Criterion-Activated Preference Synthesis: Attribution-Preserving Preference Data from Explicit Standards

Anonymous ACL submission

Abstract

Direct Preference Optimization (DPO) is notoriously sensitive to *attribution ambiguity*: when chosen and rejected responses differ along multiple entangled dimensions, the model cannot identify which behavior the preference label teaches. We propose **Criterion-Activated Preference Synthesis (CAPS)**, a criterion-first framework that turns explicit quality standards into attribution-verified preference pairs. CAPS synthesizes contexts and responses from a target criterion, then applies dual-signal scoring: a rubric-bound judge measures criterion compliance, while a blind general judge measures holistic quality. Two filters close the attribution loop: the **Empirical Activation Test (EAT)** keeps criterion-context pairs where satisfying the criterion improves blind quality, and the **Attribution Ratio (AR)** keeps pairs whose quality gap is dominated by the target criterion. Instantiated with general rubrics, medical rubrics, and Chinese legal statutes, CAPS shows strong data efficiency and domain transfer. With **20k** general synthetic pairs, it outperforms **35k** human-annotated HelpSteer3 pairs on IFEval, IFBench, AlpacaEval, and Creative Writing v3. In domain settings, CAPS improves HealthBench with 2k medical pairs and improves Qwen3-8B/14B by +4.9 to +6.6 points on legal benchmarks using only public statutes, without human preference labels. We release the resulting CAPS-Legal preference dataset.

1 Introduction

Post-training via Direct Preference Optimization (DPO; Rafailov et al., 2024) has become the dominant paradigm for LLM alignment, owing to its algorithmic simplicity over reinforcement learning from human feedback (Ouyang et al., 2022; Bai et al., 2022; Ziegler et al., 2019). However, DPO's effectiveness is acutely sensitive to the *quality* of its preference signal. We identify *attribution ambiguity* as a pervasive failure mode in current preference datasets: when a chosen response simultaneously outperforms its rejected counterpart along multiple dimensions (fluency, accuracy, formatting, depth), the resulting composite "better" label conceals *which* dimension is being learned.

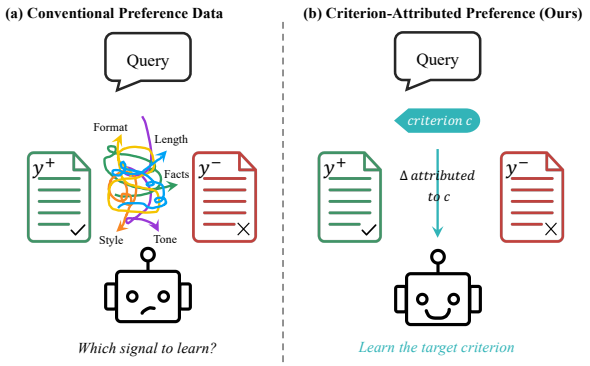

Figure 1: Comparison between conventional preference data (left) and CAPS (right). Conventional methods produce pairs with confounded quality gaps across multiple dimensions—the model cannot determine which signal to learn. CAPS activates a specific criterion c and verifies that the quality gap is primarily attributable to c via dual-signal scoring.

Wang et al. (2025a) provide direct evidence of this failure: aggregating multi-model responses—a common scaling strategy—paradoxically enables reward hacking, with attack success rates escalating from 0.12 to over 0.51 as data volume increases. A growing line of work on preference data quality (He et al., 2025; Xiao et al., 2025; Guo et al., 2025; Liu et al., 2024) consistently demonstrates that curated subsets outperform full-scale datasets across diverse filtering strategies, confirming that *data attribution matters more than data volume*.

A well-documented theoretical root cause is rank deficiency (Rafailov et al., 2024): when pairs conflate multiple quality dimensions, the data matrix collapses into a low-dimensional subspace admitting infinitely many degenerate optima, with characteristic hump-shaped reward curves (Gao et al.,

2023). The critical insight is the converse: if each pair's quality gap is localized to a *specific, identifiable criterion*, the effective preference matrix becomes higher-rank, directly mitigating overoptimization.

We propose **Criterion-Activated Preference Synthesis (CAPS)**, which reverses the conventional pipeline: instead of collecting preferences and post-hoc filtering, CAPS begins with an explicit quality criterion c and synthesizes preference pairs whose quality gap is *primarily attributable* to c. The key mechanism is *Dual-Signal Scoring*: a rubric-bound judge measures criterion compliance, while a general blind judge measures holistic quality without knowledge of c. CAPS enforces two complementary filters: the *Empirical Activation Test* (EAT) discards criterion-context combinations that fail to produce a meaningful effect on blind quality scores—selecting *learnable* criteria; the *Attribution Ratio* (AR), a bounded ranking metric, retains only pairs where the criterion dominates the quality gap—selecting *attribution-pure* pairs.

We instantiate CAPS on three criterion sources—HelpSteer3 (Wang et al., 2025b) (general), RaR Medicine (Gunjal et al., 2025) (clinical), and legal statutes—demonstrating that any codified standard can be converted into high-quality preference supervision. Experiments show that 20k general CAPS pairs outperform 35k HelpSteer3 human pairs across general benchmarks, while CAPS-Legal improves Qwen3-8B by +6.60 PLawBench (Shi et al., 2026) and +6.04 MSLR (Yu et al., 2025) over base and avoids the degradation observed with the Infinity-Instruct legal subset. Controlled scaling experiments further suggest moderate sample-efficiency gains in this setting.

Our contributions are threefold:

1. We propose CAPS, the first preference data generation framework that uses explicit criteria as the *source* rather than post-hoc descriptor of preference signal, enabling attribution-preserving synthesis from any codified domain standard.
2. We introduce two complementary verification mechanisms that close the criterion-attribution loop. **EAT** tests whether satisfying c improves blind quality in context x, filtering for criteria that *can* be learned. **AR** keeps pairs whose quality gap is dominated by the target criterion, filtering for pairs that *learn the right thing*. These group-level and pair-level checks are non-redundant.
3. We demonstrate that domain standards can be systematically converted into preference supervision, exploiting the "standard-rich, data-scarce" opportunity. In our general-domain experiments, 20k synthetic pairs outperform the 35k human-annotated HelpSteer3 baseline, suggesting improved sample efficiency in this setting; CAPS also substantially improves legal reasoning over both base models and the Infinity-Instruct legal subset baseline. We release an open-source legal preference dataset synthesized from Chinese statutes.

2 Methodology

2.1 Overview

We propose **Criterion-Activated Preference Synthesis** (CAPS), a framework that transforms explicit evaluation criteria into preference pairs with verified attribution. CAPS operationalizes this through dual-signal scoring (§2.3) and two complementary filters that operate at different granularities: **EAT** (§2.4) selects *learnable criterion-context pairs* by verifying that satisfying c measurably shifts blind quality scores; **AR** (§2.5) selects *attribution-pure preference pairs* by verifying that the rubric is the dominant driver of the quality gap within each pair. The two filters are non-redundant—EAT operates on groups and cannot catch per-pair confounding, while AR operates on individual pairs and cannot detect unlearnable criteria. Figure 2 illustrates the full pipeline.

2.2 Context Synthesis and Rollout

Given a criterion c from a criterion bank $\mathcal{C}$, a generator model G produces N diverse contexts (user queries) where c is potentially relevant, satisfying three requirements: (i) **natural quality sensitivity**—c compliance should naturally affect response quality; (ii) **ecological validity**—contexts resemble realistic deployment queries; (iii) **diversity**—variation in domain, complexity, and intent. Context synthesis makes *no prior judgment* about whether c will activate—this is deferred to EAT (§2.4), preventing confirmation bias.

For each context x, we collect responses from a diverse model pool $\mathcal{M}$ at multiple temperatures, yielding M candidates (one per model-temperature combination). The multi-source design increases response diversity and prevents quality clustering.

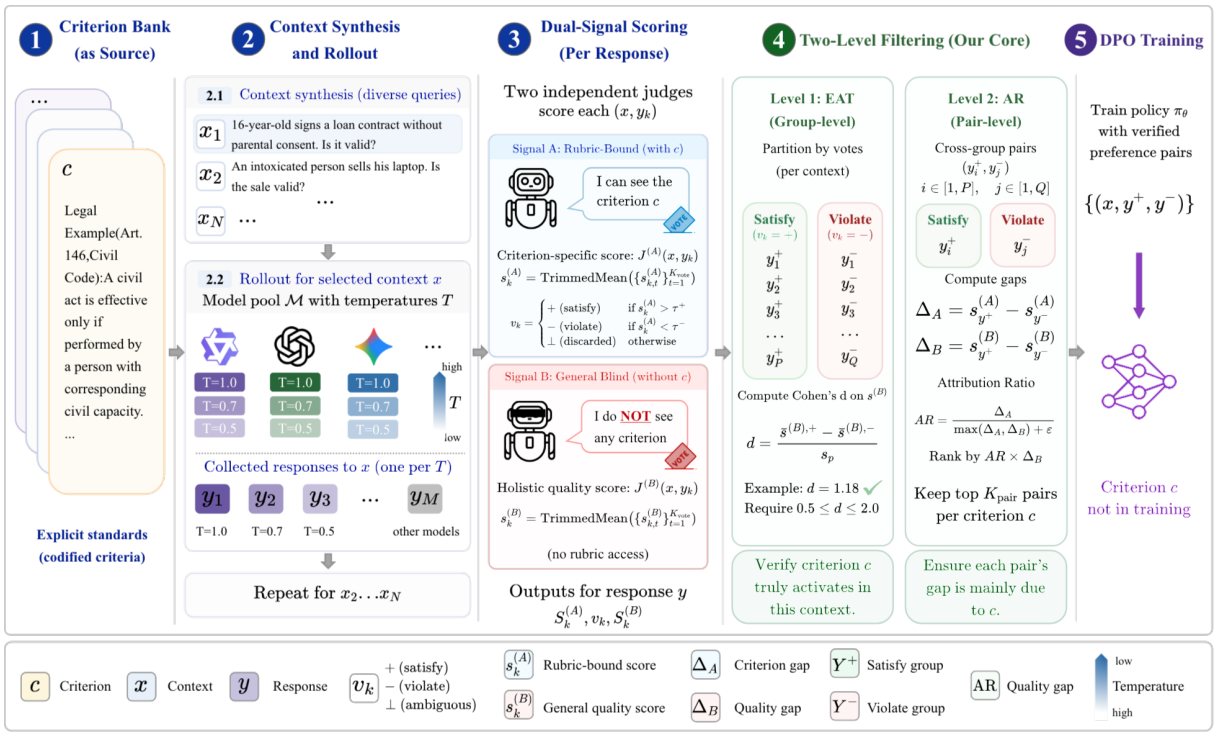

Figure 2: Overview of the CAPS pipeline. Given criteria from a criterion bank, CAPS synthesizes attribution-verified preference pairs through context synthesis, multi-source rollout, and dual-signal scoring. EAT filters at the group level (criterion activation verification); AR filters at the pair level (attribution specificity ranking). The criterion c does not enter the final DPO training—only the verified (x, y^+, y^-) triplets are used.

2.3 Dual-Signal Scoring

The core challenge is that no single signal can simultaneously verify (i) *whether* a response satisfies a given criterion and (ii) *whether* doing so improves perceived quality from a holistic standpoint. We therefore decompose preference judgment into two independent signals with distinct roles: one explicitly conditioned on the criterion (Signal A, rubric-bound), the other deliberately blind to it (Signal B, general). Both signals follow the *same* aggregation procedure—each judge is queried K_{vote} times with stochastic decoding, the highest and lowest scores are trimmed to remove outliers, and the remainder is averaged into a single scalar score. This trimmed-mean aggregation stabilizes both signals against per-call judge variance.

Signal A (Rubric-Bound). A judge $J^{(A)}$ scores each context-response pair (x, y_k) on a 1–10 scale *with full access to criterion* c, producing K_{vote} stochastic samples $\{s_{k,t}^{(A)}\}_{t=1}^{K_{\text{vote}}}$. The aggregated score is

$$s_k^{(A)} = \text{TrimmedMean}\big(\{s_{k,t}^{(A)}\}_{t=1}^{K_{\text{vote}}}\big), \qquad (1)$$

which serves two purposes. First, $s_k^{(A)}$ feeds directly into the pair-level attribution score Δ_A (§2.5). Second, $s_k^{(A)}$ partitions responses into satisfy/violate groups via a double-sided threshold:

$$v_k = \begin{cases} + \text{ (satisfy)} & \text{if } s_k^{(A)} > \tau^+ \\ - \text{ (violate)} & \text{if } s_k^{(A)} < \tau^- \\ \perp \text{ (discarded)} & \text{otherwise} \end{cases} \qquad (2)$$

Responses falling in the ambiguous middle band ($v_k = \perp$) are excluded from subsequent stages, ensuring clean group separation for EAT. The rubric-bound paradigm follows PROMETHEUS (Kim et al., 2023), with trimmed-mean aggregation further stabilizing the decision boundary (Yin et al., 2018).

Signal B (General Blind). A judge $J^{(B)}$ scores the same response on a 1–10 scale *without access to criterion* c, again with K_{vote} stochastic samples and trimmed-mean aggregation:

$$s_k^{(B)} = \text{TrimmedMean}\big(\{s_{k,t}^{(B)}\}_{t=1}^{K_{\text{vote}}}\big),$$
$$s_{k,t}^{(B)} = J^{(B)}(x, y_k). \qquad (3)$$

Unlike Signal A, Signal B is *not* thresholded and *does not* produce a group label—it is consumed downstream as a continuous quality signal, used

for both EAT's effect-size computation (§2.4) and AR's holistic gap Δ_B (§2.5). The judge evaluates quality holistically—covering fluency, factual accuracy, coherence, domain depth, and safety—a paradigm with established human-LLM agreement (Jia et al., 2026; Zheng et al., 2024). Because no rubric is provided, $s_k^{(B)}$ captures the *total* perceived quality of the response, with any criterion-relevant or confounding effect on quality manifesting in this score—a property exploited by AR in §2.5 to bound attribution. Issuing the two signals as independent API calls with disjoint prompts and no shared context window ensures prompt-level information asymmetry between them.

2.4 Empirical Activation Test (EAT)

EAT decides: *should this criterion-context pair enter the data pool at all?* Not every criterion produces a useful training signal in every context—a criterion about "citing primary sources" degenerates into noise when applied to a creative writing prompt. EAT asks: does satisfying c measurably shift the perceived quality of responses to x? If yes, (x, c) generates a learnable signal; if no, it is discarded before pair construction. This is a group-level decision made once per (x, c) over all rollouts.

Group formation. Using the labels v_k from Signal A, responses are partitioned into a satisfy group $Y^+ = \{y_k \mid v_k = +\}$ and a violate group $Y^- = \{y_k \mid v_k = -\}$, requiring $|Y^+| \geq 2$ and $|Y^-| \geq 2$ for valid variance estimation. Responses with ambiguous labels ($v_k = \perp$) are excluded. If a context fails to elicit both satisfying and violating responses (i.e., $|Y^+| < 2$ or $|Y^-| < 2$), it is deemed uninformative for criterion c and is discarded—this prevents degenerate effect-size estimates from insufficient group contrast. If *no* context passes EAT for criterion c, the criterion itself is excluded from the active bank.

Effect size computation. For each group, we compute the mean and unbiased sample variance of Signal B scores: $\bar{s}^{(B),+} = \frac{1}{n^+} \sum_{y_k \in Y^+} s_k^{(B)}$ (and analogously $\bar{s}^{(B),-}$), with sample variances σ_+^2 and σ_-^2 defined accordingly, where $n^+ = |Y^+|$ and $n^- = |Y^-|$. We then compute Cohen's d (Cohen, 1977) on these scores (crucially, *not* Signal A):

$$d = \frac{\bar{s}^{(B),+} - \bar{s}^{(B),-}}{s_p}, \quad (4)$$

$$s_p = \sqrt{\frac{(n^+ - 1)\sigma_+^2 + (n^- - 1)\sigma_-^2}{n^+ + n^- - 2}}$$

Computing d on Signal B avoids circularity: a large d on Signal A merely confirms the judge can read the rubric, whereas Signal B measures whether criterion satisfaction translates to *perceived quality improvement* from a blind evaluator—the property required for effective preference learning.

Double-sided threshold. We require $d \in [\tau_{\min}, \tau_{\max}]$. Since d is the ratio of group mean difference to pooled standard deviation s_p, extreme values arise from either component. When $d < \tau_{\min}$: either the criterion has no effect ($\bar{s}^{(B),+} \approx \bar{s}^{(B),-}$, criterion irrelevant), or the signal is drowned by within-group variance from multi-model heterogeneity or judge instability. When $d > \tau_{\max}$: either the quality gap is trivially large (e.g., on-topic vs. off-language responses, encouraging reward hacking (He et al., 2025; Wang et al., 2025a)), or near-zero variance from few coarse-grained samples produces a statistical artifact ($s_p \to 0$, inflating d despite a small absolute gap). See Appendix E for detailed case analysis.

2.5 Attribution Ratio (AR) and Pair Selection

AR decides: *is this specific pair's preference signal actually about the criterion?* Passing EAT confirms that c activates *on average* in context x, but says nothing about whether any *particular* pair isolates the criterion. A randomly drawn pair from an EAT-passing context might differ predominantly in fluency or formatting—not in criterion compliance. AR provides a per-pair audit: by comparing the criterion-specific gap Δ_A against the holistic gap Δ_B, it quantifies how much of the observable preference is attributable to c versus to confounders. This is also the central mechanism by which CAPS mitigates multi-model reward hacking (Wang et al., 2025a): stylistic differences across model scales inflate Δ_B without proportionally increasing Δ_A, yielding low AR scores that filter out confound-driven pairs (see Appendix G).

For a candidate pair (y^+, y^-) where $y^+ \in Y^+$ and $y^- \in Y^-$, we define:

$$\Delta_A = s_{y^+}^{(A)} - s_{y^-}^{(A)}, \quad \Delta_B = s_{y^+}^{(B)} - s_{y^-}^{(B)} \quad (5)$$

where Δ_A captures the criterion-specific score gap and Δ_B captures the holistic quality gap. $\Delta_A > 0$ is guaranteed by construction since $y^+ \in Y^+$ (satisfies c) and $y^- \in Y^-$ (violates c). $\Delta_B > 0$ is enforced via the directional consistency constraint below.

Attribution Ratio (AR). We define:

$$\text{AR} = \frac{\Delta_A}{\max(\Delta_A, \Delta_B) + \varepsilon} \in [0, 1] \quad (6)$$

where $\varepsilon = 10^{-8}$ prevents division by zero. The max operator normalizes the ratio, guaranteeing AR $\in [0, 1]$ regardless of whether Δ_A or Δ_B is larger—when $\Delta_A \geq \Delta_B$, AR $= \Delta_A/(\Delta_A + \varepsilon) \approx 1$ for $\Delta_A \gg \varepsilon$; when $\Delta_B > \Delta_A$, the denominator equals $\Delta_B + \varepsilon$ and AR decreases proportionally. The AR admits a natural interpretation:

- AR $\geq \tau_{\text{AR}}$ (retained): the criterion-specific gap is a substantial driver of the quality difference. Higher AR indicates stronger criterion dominance.
- AR $< \tau_{\text{AR}}$ (filtered): confounding factors dominate ($\Delta_B \gg \Delta_A$). Such pairs are excluded.

AR is used strictly as a ranking metric rather than a calibrated proportion, since LLM judge scores are fundamentally ordinal (Kim et al., 2023). Its empirical effectiveness is validated in §4.1, where high-AR pairs consistently outperform low-AR pairs at fixed data budget.

Pair quality constraints. Beyond AR $\geq \tau_{\text{AR}}$, we impose four additional constraints grounded in empirical findings:

- **Directional consistency:** We strictly require $\Delta_A > 0$ and $\Delta_B > 0$—the chosen response must score higher on *both* the criterion-specific and holistic dimensions. This eliminates paradoxical pairs where criterion satisfaction coincides with overall quality degradation.
- **Chosen quality:** $s^{(B)}_{y^+} \geq q_{\min}$, ensuring the chosen response has adequate overall quality rather than being merely "less bad" (He et al., 2025).
- **Margin bounds:** $\Delta_B \in [\Delta_{\min}, \Delta_{\max}]$, avoiding both noisy pairs (gap too small) and trivial pairs (gap too large) (Xiao et al., 2025).
- **Diversity cap:** at most K_{rep} pairs per (x, c), preventing over-representation of easy-to-distinguish contexts (Yu et al., 2026).

Ranking and selection. Valid pairs are ranked by:

$$\text{score}(y^+, y^-) = \text{AR} \times \Delta_B \quad (7)$$

This jointly rewards *attribution purity* (high AR) and *signal strength* (high Δ_B). Neither alone suffices: AR $= 1$ with $\Delta_B \approx 0$ contributes negligible gradient, while high Δ_B with low AR teaches confound-driven preferences. We select the top K_{pair} pairs per criterion-context combination (x, c).

Complementarity. EAT and AR address distinct failure modes: EAT-only filtering misses per-pair confounding (e.g., model identity driving quality gaps within an activated context (Wang et al., 2025a)); AR-only filtering misses contexts where the criterion fails to activate entirely—without EAT, all contexts advance to pair selection regardless of whether criterion satisfaction actually correlates with quality, producing nominally high-AR pairs from judge noise in uninformative contexts. Together, EAT establishes that *signal exists*; AR confirms that *signal is correctly attributed*. We empirically validate this in §4.1.

2.6 Training

We train the target policy π_θ on the selected pairs via standard DPO (Rafailov et al., 2024); criterion c is *not* used during training. This isolation ensures that all performance gains are directly attributable to data quality rather than algorithmic novelty, and makes CAPS compatible with algorithmic variants such as IPO (Azar et al., 2024) and KTO (Ethayarajh et al., 2024).

3 Experiments

We conduct a comprehensive evaluation of CAPS across three domains: general-purpose alignment, healthcare, and practical legal tasks.

3.1 Experimental Setup

Datasets and criterion sources. CAPS is instantiated on three criterion sources and synthesizes domain-specific preference pairs from each: (1) **General**: 35k pair-level criteria extracted from HelpSteer3 by Auto-Rubric → 20k synthesized preference pairs; (2) **Medical**: 22k pair-level criteria from RaR Medicine (Gunjal et al., 2025) → 2k synthesized preference pairs[1]; (3) **Legal**: ∼2k Chinese statutory provisions → 6k synthesized preference pairs (corpus details in Appendix C). General/Medical criteria are pair-level, while Legal

[1]The lower medical conversion rate is mainly due to EAT filtering: RaR-Medicine-derived rubrics often target multi-step reasoning, where activation requires highly specific contexts.

(a) General Benchmarks

Method	Data	Instruction Following				Open-ended	
		IFEval_S	IFEval_L	IFB_S	IFB_L	Alpaca	Creative
Qwen3-8B							
Base	–	82.26	85.95	25.67	30.00	57.00	59.00
HelpSteer3	35k	84.47	87.43	32.33	36.67	66.51	58.96
CAPS	20k	**86.32**	**89.46**	**35.00**	**38.33**	**68.59**	**74.53**
Qwen3-14B							
Base	–	85.40	88.54	27.67	30.00	63.23	70.42
HelpSteer3	35k	85.03	88.17	34.67	40.00	71.23	68.13
CAPS	20k	**87.43**	**90.02**	**38.33**	**42.67**	**72.49**	**79.66**
Reference							
Qwen3-32B	–	84.83	88.17	26.33	30.67	69.59	77.04
Qwen3-235B	–	87.43	90.20	39.00	41.67	82.61	86.48
Gemma-4-31B	–	91.87	94.09	49.00	52.67	77.26	86.00

(b) Medical

Method	Data	Health
Qwen3-8B		
Base	–	41.53
HelpSteer3	35k	45.19
RaR-Med.	22k	41.79
CAPS	2k	**45.50**
Qwen3-14B		
Base	–	45.34
HelpSteer3	35k	47.59
RaR-Med.	22k	44.07
CAPS	2k	**47.91**
Reference		
Qwen3-32B	–	44.36
Qwen3-235B	–	51.26
Gemma-4-31B	–	48.93

(c) Legal

Method	Data	PLaw	MSLR
Qwen3-8B			
Base	–	41.29	55.58
HelpSteer3	35k	44.94	45.03
Inf-Inst	10k	26.39	31.43
CAPS	6k	**47.89**	**61.62**
Qwen3-14B			
Base	–	46.31	60.37
HelpSteer3	35k	49.44	47.66
Inf-Inst	10k	39.98	46.33
CAPS	6k	**51.25**	**65.77**
Reference			
Qwen3-32B	–	53.06	65.08
Qwen3-235B	–	63.51	75.63
Gemma-4-31B	–	57.00	67.42

Table 1: Main results with separated general, medical, and legal evaluations. Panel (a) reports general alignment benchmarks, where CAPS uses the 20k general checkpoint. Panel (b) reports HealthBench, where CAPS uses the 2k medical checkpoint. Panel (c) reports legal benchmarks, where CAPS uses the 6k legal checkpoint. Qwen3-235B denotes Qwen3-235B-A22B-Instruct-2507. Shading indicates the corresponding CAPS data/checkpoint. **Bold** = best within each base-model block; S=strict, L=loose.

criteria are statutory provisions that can instantiate multiple contexts. Representative preference pairs from each domain are shown in Appendix K. We train one domain-specific DPO checkpoint per base model and evaluate it on the corresponding benchmarks. The legal criterion set and preference dataset are publicly released.

Baselines. We compare CAPS against five groups of baselines: (1) **Base models**: Qwen3-8B and Qwen3-14B (Yang et al., 2025); (2) **HelpSteer3 DPO**: standard DPO trained on 35k human-annotated HelpSteer3 preference pairs, evaluated on the general benchmarks and HealthBench(Arora et al., 2025); (3) **RaR-Medicine**: a medical question-answering dataset (Gunjal et al., 2025), used for SFT and evaluated on HealthBench; (4) **Infinity-Instruct legal subset**: 10k legal instructions filtered from Infinity-Instruct (Li et al., 2025) and used for SFT, evaluated on PLawBench/MSLR (details in Appendix I); and (5) **Reference models**: stronger zero-shot models, including Qwen3-32B, Qwen3-235B-A22B-Instruct-2507, and Gemma-4-31B, used to contextualize the performance of tuned 8B/14B models.

Metrics. IFEval (Zhou et al., 2023) and IFBench (Pyatkin et al., 2025)(strict/loose instruction-following), AlpacaEval (Dubois et al., 2025) LC, Creative Writing v3 (Paech, 2024), HealthBench (clinical reasoning), PLawBench and MSLR (legal reasoning/retrieval).

Implementation. For generation and dual-signal judging, we employ DeepSeek-V3.2 (DeepSeek-AI et al., 2025) in non-thinking mode[2] and construct rollout pools from multiple model-temperature combinations. CAPS checkpoints are trained with standard DPO, while the Infinity-Instruct legal subset uses SFT on instruction-responses. We report averages over three runs; detailed hyperparameters, reproducibility settings, artifacts, and responsible-use notes are provided in Appendices D and J.

3.2 Main Results

Table 1 summarizes results across the general, medical, and legal settings. CAPS improves the corresponding base models in all three domains while using compact synthesized preference sets.

General tasks. CAPS consistently improves over base and outperforms the 35k HelpSteer3 DPO baseline with only 20k synthetic pairs. The most salient gain is Creative Writing, where CAPS-8B approaches Qwen3-32B (74.53 vs. 77.04) and CAPS-14B surpasses it (79.66 vs. 77.04). This pattern is consistent with CAPS reinforcing controllable aspects of open-ended generation—instruction realization, structure, and style—rather than only optimizing narrow instruction-following scores. As a result, CAPS improves Creative Writing without degrading other general benchmarks.

Medical domain. CAPS improves HealthBench over base at both scales and roughly matches the much larger 35k HelpSteer3 DPO baseline using only 2k medical pairs (45.50 vs. 45.19 for 8B; 47.91

[2]DeepSeek API thinking-mode guide: https://api-docs.deepseek.com/guides/thinking_mode.

vs. 47.59 for 14B), which we treat as data-efficient parity. More notably, CAPS outperforms direct SFT on the 22k RaR-Medicine subset, which provides little benefit for 8B and degrades 14B, suggesting that verified preference conversion is a safer use of this source than direct instruction tuning. The modest gains likely reflect source–benchmark mismatch: RaR-Medicine focuses on clinical reasoning (Gunjal et al., 2025), while HealthBench also covers assistant-facing skills such as uncertainty handling and communication.

Legal domain. CAPS-Legal shows the clearest advantage in a standard-rich domain: using only public statutes and no human preference labels, it improves both 8B and 14B models over base and substantially outperforms the 10k Infinity-Instruct legal subset. The 35k HelpSteer3 baseline also does not reliably transfer to PLawBench/MSLR, especially degrading MSLR, suggesting that explicit statutory criteria provide more targeted supervision than either general preference data or legal-domain instruction data alone. CAPS-14B also slightly exceeds Qwen3-32B on MSLR (65.77 vs. 65.08).

4 Analysis

4.1 Capability Gains Beyond Data Scale and Distillation

Data efficiency. Under a fixed 5k HelpSteer3-rubric budget, CAPS matches the 5k Original-DPO baseline with 3k synthesized pairs and modestly outperforms the 10k Original-DPO baseline at 6.5k pairs (Figure 3). Further scaling to 10k–13k pairs brings only marginal gains, suggesting that criterion-bound synthesis can improve data efficiency, though the gains are moderate in this setting.

Distillation control. Table 2 shows that conventional rollout-SFT distillation does not explain CAPS gains: training Qwen3-8B on responses from stronger models slightly reduces IFBench performance relative to the base model. Since Qwen3-8B is already a strong base model, direct SFT on generated responses may disrupt its instruction-following behavior; in contrast, full CAPS DPO reaches 35.00/38.33, indicating that the key signal is contrastive supervision over criterion-activated pairs.

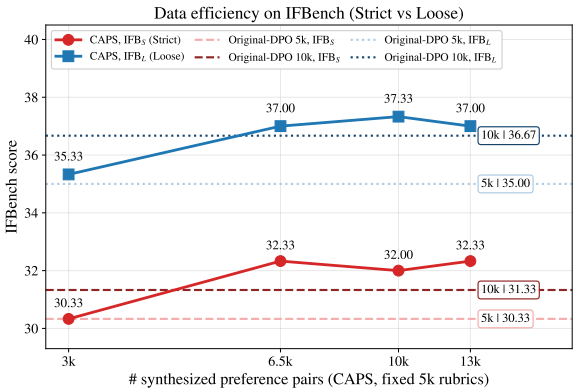

Figure 3: Data-efficiency comparison on IFBench under a fixed 5k pair-level rubric budget. CAPS (red) synthesizes varying numbers of preference pairs from the same rubrics; Original-DPO baselines (gray) use 5k and 10k sampled HelpSteer3 pairs. CAPS at 6.5k pairs already exceeds the 10k Original-DPO baseline on both metrics and saturates beyond.

Method	IFB$_S$	IFB$_L$
Qwen3-8B (Base)	25.67	30.00
Qwen3-235B Rollout SFT	24.33	28.67
Gemma-4-31B Rollout SFT	23.33	26.00
CAPS DPO (Full)	**35.00**	**38.33**

Table 2: Distillation-control baselines on IFBench. Rollout SFT denotes SFT on answers generated by Qwen3-235B or Gemma-4-31B from the original 35k HelpSteer3 queries.

4.2 What Drives CAPS Gains?

We design two complementary ablations to validate CAPS's core design. Table 3 provides a pipeline-level comparison against single-judge controls; Table 4 isolates the contribution of each filtering component.

Structured synthesis matters beyond single-judge signals. As shown in Table 3, random pairing provides a sanity check: it improves IFB$_S$ but hurts PLawBench, indicating that rollout diversity alone is not sufficient. More importantly, both single-judge controls improve over random selection, but each is limited by its signal: the general judge is stronger on IFBench, while the rubric judge is stronger on PLawBench, where legal correctness is tied to explicit statutory criteria. Full CAPS performs best in both settings (35.00 IFB$_S$, 47.89 PLaw), suggesting that combining rubric-bound and general signals through EAT+AR is more effective than relying on either judge alone.

Setting	IFB_S	PLaw
Qwen3-8B Base	25.67	41.29
Random	29.67	33.39
General judge	31.67	42.49
Rubric judge	31.00	44.10
CAPS(Rubric+General Judge)	**35.00**	**47.89**

Table 3: Pipeline-level comparison. "Random" samples preference pairs from the rollout pool without judge-based filtering. "General judge" and "Rubric judge" use only Signal B or Signal A for single-signal pair selection, respectively, and therefore cannot apply the full EAT+AR verification. "CAPS synthesis" applies the full dual-signal EAT+AR pipeline.

Setting	IFB_S	PLaw
Full CAPS	35.00	47.89
– AR ranking	33.00	46.95
– AR (entire)	31.33	46.65
– EAT	28.67	46.21

Table 4: Component ablation (subtractive). Starting from Full CAPS, we progressively remove: AR-based ranking (pairs selected randomly among AR-passing candidates), AR filtering entirely (all EAT-passed pairs retained), and EAT (all criterion-context pairs advance to pair selection).

Each component contributes incrementally. Table 4 shows progressive degradation as AR ranking, AR filtering, and EAT are removed, consistent with their roles in Section 2.4–2.5. The effect is stronger on IFBench: removing EAT drops IFB_S from 35.00 to 28.67, suggesting that general rubrics rely more on group-level activation checks. On PLawBench, the drops are smaller but consistent (47.89 to 46.21), likely because statutory criteria are already closely aligned with the target domain. Overall, EAT controls criterion-context learnability, while AR improves pair-level attribution and ranking.

5 Related Work

Preference data quality and synthesis. Scaling approaches (Cui et al., 2024; Chen et al., 2024; Yuan et al., 2024) produce large preference datasets without controlling *what* differentiates chosen from rejected. CLAIR (D'Oosterlinck et al., 2025) constrains via minimal edits, but only 43% Jaccard similarity between intended and actual attributions. Quality-filtering work (Wang et al., 2025a; He et al., 2025; Xiao et al., 2025; Guo et al., 2025; Liu et al., 2024) demonstrates curated subsets outperform full-scale data, but all apply *post-hoc filtering*. CAPS *constructs* attribution-traceable pairs from explicit criteria by design.

Rubric-guided methods. Rubric-based approaches span four layers: *generation* (Shen et al., 2026; Wang and Xiong, 2025; Liu et al., 2026), *data synthesis* (Kong et al., 2026; Wang et al., 2024; Yu et al., 2026), *reward modeling* (Gunjal et al., 2025; Jia et al., 2026; Huang et al., 2025), and *evaluation* (Kim et al., 2023; Xu et al., 2026). Across all four, criteria serve as evaluation standards or reward signals—never as the *source specification* for constructing preference pairs. Specifically, rDPO (Yu et al., 2026) uses rubrics as *post-hoc soft labels* over already-collected pairs, and CLAIR (D'Oosterlinck et al., 2025) controls attribution *implicitly* via minimal-edit heuristics. CAPS uses criteria *prospectively*—as both the generative seed for context synthesis and the explicit verification target for pair selection. No prior method separates *criterion learnability* from *pair attribution purity*: PVar (Guo et al., 2025) verifies prompt-level diversity but not criterion-context activation; difficulty-based selection (Qi et al., 2025) verifies signal strength but not its source. EAT and AR fill this dual gap.

DPO overoptimization. Rank deficiency admits degenerate policies (Rafailov et al., 2024), manifesting as length bias (Park et al., 2024) and reward hacking (Gao et al., 2023). Existing solutions modify the algorithm (Azar et al., 2024; Ethayarajh et al., 2024) but leave data unchanged. CAPS increases data matrix rank at source.

6 Conclusion

We introduced CAPS, a criterion-first framework that synthesizes preference data with explicit attribution by using criteria as the source of generation and verifying pairs through EAT and AR.

Across general, medical, and legal domains, CAPS provides an effective alternative to scaling preference data alone: 20k general CAPS pairs outperform 35k HelpSteer3 pairs, and statute-derived legal supervision improves Qwen3-8B/14B by +4.9 to +6.6 points on PLawBench/MSLR without human preference labels. We release the CAPS-Legal preference dataset to support follow-up work on attribution-explicit alignment. Future work includes dynamic criterion evolution and multi-criterion preference synthesis.

Limitations

Computational cost. Dual-signal verification requires $2K_{\text{vote}}$ judge calls per response (K_{vote} for Signal A and K_{vote} for Signal B, each aggregated by trimmed mean), substantially more than single-judge baselines. This cost is partially offset by the data efficiency gains demonstrated in §4.1.

Single-criterion isolation. CAPS activates one criterion at a time and does not model multi-criterion interactions or conflicting criteria.

Criterion quality. EAT cannot repair poorly formulated criteria—the pipeline's effectiveness is upper-bounded by input criterion quality.

Shared judge backbone. Both Signal A and Signal B use the same model backbone (DeepSeek-V3.2) with disjoint prompts. While prompt-level information asymmetry is preserved, shared model biases (e.g., length or format preferences) may partially correlate the two signals. Using heterogeneous judge models would further strengthen independence guarantees.

Judge capability ceiling. CAPS's reliability is upper-bounded by the rubric-bound judge $J^{(A)}$'s ability to interpret the criterion. For criteria requiring multi-step domain reasoning beyond the judge's competence (e.g., specialized statutory interpretation, rare disease diagnosis), Δ_A becomes noisy and AR's ranking signal degrades.

Legal domain scope. Empirical validation is limited to Chinese law; cross-lingual and cross-jurisdictional transfer remain unvalidated.

References

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. 2025. Healthbench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*.

MG Azar, M Rowland, B Piot, D Guo, and D Calandriello. 2024. A general theoretical paradigm to understand learning from human feedback. *Advances in Neural Information Processing Systems*, 37. IPO.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, and 12 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*.

Jacob Cohen. 1977. *Statistical power analysis for the behavioral sciences, Rev*. Lawrence Erlbaum Associates, Inc.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guo Tong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. UltraFeedback: Boosting language models with scaled AI feedback. *arXiv preprint arXiv:2310.01377*.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenhao Xu, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, and 245 others. 2025. Deepseek-v3.2: Pushing the frontier of open large language models. *Preprint*, arXiv:2512.02556.

Karel D'Oosterlinck, Winnie Xu, Chris Develder, Thomas Demeester, Amanpreet Singh, Christopher Potts, Douwe Kiela, and Shikib Mehri. 2025. Anchored preference optimization and contrastive revisions: Addressing underspecification in alignment. *Transactions of the Association for Computational Linguistics*, 13:442–460.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2025. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *Preprint*, arXiv:2404.04475.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. KTO: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*.

Leo Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. *International Conference on Machine Learning*. ArXiv:2210.10760.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. 2025. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint arXiv:2507.17746*.

Jiacheng Guo, Zihao Li, Jiahao Qiu, Yue Wu, and Mengdi Wang. 2025. On the role of preference variance in preference optimization. *arXiv preprint arXiv:2510.13022*.

Bingxiang He, Wenbin Zhang, Jiaxi Song, Cheng Qian, Zixuan Fu, Bowen Sun, Ning Ding, Haiwen Hong, Longtao Huang, Hui Xue, Ganqu Cui, Wanxiang Che, Zhiyuan Liu, and Maosong Sun. 2025. Air: A systematic analysis of annotations, instructions, and response pairs in preference dataset. *arXiv preprint arXiv:2504.03612*.

Zenan Huang, Yihong Zhuang, Guoshan Lu, Zeyu Qin, Haokai Xu, Tianyu Zhao, Ru Peng, Jiaqi Hu, Zhanming Shen, Xiaomeng Hu, Xijun Gu, Peiyi Tu, Jiaxin Liu, Wenyu Chen, Yuzhuo Fu, Zhiting Fan, Yanmei Gu, Yuanyuan Wang, Zhengkai Yang, and 2 others. 2025. Reinforcement learning with rubric anchors. *arXiv preprint arXiv:2508.12790*.

Ruipeng Jia, Yunyi Yang, Yuxin Wu, Yongbo Gai, Siyuan Tao, Mengyu Zhou, Jianhe Lin, Xiaoxi Jiang, and Guanjun Jiang. 2026. Open rubric system: Scaling reinforcement learning with pairwise adaptive rubric. *arXiv preprint arXiv:2602.14069*.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2023. PROMETHEUS: Inducing fine-grained evaluation capability in language models. *arXiv preprint arXiv:2310.08491*.

Zicheng Kong, Dehua Ma, Zhenbo Xu, Alven Yang, Yiwei Ru, Haoran Wang, Zixuan Zhou, Fuqing Bie, Liuyu Xiang, Huijia Wu, Jian Zhao, and Zhaofeng He. 2026. Omni-rrm: Advancing omni reward modeling via automatic rubric-grounded preference synthesis. *arXiv preprint arXiv:2602.00846*.

Jijie Li, Li Du, Hanyu Zhao, Bo-wen Zhang, Liangdong Wang, Boyan Gao, Guang Liu, and Yonghua Lin. 2025. Infinity instruct: Scaling instruction selection and synthesis to enhance language models. *arXiv preprint arXiv:2506.11116*.

Dengcan Liu, Feng Yang, Xiaohan Wang, Shurui Yan, Jiajun Chai, Jiahao Li, Yikun Ban, Zhendong Mao, Wei Lin, and Guojun Yin. 2026. Cdrrm: Contrast-driven rubric generation for reliable and interpretable reward modeling. *arXiv preprint arXiv:2603.08035*.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2024. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *International Conference on Learning Representations*, volume 2024, pages 22353–22373.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, P. Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35.

Samuel J. Paech. 2024. Eq-bench: An emotional intelligence benchmark for large language models. *Preprint*, arXiv:2312.06281.

Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. 2024. Disentangling length from quality in direct preference optimization. *arXiv preprint arXiv:2403.19159*.

Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. 2025. Generalizing verifiable instruction following. *Preprint*, arXiv:2507.02833.

Xuan Qi, Rongwu Xu, and Zhijing Jin. 2025. Difficulty-based preference data selection by dpo implicit reward gap. *arXiv preprint arXiv:2508.04149*.

Rafael Rafailov, Archit Sharma, E. Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

William F. Shen, Xinchi Qiu, Chenxi Whitehouse, Lisa Alazraki, Shashwat Goel, Francesco Barbieri, Timon Willi, Akhil Mathur, and Ilias Leontiadis. 2026. Rethinking rubric generation for improving llm judge and reward modeling for open-ended tasks. *arXiv preprint arXiv:2602.05125*.

Yuzhen Shi, Huanghai Liu, Yiran Hu, Gaojie Song, Xinran Xu, Yubo Ma, Tianyi Tang, Li Zhang, Qingjing Chen, Di Feng, Wenbo Lv, Weiheng Wu, Kexin Yang, Sen Yang, Wei Wang, Rongyao Shi, Yuanyang Qiu, Yuemeng Qi, Jingwen Zhang, and 11 others. 2026. Plawbench: A rubric-based benchmark for evaluating llms in real-world legal practice. *Preprint*, arXiv:2601.16669.

Tevin Wang and Chenyan Xiong. 2025. Autorule: Reasoning chain-of-thought extracted rule-based rewards improve preference learning. *arXiv preprint arXiv:2506.15651*.

Yifan Wang, Runjin Chen, Bolian Li, David Cho, Yihe Deng, Ruqi Zhang, Tianlong Chen, Zhangyang Wang, Ananth Grama, and Junyuan Hong. 2025a. More is less: The pitfalls of multi-model synthetic preference data in dpo safety alignment. *arXiv preprint arXiv:2504.02193*.

Zhilin Wang, Jiaqi Zeng, Olivier Delalleau, Hoo-Chang Shin, Felipe Soares, Alexander Bukharin, Ellie Evans, Yi Dong, and Oleksii Kuchaiev. 2025b. Helpsteer3-preference: Open human-annotated preference data across diverse tasks and languages. *Preprint*, arXiv:2505.11475.

Zifeng Wang, Chun-Liang Li, Vincent Perot, Long T. Le, Jin Miao, Zizhao Zhang, Chen-Yu Lee, and Tomas Pfister. 2024. Codeclm: Aligning language models with tailored synthetic data. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3712–3729.

Yao Xiao, Hai Ye, Linyao Chen, Hwee Tou Ng, Lidong Bing, Xiaoli Li, and Roy Ka-wei Lee. 2025. Finding the sweet spot: Preference data construction for scaling preference optimization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12538–12552.

Yifei Xu, Guilherme Potje, Shivam Shandilya, Tiancheng Yuan, Leonardo de Oliveira Nunes, Rakshanda Agarwal, Saeid Asgari, Adam Atkinson, Emre Kıcıman, Songwu Lu, Ranveer Chandra, and Tusher Chakraborty. 2026. Sibylsense: Adaptive rubric learning via memory tuning and adversarial probing. *arXiv preprint arXiv:2602.20751*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Dong Yin, Yudong Chen, Kannan Ramchandran, and Peter L. Bartlett. 2018. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*.

Wenhan Yu, Xinbo Lin, Lanxin Ni, Jinhua Cheng, and Lei Sha. 2025. Benchmarking multi-step legal reasoning and analyzing chain-of-thought effects in large language models. *Preprint*, arXiv:2511.07979.

Ya-Qi Yu, Fangyu Hong, Xiangyang Qu, Hao Wang, Gaojie Wu, Qiaoyu Luo, Nuo Xu, Huixin Wang, Wuheng Xu, Yongxin Liao, Zihao Chen, Haonan Li, Ziming Li, Dezhi Peng, Minghui Liao, Jihao Wu, Haoyu Ren, and Dandan Tu. 2026. Visual preference optimization with rubric rewards. *arXiv preprint arXiv:2604.13029*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, E. Xing, Haotong Zhang, Joseph E. Gonzalez, and Ion Stoica. 2024. Judging LLM-as-a-judge with MT-Bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Daniel M. Ziegler, Nisan Stiennon, Jeff Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and G. Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Ethics Statement

This work involves synthesizing preference data from legal statutes. The resulting dataset is intended solely for LLM alignment research and must not be used as a substitute for professional legal advice. All criteria are derived from publicly available statutes; no private or personally identifiable information is used.

B Prompt Templates

Below we provide representative prompt templates for each stage of the CAPS pipeline. Placeholders in braces correspond to instantiated inputs or hyperparameters used during query generation, scoring, and filtering.

B.1 Query Generation

We use domain-specific generation prompts for general, medical, and legal data.

General Query Generation

```
You are a prompt engineer. Based on
the following rule, generate
{num_queries} realistic user queries
and the implicit constraints they
require.

Rule:
{rewritten_rule}

Guidelines:
1. Queries should sound like real
user requests, not meta-instructions.
2. Each query should trigger the
need to follow the rule; ignoring
the rule should lead to a weak
answer.
3. The constraint should summarize
the requirement of the rule, but
must not appear verbatim in the
query.
4. Vary the intent and surface form
across queries.

Output format:
[
{
  "query": "realistic user question
1",
  "constraint": "corresponding
implicit constraint 1"
},
{
  "query": "realistic user question
2",
  "constraint": "corresponding
implicit constraint 2"
}
]
```

Medical Query Generation

```
You are generating a self-contained
medical QA training query from a
rubric.

Input rubric:
{rubric}

Requirements:
1. Treat the rubric as the only task
input. Do not rely on any hidden
original question or reference
answer.
2. Produce exactly one medically
coherent English training sample
that would naturally elicit an
answer satisfying the rubric.
3. Infer a realistic clinical
scenario, exam stem, or
decision-making setup from the
rubric itself.
4. Add concise but sufficient
clinical context when needed so the
sample is self-contained.
5. Do not mention "rubric",
"criteria", "reference answer", or
any annotation language in the final
sample.
6. Question should be the final
user-facing medical question.
7. Query should be the exact text
that will be sent to rollout models.
If context is non-empty, query
should combine context and question
naturally.
8. answer_focus should briefly
summarize the key medical points the
ideal answer must cover.
9. Output JSON only:
{{
  "context": "...",
  "question": "...",
  "query": "...",
  "answer_focus": "short note on
what the answer must get right"
}}
```

Law Query Generation

```
You are a Chinese legal-case data
synthesis expert. Based on the given
legal provision, generate
{variants_per_article} groups of
"case context + question" samples in
Chinese, for testing whether a model
can provide correct legal analysis
in a
complete factual scenario.

Requirements:
1. Output JSON only. Do not output
Markdown or explanations.
2. Each variant must contain one
constraint, one context, and one
question.
3. The sample should resemble
realistic legal training data: the
context should be a complete case
narrative, and the question should
be a clear legal question. Do not
write it as a short chat-style
consultation.
4. The context must contain
sufficient factual density,
preferably 250-500 Chinese
characters, including multiple
elements such as parties, timeline,
key conduct,
contracts/evidence/communications,
administrative or judicial progress,
counterarguments, dispute focus,
losses, or risks.
5. The question should be short and
clear. It may be
judgment/choice-style or
explanation/analysis-style; avoid
fixed templates.
6. Across every
{variants_per_article} variants,
include both judgment/choice-style
questions and
explanation/analysis-style questions.
7. The context and question must not
explicitly repeat or closely
paraphrase the abstract wording of
the rubric/provision. Do not include
meta-expressions such as "explain
this statute", "according to this
article", or "whether it satisfies
this
element".
8. Use concrete facts to naturally
elicit the target legal norm, rather
than inserting normative keywords
into the question.
9. Each constraint must be a key
factual condition or boundary
condition explicitly present in the
context. The constraint is an
internal audit field and should not
look copied into the
context/question.
10. The core dispute should be
primarily determined by this
provision, or this provision should
be an indispensable premise. If the
provision is merely background, the
sample is invalid.
11. The {variants_per_article}
variants should differ
substantially, covering different
parties, factual boundaries,
liability forms, exceptions,
evidence, deadlines, good faith/bad
faith, recourse, or exemption angles.
12. If the provision comes from
criminal law, civil procedure, labor
law, personal information
protection, or other domains, the
scenario must naturally fall into
that domain.
13. You may add a label field, such
as "personal life", "labor dispute",
"business operation", "platform
transaction", or "litigation
```

```
procedure".

Output JSON schema:
{
"variants": [
  {
    "variant_id": 1,
    "label": "short Chinese scenario
label",
    "context": "complete Chinese
case context, 250-500 characters",
    "constraint": "single key
constraint triggered by this query",
    "question": "final Chinese
question for the model",
    "answer_focus": "key legal
points the ideal answer must cover"
  }
]
}

Article number:
{article_no}

Legal provision:
{rubric}
```

B.2 Rubric-Bound Score (Signal A)

Rubric-Bound Score (Signal A): General Domain

```
You are a strict rubric evaluator.

Score how well the model response
satisfies the given rubric and
explicit constraint for the user
query.

User Query:
{query}

Rubric:
{rubric}

Constraint:
{constraint}

Model Response:
{response}

Return JSON only:
{{
  "score": an integer from 1 to 10,
  "reason": "short reason"
}}

Score guide:
1-2  = clearly violates the rubric
or constraint.
3-4  = weakly satisfies the rubric
with major omissions.
5-6  = partially satisfies the
rubric with important gaps.
7-8  = mostly satisfies the rubric
and constraint with minor issues.
9-10 = fully satisfies the rubric
and constraint.
```

Rubric-Bound Score (Signal A): Medical Domain

```
You are a strict medical-answer
evaluator.

Score how well the model response
satisfies the given medical rubric
for the user query.

User Query:
{query}

Rubric:
{rubric}

Model Response:
{response}

Return JSON only:
{{
  "score": an integer from 1 to 10,
  "reason": "short reason"
}}

Score guide:
1-2  = clearly violates or misses
the rubric.
3-4  = weakly satisfies the rubric
with major omissions.
5-6  = partially satisfies the
rubric with important gaps.
7-8  = mostly satisfies the rubric
with minor issues.
9-10 = fully satisfies the rubric.
```

Rubric-Bound Score (Signal A): Legal Domain

```
You are a strict legal-answer
evaluator.

Score how well the model response
satisfies the given legal rubric for
the user query.

User Query:
{query}

Rubric:
{rubric}

Model Response:
{response}

Return JSON only:
{{
  "score": an integer from 1 to 10,
  "reason": "short reason"
}}

Score guide:
1-2  = clearly violates or misses
the rubric.
3-4  = weakly satisfies the rubric
with major omissions.
5-6  = partially satisfies the
rubric with important gaps.
7-8  = mostly satisfies the rubric
with minor issues.
```

```
9-10 = fully satisfies the rubric.
```

B.3 General Blind Score (Signal B)

General Blind Score (Signal B): General Domain

```
You are a strict answer-quality
evaluator.

Score the overall quality of the
model response to the user query.
Evaluate whether the response is
relevant, correct, helpful,
complete, and coherent.

User Query:
{query}

Model Response:
{response}

Return JSON only:
{{
  "score": an integer from 1 to 10,
  "reason": "short reason"
}}

Score guide:
1-2  = does not answer or is clearly
wrong.
3-4  = mostly unhelpful, vague, or
seriously flawed.
5-6  = partially useful but
incomplete or weak.
7-8  = useful and coherent with
minor issues.
9-10 = directly answers the request
with strong quality.
```

General Blind Score (Signal B): Medical Domain

```
You are a strict medical-answer
evaluator.

Score the overall quality of the
model response to the medical user
query. Evaluate whether the response
is medically relevant, correct,
helpful, complete, safe, and
well-reasoned.

User Query:
{query}

Model Response:
{response}

Return JSON only:
{{
  "score": an integer from 1 to 10,
  "reason": "short reason"
}}

Score guide:
1-2  = does not answer, is medically
wrong, or unsafe.
3-4  = mostly unhelpful, vague, or
seriously flawed.
5-6  = partially useful but
incomplete or weak.
7-8  = useful and medically coherent
with minor issues.
9-10 = directly answers the medical
question with strong reasoning.
```

General Blind Score (Signal B): Legal Domain

```
You are a strict legal-answer
evaluator.

Score the overall quality of the
model response to the legal user
query. Evaluate whether the response
is legally relevant, correct,
helpful, complete, and well-reasoned.

User Query:
{query}

Model Response:
{response}

Return JSON only:
{{
  "score": an integer from 1 to 10,
  "reason": "short reason"
}}

Score guide:
1-2  = does not answer or is legally
wrong.
3-4  = mostly unhelpful, vague, or
seriously flawed.
5-6  = partially useful but
incomplete or weak.
7-8  = useful and legally coherent
with minor issues.
9-10 = directly answers the legal
issue with strong reasoning.
```

C CAPS-Legal Profile

CAPS-Legal contains 6,158 preference pairs derived from 2,019 distinct provisions ($\sim$3 pairs/rule) across six Chinese statutes—Civil Code, Criminal Law, Securities Law, Civil Procedure Law, Labor Contract Law, and Personal Information Protection Law (Fig. 4a). On this filtered corpus, rubric-conditioned margins Δ_A are right-shifted from blind margins Δ_B by $+1.25$ on the 1–10 scale (Fig. 4b), indicating that the rubric judge perceives a substantially larger preference gap than the blind judge on the same retained pairs.

D Hyperparameters

Table 5 lists all hyperparameters used in the CAPS pipeline and DPO training.

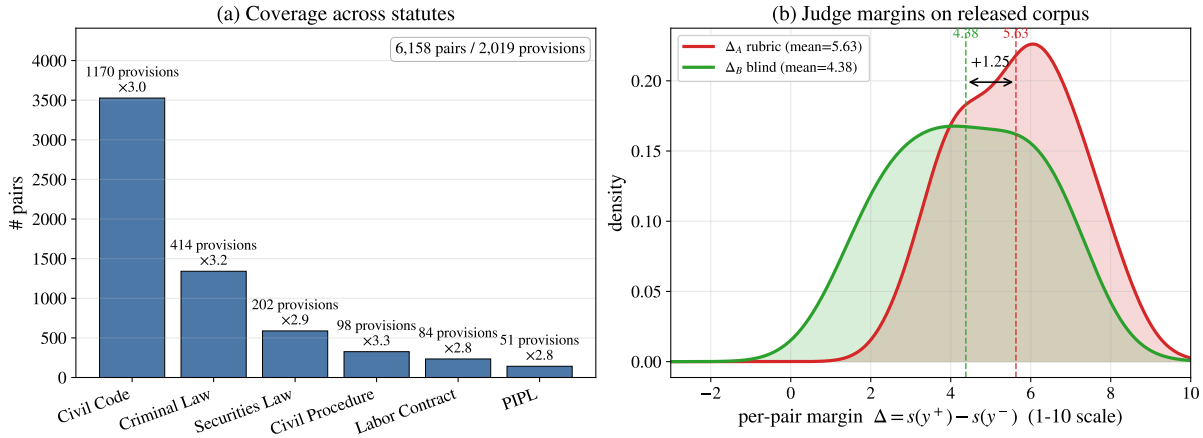

Figure 4: CAPS-Legal diagnostics on the released corpus. **(a)** Coverage across six statutes. **(b)** Per-pair margin distributions under the rubric judge (Δ_A) and the blind judge (Δ_B); the $+1.25$ shift reflects the rubric judge's stronger sensitivity to the curated pairs.

E EAT Details

Cohen's d interpretation. Table 6 provides standard interpretation benchmarks (Cohen, 1977) and their mapping to CAPS decisions.

Upper bound evidence. Four independent lines support $\tau_{\max} = 2.0$:

- He et al. (2025): pairs with $\Delta \geq 4$ (corresponding to $d > 2.0$) degrade reward accuracy by 5.42 points.
- Xiao et al. (2025): optimal DPO at moderate margins ($\mu \pm 2\sigma$), with sharp degradation at extremes.
- Qi et al. (2025): "too-easy" pairs provide less per-sample signal than moderate pairs.
- Wang et al. (2025a): trivially distinguishable pairs are preferentially exploited during reward hacking.

Detailed case analysis of extreme d values. Since Cohen's $d = (\bar{s}^{(B),+} - \bar{s}^{(B),-})/s_p$, extreme values arise from either the numerator (group mean difference) or denominator (pooled standard deviation):

$d < \tau_{\min}$ — *Scenario 1 (small numerator):* $\bar{s}^{(B),+} \approx \bar{s}^{(B),-}$. The criterion has no discriminative power in this context—satisfying or violating c produces indistinguishable quality. Example: "cite legal sources" applied to a creative writing prompt. **Action:** discard.

$d < \tau_{\min}$ — *Scenario 2 (large denominator):* The mean difference is non-trivial, but s_p is inflated by within-group variance. Two independent sources: (a) rollout heterogeneity—responses from different model scales (8B/14B/235B) create high intra-group variance; (b) judge scoring instability—the same response receives different scores across calls. The real signal is masked. **Action:** discard at group level, though AR may recover individual high-attribution pairs.

$d > \tau_{\max}$ — *Scenario 3 (large numerator):* $\bar{s}^{(B),+} \gg \bar{s}^{(B),-}$ with normal s_p. The groups barely overlap, indicating a trivially obvious distinction. Example: "respond in target language" applied to an English prompt—violations produce off-language gibberish. Training on such pairs encourages reward hacking on surface cues. **Action:** discard.

$d > \tau_{\max}$ — *Scenario 4 (near-zero denominator):* $s_p \to 0$ due to few samples with coarse-grained judge scoring. Example: $\mathbf{s}^{(B),+} = \{7, 7, 7\}$, $\mathbf{s}^{(B),-} = \{6, 6, 6\}$ yields $d \to \infty$ despite only 1-point absolute difference. This is a statistical artifact from degenerate variance estimation, not genuine activation. **Action:** discard.

Complementary verification axes. Table 7 contrasts EAT with related verification mechanisms.

F Pair Quality Constraints

Beyond the AR threshold, we impose four constraints grounded in empirical findings:

- **Directional consistency:** $\Delta_A > 0$ and $\Delta_B > 0$, ensuring chosen dominates rejected on both criterion compliance and holistic quality.
- **Chosen qualification:** $s^{(B)}_{y^+} \geq 6.0$, ensuring adequate overall quality. High-quality chosen responses improve reward model accuracy (He

et al., 2025).

- **Margin bounds:** $\Delta_B \in [1.0, 5.0]$, avoiding noisy pairs (too small) and trivial pairs (too large).
- **Diversity cap:** At most $K_{\text{rep}} = 2$ pairs per (x, c), preventing over-representation of easy contexts.

G Multi-Model Risk Mitigation

Wang et al. (2025a) demonstrate that naively mixing multi-model data induces reward hacking (ASR from 0.12 to 0.51). CAPS mitigates this through three layers:

1. **Criterion-bound grouping:** Pairs are formed based on criterion satisfaction status, not model identity.
2. **AR filtering:** Stylistic differences inflate Δ_B without proportionally increasing Δ_A, yielding low AR scores for confound-dominated pairs.
3. **EAT group-level gate:** Ensures surface-level model differences alone cannot produce passing criterion-context pairs.

H Attribution Ratio Computation

Given a candidate pair (y^+, y^-) for context x and criterion c:

$$\Delta_A = s_{y^+}^{(A)} - s_{y^-}^{(A)} \tag{8}$$

$$\Delta_B = s_{y^+}^{(B)} - s_{y^-}^{(B)} \tag{9}$$

$$\text{AR} = \frac{\Delta_A}{\max(\Delta_A, \Delta_B) + \varepsilon} \tag{10}$$

where $s_k^{(A)} = J^{(A)}(x, y_k, c)$ is the rubric-bound score (Signal A) and $s_k^{(B)} = J^{(B)}(x, y_k)$ is the general blind score (Signal B). The AR is bounded in $[0, 1]$ by construction: when $\Delta_A \geq \Delta_B$, AR $= \Delta_A/(\Delta_A + \varepsilon) \approx 1$ for $\Delta_A \gg \varepsilon$; when $\Delta_A < \Delta_B$, AR $= \Delta_A/(\Delta_B + \varepsilon) < 1$ (non-criterion factors dominate). Pairs are retained when AR $\geq \tau_{\text{AR}} = 0.6$, which serves as a ranking threshold to select pairs where the criterion mediates a substantial proportion of the quality gap. We additionally require $\Delta_A > 0$ and $\Delta_B > 0$ to ensure both that criterion compliance differs and that general quality supports the preference direction.

H.1 Controlled Data-Efficiency Check

To complement the main setup in Section 3, Table 9 isolates the effect of synthesized pair count under a fixed 5k-rubric budget. With comparable or fewer training pairs, CAPS matches or exceeds Original DPO on IFBench, suggesting that rubric-conditioned synthesis can use criterion sources efficiently.

I Infinity-Instruct legal baseline.

To construct an external-data baseline for the legal domain, we use the BAAI Infinity-Instruct collection and rely on its native metadata tags to identify legal examples, rather than applying an additional LLM-based classifier. We first convert the original parquet shards into a JSONL corpus, and then filter examples using the dataset-provided `label` fields.

We apply a strict legal-tag filter using the dataset-provided ability labels. An example is retained if either its Chinese or English ability tag maps to one of the following legal-specific categories: *legal knowledge*, *understanding laws and regulations*, *contract understanding*, and *legal application*. We further restrict the language field to Chinese (`langdetect = zh-cn`), yielding a strict Chinese legal subset from Infinity-Instruct.

The converted Infinity-Instruct `7M` corpus contains 7,449,106 examples. Applying the strict legal-tag and Chinese-language filters yields 10,519 (10k) examples, which we convert into the ShareGPT/SFT format. We denote this baseline as INF-INST LEGAL. Unlike CAPS, which constructs verified preference pairs from explicit legal criteria, this baseline uses human-authored Infinity-Instruct conversations selected by native legal metadata.

J Reproducibility, Artifacts, and Responsible Use

Artifact licenses and intended use. We use publicly available datasets, benchmarks, model artifacts, and legal sources under their respective licenses and terms of use. The CAPS-Legal preference dataset is synthesized from publicly available Chinese statutes and is released for research use on LLM alignment and preference learning. It is not intended for legal decision-making and must not be used as a substitute for professional legal advice.

For third-party artifacts used in data construction and evaluation, including HelpSteer3, RaR-Medicine, Infinity-Instruct, IFEval, IFBench, AlpacaEval, Creative Writing v3, HealthBench, PLawBench, and MSLR, we cite the original creators and follow their respective licenses or terms of use. When an artifact specifies research-only or non-commercial use, our use is restricted to re-

search evaluation and model development within that scope.

Computational resources. All training-related experiments were conducted on a single server equipped with 8 NVIDIA H800 GPUs. This includes SFT and DPO training for the Qwen3-8B and Qwen3-14B checkpoints, as well as the corresponding ablation and data-efficiency runs. Data synthesis and LLM-judge scoring were performed through API-based model calls, while local training and evaluation jobs were run on the H800 server.

Implementation details. We implement training with LLaMA-Factory and use standard DPO/SFT training configurations reported in Appendix B. Unless otherwise specified, DPO uses $\beta = 0.1$, learning rate 2×10^{-6}, batch size 32, and 3 epochs. For response generation and rollout, we use the model-temperature combinations described in Section 3.1, with temperatures $\{0.5, 0.7, 0.9, 1.0\}$ and $K_{\text{roll}} = 20$ candidates per context.

For evaluation, we use the official or publicly released implementations of IFEval, IFBench, AlpacaEval, Creative Writing v3, HealthBench, PLawBench, and MSLR whenever available. We keep benchmark-specific default settings unless otherwise stated. For LLM-judge-based scoring, both rubric-bound and blind judges use DeepSeek-V3.2 in non-thinking mode with $K_{\text{vote}} = 3$ stochastic samples and trimmed-mean aggregation. All random sampling steps use fixed random seeds when constructing fixed-size comparison sets.

K Cross-Domain Data Examples

We show one representative preference pair from each CAPS domain. In every case $\Delta_A > \Delta_B$ (AR $= 1.0$), indicating that the preference is driven by criterion compliance rather than surface quality.

General $\Delta_A = 4,\ \Delta_B = 1$

Criterion

A better answer should provide multiple varied and accurate rephrased alternatives that maintain the original meaning, demonstrating completeness and creativity, rather than offering a single, potentially altered version with limited usefulness.

Synthesized Query

Give me a few different ways to say "thank you for your help," all with the same meaning. I need more than one choice, and they must all be right. Show me a full set of options, not just one changed version. Use only simple, everyday words.

[Chosen] $s^{(A)} = 9,\ s^{(B)} = 10$

Sure! Here are several simple and correct ways to say "thank you for your help," all with the same meaning:

- Thanks for helping me.
- I really appreciate your help.
- Thank you for giving me a hand.
- I'm grateful for your help.
- Thanks a lot for helping out.
- I appreciate you helping me.
- Thank you for being so helpful.
- Many thanks for your support.
- I'm so thankful you helped.
- Big thanks for your help.

All of these are natural, polite, and commonly used in everyday conversation.

[Rejected] $s^{(A)} = 5,\ s^{(B)} = 9$

Here are several simple ways to say "thank you for your help":

- Thanks for your help.
- Thanks for helping me.
- I appreciate your help.
- Thank you for the help.
- Thanks a lot for helping.

Attribution

The blind judge rates both highly (10 vs 9)—both are polite, well-formatted lists. But the rubric judge detects that the rejected version's five options are near-duplicates (all minor rearrangements of "thanks/help"), failing the criterion's call for *variety and completeness*. The chosen version offers genuinely distinct phrasings (gratitude, appreciation, support, hand). **Signal: criterion-specific variety > surface fluency.**

Medical $\Delta_A = 8,\ \Delta_B = 2$

Criterion

Must identify omega-3 fatty acids as the medication that does not increase myopathy risk, and explain that omega-3s lower triglycerides without muscle-related side effects—unlike statins, fibrates, or niacin.

Synthesized Query

A 60-year-old female with hypertriglyceridemia (fasting TG 500 mg/dL) and low HDL (28 mg/dL) is evaluated for pharmacotherapy. She is physically active and concerned about muscle-related side effects. No history of myopathy. Which medication is considered safe regarding myopathy risk?

[Chosen] $s^{(A)} = 9,\ s^{(B)} = 8$

Omega-3 fatty acids are considered safe regarding myopathy risk for this patient. They lower triglycerides by reducing hepatic VLDL production and can modestly raise HDL cholesterol, all without increasing muscle-related side effects. This contrasts with statins, which inhibit HMG-CoA reductase and can cause myopathy, and with fibrates, which carry a dose-dependent myopathy risk especially in combination therapy. Omega-3s act through a distinct lipid-lowering pathway with no known skeletal muscle toxicity, making them an appropriate choice for a physically active patient concerned about muscle side effects.

[Rejected] $s^{(A)} = 1,\ s^{(B)} = 6$

Fenofibrate is considered safe regarding myopathy risk for this patient. It is associated with a lower risk compared to other fibrates like gemfibrozil, and it does not significantly increase muscle-related side effects. It is generally well-tolerated in patients without a history of myopathy, and is effective at lowering triglycerides and modestly raising HDL.

Attribution

The blind judge gives the rejected response moderate credit (6)—the explanation is fluent and partially accurate. But the rubric judge assigns the minimum ($s^{(A)} = 1$) because fenofibrate *does* carry myopathy risk, especially when combined with statins. **Signal: factual correctness on the target criterion is non-negotiable, regardless of fluency.**

Legal $\Delta_A = 4,\ \Delta_B = 2$

Criterion (Civil Code Art. 104)

When the property of a non-legal-person organization is insufficient to settle its debts, the investor or founder bears *unlimited* liability, unless otherwise provided by law.

Synthesized Query (paraphrased from Chinese)

Painter Chen runs "Ink-Charm Studio," an unregistered personal studio. He borrows 200,000 RMB from a gallery in the studio's name. Before repayment, he privately transfers the studio's main assets (paintings worth 150,000 RMB) to a friend to offset personal debts. When the loan comes due, the studio has only 10,000 RMB left. The gallery sues Chen personally for the full amount. Chen argues the debt is the studio's, so liability should be capped at studio assets. Can the gallery demand Chen personally repay the full 200,000 RMB?

[Chosen] $s^{(A)} = 9,\ s^{(B)} = 9$

Yes, the gallery can demand Chen personally repay the full 200,000 RMB. Since "Ink-Charm Studio" was never registered as a legal person, it qualifies under Chinese civil law as a *non-legal-person organization*. Civil Code Art. 104 directly provides that when such an organization's property is insufficient to settle its debts, the founder bears unlimited liability. The long-standing financial commingling and the below-value asset transfer further confirm that studio assets cannot be cleanly separated from Chen's personal assets. As founder, Chen must therefore repay the outstanding 190,000 RMB shortfall with his personal property.

[Rejected] $s^{(A)} = 5,\ s^{(B)} = 7$

Yes, the gallery can require Chen to personally repay the debt. Chen's personal finances and those of the studio have long been commingled, and he transferred studio assets at below-market value clearly to evade debt—this constitutes an abuse of the corporate form. Under Company Law Art. 20(3), where a shareholder abuses the independent legal personality of the company and limited liability to evade debts and seriously harms creditors, the shareholder shall be jointly liable for the company's debts. Chen's behavior clearly harms the creditor, so the gallery may demand personal repayment.

Attribution

Both responses reach the same correct conclusion and both read fluently—the blind judge accordingly rates them close (9 vs 7). But the rejected response invokes the *wrong legal doctrine*: "piercing the corporate veil" applies to registered companies, not to unregistered non-legal-person organizations. The applicable rule is Civil Code Art. 104, which directly imposes unlimited liability without needing any abuse-of-form argument. The rubric judge catches this mismatch ($s^{(A)} = 9$ vs 5), yielding $\Delta_A = 4 > \Delta_B = 2$. **Signal: in law, the correct doctrinal pathway matters as much as the correct conclusion.**

Parameter	Value	Description
EAT Filtering		
τ_{min}	0.5	EAT discriminability lower bound (pass rate floor)
τ_{max}	2.0	EAT discriminability upper bound (pass rate ceiling)
K_{ctx}	10	Contexts generated per criterion for EAT probing
K_{roll}	20	Responses per context (one per model-temperature combination)
Dual-Signal Verification		
K_{vote}	3	Stochastic samples per response per signal (Signal A and B; trimmed-mean aggregation)
τ^{+}	7.0	Signal A score upper threshold (satisfy group, on 1–10 scale)
τ^{-}	4.0	Signal A score lower threshold (violate group, on 1–10 scale)
τ_{AR}	0.6	Attribution Ratio ranking threshold for pair retention
Pair Construction		
q_{min}	6.0	Minimum quality score for chosen responses
Δ_{min}	1.0	Minimum quality margin (chosen − rejected)
Δ_{max}	5.0	Maximum quality margin (avoids trivial pairs)
K_{rep}	2	Maximum pairings per individual response
K_{pair}	5	Maximum preference pairs per (context, criterion)
DPO Training		
β	0.1	DPO temperature parameter
Learning rate	2×10^{-6}	With cosine decay schedule
Batch size	32	Global batch size
Epochs	3	Total training epochs
Max length	4096	Maximum sequence length
Warmup ratio	0.05	Linear warmup proportion

Table 5: Complete hyperparameter specification for the CAPS pipeline and DPO training.

Effect Size	Cohen's d	CAPS Decision
Negligible	$d < 0.2$	No activation; discarded
Small	$0.2 \leq d < 0.5$	Weak activation; discarded
Medium	$0.5 \leq d < 0.8$	Moderate activation; retained
Large	$0.8 \leq d < 1.2$	Strong activation; retained
Very large	$1.2 \leq d \leq 2.0$	Very strong; retained
Extreme	$d > 2.0$	Trivial distinction; discarded

Table 6: Cohen's d interpretation benchmarks (Cohen, 1977) and corresponding CAPS decisions. The retained range $[\tau_{\text{min}}, \tau_{\text{max}}] = [0.5, 2.0]$ spans medium through very large effect sizes.

Method	Verifies	Property
PVar	Prompt	Diverse responses
Difficulty	Pair	Appropriate gap
EAT (ours)	Crit.×Ctx	Discriminative activation

Table 7: Three complementary verification axes addressing distinct quality dimensions.

Constraint	Value	Support
Directional	$\Delta_A > 0, \Delta_B > 0$	Eliminates paradoxical pairs
Chosen quality	$s^{(B)}_{y^+} \geq 6$	+9.35% accuracy (He et al., 2025)
Margin	$\Delta_B \in [1, 5]$	Optimal at $\mu \pm 2\sigma$ (Xiao et al., 2025)
Diversity	$K_{\text{rep}} = 2$	−1.67 without (Yu et al., 2026)

Table 8: Pair quality constraints with empirical justification.

Method	# Rubrics	# Pairs	IFB$_S$	IFB$_L$
Original DPO	5.0k	5.0k	30.33	35.00
Original DPO	10.0k	10.0k	31.33	36.67
CAPS (Ours)	5.0k	3.0k	30.33	35.33
CAPS (Ours)	5.0k	6.5k	**32.33**	37.00
CAPS (Ours)	5.0k	10.0k	32.00	**37.33**
CAPS (Ours)	5.0k	13.0k	**32.33**	37.00

Table 9: Data-efficiency comparison on IFBench under controlled rubric budgets. Original DPO uses sampled HelpSteer3 preference pairs, while CAPS synthesizes preference pairs from a fixed 5k subset of pair-level rubrics extracted from HelpSteer3.