$, and stage 3 proposed edits to all five. The gate rejected two: the $f_3$ Prefer clause fell to a content recall of $0.57$

against the required 0.60, and the $f_6$ refinement was not cited by the cross-benchmark diagnosis. The repair pass fixed neither, and the orchestrator reverted both, leaving three accepted edits—$f_4$'s Prefer gained "from known standards", $f_5$'s Prefer gained "with no truncation", and $f_7$'s clauses gained a persona clause. The candidate differs from round 1's output in exactly these three clauses.

**A round that changes nothing.** The same pipeline ran against an eight-billion-parameter policy. Attribution judged every factor it attributed fully covered, with strong confidence for five of them, and returned zero REFINE candidates, while stage 2 nonetheless named improvement directions for four factors—precisely the disagreement the gate exists to arbitrate. Stage 3 proposed refinements to three factors, and the gate rejected the pass with fifteen violations: four factors missing from the ledger, three REFINE inputs that did not match the frozen wording, three REFINEs unsupported by attribution, and one proposed clause containing a digit. The repair pass returned all eight factors as KEEP and passed cleanly, leaving the rubric unchanged. The round is therefore a no-op reported as a no-op: the specification was already adequate at the resolution of this feedback, so no edit was licensed, and the disagreement between stage 2 and stage G is resolved in favour of attribution by construction.

# H Additional Results and Protocol Details

This appendix gives the measurement and protocol detail behind the main-text results: evaluation protocols, training-protocol audits, and the per-cell measurements that the main tables summarize.

## H.1 Arena-Hard: protocol and win rates

**The judge and the comparison.** Every condition is judged against one fixed anchor answer (the standard Arena-Hard baseline) with the candidate always in slot B, so Arena-Hard here is a *fixed-anchor win rate*, not a head-to-head between arms. Per-item outcomes are binary (beat the anchor, or not), with ties counted as half-wins in the win-rate metric; a binary exact McNemar applied to a half-win metric requires the tie rule to be stated, and the RubricOpt-R2 (REFINE) − RubricOpt-R2 (KEEP) comparison is therefore reported separately (Sec. E.1) rather than merged into this table.

## H.2 Training protocol details

The loss is DPO with $\beta = 1.0$, $-\log \sigma(\beta[(\log \pi_\theta - \log \pi_{\text{ref}})^+ - (\log \pi_\theta - \log \pi_{\text{ref}})^-])$, and the reference is recomputed every step with the adapter disabled and no log-probability cache. Padding is excluded by the label mask. The trainer asserts that conditioning leaves the response tokens unchanged, that the criterion suffix appears in the conditioned prompt and not in the unconditioned one, and that the criterion lands inside the masked prompt region; it prints the masked-token delta (+43 for the single-factor pilot block, 43–64 for the formal blocks), so criterion tokens enter neither the numerator nor the denominator of the margin. The exposure mask is identical across the two continuation branches, and both start from the same frozen RubricOpt-R1 checkpoint, which is what the matched-refinement comparison requires.

**Semantic content against generic training context.** Because every criterion arm gains on Arena-Hard and the gains do not order by factor set, the question arises whether any extra context would do the same. Three partial controls exist; the one that would settle the question does not.

Table 26: Composition and length of the four formal criterion blocks, measured with the Qwen3-4B tokenizer. The four arms span 70 to 234 suffix tokens.

| Arm | Factors | Clauses | Suffix tokens | Avg. 7-bench |
|---|---|---|---|---|
| Common | $f_2$, $f_5$ | 2/2 | 70 | 58.46 |
| Common + Specific | $f_2$, $f_4$, $f_5$, $f_7$, $f_8$ | 5/5 | 151 | 58.31 |
| Common + Low | $f_1$, $f_2$, $f_3$, $f_5$, $f_6$ | 5/5 | 153 | 58.34 |
| RubricOpt-R1 | All eight | 8/8 | 234 | 58.48 |

The strongest available contrast is Common+Specific against Common+Low: identical clause count ($5/5$), near-identical suffix length (151 against 153 tokens, $1.3\%$ apart), and a different second factor set—the diagnosis-selected narrow-scope factors against the zero-demand factors. On the seven official benchmarks they are indistinguishable ($+0.15$, $-0.51$, $+0.02$, $-0.13$, $+0.26$, $+0.02$, $-0.01$ percentage points; average $-0.03$). The diagnosis describes the model–task–data state rather than predicting where the gains land.

**The one real context control.** The pilot's $f_7$ arm is a same-interface, same-conditioned-row-indices ($7{,}173/36{,}264$), same-$q$ criterion that differs from the $f_2$ arm only in its text (21 against 22 tokens). Against ordinary DPO it moves *down* on MuSR pass@4 ($p = 0.0127$, discordant $3/14$) while $f_2$ moves up on all four benchmarks, so a matched-length context is not sufficient to reproduce the $f_2$ direction at pilot scale, so at pilot scale the direction tracks the criterion's text rather than the presence of a matched-length context.

**Judge.** All factor scores come from one blind judge, a Qwen3-8B model running one version-tagged prompt, and its reproducibility is measured rather than assumed. Tab. 27 reports agreement against an independent rescorer, exact agreement and ordinal correlation on 240 repeated responses, and the inter-factor correlation that shows the eight axes do not collapse onto a single quality scale. The judge's configuration is in appendix B.3.

Table 27: Measurement quality of the factor judge. Agreement is the blind pairwise audit against the direction implied by independent per-response scores; the decisive column is the number of comparisons that could enter it. Repeat scoring is a separate experiment on 240 shared responses.

| Quantity | Unit | Value | Denominator | Note |
|---|---|---|---|---|
| Decisive-only agreement, mean | factor | 0.741 | 868 | over eight factors |
| Agreement, highest | $f_1$ task comprehension | 0.909 | 11 | small denominator |
| Agreement, lowest | $f_8$ efficiency | 0.559 | 247 | largest denominator |
| Repeat scoring, exact match | factor | 0.846–0.992 | 240 | per factor |
| Repeat scoring, Pearson | factor | 0.530–0.905 | 240 | weakest $f_4$ |
| Mean score shift on repeat | factor | $\leq 0.025$ | 240 | every factor |
| Inter-factor correlation, mean $\lvert r\rvert$ | pair | 0.195 | 108,685 | production prompt |
| Inter-factor correlation, max $\lvert r\rvert$ | pair | 0.362 | 108,685 | production prompt |
| Collapse control, max $\lvert r\rvert$ | pair | 1.000 | 288 | prior prompt revision |
| Distinct score vectors | prompt | 822 vs 4 | — | production vs control |

### H.3 The RubricOpt trace

**The pipeline that actually ran.** Five stages, in execution order: per-batch benchmark diagnostic; cross-benchmark synthesis; factor attribution; conservative criterion update; rubric audit. Stage 2 may only raise an issue that recurs across samples *and* benchmarks; an isolated example is explicitly not a rubric gap. Attribution is the gate-keeper: a failure the current wording states in full yields KEEP, and only a failure judged not fully covered licenses REFINE.

**The gate is deterministic and it caught the model.** Round 1's binding gate applies eight checks: all eight factors present exactly once with a KEEP/REFINE decision; the REFINE input clauses must match the frozen canonical wording byte-for-byte; content-token recall from old to new clause $\geq 0.60$ with the clause length inside $[-5, +10]$ words; one matched prefer/avoid pair per factor; no benchmark names, digits, intensifiers or example markers; every REFINE factor traceable to an attribution judging the failure not fully covered; bounded edit magnitude; and cross-factor vocabulary imports as warnings. Round 2 adds two checks, the last being that the decision ledger matches the rendered rubric by recomputed hash.

Table 28: Round-1 edit ledger. The model's own audit passed all eight checks on a candidate the deterministic gate then rejected, which is the clearest available evidence that the gate is load-bearing.

| Cycle | Outcome | Reason |
|---|---|---|
| Candidate 1 | gate FAIL | the proposed $f_3$ prefer clause does not match the frozen wording (the model dropped "in a calibrated way"); warning: the new $f_3$ text imports $f_2$ vocabulary |
| Repair pass | gate FAIL | model returned the candidate unchanged |
| Resolution | $f_3 \rightarrow$ KEEP | evidence also occurs outside technical domains; not supportable |
| Final | $f_4$ REFINE | recall 1.00, no foreign vocabulary, attribution-traceable |

The accepted edit adds seven words across the two clauses and is quoted in Sec. 5.3.

**An early 4B pass that returned all-KEEP.** The all-KEEP pass recorded here is an early Qwen3-4B pass: its input policy is RUBRICOPT-R2 (REFINE), its input rubric is $R_1$, and it leaves $R_1$ unchanged. It is distinct from both the later 4B pass that accepted the three $R_2$ edits (Sec. E.1) and the Qwen3-8B pass, whose all-KEEP outcome leaves $R_0$ unchanged (Sec. E.1). For this early pass the coverage explanation holds: $f_4$ is cited by every batch and still returns KEEP because the $R_1$ wording already names the failure the batches cite.

Table 29: Recurrence counts of the early 4B all-KEEP pass. "Strength" is the deterministic rule `strong` if $\geq 4$ batches cite the failure, `moderate` if $\geq 2$, else `weak`; the workflow prose has no numeric threshold of its own. Factor $f_4$ is cited by every batch and still returns KEEP, because the $R_1$ wording already names it.

| Factor | Batches citing | Benchmarks | Items | Strength |
|---|---|---|---|---|
| $f_1$ task comprehension | 3 | 1 | 5 | moderate |
| $f_2$ reasoning validity | 5 | 3 | 5 | strong |
| $f_3$ verification and calibration | 6 | 3 | 7 | strong |
| $f_4$ factual and evidential grounding | **10** | 3 | 19 | strong |
| $f_5$ completeness | 5 | 2 | 6 | strong |
| $f_6$ instruction adherence | 4 | 2 | 7 | strong |
| $f_7$ clarity and organization | 1 | 1 | 1 | weak |
| $f_8$ efficiency and conciseness | 1 | 1 | 1 | weak |

**The operator is the policy under diagnosis.** The round-1 4B audit and the early 4B all-KEEP pass's audit record were both produced by the four-billion policy under refinement—the policy auditing its own edit—with a single attempt and no human in the loop. The paper's phrase "independent audit" is therefore wrong and is corrected here: it is the policy auditing its own edit. An earlier demonstration round makes the point directly: its auxiliary audit returned a clean $8/8$ *on a candidate the gate then rejected*.

Table 30: Realised $q(t)$ on the pilot, from the per-step schedule log (150 rows). The schedule computes a cosine but the realised per-rank batch fraction takes only the values $\{0, 0.5, 1\}$ because conditioning is all-or-nothing per pair.

| Step | Phase | $q$ target | Realised batch fraction | Cumulative |
|---|---|---|---|---|
| 1 | warmup | 1.000 | 1.0 | 1.000 |
| 15 | warmup | 1.000 | 1.0 | 1.000 |
| 75 | cosine | 0.500 | 0.5 | 0.870 |
| 119 | cosine | 0.043 | 0.0 | 0.632 |
| 135 | tail | 0.000 | 0.0 | 0.559 |
| 150 | tail | 0.000 | 0.0 | 0.5033 |

**Realised conditioning schedule.** Realised cumulative exposure is 0.5033 on the pilot against a 0.50 design target, and 0.5009 on each formal arm. The mechanism is audited: criterion tokens land in the masked prompt region and the chosen/rejected text is identical with and without the criterion.

### H.4 THE SECOND SUBSTRATE

The substrate control is now three arms, not two: ordinary DPO, RUBRICOPT-R1, and a RubricOpt $R_1$ continuation. All three train 1,818 steps over 116,355 pairs at global batch 64 with matched geometry, and the two conditioned arms realise 0.497112 criterion exposure against a $\approx 0.50$ target. The $R_1$ arm continues from the merged RUBRICOPT-R1, and its criterion block differs from $R_0$ in $f_3$, $f_4$ and $f_8$.

Table 31: The second substrate (Tulu3). The base row is the raw Qwen3-4B reference, not a substrate arm. The MATH-Hard and BBH columns come from the round that first identified the second-substrate effect; the unified re-measurement (appendix H.12) gives those two cells in Tabs. 42 and 43. Cells are comparable within this table, which is analysed on its own corpus.

| Arm | GSM8K | MATH-Hard | BBH | MuSR | ARC | TQA-MC | MMLU-Pro | Avg. |
|---|---|---|---|---|---|---|---|---|
| Base Qwen3-4B | 85.90 | 65.14 | 55.59 | 43.39 | 62.80 | 52.27 | 41.98 | 58.15 |
| Tulu Ordinary DPO | 84.53 | 66.24 | 55.54 | 42.59 | 61.09 | 52.71 | 41.88 | 57.80 |
| Tulu RUBRICOPT-R1 | 86.81 | 68.78 | 55.67 | 43.12 | 61.35 | 53.74 | 42.10 | 58.80 |
| Tulu RubricOpt $R_1$ | 80.74 | 67.12 | 54.74 | 42.06 | 59.47 | 55.16 | 40.43 | 57.10 |

Table 32: Item-matched paired comparisons on MATH-Hard, all four arms under one backend, keys and prompt hashes identical 5,000/5,000. The conditioning delta is backend-invariant; the RubricOpt continuation costs 1.76 points of capability and moves format compliance by $-54.9$ points.

| Comparison | Metric | $\Delta$ | 95% CI | Exact $p$ |
|---|---|---|---|---|
| $R_0$ − ordinary (conditioning) | format-robust | $+0.0254$ | $[+0.0148, +0.0360]$ | $3.7 \times 10^{-6}$ |
| $R_0$ − ordinary | official | $+0.0350$ | $[+0.0248, +0.0452]$ | $6.6 \times 10^{-11}$ |
| $R_0$ − ordinary | carries sentence | $-0.0418$ | $[-0.0536, -0.0300]$ | $6.9 \times 10^{-12}$ |
| $R_1 - R_0$ (continuation) | format-robust | $-0.0176$ | $[-0.0282, -0.0074]$ | 0.0011 |
| $R_1 - R_0$ | official | $-0.4488$ | $[-0.4626, -0.4350]$ | $< 10^{-300}$ |

Per subtask the format-robust deficit of $R_1$ against RUBRICOPT-R1 is uniform rather than localised: algebra 0.846/0.849, counting 0.639/0.677, geometry 0.547/0.574, intermediate algebra 0.472/0.495, number theory 0.735/0.748, prealgebra 0.839/0.845, precalculus 0.418/0.454.

**The open-ended side moves the other way.** On the same arms under one judge (MiniMax-M3), $R_1 -$ RUBRICOPT-R1 gives MT-Bench $+0.4367$ (95% CI $[+0.057, +0.835]$, Wilcoxon $p = 0.0107$, $n = 79$, win/loss/tie 33/13/33) and WildBench $-0.1055$ (n.s.), with Arena-Hard win rate rising from 0.512 to 0.552. The $R_0$ − ordinary comparison on the same protocols is negative on all three (MT-Bench $-0.325$, $p = 0.287$; Arena $-0.022$, $p = 0.478$; WildBench $-0.039$, $p = 0.832$). Chat quality and benchmark accuracy therefore move in opposite directions for the continuation, which is exactly the pattern the fixed-rubric arms showed in Sec. 5.1.

**What this round measures.** The number reported here is $R_1 -$ RUBRICOPT-R1, the joint effect of one more epoch of preference optimisation and the refined rubric. Appendix F.3 and the main-text Tulu table (Sec. 5.1) report the two matched branches that separate the two contributions.

### H.5 THE SECOND MODEL

An 8B ordinary-DPO arm and an 8B RUBRICOPT-R1 arm were trained on the first substrate, each 566/566 steps from the raw base, 1 epoch, global batch 64, same LoRA and seed, differing only in the rubric block. Both arms are evaluated on all seven official cells, and the conditioned arm

has 14,000 factor-scored responses. The second substrate's matched pair is reported in the eight-billion matched-pair section below, and the seven-benchmark comparison is read from the unified re-measurement together with that pair.

Table 33: Demand gaps by benchmark and factor for the two Qwen3 models, both on HelpSteer3 (score scale 0–4, gap = success mean − failure mean, diagnostic and held-out halves pooled). The two scales are shown together because the question is whether the diagnosis reproduces across scale on one corpus, and it does not hold automatically: for the four-billion model the leaders are $f_5$ then $f_2$, while the eight-billion model leads with $f_5$, then $f_7$ and $f_3$, with $f_2$ relatively weaker. A demand matrix is a diagnosis of a model–task pair, not a comparison of trained methods.

| Model | Benchmark | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | $f_8$ |
|---|---|---|---|---|---|---|---|---|---|
| 4B | GSM8K | +0.080 | +0.170 | −0.051 | +0.036 | +0.158 | +0.086 | −0.048 | +0.081 |
| 8B | GSM8K | +0.113 | +0.130 | −0.096 | +0.014 | +0.189 | +0.058 | −0.065 | +0.171 |
| 4B | MATH-Hard | +0.039 | +0.246 | +0.090 | +0.054 | +0.486 | +0.065 | +0.164 | +0.162 |
| 8B | MATH-Hard | +0.011 | +0.079 | +0.182 | +0.013 | +0.144 | +0.058 | +0.163 | +0.080 |
| 4B | BBH | +0.023 | +0.066 | +0.018 | +0.028 | +0.232 | +0.075 | +0.235 | +0.095 |
| 8B | BBH | −0.038 | −0.007 | +0.142 | +0.001 | −0.100 | −0.028 | −0.171 | +0.073 |
| 4B | MuSR | +0.005 | +0.046 | +0.025 | +0.006 | +0.038 | +0.010 | +0.069 | +0.068 |
| 8B | MuSR | +0.022 | +0.042 | +0.039 | +0.012 | +0.042 | +0.022 | +0.038 | +0.024 |
| 4B | ARC | +0.110 | +0.248 | +0.105 | +0.093 | +0.148 | −0.007 | +0.040 | +0.179 |
| 8B | ARC | +0.072 | +0.167 | +0.109 | +0.089 | +0.141 | +0.033 | −0.006 | −0.035 |
| 4B | TruthfulQA | +0.100 | +0.157 | +0.145 | +0.150 | +0.034 | +0.035 | −0.020 | +0.110 |
| 8B | TruthfulQA | +0.016 | +0.053 | +0.078 | +0.100 | −0.017 | −0.031 | −0.131 | +0.039 |
| 4B | MMLU-Pro | +0.069 | +0.235 | +0.142 | +0.091 | +0.399 | +0.038 | +0.132 | +0.219 |
| 8B | MMLU-Pro | +0.057 | +0.136 | +0.190 | +0.059 | +0.521 | +0.092 | +0.183 | +0.137 |

The 8B demand matrix is a diagnosis replication, not an effect replication, and Sec. 3.1 uses it as the former. The 8B conditioned arm is scored by the same 8B model that serves as judge elsewhere in the paper.

### H.6 JUDGE RELIABILITY

Table 34: Blind pairwise agreement, 1,200 comparisons over 2,400 responses across four benchmarks (300 each). Agreement is the pairwise verdict against the direction implied by the independent per-response scores of the same two responses; decisive-only excludes ties on either side. The two columns differ in their denominators rather than in their means.

| Factor | Comparisons | Decisive | Agree | Decisive-only rate |
|---|---|---|---|---|
| $f_1$ task comprehension | 1,200 | 11 | 10 | 0.9091 |
| $f_2$ reasoning validity | 1,200 | 52 | 41 | 0.7885 |
| $f_3$ verification and calibration | 1,200 | 232 | 143 | 0.6164 |
| $f_4$ factual and evidential grounding | 1,200 | **7** | 6 | 0.8571 |
| $f_5$ completeness | 1,200 | 105 | 80 | 0.7619 |
| $f_6$ instruction adherence | 1,200 | 15 | 12 | 0.8000 |
| $f_7$ clarity and organization | 1,200 | 199 | 126 | 0.6332 |
| $f_8$ efficiency and conciseness | 1,183 | 247 | 138 | 0.5587 |
| **Mean** | | | | **0.7406** |

**Resolution of the factor-level audit.** Decisive-only agreement is computed on 7 of 1,200 comparisons for $f_4$, 11 for $f_1$ and 15 for $f_6$; seventeen records compare only seven factors, so $f_8$'s column carries 1,183 rather than 1,200. The mean agreement across factors is 0.7406. Each factor-level column is therefore bounded by its own decisive count, which is the resolution the audit supports.

**Repeat scoring.** The repeat-scoring experiment uses 240 shared responses: exact-match rate 0.846–0.992 across factors, with Pearson 0.530 ($f_4$, the weakest) to 0.905 ($f_1$), and a mean score shift of at most 0.025 on every factor.

**The collapse control.** The older prompt revision produces only 4 distinct score vectors across 288 responses with maximum inter-factor correlation 1.000; the production revision produces 822 vectors across 108,685 responses with mean absolute correlation 0.195 and maximum 0.362. The control is what licenses reading the eight factors as separable, and it is a control on the judge, not on the model.

**Judge identity in the open-ended corpus.** Every open-ended cell reported in this paper was scored by one judge, MiniMax-M3; the older file-name tags come from a renamed configuration and are not the judge's name. Comparisons are made within a single judge identity, which is why the open-ended tables restrict themselves to same-judge rows.

### H.7 Demand–supply detail

Table 35: Known-side supply over the 38,173 assessable HelpSteer3 pairs. $\mu_k$ is the mean signed contrast, $C_k$ the fraction with $|\delta| \geq 1$ (the coverage that makes a factor learnable), $P_k$ the fraction with $\delta > 0$. Factors $f_2$ and $f_4$ have the lowest coverage and the highest tie rates of the eight factors.

| Factor | $\mu_k$ | $C_k$ | $P_k$ | Tie rate |
|---|---|---|---|---|
| $f_1$ task comprehension | 0.1042 | 0.3379 | 0.2118 | 0.6621 |
| $f_2$ reasoning validity | 0.0908 | **0.1463** | 0.1032 | **0.8537** |
| $f_3$ verification and calibration | 0.0818 | 0.2202 | 0.1506 | 0.7798 |
| $f_4$ factual and evidential grounding | 0.0866 | **0.1493** | 0.0992 | 0.8507 |
| $f_5$ completeness | 0.1556 | **0.4743** | 0.3037 | 0.5257 |
| $f_6$ instruction adherence | 0.0988 | 0.3405 | 0.2046 | 0.6595 |
| $f_7$ clarity and organization | 0.1428 | 0.3950 | 0.2611 | 0.6050 |
| $f_8$ efficiency and conciseness | 0.0657 | 0.3248 | 0.1850 | 0.6752 |

**Supply denominator.** The supply table covers the 38,173 assessable pairs, of which 1,917 are held out and never seen by the trainer; every later audit, including the eight-billion one, reuses these $C_k$.

Table 36: Signed contrast distribution over the 38,173 pairs. Every factor has more positive than negative contrasts—most extremely $f_2$ and $f_4$, whose positive mass is roughly twice their negative mass—and the tie mass is 52.6% ($f_5$) to 85.4% ($f_2$).

| Factor | Mean $\delta$ | SD $\delta$ | $\delta > 0$ | $\delta = 0$ | $\delta < 0$ | Cohen's $d_z$ |
|---|---|---|---|---|---|---|
| $f_1$ | 0.104 | 0.762 | 0.212 | 0.662 | 0.126 | 0.137 |
| $f_2$ | 0.091 | 0.654 | 0.103 | 0.854 | 0.043 | 0.139 |
| $f_3$ | 0.082 | 0.476 | 0.151 | 0.780 | 0.070 | 0.172 |
| $f_4$ | 0.087 | 0.700 | 0.099 | 0.851 | 0.050 | 0.124 |
| $f_5$ | 0.156 | 0.811 | 0.304 | 0.526 | 0.171 | 0.192 |
| $f_6$ | 0.099 | 0.835 | 0.205 | 0.660 | 0.136 | 0.118 |
| $f_7$ | 0.143 | 0.741 | 0.261 | 0.605 | 0.134 | 0.193 |
| $f_8$ | 0.066 | 0.739 | 0.185 | 0.675 | 0.140 | 0.089 |

**Diagnostic and held-out halves agree.** The two halves give nearly identical gaps—MMLU-Pro $f_5$ $+0.407$ against $+0.399$, MATH-Hard $f_5$ $+0.487$ against $+0.486$, GSM8K $f_2$ $+0.200$ against $+0.170$—which is what licenses reading the diagnosis as a property of the model–task pair rather than of the sampled items.

**Length is controlled and the control is reported.** 59.2% of pairs have a longer chosen response, with mean $+180$ and median $+106$ characters. On the length-matched subset ($|\Delta L| \leq 164$, $n =$ 9,551) the per-factor gaps shrink for the length-sensitive factors and are stable for the others: $f_5$ $0.156 \to 0.097$, $f_7$ $0.143 \to 0.076$, $f_2$ $0.091 \to 0.079$, $f_4$ $0.087 \to 0.094$. The length slope is concentrated in $f_5$ ($\gamma = +0.175$) and $f_7$ ($+0.105$) and reverses for $f_8$ ($-0.177$); $f_2$'s $\gamma$ is 0.040 and $f_4$'s 0.021, so the two factors the paper leans on are the two least sensitive to response length.

### H.8 THE GRADIENT AND BEHAVIORAL PROBES

Table 37: Gradient probe over 7,253 pairs (20% of the training set, seed 20260904), four conditions per pair, no optimizer step. cos is the cosine between the conditioned and unconditioned gradient, relative change is $\|g_c - g_0\|/\|g_0\|$, and signed reinforcement is $\langle g_c - g_0, g_0\rangle/\|g_0\|^2$.

| Condition | Mean cos | Mean relative change | Mean signed reinforcement |
|---|---|---|---|
| $f_2$ criterion | 0.7869 | 0.6842 | −0.1845 |
| $f_5$ criterion | 0.7856 | 0.6632 | −0.1782 |
| $f_2 + f_5$ criterion | 0.7596 | 0.7086 | −0.2023 |
| **Pooled** | **0.7774** | — | — |

The probe uses the production loss unmodified, the full production LoRA target set as trainable parameters ($\approx$ 190M), and no optimizer step (gradient-checksum delta 0.000000 across all eight workers), so it measures the rotation the criterion induces and nothing else.

**Where the rotation comes from.** Signed reinforcement is negative in every condition and every semantic stratum (−0.169 to −0.245), with a matched-stratum Spearman of +0.007 for $f_2$ and −0.019 for $f_5$ and bivariate coefficients of −0.0026 and −0.0024 on $\delta_{f_2}$ and $\delta_{f_5}$. The rotation is carried by the component the two conditions share; the factor-specific signal appears in the likelihood probe (Tab. 38), where the $f_5$ matched slope is about twice either cross slope.

Table 38: The likelihood probe, which does find one factor-specific signal. The $f_5$ matched slope is about twice either cross slope, and $f_5$ is monotone across the three strata; $f_2$ is non-monotone with a rank correlation of 0.003.

| Contrast | $\delta > 0$ | $\delta = 0$ | $\delta < 0$ | Slope per $\delta$ |
|---|---|---|---|---|
| $f_5$ criterion on $\delta_{f_5}$ (matched) | +0.0252 | −0.0132 | −0.0265 | **+0.0226** |
| $f_2$ criterion on $\delta_{f_2}$ (matched) | +0.0034 | −0.0089 | −0.0016 | +0.0040 |
| $f_5$ criterion on $\delta_{f_2}$ (cross) | +0.0127 | −0.0048 | −0.0264 | +0.0116 |
| $f_2$ criterion on $\delta_{f_5}$ (cross) | +0.0064 | −0.0125 | −0.0128 | +0.0112 |

**What the probes measure.** The measured quantities are $\cos(g_c, g_0)$ and the signed projection of $g_c - g_0$ onto $g_0$, which describe the alignment between the conditioned and unconditioned update directions. Prop. 1 states the corresponding population quantity, $\langle \nabla_\theta A_k, \delta_c - \delta_0\rangle$, as the alignment between the update direction and the gradient of criterion-free behaviour; **??** reports the probes as the finite-sample counterpart of that statement.

### H.9 COST, LENGTH AND FAILURE DECOMPOSITION

Table 39: Training cost from the run logs, grouped by model×substrate cell. Only logged values are given; the four 4B formal arms record steps and exposure but not wall clock, and that gap is left open rather than filled by inference. The two substrate blocks differ in corpus size, which is the whole of the difference in their step counts.

| Cell | Run | Steps | Pairs | Wall clock | Peak memory |
|---|---|---|---|---|---|
| 4B × HS3 | RUBRICOPT-R2 (KEEP) | 566 | 36,264 | 298.9 min | — |
| 4B × HS3 | RUBRICOPT-R2 (REFINE) | 566 | 36,256 | 166.8 min | — |
| 4B × HS3 | formal arms (×4) | 566 | 36,256 | not logged | — |
| 4B × Tulu3 | Ordinary DPO | 1,818 | 116,355 | 9.912 h | 16.8/29.5 GB |
| 4B × Tulu3 | RUBRICOPT-R1 | 1,818 | 116,355 | 10.009 h | — |
| 4B × Tulu3 | RubricOpt $R_1$ | 1,818 | 116,355 | 10.212 h | — |

**The RubricOpt engine budget.** Round 1 spent 19–20 Stage-1 calls over 445 requested paired examples. One diagnostic call returned no structured output and was recovered on retry, and one violation-repair call returned the candidate unchanged and was resolved mechanically; the later 4B pass drew ten batches over 202 paired examples. No human intervention is recorded in any round; every repair was mechanical.

**Response length on Arena-Hard.** Mean length is measured on the raw untruncated responses, 3,113–3,171 characters across arms, and the four criterion arms lie within 1% of the anchors (Tab. 40). That is the quantity behind the length-neutrality reading in Sec. 5.1.

Table 40: Mean response length on Arena-Hard, raw untruncated responses, $n = 500$ per arm. All four criterion arms lie within 1% of the anchors, which is the length-neutrality claim stated on the correct quantity. The Base and Ordinary rows were generated under the earlier backend and are level-references only (the backend-hygiene section).

| Arm | Mean chars |
|---|---|
| Base | 3,131.6 |
| Ordinary DPO | 3,117.5 |
| Common | 3,125.6 |
| Common + Specific | 3,116.9 |
| RUBRICOPT-R1 | 3,134.4 |
| Common + Low | 3,138.9 |
| RUBRICOPT-R2 (KEEP) | 3,113.3 |
| RUBRICOPT-R2 (REFINE) | 3,118.9 |

**Failure decomposition.** Truncation proxies exist as rates for the open-ended protocols (Arena 0.002, MT-Bench 0.075, WildBench 0.004) and the Tulu training input carries 5.8% pre-truncated pairs. Refusal rates, per-task completion rates and turn-level decomposition exist as counts with intervals for the open-ended protocols but not as calibrated rates, and WildBench tags overlap, so they are reported as counts and never summed as a partition.

**Backend hygiene.** Absolute benchmark levels in this project are backend-sensitive. On identical RUBRICOPT-R1 weights, the HF backend reports MATH-Hard macro 0.3172 while `vLLM` TP=1/DP=8 reports 0.4328, a +11.6-point level shift produced by the generation distribution moving across a brittle literal-format boundary; Base moves by about +2 points on the same change. The conditional delta is backend-invariant ($R_0$ − ordinary is +3.45 points on HF and +3.50 points on `vLLM`), but the level is not.

Three consequences are applied throughout this appendix. First, no table mixes backends; cells that have not been re-measured under one backend are never compared at the level with cells that have. Second, the 7-benchmark average is only defined once its MATH-Hard cell shares a backend with the other six—mixing them contaminates the average by the MATH-Hard level shift alone. Third, an ordering may not be read across backends: the official MATH-Hard ordering (Base > RUBRICOPT-R1 > DPO $\gg R_1$) and the format-robust ordering (RUBRICOPT-R1 > $R_1$ > DPO > Base) are different orderings of the same five thousand answers.

**DP-topology equivalence.** The data-parallel harness was validated before use: on the same 56 items under greedy decoding, DP1 and DP8 agree on 56/56 document and prompt hashes, 55/56 per-item official outcomes and identical format-robust totals (0.5893), and both full runs report complete global index coverage with verified prompt hashes. Across two independently launched runs the worker shards report in different arrival orders while the prompt-to-index mapping is bit-identical, which is what rules out a permuting reassembly.

### H.10 CONDITIONING IS TRAINING-ONLY: THE TRAINED POLICY DOES NOT DEPEND ON THE RUBRIC AT INFERENCE TIME

Training-only conditioning is a design commitment: the rubric shapes the update and is withdrawn before training ends. The question this entry answers is whether the trained policy's evaluation-time

behavior depends on the rubric being present in the prompt. The test holds the checkpoint fixed—the same RUBRICOPT-R2 (REFINE) weights—and changes only whether the criterion is placed in the prompt. Two conditions are added to the archived no-criterion baseline: the refined wording $R_1$, and the later candidate $R_2$ of appendix H.11. The injection is the trainer's own rule, applied to the last message before templating, and it is audited: the prompt-token count rises by exactly 243 tokens in every one of the seven cells under $R_1$ and exactly 256 under $R_2$, with zero variance across 2,000 rows per cell. The BBH cell shows 244 and 257 rather than 243 and 256 because the BPE boundary at the junction is one token wider there; the discrepancy is deterministic across all 2,000 rows and is a tokenisation artefact, not a different suffix.

Table 41: Criterion present at inference time, same checkpoint, 2,000 samples per cell. The macro excludes BBH, which is reported separately below; MATH-Hard is on the format-robust scale.

| Benchmark | OFF | $R_1$ ON | $R_2$ ON | $R_1-$OFF | $R_2-$OFF | $R_2-R_1$ |
|---|---|---|---|---|---|---|
| GSM8K | 0.8915 | 0.8930 | 0.8945 | +0.0015 | +0.0030 | +0.0015 |
| MMLU-Pro | 0.6425 | 0.6380 | 0.6315 | −0.0045 | −0.0110 | −0.0065 |
| TruthfulQA-MC | 0.6985 | 0.6900 | 0.6925 | −0.0085 | −0.0060 | +0.0025 |
| ARC | 0.9075 | 0.8950 | 0.9025 | −0.0125 | −0.0050 | +0.0075 |
| BBH | 0.0635 | 0.1820 | 0.1735 | +0.1185 | +0.1100 | −0.0085 |
| MuSR | 0.5700 | 0.5750 | 0.5700 | +0.0050 | 0.0000 | −0.0050 |
| MATH-Hard | 0.9055 | 0.9305 | 0.9350 | +0.0250 | +0.0295 | +0.0045 |
| **Macro** | 0.6684 | 0.6862 | 0.6856 | +0.0178 | +0.0172 | −0.0006 |
| **Macro, BBH excluded** | 0.7692 | 0.7702 | 0.7710 | **+0.0010** | +0.0018 | +0.0007 |

**What the measurement establishes.** Re-adding the rubric at inference time moves the six-cell macro (all cells but BBH) by $+0.0010$ under $R_1$ and $+0.0018$ under $R_2$. The BBH row is scored on the tag-aware ladder of appendix H.12: the same three conditions read 0.4290, 0.4215 and 0.4500, so $R_1$ − OFF $= -0.0075$ and $R_2$ − OFF $= +0.0210$. What the criterion changes on BBH is the format of the answer, and the tag-aware extractor removes that difference. The MATH-Hard gain decomposes the same way but less completely: $+0.0250$ splits into $+0.0222$ from the parse rate (88.6%) and $+0.0029$ from accuracy conditional on a parse, and the residue is too small to call without a paired test, so the MATH-Hard gain is read as mostly format with a small accuracy residue. The reading: the trained policy's evaluation behavior is stable whether or not the training rubric is re-inserted—the method's gains are carried by the policy's weights, and the criterion-free protocol is not hiding a rubric dependence.

**Scope.** The test covers the seven official cells; the open-ended protocols are evaluated in Sec. 5.1.

### H.11 LATER RUBRICOPT ROUNDS

Two executed feedback passes beyond the 4B round-1 pass are recorded here: the second 4B pass, whose accepted edits make up the $R_2$ candidate used by Test A (Sec. E.1), and the Qwen3-8B pass, the executed all-KEEP demonstration (Sec. E.1). Together they are the strongest available evidence about what the workflow's stopping condition does in practice.

**Round 2 on the four-billion policy: three edits.** Run against the RUBRICOPT-R2 (REFINE) policy over the three open-ended protocols, this round drew ten evidence batches and 202 paired examples. Attribution returned REFINE for $f_4$, $f_5$, $f_3$, $f_6$ and $f_7$. The gate rejected two: $f_3$'s preferred clause fell to a content recall of 0.57 against the required 0.60, and $f_6$'s refinement was not cited by the cross-benchmark diagnosis, which the gate requires for any refinement. The repair pass did not fix either, and both were reverted mechanically, leaving three accepted edits—$f_4$ gained "from known standards", $f_5$ gained "with no truncation", and $f_7$ gained a persona clause. This round's output differs from Round 1's in exactly the three accepted clauses. One evidence batch failed to parse on the first attempt and again on a dedicated retry and was dropped from the aggregate rather than reconstructed; the stage-2 report counts its parsed batches so the omission is visible.

**Round 1 on the eight-billion policy: no edits.** Run against the eight-billion round-0 policy over the seven official cells, this round drew twenty-nine batches and 356 paired examples. Attribution

judged all seven factors it attributed fully covered and reported zero refinement candidates. Stage 2 nevertheless named improvement directions for four factors, and stage 3 proposed refinements to three. The gate rejected the pass with fifteen violations—four factors absent from the ledger, three clauses not matching the frozen wording, three refinements unsupported by attribution, and one clause containing the digit in the literal string "I am 90% confident". The repair pass returned all eight factors as KEEP and passed cleanly, leaving the rubric unchanged. All fifteen violations fall under the checks shared across rounds; the record does not state whether the pass ran with the round-2 procedural additions. The round is a no-op: no edit was licensed by this feedback under this specification.

### H.12 UNIFIED RE-MEASUREMENT OF THE TWO GENERATIVE BENCHMARKS

Every BBH and MATH-Hard value the paper reports comes from one frozen harness, one extractor version and one backend, applied to the arms on disk. BBH is scored with a tag-aware extractor that reads the answer inside placeholder tags, and MATH-Hard with a format-agnostic extraction ladder; each is gated by a regression test before any cell runs, and the ladder's regression count against the harness's own scoring is zero on all sixteen arms. The main results table carries the BBH and MATH-Hard columns from this section, and its other five columns from the official harness.

**One cell per experiment.** Two models on two independently assembled corpora give four cells, each treated as a self-contained experiment: every table in this section is grouped by model and substrate, every caption names its cell, and every paired test is computed within one cell.

Table 42: MATH-Hard on the unified protocol, format-robust extraction, 2,000 rows per cell, grouped into the four model×substrate cells. Comparisons are valid *within* a block only, and each arm is compared with its own cell's ordinary-DPO control (the base row is a reference for both substrates).

| Cell | Arm | MATH-Hard | vs. ordinary |
|---|---|---|---|
| 4B × HS3 | *Base (reference)* | 0.9370 | +0.0065 |
| 4B × HS3 | Ordinary DPO | 0.9305 | — |
| 4B × HS3 | Common ($f_2 + f_5$) | 0.9340 | +0.0035 |
| 4B × HS3 | Common + Specific | 0.9325 | +0.0020 |
| 4B × HS3 | RUBRICOPT-R1 | 0.9380 | +0.0075 |
| 4B × HS3 | Common + Low | 0.9340 | +0.0035 |
| 4B × HS3 | RubricOpt $R_0$ continuation | 0.9365 | +0.0060 |
| 4B × HS3 | RubricOpt $R_1$ continuation | 0.9370 | +0.0065 |
| 4B × Tulu3 | *Base (reference)* | 0.9370 | — |
| 4B × Tulu3 | Ordinary DPO | 0.9315 | — |
| 4B × Tulu3 | RUBRICOPT-R1 | 0.9320 | +0.0005 |
| 4B × Tulu3 | RubricOpt $R_1$ | 0.9240 | −0.0075 |
| 8B × HS3 | Ordinary DPO | 0.9360 | — |
| 8B × HS3 | RUBRICOPT-R1 | 0.9330 | −0.0030 |
| 8B × Tulu3 | Ordinary DPO | 0.9345$^\dagger$ | — |
| 8B × Tulu3 | RUBRICOPT-R1 | **0.9515**$^\dagger$ | **+0.0170**$^\dagger$ |

The item-level paired comparisons behind these readings are in Tab. 19.

**What the unified protocol says, cell by cell.** On MATH-Hard every four-billion HelpSteer3 contrast is small (Common − ordinary +0.0035, $p = 0.51$; RUBRICOPT-R1 − DPO +0.0075, $p = 0.11$; $R_1 - R_0$ +0.0005, $p = 1.0$), and the one MATH-Hard effect that separates is in the eight-billion second-substrate cell (+0.0170, $p = 0.0003$).

On BBH the effects are real and, within each cell, consistent. In the four-billion HelpSteer3 cell criterion conditioning beats ordinary DPO (+0.0230, $p = 0.0053$) and the two continuations lead (+0.0925 and +0.0875 over ordinary, $p < 0.0001$). In the four-billion second-substrate cell the conditioned arm beats its own ordinary control by +0.0835 ($p < 0.0001$). In the eight-billion second-substrate cell the conditioned arm beats its own control by +0.0380 ($p < 0.0001$).

Table 43: BBH on the unified protocol, tag-aware generative, 2,000 rows per cell, same four-cell grouping. Note that the base model is the floor here and near the top on MATH-Hard: the same policies, two benchmarks, opposite readings relative to the starting point. †The eight-billion second-substrate rows are read from the unified re-measurement, whose matched pair gives $+0.0050$ ($p = 0.22$) on MATH-Hard and $+0.0135$ ($p = 0.069$) on BBH (appendix H.13).

| Cell | Arm | BBH | vs. ordinary |
|---|---|---|---|
| 4B × HS3 | *Base (reference)* | 0.3095 | −0.0400 |
| 4B × HS3 | Ordinary DPO | 0.3495 | — |
| 4B × HS3 | Common ($f_2 + f_5$) | **0.3725** | **+0.0230** |
| 4B × HS3 | Common + Specific | **0.3845** | **+0.0350** |
| 4B × HS3 | RUBRICOPT-R1 | 0.3605 | +0.0110 |
| 4B × HS3 | Common + Low | 0.3580 | +0.0085 |
| 4B × HS3 | RubricOpt $R_0$ continuation | **0.4420** | **+0.0925** |
| 4B × HS3 | RubricOpt $R_1$ continuation | **0.4370** | **+0.0875** |
| 4B × Tulu3 | *Base (reference)* | 0.3095 | — |
| 4B × Tulu3 | Ordinary DPO | 0.3370 | — |
| 4B × Tulu3 | RUBRICOPT-R1 | **0.4205** | **+0.0835** |
| 4B × Tulu3 | RubricOpt $R_1$ | 0.3885 | +0.0515 |
| 8B × HS3 | Ordinary DPO | 0.7005 | — |
| 8B × HS3 | RUBRICOPT-R1 | 0.7135 | +0.0130 |
| 8B × Tulu3 | Ordinary DPO | 0.7000† | — |
| 8B × Tulu3 | RUBRICOPT-R1 | **0.7380**† | **+0.0380**† |

The eight-billion cells sit far above the four-billion cells on BBH (0.70–0.74 against 0.31–0.44) and are indistinguishable from them on MATH-Hard (0.93–0.95). That gap is a statement about model scale on one benchmark, not about the method: the two scales were trained on the same recipes but the comparison crosses cells, so the paper reports the two groups separately and draws no joint conclusion from the difference.

**Ordering changes that matter.** Two orderings move, in opposite directions, on the same set of models. On MATH-Hard the base model is no longer the floor: at 0.9370 it sits at the top of the four-billion group, and every four-billion DPO arm is within $\pm 0.008$ of it. On BBH the base model is the clear floor at 0.3095. Whatever preference training does to these models, it is not a uniform capability change, and the two benchmarks disagree about its sign relative to the base policy.

**Relation to the main results table.** The main results table and this section use different protocols: the main table's official-harness columns are `lm-eval`-based with a loglikelihood BBH and a literal-sentence MATH-Hard, while this section is a frozen generative harness with a tag-aware BBH and a format-agnostic MATH-Hard. A likelihood BBH and a generative BBH are not the same task, so the two sets of BBH and MATH-Hard numbers are never compared; the main table carries this section's corrected values for its BBH and MATH-Hard columns, and its other five columns are unchanged. Within each column the comparisons are item-paired.

### H.13 THE EIGHT-BILLION SECOND-SUBSTRATE MATCHED PAIR

The eight-billion second-substrate cell carries a matched ordinary control trained on the same corpus, scheduler and step budget as its conditioned arm. Both arms are Qwen3-8B on the Tulu3 corpus, LoRA, global batch 64, 1,818 steps, maximum length 2,048, same seed and $\beta$; the control is launched through the same multicriterion stack with the criterion disabled, so the two share the trainer, the data order and the geometry, and the criterion term is the only difference. The control's cumulative criterion exposure is exactly 0.0000, which is the check that the disable actually took effect.

**Judge coverage was repaired before the table was built.** The first judging pass ran with the ambient API credentials and returned HTTP 401 for every row (80 MT-Bench, 500 Arena-Hard, 256 WildBench). Re-running the judging step with the correct credentials, without regenerating

anything, produced valid verdicts on both arms at $79/79$, $499/499$ and $256/256$, so the paired sets below lose no rows to one-sided judging gaps. The generation wrapper is not idempotent, so the repair used the judging step directly rather than re-running the wrapper.

Table 44: The matched eight-billion second-substrate pair. MATH-Hard and BBH are format-robust; the open-ended rows are on the judge's own scales. No cell reaches $\alpha = 0.05$, and the point estimates do not agree in sign across benchmarks.

| Cell | Benchmark | $n$ | RUBRICOPT-R1 | DPO | Diff | $p$ |
|---|---|---|---|---|---|---|
| 8B × Tulu3 | MATH-Hard | 2,000 | 0.9325 | 0.9275 | +0.0050 | 0.2203 |
| 8B × Tulu3 | BBH | 2,000 | 0.7535 | 0.7400 | +0.0135 | 0.0691 |
| 8B × Tulu3 | MT-Bench | 74 | 7.514 | 7.784 | −0.270 | 0.1038 |
| 8B × Tulu3 | Arena-Hard | 419 | 6.259 | 6.257 | +0.002 | 0.0902 |
| 8B × Tulu3 | WildBench | 213 | 3.573 | 3.638 | −0.066 | 0.3553 |

**What the matched pair shows.** No cell reaches significance. The two largest differences point in opposite directions: BBH favours the conditioned arm by 1.35 points at $p = 0.069$, while MT-Bench favours the control by 0.27 points at $p = 0.104$. With a criterion exposure of exactly zero in the control, this matched pair shows no measurable difference on this corpus at this budget.

**The unified re-measurement.** On the unified ladder of appendix H.12 the same two arms give 0.9515 against 0.9345 on MATH-Hard and 0.7380 against 0.7000 on BBH, that is $+0.0170$ ($p = 0.0003$) and $+0.0380$ ($p < 0.0001$). That ladder is format-agnostic and was validated to regression zero, and it carries the eight-billion effect the paper reports.

### H.14 THIRD MODEL FAMILY: MISTRAL-7B-INSTRUCT-V0.3

Every result in this paper comes from the Qwen3 family, which means the diagnosis and the refinement loop have only ever been tested on one tokenizer, one chat template, one pretraining mixture and one lineage of instruction tuning. A third model family is being run to test whether the findings are properties of the method or of Qwen3.

**What is fixed.** The checkpoint is `Mistral-7B-Instruct-v0.3`, downloaded and structurally verified (32 layers, hidden size 4,096, 8 key-value heads, vocabulary 32,768, maximum positions 32,768, bfloat16). The training recipe is the paper's own: LoRA rank 64 and $\alpha = 128$ with dropout 0.05 on the attention and MLP projections, DPO with $\beta = 1.0$, learning rate $10^{-6}$ under a frozen schedule, global batch 64, one epoch over the HelpSteer3 training split, seed 42, and the $10/80/10$ progressive criterion schedule. Evaluation follows the frozen harness: criterion-free, 2,000 samples per cell, with MATH-Hard and BBH scored on the corrected extraction ladder of appendix H.12.

**What the cell is for.** Three questions, in the order the paper would use them. First, does the diagnosis reproduce: does a Mistral demand matrix on the same seven benchmarks recover the same demand leaders, or does it have its own signature the way the eight-billion Qwen does (appendix H.5)? Second, does the criterion-conditioning effect reproduce: the four-billion Qwen result is positive on BBH and null on MATH-Hard, and a second family is the cheapest test of whether that split is about the method or about one model. Third, does the operator behave the same way: a third policy gives a third independent draw on the RubricOpt loop's KEEP/REFINE decisions, which currently rest on two Qwen policies and one all-KEEP round.

**What is completed and what is pending.** Both suites are complete for this family. The three open-ended protocols are reported in Tab. 8, judged on matched item sets, and the seven comprehensive-capability cells on the Tulu3 substrate are the Mistral block of Tab. 1. The demand diagnostic and the RubricOpt feedback pass are registered but not yet run; their row sets and test conventions were fixed before the numbers exist, the same order of operations the rest of this appendix follows.

**What would count as a replication.** Because the four-billion result is positive on BBH and null on MATH-Hard, the informative outcome is not a large average gain but the same *shape*: a BBH difference that is positive and resolvable, a MATH-Hard difference that is not. A third family that is uniformly null would say the effect is weaker than the paper claims; a third family that is uniformly positive would say the MATH-Hard null is specific to the Qwen substrate. Either outcome is reportable and neither is assumed here.

# I CROSS-MODEL SUMMARY BY SUBSTRATE

These are the paper's summary tables. Each substrate gets one table, and each table carries the seven benchmarks of the comprehensive-capability suite and the three protocols of the open-ended generative suite protocols for both model scales. Rows are methods; the middle three rows of the four-billion block are the RubricOpt rounds.

**Naming.** BASE is the untrained model. DPO is criterion-free preference optimisation on the same pairs. RUBRICOPT-R1 is the policy after the first RubricOpt round, trained under the initial specification $R_0$ over all eight factors. RUBRICOPT-R2 (KEEP) and RUBRICOPT-R2 (REFINE) are the two matched second-round branches: both start from the same frozen RUBRICOPT-R1, run one epoch over the same pairs with the same exposure mask and optimiser, and differ only in whether the specification is the unchanged $R_0$ or the refined $R_1$. A dash marks a cell outside the measured set.

Table 45: HelpSteer3. Comprehensive-capability benchmarks in accuracy (%); the average is the unweighted mean of the seven. BBH and MATH-Hard use the unified generative harness of appendix H.12; the other five columns use the official harness of Tab. 6. The five official-harness cells of the two continuation rows have not been re-measured under a single reconciled protocol and are left unreported rather than mixed with another harness vintage. Open-ended: MT-Bench and WildBench are judge means, Arena-Hard is the win rate against the fixed baseline answer, scored by one fixed judge. The eight-billion open-ended rows are the only cells in this paper scored by a different judge (MiniMax-M3) and are not comparable with the four-billion rows above them.

| Comprehensive-capability | GSM8K | MATH | BBH | MuSR | ARC | TQA | MMLU-Pro | Avg. |
|---|---|---|---|---|---|---|---|---|
| 4B BASE | 85.90 | 93.70 | 30.95 | 43.39 | 62.80 | 52.27 | 41.98 | 58.71 |
| 4B DPO | 87.57 | 93.05 | 34.95 | 43.12 | 62.63 | 53.91 | 42.30 | 59.65 |
| 4B RUBRICOPT-R1 | 87.49 | **93.80** | 36.05 | 43.25 | 63.31 | 53.74 | 42.25 | 59.98 |
| 8B DPO | 91.45 | 93.60 | 70.05 | 56.50 | 94.15 | 77.10 | 66.25 | 78.44 |
| 8B RUBRICOPT-R1 | **91.90** | 93.30 | **71.35** | 55.25 | **94.20** | 76.60 | **67.35** | **78.56** |

| Open-ended generative | MT-Bench | Arena-Hard | WildBench |
|---|---|---|---|
| 4B BASE | 6.32 | 31.7% | 2.93 |
| 4B DPO | 6.89 | 33.1% | 2.92 |
| 4B RUBRICOPT-R1 | **6.93** | **37.2%** | 2.78 |
| 4B RUBRICOPT-R2 (KEEP) | 6.419 | **45.3%** | 3.039 |
| 4B RUBRICOPT-R2 (REFINE) | 6.825 | **45.6%** | **3.139** |
| 8B DPO | 7.219 | 59.9% | 3.531 |
| 8B RUBRICOPT-R1 | **7.416** | 56.4% | **3.565** |

**What the two tables say together.** Each substrate shows a gain on the evaluation family its pairs can teach, and the second round extends it where the matched design licenses a read. On HelpSteer3 the conditioning effect appears on the open-ended protocols (Arena-Hard 33.1% to 37.2% at round 1, 45.3% and 45.6% after the second round) and on the BBH column of the official set; on Tulu3 it appears on the comprehensive-capability suite (average 58.52 to 60.34 at four billion, driven by BBH +8.35), while the open-ended protocols move flat to nominally negative. Neither substrate shows a gain in both families at once: per-substrate evaluation exposes the difference, and the behavioral diagnosis provides the basis for interpreting it—on HelpSteer3, the task-relevant behaviors the data rarely contrasts are where its benchmark columns stay flat (Sec. 5.2). The corresponding Tulu3 profile has since been measured and is markedly denser—it contrasts seven of the eight factors more often than HelpSteer3 does and has no sparse factor (Fig. 5)—so on that corpus the criterion has less left to add, which is the reading this paragraph would predict.

Table 46: Tulu3. Same layout. All open-ended cells in this table are scored by one judge (MiniMax-M3), so the four-billion and eight-billion rows are comparable here, unlike in Tab. 45. The second-round RUBRICOPT-R2 (KEEP) branch was never trained on this substrate, so the only RubricOpt continuation shown is RUBRICOPT-R2 (REFINE), the branch this substrate carries. The eight-billion BBH and MATH-Hard cells come from the unified re-measurement of appendix H.12 and carry no separate paired claim here.

| Comprehensive-capability | GSM8K | MATH | BBH | MuSR | ARC | TQA | MMLU-Pro | Avg. |
|---|---|---|---|---|---|---|---|---|
| 4B BASE | 85.90 | 93.70 | 30.95 | 43.39 | 62.80 | 52.27 | 41.98 | 58.71 |
| 4B DPO | 84.53 | 93.15 | 33.70 | 42.59 | 61.09 | 52.71 | 41.88 | 58.52 |
| 4B RUBRICOPT-R1 | **86.81** | **93.20** | **42.05** | **43.12** | **61.35** | **53.74** | **42.10** | **60.34** |
| 4B RUBRICOPT-R2 (REFINE) | 80.74 | 92.40 | 38.85 | 42.06 | 59.47 | 55.16 | 40.43 | 58.44 |
| 8B DPO | 91.80 | 93.45 | 70.00 | 55.20 | 94.65 | 76.80 | 67.35 | 78.46 |
| 8B RUBRICOPT-R1 | 91.40 | **95.15** | **73.80** | 55.15 | **95.00** | **77.85** | **68.30** | **79.52** |

| Open-ended generative | MT-Bench | Arena-Hard | WildBench |
|---|---|---|---|
| 4B DPO | 7.006 | 49.5% | 3.383 |
| 4B RUBRICOPT-R1 | 6.681 | 47.4% | 3.344 |
| 4B RUBRICOPT-R2 (REFINE) | **7.133** | 44.2% | 3.238 |
| 8B DPO | 7.595 | 58.0% | 3.641 |
| 8B RUBRICOPT-R1 | 7.316 | **61.1%** | 3.594 |

**Scale.** The eight-billion policy leads the four-billion policy on every benchmark of the comprehensive-capability suite benchmark in both substrates, and the lead is already present in the criterion-free DPO rows, which never activate a criterion. The scale difference is therefore a property of the base models and not of the method, and the two scales are reported side by side rather than combined.

# J STATISTICAL SUMMARY

This appendix collects the quantities a reader needs in order to judge how much of each claim is signal: effect sizes with intervals for every paired comparison, reliability statistics for the judge, and the resolution of the design. Nothing here is new evidence; it is the same measurements expressed in comparable units.

## J.1 EFFECT SIZES FOR EVERY PAIRED COMPARISON

All comparisons are item-level and paired on the prompt and sample index, which is why the intervals are narrow relative to the differences. The odds ratio is the McNemar ratio of discordant pairs (the count favouring the first method divided by the count favouring the second), so a value below one favours the first method; Cohen's $h$ is the arcsine-transformed difference of the two proportions and is reported as a signed effect size on the same scale as the difference.

Three readings follow from the effect sizes that the bare differences do not give. First, the two BBH results that the paper leans on have odds ratios of $0.52$ and $0.41$: the modified method wins the discordant pairs roughly twice as often as it loses them, which is a large paired effect even though the accuracy difference is under nine points. Second, the refinement comparison is measured tightly on MATH-Hard: RUBRICOPT-R2 (REFINE) against RUBRICOPT-R2 (KEEP) moves accuracy by $+0.05$ points with an odds ratio of $0.97$, and the $95\%$ interval $[-0.7, +0.8]$ points bounds the edit's incremental effect on that benchmark at about $0.8$ points; on BBH the interval is wider, $[-2.1, +1.1]$ points, so the two benchmarks bound the edit at different widths, and both intervals are item-paired within one seed. Third, the eight-billion second-substrate MATH-Hard cell (odds ratio $0.42$; appendix H.13) points the same way as the BBH results, so the paper's positive findings are not confined to one benchmark.

Table 47: Effect sizes for every paired comparison, grouped by model×substrate cell. The odds ratio is $b_{01}/b_{10}$ from the discordant pairs; values below one favour the first method. $h$ is Cohen's $h$. $p$ is the exact McNemar test. Comparisons are valid within a cell only.

| Cell | Bench. | Comparison | $\Delta$ | OR | $h$ | $p$ |
|---|---|---|---|---|---|---|
| 4B HS3 | MATH | Common − DPO | +0.0035 | 0.84 | +0.014 | 0.505 |
| 4B HS3 | MATH | RUBRICOPT-R1 − DPO | +0.0075 | 0.67 | +0.030 | 0.105 |
| 4B HS3 | MATH | RUBRICOPT-R1 − Common | +0.0040 | 0.79 | +0.016 | 0.403 |
| 4B HS3 | MATH | RUBRICOPT-R2 (REFINE) − RUBRICOPT-R2 (KEEP) | +0.0005 | 0.97 | +0.002 | 1.000 |
| 4B HS3 | MATH | RUBRICOPT-R2 (KEEP) − RUBRICOPT-R1 | −0.0015 | 1.08 | −0.006 | 0.824 |
| 4B Tulu3 | MATH | RUBRICOPT-R1 − DPO | +0.0005 | 0.98 | +0.002 | 1.000 |
| 8B HS3 | MATH | RUBRICOPT-R1 − DPO | −0.0030 | 1.15 | −0.012 | 0.594 |
| 8B Tulu3 | MATH | RUBRICOPT-R1 − DPO | +0.0170 | **0.42** | **+0.074** | **0.0003** |
| 4B HS3 | BBH | Common − DPO | +0.0230 | **0.70** | **+0.048** | **0.0053** |
| 4B HS3 | BBH | RUBRICOPT-R1 − DPO | +0.0110 | 0.84 | +0.023 | 0.179 |
| 4B HS3 | BBH | RUBRICOPT-R1 − Common | −0.0120 | 1.22 | −0.025 | 0.141 |
| 4B HS3 | BBH | RUBRICOPT-R2 (REFINE) − RUBRICOPT-R2 (KEEP) | −0.0050 | 1.09 | −0.010 | 0.563 |
| 4B HS3 | BBH | RUBRICOPT-R2 (KEEP) − RUBRICOPT-R1 | +0.0815 | **0.41** | **+0.167** | **< 0.0001** |
| 4B Tulu3 | BBH | RUBRICOPT-R1 − DPO | +0.0835 | **0.52** | **+0.172** | **< 0.0001** |
| 8B HS3 | BBH | RUBRICOPT-R1 − DPO | +0.0130 | 0.82 | +0.029 | 0.128 |
| 8B Tulu3 | BBH | RUBRICOPT-R1 − DPO | +0.0380 | **0.60** | **+0.085** | **< 0.0001** |

### J.2 JUDGE MEASUREMENT QUALITY

The judge's own reproducibility is summarised in Tab. 27, and the design of that audit is in appendix B.3.

### J.3 DISTRIBUTIONAL SHAPE

The factor contrasts that the supply analysis is built on are ordinal and heavily tied, and the tie mass is large enough to matter for any method that treats them as continuous.

Table 48: Shape of the pairwise factor contrast distribution over the 38,173 assessable HelpSteer3 pairs. The tie mass runs from 52.6% to 85.4%, so for most factors most pairs carry no information about that factor.

| Factor | Mean $\delta$ | Fraction $\delta > 0$ | Fraction $\delta = 0$ | Fraction $\delta < 0$ |
|---|---|---|---|---|
| $f_1$ task comprehension | +0.104 | 0.212 | 0.662 | 0.126 |
| $f_2$ reasoning validity | +0.091 | 0.103 | **0.854** | 0.043 |
| $f_3$ verification | +0.082 | 0.151 | 0.780 | 0.070 |
| $f_4$ factual grounding | +0.087 | 0.099 | **0.851** | 0.050 |
| $f_5$ completeness | +0.156 | 0.304 | **0.526** | 0.171 |
| $f_6$ instruction adherence | +0.099 | 0.205 | 0.660 | 0.136 |
| $f_7$ clarity and organization | +0.143 | 0.261 | 0.605 | 0.134 |
| $f_8$ efficiency and conciseness | +0.066 | 0.185 | 0.675 | 0.140 |

Every factor has more positive than negative mass, which is a property of the corpus rather than of the method: chosen responses tend to score higher on every factor. The informative quantity is therefore not the sign balance but the tie mass, because a tied pair cannot teach anything about that factor. Factors $f_2$ and $f_4$ are tied on 85% of pairs, and it is why the paper's coverage statistic $C_k$ rather than the mean contrast is the number that determines whether a factor is learnable from this corpus.

# K A Probabilistic Analysis of Semantic Rubric Steering

Semantic Rubric Steering changes the context in which a preference update is computed, whereas its benefit is measured under the policy without that context. This appendix connects these two distributions. The first result expresses the rubric's incremental effect on task success in the same semantic coordinates as the demand diagnosis. The second bounds how much success probability can fall during the remaining training after the rubric is withdrawn. Both results are conditional statements with explicit, measurable terms; neither assumes that the eight factors completely describe task success.

## K.1 Training randomness and the deployment distribution

**The probability space.** Fix a task $B$, a prompt distribution $\nu$, a reference policy, and the empirical distribution of preference triples $\xi = (\boldsymbol{x}', \boldsymbol{y}^+, \boldsymbol{y}^-)$ in $\mathcal{D}$. At a fixed parameter state $\theta$, write $c = \mathrm{crit}(\mathcal{G})$ and $0 = \varnothing$. Independently of $\xi$, draw $U = c$ with probability $q$ and $U = 0$ otherwise; both sides of the pair use the same $U$. Thus

$$\Pr(\xi, U) = \Pr_{\mathcal{D}}(\xi)\big[q\,\mathbf{1}\{U = c\} + (1-q)\,\mathbf{1}\{U = 0\}\big]. \tag{17}$$

This is a distribution over training examples and contexts. The deployment law is instead

$$P_\theta(\boldsymbol{x}, \boldsymbol{y}) = \nu(\boldsymbol{x})\pi_\theta(\boldsymbol{y} \mid \boldsymbol{x}, 0), \qquad p_B(\theta) = \mathbb{E}_{P_\theta}[o_B(\boldsymbol{x}, \boldsymbol{y})], \tag{18}$$

where the fixed binary function $o_B$ records task success. In particular, deployment does not average rubric-present and rubric-absent generations.

For a sufficient regularity setting, take a finite prompt set and finite response spaces (a finite vocabulary with a fixed generation cap), with a strictly positive, continuously differentiable policy on a neighborhood of every parameter path considered. All response scores and outcomes are fixed functions on these spaces, and $0 < p_B(\theta) < 1$. These assumptions justify interchanging differentiation and expectation. The results concern sampling from this policy; a greedy answer or an argmax benchmark decision is a different, potentially discontinuous readout.

**The implemented preference score.** To distinguish sequence probabilities from length-normalized training scores, let $L_\pm > 0$ be the fixed number of response tokens on each side and define

$$\begin{aligned} \ell^u_\pm(\theta; \xi) &= \frac{1}{L_\pm} \log \pi_\theta(\boldsymbol{y}^\pm \mid \boldsymbol{x}', u), \\ \widetilde{M}_u(\theta; \xi) &= (\ell^u_+ - \ell^u_{+,\mathrm{ref}}) - (\ell^u_- - \ell^u_{-,\mathrm{ref}}), \qquad \widetilde{\kappa}_u = \beta\sigma(-\beta\widetilde{M}_u). \end{aligned} \tag{19}$$

This is the mean-per-token convention recorded in the training protocol (Tab. 23). Response tokens and their lengths are identical across contexts, and rubric tokens enter neither side. The score $\ell^u_\pm$ is not the log probability of a newly normalized response distribution. For the sequence-sum convention in Eq. (6), replace $1/L_\pm$ by 1; the analysis below then holds with the corresponding training gradients.

Differentiating the logistic preference loss (Rafailov et al., 2023) gives the pair descent direction and its mean:

$$d_u(\xi) = \nabla_\theta \ell^u_+ - \nabla_\theta \ell^u_-, \qquad v_u = \mathbb{E}_{\xi\sim\mathcal{D}}[\widetilde{\kappa}_u d_u(\xi)], \qquad v_q = v_0 + q\Delta v, \quad \Delta v = v_c - v_0. \tag{20}$$

Here $v_u$ is the negative mean loss gradient. The last equality is exact for the mean raw gradient at a shared parameter state; it is not an equality between AdamW updates. The reference is frozen when differentiating.

**How a training pair changes deployment probability.** Let $S_0(\boldsymbol{x}, \boldsymbol{y}) = \nabla_\theta \log \pi_\theta(\boldsymbol{y} \mid \boldsymbol{x}, 0)$ be the *full-sequence* deployment score, without length normalization. Define

$$\begin{aligned} K_u(\boldsymbol{x}, \boldsymbol{y}; \xi) &= \langle S_0(\boldsymbol{x}, \boldsymbol{y}), d_u(\xi)\rangle, \\ h(\boldsymbol{x}, \boldsymbol{y}) &= \langle S_0(\boldsymbol{x}, \boldsymbol{y}), \Delta v\rangle \\ &= \mathbb{E}_{\xi\sim\mathcal{D}}[\widetilde{\kappa}_c K_c(\boldsymbol{x}, \boldsymbol{y}; \xi) - \widetilde{\kappa}_0 K_0(\boldsymbol{x}, \boldsymbol{y}; \xi)]. \end{aligned} \tag{21}$$

The function $h$ is the incremental rate of change of the deployment log probability along the rubric-induced change of direction. It has $\mathbb{E}_{P_\theta}[h \mid \boldsymbol{x}] = 0$, since the conditional mean of $S_0$ is zero. Eq. (21) makes the data dependence explicit: each fixed preference pair contributes through its context-dependent gradient alignment with a deployment response. No factor scores are inserted into the training loss.

### K.2 When semantic steering improves success probability

All expectations in this subsection are under $P_\theta$ at one fixed state. Write $p = p_B(\theta)$, $z = (z_1, \ldots, z_K)^\top$, $Z = z - \mathbb{E}[z]$, and

$$D = \mathbb{E}[z \mid o_B = 1] - \mathbb{E}[z \mid o_B = 0], \qquad \Sigma = \mathbb{E}[ZZ^\top]. \tag{22}$$

The vector $D$ is the population version of $D_{m,B,k}$ for $m = \pi_\theta$ when diagnostic responses follow this sampling law. Define the least-squares projections of $h$ and the centered outcome onto the semantic coordinates:

$$\begin{aligned} a &= \Sigma^\dagger \mathbb{E}[Zh], & r_h &= h - a^\top Z, & \epsilon_h^2 &= \mathbb{E}[r_h^2], \\ b &= \Sigma^\dagger \mathbb{E}[Z(o_B - p)], & r_o &= o_B - p - b^\top Z, & \epsilon_o^2 &= \mathbb{E}[r_o^2]. \end{aligned} \tag{23}$$

Here $\Sigma^\dagger$ is the Moore–Penrose pseudoinverse, so correlated or redundant factors are permitted. The coefficients $a$ summarize the change in deployment probabilities induced by a rubric; they are derived quantities, not trainable factor weights or a linear reward-model assumption.

**Proposition 2 (Semantic decomposition of the incremental success odds)** *Under the preceding assumptions, let $\theta_q(\eta) = \theta + \eta v_q$ and $\theta_0(\eta) = \theta + \eta v_0$, with directions held fixed at $\theta$. Writing $\operatorname{logit}(p) = \log[p/(1-p)]$, the exact directional identity is*

$$\frac{d}{d\eta}\left\{\operatorname{logit} p_B(\theta_q(\eta)) - \operatorname{logit} p_B(\theta_0(\eta))\right\}\Big|_{\eta=0} = q\big(D^\top a + \mathcal{R}\big), \tag{24}$$

*where*

$$\mathcal{R} = \frac{\mathbb{E}[r_o r_h]}{p(1-p)}, \qquad |\mathcal{R}| \le \frac{\epsilon_o \epsilon_h}{p(1-p)}, \qquad \epsilon_o^2 = p(1-p) - p^2(1-p)^2 D^\top \Sigma^\dagger D. \tag{25}$$

*Consequently, if $q > 0$ and*

$$D^\top a > \frac{\epsilon_o \epsilon_h}{p(1-p)}, \tag{26}$$

*then $p_B(\theta_q(\eta)) > p_B(\theta_0(\eta))$ for all sufficiently small positive $\eta$. If $D^\top a < -\epsilon_o \epsilon_h / [p(1-p)]$, the strict inequality is reversed for sufficiently small positive $\eta$.*

**Proof.** For any fixed direction $w$, differentiating the deployment expectation gives

$$\nabla_\theta p_B^\top w = \mathbb{E}[(o_B - p)\langle S_0, w\rangle]. \tag{27}$$

Taking $w = \Delta v$ and dividing by $p(1-p)$ expresses the incremental log-odds derivative as

$$\frac{\mathbb{E}[(o_B - p)h]}{p(1-p)} = \mathbb{E}[h \mid o_B = 1] - \mathbb{E}[h \mid o_B = 0]. \tag{28}$$

Likewise, conditioning on the binary outcome gives $\mathbb{E}[Z(o_B - p)] = p(1-p)D$.

The normal equations for Eq. (23) hold even when $\Sigma$ is singular. Indeed, $w \in \ker \Sigma$ implies $\mathbb{E}[(w^\top Z)^2] = 0$, hence $w^\top Z = 0$ almost surely; both covariance vectors being projected are therefore in the range of $\Sigma$. Thus $\mathbb{E}[Zr_h] = \mathbb{E}[Zr_o] = 0$, and

$$\mathbb{E}[(o_B - p)h] = p(1-p)D^\top a + \mathbb{E}[r_o r_h]. \tag{29}$$

Cauchy–Schwarz bounds the last expectation by $\epsilon_o \epsilon_h$. The least-squares residual variance for $o_B - p$ is $p(1-p) - \mathbb{E}[Z(o_B - p)]^\top \Sigma^\dagger \mathbb{E}[Z(o_B - p)]$, which gives Eq. (25). Finally, $v_q - v_0 = q\Delta v$ proves Eq. (24). A strictly positive derivative of the log-odds difference, which is zero at $\eta = 0$, makes that difference positive for sufficiently small positive $\eta$; the logit is strictly increasing. The negative case follows identically. □

**What the decomposition explains.** The term $D^\top a$ couples a task-side measurement to the actual effect of the training context: a rubric is locally useful when its induced probability change favors the behaviors associated with success strongly enough to dominate the residual. Positive demand alone does not imply positive transfer, since the rubric need not produce a positive coefficient on that factor. Conversely, if either the centered task outcome or the incremental probability change lies entirely in the linear span of the measured factors, its corresponding residual is zero and the semantic decomposition is exact. Otherwise, the residual explicitly retains behavior outside that span. This extends Prop. 1 from a generic behavioral gradient to a task-success readout in the paper's diagnostic coordinates.

The data-side contribution is the expectation over pairs in Eq. (21). Changing the substrate can change both $a$ and the residual even with the same rubric. The coverage $C_k$ and signed contrasts $(\mu_k, P_k)$ describe observed factor differences, but do not determine $K_u$ or $a$. In particular, a tie in an ordinal factor score does not force a pair's policy gradient to vanish. Thus sparse measured supply motivates a diagnosis; it is not an impossibility theorem for learning that behavior. This distinction also explains why factor coverage or gradient cosine alone cannot establish the sign in Eq. (26).

### K.3 When gains survive rubric withdrawal

A positive local steering effect must persist until deployment. To quantify this requirement without replacing the actual optimizer by gradient flow, consider any piecewise continuously differentiable parameter path $\theta(t)$, $\tau \le t \le T$. For discrete training, this can be the linear interpolation through the realized checkpoints. Let $\tau$ mark the end of rubric exposure, and let $T$ be the final checkpoint. All probabilities below remain criterion-free, including at $\tau$. Define the deployment Fisher matrix and the length of the remaining path by

$$F(\theta) = \mathbb{E}_{P_\theta}[S_0 S_0^\top], \qquad \mathcal{L}_{\tau:T} = \int_\tau^T \sqrt{\dot\theta(t)^\top F(\theta(t))\dot\theta(t)}\, dt. \tag{30}$$

This length measures change in output probabilities, not Euclidean distance between parameter vectors. It needs only a directional second moment of the score, not an inverse Fisher matrix.

**Proposition 3 (Retention after withdrawal)** *Suppose the probability-space assumptions above hold along the path and $\mathcal{L}_{\tau:T} < \infty$. With $\alpha_t = \arcsin\sqrt{p_B(\theta(t))} \in (0, \pi/2)$,*

$$|\alpha_T - \alpha_\tau| \le \tfrac{1}{2}\mathcal{L}_{\tau:T}. \tag{31}$$

*In particular,*

$$p_B(\theta(T)) \ge \sin^2\left(\max\left\{0, \arcsin\sqrt{p_B(\theta(\tau))} - \tfrac{1}{2}\mathcal{L}_{\tau:T}\right\}\right). \tag{32}$$

*For a comparison policy with success probability $p_\star \in (0,1)$ under the same prompt distribution and outcome function, a sufficient condition to retain a strict advantage at $T$ is*

$$p_B(\theta(\tau)) > p_\star, \qquad \mathcal{L}_{\tau:T} < 2\left(\arcsin\sqrt{p_B(\theta(\tau))} - \arcsin\sqrt{p_\star}\right). \tag{33}$$

*The constant $1/2$ in Eq. (31) is sharp.*

**Proof.** Write $p_t = p_B(\theta(t))$ and $h_t = \langle S_0, \dot\theta(t)\rangle$. The score identity gives $\mathbb{E}[h_t] = 0$ and $\dot p_t = \mathbb{E}[(o_B - p_t)h_t]$. Hence

$$|\dot p_t| \le \sqrt{p_t(1-p_t)}\sqrt{\mathbb{E}[h_t^2]} = \sqrt{p_t(1-p_t)}\sqrt{\dot\theta(t)^\top F(\theta(t))\dot\theta(t)}. \tag{34}$$

Since $\dot\alpha_t = \dot p_t/[2\sqrt{p_t(1-p_t)}]$, integration on each smooth segment and the triangle inequality prove Eq. (31). Clipping its lower bound at zero and using monotonicity of $\sin^2$ on $[0, \pi/2]$ gives Eq. (32) and Eq. (33). For sharpness, consider one prompt and two responses, with success probabilities $(\sin^2\alpha, \cos^2\alpha)$ and $\alpha \in (0, \pi/2)$. The Fisher information for parameter $\alpha$ is 4. Every monotone path then has $\mathcal{L}_{\tau:T} = 2|\alpha_T - \alpha_\tau|$, achieving equality. □

**The role of the withdrawal schedule.** The bound explains why a training-only rubric can leave a persistent effect: removing the text changes future updates but does not reset the learned policy. For the idealized mean-gradient path $\dot{\theta} = v_0 + q\Delta v$, the Fisher seminorm satisfies

$$\sqrt{\dot{\theta}^\top F \dot{\theta}} \le \sqrt{v_0^\top F v_0} + q\sqrt{\Delta v^\top F \Delta v}. \tag{35}$$

All terms are evaluated at the evolving policy. Withdrawal removes the explicit rubric component as $q$ reaches zero, while the ordinary preference updates continue to move deployment probabilities. Eq. (33) states how small that remaining movement must be to preserve an existing advantage. The cosine schedule alone does not impose this condition or prove that it is an optimal schedule.

For actual optimizer increments $\delta_j = \theta_{j+1} - \theta_j$, linear interpolation gives the exact path length

$$\mathcal{L}_{\tau:T} = \sum_{j=\tau}^{T-1} \int_0^1 \sqrt{\delta_j^\top F(\theta_j + s\delta_j)\delta_j}\, ds. \tag{36}$$

This form applies to a realized AdamW trajectory, including momentum and weight decay. Endpoint score measurements can approximate the integral but are not certified upper bounds without additional control between checkpoints. The comparison $p_\star$ can be the final ordinary-DPO policy; its value and the tail length must both be established to invoke the sufficient condition.

### K.4 Connection to the evidence and to rubric refinement

The two results separate the creation of a useful probability change from its retention. The success-odds decomposition identifies how the same labeled pairs can have different task effects under different rubrics and different substrates. The retention bound states when those effects can survive a final phase with no rubric. These are applications of score-function identities, least-squares projection, and Cauchy–Schwarz to the method, rather than claims of new general probability inequalities.

The same first result analyzes an accepted wording edit: at a shared policy and reference, replace $\Delta v = v_c - v_0$ by $v_{R_1} - v_{R_0}$, where each velocity is computed on the same preference pairs. It then describes the local edit increment under a common exposure rate. Passing the deterministic textual gate does not enforce the sign of $D^\top a + \mathcal{R}$. Thus an evidence-grounded edit can pass the gate without improving the task beyond unchanged-rubric continuation, as in the tested BBH comparison. This is a local interpretation, not a monotonic-improvement guarantee for recursive training.

The reported benchmark gains, context-gradient probes, and factor-score shifts measure different parts of this account. Neither the projection coefficients and residuals nor the withdrawal path length have been measured in the current experiments. Checking the first condition requires rubric-free policy samples, their outcomes and factor scores, and paired context gradients at the same checkpoint. Checking retention requires rubric-free success probabilities and a bound on the subsequent probability path length. Finite-sample estimates would additionally require uncertainty analysis. The existing results therefore establish the empirical effects reported in the main text; the propositions specify testable conditions for explaining those effects without treating an unmeasured mechanism as an established finding.