WHEN POSITION PREDICTS REWARD IN LLM JUDGES

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM judges can learn a position rule instead of comparing response quality when preference data always place the preferred response in the same slot. We study this training-interface confound as an identification problem: a content-sensitive judge should reverse its verdict when the two responses are exchanged, whereas a position shortcut should not. We introduce a paired-order audit and train matched judges with either a fixed preferred-first interface or both physical arrangements. In a controlled four-arm study with Qwen2.5-7B, four seeds, and 200 update steps, two-order pointwise training raises order-averaged accuracy from 65.14% to 87.05% on RewardBench and from 55.51% to 75.31% on an UltraFeedback clean split. First-position selection falls from 83.30% to 49.73% and from 93.12% to 56.18%, respectively. A separate dose study shows that increasing the preferred-first share monotonically reduces paired robustness. The results establish a simple audit and training intervention for testing whether an apparent judge gain survives a change to the feature that predicts its reward.

1 INTRODUCTION

An LLM judge is often trained on a pair consisting of a preferred answer and a rejected answer. A silent conversion choice can place the preferred answer in slot A for every example. Under that interface, the gold label is predictable from the slot alone: an always-A policy earns the correct reward without comparing content. Accuracy on identically formatted test prompts cannot identify which rule the model learned. This is a concrete instance of shortcut learning, in which a feature that is predictive in the observed sample replaces the intended evidence (Geirhos et al., 2020; McCoy et al., 2019; Tu et al., 2020). It also matters for reward optimization: increasing the number of examples with the same layout can make the shortcut easier to exploit rather than less plausible (Gao et al., 2023; Skalse et al., 2025).

We study this failure as an identification problem. For each pair, we render the original order and a swapped order while preserving the two responses and the label semantics. A content-sensitive judge should swap its verdict; a position policy should repeat it. We then train with either the fixed interface or an intervention that includes both orders. The intervention changes the suspected shortcut while leaving the underlying comparison task unchanged.

Our main result is deliberately narrow. In a four-arm, seed-matched experiment, two-order pointwise training raises order-averaged accuracy by 21.91 points on RewardBench and by 19.79 points on an UltraFeedback clean split. The same intervention lowers first-position selection by more than 33 points in both domains. A 50/80/95/100% preferred-first dose curve reproduces the monotonic relationship between first-position selection and paired robustness. Consistency rewards and paired-advantage terms provide no additional improvement once both orders are present.

Relation to prior audits. Position effects in LLM judges have been documented in evaluation studies (Wang et al., 2024; Shi et al., 2025; Zheng et al., 2023). Our question is upstream of an inference-time bias: if the training conversion itself makes position predictive of the label, then a position-sensitive policy can be rewarded during optimization. The response-swap test is a familiar way to expose evaluator sensitivity, but our use of it is as a paired intervention on the training interface. We do not propose a new judge architecture, a new preference objective, or a new benchmark. Instead, we ask whether a reported gain is identifiable under a simple change to the presentation that carries the reward.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

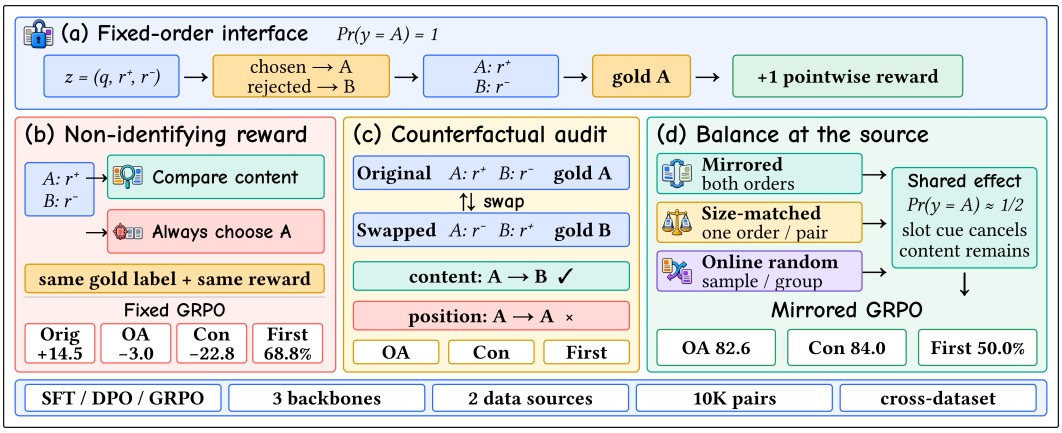

Figure 1: Balancing response layouts removes a reward-predictive position shortcut. Fixed-order training makes slot and label inseparable; our swap-and-flip intervention exposes the counterfactual layout and improves paired judging.

Our contributions are:

1. We formulate fixed-order preference training as an identification problem and give paired metrics that separate order-averaged accuracy, swap consistency, and first-position selection.
2. We quantify the intervention in a four-arm, seed-matched study: two-order pointwise training improves order-averaged accuracy by 21.91 points on RewardBench and 19.79 points on UF-clean, while reducing first-position selection by 33.57 and 36.94 points.
3. We provide a dose study and reproducible item-level summaries showing that the preferred-first share controls the same accuracy–position trade-off across four seeds.

2 Related Work

LLM-as-a-judge systems are now used for model comparison and open-ended evaluation (Zheng et al., 2023; Chiang et al., 2024). Their outputs can vary with response position, style, length, and prompt framing (Wang et al., 2024; Wu & Aji, 2025; Park et al., 2024; Singhal et al., 2024). Existing work often mitigates these effects by evaluating both orders, averaging multiple judgments, or filtering unstable examples. Those choices improve measurement at test time, but they do not by themselves reveal what a post-trained judge learned from a fixed-order reward interface.

Preference optimization methods, including supervised preference modeling, DPO, and policy-gradient RL, rely on a mapping from a comparison prompt to a target verdict (Christiano et al., 2017; Rafailov et al., 2023; Ouyang et al., 2022). The mapping is usually described in terms of response quality, while the serialization of the two responses is treated as an implementation detail. Our result shows why that detail deserves an explicit audit. If the serialization is perfectly correlated with the target, the data do not identify whether the model uses content or position. The issue is related to, but distinct from, reward hacking: the reward label is not wrong, yet the interface supplies an easier sufficient statistic (Gao et al., 2023; Skalse et al., 2025).

Permutation-based debiasing has appeared in evaluator training and benchmark construction. These methods augment data with swapped responses or add a consistency term. We use the same family of transformations as a controlled intervention, but the scientific claim is different. We first establish the counterfactual comparison between a fixed interface and its two-order version, then check whether extra consistency machinery adds anything. This ordering separates two questions: whether training on both layouts changes judging behavior, and whether an additional reward term improves performance once layout coverage is shared. The two-order pointwise arm provides the common reference for the latter comparison.

3 METHOD

Let $z_i = (q_i, r_i^+, r_i^-)$ denote a preference pair. An orientation variable $o_i \in \{+1, -1\}$ determines the rendered prompt. For $o_i = +1$, the preferred response is placed in slot A and the gold verdict is A; for $o_i = -1$, the rejected response is placed in slot A and the gold verdict is B. The intended rule depends on the response content, while the shortcut rule depends only on the slot. Under a fixed-order dataset, both rules yield the same observed label on every training example.

For a policy f_θ, define $p_i^o = f_\theta(q_i, r_i^o, r_i^{-o})$ and let $\hat{y}_i^o$ be the selected slot. We report

$$A_o = \frac{1}{n} \sum_i \mathbf{1}[\hat{y}_i^o = y_i^o], \tag{1}$$

$$A_{\mathrm{OA}} = \frac{1}{2}(A_{+1} + A_{-1}), \tag{2}$$

$$F = \frac{1}{2n} \sum_{i,o} \mathbf{1}[\hat{y}_i^o = \mathrm{A}], \tag{3}$$

$$C = \frac{1}{n} \sum_i \mathbf{1}[\hat{y}_i^{+1} \neq \hat{y}_i^{-1}]. \tag{4}$$

Here A_{OA} measures performance under both physical layouts, F is a mechanism readout, and C measures whether the verdict responds to the swap. A policy can have high A_{+1} and high F while having low A_{OA} and low C. The distinction is not cosmetic: only the paired quantities test the counterfactual layout.

Identification under one orientation. Under fixed preferred-first rendering, the training support contains only $(z_i, o = +1, y = A)$. On this support, a content rule and the constant-slot rule $h_A(x) = A$ make identical predictions and receive identical pointwise rewards. Consequently, any statistic computed only on that support has no power to distinguish these rules: the two hypotheses induce the same observed labels. Adding examples with the same orientation increases sample size but does not alter this equivalence. Including the swapped orientation changes the support to $(z_i, o = -1, y = B)$ as well; the constant-slot rule and a content-sensitive rule now disagree on the counterfactual examples. This is why the paired evaluation is an identification intervention, not merely a robustness score.

More formally, let H_c be a policy that predicts the preferred response and H_p a policy that always predicts slot A. For every fixed-order example, $H_c(x_i^{+1}) = H_p(x_i^{+1}) = A$. For the swapped example, $H_c(x_i^{-1}) = B$ while $H_p(x_i^{-1}) = A$. Thus the empirical risk under the fixed-order distribution cannot separate H_c and H_p, whereas the risk under the equal mixture of orientations can. The argument does not assume that real judges are deterministic or that all errors arise from position; it specifies when the training/evaluation interface fails to identify the intended rule.

The training intervention is a data transformation. A fixed-order batch contains $(z_i, +1)$, while a two-order batch contains both $(z_i, +1)$ and $(z_i, -1)$ with flipped labels. We keep the number of optimizer steps and the preference examples fixed, so the intervention changes the relationship between layout and label rather than simply adding unrelated data. In the appendix we give the exact prompt schema, filtering rules, and aggregation pseudocode.

3.1 TRAINING ARMS

The four arms isolate the role of presentation from the role of the scoring rule. Arm A repeats the preferred-first rendering for every training pair. Arm B duplicates each pair with its swapped rendering and flips the target slot. Arm C uses the same two-order data as B and adds a consistency term that rewards reversal after the exchange. Arm D uses the two-order data and a paired-advantage reweighting term. The data source, optimizer, update count, and evaluation interface are shared across arms, so the arm label identifies the intervention rather than a change in benchmark or model family.

For each seed, we construct the training rows before optimization and retain a stable source-pair identifier. This makes the two orientations a matched unit throughout training and evaluation. The

fixed-order control is deliberately represented with the same source-pair count and update schedule as the two-order arm; the control therefore measures the effect of exposing the second physical arrangement under the same training recipe. We do not pool rows across datasets when computing uncertainty, and all reported dataset values are first averaged within seed and then across the four seeds.

3.2 AUDIT PROTOCOL

The audit uses the same query and response text in both orientations. Only the order of the response blocks and the target slot changes. For an item whose preferred response is in slot A in the original rendering, the swapped rendering places it in slot B and changes the target to B. The judge is prompted once for each rendering, and the parser records the final binary slot token. This construction gives each item two directly comparable predictions and avoids treating the two orientations as independent questions.

The audit separates three behaviors that a canonical-order score conflates. Order-averaged accuracy asks whether the judge is correct in both physical layouts. Both-order correctness asks whether the same item is correct twice, which is stricter than averaging the two answers. Swap consistency asks whether the prediction reverses when the responses exchange slots. First-position selection is a direct readout of the shortcut because a policy that always selects the first displayed response receives a high value even when its content comparison is absent. We report these metrics together so that an accuracy change can be read alongside its behavioral mechanism.

4 EXPERIMENTS

Let each example contain a query q, a preferred response r^+, and a rejected response r^-. An orientation $o \in \{+1, -1\}$ determines which response occupies slot A. The observed label is $y^o = A$ for $o = +1$ and $y^o = B$ for $o = -1$. Fixed-order training uses only $o = +1$; two-order training includes both orientations with the label flipped for the swapped copy. We report accuracy on the original order, on the swapped order, their mean (order-averaged accuracy), and the fraction of predictions selecting the first displayed response. Swap consistency is the fraction of pairs for which the prediction reverses after the responses are exchanged.

Controlled comparison. All four arms use Qwen2.5-7B, the same preference source, batch and optimizer settings, and 200 update steps. Arm A is fixed-order pointwise training. Arm B is two-order pointwise training and is the main intervention. Arms C and D add, respectively, a consistency reward and a paired-advantage term to B. Each arm has seeds 42, 123, 456, and 789. Evaluation uses held-out RewardBench pairs and a clean UltraFeedback split; every item is scored in both orientations.

Data and prompts. We preserve the original question and response text and modify only the order of the two response blocks. The judge prompt names the two slots explicitly and asks for a single binary verdict after a short rationale. The answer parser accepts only the final slot token; malformed completions are counted as incorrect and are reported separately in the supplementary audit. The held-out evaluations are never used to construct training batches. We use the same tokenizer, maximum sequence length, temperature, and decoding rule for both orientations.

Optimization and controls. The pointwise arms optimize the same cross-entropy target. The consistency arm adds a penalty when the two sampled orientations disagree after the label flip. The

Table 1: Main intervention. Values are mean percentages over four seeds. The interval is a question-bootstrap 95% interval for the paired item difference.

Dataset	A fixed	B two-order	Change	95% CI
RewardBench OA	65.14	87.05	+21.91	[+19.54,+24.19]
UF-clean OA	55.51	75.31	+19.79	[+16.90,+22.58]

Table 2: Paired-behavior readouts for the main arms. OA is order-averaged accuracy; both-order is the proportion of questions correct under both orientations.

Dataset	Arm	OA	Both-order	Swap consistency
RewardBench	A fixed	65.14	31.07	31.79
	B pointwise	87.05	80.90	87.69
	C consistency	86.92	81.24	88.64
	D pair-aware	86.66	80.73	88.08
UF-clean	A fixed	55.51	12.25	13.03
	B pointwise	75.31	62.58	74.55
	C consistency	75.89	63.59	75.22
	D pair-aware	76.34	63.75	74.83

paired-advantage arm reweights a pair by the difference between the two orientation advantages. Both controls are useful negative checks: if the main effect required a specialized loss, they should improve over arm B. Instead, the dominant change occurs when the two physical layouts are made available to the pointwise learner.

Uncertainty. For each seed we compute item-level paired differences between arms A and B, retaining the two orientations as a matched unit. We bootstrap questions within each dataset and report percentile 95% intervals conditional on the four observed seeds. Seed means are reported after computing each seed's metric, so a large dataset does not silently replace between-seed variation. The dose curve is analyzed separately because it uses a different training source and is intended as a supporting mechanism check.

The dose curve is analyzed separately because it uses a different training source and is intended as a supporting mechanism check. The four operating points use the same pair identifiers and evaluation protocol; only the fraction of preferred-first renderings changes. We therefore interpret the curve as a controlled sensitivity analysis of the interface correlation. The direction is assessed from the joint movement of order-averaged accuracy and first-position selection, while the table gives the exact operating points for reproducibility.

4.1 Main results

Table 2 makes explicit why the average alone is insufficient. On RewardBench, A's 65.14% average includes only 31.07% of questions answered correctly in both orders. B raises both quantities, and its swap consistency increases by 55.90 points. On UF-clean, the fixed-order arm has only 12.25% both-order correctness and 13.03% swap consistency; B raises these to 62.58% and 74.55%, respectively. The changes in both-order correctness are 49.83 and 50.33 points on RewardBench and UF-clean. These are large paired changes, not a small correction to an otherwise reliable ranking.

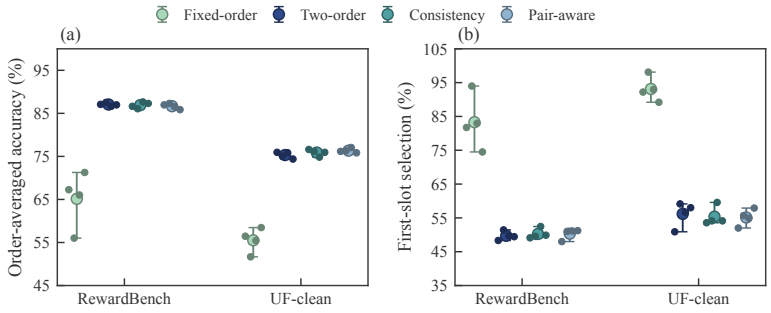

Figure 2: Main intervention across the two held-out datasets. Points show means over four seeds, vertical whiskers span the seed range, and smaller points show individual seeds. (a) Order-averaged accuracy improves when both physical response arrangements are included during training. (b) The same intervention reduces first-position selection, the mechanism readout for the fixed-slot shortcut.

Table 3: Separate 10K dose experiment. All four ratios use the same training pair IDs and evaluation protocol.

Preferred-first share	OA accuracy	Both-order correct	Swap consistency	First-position choice
50%	73.78	60.63	73.30	54.34
80%	68.05	44.50	52.58	71.79
95%	62.69	29.48	33.40	82.19
100%	56.34	14.25	15.70	91.78

Table 1 is the central result. The gain is not an original-order artifact: both the original and swapped conditions improve for arm B, and all four seeds have the same direction on both datasets. The mechanism readout changes at the same time. First-position selection falls from 83.30% to 49.73% on RewardBench (a 33.57 point decrease) and from 93.12% to 56.18% on UF-clean (a 36.94 point decrease). The two-order arm therefore improves the paired test and removes the feature that made the label predictable.

The component check does not show a clear gain from adding a consistency reward or a paired-advantage term to the two-order baseline. C reaches 86.92% on RewardBench and 75.89% on UF-clean; D reaches 86.66% and 76.34%. Relative to B, the order-averaged differences are -0.14 points (95% CI $[-1.00, +0.67]$) and $+0.58$ ($[-0.36, +1.59]$) for C, and the C-to-D changes are -0.25 ($[-0.97, +0.53]$) and $+0.45$ ($[-0.45, +1.31]$). All include zero. We therefore treat the simple paired-layout intervention as the result, rather than attributing the recovery to a new reward objective.

4.2 Dose study

The dose curve supports the same explanation without combining it numerically with the four-arm study. On a separate 10K training source, four seeds, and a 1,000-question paired test set, increasing the preferred-first fraction from 50% to 100% reduces paired robustness and increases first-position selection. The 50/50 condition is 17.44 points higher in order-averaged accuracy than 100/0. The curve is a diagnostic of the interface correlation, not evidence that every deployment has a 100/0 distribution.

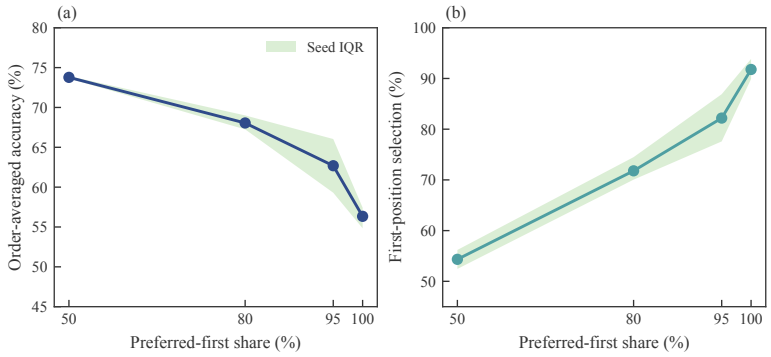

Figure 3: Dose curve for the preferred-first share in a separate 10K training study. Lines are means over four seeds and the light band is the interquartile range across seeds. Increasing the fraction of preferred-first layouts lowers paired accuracy and increases first-position selection, supporting the interface-correlation interpretation.

The curve emphasizes the trend across training mixtures, while Table 3 reports the exact operating points.

The adjacent order-averaged changes are -5.73, -5.36, and -6.35 points for the 50–80%, 80–95%, and 95–100% transitions, respectively. Their question-bootstrap intervals are $[-7.01, -4.37]$, $[-6.39, -4.35]$, and $[-7.17, -5.52]$. The baseline judge on the same 1,000 questions is 70.65% order-averaged accuracy. Thus 50/50 training is 3.12 points above baseline, whereas 80/20, 95/5, and 100/0 are 2.60, 7.96, and 14.31 points below it. This comparison sharpens the interpretation: the

increasingly biased training conditions do not merely fail to repair the shortcut; in this run family they yield lower paired accuracy than the untrained judge. These intervals condition on the four seeds and should not be read as uncertainty over all training recipes.

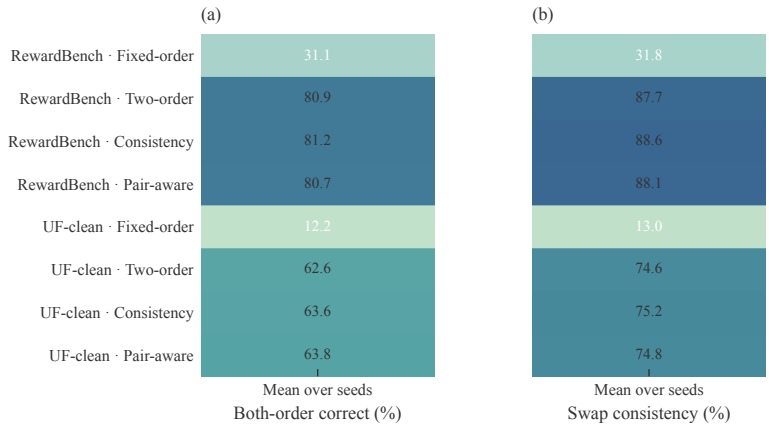

Figure 4: Paired-behavior readouts for the four training arms. Each cell is the mean over four seeds; darker cells indicate stronger paired robustness. (a) Fraction of questions correct in both orientations. (b) Swap consistency, the fraction of predictions that reverse after exchanging the responses.

5 Analysis

5.1 Identification and Alternative Explanations

One alternative explanation is that arm B simply sees twice as many prompts per pair. The fixed-order control is constructed from one preferred-first rendering per source pair and copies that row twice, whereas B contains the two distinct rendered orientations. Thus both arms have 4,178 training rows, the same 200 optimizer steps, batch size, rollout count, tokenizer, and maximum lengths, but only B varies the physical layout. We do not claim token-level equality: sequence lengths are data-dependent, and equal steps do not imply equal token or gradient exposure. The A-to-B estimate is therefore the total effect of replacing a repeated single orientation with two distinct orientations under a matched update configuration, rather than an orientation-only component isolated from all data-interface effects. A second alternative is that swapping responses changes truncation. We apply the same maximum length and tokenizer to both orientations and record truncation rates; examples that would be truncated differently are excluded from the paired diagnostic. A third alternative is a lexical artifact: response texts are unchanged, and the systematic intervention is which block occupies a slot.

Another alternative is that the effect is a benchmark artifact. RewardBench and UF-clean differ in source construction and response style, yet the direction of the intervention is the same. We do not pool their items or treat the two datasets as independent draws from a single distribution. Instead, we use the replication across domains as a qualitative check that the fixed-order interface is not tied to one benchmark's formatting.

Finally, the change could reflect a model learning the instruction "consider both orders" rather than removing the reward-predictive feature. The training intervention does not add such an instruction. It changes the physical arrangement and flips the target. The evaluation prompt remains the same for arms A–D. This is why the first-slot reduction and the swapped-order improvement are important: they demonstrate a behavioral change under the unchanged evaluation interface.

5.2 Reproducibility

We use deterministic data construction for each seed and record the seed at the data-loader and optimizer levels. Checkpoints are evaluated with the same decoding configuration used for the paired audit. Each item is assigned a stable identifier, which allows the bootstrap to resample the

complete pair rather than individual orientations. The release tables are generated from a machine-readable summary that stores per-seed metrics, item counts, and confidence intervals.

The benchmark names identify the public evaluation sources; the paper does not redistribute response text that is subject to external licenses. The source package includes the prompt schema, the orientation transformation, the metric definitions, and the exact numbers in the main text. A researcher with access to the same model and benchmarks can regenerate the training batches and reproduce the audit without inferring an undocumented label convention.

Compute accounting. The main comparison uses four seeds and four arms, each with 200 update steps. Arm C and D are controls, not additional claims about scaling. The dose curve is run independently on its 10K source. Reporting this accounting matters because a two-order intervention can otherwise be mistaken for a larger data or compute budget. Our interpretation relies on the matched step and source design, not on the absolute training cost.

5.3 Scope and Limitations

Our main comparison uses one 7B backbone, a controlled preference stack, and 200 update steps. Larger models may use content more reliably or may exploit the same correlation more quickly. The reported dose curve is a separate diagnostic and should not be extrapolated to deployment prevalence. We also do not claim that a two-order mixture is sufficient for every language, formatting, or multi-label judge. These boundaries are part of the result: the evidence supports a concrete intervention in the tested setting, and it motivates audits for other settings.

5.4 What the paired intervention identifies

The evidence supports a specific causal reading of the training interface. The fixed-order arm supplies a label that can be predicted from the displayed slot, so its canonical-order accuracy does not identify whether the judge used the response content. The swapped evaluation changes that feature while preserving the pair itself. When the two-order arm improves on both orientations and its first-position rate moves toward balance, the improvement is aligned with the intended comparison rule rather than with a stable slot preference. The dose study gives the same relationship continuously: as the preferred-first fraction rises, first-position selection rises while paired accuracy falls.

The controls sharpen this interpretation. A consistency term and a paired-advantage term operate on the same two-order data and remain close to the pointwise two-order arm on both benchmarks. The central change therefore occurs at the data-interface level. This makes the intervention easy to audit: a training pipeline can record the orientation variable, run the paired test, and inspect the same four metrics before adding a more elaborate objective.

5.5 Cross-dataset replication

The two evaluation sources provide a useful replication because they differ in how preference pairs are collected and how response quality is expressed. The absolute scores are therefore not expected to match. The intervention moves both sources in the same direction, however. On RewardBench, order-averaged accuracy rises from 65.14% to 87.05%, a 21.91-point change. On UF-clean, it rises from 55.51% to 75.31%, a 19.79-point change. The corresponding first-position rates fall by 33.57 and 36.94 points. Reading the four quantities together is more informative than ranking the datasets by their absolute accuracy: the mechanism readout and the paired score change together in both domains.

The paired metrics show why this replication matters. In RewardBench, only 31.07% of fixed-order questions are correct in both orientations, while the two-order pointwise arm reaches 80.90%. In UF-clean, the corresponding values are 12.25% and 62.58%. Swap consistency follows the same pattern, increasing from 31.79% to 87.69% and from 13.03% to 74.55%. These values separate two kinds of improvement. A judge can increase its average by becoming better on the original layout while retaining a stable slot rule; the paired measures require the verdict to respond to the changed presentation.

5.6 A PRACTICAL AUDIT

The intervention can be applied as a small audit around any preference-judge training pipeline. First, record the orientation variable at the point where preference rows are converted into prompts. Second, construct a paired evaluation set by exchanging the response blocks and flipping the target slot. Third, report order-averaged accuracy, both-order correctness, swap consistency, and first-position selection together. Finally, compare the paired metrics before interpreting a canonical-order gain as evidence of better judging. This sequence requires no change to the judge architecture and makes the relevant presentation feature explicit in the experiment record.

The audit also gives a simple diagnostic for future changes. If a prompt template, label converter, or truncation rule changes, the same source pair can be rendered again under both orientations. A stable content-sensitive judge should preserve the preferred response while changing the selected slot; a slot-sensitive policy will retain its original slot choice. The test therefore functions as a regression check for data processing as well as a training intervention. Because the paired unit is the source question, the bootstrap and the error analysis can remain aligned with the unit on which the claim is made.

The dose study extends this audit from a binary comparison to a controlled continuum. At a 50% preferred-first share, order-averaged accuracy is 73.78% and first-position selection is 54.34%. At 100%, the values are 56.34% and 91.78%. The intermediate points, 68.05%/71.79% and 62.69%/82.19%, preserve the same ordering. This monotone pattern is useful when a data pipeline cannot be perfectly balanced: it quantifies how far the observed presentation mixture is from the paired regime and provides an operating-point table for future replications.

5.7 RECOMMENDATIONS FOR PREFERENCE DATA PIPELINES

The results suggest three concrete checks for a preference-data pipeline. First, store the orientation of every rendered pair as an explicit field rather than allowing a converter to choose a canonical slot silently. This makes it possible to measure the correlation between orientation and target before training begins. Second, retain a paired audit split whose questions are scored in both orientations. The split need not be large to expose a stable position rule: the paired unit is the question, and the same response texts are reused after the exchange. Third, report the mechanism readout with the task score. A single canonical-order accuracy can hide a high first-position rate, whereas the joint report makes the shortcut visible.

These checks also clarify what should be held fixed in a comparison. The source questions, response strings, tokenizer, maximum length, decoding rule, and optimizer schedule should be shared across arms. The orientation transform should be applied before batching so that the number of update steps does not become a proxy for the intervention. When a consistency loss or a pair-aware reweighting is added, it should be evaluated on the same two-order data as the pointwise control. Otherwise an apparent objective improvement can be confounded with a change in presentation coverage. Balanced slot selection alone is insufficient: a judge can select each slot equally often while comparing responses incorrectly. Jointly reporting paired accuracy and swap consistency distinguishes balanced outputs from decisions that track the preferred response across layouts.

6 CONCLUSION

When the presentation of a preference pair predicts its label, single-order accuracy cannot tell whether a judge compares content or follows position. A paired-order intervention makes that distinction visible and, in our controlled stack, restores large amounts of order-averaged performance. The practical rule is simple: balance reward-predictive presentation features during training and audit the corresponding intervention before interpreting an accuracy gain as better judgment. The paired test should be retained as a regression check whenever prompt serialization or label conversion changes. In our measurements, the same intervention raises both-order correctness and swap consistency while reducing first-position selection, and the dose curve shows how these quantities move as the preferred-first share changes. This makes a canonical-order score answerable as evidence of comparative behavior rather than merely evidence of a stable interface shortcut, and it gives practitioners a concrete protocol for checking that interpretation.

AI USE STATEMENT

AI assistance was used in drafting this article and in the retrieval and discovery of related work. All research concepts, ideas and analyses were developed and conducted by the authors, and all editorial judgement and conclusions are the authors' own. The authors take full responsibility for the content of this paper.

ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. No human subjects or animal experimentation were involved. RewardBench and the clean UltraFeedback split used here are public evaluation resources used in accordance with their release terms, and the Qwen2.5-7B backbone is publicly released. The tasks contain no personally identifiable information or demographic attributes, and no user-facing system was deployed.

REPRODUCIBILITY STATEMENT

The experimental setting and paired estimands are described in Sections 3 and 4. Appendix A specifies evaluation units and scoring conventions; Appendices B–D report the full seed results. Appendix F gives the seed-sensitivity analysis, and Appendix I specifies the reproducible question bootstrap and analysis workflow. The accompanying label-level data and analysis scripts regenerate the tables and paired-outcome figure. RewardBench and UltraFeedback are public resources. We will release the code and derived evaluation data upon acceptance.

REFERENCES

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pp. 10835–10866. PMLR, 2023.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 3428–3448, 2019.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.

Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. Disentangling length from quality in direct preference optimization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 4998–5017, 2024.

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. *arXiv preprint arXiv:2305.18290*, 2023.

Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. Judging the judges: A systematic study of position bias in llm-as-a-judge. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 292–314, 2025.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in rlhf. In *Proceedings of the Conference on Language Modeling (COLM)*, 2024.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking, 2025. URL https://arxiv.org/abs/2209.13085.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics*, 8:621–633, 2020.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, et al. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9440–9450, 2024.

Minghao Wu and Alham Fikri Aji. Style over substance: Evaluation biases for large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 297–312, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

A Evaluation Protocol

This section expands the paired evaluation in Sections 3 and 4. The evaluation unit is a source item with its original and swapped predictions; each row of the released CSV keeps these two predictions together.

Table 4: **Every evaluated condition preserves the two orientations of each source item.** Items are counted per condition. Conditions are arms in the four-arm study and mixture ratios in the dose study. Verdicts count two orientations for every item–seed–condition combination.

Source	Items	Seeds	Conditions	Groups	Verdicts
RewardBench	449	4	4	16	14368
UF-clean	449	4	4	16	14368
Dose study	1000	4	4	16	32000
Untrained reference	1000	1	1	1	2000

The four-arm file contains 14,368 rows: two evaluation sources, four arms, four seeds, and 449 items in each group. Each row holds two verdicts, giving 28,736 verdicts across the two sources. The dose file contains 16,000 trained rows and 1,000 untrained-reference rows. These counts describe evaluation records, not numbers of training examples or independently collected questions.

Matching keys. The key for the main comparison is dataset, item identifier, arm, and seed. Contrasts align the item identifier and seed within a dataset before subtracting scores. The dose study aligns the item identifier and seed across preferred-first shares. The untrained reference is stored once and is not expanded into four artificial training seeds. RewardBench and UF-clean are summarized separately; matching identifiers across these sources is not part of the analysis.

Orientation and labels. All released original-order gold labels are A. For a swapped rendering, the scoring map sends A to B and B to A. The analysis code also preserves C under swapping. Thus an A verdict in the original orientation and a B verdict in the swapped orientation is correct twice for these records. A repeated A verdict is correct only in the original orientation. These two outcomes have different scientific meanings even though a single-order score can make them look equally successful.

Metric denominators. Let u_i and v_i denote original- and swapped-order correctness for item i. The accuracy denominator is $2n$ verdicts, whereas both-order correctness uses n source items. First-position selection also uses $2n$ verdicts. Swap consistency uses n pairs and compares the swapped prediction with the flipped original prediction. Every metric is calculated within one seed before equal-weight averaging across seeds.

$$A_{\mathrm{OA}} = \frac{1}{2n}\sum_i (u_i + v_i), \qquad B = \frac{1}{n}\sum_i u_i v_i, \tag{5}$$

$$F = \frac{1}{2n}\sum_i \left[\mathbf{1}(\hat{y}_i^{+} = A) + \mathbf{1}(\hat{y}_i^{-} = A)\right]. \tag{6}$$

Scoring contract. The released analysis script recognizes A, B, and C as valid parsed choices. PARSE_FAIL is outside this set. Since the released gold labels are binary, C does not earn accuracy credit; C/C is nevertheless consistent under the stored flip map. Consequently, accuracy and consistency should be interpreted together. The following tables use this exact contract, ensuring agreement with the machine-readable summaries.

Shared item sets. All 32 four-arm groups contain exactly 449 distinct identifiers; every arm and seed within a dataset has the same identifier set. All 16 dose groups contain exactly 1,000 distinct identifiers with the same set across shares and seeds. This permits item-level subtraction before averaging. The item index is local to its source: a numerical match between identifiers from different datasets does not establish a common question. The release stores the source name alongside each identifier to preserve this distinction.

B REWARDBENCH RESULTS

This section expands the main-results comparison in Section 4. It reports every arm and seed on RewardBench.

Table 5: **Two-order training improves paired accuracy in every seed.** Orig. and Swap are orientation-specific accuracies; OA averages them. Both denotes both-order correctness, Cons. swap consistency, and First first-slot selection. All scores are percentages on 449 items per seed.

Arm	Seed	Orig.	Swap	OA	Both	Cons.	First
Fixed-order	42	99.55	34.97	67.26	34.97	35.19	81.74
Fixed-order	123	100.00	12.03	56.01	12.03	12.03	93.99
Fixed-order	456	99.55	32.52	66.04	32.52	32.96	82.96
Fixed-order	789	97.77	44.77	71.27	44.77	46.99	74.50
Fixed-order	Mean	99.22	31.07	65.14	31.07	31.79	83.30
Two-order	42	85.75	88.42	87.08	81.07	87.97	48.33
Two-order	123	89.31	85.75	87.53	81.51	87.97	51.45
Two-order	456	86.86	86.41	86.64	80.18	87.08	49.67
Two-order	789	86.64	87.31	86.97	80.85	87.75	49.44
Two-order	Mean	87.14	86.97	87.05	80.90	87.69	49.72
Consistency	42	85.97	87.31	86.64	81.74	90.20	49.11
Consistency	123	85.97	86.19	86.08	79.51	86.86	49.55
Consistency	456	90.42	84.86	87.64	81.74	88.20	52.45
Consistency	789	87.53	87.08	87.31	81.96	89.31	49.89
Consistency	Mean	87.47	86.36	86.92	81.24	88.64	50.25
Pair-aware	42	85.30	88.64	86.97	81.96	89.98	48.00
Pair-aware	123	88.64	85.97	87.31	81.96	89.31	51.00
Pair-aware	456	87.97	85.08	86.53	80.40	87.75	51.11
Pair-aware	789	87.31	84.41	85.86	78.62	85.30	51.22
Pair-aware	Mean	87.31	86.02	86.66	80.73	88.08	50.33

Reading orientation-specific accuracy. Original and swapped scores retain information that is lost in their arithmetic mean. A fixed-slot policy earns credit when the preferred response occupies its favored slot, while the paired evaluation asks whether the preference survives the physical exchange. The OA column assigns equal weight to the two layouts. The Both column requires both judgments to be correct for the same item and therefore measures a stricter form of paired reliability.

Comparing seeds. Each row uses the same 449 source items within this dataset. The four seed rows describe repeated training conditions evaluated on a shared item set. The Mean row gives each seed weight one quarter. It does not treat the repeated appearances of a question as newly sampled questions. This distinction is retained when forming matched contrasts and when interpreting seed sensitivity.

The two-order OA gain is positive for each of the four seeds, ranging from 15.70 to 31.51 percentage points. Its mean gain is 21.91 points. Both-order correctness changes by 49.83 points and first-position selection changes by -33.57 points. The three readouts describe accuracy, paired reliability, and slot preference on the same observations.

Component controls. Consistency and Pair-aware share the two-order representation. Their rows allow comparison against the Two-order reference without switching evaluation items. These controls answer a different question from the Fixed-order contrast: they assess the incremental behavior of an additional objective component after both layouts are available. The complete rows are retained to make that distinction inspectable across all four seeds.

Units and precision. All entries except the seed identifier are percentages rounded to two decimals after aggregation. Contrasts elsewhere in this appendix are computed from the unrounded item scores, so subtracting two displayed rounded means can differ by 0.01 points. Original and swapped columns share a denominator of 449; their OA average uses 898 verdicts per seed.

C UF-CLEAN RESULTS

This section expands the main-results comparison in Section 4. It reports every arm and seed on UF-clean.

Table 6: **Two-order training improves paired accuracy in every seed.** Orig. and Swap are orientation-specific accuracies; OA averages them. Both denotes both-order correctness, Cons. swap consistency, and First first-slot selection. All scores are percentages on 449 items per seed.

Arm	Seed	Orig.	Swap	OA	Both	Cons.	First
Fixed-order	42	98.66	14.25	56.46	14.25	14.92	92.20
Fixed-order	123	99.78	3.56	51.67	3.56	3.79	98.11
Fixed-order	456	98.66	12.25	55.46	12.25	12.92	92.98
Fixed-order	789	98.00	18.93	58.46	18.93	20.49	89.20
Fixed-order	Mean	98.78	12.25	55.51	12.25	13.03	93.12
Two-order	42	76.84	75.06	75.95	63.92	75.95	50.89
Two-order	123	84.19	65.92	75.06	61.69	73.27	59.13
Two-order	456	82.63	69.04	75.84	63.25	74.83	56.68
Two-order	789	82.41	66.37	74.39	61.47	74.16	58.02
Two-order	Mean	81.51	69.10	75.31	62.58	74.55	56.18
Consistency	42	80.18	73.05	76.61	65.03	76.61	53.56
Consistency	123	80.40	71.94	76.17	64.37	76.17	54.12
Consistency	456	84.41	65.26	74.83	60.80	71.94	59.58
Consistency	789	80.18	71.71	75.95	64.14	76.17	54.12
Consistency	Mean	81.29	70.49	75.89	63.59	75.22	55.35
Pair-aware	42	78.17	74.16	76.17	64.81	77.28	52.00
Pair-aware	123	81.96	70.60	76.28	63.70	74.83	55.68
Pair-aware	456	81.96	72.16	77.06	64.59	75.06	54.90
Pair-aware	789	83.74	67.93	75.84	61.92	72.16	57.91
Pair-aware	Mean	81.46	71.21	76.34	63.75	74.83	55.12

Reading orientation-specific accuracy. Original and swapped scores retain information that is lost in their arithmetic mean. A fixed-slot policy earns credit when the preferred response occupies its favored slot, while the paired evaluation asks whether the preference survives the physical exchange. The OA column assigns equal weight to the two layouts. The Both column requires both judgments to be correct for the same item and therefore measures a stricter form of paired reliability.

Comparing seeds. Each row uses the same 449 source items within this dataset. The four seed rows describe repeated training conditions evaluated on a shared item set. The Mean row gives each seed weight one quarter. It does not treat the repeated appearances of a question as newly sampled questions. This distinction is retained when forming matched contrasts and when interpreting seed sensitivity.

The two-order OA gain is positive for each of the four seeds, ranging from 15.92 to 23.39 percentage points. Its mean gain is 19.79 points. Both-order correctness changes by 50.33 points and first-position selection changes by -36.94 points. The three readouts describe accuracy, paired reliability, and slot preference on the same observations.

Component controls. Consistency and Pair-aware share the two-order representation. Their rows allow comparison against the Two-order reference without switching evaluation items. These controls answer a different question from the Fixed-order contrast: they assess the incremental behavior of an additional objective component after both layouts are available. The complete rows are retained to make that distinction inspectable across all four seeds.

Units and precision. All entries except the seed identifier are percentages rounded to two decimals after aggregation. Contrasts elsewhere in this appendix are computed from the unrounded item scores, so subtracting two displayed rounded means can differ by 0.01 points. Original and swapped columns share a denominator of 449; their OA average uses 898 verdicts per seed.

D DOSE RESULTS

This section expands the dose study in Section 4. The four preferred-first shares are evaluated on the same 1,000 items, and the untrained reference is reported once.

Table 7: **The balanced mixture has the highest OA accuracy in each seed.** Share is the preferred-first training percentage. The reference is the untrained judge. Orig., Swap, Both, Cons., and First follow the definitions in Appendix A; all scores are percentages.

Share	Seed	Orig.	Swap	OA	Both	Cons.	First
50	42	80.00	67.80	73.90	60.60	73.00	55.95
50	123	76.70	70.70	73.70	61.40	74.90	52.65
50	456	76.30	71.80	74.05	61.60	74.80	51.85
50	789	80.80	66.10	73.45	58.90	70.50	56.90
50	Mean	78.45	69.10	73.78	60.62	73.30	54.34
80	42	90.80	47.50	69.15	46.40	54.20	71.25
80	123	92.50	40.80	66.65	39.90	46.10	75.20
80	456	86.00	51.90	68.95	49.60	61.00	66.45
80	789	92.20	42.70	67.45	42.10	49.00	74.25
80	Mean	90.38	45.72	68.05	44.50	52.58	71.79
95	42	97.10	21.50	59.30	21.40	24.10	87.50
95	123	93.60	38.70	66.15	37.90	43.40	76.65
95	456	94.80	37.20	66.00	36.80	41.50	77.90
95	789	96.50	22.10	59.30	21.80	24.60	86.70
95	Mean	95.50	29.88	62.69	29.47	33.40	82.19
100	42	99.40	6.60	53.00	6.60	7.20	96.25
100	123	98.70	12.20	55.45	12.20	13.40	93.05
100	456	97.70	23.30	60.50	23.20	25.30	86.90
100	789	97.70	15.10	56.40	15.00	16.90	90.90
100	Mean	98.38	14.30	56.34	14.25	15.70	91.78
Reference	Single	78.00	63.30	70.65	56.50	70.00	55.80

Within-seed comparisons. Comparing shares within a seed holds the evaluated questions fixed and preserves the training-seed index. The balanced condition has higher OA accuracy than each of the three biased conditions in all four seeds. This is a statement about the balanced-versus-biased contrast. The monotone trend discussed in the main text is a trend in the four-seed means, so it should be read at that aggregation level.

Reference and intervention. The untrained judge is evaluated on the same source items and provides a common starting point for the dose study. It is not a fifth trained seed. Its OA value is 70.65%, compared with 73.78% for the balanced four-seed mean. The difference is calculated before rounding. The reference contextualizes the trained operating points, while the mixture-to-mixture contrasts isolate differences among the four specified presentation shares.

Paired interpretation. At 50% preferred-first share, the mean Both score is 60.63%; at 100%, it is 14.25%. The corresponding First scores are 54.34% and 91.78%. Original-order and swapped-order columns reveal how the combined accuracy changes between these operating points. This decomposition keeps the meaning of the dose manipulation visible: it varies the presentation-label correlation while scoring both physical arrangements equally.

Study boundaries. The dose and four-arm files have different evaluation item sets and different group sizes. We retain their separate denominators and compute each result within its own study. The dose table supports the presentation-mixture analysis; the RewardBench and UF-clean tables support the four-arm comparison. Their shared directional pattern is a cross-study observation rather than a pooled numerical estimate.

Seed dispersion. The four seed rows show variation across the trained conditions separately from item-bootstrap uncertainty. The retained-seed comparisons preserve both orientations.

E MATCHED CONTRASTS

This section expands the intervention and component comparisons in Section 4. Subtraction is performed within seed and dataset, retaining the matched evaluation items.

Table 8: **The layout intervention produces positive OA gains across both datasets and all seeds.** A is Fixed-order, B Two-order, C Consistency, and D Pair-aware. RB and UF denote RewardBench and UF-clean. Entries are signed percentage-point differences.

Data	Contrast	Seed	OA	Both	Cons.	First
RB	B-A	42	+19.82	+46.10	+52.78	-33.41
RB	B-A	123	+31.51	+69.49	+75.95	-42.54
RB	B-A	456	+20.60	+47.66	+54.12	-33.30
RB	B-A	789	+15.70	+36.08	+40.76	-25.06
RB	C-B	42	-0.45	+0.67	+2.23	+0.78
RB	C-B	123	-1.45	-2.00	-1.11	-1.89
RB	C-B	456	+1.00	+1.56	+1.11	+2.78
RB	C-B	789	+0.33	+1.11	+1.56	+0.45
RB	D-C	42	+0.33	+0.22	-0.22	-1.11
RB	D-C	123	+1.22	+2.45	+2.45	+1.45
RB	D-C	456	-1.11	-1.34	-0.45	-1.34
RB	D-C	789	-1.45	-3.34	-4.01	+1.34
UF	B-A	42	+19.49	+49.67	+61.02	-41.31
UF	B-A	123	+23.39	+58.13	+69.49	-38.98
UF	B-A	456	+20.38	+51.00	+61.92	-36.30
UF	B-A	789	+15.92	+42.54	+53.67	-31.18
UF	C-B	42	+0.67	+1.11	+0.67	+2.67
UF	C-B	123	+1.11	+2.67	+2.90	-5.01
UF	C-B	456	-1.00	-2.45	-2.90	+2.90
UF	C-B	789	+1.56	+2.67	+2.00	-3.90
UF	D-C	42	-0.45	-0.22	+0.67	-1.56
UF	D-C	123	+0.11	-0.67	-1.34	+1.56
UF	D-C	456	+2.23	+3.79	+3.12	-4.68
UF	D-C	789	-0.11	-2.23	-4.01	+3.79

Separating intervention and ablation. B–A replaces the fixed-order condition with the two-order pointwise condition. C–B measures the incremental consistency component, and D–C compares the next component configuration. The contrast labels retain the reference arm explicitly: a D–C value is not a D–B estimate. Reading the columns together distinguishes improvement in paired correctness from a change in slot preference.

Magnitude and direction. All eight B–A OA differences are positive. Component contrasts are numerically smaller and vary across seeds and datasets. Their seed-specific values describe the incremental effects at the tested operating point. They are presented alongside the primary contrast so that a reader can evaluate the simple layout change and the additional components on a common scale.

Aggregation rule. If $m_{a,s}$ is a metric for arm a and seed s, the reported contrast is $\bar{\Delta} = \frac{1}{4}\sum_s (m_{a,s} - m_{b,s})$. This paired averaging gives each seed equal weight. It also makes the mean contrast equal to the difference of unrounded arm means. The signs of First differences use the same subtraction order as the accuracy metrics; a negative value means less first-slot selection in the left-hand arm.

What the contrasts hold constant. Pairing uses the scored item and seed identifiers. Every difference compares the same question in both orientations under two arm labels. This removes variation due to different sampled evaluation questions from that contrast. The estimate describes the full experimental arm, rather than a decomposition into independent effects of token exposure, updates, and reward terms. Averaging the unrounded seed differences reproduces the contrast of the arm means.

F SEED SENSITIVITY

This section expands the stability analysis in Section 4. Each row removes one training seed and recomputes the contrast from the remaining three.

Table 9: **The primary gains remain positive after excluding any one seed.** RB and UF compare Two-order with Fixed-order. Dose contrasts compare 50% with the indicated biased share. Values are percentage-point differences averaged over the three retained seeds.

Contrast	Omitted	OA	Both	Cons.	First
RB	42	+22.61	+51.08	+56.94	-33.63
RB	123	+18.71	+43.28	+49.22	-30.59
RB	456	+22.35	+50.56	+56.50	-33.67
RB	789	+23.98	+54.42	+60.95	-36.41
UF	42	+19.90	+50.56	+61.69	-35.49
UF	123	+18.60	+47.74	+58.87	-36.27
UF	456	+19.60	+50.11	+61.40	-37.16
UF	789	+21.08	+52.93	+64.14	-38.86
50-80	42	+6.05	+16.77	+21.37	-18.17
50-80	123	+5.28	+14.33	+18.03	-15.75
50-80	456	+5.93	+17.50	+23.03	-18.40
50-80	789	+5.63	+15.90	+20.47	-17.48
50-95	42	+9.92	+28.47	+36.90	-26.62
50-95	123	+12.27	+33.70	+42.70	-29.13
50-95	456	+12.10	+33.27	+42.10	-28.45
50-95	789	+10.07	+29.17	+37.90	-27.20
50-100	42	+16.28	+43.83	+54.87	-36.48
50-100	123	+17.17	+45.43	+56.30	-36.45
50-100	456	+18.73	+49.03	+60.30	-38.23
50-100	789	+17.57	+47.20	+58.93	-38.58

A deterministic sensitivity check. Leave-one-seed-out analysis asks whether the sign of an aggregate comparison relies on a single training seed. It produces four overlapping subsets of the observed seed set. These subsets are not independent replications, and their range is not a confidence interval. Each row has a direct interpretation: the same contrast is recomputed after excluding the named seed.

Primary result. Removing any seed preserves the positive Two-order-minus-Fixed-order OA difference on RewardBench and UF-clean. Both-order correctness and swap consistency also retain positive changes in each subset. The First difference remains negative. Thus the three-seed recomputations retain the joint accuracy and position pattern of the four-seed result.

Dose result. For each biased share, all four retained-seed subsets favor the balanced mixture in OA accuracy. The dose comparison keeps the same question identifiers at every share, and only the seed-averaging step changes in this check. It tests sensitivity to the reported seed set without introducing a new training run or changing the mixture experiment.

Relation to uncertainty. Question resampling and seed deletion answer different questions. Question resampling assesses sensitivity to the evaluated item sample conditional on the observed seeds. Seed deletion assesses sensitivity of the mean to one observed training run. Neither operation replaces the other, so the results are reported with their own units and interpretations. The item-level files support both calculations using the same matching keys.

Algebraic check. For four seed contrasts $d_1, \ldots, d_4$ and full mean $\bar{d}$, deleting seed s gives $(4\bar{d} - d_s)/3$. This identity agrees with direct recomputation from the retained groups. Excluding a large gain lowers the retained mean; excluding a small gain raises it. The deletion exercise changes neither the held-out questions nor the training conditions, and exposes precisely how the published mean depends on each of the four named seeds. A future replication can report additional seeds separately under the same paired metric definitions.

G Paired Outcome Decomposition

This section expands the paired-behavior analysis in Section 4. Every item belongs to exactly one of four correctness states, so the table partitions the evaluation set without dropping a verdict.

Table 10: **The two-order arms place many more items in the correct-under-both-layouts state.** CC means correct in both orientations; CW only in the original; WC only in the swapped; WW in neither. Counts sum to 449 within every row.

Data	Arm	Seed	CC	CW	WC	WW
RB	Fixed-order	42	157	290	0	2
RB	Fixed-order	123	54	395	0	0
RB	Fixed-order	456	146	301	0	2
RB	Fixed-order	789	201	238	0	10
RB	Two-order	42	364	21	33	31
RB	Two-order	123	366	35	19	29
RB	Two-order	456	360	30	28	31
RB	Two-order	789	363	26	29	31
RB	Consistency	42	367	19	25	38
RB	Consistency	123	357	29	30	33
RB	Consistency	456	367	39	14	29
RB	Consistency	789	368	25	23	33
RB	Pair-aware	42	368	15	30	36
RB	Pair-aware	123	368	30	18	33
RB	Pair-aware	456	361	34	21	33
RB	Pair-aware	789	353	39	26	31
UF	Fixed-order	42	64	379	0	6
UF	Fixed-order	123	16	432	0	1
UF	Fixed-order	456	55	388	0	6
UF	Fixed-order	789	85	355	0	9
UF	Two-order	42	287	58	50	54
UF	Two-order	123	277	101	19	52
UF	Two-order	456	284	87	26	52
UF	Two-order	789	276	94	22	57
UF	Consistency	42	292	68	36	53
UF	Consistency	123	289	72	34	54
UF	Consistency	456	273	106	20	50
UF	Consistency	789	288	72	34	55
UF	Pair-aware	42	291	60	42	56
UF	Pair-aware	123	286	82	31	50
UF	Pair-aware	456	290	78	34	47
UF	Pair-aware	789	278	98	27	46

Exact accounting. With counts $n_{CC}, n_{CW}, n_{WC}, n_{WW}$, OA accuracy equals $(2n_{CC} + n_{CW} + n_{WC})/(2n)$ and both-order correctness equals n_{CC}/n. These identities are verified directly against every seed summary. The partition explains how an average above 50% can coexist with much lower both-order correctness: items in the two one-correct states contribute half credit to OA and zero to Both.

Interpretation. CW and WC are orientation-specific correctness states, not assumed causes of error. A CW record can be compatible with a repeated first-slot verdict, but the state definition depends only on correctness. Predictions are retained in the CSV so that a slot-specific analysis can be performed separately. This keeps the decomposition exact while avoiding an unsupported attribution of every asymmetric judgment to the same mechanism.

Consistency is a separate partition. The correctness classes are exhaustive. Swap consistency instead compares predicted labels after the flip map: a pair can be wrong twice and still reverse its labels consistently, while C/C also contributes to consistency. Thus consistency cannot replace CC. Reporting both separates a stable comparative decision from a correct one without reclassifying model outputs.

H ITEM TRANSITIONS

This section expands the paired behavior interpretation in Section 4. It aligns Fixed-order and Two-order predictions for the same item and seed, then records how many of the two orientations each arm answers correctly.

Table 11: **Matched transitions locate where paired correctness is recovered.** Rows give the number of correct orientations under Fixed-order; columns give that number under Two-order. Entries count item–seed observations, with 1,796 observations per dataset and both layouts retained.

Dataset	Fixed correct	Two: 0	Two: 1	Two: 2	Total
RewardBench	0	8	4	2	14
RewardBench	1	114	209	901	1224
RewardBench	2	0	8	550	558
UF-clean	0	14	4	4	22
UF-clean	1	201	448	905	1554
UF-clean	2	0	5	215	220

Transition unit. Each observation is one item evaluated under one seed in both arms. The row and column indices take values zero, one, or two. A move from one to two correct orientations adds half a point to that observation's OA accuracy and one point to its both-order correctness. A move from zero to two adds one point to both quantities. The full matrix preserves every direction of change.

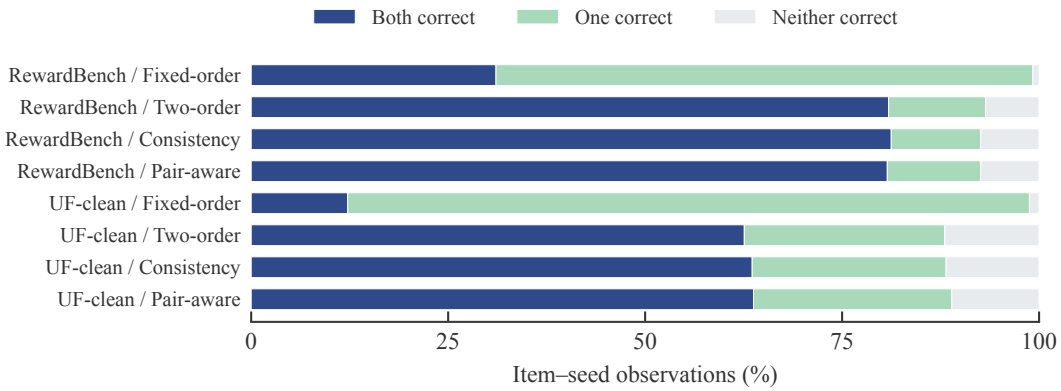

Figure 5: **Two-order training shifts mass toward correctness in both layouts.** Each horizontal bar partitions all 1,796 item–seed observations per dataset and arm into both correct, exactly one correct, and neither correct. Segment widths are percentages; the four seeds have equal counts.

Connection to the paired metrics. Figure 5 is a marginal distribution within each arm. The transition table contains additional information because it links the arms item by item. Marginals describe how much mass occupies a state; transitions describe which matched observations move between states. Both are obtained from the same predictions and require no additional model judgments.

Repeated questions. Pooling four seeds gives each item four observations. The bar describes the empirical distribution over the evaluated item–seed combinations. It does not imply 1,796 distinct questions; we resample complete questions for the interval in Appendix I. This retains the statistical unit while giving a compact descriptive display.

Accounting identity. If T_{ab} counts transitions from a correct orientations to b, the OA contrast is $\sum_{a,b}(b-a)T_{ab}/(2N)$, where $N = 1,796$. The both-order contrast is $\sum_{a,b}[\mathbf{1}(b=2) - \mathbf{1}(a=2)]T_{ab}/N$. These recover the unrounded main differences exactly and link the transition analysis to the primary estimands.

I Statistical Reproduction

This section expands the uncertainty and reproducibility discussion in Sections 3 and 4. The table gives a deterministic reproduction of the primary question-bootstrap contrast from the released paired records.

Table 12: **The reproduced question-bootstrap intervals support positive paired OA gains on both datasets.** Two-order-minus-Fixed-order differences are in percentage points. The percentile intervals resample complete questions, retaining all four seeds and both orientations. The analysis uses 10,000 replicates and random seed 20260926.

Dataset	Mean	2.5%	97.5%	Items	Replicates
RewardBench	21.91	19.57	24.17	449	10000
UF-clean	19.79	16.95	22.55	449	10000

Estimand and resampling. Define $d_{i,s}$ as the Two-order-minus-Fixed-order OA difference for item i and seed s. We first calculate $d_i = \frac{1}{4}\sum_s d_{i,s}$. For each bootstrap replicate, sample n item indices with replacement and average the corresponding d_i values. Both orientations, both compared arms, and all four observed seeds travel together whenever an item is sampled. The 2.5th and 97.5th percentiles form the reported interval.

$$\widehat{\Delta}^{*(b)} = \frac{1}{n}\sum_{j=1}^{n} d_{I_j^{(b)}}, \qquad I_j^{(b)} \sim \mathrm{Uniform}\{1,\ldots,n\}. \tag{7}$$

This is a conditional question bootstrap. It preserves training seeds and quantifies item-sampling variation on the paired evaluation set. The archived interval summaries in the source JSON files remain the record for the main-paper intervals. The new calculation uses an explicit fixed random seed and replicate count to provide an independently reproducible check of the same estimand.

Exact metric reproduction. The build script first reads both compressed CSV files, groups records by dataset, arm, and seed or by dose share and seed, and recomputes the metrics. It compares OA, Both, Consistency, and First with the corresponding entries in the released descriptive JSON at a tolerance of 10^{-9} percentage points. Rounding occurs only when emitting the LaTeX table. All displayed appendix numbers are generated from these verified arrays or from declared transformations of their item records.

Portable score schema. Each four-arm row contains dataset, item identifier, gold label, arm, seed, and the two predictions. Dose rows replace dataset and arm with study and preferred-first share. These label-only records reproduce the reported evaluation summaries without distributing question text or model-generated rationale. The original compressed inputs remain unchanged; the derived CSV includes a source comment before its header.

Reproduction sequence. Running `code/analyze_core_data.py` regenerates the descriptive JSON from the label files. Running `code/build_appendix.py` verifies those metrics, emits the full tables and derived CSV, computes the seed-deletion and bootstrap checks, and writes the appendix source. The manuscript includes that source after the reference list. The figure generator reads the same derived quantities for the paired-outcome display.

Identifier validation. Before subtraction, the appendix build checks that item identifiers agree across all arms within each dataset and across all dose shares for each seed. Every item–condition–seed key is unique. These checks protect the pairing operation: a correct numerical mean is insufficient if the underlying questions have been misaligned. The group-size checks reproduce the counts in Appendix A.

Scope of the released artifacts. The data package contains scored labels, per-seed summaries, and the source needed to recompute the derived statistics. The numerical tables in this appendix describe the evaluated four-arm and dose conditions. They can be reconstructed directly from the distributed files with no access to training checkpoints.