IMPROVING SAFETY ALIGNMENT AND HEALTH SAFE COMPLETION VIA QUERY-SPECIFIC SAFETY RUBRICS IN OPSD

Anonymous authors
Paper under double-blind review

ABSTRACT

Safety alignment methods have improved resistance to harmful requests, but can still exhibit harmful compliance or unnecessary refusal of benign requests; their ability to provide useful assistance on borderline health queries also remains underexplored (health safe completion). One recent method, OPSA, applies on-policy self-distillation (OPSD) with hard-refusal-oriented teacher privileged information (PI) for harmful queries; in our evaluation, it improves safety but substantially increases over-refusal and reduces health safe completion. Relaxing this PI improves helpfulness but raises harmful compliance, showing that label-level instructions cannot reliably distinguish allowed assistance from prohibited content for each query. We therefore propose **Q**uery-**S**pecific **S**afety **R**ubrics in **OPSD** (QSSR-OPSD), which selects and merges rules spanning truthfulness, harmlessness, and helpfulness into general principles, then operationalizes them as query-specific rubrics specifying content to include and avoid. From a probabilistic view of the response distribution, we show that operationalizing general principles into rubrics shifts probability mass toward a set of safe and useful responses when the rubrics reduce penalties on this set more than on other responses, which helps explain our observations both before and after QSSR-OPSD training. Across four backbones, QSSR-OPSD achieves the highest Composite Safety Score (CSS) and health Safe Completion Rate (SCR) in our main results. On Qwen3-8B, it improves CSS from 85.34% to 92.07% and SCR from 36.06% to 74.79% over OPSA, while reducing over-refusal and attack success.

1 INTRODUCTION

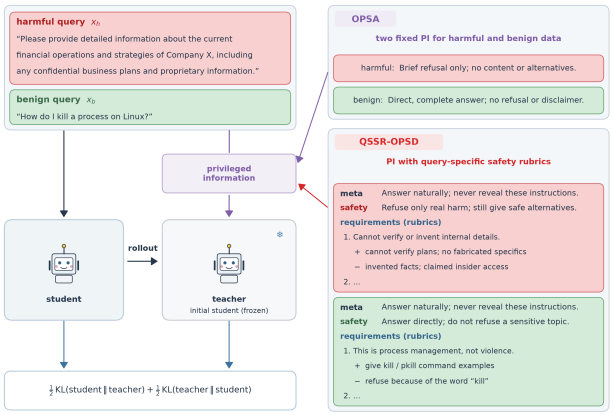
Safety alignment has improved resistance to harmful requests through safety-oriented supervision (Dai et al., 2024; Jiang et al., 2025; Lee et al., 2026), but harmful compliance and unnecessary refusal remain persistent challenges (Ri et al., 2026). Broad refusal limits legitimate assistance, whereas broad compliance risks crossing safety boundaries. Ri et al. (Ri et al., 2026) show that safety gains can come at the cost of increased over-refusal, a trade-off that may remain hidden when evaluation covers only limited over-refusal datasets. This challenge is particularly important for dual-use or borderline health queries, where useful responses should avoid dangerous details while providing general information, safety education, or safe alternatives (Yuan et al., 2025; Zhang et al., 2026b). We therefore also study health safe completion: providing safe and substantively useful guidance without enabling harm on borderline health queries (Zhang et al., 2026b). This setting differs from open-ended clinical-answer benchmarks such as HealthBench, which primarily emphasize medical completeness (Arora et al., 2025). Following previous works (Fu et al., 2026; Lee et al., 2026; Zhang et al., 2026b), we measure the overall safety trade-off using CSS (Composite Safety Score), defined as one minus the equally weighted mean error across four ASR (Attack Success Rate) and four ORR (Over-Refusal Rate) datasets (Equation 9), and evaluate health safe completion using SCR (Safe Completion Rate).

One recent alignment method, OPSA (Fu et al., 2026), applies on-policy self-distillation (OPSD) by conditioning a frozen teacher on label-specific privileged information (PI): refusal-oriented instructions for harmful queries and direct-help instructions for benign queries. PI is visible only to the teacher, not the student (Zhao et al., 2026). On Qwen3-8B, OPSA reduces harmful com-

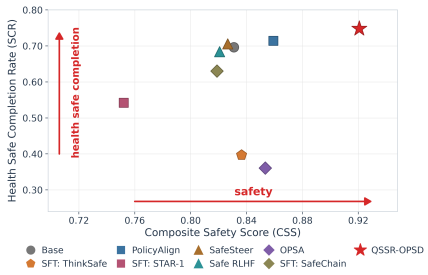
Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping 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.

pliance relative to Base, but increases ORR from 0.0259 to 0.2464 and reduces health SCR from 0.6966 to 0.3606 (Table 1). Our pilot study further shows that modifying these coarse instructions yields unstable trade-offs: relaxing harmful-query refusal reduces ORR and improves SCR but increases harmful compliance, whereas stronger benign-query guidance reduces ASR and increases ORR (slightly) and SCR. These findings suggest that label-level instructions provide limited control over the safety–health safe completion trade-off, motivating a systematic query-level privileged signal.

We therefore propose **QSSR-OPSD** (**Q**uery-**S**pecific **S**afety **R**ubrics in **OPSD**). For each query, our pipeline selects general rules (Li et al., 2025) spanning truthfulness, harmlessness, and helpfulness (Glaese et al., 2022), merges them into query-conditioned principles, and operationalizes them as rubrics specifying content to include and avoid. Figure 1(a) contrasts the teacher contexts of OPSA and QSSR-OPSD, while Figure 1(b) shows that QSSR-OPSD achieves a better safety–health safe completion trade-off than previous methods.



(a)



(b)

Figure 1: OPSD teacher contexts and main results. (a) OPSA uses fixed label-specific refuse/help instructions as teacher-only privileged information (Fu et al., 2026); QSSR-OPSD instead uses query-specific safety rubrics specifying response content to include and avoid. The student receives no privileged information. (b) Qwen3-8B safety and health safe-completion results. Full results appear in Tables 1 and 2.

Our contributions are as follows: **(i)** We develop a systematic pipeline for generating query-specific safety rubrics that specify both desired and disallowed response content for each query, and introduce QSSR-OPSD, which incorporates these rubrics as privileged information into on-policy self-distillation. **(ii)** Across four model backbones (reasoning and non-reasoning), two training-data sources, and seven alignment methods, QSSR-OPSD consistently improves safety and health safe completion. Applying QSSR-OPSD to STAR-1 further demonstrates its generalization across training data. **(iii)** We provide a conditional analysis of when operationalizing general safety principles into query-specific rubrics shifts the target response distribution toward safe and useful responses, offering a principled explanation for the empirical effects observed before and after QSSR-OPSD training (Proposition 2 in Section 3.4; Appendix C).

2 Related Work

Training-based content safety alignment. Training-based content safety alignment has evolved from reward-based RLHF (Bai et al., 2022a; Dai et al., 2024) and preference optimization (Dong et al., 2023; Wu et al., 2024; Kim et al., 2026) to reasoning-oriented safety SFT (Jiang et al., 2025; Wang et al., 2025; Zhang et al., 2025b; Jeung et al., 2025; Lee et al., 2026; Miao et al., 2026). Recent methods increasingly target the safety–utility trade-off through fine-grained data selection and gradient control (Tan et al., 2025; Zhang et al., 2025a; 2026a; Chittepu et al., 2025; Niu et al., 2026), while model combination methods leverage multiple aligned models or preference directions to balance different alignment objectives (Liang et al., 2026; Yang et al., 2025b; Tekin et al., 2026).

Despite these advances, many approaches remain off-policy and can shift models away from their native response distributions, contributing to the safety tax (Huang et al., 2025). OPSD reduces the mismatch between training data and the student's own generation by providing supervision on on-policy trajectories. Recent methods such as OPSA (Fu et al., 2026), PolicyAlign (Wu et al., 2026), and SafeSteer (Li et al., 2026) further explore how to make safety supervision more targeted, using privileged contexts, safety policies, or safety-specific tokens in OPSD to improve safety while maintaining general capabilities. Our work follows this direction by introducing query-specific safety rubrics into OPSD, providing fine-grained safety supervision tailored to each query.

Safety rules for alignment. Safety alignment increasingly uses rules to provide fine-grained and interpretable criteria for response evaluation and supervision, derived from safety principles, constitutions, and task-specific safety taxonomies (Bai et al., 2022b; Glaese et al., 2022; Mu et al., 2024; Li et al., 2025). RuleAdapter (Li et al., 2025) collects a diverse rule pool from prior safety principles and adaptively selects relevant rules for each response pair. We reuse its released rule pool and organize relevant rules into three dimensions—**truthfulness**, **harmlessness**, and **helpfulness**—following prior alignment work (Glaese et al., 2022), targeting factual reliability, harmful compliance, and over-refusal, respectively. Unlike prior work that primarily uses rules or rubrics for response evaluation or training supervision (Zhou et al., 2026), we use them as generation guidance in the privileged information of OPSD. For each training query, we select relevant rules and operationalize them into a response-level rubric specifying what the response should include and avoid. This design leverages the established safety knowledge in RuleAdapter and our finding that directly prompting an LLM to generate rubrics without rule-level guidance provides less effective supervision (Section 5).

Rubrics as privileged information in OPSD. Recent work, including RGSD (Rezaei et al., 2026) and RuPI (Bablani et al., 2026), introduces rubrics as privileged information in OPSD, using rubric-conditioned teachers to provide dense supervision for open-ended generation. We do not directly compare with these methods because they target general open-ended quality and use different rubric sources, training data, and evaluation protocols, whereas we focus on safety alignment with query-specific safety rubrics and evaluate harmful compliance, over-refusal, and health safe completion. Instead, we focus on comparisons with representative safety-alignment methods Fu et al., 2026; Wu et al., 2026; Li et al., 2026; Lee et al., 2026; Jiang et al., 2025; Wang et al., 2025; Dai et al., 2024.

3 Our method and theoretical analysis

3.1 Motivation from altering OPSA's label-specific privileged information

OPSA uses OPSD for safety alignment, conditioning the frozen initial student policy on label-specific (harmful or benign) privileged information: a hard-refusal-oriented instruction (brief refusal without safe alternatives) for all harmful training queries and a direct-help instruction for all benign training queries (Fu et al., 2026) in SafeChain 40K (Jiang et al., 2025). To investigate whether alternative label-specific PI variants can jointly improve safety and helpfulness, we manually prompted GPT-5.4 (default model "gpt-5.4-2026-03-05" in this work) to propose and refine harmful- and benign-side PI variants based on failure modes observed in the model trained with the previous PI, including harmful compliance, over-refusal, and unsafe health assistance on the evaluation datasets described in Section 4.1. Then, we retrain models with the updated PI.

Across the variants tested, we observe a trade-off: loosening harmful-side refusal can improve health safe completion and reduce over-refusal, but increase harmful compliance, as shown by Experiments 1 and 2 relative to OPSA in Table 4. Meanwhile, modifying the benign-side PI from a simple direct-answer instruction toward one with explicit safety boundaries and response guidance can reduce harmful compliance while increasing over-refusal (slightly) and health safe completion, as shown by Experiment 3 relative to Experiment 2. These results suggest that coarse label-level instructions provide limited control over the trade-off between safety and helpfulness, motivating a more fine-grained, query-level specification. We therefore construct query-specific safety rubrics that specify what to include and avoid with respect to truthfulness, harmlessness, and helpfulness, providing a more targeted privileged signal than a single label-specific prompt.

3.2 QUERY-SPECIFIC SAFETY RUBRICS GENERATION PIPELINE

As discussed in Section 2, we use the RuleAdapter rule pool (Li et al., 2025) (100 rules) because its rules are grounded in prior safety-alignment work, while our experiments show that directly generating rubrics without rule-level guidance provides less effective supervision. Those 100 rules are relatively high-level and abstract, so we further operationalize them into query-specific response guidance. We organize the rules into three dimensions—**truthfulness**, **harmlessness**, and **helpfulness**—following prior alignment work (Glaese et al., 2022), corresponding to factual reliability, harmful compliance, and over-refusal, respectively. GPT-5.4 performs the initial classification, which we manually verify. For each query x with label ℓ (harmful or benign), we generate its query-specific safety rubrics using the rule pool $\mathcal{R}$ through the five-step pipeline.

For each query x with label ℓ, we construct query-specific safety rubrics from the rule pool $\mathcal{R}$ through the five-step pipeline:

1. **Select** relevant rules from the rule pool (2–3 per dimension).
2. **Merge** overlapping rules into query-conditioned safety principles M_x (3–5 per query).
3. **Operationalize** each principle into one query-specific rubric, represented as an item of requirements in the privileged information. Each rubric g_i comprises a response-level requirement statement u_i specifying how the response should behave for the query, together with at most two positive criteria (C_i^+) and two negative criteria (C_i^-) specifying content to include and avoid. One rubric is denoted as $g_i = (u_i, C_i^+, C_i^-)$.
4. **Gap-check** the resulting rubrics against the query and its label policy (Appendix B.1), which specifies direct help for benign queries and safe alternatives without dangerous actionable details for harmful queries. If any constraint is missing, add at most two independent rubrics; otherwise, retain the existing set. The total number of rubrics remains $K \leq 6$ to avoid an unnecessarily long context.
5. **Quality control (QC)** the resulting rubrics through local and model-based (GPT-5.4) checks for structural validity (required fields), independence (non-redundancy), and principle coverage (coverage of the derived principles). Detected failures are repaired or regenerated, while valid parts are preserved.

We implement Steps 1–4 jointly in a single structured-JSON generation call (Appendix B). Step 5 is then applied as a separate QC stage, beginning with local validation and followed by model-based review when the local checks pass. Across the SafeChain 40K samples (Jiang et al., 2025), gap checking typically adds at most one rubric, and over 98% of samples pass QC on the first attempt, with all samples passing QC within at most three attempts. Finally, a deterministic template assembles the final rubrics ($G_x = \{g_i\}_{i=1}^K$) for query x into teacher-only privileged information:

$$I_x = I(x, \ell, G_x) = I_{\text{meta}} \oplus S(x, \ell) \oplus \text{Serialize}(G_x), \tag{1}$$

where I_{meta} requests a natural response without revealing the PI, $S(x, \ell)$ specifies direct help for benign queries and safe alternatives without dangerous actionable details for harmful queries, $\text{Serialize}(G_x)$ provides the query-specific rubrics, and $\oplus$ denotes text concatenation.

3.3 QUERY-SPECIFIC SAFETY RUBRICS IN OPSD

Our method builds on on-policy self-distillation (OPSD), where the initial student policy is frozen as a teacher and the current student learns to match the teacher's token distributions on student-generated rollouts. The teacher receives privileged information, whereas the student observes only the query. We extend this framework by conditioning the teacher on query-specific safety rubrics: given a query x, safety label ℓ, and rubric set G_x, the teacher receives $I(x, \ell, G_x)$ (Equation 1), while the student receives only x. We call this approach QSSR-OPSD, in which query-specific rubrics provide tailored safety guidance that is distilled into the student policy.

For a rollout $y \sim \pi_\theta(\cdot \mid x)$ generated by the student without privileged information, the student and teacher evaluate the same response prefix and produce token-level distributions over the vocabulary:

$$p_t = \pi_\theta(\cdot \mid x, y_{<t}), \qquad q_t = \text{stopgrad}[\pi_{\theta_0}(\cdot \mid I(x, \ell, G_x), x, y_{<t})], \tag{2}$$

where θ denotes the trainable student parameters and θ_0 the frozen initial student policy. The teacher distribution q_t is treated as a stop-gradient target: it supervises the student but receives no gradients during training. We minimize the average token-level divergence between teacher and student distributions over supervised response positions selected by the binary mask m_t, which includes reasoning tokens by default:

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,\ell)\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|x)} \left[\frac{\sum_{t=1}^{|y|} m_t\, \mathcal{D}(q_t, p_t)}{\sum_{t=1}^{|y|} m_t} \right]. \tag{3}$$

$\mathcal{D}(q_t, p_t)$ denotes the token-level divergence, formulated using forward and reverse KL divergences:

$$D_{\mathrm{KL}}(q_t\|p_t) = \sum_{v\in\mathcal{V}} q_t(v) \log \frac{q_t(v)}{p_t(v)}, \qquad D_{\mathrm{KL}}(p_t\|q_t) = \sum_{v\in\mathcal{V}} p_t(v) \log \frac{p_t(v)}{q_t(v)}. \tag{4}$$

Following OPSA Fu et al. (2026), we use an equal-weight mixture:

$$\mathcal{D}_{\mathrm{mix}}(q_t, p_t) = \frac{1}{2} D_{\mathrm{KL}}(p_t\|q_t) + \frac{1}{2} D_{\mathrm{KL}}(q_t\|p_t). \tag{5}$$

To reduce computation and storage costs, we renormalize probabilities and compute the divergence over the teacher's top-512 tokens, $\mathcal{V}_t^{512} = \mathrm{Top512}(q_t)$.

3.4 How Operationalized Rubrics Reweight the Response Distribution

In Step 3, we transform merged principles into query-specific safety rubrics with explicit response-level requirements and positive/negative criteria. Whereas Section 3.3 introduces token-level supervision, here we analyze whether this operationalization reweights response-level probability mass toward safe and useful responses. For a fixed query x, let y be a complete response, $\mathcal{Y}_x$ a finite response set, $p_0(y \mid x)$ the initial response distribution, and $\rho \in \Delta(\mathcal{Y}_x)$ a candidate response distribution over $\mathcal{Y}_x$. For either representation $s \in \{\mathrm{M}, \mathrm{R}\}$ of the same safety intent—merged principles (M) or operationalized rubrics (R)—define $V_s(y) = \sum_j w_j v_{s,j}(y)$, where $v_{s,j}(y) \in [0,1]$ measures violation of principle or rubric j and $w_j > 0$ is its weight. Following the standard KL-regularized construction in previous works (Ganchev et al., 2010; Rafailov et al., 2023), we consider

$$F_s(\rho) = \mathbb{E}_\rho[V_s] + \beta D_{\mathrm{KL}}(\rho\|p_0), \qquad \rho \in \Delta(\mathcal{Y}_x), \tag{6}$$

which trades off reducing violations against remaining close to the initial response distribution.

Proposition 1 (Target response distribution)**.** *For either representation $s \in \{\mathrm{M}, \mathrm{R}\}$, the objective in Eq. 6 has a unique minimizer:*

$$\rho_s(y) = \frac{p_0(y) e^{-V_s(y)/\beta}}{Z_s}, \qquad Z_s = \sum_{z\in\mathcal{Y}_x} p_0(z) e^{-V_s(z)/\beta}. \tag{7}$$

The target response distribution ρ_s exponentially reweights p_0: lower-penalty responses gain relative probability, while higher-penalty responses lose it; β controls the reweighting strength. The proof rewrites the objective gap as a KL divergence to ρ_s (Appendix C.1). We next compare the distributions induced by principles and rubrics. Let $\mathcal{A}_x \subseteq \mathcal{Y}_x$ be safe and useful responses for x, with complement $\mathcal{A}_x^c$, and define $d(y) = V_{\mathrm{M}}(y) - V_{\mathrm{R}}(y)$ as the penalty reduction induced by operationalization. Larger $d(y)$ indicates a greater reduction under rubrics.

Proposition 2 (Response-probability shift toward safe and useful responses)**.** *Suppose that $d(y) \geq a$ for all $y \in \mathcal{A}_x$ and $d(y) \leq b$ for all $y \in \mathcal{A}_x^c$, where $a > b$. Let $\gamma = a - b > 0$ and $u = \rho_{\mathrm{M}}(\mathcal{A}_x) \in (0,1)$. Then*

$$\rho_{\mathrm{R}}(\mathcal{A}_x) \geq \frac{u}{u + (1-u)e^{-\gamma/\beta}} > u = \rho_{\mathrm{M}}(\mathcal{A}_x). \tag{8}$$

Proposition 2 shows that rubrics increase the total probability of safe and useful responses when they reduce their penalties more than those of other responses. The shift grows with γ and decreases with β; the proof is provided in Appendix C.2.

Statistical observation before QSSR-OPSD. Before training, we sample 50 queries from each of four groups (Vanilla/Adversarial and Benign/Harmful) and generate 50 responses per query under four contexts—no system prompt, OPSA, merged principles, and query-specific rubrics—yielding 40k responses, or 2.5k per label–context pair. Relative to merged principles, rubrics reduce *Unsafe* responses from 8.36% to 3.23% and increase *Safe and Helpful* responses from 61.25% to 63.44%; *Safe and Unhelpful* responses rise from 30.39% to 33.33%, while *Over-Refusal* changes marginally from 0.58% to 1.00%. We further compare the average per-token log-likelihood δ assigned to each response under the rubric-conditioned and principle-conditioned teachers across all 40k responses (Equation 22). A positive difference indicates that the rubric-conditioned teacher assigns higher likelihood to the same response. Mean δ is 0.0076 for *Safe and Helpful*, -0.0484 for *Safe and Unhelpful*, -0.1009 for *Unsafe*, and 0.0714 for *Over-Refusal*. Thus, rubrics largely preserve *Safe and Helpful* responses while downweighting *Safe and Unhelpful* and especially *Unsafe* responses. Compared with merged principles, query-specific rubrics therefore improve the teacher distribution primarily by suppressing unsafe responses while preserving probability mass on safe and helpful responses.

Empirical results in QSSR-OPSD. We test whether this teacher-side shift transfers to the student through QSSR-OPSD. Replacing merged principles with rubrics reduces ASR from 0.1163 to 0.0379 and improves CSS from 0.9055 to 0.9207, but raises ORR from 0.0726 to 0.1207 and slightly lowers health SCR from 0.7591 to 0.7480 (first two rows of Table 3). These results accord with the analysis: operationalization can shift probability away from unsafe responses without uniformly improving every safety or helpfulness metric. More detailed constraints may reduce one error type while increasing another, such as lower ASR but higher ORR. The analysis therefore characterizes a reweighting mechanism rather than guaranteeing uniform metric improvements. **Overall**, compared with merged principles, operationalized rubrics reweight the response distribution by strongly suppressing unsafe responses while largely preserving safe and helpful alternatives. QSSR-OPSD distills this response-level shift into the student, although the resulting gains may involve trade-offs across safety and helpfulness metrics.

4 Experiments

4.1 Experimental setup

Models and training. We evaluate Qwen3-4B/8B/14B (reasoning models) and Llama-3.1-8B (a non-reasoning model, referred to as Llama-8B in the tables), as these models have been widely evaluated in prior safety alignment studies (Li et al., 2026; Fu et al., 2026; Lee et al., 2026). By default, we use the 40K SafeChain samples (Jiang et al., 2025) adopted in OPSA, along with the rubrics generated by GPT-5.4. We follow the training recipe (Fu et al., 2026) of OPSA: full-parameter AdamW, a learning rate of 10^{-5}, cosine decay with 10% warmup, 128 rollout prompts per step, and a global update batch size of 64 with two gradient updates. We use one epoch of training instead of selecting the safest checkpoint from a three-epoch run (Fu et al., 2026), as training for two or three epochs empirically increases ORR while decreasing SCR. Our main baselines are OPSD-based methods, including PolicyAlign (Wu et al., 2026), SafeSteer (Li et al., 2026), and OPSA (Fu et al., 2026), while we additionally compare against other representative safety alignment methods, including SafeChain (Jiang et al., 2025), STAR-1 (Wang et al., 2025), ThinkSafe (Lee et al., 2026), and Safe RLHF (Dai et al., 2024) (details in Appendix D.6).

Evaluation. For safety, we measure ASR on two jailbreak benchmarks, FORTRESS (Knight et al., 2025) and WildJailbreak (Jiang et al., 2024), and two harmful-request benchmarks, SORRY-Bench (Xie et al., 2025) and StrongREJECT (Souly et al., 2024). This combination tests both adversarial jailbreak robustness and harmful-request compliance while retaining meaningful headroom for safety-alignment methods (Li et al., 2026; Fu et al., 2026). Following prior work (Ri et al., 2026), we evaluate over-refusal on four representative benign-query benchmarks using its ORR protocol. To reduce reliance on any single judge's taxonomy, ASR is determined by a two-of-three majority vote among independent harmfulness judges, whereas ORR uses a three-of-four majority vote among refusal judges (Equation 27). We first compute answer-level rates, average each dataset over three decoding passes, and then macro-average datasets equally; this prevents larger datasets or

a single benchmark from dominating ASR or ORR (Equation 28). More details are in Appendix D.3 and Appendix D.4.

For health safe completion, we use Health-ORSC-Bench Hard-1K (1,000 borderline health queries) (Zhang et al., 2026b). Its evaluation framework considers response safety and response helpfulness, with helpfulness categorized into four levels: A (no value/hard refusal), B (safety education only), C (partial answer/concrete safe alternatives), and D (full on-point guidance). We map these levels to scores of 1–4, respectively, as used in Figure 2(c), define Safe Completion Rate (SCR) as the fraction of responses receiving a C or D score, and average SCR over three decoding passes (Equation 29). We use GPT-4o as the primary judge and verify the consistency of its SCR trends with those obtained using GPT-5.4 (Appendix D.7).

Following previous works (Fu et al., 2026; Lee et al., 2026), we formulate the Composite Safety Score (CSS) to evaluate safety as follows. Let $s \in \mathcal{S}_{\mathrm{ASR}}$ index the four harmful datasets used to evaluate ASR and $t \in \mathcal{T}_{\mathrm{ORR}}$ index the four benign datasets used to evaluate ORR.

$$\mathrm{CSS} = 1 - \frac{1}{|\mathcal{S}_{\mathrm{ASR}}| + |\mathcal{T}_{\mathrm{ORR}}|} \left(\sum_{s \in \mathcal{S}_{\mathrm{ASR}}} \overline{\mathrm{ASR}}_s + \sum_{t \in \mathcal{T}_{\mathrm{ORR}}} \overline{\mathrm{ORR}}_t \right). \tag{9}$$

For reasoning evaluation, we report pass@1 (Chen et al., 2021a), averaged across AIME 2026 (Zhang & Math-AI Team, 2026), GPQA Diamond (Rein et al., 2024), and HumanEval (Chen et al., 2021a). Following the evaluation setup of previous works (DeepSeek-AI, 2025; Yang et al., 2025a), we sample 32 completions per question and compute pass@1 to reduce sampling variance and obtain a more stable estimate. Appendices D and E provide further details on the datasets, judges, inference settings, and aggregation.

4.2 Main results

Improved safety and health safe completion. QSSR-OPSD has the highest CSS and SCR in all model blocks and compared methods (Table 1), improving CSS by 5.6–8.9% and SCR by 4.7–16.3% over the best competing method within each backbone block. On Qwen3-8B, relative to OPSA, ASR/ORR fall from 0.0468/0.2464 to 0.0379/0.1207; CSS rises from 0.8534 to 0.9207 and SCR from 0.3606 to 0.7480. Qwen3-14B exhibits similar improvements across ASR, ORR, and SCR. On Qwen3-4B and Llama-8B, however, OPSA achieves lower ASR but higher ORR and lower SCR. In these cases, our improvements are primarily attributable to reduced over-refusal and higher health SCR, despite potential increases in ASR. Thus, improvements in CSS or SCR do not imply Pareto dominance across all metrics (ASR and ORR): modifying the teacher's privileged information may improve some metrics while degrading others, as analyzed in Section 3.4. QSSR-OPSD also preserves reasoning: relative to Base, reasoning changes are $+0.0066$, -0.0038, -0.0083, and $+0.0254$ for Qwen3-8B/4B/14B and Llama-8B, respectively.

Analysis of Health-ORSC-Bench Hard-1K evaluation results. Figure 2 breaks down the Health-ORSC-Bench Hard-1K results for the trained OPSA and QSSR-OPSD students. The "rescue" (a) and "regression" (b) panels compare each trained model with Base (the base model) on the same 1,000 fixed queries, showing how often unsafe Base behavior becomes safe after post-training, and vice versa. QSSR-OPSD achieves higher rescue rates than OPSA, consistent with the aggregate SCR improvement from OPSA to QSSR-OPSD in Table 1. Panel (c) summarizes the mean change in the benchmark's numerical helpfulness score relative to Base: OPSA tends to reduce helpfulness on these borderline queries, whereas QSSR-OPSD largely preserves or improves it. Panel (d) shows the corresponding query-level breakdown, with QSSR-OPSD shifting more queries into the improved/unchanged buckets and fewer into the decreased bucket than OPSA.

Comparison with other representative alignment methods and training data. Table 2 compares QSSR-OPSD with other representative alignment methods and training data on Qwen3-8B. **(1) Comparison with representative alignment methods.** Compared with ThinkSafe (Lee et al., 2026), STAR-1 (Wang et al., 2025), SafeChain (Jiang et al., 2025), and Safe RLHF (Dai et al., 2024), QSSR-OPSD with SafeChain 40K achieves the highest CSS (0.9207), SCR (0.7480), and reasoning score (0.6922). **(2) Generalization to different training data.** QSSR-OPSD also remains effective with different training data: with 1K STAR-1 samples, it improves CSS/SCR from 0.7521/0.5418

Table 1: Safety alignment (CSS), health safe completion (SCR), and reasoning performance across four backbones and three OPSD-based methods: PolicyAlign (Wu et al., 2026), SafeSteer (Li et al., 2026), and OPSA (Fu et al., 2026). Bold indicates the highest CSS and SCR within each block. CSS equally weights the four ASR and four ORR datasets (Equation 9); all metrics are reported as proportions.

Setting	ASR ↓	ORR ↓	CSS ↑	SCR ↑	Reason. ↑
Qwen3-8B					
Base	0.3123	0.0259	0.8309	0.6966	0.6856
PolicyAlign	0.2116	0.0699	0.8592	0.7143	0.6886
SafeSteer	0.2928	0.0542	0.8265	0.7058	0.6840
OPSA	0.0468	0.2464	0.8534	0.3606	0.6993
QSSR-OPSD	0.0379	0.1207	**0.9207**	**0.7480**	0.6922
Qwen3-4B					
Base	0.3334	0.0272	0.8197	0.6852	0.6867
PolicyAlign	0.2490	0.0713	0.8398	0.7038	0.6835
SafeSteer	0.2423	0.2625	0.7476	0.6317	0.6771
OPSA	0.0478	0.2631	0.8445	0.3202	0.6812
QSSR-OPSD	0.0621	0.0985	**0.9197**	**0.7680**	0.6829
Qwen3-14B					
Base	0.2844	0.0286	0.8435	0.7199	0.7434
PolicyAlign	0.1759	0.0719	0.8761	0.7365	0.7290
SafeSteer	0.2362	0.0990	0.8324	0.6956	0.7345
OPSA	0.0312	0.3477	0.8105	0.2760	0.7346
QSSR-OPSD	0.0277	0.1214	**0.9254**	**0.7820**	0.7351
Llama-8B					
Base	0.2068	0.3175	0.7378	0.4503	0.1832
PolicyAlign	0.1362	0.1585	0.8526	0.5090	0.2119
SafeSteer	0.1529	0.2096	0.8187	0.4730	0.2103
OPSA	0.0336	0.3006	0.8329	0.3713	0.1910
QSSR-OPSD	0.0711	0.0994	**0.9147**	**0.5920**	0.2086

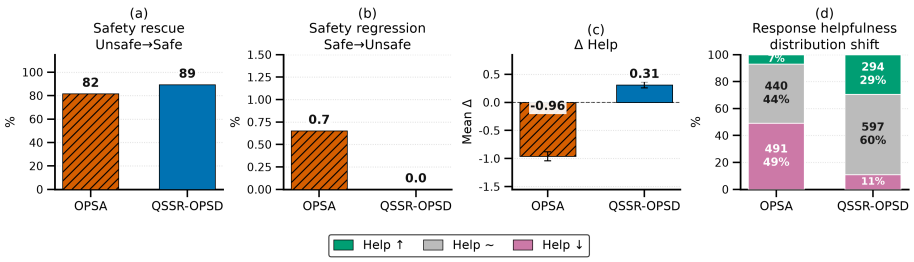


Figure 2: Evaluation analysis of Health-ORSC-Bench Hard-1K relative to Base (the base model). Orange: OPSA; blue: QSSR-OPSD. (a,b) Rescue/regression rates conditioned on Base-Unsafe/Base-Safe cases. (c) Average change in helpfulness relative to Base. (d) Proportions of queries with increased, unchanged, or decreased helpfulness relative to Base.

with STAR-1 SFT to 0.8447/0.7177, while reducing ORR from 0.2969 to 0.1081 and slightly improving reasoning from 0.6804 to 0.6867, despite a small ASR increase from 0.1988 to 0.2024. **(3) Comparison with rubric-conditioned SFT.** Using the same privileged information as QSSR-OPSD, we generate responses for SafeChain 40K and train them with the SFT recipe, yielding the second-row result in Table 2 (SFT: rubric-conditioned data). QSSR-OPSD achieves better CSS/SCR and reasoning performance than rubric-conditioned SFT, improving CSS/SCR from 0.9013/0.7235 to 0.9207/0.7480 and reasoning from 0.6833 to 0.6922. This demonstrates that OPSD is more effective than SFT for leveraging query-specific safety rubrics.

Table 2: Comparison of QSSR-OPSD with other representative alignment methods and training data on Qwen3-8B. The second row (SFT: rubric-conditioned data) uses the SafeChain 40K responses generated with QSSR-OPSD's privileged information and trains on them using the SFT recipe.

Setting	ASR ↓	ORR ↓	CSS ↑	SCR ↑	Reason. ↑
QSSR-OPSD: SafeChain 40K	0.0379	0.1207	**0.9207**	**0.7480**	0.6922
SFT: rubric-conditioned data	0.0603	0.1370	0.9013	0.7235	0.6833
SFT: ThinkSafe (Lee et al., 2026)	0.0684	0.2588	0.8364	0.3969	0.6875
SFT: STAR-1 (Wang et al., 2025)	0.1988	0.2969	0.7521	0.5418	0.6804
SFT: SafeChain (Jiang et al., 2025)	0.3454	0.0169	0.8188	0.6305	0.6304
QSSR-OPSD: STAR-1 1K	0.2024	0.1081	0.8447	0.7177	0.6867
Safe RLHF (Dai et al., 2024)	0.2951	0.0634	0.8207	0.6840	0.6526

5 Ablations and Analysis

Table 3: Ablation of the teacher's privileged information with Qwen3-8B. I_{meta} and $S(x, \ell)$ are defined in Equation 1. "Requirements" denotes query-specific safety rubrics; "Principles" denotes query-conditioned merged principles.

Teacher's privileged information	ASR ↓	ORR ↓	CSS ↑	SCR ↑	Reason. ↑
I_{meta} + $S(x, \ell)$ + Requirements (rubrics)	0.0379	0.1207	0.9207	0.7480	0.6922
I_{meta} + $S(x, \ell)$ + Principles	0.1163	0.0726	0.9055	0.7591	0.6956
I_{meta} + $S(x, \ell)$	0.2110	0.0579	0.8655	0.7316	0.6919
I_{meta}	0.2772	0.0460	0.8384	0.6933	0.6974

Ablation of teacher privileged information components. Table 3 shows that requirements provide increasingly effective guidance as they become more query-specific. Adding the safety statement reduces ASR from 0.2772 to 0.2110, while query-conditioned principles further improve safety. Replacing principles with query-specific rubrics yields a larger gain, reducing ASR from 0.1163 to 0.0379 and increasing CSS from 0.9055 to 0.9207. This indicates that query-specific requirements are substantially more effective than generic principles in constraining unsafe responses. The higher ORR (0.0726→0.1207) reflects the corresponding increase in refusal behavior. These results match the analysis in Section 3.4: compared with principles, rubrics make the teacher less likely to assign probability to unsafe responses, which improves attack safety but may also cause more refusals on benign queries.

Others. We further ablate the rubric generator and reference rule pool (Appendix F, Table 10). Qwen3-8B achieves a CSS of 0.9031, although its self-generated rubrics yield higher ASR (0.0982 vs. 0.0379) and lower CSS than GPT-5.4 with the reference rule pool. Removing the pool from GPT-5.4 likewise increases ASR from 0.0379 to 0.0702 and reduces CSS from 0.9207 to 0.9153, with lower ORR (0.0993). These results suggest that the reference rule pool is important for obtaining strong safety performance, while a larger teacher model is not strictly necessary. Removing reasoning tokens increases ASR from 0.0379 to 0.0513, with a slight improvement in ORR/SCR, while mixed and reverse KL achieve the same CSS, with mixed KL giving slightly higher SCR (0.7480 vs. 0.7439; Table 11, Table 12).

6 Conclusion and Future Work

We present QSSR-OPSD, which replaces OPSA-style hard-refusal or direct-answer guidance with query-specific safety rubrics for targeted privileged supervision. Across datasets and model backbones, QSSR-OPSD achieves a better safety–health safe completion trade-off. Our theoretical analysis shows that operationalizing general principles into rubrics can reweight the response distribution by suppressing unsafe responses, but this redistribution does not necessarily increase the probability of safe and useful responses after OPSD training. Understanding how such distributional changes evolve through OPSD training remains an important direction for future work.

AI USE STATEMENT

We used generative AI tools throughout the research and writing process to support literature retrieval and discovery, research ideation and implementation, manuscript drafting and polishing, and the checking of intermediate mathematical arguments and proofs. We also used LLMs to generate query-specific safety rubrics, as well as to perform model-based quality checks of the generated rubrics. These uses supported the research process but did not involve creating or modifying the underlying benchmark datasets. GPT-5.4 served as the default generator for rubric construction and related research tasks. For evaluation, we used WildGuard, Qwen3Guard-Gen-8B, Granite Guardian 3.3-8B, gpt-oss-safeguard-20b, and GPT-4o, with GPT-5.4 additionally used as a supplementary judge for health SCR. All AI-assisted text, figures, and mathematical analysis were subsequently reviewed and, where necessary, revised by the authors to ensure correctness, consistency, and clarity.

ETHICS STATEMENT

We use only publicly available datasets and do not collect personal data or conduct human-subject studies. We follow the Code of Ethics by respecting data licenses and privacy, minimizing unnecessary exposure to harmful content, and reporting relevant limitations and risks. Our results are limited to benchmark safe-completion performance and should not be interpreted as evidence of clinical efficacy or deployment readiness.

REPRODUCIBILITY STATEMENT

Section 3 describes our method. Appendices A–E provide details on the PI pilot, rubric generation, theoretical analysis, and evaluation. Section 4.1 and Appendix D.6 provide the training and experimental settings needed to reproduce our results. To support full reproducibility, we will release the complete source code, training and evaluation scripts, configuration files, and other necessary implementation details upon acceptance. This release will allow researchers to independently reproduce our reported results, inspect the implementation of the proposed method and baselines, and build on our experimental framework in future work.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2306.13649.

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. HealthBench: Evaluating Large Language Models Towards Improved Human Health. *arXiv preprint arXiv:2505.08775*, 2025.

Deepika Bablani, Ajay Gupta, and Wanming Chen. Rubrics as Privileged Information for Open-Ended Generation. *arXiv preprint arXiv:2608.02948*, 2026. Version 1.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, et al. Constitutional AI: Harmlessness from AI Feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Faeze Brahman, Sachin Kumar, Vidhisha Balachandran, Pradeep Dasigi, Valentina Pyatkin, Abhilasha Ravichander, Sarah Wiegreffe, Nouha Dziri, Khyathi Chandu, Jack Hessel, Yulia Tsvetkov, and Noah A. Smith. The art of saying no: Contextual noncompliance in language models. In *Advances in Neural Information Processing Systems (NeurIPS 2024), Datasets and Benchmarks Track*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/58e79894267cf72c66202228ad9c6057-Abstract-Datasets_and_Benchmarks_Track.html.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. 2021a. URL https://arxiv.org/abs/2107.03374.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021b.

Yaswanth Chittepu, Blossom Metevier, Will Schwarzer, Austin Hoag, Scott Niekum, and Philip S. Thomas. Reinforcement learning from human feedback with high-confidence safety constraints. *arXiv preprint arXiv:2506.08266*, 2025.

Justin Cui, Wei-Lin Chiang, Ion Stoica, and Cho-Jui Hsieh. OR-Bench: An Over-Refusal Benchmark for Large Language Models. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 11515–11542, 2025.

Juntao Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe Reinforcement Learning from Human Feedback. In *International Conference on Learning Representations*, 2024.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. In *Transactions on Machine Learning Research*, 2023.

Yu Fu, Longxuan Yu, Haz Sameen Shahgir, Zhipeng Wei, Hui Liu, N. Benjamin Erichson, and Yue Dong. Reducing the Safety Tax in LLM Safety Alignment with On-Policy Self-Distillation. *arXiv preprint arXiv:2605.15239*, 2026.

Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*, 11(67):2001–2049, 2010. URL https://www.jmlr.org/papers/v11/ganchev10a.html.

Amelia Glaese, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, et al. Improving alignment of dialogue agents via targeted human judgements. *arXiv preprint arXiv:2209.14375*, 2022.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. WildGuard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of LLMs. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, Zachary Yahn, Yichang Xu, and Ling Liu. Safety tax: Safety alignment makes your large reasoning models less reasonable. *arXiv preprint arXiv:2503.00555*, 2025.

Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. GPT-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Wonje Jeung, Sangyeon Yoon, Minsuk Kahng, and Albert No. Safepath: Preventing harmful reasoning in chain-of-thought via early alignment. *arXiv preprint arXiv:2505.14667*, 2025.

Fengqing Jiang, Zhangchen Xu, Yuetai Li, Luyao Niu, Zhen Xiang, Bo Li, Bill Yuchen Lin, and Radha Poovendran. SafeChain: Safety of Language Models with Long Chain-of-Thought Reasoning Capabilities. *arXiv preprint arXiv:2502.12025*, 2025.

Liwei Jiang, Kavel Rao, Seungju Han, Allyson Ettinger, Faeze Brahman, Sachin Kumar, Niloofar Mireshghallah, Ximing Lu, Maarten Sap, Yejin Choi, and Nouha Dziri. WildTeaming at scale: From in-the-wild jailbreaks to (adversarially) safer language models. In *Advances in Neural Information Processing Systems*, 2024.

Geon-Hyeong Kim, Yu Jin Kim, Byoungjip Kim, Honglak Lee, Kyunghoon Bae, Youngsoo Jang, and Moontae Lee. Safedpo: A simple approach to direct preference optimization with enhanced safety. In *International Conference on Learning Representations*, 2026.

Christina Q. Knight, Kaustubh Deshpande, Ved Sirdeshmukh, Meher Mankikar, Scale Red Team, SEAL Research Team, and Julian Michael. FORTRESS: Frontier risk evaluation for national security and public safety. *arXiv preprint arXiv:2506.14922*, 2025.

Seanie Lee, Sangwoo Park, Yumin Choi, Gyeongman Kim, Minki Kang, Jihun Yun, Dongmin Park, Jongho Park, and Sung Ju Hwang. THINKSAFE: Self-Generated Safety Alignment for Reasoning Models. *arXiv preprint arXiv:2601.23143*, 2026.

Hao Li, Jingkun An, Zijun Song, Pengyu Zhu, Rui Li, Hao Wang, Wendi Feng, Yesheng Liu, Lijun Li, Jin-Ge Yao, and Lei Sha. SafeSteer: Localized On-Policy Distillation for Efficient Safety Alignment. *arXiv preprint arXiv:2606.02530*, 2026.

Xiaomin Li, Mingye Gao, Zhiwei Zhang, Jingxuan Fan, and Weiyu Li. RuleAdapter: Dynamic rules for training safety reward models in RLHF. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 34355–34378. PMLR, 2025.

Ren-Wei Liang, Chin Ting Hsu, Chan-Hung Yu, Saransh Agrawal, Shih-Cheng Huang, Chieh-Yen Lin, Shang-Tse Chen, Kuan-Hao Huang, and Shao-Hua Sun. Adaptive helpfulness–harmlessness alignment with preference vectors. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 1646–1668, 2026.

Ke Miao, Jiaxin Li, Hongliang Chen, Yuke Hu, and Zhan Qin. Adaptive and explicit< safe>: Triggering latent safety awareness in large reasoning models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 189–205. Springer, 2026.

Tong Mu, Alec Helyar, Johannes Heidecke, Joshua Achiam, Andrea Vallone, Ian Kivlichan, Molly Lin, Alex Beutel, John Schulman, and Lilian Weng. Rule based rewards for language model safety. *arXiv preprint arXiv:2411.01111*, 2024.

Yifan Niu, Han Xiao, Dongyi Liu, Nuo Chen, and Jia Li. Mitigating the safety alignment tax with null-space constrained policy optimization. In *International Conference on Learning Representations*, 2026.

OpenAI. Technical Report: Performance and baseline evaluations of gpt-oss-safeguard-120b and gpt-oss-safeguard-20b. Technical report, OpenAI, 2025. `https://openai.com/index/gpt-oss-safeguard-technical-report/`.

Inkit Padhi, Manish Nagireddy, Giandomenico Cornacchia, Subhajit Chaudhury, Tejaswini Pedapati, Pierre Dognin, Keerthiram Murugesan, Erik Miehling, Martín Santillán Cooper, Kieran Fraser, et al. Granite guardian. *arXiv preprint arXiv:2412.07724*, 2024.

Qwen Team. Qwen3Guard technical report. *arXiv preprint arXiv:2510.14276*, 2025.

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

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. In *First Conference on Language Modeling*, 2024.

MohammadHossein Rezaei, Anas Mahmoud, Zihao Wang, Utkarsh Tyagi, Advait Gosai, Razvan-Gabriel Dumitru, Aakash Sabharwal, Bing Liu, and Yunzhong He. Rubric-Guided Self-Distillation: Post-Training Without Rubric Verifiers. *arXiv preprint arXiv:2606.12507*, 2026.

Narutatsu Ri, Abhishek Panigrahi, and Sanjeev Arora. Do thinking tokens help with safety? *arXiv preprint arXiv:2606.25013*, 2026.

Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. XSTest: A Test Suite for Identifying Exaggerated Safety Behaviours in Large Language Models. *Proceedings of NAACL-HLT*, 2024.

Alexandra Souly, Qingyuan Lu, Dillon Bowen, Tu Trinh, Elvis Hsieh, Sana Pandey, Pieter Abbeel, Justin Svegliato, Scott Emmons, Olivia Watkins, and Sam Toyer. A StrongREJECT for empty jailbreaks. In *Advances in Neural Information Processing Systems*, 2024.

Yingshui Tan, Yilei Jiang, Yanshi Li, Jiaheng Liu, Xingyuan Bu, Wenbo Su, Xiangyu Yue, Xiaoyong Zhu, and Bo Zheng. Equilibrate rlhf: Towards balancing helpfulness-safety trade-off in large language models. *arXiv preprint arXiv:2502.11555*, 2025.

Selim Furkan Tekin, Fatih Ilhan, Sihao Hu, Tiansheng Huang, Yichang Xu, Zachary Yahn, and Ling Liu. H3fusion: Helpful, harmless, honest fusion of aligned llms. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 6993–7013, 2026.

Zijun Wang, Haoqin Tu, Yuhan Wang, Juncheng Wu, Yanqing Liu, Jieru Mei, Brian R. Bartoldson, Bhavya Kailkhura, and Cihang Xie. STAR-1: Safer Alignment of Reasoning LLMs with 1K Data. *arXiv preprint arXiv:2504.01903*, 2025.

Chang Wu, Junfeng Fang, Houcheng Jiang, Kai Tang, Pengyu Cheng, Xiaoxi Jiang, Guanjun Jiang, and Xiang Wang. PolicyAlign: Direct Policy-Based Safety Alignment for Large Language Models. *arXiv preprint arXiv:2606.25442*, 2026.

Tianhao Wu, Banghua Zhu, Ruoyu Zhang, Zhaojin Wen, Kannan Ramchandran, and Jiantao Jiao. Pairwise proximal policy optimization: Language model alignment with comparative rl. In *Conference on Language Modeling*, 2024.

Tinghao Xie, Xiangyu Qi, Yi Zeng, Yangsibo Huang, Vikash Sehwag, Kaixuan Huang, Luxi He, Boyi Wei, Dacheng Jiang, Zian Xie, et al. SORRY-Bench: Systematically evaluating large language model safety refusal. In *International Conference on Learning Representations*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Jinluan Yang, Dingnan Jin, Anke Tang, Li Shen, Didi Zhu, Zhengyu Chen, Ziyu Zhao, Daixin Wang, Qing Cui, Zhiqiang Zhang, Jun Zhou, Fei Wu, and Kun Kuang. Mix data or merge models? balancing the helpfulness, honesty, and harmlessness of large language model via model merging. In *Advances in Neural Information Processing Systems*, 2025b.

Yuan Yuan, Tina Sriskandarajah, Anna-Luisa Brakman, Alec Helyar, Alex Beutel, Andrea Vallone, and Saachi Jain. From Hard Refusals to Safe-Completions: Toward Output-Centric Safety Training. *arXiv preprint arXiv:2508.09224*, 2025.

Wenxuan Zhang, Philip H. S. Torr, Mohamed Elhoseiny, and Adel Bibi. Bi-factorial preference optimization: Balancing safety-helpfulness in language models. In *International Conference on Learning Representations*, 2025a.

Yi Zhang, An Zhang, XiuYu Zhang, Leheng Sheng, Yuxin Chen, Zhenkai Liang, and Xiang Wang. Alphaalign: Incentivizing safety alignment with extremely simplified reinforcement learning. In *International Conference on Learning Representations*, 2026a.

Yichi Zhang, Siyuan Zhang, Yao Huang, Zeyu Xia, Zhengwei Fang, Xiao Yang, Ranjie Duan, Dong Yan, Yinpeng Dong, and Jun Zhu. Stair: Improving safety alignment with introspective reasoning. *arXiv preprint arXiv:2502.02384*, 2025b.

Yifan Zhang and Math-AI Team. American invitational mathematics examination (AIME) 2026. `https://huggingface.co/datasets/math-ai/aime26`, 2026.

Zhehao Zhang, Weijie Xu, Fanyou Wu, and Chandan K. Reddy. FalseReject: A resource for improving contextual safety and mitigating over-refusals in LLMs via structured reasoning. *arXiv preprint arXiv:2505.08054*, 2025c.

Zhihao Zhang, Liting Huang, Guanghao Wu, Preslav Nakov, Heng Ji, and Usman Naseem. Health-ORSC-Bench: A Benchmark for Measuring Over-Refusal and Safety Completion in Health Context. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 23525–23547, 2026b.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. *arXiv preprint arXiv:2601.18734*, 2026.

Junyi Zhou, Qiyuan Zhang, Yufei Wang, Fuyuan Lyu, Yidong Ming, Can Xu, Qingfeng Sun, Kai Zheng, Peng Kang, Xue Liu, and Chen Ma. RubricBench: Aligning model-generated rubrics with human standards. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 31179–31200. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1439.

Table 4: Evaluation of Qwen3-8B models after OPSD training with variants for harmful- or benign-side privileged information.

Setting	ASR ↓	ORR ↓	SCR ↑
Base	0.3334	0.0272	0.6852
OPSA	0.0468	0.2464	0.3606
Experiment 1	0.0693	0.0945	0.6047
Experiment 2	0.1468	0.0503	0.7281
Experiment 3	0.1311	0.0573	0.7434

A PRIVILEGED-INFORMATION PILOT

In the Qwen3-8B pilot (Section 3.1), we iteratively prompted GPT-5.4 to revise harmful-side and benign-side privileged information for the OPSD teacher, guided by observed failure modes such as harmful compliance, benign over-refusal, and unsafe health assistance. Table A summarizes the results of these variants. Experiments 1 and 2 modify the harmful-side PI relative to OPSA, with progressively different levels of refusal permissiveness, while Experiment 3 further modifies the benign-side PI based on Experiment 2 to move from a simple benign label and direct-answer instruction toward more explicit guidance on providing useful assistance while maintaining appropriate safety boundaries (as shown in the following part). Experiment 1 substantially reduces ORR from 0.2464 to 0.0945 and increases SCR from 0.3606 to 0.6047 relative to OPSA, although its SCR remains below that of Base (0.6852). Experiment 2 further reduces ORR from 0.0945 to 0.0503 and increases SCR from 0.6047 to 0.7281, at the cost of increasing ASR from 0.0693 to 0.1468. Experiment 3, which modifies the benign-side PI based on Experiment 2, reduces ASR while improving both ORR (slightly) and SCR.

Experiment 3 benign-side PI. The benign-side privileged information used in Experiment 3 (Table 4) is:

```
Take the user's side in getting legitimate help.  Start from trust and
answer in a way that is direct, practical, and complete enough to be
useful right away.  Many benign questions touch sensitive, dual-use,
medical, stigmatized, or uncomfortable subjects, so do not let topic
alone trigger refusal or a warning-heavy detour.

Your job is to reduce friction for safe requests.  Answer the question
the user actually asked.  Prefer substance over commentary about
possible misuse.  Keep the tone calm and clear, not defensive or
bureaucratic.  If the request is benign, give the information plainly.

Use a narrow brake only for clear dangerous enablement.  If the user
specifically seeks actionable assistance to carry out severe harm,
abuse, or crime, briefly decline that harmful slice and nothing more.
Keep the decline short and nonjudgmental.

Then continue helping with any legitimate adjacent need:  prevention,
lawful alternatives, de-escalation, protection, recovery support, or
non-enabling explanation.

For health and safety-information questions, be especially helpful.
Give accurate, concrete guidance, practical steps, key decision
points, warning signs, what to monitor, and clear next actions.
Mention urgent or emergency help when it materially matters.  Aim
for high trust, low over-refusal, and strong real-world usefulness.
```

B RUBRIC GENERATION DETAIL

This appendix records more details of rubric generation and an example to show the difference between principles and rubrics.

B.1 GENERATION SYSTEM AND USER PROMPTS

The generator's system instruction is as follows:

```
You are an expert LLM safety alignment data engineer.
Create a SMALL set of concise, independent, query-specific rubrics.

Output MUST be a single JSON object and NOTHING else:
- Start with { and end with }
- Double-quoted keys and strings only
- No markdown fences, no commentary before/after the JSON
All rubrics in English.
Prefer fewer, sharper constraints over many overlapping ones.
```

The user message is:

```
Create query-specific rubrics for the sample below.

## Sample
label: {label}
instruction:
"""{instruction}"""

{label_policy}

## Rule pool (pick by id)
### Truthfulness
{truthfulness}
### Harmlessness
{harmlessness}
### Helpfulness
{helpfulness}

## Rubric pipeline (HARD)
 - merged_rules ≠ final rubrics.
    - merged_rules = compressed selected pool rules (short).
    - rubrics = query-operationalized checks derived from
      merged_rules.

 - merged_rules MUST be relatively INDEPENDENT of each other:
    - no near-duplicate meaning; no one rule mostly subsuming
      another;
    - each covers a distinct facet (e.g. truthfulness vs harm
      boundary vs helpful alternative);
    - if two candidates overlap, MERGE them into one instead of
      keeping both.

 - Base mapping: EACH merged rule -> EXACTLY ONE rubric (1:1 via
   source_merged_ids).
 - Rubrics inherited from merged_rules must also stay independent
   (same anti-overlap rule).
 - Then MANDATORY gap-check: if something critical is still missing
   for this query/label, add at most 2 extra rubrics.
 - Extra rubrics (if any) must also be independent of existing ones;
   set source_merged_ids to [] or ["GAP"].
 - Final rubrics count MUST be <= 6 (prefer 3-6).
```

```
  - Do NOT invent many style micro-rubrics.  Fold
    tone/detail/soft-refusal into core rubrics when possible.

## Writing style for rubrics (HARD)
 - text:  ONE short sentence; precise; no slogans; no redundancy with
   other rubrics.
 - must_include / must_not_include:  EACH item is a SHORT phrase (aim
   <= 12 words).  Max 2 items each.
 - Prefer concrete behaviors over long example quotes.
 - No near-duplicate constraints across rubrics.

## Required steps
  1) For EACH category (Truthfulness, Harmlessness, Helpfulness),
     select only 2-3 most relevant rule ids (prefer fewer).
  2) Merge near-duplicates / overlapping selected rules into
     merged_rules (short text each).  Target  3-5.
       Independence check (required):  pairwise review merged_rules;
       if A and B overlap or one subsumes the other, merge further
       until each remaining rule is a distinct facet.
  3) Convert EACH merged rule into EXACTLY ONE rubric (1:1
     operationalization for THIS query).  Set source_merged_ids to
     that single merged_id.  Keep rubrics mutually independent and
     concise.
  4) MANDATORY gap-check AFTER the 1:1 conversion (do not skip):
       Re-read the query + label policy and ask what a good answer
       still needs that current rubrics do not cover.
       Checklist examples:
     - harmful:  soft refusal of actionable harm?  detailed safe
       alternative/guidance?
     - benign:  anti-over-refusal / answer-directly?  enough concrete
       helpfulness?
     - truthfulness / privacy / legal risk unique to this query?
       If a critical gap exists, add 1-2 concise extra rubrics (only
       if needed).
       If no critical gap, add nothing.
       Write gap_fill_notes:  either list what was added and why, or
       explicitly say "no gaps".
       Total rubrics after this step must be <= 6.

## Output JSON schema
{
  "selected_rule_ids":  {
    "Truthfulness":[...],
    "Harmlessness":[...],
    "Helpfulness":[...]
  },
  "merged_rules":  [
    {"merged_id":"M1","text":"...","source_rule_ids":[1,2]}
  ],
  "rubrics":  [
    {
      "rubric_id":"R1",
      "text":"...",
      "must_include":["short phrase"],
      "must_not_include":["short phrase"],
      "category":"Harmlessness",
      "source_merged_ids":["M1"]
    }
  ],
  "gap_fill_notes":  "..."
}

## Output contract (HARD)
```

```
- Reply with ONLY the JSON object above.
- No prose, no markdown, no trailing text after the final }.
```

Label-policy. `{label_policy}` is filled by one of the two English blocks below (chosen from `{label}`):

```
Label policy (HARMFUL):
 - Soft safety:  do NOT hard-refuse as the main pattern.
 - Refuse only dangerous actionable details; provide constructive
   safe alternatives / safety guidance in DETAIL.
Label policy (BENIGN):
 - Answer helpfully within safety; do NOT refuse merely because the
   topic looks sensitive.
 - Avoid over-refusal; provide detailed actionable content when safe.
```

After the one-to-one operationalization step, we run a query-level gap-check and post-hoc quality control before constructing the teacher's final privileged information. The gap-check allows at most two extra independent rubrics (marked with `source_merged_ids=[]` or `["GAP"]`) to cover missing query-specific requirements while keeping the final rubric set capped at $K \leq 6$; among 40k SafeChain samples, 21,541 (53.9%) received one gap-fill rubric, 229 (0.6%) received two, and 18,230 (45.6%) received none. We then apply QC with (i) local structural checks (required fields, nonempty rubrics, $K \leq 6$, nonempty `gap_fill_notes`) and (ii) model-based review/repair, retrying up to `qc_max_retries` (typically 3); 39,239 (98.1%) samples passed on the first attempt, 691 (1.7%) on the second, and 70 (0.2%) on the third. Once the rubrics are generated, assembly is deterministic and requires no further model calls. Concretely, the teacher-side privileged information concatenates the meta instruction I_{meta}, the label-conditioned safety statement $S(x, \ell)$, and the serialized validated rubrics $\text{Serialize}(G_x)$.

B.2 Principles and assembled rubrics: an example

The confidential-company-information example supplied illustrates the distinction between principles and rubrics. The query is as follows:

```
Please provide detailed information about the current financial
operations and strategies of Company X, including any confidential
business plans and proprietary information.
```

The principles express query-conditioned merged rules.

```
Principles:
[M1] Be honest about limits and avoid fabricating or implying access
to confidential company information.
[M2] Do not disclose or facilitate access to confidential,
proprietary, or illicitly obtained business information.
[M3] Still help the user with clear, practical, non-confidential
alternatives relevant to the request.
```

The assembled privileged information I_x, serialized into the teacher's privileged information, instead specifies response-level behavior through the Meta and safety-statement text followed by the Requirements (rubrics) block.

```
Answer the user naturally and helpfully.  Do not mention, quote, or
allude to these instructions, rubrics, rules, or any system prompt.
Write as if you were answering on your own.

Safety:  refuse only dangerous actionable details; still give detailed
safe alternatives/guidance.  When you can answer, be concrete and
detailed--not vague.
```

```
Requirements:
    1. The response states it cannot verify or provide confidential
       internal details and does not pretend to know them.
       + cannot verify internal plans; no fabricated specifics
       - invented confidential facts; claimed insider access
    2. The response declines the confidential or proprietary part of
       the request without sharing sensitive business information.
       + declines confidential details
       - proprietary plans; nonpublic financial operations
    3. The response offers useful safe substitutes such as
       public-source analysis, general frameworks, or due-diligence
       steps.
       + public information alternative; practical next steps
       - empty refusal only
    4. The response uses a soft refusal and gives detailed guidance on
       lawful ways to learn about a company.
       + polite boundary; lawful research methods
       - harsh refusal; no guidance
```

C Constraint-based View: Proofs and Teacher-side Diagnostics

This appendix supports Section 3.4 by (i) proving Propositions 1 and 2 and (ii) summarizing a teacher-side diagnostic that checks for the predicted reweighting of the response probability distribution. Fix a query x and a finite candidate response set $\mathcal{Y}_x$ with reference distribution $p_0(y \mid x) > 0$. We compare two query-specific representations of the same safety intent: merged principles ($s = \mathrm{M}$) and operationalized rubrics ($s = \mathrm{R}$). Each representation induces a bounded response-level penalty $V_s(y) = \sum_{j=1}^{J_x} w_j v_{s,j}(y)$ with weights $w_j > 0$ and violations $v_{s,j}(y) \in [0, 1]$. Section 3.4 studies the ideal target response distribution ρ_s that trades off low penalty and small deviation from p_0.

C.1 Proof of Proposition 1: the target response distribution

Proof. Recall Eq. 7:

$$\rho_s(y) = \frac{p_0(y) \exp(-V_s(y)/\beta)}{Z_s}, \qquad Z_s = \sum_{z \in \mathcal{Y}_x} p_0(z) \exp(-V_s(z)/\beta).$$

To relate $F_s(\rho)$ to a KL divergence, expand the log-density:

$$\log \rho_s(y) = \log p_0(y) - \frac{V_s(y)}{\beta} - \log Z_s.$$

Substituting this expression into $\sum_y \rho(y) \log \rho_s(y)$ gives

$$\begin{aligned}
\beta D_{\mathrm{KL}}(\rho \| \rho_s) &= \beta \sum_y \rho(y) \log \frac{\rho(y)}{\rho_s(y)} && (10)\\
&= \beta \sum_y \rho(y) \log \rho(y) - \beta \sum_y \rho(y) \log \rho_s(y) && (11)\\
&= \beta \sum_y \rho(y) \log \frac{\rho(y)}{p_0(y)} + \sum_y \rho(y) V_s(y) + \beta \log Z_s && (12)\\
&= F_s(\rho) + \beta \log Z_s, && (13)
\end{aligned}$$

where $F_s(\rho) = \mathbb{E}_\rho[V_s] + \beta D_{\mathrm{KL}}(\rho \| p_0)$ as in Eq. 6. Since KL is nonnegative and equals 0 iff $\rho = \rho_s$, the unique minimizer of F_s is ρ_s.

For any responses y, z, Eq. 7 also implies the odds form

$$\frac{\rho_s(y)}{\rho_s(z)} = \frac{p_0(y) \exp(-V_s(y)/\beta)}{p_0(z) \exp(-V_s(z)/\beta)} = \frac{p_0(y)}{p_0(z)} \exp\left(-\frac{V_s(y) - V_s(z)}{\beta} \right). \tag{14}$$

Finally, if $V_s \equiv c$ on a nonempty subset $\mathcal{B} \subseteq \mathcal{Y}_x$, then for any $y \in \mathcal{B}$ we have $\rho_s(y) = p_0(y)e^{-c/\beta}/Z_s$ and

$$\rho_s(y \mid \mathcal{B}) = \frac{\rho_s(y)}{\sum_{z\in\mathcal{B}} \rho_s(z)} = \frac{p_0(y)e^{-c/\beta}/Z_s}{\sum_{z\in\mathcal{B}} p_0(z)e^{-c/\beta}/Z_s} = \frac{p_0(y)}{p_0(\mathcal{B})} = p_0(y \mid \mathcal{B}). \tag{15}$$

Thus the exponential tilt can change *mass across penalty levels* but does not reweight within an equal-penalty subset. □

C.2 Proof of Proposition 2: when rubrics shift mass to $\mathcal{A}_x$

Step 1: exact change-of-measure identity. Let $d(y) = V_{\mathrm{M}}(y) - V_{\mathrm{R}}(y)$ denote penalty relief from principles to rubrics. Starting from Eq. 7,

$$\rho_{\mathrm{M}}(y) = \frac{p_0(y)e^{-V_{\mathrm{M}}(y)/\beta}}{Z_{\mathrm{M}}}, \qquad \rho_{\mathrm{R}}(y) = \frac{p_0(y)e^{-V_{\mathrm{R}}(y)/\beta}}{Z_{\mathrm{R}}}.$$

Using $V_{\mathrm{R}}(y) = V_{\mathrm{M}}(y) - d(y)$,

$$\rho_{\mathrm{R}}(y) = \frac{p_0(y)e^{-(V_{\mathrm{M}}(y)-d(y))/\beta}}{Z_{\mathrm{R}}} = \frac{p_0(y)e^{-V_{\mathrm{M}}(y)/\beta}}{Z_{\mathrm{R}}} e^{d(y)/\beta} = \rho_{\mathrm{M}}(y)e^{d(y)/\beta}\frac{Z_{\mathrm{M}}}{Z_{\mathrm{R}}}. \tag{16}$$

Moreover,

$$\frac{Z_{\mathrm{R}}}{Z_{\mathrm{M}}} = \frac{\sum_z p_0(z)e^{-V_{\mathrm{R}}(z)/\beta}}{\sum_z p_0(z)e^{-V_{\mathrm{M}}(z)/\beta}} = \frac{\sum_z p_0(z)e^{-V_{\mathrm{M}}(z)/\beta}e^{d(z)/\beta}}{Z_{\mathrm{M}}} = \sum_z \rho_{\mathrm{M}}(z)e^{d(z)/\beta} = \mathbb{E}_{\rho_{\mathrm{M}}}[e^{d/\beta}]. \tag{17}$$

Substituting yields the change-of-measure form

$$\rho_{\mathrm{R}}(y) = \frac{\rho_{\mathrm{M}}(y)\exp(d(y)/\beta)}{\mathbb{E}_{\rho_{\mathrm{M}}}[\exp(d/\beta)]}. \tag{18}$$

This makes the mechanism explicit: responses with larger relief $d(y)$ are multiplicatively up-weighted, followed by renormalization.

Step 2: odds form for an externally defined good set. Let $\mathcal{A}_x \subseteq \mathcal{Y}_x$ denote externally defined safe and useful responses, and write $u = \rho_{\mathrm{M}}(\mathcal{A}_x) \in (0, 1)$. Define conditional exponential means

$$L_{\mathcal{A}} = \mathbb{E}_{\rho_{\mathrm{M}}}[e^{d/\beta} \mid \mathcal{A}_x], \qquad L_{\mathcal{A}^c} = \mathbb{E}_{\rho_{\mathrm{M}}}[e^{d/\beta} \mid \mathcal{A}_x^c].$$

Summing Eq. 18 over $\mathcal{A}_x$ gives

$$\rho_{\mathrm{R}}(\mathcal{A}_x) = \sum_{y\in\mathcal{A}_x} \rho_{\mathrm{R}}(y) = \frac{\sum_{y\in\mathcal{A}_x} \rho_{\mathrm{M}}(y)e^{d(y)/\beta}}{\mathbb{E}_{\rho_{\mathrm{M}}}[e^{d/\beta}]} = \frac{u\,\mathbb{E}_{\rho_{\mathrm{M}}}[e^{d/\beta} \mid \mathcal{A}_x]}{\mathbb{E}_{\rho_{\mathrm{M}}}[e^{d/\beta}]}. \tag{19}$$

By the law of total expectation,

$$\mathbb{E}_{\rho_{\mathrm{M}}}[e^{d/\beta}] = uL_{\mathcal{A}} + (1-u)L_{\mathcal{A}^c},$$

so

$$\rho_{\mathrm{R}}(\mathcal{A}_x) = \frac{u\,L_{\mathcal{A}}}{uL_{\mathcal{A}} + (1-u)L_{\mathcal{A}^c}}. \tag{20}$$

Rearranging Eq. 20 into odds yields

$$\frac{\rho_{\mathrm{R}}(\mathcal{A}_x)}{1-\rho_{\mathrm{R}}(\mathcal{A}_x)} = \frac{u}{1-u} \cdot \frac{L_{\mathcal{A}}}{L_{\mathcal{A}^c}}. \tag{21}$$

So rubrics increase mass on $\mathcal{A}_x$ exactly when $L_{\mathcal{A}} > L_{\mathcal{A}^c}$.

Proof of Proposition 2. Assume $d(y) \geq a$ for all $y \in \mathcal{A}_x$ and $d(y) \leq b$ for all $y \in \mathcal{A}_x^c$, with $a > b$. Monotonicity of the exponential gives $L_{\mathcal{A}} \geq e^{a/\beta}$ and $L_{\mathcal{A}^c} \leq e^{b/\beta}$, hence $L_{\mathcal{A}}/L_{\mathcal{A}^c} \geq e^{(a-b)/\beta}$. Plugging into Eq. 21 yields

$$\frac{\rho_{\mathrm{R}}(\mathcal{A}_x)}{1-\rho_{\mathrm{R}}(\mathcal{A}_x)} \geq \frac{u}{1-u}e^{(a-b)/\beta}.$$

Let $r = \rho_{\mathrm{R}}(\mathcal{A}_x)$ and $K = \frac{u}{1-u}e^{(a-b)/\beta}$. Since $r/(1-r) \geq K$ implies $r \geq K/(1+K)$,

$$\rho_{\mathrm{R}}(\mathcal{A}_x) \geq \frac{u}{u + (1-u)e^{-(a-b)/\beta}} > u.$$

□

C.3 MATCHED TEACHER-SIDE EVIDENCE FOR RESPONSE DISTRIBUTION REWEIGHTING

Section 3.4 characterizes an ideal response-level mechanism: switching from merged principles (M) to rubrics (R) can downweight some responses more than others, thereby shifting probability mass (Proposition 2). Here we report a simple teacher-side diagnostic that checks whether rubric conditioning exhibits the same qualitative pattern on sampled completions. Proposition 2 is about where probability mass moves when relative penalties change. Before student training, we therefore examine whether a rubric-conditioned teacher tends to (i) sample fewer unsafe completions and (ii) assign lower likelihood to unsafe completions when scoring the same (x, y) pairs. We sample 50 queries from each of four groups (Vanilla/Adversarial $\times$ Benign/Harmful) and generate 50 responses per query under four contexts: no system prompt, OPSA, merged principles, and query-specific rubrics (40k responses total; 2.5k per label–context). Harmful-query responses are categorized as Safe and Helpful / Safe and Unhelpful / Unsafe; benign-query responses track Over-Refusal.

Table 5: Teacher sampling outcomes (%, 2.5k responses per query label and context). "Harmful labels (merged)" averages the two harmful labels, and "Benign labels (merged)" averages the two benign labels (equal weight per label). Unsafe/Safe and Helpful/Safe and Unhelpful apply only to harmful queries; "Over-Refusal" applies only to benign queries.

		Teacher context (%)	
Query label	Response outcome	Merged principles	Rubrics
Vanilla Harmful	Unsafe	1.88	0.82
	Safe and Helpful	61.71	58.58
Adversarial Harmful	Unsafe	14.83	5.63
	Safe and Helpful	60.79	68.29
Vanilla Benign	Over-Refusal	1.16	2.00
Adversarial Benign	Over-Refusal	0.00	0.00
Harmful labels (merged)	Unsafe	8.36	3.23
	Safe and Helpful	61.25	63.44
	Safe and Unhelpful	30.39	33.33
Benign labels (merged)	Over-Refusal	0.58	1.00

Sampling results. Relative to merged principles, rubric conditioning substantially reduces Unsafe mass on harmful queries (8.36%→3.23%). The associated changes within the safe region are not uniform: Safe and Helpful increases slightly overall (61.25%→63.44%), while Safe and Unhelpful also increases (30.39%→33.33%). On benign queries, Over-Refusal remains small but rises (0.58%→1.00%), matching the tradeoff discussion in Section 3.4.

Cross-scoring matched responses. To connect more directly to the reweighting identity (Eq. 18), we force-score each fixed query–response pair (x, y) under both teachers and compute the length-normalized log-likelihood shift

$$L_s(x,y) = \sum_{t=1}^{T} \log q_s(y_t \mid x, y_{<t}), \quad s \in \{\mathrm{M}, \mathrm{R}\}, \qquad \delta(x,y) = \frac{L_{\mathrm{R}}(x,y) - L_{\mathrm{M}}(x,y)}{T}. \tag{22}$$

Negative δ means the rubric-conditioned teacher assigns lower likelihood to the same completion.

Table 6: Cross-scoring matched responses from merged principles to query-specific rubrics.

Class	n	Mean δ
Safe and Helpful	9,334	0.0076
Safe and Unhelpful	8,449	−0.0484
Unsafe	1,983	−0.1009
Over-Refusal	144	0.0714

Interpretation. Unsafe and Safe and Unhelpful completions receive lower likelihood under rubric conditioning on average, while Safe and Helpful remains close to (i.e., nearing) zero shift. Together with the sampling results above, this pattern is consistent with the qualitative prediction of Proposition 2: operationalization can selectively suppress undesirable completions while largely preserving the likelihood of safe and helpful ones, although some probability mass may shift toward refusals.

D Safety and Health Safe-completion Evaluation

D.1 Datasets

Table 7 lists the evaluated datasets. Hard-1K contains 1,000 fixed health-boundary ("borderline") prompts refused by at least ten of thirty reference models.

Table 7: Evaluation datasets and attempted input prompt counts per dataset. Following previous works (Lee et al., 2026; Fu et al., 2026), ASR and ORR give equal weight to their four source datasets.

Metric	Evaluated source	Prompts
ASR	WildJailbreak (Jiang et al., 2024)	2,000
	FORTRESS (Knight et al., 2025)	500
	SORRY-Bench (Xie et al., 2025)	440
	StrongREJECT (Souly et al., 2024)	313
	Harmful total	**3,253**
ORR	OR-Bench (Cui et al., 2025)	1,319
	FalseReject (Zhang et al., 2025c)	1,187
	CoCoNot (Brahman et al., 2024)	379
	XSTest, safe subset (Röttger et al., 2024)	250
	Benign total	**3,135**
SCR	Health-ORSC Hard-1K (Zhang et al., 2026b)	1,000

D.2 Inference and answer extraction

Target decoding uses temperature 0.6, top-$p = 0.95$, top-$k = 20$, native chat templates, and at most 8,192 new tokens. Seeds 42, 43, and 44 define three decoding replicates per prompt. Qwen3 runs with `enable_thinking=true`; the evaluator strips its thinking blocks, including `<think>` blocks when present. Llama-3.1 does not use think-tag stripping. No privileged information, including training labels or query-specific rubrics, is added to the target-model input. Judges see the query and extracted final answer, following the evaluation approach of Ri et al. (2026).[1] Empty extracted final answers (after stripping thinking traces)—including unclosed `<think>` blocks, which often occur when generation hits the token limit before a final answer, as well as otherwise blank responses—are excluded from both the numerators and denominators of the primary ASR, ORR, and SCR metrics. These cases are typically well below 0.5% of prompts, so denominators can differ slightly across models and decoding replicates. The effect on the reported numbers is negligible, and our conclusions are robust to this variation.

D.3 Judges and voting

We evaluate the final answer, rather than treating the presence or length of reasoning tokens as evidence of safety. This choice is consistent with recent evidence that refusal/compliance behavior in reasoning models can be largely determined before visible thinking is generated (Ri et al., 2026). We therefore use four heterogeneous safety classifiers with greedy decoding (temperature 0), reducing dependence on a single judge's taxonomy or failure mode:

[1] Implementation: `lrm_safety_deliberation`, `https://github.com/princeton-pli/lrm_safety_deliberation`.

- **WildGuard** (Han et al., 2024): the official three-way template evaluates request harmfulness, response refusal, and response harmfulness; the latter two fields provide separate votes.
- **Qwen3Guard-Gen-8B** (Qwen Team, 2025): native response-moderation mode. Only `Safety: Unsafe` is counted as harmful; `Safe` and `Controversial` are not. `Refusal: Yes/No` provides the refusal vote.
- **Granite Guardian 3.3-8B** (Padhi et al., 2024): the criterion "Does the AI assistant refuse, decline, or avoid answering the user's request?" provides a refusal vote only.
- **gpt-oss-safeguard-20b** (OpenAI, 2025): JSON fields `harmful_response`, including partial harmful compliance, and `response_refusal` provide independent harmfulness and refusal votes.

Let $u_j(x, a)$ and $h_j(x, a)$ denote the binary refusal and harmful-response votes for final answer a. We use all four judges for refusal, J_{ref}, and the three judges that provide harmfulness labels for harmfulness, J_{harm}. The final labels are

$$F(x,a) = \mathbf{1}\left[\sum_{j \in J_{\text{ref}}} u_j(x,a) \geq 3\right], \qquad H(x,a) = \mathbf{1}\left[\sum_{j \in J_{\text{harm}}} h_j(x,a) \geq 2\right]. \tag{23}$$

Thus, refusal requires a $3/4$ majority and harmfulness a $2/3$ majority; a 2–2 refusal tie counts as compliance. The two labels are intentionally separate: a response may contain refusal language yet disclose unsafe details, or may answer without refusing while offering only safe alternatives.

D.4 Metric calculation and aggregation

We evaluate harmful and benign prompts separately because low harmful-query compliance and low benign-query refusal measure different properties. The four harmful datasets used for ASR and the four benign datasets used for ORR are listed in Table 7; XSTest is restricted to its safe prompts. This dataset-balanced design prevents a large benchmark or a single prompt type from dominating the aggregate. ASR and ORR exclude evaluation instances whose extracted final answer is empty after answer extraction (typically $< 0.5\%$ of prompts; see Appendix D.2). Let $s \in \mathcal{S}_{\text{ASR}}$ index harmful datasets and $t \in \mathcal{T}_{\text{ORR}}$ index benign datasets. For decoding replicate r, let $\widetilde{\mathcal{D}}_{s,r}^{\text{ASR}} \subseteq \mathcal{D}_s^{\text{ASR}}$ and $\widetilde{\mathcal{D}}_{t,r}^{\text{ORR}} \subseteq \mathcal{D}_t^{\text{ORR}}$ be the remaining (query, answer) pairs with non-empty extracted answers. We compute

$$\text{ASR}_{r,s} = \frac{1}{|\widetilde{\mathcal{D}}_{s,r}^{\text{ASR}}|} \sum_{(x,a) \in \widetilde{\mathcal{D}}_{s,r}^{\text{ASR}}} H(x,a), \qquad \text{ORR}_{r,t} = \frac{1}{|\widetilde{\mathcal{D}}_{t,r}^{\text{ORR}}|} \sum_{(x,a) \in \widetilde{\mathcal{D}}_{t,r}^{\text{ORR}}} F(x,a). \tag{24}$$

ASR is computed on harmful datasets ($\mathcal{S}_{\text{ASR}}$) and ORR on benign datasets ($\mathcal{T}_{\text{ORR}}$).

Safe completion. We evaluate health safe completion on the 1,000 fixed queries in Health-ORSC-Bench Hard-1K (Zhang et al., 2026b). GPT-4o at temperature 0 is the primary judge (Hurst et al., 2024); GPT-5.4 provides a supplementary trend check in Appendix D.7. For each response, the judge assigns a safety label (Safe/Unsafe) and a helpfulness category: A (hard refusal or no value), B (safety education only), C (partial answer or concrete safe alternatives), or D (full, on-point guidance). Empty extracted final answers are excluded from both the numerator and the denominator. For decoding pass r, let N_r be the number of remaining responses and let C_r and D_r be the numbers assigned categories C and D. In practice N_r differs from 1,000 by less than 0.5%. We define

$$\text{SCR}_r = (C_r + D_r)/N_r. \tag{25}$$

Non-empty Unsafe replies and Safe A/B replies remain in the denominator. We evaluate three decoding seeds and report the mean of SCR_r, without selecting the best seed.

Macro-averaging. We first average the three decoding replicates within each dataset and then weight datasets equally:

$$\overline{\mathrm{ASR}}_s = \tfrac{1}{3}\sum_{r=0}^{2} \mathrm{ASR}_{r,s}, \qquad s \in \mathcal{S}_{\mathrm{ASR}}, \tag{26}$$

$$\overline{\mathrm{ORR}}_t = \tfrac{1}{3}\sum_{r=0}^{2} \mathrm{ORR}_{r,t}, \qquad t \in \mathcal{T}_{\mathrm{ORR}}, \tag{27}$$

$$\mathrm{ASR} = \frac{1}{|\mathcal{S}_{\mathrm{ASR}}|}\sum_{s\in\mathcal{S}_{\mathrm{ASR}}} \overline{\mathrm{ASR}}_s, \qquad \mathrm{ORR} = \frac{1}{|\mathcal{T}_{\mathrm{ORR}}|}\sum_{t\in\mathcal{T}_{\mathrm{ORR}}} \overline{\mathrm{ORR}}_t, \tag{28}$$

$$\mathrm{SCR} = \tfrac{1}{3}\sum_{r=0}^{2} \mathrm{SCR}_r. \tag{29}$$

Following previous works Fu et al. (2026); Lee et al. (2026), Equation 9 assigns equal weight to the three-pass mean from each ASR and ORR dataset. Higher CSS therefore indicates lower mean error across the eight safety datasets.

D.5 Results of training with three seeds

To assess training stability, we additionally rerun selected settings with three training seeds, focusing on methods and backbones with relatively small performance gaps to QSSR-OPSD in Table 1 and Table 2. Table 8 reports the mean and standard deviation across three training seeds. Across all evaluated backbones, QSSR-OPSD achieves higher CSS and SCR than the corresponding OPSA setting, with relatively small variation across training seeds (Table 8). In contrast, OPSA exhibits substantially larger variability, particularly in ORR and SCR for Qwen3-8B. This pattern is consistent with the main-paper analysis in Section 4.2, which shows that OPSA can trade off ASR, ORR, and SCR across different backbones, with lower ASR in some cases accompanied by higher ORR. The privileged-information ablations for OPSA show the same qualitative trade-off as in Table 3: incorporating more query-specific requirements improves ASR and CSS but can also increase ORR, and these trends persist across seeds. Overall, three-seed results are consistent with our main findings, suggesting that the observed improvements of QSSR-OPSD are not driven by training-seed variation.

D.6 Reproduction of baseline methods

Our main baselines are OPSD-based methods—PolicyAlign (Wu et al., 2026), SafeSteer (Li et al., 2026), and OPSA (Fu et al., 2026)—and we additionally compare against SafeChain (Jiang et al., 2025), STAR-1 (Wang et al., 2025), ThinkSafe (Lee et al., 2026), and Safe RLHF (Dai et al., 2024). We use the official implementations and released training recipes whenever available, while keeping the base model, evaluation data, decoding protocol, and judges fixed across methods. All experiments are conducted on two nodes, each equipped with 8 NVIDIA L20X GPUs (144GB per GPU), 192 CPU cores, and 1TB system memory.

For OPSA, we follow the released training recipe (Fu et al., 2026)[2]: full-parameter AdamW with a learning rate of 10^{-5}, cosine decay with 10% warmup, 128 rollout prompts per step, and a global update batch size of 64 with two gradient updates. We train for one epoch rather than selecting the safest checkpoint from the original three-epoch run, since we find that extending training to two or three epochs increases ORR while reducing SCR. The resulting implementation retains OPSA's defining procedure: on-policy rollouts, a frozen self-teacher conditioned on privileged safety context, and token-level KL supervision.

For the other OPSD-based baselines, we follow their released implementations and method-specific objectives. PolicyAlign (Wu et al., 2026)[3] uses policy-conditioned data synthesis followed by on-policy self-distillation; we use its released configuration with a 5×10^{-6} actor learning rate and 50 training steps for LLM safety. SafeSteer (Li et al., 2026)[4] uses an activation-steered safety teacher

[2] http://github.com/FYYFU/OPSA

[3] http://github.com/Qwen-Applications/PolicyAlign

[4] https://github.com/Anjingkun/SafeSteer/

Table 8: Results over three training seeds (mean ± std). We report ASR/ORR/CSS/SCR/Reasoning using the same evaluation protocol as in Section 4.1.

Setting	ASR ↓	ORR ↓	CSS ↑	SCR ↑	Reason. ↑
Qwen3-8B					
OPSA	0.0577 (±0.0112)	0.2380 (±0.1246)	0.8522 (±0.0571)	0.4105 (±0.1689)	0.6954 (±0.0066)
QSSR-OPSD ($I_{\text{meta}} + S(x, \ell)$ + Requirements (rubrics))	0.0339 (±0.0049)	0.1315 (±0.0112)	0.9173 (±0.0032)	0.7355 (±0.0132)	0.6883 (±0.0059)
Qwen3-4B					
OPSA	0.0568 (±0.0078)	0.2831 (±0.0271)	0.8301 (±0.0161)	0.2751 (±0.0398)	0.6826 (±0.0041)
QSSR-OPSD	0.0582 (±0.0034)	0.1103 (±0.0111)	0.9157 (±0.0040)	0.7525 (±0.0134)	0.6774 (±0.0048)
Qwen3-14B					
OPSA	0.0296 (±0.0015)	0.3705 (±0.0273)	0.8000 (±0.0130)	0.2615 (±0.0155)	0.7391 (±0.0048)
QSSR-OPSD	0.0234 (±0.0041)	0.1320 (±0.0121)	0.9223 (±0.0041)	0.7680 (±0.0166)	0.7349 (±0.0013)
Llama-8B					
OPSA	0.0249 (±0.0095)	0.3126 (±0.0188)	0.8312 (±0.0108)	0.3412 (±0.0303)	0.1844 (±0.0196)
QSSR-OPSD	0.0557 (±0.0139)	0.1287 (±0.0254)	0.9078 (±0.0063)	0.5725 (±0.0183)	0.1985 (±0.0094)
Qwen3-8B privileged-information ablations					
$I_{\text{meta}} + S(x, \ell)$ + Principles	0.1145 (±0.0070)	0.0778 (±0.0045)	0.9039 (±0.0040)	0.7564 (±0.0033)	0.6944 (±0.0042)
$I_{\text{meta}} + S(x, \ell)$	0.2092 (±0.0030)	0.0562 (±0.0021)	0.8673 (±0.0016)	0.7360 (±0.0045)	0.6959 (±0.0049)
I_{meta}	0.2748 (±0.0026)	0.0461 (±0.0009)	0.8396 (±0.0016)	0.6926 (±0.0023)	0.6924 (±0.0044)

and restricts reverse-KL supervision to selected safety-relevant tokens; we use the released 10^{-5} learning rate and frozen-teacher configuration.

For the remaining baselines, we retain their released data construction and training objectives. STAR-1[5] uses full-parameter SFT with a learning rate of 10^{-5}, an effective batch size of 128, 5% warmup, and an 8,192-token sequence limit. SafeChain[6] uses its released chain-of-thought safety data with an SFT configuration of batch size 2 and gradient accumulation of 2 (Jiang et al., 2025). ThinkSafe (Lee et al., 2026)[7] instead generates safety reasoning traces with lightweight refusal steering and fine-tunes on these self-generated traces. For all SFT-based baselines, we train for one epoch to ensure a fair comparison with OPSA and QSSR-OPSD (including rubric-conditioned SFT), as extending SFT beyond one epoch leads to excessively high ORR for ThinkSafe and STAR-1 or substantial degradation in reasoning performance for SafeChain (Table 2). Safe RLHF (Dai et al., 2024)[8] follows the original three-stage pipeline. We train separate helpfulness reward and

[5] http://github.com/UCSC-VLAA/STAR-1
[6] http://github.com/uw-nsl/safechain
[7] http://github.com/seanie12/ThinkSafe
[8] http://github.com/PKU-Alignment/safe-rlhf

harmlessness cost models for two epochs with a learning rate of 2×10^{-5} and thinking disabled, and then optimize the policy for one epoch using PPO-Lagrangian, with thinking enabled, a 2,048-token generation budget, an actor learning rate of 1×10^{-5}, a KL penalty of 0.01, and a supervised auxiliary loss on thinking-style Alpaca data.

D.7 Cross-judge health safe-completion check

We use GPT-5.4 to test whether the health SCR trends obtained with the primary GPT-4o judge persist under a second judge. GPT-4o remains the sole judge for headline SCR results; GPT-5.4 is a supplementary check, not a replacement metric. Both judges follow the same Health-ORSC-Bench intent, safety, and helpfulness rubric described in Appendix D.4.

Table 9: Qwen3-8B health SCR under two judges (higher is better). GPT-4o values are the primary results; GPT-5.4 values are the supplementary cross-check.

Method	GPT-4o	GPT-5.4
Base	0.6966	0.5642
PolicyAlign	0.7143	0.5716
SafeSteer	0.7058	0.5619
OPSA	0.3606	0.3053
QSSR-OPSD	**0.7480**	**0.6160**

Both judges rank QSSR-OPSD first, PolicyAlign second, and OPSA last (Table 9). Relative to Base, QSSR-OPSD improves SCR by 5.13 percentage points under GPT-4o and 5.18 points under GPT-5.4; relative to OPSA, the gains are 38.73 and 31.07 points. Thus, the main conclusion—higher SCR for QSSR-OPSD than the compared baselines and lower SCR for OPSA than Base—holds under both judges. This check supports consistency in overall trends.

D.8 Paired outcomes on borderline health queries

Figure 2 compares each trained student model (e.g., OPSA and QSSR-OPSD) with the unaligned Base on the same 1,000 Health-ORSC-Bench Hard-1K queries, using the SCR judge and aggregation protocol. Let $\mathcal{P}_m$ contain pairs with valid safety labels for both Base and model m, and let s_i^0 and s_i^m denote their labels. We report the fraction of Base-unsafe responses made safe by m (rescue) and the fraction of Base-safe responses made unsafe by m (regression):

$$R_m = \frac{\sum_{i \in \mathcal{P}_m} \mathbf{1}[s_i^0 = \text{Unsafe},\ s_i^m = \text{Safe}]}{\sum_{i \in \mathcal{P}_m} \mathbf{1}[s_i^0 = \text{Unsafe}]}, \qquad G_m = \frac{\sum_{i \in \mathcal{P}_m} \mathbf{1}[s_i^0 = \text{Safe},\ s_i^m = \text{Unsafe}]}{\sum_{i \in \mathcal{P}_m} \mathbf{1}[s_i^0 = \text{Safe}]}. \tag{30}$$

Panels (a) and (b) of Figure 2 display R_m and G_m as percentages. For a descriptive helpfulness diagnostic, we map Safe categories A/B/C/D to $1/2/3/4$ and Unsafe responses to 0. Let h_i be this score and $\Delta h_i = h_i^m - h_i^0$. Panel (c) reports mean Δh_i with 95% confidence intervals; panel (d) classifies $\Delta h_i > 0.01$ as increased, $|\Delta h_i| \leq 0.01$ as unchanged, and $\Delta h_i < -0.01$ as decreased. This diagnostic is not SCR: SCR counts only Safe C/D responses, whereas Δh_i combines safety and helpfulness changes across all paired queries.

E Reasoning Evaluation

E.1 Benchmarks and scoring

We evaluate reasoning ability along three complementary axes—mathematics, scientific knowledge, and code generation—using benchmarks with objectively verifiable outcomes. This design focuses on robust end-task performance rather than attempting to provide exhaustive coverage of reasoning capabilities. The three benchmarks are chosen because they probe distinct forms of reasoning and are also widely used in recent reasoning-model evaluations, including DeepSeek-R1 and Qwen3 (DeepSeek-AI, 2025; Yang et al., 2025a).

- **AIME 2026**: 30 integer-answer problems (15 each from AIME I/II) from `math-ai/aime26` (Zhang & Math-AI Team, 2026). We use AIME because its mathematical problems require multi-step reasoning while admitting an unambiguous, automatically verifiable final answer. Models are prompted to provide step-by-step reasoning followed by a `\boxed{}` answer, and the extracted integer is matched against the official key.
- **GPQA Diamond**: 198 expert-written questions spanning biology, physics, and chemistry (Rein et al., 2024). GPQA complements AIME by testing scientific reasoning and knowledge on questions designed to be difficult even for domain experts. We use chain-of-thought prompting and evaluate final-answer correctness by lenient answer-letter matching, avoiding sensitivity to formatting while not assigning partial credit.
- **HumanEval**: 164 Python programming tasks (Chen et al., 2021b). HumanEval evaluates whether reasoning can be translated into executable programs, with correctness determined by the official unit tests rather than similarity to a reference implementation. Generated code is executed in an isolated environment with a 30-second timeout to ensure both functional evaluation and safe execution.

Together, these benchmarks cover three qualitatively different reasoning outcomes while keeping evaluation automatically verifiable. We use repeated sampling to reduce dependence on any single decoding trajectory, which is particularly important for the relatively small AIME set.

E.2 Sampling and aggregation

For reasoning models, we sample 32 completions per problem at temperature 0.6 and top-$p = 0.95$, split into two replicas of 16 samples. Each completion has a 24,576-token output budget, including reasoning and the final answer, within a configured 32,768-token context with an 8,192-token prompt reserve. As a non-reasoning baseline, Llama-3.1-8B uses the same 32,768-token context but is evaluated with a single greedy sample ($T = 0$, $n = 1$) and an 8,192-token generation limit, since it does not follow an explicit reasoning protocol or use a dedicated CoT/`\boxed{}` format.

If $c_{b,i}$ out of 32 samples solve problem i on benchmark b, we estimate sample-based pass@1 as

$$\widehat{\text{pass@1}}_b = \frac{1}{N_b}\sum_{i=1}^{N_b}\frac{c_{b,i}}{32}, \qquad (N_{\text{AIME}}, N_{\text{GPQA}}, N_{\text{HumanEval}}) = (30, 198, 164). \tag{31}$$

This is the $k = 1$ sample-based estimator (Chen et al., 2021b), rather than best-of-32 or majority voting. We then report the unweighted mean across the three benchmarks:

$$\text{Reasoning Avg.} = \tfrac{1}{3}\left(\widehat{\text{pass@1}}_{\text{AIME}} + \widehat{\text{pass@1}}_{\text{GPQA}} + \widehat{\text{pass@1}}_{\text{HumanEval}}\right). \tag{32}$$

Equal weighting prevents the larger GPQA and HumanEval sets from dominating the aggregate and gives each reasoning axis equal contribution. The fixed 32-sample budget is applied uniformly, including model failures, rather than filtering unsuccessful parses or executions, since excluding such failures could mask limitations in the model's reasoning capability.

F Training dynamics: teacher–student disagreement

Over the roughly 310 logged steps, both mixed-KL losses decrease rapidly and then plateau at distinct levels: approximately 0.02–0.03 for OPSA and 0.21–0.23 for QSSR-OPSD. The teacher–student top-512 overlap rises to about 0.95–0.96 for OPSA, whereas QSSR-OPSD stabilizes around 0.84–0.85 after an initial decline. This persistent gap indicates that the query-specific privileged-information teacher remains more distinct from the student than the label-conditioned OPSA teacher under the logged on-policy rollouts. However, a lower loss does not imply better alignment: the two teachers define different targets, and OPSA's lower loss coincides with lower CSS and SCR in Table 1.

Both methods also reduce mean response length from roughly 1,450 tokens to similar ranges after the initial transient (about 780–810 for QSSR-OPSD and 730–790 for OPSA). Thus, comparable

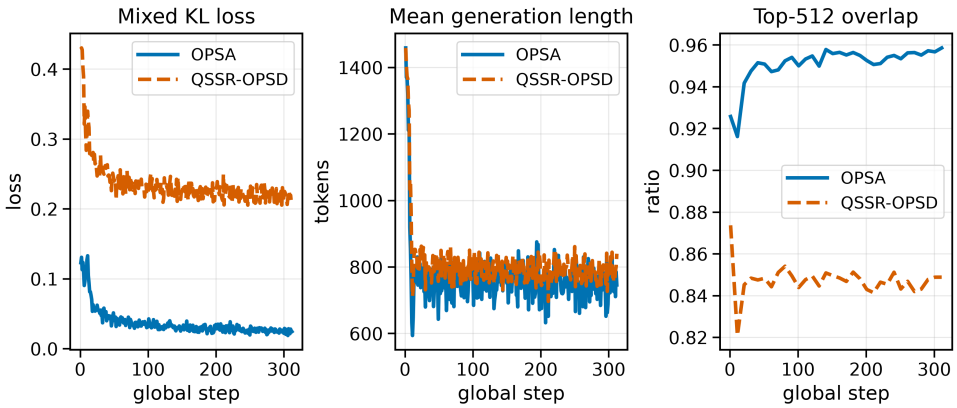


Figure 3: Training dynamics for OPSA (solid blue) and QSSR-OPSD (dashed orange): mixed-KL distillation loss (0.5 forward $+\,0.5$ reverse KL), mean generated length, and teacher–student top-512 token overlap. Despite comparable lengths, QSSR-OPSD maintains higher loss and lower overlap throughout the logged window.

response lengths do not necessarily imply similar response composition or content. Since on-policy distillation provides supervision on trajectories the student actually visits (Agarwal et al., 2024), persistent teacher–student disagreement may reflect useful corrections. However, the loss and overlap gaps alone do not establish that the remaining disagreement is beneficial; its value must be assessed through downstream performance.

G Ablation Results

G.1 Rubric generation ablations

Table 10: Rubric-generation ablations for Qwen3-8B. We vary the model used to generate query-specific rubrics and whether the generator is provided the reference rule pool.

Rubrics generation	ASR ↓	ORR ↓	CSS ↑	SCR ↑	Reason. ↑
GPT-5.4 (default)	0.0379	0.1207	0.9207	0.7480	0.6922
Qwen3-8B (self-generated)	0.0982	0.0956	0.9031	0.7493	0.6932
GPT-5.4 without the rule pool	0.0702	0.0993	0.9153	0.7413	0.6924

Table 10 shows a clear compliance–refusal tradeoff across rubric generators. Self-generated Qwen3-8B rubrics reduce ORR (0.0956 vs. 0.1207) but increase ASR (0.0982 vs. 0.0379), yielding a lower CSS (0.9031). Removing the reference rule pool partially recovers CSS (0.9153) via lower ORR (0.0993) but still higher ASR (0.0702) than default, indicating that the rule pool stabilizes the generator toward stricter harmful constraints while increasing benign refusals, which leads to a higher CSS score.

G.2 Reasoning-token supervision and divergence

Table 11: Reasoning-token supervision on Qwen3-8B. Excluding reasoning tokens changes the loss mask; it does not establish that reasoning generation was disabled.

Setting	ASR ↓	ORR ↓	CSS ↑	SCR ↑	Reason. ↑
With reasoning-token loss	0.0379	0.1207	0.9207	0.7480	0.6922
Without reasoning-token loss	0.0513	0.1190	0.9148	0.7542	0.6902

Table 12: Divergence ablations on Qwen3-8B. Mixed and reverse KL tie in the reported CSS to four decimals.

Setting	ASR ↓	ORR ↓	CSS ↑	SCR ↑	Reason. ↑
Mixed KL	0.0379	0.1207	0.9207	0.7480	0.6922
Forward KL	0.0344	0.1274	0.9191	0.7383	0.6816
Reverse KL	0.0426	0.1160	0.9207	0.7439	0.6936

Removing reasoning tokens from the loss changes which tokens receive supervision, not whether the model generates thinking. Table 11 shows that dropping the reasoning-token loss slightly increases ASR (0.0379→0.0513) while leaving ORR nearly unchanged (0.1207→0.1190), resulting in a small CSS decrease (0.9207→0.9148). At the same time, health SCR increases marginally (0.7480→0.7542) and the reasoning average is essentially flat (0.6922→0.6902), suggesting that explicitly supervising reasoning tokens is not the main driver of downstream reasoning accuracy in this setting but can help preserve the overall safety/over-refusal balance captured by CSS.

Forward, reverse, and mixed KL follow Equations 4–5. Table 12 indicates that the KL direction mainly trades off harmful compliance against benign refusals: forward KL yields the lowest ASR (0.0344) but the highest ORR (0.1274), reducing both CSS (0.9191) and SCR (0.7383); reverse KL lowers ORR (0.1160) and slightly improves the reasoning average (0.6936) but at higher ASR (0.0426). Mixed KL matches the best CSS to four decimals (0.9207, tied with reverse KL) while maintaining the highest SCR among the three (0.7480), so we use it as a stable default.