FAMILY-DEPENDENT CHANGES IN TOKEN-LEVEL PROMPT DEPENDENCE ACROSS POST-TRAINING CHECKPOINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-training is intended to make language models follow their inputs more closely, but whether it changes how strongly a stated fact supports the model's answer, and in which direction, remains unclear. Response-level evaluations do not answer this question, because they do not show how individual answer-token probabilities respond when only the stated fact changes. We study it with Token-Level Matched-Substitution Sensitivity: holding the gold response prefix and target token fixed, we replace the factual prompt with a relation-matched alternative and average the resulting log-probability difference over the answer span to obtain Answer Prompt Dependence (Answer PD). We compare Base and post-trained checkpoints from four model families (Qwen3-1.7B, Mistral-7B, SmolLM2-1.7B, and OLMo 2 7B) on CounterFact and ParaRel under one fixed plain-text prompt format. The direction of change depends on the family: on CounterFact, Qwen3 decreases Answer PD ($\Delta = -1.81$), whereas Mistral, SmolLM2, and OLMo 2 increase it ($+2.30$, $+0.75$, and $+3.78$). Across OLMo 2's released Base, SFT, DPO, and Instruct (RLVR) checkpoints, Answer PD rises at every stage on both datasets, and about two-thirds of the total increase occurs at SFT. In a paraphrase control that preserves the fact, wording sensitivity changes in the opposite direction to Answer PD in all three evaluated families, with paired intervals excluding zero for Qwen3 and SmolLM2. Leave-one-relation-out analyses preserve every evaluated aggregate direction. Post-training should therefore not be assumed to change factual prompt dependence uniformly; our results characterize released checkpoints under a fixed protocol, not causal effects of individual training stages.

1 INTRODUCTION

Language models choose each next token from two kinds of context: the prompt and the answer text that has already been written. Consider the prompt "The capital of France is ...". If the model scores the answer token "Paris" after the fixed answer text "The answer is", the probability of "Paris" depends on both the factual statement and the preceding answer text. We use x for the prompt, h_t for the reference-answer text fixed before the scored token, and y_t for the answer token. Whole-answer accuracy tells us whether the model produced the right answer, but not how changing one fact changes support for the same token.

Previous work shows that model behavior can change with wording, prompt templates, and factual associations (Chatterjee et al., 2024; Mizrahi et al., 2024; Jiang et al., 2020), but these studies usually score a complete answer or task. Post-training raises a further question. Instruction tuning and preference optimization are intended to make models follow their inputs more closely (Wei et al., 2022; Mishra et al., 2022; Sanh et al., 2022; Ouyang et al., 2022; Rafailov et al., 2023), yet analyses of post-trained models find that their effects on token-level predictions are uneven: differences between aligned and base models concentrate on a small subset of token positions (Lin et al., 2024; Qi et al., 2025), preference optimization can shift probability mass toward already-likely outputs (Ren and Sutherland, 2025), and reliance on supplied context can first rise and then fall during instruction tuning (Goyal et al., 2025). We therefore ask a narrower question: *with the answer context held fixed, does post-training change how strongly a stated fact supports the same answer token, and does it change it in the same direction across model families?*

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.

For a concrete contrast, replace "The capital of France is ..." with "The capital of Japan is ...". Both prompts express the same country-to-capital relation, but they use different subjects and have different correct answers. We call this a relation-matched factual substitution. We feed the same reference-answer prefix to the model and score the same next token under both prompts, rather than generating a new answer. This fixed-context procedure uses teacher forcing. We call the token-level comparison *Token-Level Matched-Substitution Sensitivity*.

We define the token-level signed difference as

$$\mathrm{PD}_t = \log p_\theta(y_t \mid x_{\mathrm{orig}}, h_t) - \log p_\theta(y_t \mid x_{\mathrm{swap}}, h_t).$$

For multi-token answers, we average PD_t over the gold answer span to obtain Answer Prompt Dependence (Answer PD). A positive value means that the original fact supports the fixed answer token more; a value near zero means that the substitution makes little difference; and a negative value means that the substituted fact supports the same token more. Section 3 states what Answer PD does and does not measure.

We also use a paraphrase control that changes the wording while preserving the fact and answer, which separates sensitivity to factual substitution from sensitivity to wording alone. We evaluate four model families under one shared plain-text prompt format (Plaintext B). For Qwen3, Mistral, and SmolLM2, we compare Base with the corresponding Post or Instruct endpoint. OLMo 2 7B additionally releases SFT and DPO checkpoints, and its final Instruct checkpoint is trained with RLVR on top of DPO (Team OLMo et al., 2024); we evaluate this staged series on both CounterFact and ParaRel. Our main findings are as follows.

The direction of change depends on the model family. On CounterFact, Answer PD decreases from Base to Post for Qwen3 ($\Delta = -1.81$), but increases from Base to Instruct for Mistral ($+2.30$), SmolLM2 ($+0.75$), and OLMo 2 ($+3.78$). The same endpoint directions are also observed on ParaRel, with similar magnitudes across the four families (§5.1). This consistency across datasets suggests that the contrast is not specific to CounterFact.

Within OLMo 2, most of the increase occurs at SFT. Mean Answer PD rises at every released stage on both datasets, with the largest change appearing between Base and SFT. On CounterFact, SFT accounts for about two-thirds of the total increase ($+2.59$ of $+3.78$), DPO contributes a further $+0.94$, and the final RLVR step adds $+0.25$. ParaRel shows a very similar stage-wise pattern (§5.2), indicating that most of the checkpoint-level increase is already present after SFT.

Factual and wording sensitivity move in opposite directions. In the paraphrase control, the fact and answer remain fixed while only the wording changes. The change in wording sensitivity has the opposite sign to the change in Answer PD for all three evaluated families. The paired intervals exclude zero for Qwen3 and SmolLM2, while Mistral shows little change in wording sensitivity (§5.4). This pattern separates sensitivity to factual substitution from sensitivity to surface wording.

Removing any single relation leaves the overall direction of each checkpoint contrast unchanged. Although some individual relations move differently from the aggregate trend, none of them is sufficient to reverse the overall result (§5.3).

Figure 1 summarizes the evaluation pipeline. We contribute (i) a controlled token-level measurement that isolates factual substitution under a fixed answer context, (ii) a four-family checkpoint comparison under a shared interface, including the staged OLMo 2 trajectory on two datasets, and (iii) robustness analyses across paraphrase, donor, relation-removal, and dataset variation. Because the evaluated models are released checkpoints rather than models trained under controlled interventions, these comparisons describe differences between checkpoints rather than causal effects of individual training stages; estimating such effects would require controlled retraining or quasi-experimental designs over checkpoints (Lesci et al., 2024).

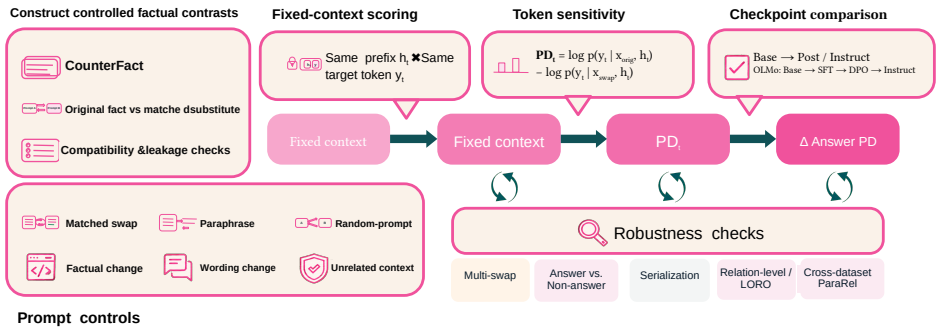

Figure 1: Overview of the evaluation framework. Relation-matched factual contrasts are scored with a fixed gold response prefix and target token to obtain Answer PD, followed by checkpoint comparison and robustness analyses across controls and datasets.

2 Related Work

Prompt sensitivity. Recent prompt-sensitivity work has moved beyond aggregate task accuracy toward instance-level and probability-based analyses. POSIX measures likelihood changes under rewording, while multi-prompt evaluation shows that phrasing can alter both measured performance and model rankings (Chatterjee et al., 2024; Mizrahi et al., 2024). ProSA makes this instance-level perspective explicit with PromptSensiScore and decoding-confidence analyses (Zhuo et al., 2024). The observed sensitivity also depends on evaluation choices such as likelihood scoring and answer matching (Hua et al., 2025), and meaning-preserving variants can be connected to gradients and next-token log probabilities (Liu and Chu, 2026). These studies vary the wording of a prompt; we instead hold the prompt template, response prefix, and scored token fixed and vary only the stated fact.

Factual knowledge probing. Factual-knowledge probing provides the setting for these comparisons (Youssef et al., 2023). LAMA tests factual completions with fill-in-the-blank queries (Petroni et al., 2019), CounterFact provides subject–relation–object records used in factual analysis and editing (Meng et al., 2022), ParaRel tests consistency across alternative relation phrasings (Elazar et al., 2021), and temporally consistent probing extends this question to time-indexed queries (Bajpai et al., 2024). Probing usually asks whether a model retrieves the correct object; we instead use CounterFact and ParaRel to build relation-matched pairs whose original and substituted facts compete for the same answer token. Neither dataset is used to modify model parameters.

Post-training methods. Post-training aims to change how pretrained models follow instructions, use context, and align responses. Work on instruction tuning studies both task-following gains and behavioral or internal shifts after tuning (Ghosh et al., 2024; Wu et al., 2024), and the composition of context-augmented instruction data can change how models rely on supplied context versus parametric knowledge (Lee et al., 2026). LIMA motivates the hypothesis that alignment can impose response formats with relatively little supervised data, whereas DPO provides a direct preference-optimization route without a separate reward-model stage (Zhou et al., 2023; Rafailov et al., 2023). We study released checkpoints of Qwen3, Mistral 7B, SmolLM2, and OLMo 2 (Yang et al., 2025; Jiang et al., 2023; Ben Allal et al., 2025; Team OLMo et al., 2024).

How post-training changes model predictions. A growing line of work analyzes what post-training changes rather than how well post-trained models perform. Lin et al. (2024) find that the token distributions of aligned and base models differ mainly at a small subset of positions, and Qi et al. (2025) show that the divergence introduced by safety alignment is concentrated in the first few output tokens. Ren and Sutherland (2025) decompose the step-wise learning dynamics of SFT and DPO, showing that SFT can raise the probability of responses written for other questions and that off-policy DPO squeezes probability mass toward already-likely outputs. Goyal et al. (2025) find that context reliance under knowledge conflict first rises and then falls during instruction tuning, and trace the decline to examples whose answers are already predictable without the context, identified

by their target perplexity when the context is removed. At the level of released checkpoints, Yue et al. (2025) compare base and RLVR-trained models across families and find that RLVR mainly sharpens the base distribution, and Kim et al. (2025) track a knowledge-related diagnostic across OLMo pre-training checkpoints and relate it to knowledge acquisition. These studies examine safety, response confidence, context reliance, reasoning coverage, or parametric memory use. Our estimand is complementary and narrower: the fixed-prefix support that a stated fact provides for the same answer token, compared across post-training checkpoints of several families under one scoring protocol.

3 METHOD

3.1 FROM NEXT-TOKEN PREDICTION TO A CONTROLLED PROMPT COMPARISON

Language models predict a response one token at a time. Let x denote the factual prompt and let $y = (y_1, \ldots, y_T)$ denote the reference response. For a target token y_t, let $h_t = y_{<t}$ be the response prefix that comes before that token. This prefix is called the gold response prefix because it is taken from the reference answer, not generated by the model. The model therefore assigns the target token a conditional probability $p_\theta(y_t \mid x, h_t)$ given both the prompt and the already supplied answer text.

The same token-level conditional likelihood is the quantity that supervised fine-tuning (SFT) optimizes:

$$\mathcal{L}_{\mathrm{SFT}} = -\mathbb{E}_{(x,y)}\left[\sum_t \log p_\theta(y_t \mid x, h_t)\right], \qquad h_t = y_{<t}.$$

This connection explains why the direction in which post-training changes our score is not fixed in advance. SFT directly raises $\log p_\theta(y_t \mid x, h_t)$ for training pairs, but learning one response also changes the model's predictions for other prompts: Ren and Sutherland (2025) show that SFT can increase the probability of responses written for other questions, and that off-policy preference optimization pushes probability mass toward already-likely outputs. For a fixed answer token scored under a mismatched prompt, the first effect can raise its support and the second can lower it. Which effect dominates for a released checkpoint is therefore an empirical question; our diagnostic measures the outcome without retraining models or attributing it to individual optimization steps.

For interpretation, the conditional log probability decomposes as

$$\log p_\theta(y_t \mid x, h_t) = \underbrace{\log p_\theta(y_t \mid h_t)}_{\text{prefix-only support}} + \underbrace{\log \frac{p_\theta(y_t \mid x, h_t)}{p_\theta(y_t \mid h_t)}}_{\text{prompt-associated log-ratio}}. \tag{1}$$

The first term is the support that the response prefix provides on its own. The second has the same form as the per-example terms of conditional cross-mutual information, which measures context usage by comparing one model's log probabilities with and without a context (Fernandes et al., 2021). It is a property of the model's predictions, not an estimate of mutual information in the data. Our score does not compare a prompt with its absence; it compares two prompts, so the prefix-only term cancels (Eq. 2).

3.2 TOKEN-LEVEL MATCHED-SUBSTITUTION SENSITIVITY

We first describe the comparison with a concrete example. An original prompt might be "The capital of France is ...", while a matched substitute is "The capital of Japan is ...". Both prompts express the country–capital relation, but they change the subject and the associated answer: France is paired with Paris, whereas Japan is paired with Tokyo. In general, x_{orig} denotes the original factual prompt and x_{swap} denotes a relation-matched factual substitute that preserves the relation while changing the subject and answer.

For the primary comparison, we choose a target token from the original reference answer and use the same target token y_t and the same gold response prefix h_t under both prompts. Teacher forcing means that this prefix is supplied directly to the model before the target token is scored. The model is not asked to generate a response. This is a scoring procedure rather than a free-generation evaluation.

It may seem unusual to score the token "Paris" under the Japan prompt, where the correct answer would be Tokyo. That is intentional. We are not asking what answer the model would freely generate

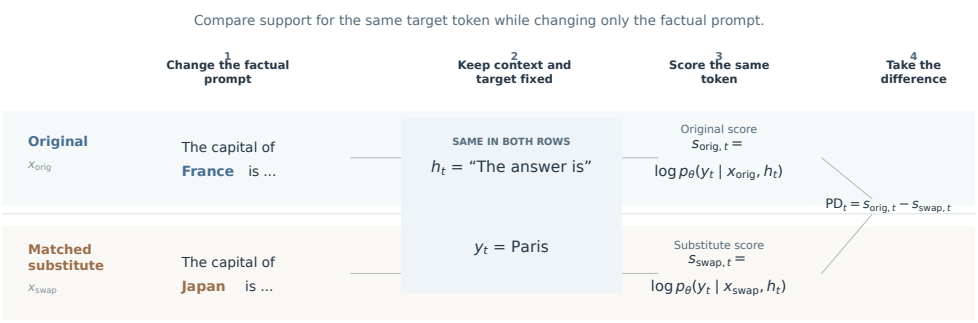

Figure 2: Token-Level Matched-Substitution Sensitivity. The factual prompt changes while the gold response prefix h_t and target token y_t remain fixed. Under teacher forcing, the model scores the same target token under the original and matched-substitute prompts; their signed conditional log-probability difference defines token-level PD_t.

for Japan. We are asking: if the target remains Paris and the preceding answer context remains exactly the same, how much does the model's support for Paris change when the factual prompt is changed from France to Japan? The only systematic change in this paired comparison is the factual prompt.

Figure 2 summarizes this paired scoring procedure.

We call the signed difference

$$\mathrm{PD}_t = \log p_\theta(y_t \mid x_{\mathrm{orig}}, h_t) - \log p_\theta(y_t \mid x_{\mathrm{swap}}, h_t)$$

token-level Prompt Dependence (PD). A positive value means that the original prompt gives greater support to the fixed target token, a value near zero means that the substitution changes that support little, and a negative value means that the substitute prompt gives greater support to the same fixed token. Substituting Eq. 1 gives

$$\mathrm{PD}_t = \log \frac{p_\theta(y_t \mid x_{\mathrm{orig}}, h_t)}{p_\theta(y_t \mid h_t)} - \log \frac{p_\theta(y_t \mid x_{\mathrm{swap}}, h_t)}{p_\theta(y_t \mid h_t)}, \tag{2}$$

so PD contrasts the prompt-associated log-ratios of the two facts and does not contain the prefix-only term.

What PD does and does not measure. Three properties hold by construction. First, PD does not depend on how strongly the prefix alone predicts y_t. Second, negative PD is a valid observation rather than an error. Third, for a fixed model, PD is the effect of an input-level substitution on the log probability of the original answer token; the substitution replaces one subject with another of the same relation, the symmetric token replacement used by Zhang and Nanda (2024) to study factual recall, which keeps the altered prompt in distribution. Two further properties must be checked rather than assumed. Because PD is measured in nats, a checkpoint whose predictive distributions become sharper can show larger PD without any fact-specific change; the random-prompt control in Table 1 addresses this. And differences in PD between released checkpoints describe those checkpoints: they do not identify the effect of any training stage, and PD does not indicate where factual knowledge is stored.

3.3 Answer-Level and Checkpoint-Level Comparison

An answer can contain one token or several tokens. Let $\mathcal{A}$ be the set of tokens in the gold answer span. We average token-level PD within each sample to obtain the sample-level Answer PD:

$$\text{Answer PD} = \frac{1}{|\mathcal{A}|} \sum_{t \in \mathcal{A}} \mathrm{PD}_t, \qquad \text{where } \mathcal{A} \text{ is the gold answer-token set.}$$

The token average is performed inside each sample first. Samples, rather than individual tokens, are the units used for means, paired bootstrap intervals, and checkpoint contrasts; this avoids treating

Table 1: Auxiliary checks. Each check changes one element of the primary comparison; the last column states the outcome that would undermine the primary reading.

Check	What changes	Outcome that would undermine the primary reading
Paraphrase	Wording changes; fact and answer fixed	Wording sensitivity changes in the same way as Answer PD
Random prompt	Substitute taken from an unrelated relation	Answer PD under unrelated prompts changes as much as under matched substitutes
Multi-swap	Up to three donors, averaged in probability space	The checkpoint difference depends on a particular donor
Answer vs. non-answer	PD averaged over non-answer response tokens	Substitution effects are spread evenly over the response
New target	The substitute answer is scored under the substitute prompt	The original-answer effect is not mirrored for the substitute answer

tokens from the same answer as independent observations. We average rather than sum so that answers with different numbers of tokens contribute on the same scale. Under teacher forcing, later answer tokens can often be predicted from earlier answer tokens, so their PD is typically smaller than that of the first answer token; because tokenizers also differ across families, we compare the direction and size of within-family changes rather than absolute Answer PD across families.

The checkpoint change discussed in the results is

$$\Delta \,\text{Answer PD} = \text{Answer PD}_{\text{later}} - \text{Answer PD}_{\text{base}}.$$

For Qwen3, Mistral, and SmolLM2, later denotes the released Post or Instruct endpoint. For OLMo 2, we compare Base, SFT, DPO, and Instruct (RLVR) and also report adjacent checkpoint contrasts.

Shared serialization. All checkpoints are scored with the same plain-text wrapper (Appendix B.1), including post-trained checkpoints that are usually queried through a chat template. Applying each checkpoint's own template would change the input between the Base and later checkpoints, so a checkpoint contrast would mix changes in the weights with changes in the serialization. A shared wrapper keeps the input identical across checkpoints, following the practice of evaluating base and post-trained models with the same prompt (Yue et al., 2025). The cost is that post-trained checkpoints are evaluated outside their usual interface, which limits the scope of our conclusions (Section 6).

Controls. Each auxiliary check changes one element of the primary comparison and asks whether the checkpoint difference survives (Table 1). Exact definitions are given in Appendix A; none of them changes the primary PD definition or the primary sample set.

4 Experimental Setup

Table 2 summarizes the checkpoints, datasets, and analyses; details follow.

Models. We select families that release a pretrained base checkpoint together with a post-trained counterpart, so that each contrast compares two checkpoints derived from the same base model: Qwen3-1.7B (Base→Post), Mistral-7B-v0.3 (Base→Instruct), SmolLM2-1.7B (Base→Instruct), and OLMo 2 7B (Yang et al., 2025; Jiang et al., 2023; Ben Allal et al., 2025; Team OLMo et al., 2024). We use OLMo 2 for the staged analysis because it also releases its intermediate post-training checkpoints: SFT, DPO, and the final Instruct model, which is trained with RLVR on top of DPO. Exact revisions are listed in Appendix B.3.

Data. The primary evaluation uses 252 relation-matched pairs from CounterFact covering 24 relations (Meng et al., 2022). A pair is retained only if the substitute expresses the same relation with a different subject and answer, neither prompt contains the target answer, and the answer span aligns

Table 2: Checkpoints and analysis coverage. Auxiliary diagnostics are non-answer PD, the new-target check, and multi-swap donor robustness.

Family	Checkpoints compared	CounterFact	ParaRel	Paraphrase	Auxiliary
Qwen3-1.7B	Base → Post	✓	✓	✓	✓
Mistral-7B-v0.3	Base → Instruct	✓	✓	✓	✓
SmolLM2-1.7B	Base → Instruct	✓	✓	✓	✓
OLMo 2 7B	Base → SFT → DPO → Instruct (RLVR)	✓	✓	–	–

with the response tokens. ParaRel provides a second evaluation of 230 pairs over the 23 relations it shares with CounterFact, balanced at ten pairs per relation and excluding facts that appear in the CounterFact set (Elazar et al., 2021). Cross-dataset comparisons restrict CounterFact to the same 23 relations. Construction details are in Appendix B.2.

Scoring. Every checkpoint is scored with the same plain-text wrapper (Appendix B.1) in bfloat16, with the fixed response "The answer is *answer*" supplied under teacher forcing. Answer tokens are the response tokens whose character spans overlap the verified answer span. For the paraphrase control, sample-level Answer PS is the absolute difference between the mean answer-token log probabilities under the original and paraphrased prompts; larger values indicate greater sensitivity to wording. The paraphrase set contains 83 verified pairs.

Statistics. The statistical unit is the sample. Checkpoint contrasts use paired percentile bootstrap intervals with $B = 5000$ resamples of complete paired examples. We also report direct relation-level changes and leave-one-relation-out (LORO) aggregates, which recompute the sample-weighted mean after removing one complete relation. Seeds and implementation details are in Appendix B.4.

5 Results

We first report how Answer PD changes between released checkpoints (§5.1–§5.2), then test whether these changes are specific to the stated fact (§5.3), and finally compare them with changes in wording sensitivity (§5.4). Table 3 summarizes the main contrasts.

Table 3: Main Answer PD results across model families, datasets, and controls. Δ is the later-minus-earlier checkpoint change; intervals are paired sample-level 95% bootstrap CIs.

Four-family endpoint comparisons

		CounterFact				ParaRel			
Family	Endpoint	Base	Later	Δ	95% CI	Base	Later	Δ	95% CI
Qwen3	B→Post	**3.38**	1.58	−1.81	[−2.18, −1.43]	**3.45**	1.70	−1.75	[−2.12, −1.40]
Mistral	B→Inst.	**3.25**	**5.55**	**+2.30**	[+2.01, +2.59]	**3.39**	**5.36**	**+1.97**	[+1.70, +2.25]
SmolLM2	B→Inst.	3.10	**3.85**	**+0.75**	[+0.51, +0.99]	3.33	**4.02**	**+0.69**	[+0.46, +0.92]
OLMo 2 7B	B→Inst.	**3.73**	**7.51**	**+3.78**	[+3.34, +4.20]	**4.02**	**7.85**	**+3.83**	[+3.37, +4.28]

OLMo 2 7B staged comparisons

Contrast	CounterFact				ParaRel			
	Δ	95% CI	Rel.	LORO	Δ	95% CI	Rel.	LORO
SFT - Base	**+2.59**	[+2.27, +2.91]	23/24	24/24	**+2.63**	[+2.27, +3.00]	23/23	23/23
DPO - Base	**+3.53**	[+3.11, +3.92]	23/24	24/24	**+3.58**	[+3.14, +4.00]	22/23	23/23
Instruct - Base	**+3.78**	[+3.34, +4.20]	23/24	24/24	**+3.83**	[+3.37, +4.28]	22/23	23/23
DPO - SFT	+0.94	[+0.80, +1.08]	23/24	24/24	+0.95	[+0.81, +1.09]	22/23	23/23
Instruct - DPO	+0.25	[+0.20, +0.30]	23/24	24/24	+0.25	[+0.21, +0.30]	23/23	23/23

Paraphrase control

Family	Factual Δ	Paraphrase Δ	95% CI
Qwen3	**-1.81**	**+0.46**	[+0.19, +0.75]
Mistral	**+2.30**	**-0.04**	[−0.28, +0.21]
SmolLM2	**+0.75**	**-0.44**	[−0.73, −0.17]

Note. Rel. and LORO report positive relation-level changes and positive leave-one-relation-out runs, respectively. Boldface highlights Δ values, not better performance. OLMo 2 7B was not evaluated in the paraphrase control.

5.1 The Direction of Change Depends on the Model Family

On CounterFact, Answer PD decreases from Base to Post for Qwen3 ($\Delta = -1.81$, 95% CI $[-2.18, -1.43]$), but increases from Base to Instruct for Mistral ($+2.30$, $[+2.01, +2.59]$), SmolLM2 ($+0.75$, $[+0.51, +0.99]$), and OLMo 2 ($+3.78$, $[+3.34, +4.20]$; Figure 3). ParaRel shows the same four directions with similar magnitudes (Qwen3 -1.75, Mistral $+1.97$, SmolLM2 $+0.69$, OLMo 2 $+3.83$). At the relation level, 23 of 24 CounterFact relations decrease for Qwen3, while 24, 18, and 23 of 24 increase for Mistral, SmolLM2, and OLMo 2, respectively (Appendix Tables B1–B2).

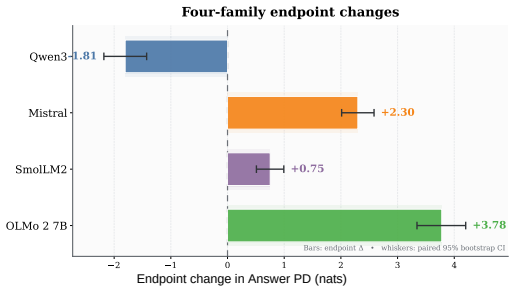

Figure 3: Base-to-later change in mean Answer PD on CounterFact. Qwen3 moves in the opposite direction to the other three families, and every paired 95% bootstrap interval (whiskers) excludes zero. The dashed line marks zero.

5.2 Within OLMo 2, Most of the Increase Occurs at SFT

Mean Answer PD rises at every released OLMo 2 checkpoint on both datasets, and all three adjacent paired intervals lie above zero (Appendix Table B3). On CounterFact, SFT contributes +2.59 of the total +3.78 increase (68%), DPO adds +0.94 (25%), and the final RLVR step adds +0.25 (7%); ParaRel shows a nearly identical sequence (+2.63, +0.95, +0.25). At the sample level, DPO-to-Instruct changes are closer to zero than the earlier transitions (Figure 4b).

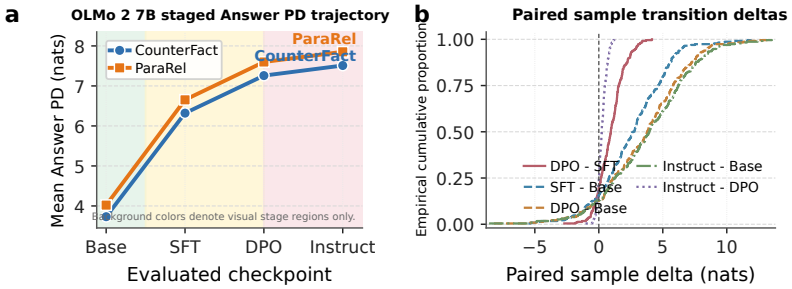

Figure 4: OLMo 2 7B across released checkpoints. (a) Mean Answer PD rises at every stage on CounterFact and ParaRel, with the largest step at SFT. (b) Cumulative distributions of per-example changes; DPO-to-Instruct changes cluster near zero. Adjacent paired intervals are reported in Appendix Table B3.

5.3 Is the Change Specific to the Stated Facts?

A checkpoint difference in Answer PD could also arise if a single relation dominated the average, if a particular substitute drove the result, or if post-training changed support for the whole response rather than for the answer. We test each possibility.

Is the result driven by a single relation? Removing any one relation leaves the sign of every aggregate contrast unchanged. Qwen3 remains negative and Mistral and SmolLM2 remain positive in all 24 CounterFact runs, and all five OLMo 2 contrasts remain positive in all 24 CounterFact and 23 ParaRel runs; for example, the OLMo 2 SFT$-$Base contrast ranges from $+2.42$ to $+2.73$ across held-out relations (Appendix Figure B4; Tables B7–B8).

Does it depend on the choice of substitute? Largely not. Among the 174 examples with at least two legal donors, single- and multi-substitute Answer PD are strongly correlated (Spearman $\rho = 0.85$) and agree in sign for 90.2% of examples (Appendix Figure B1b).

Is it confined to the answer span? The change is concentrated in the answer span. Over the non-answer response tokens, PD changes between checkpoints by at most 0.064 nats in absolute value across the three endpoint families and both datasets, compared with 0.69 to 2.30 nats for Answer PD (Appendix Table B5).

Is it mirrored for the substitute answer? Yes. The substitute answer shows changes in the same direction and of similar size in all six family–dataset pairs, including -1.80 for Qwen3 and $+1.98$ for Mistral on CounterFact (Appendix Table B6).

5.4 Factual and Wording Sensitivity Move in Opposite Directions

The paraphrase control changes only the wording while keeping the fact and answer fixed. Across 83 verified pairs, wording sensitivity increases for Qwen3 (Δ PS = +0.46, [+0.19, +0.75]), decreases for SmolLM2 (-0.44, [-0.73, -0.17]), and changes little for Mistral (-0.04, [-0.28, +0.21]; Figure 5). These directions are opposite to the corresponding Answer PD changes. OLMo 2 was not evaluated in this control.



Figure 5: Paraphrase control for Qwen3, Mistral, and SmolLM2. (a) Base-to-later change in mean Answer PS with paired 95% bootstrap intervals. Qwen3 and SmolLM2 move opposite to their Answer PD changes; Mistral's interval includes zero. (b) Per-example change distributions; the dashed line marks zero.

6 Discussion and Limitations

Answer PD changes differently across the evaluated checkpoints: under Plaintext B, it decreases for Qwen3 but increases for Mistral, SmolLM2, and OLMo 2. The same endpoint directions hold on ParaRel and under leave-one-relation-out analysis. The paraphrase control uses a smaller subset without staged OLMo, so we treat it only as a check on wording sensitivity.

Answer PD measures how a fixed answer token responds to factual-prompt substitution with the response prefix fixed. It does not measure free-generation accuracy or the amount or location of factual knowledge, and it depends on the prompt interface.

These checkpoint comparisons are descriptive rather than causal because the released checkpoints are not controlled interventions. The OLMo Base–SFT–DPO–Instruct sequence therefore shows checkpoint differences, not stage effects. Our conclusions are limited to the evaluated datasets, checkpoints, and interfaces.

7 Conclusion

Our results show that factual-prompt dependence does not change uniformly across post-trained checkpoints. Under the same scoring protocol, different model families show different changes in Answer PD, while the paraphrase control shows a different pattern on the verified subset. These findings describe the evaluated checkpoints and should not be interpreted as causal effects of individual training stages.

AI USE STATEMENT

Generative AI tools were used for code development and debugging, figure refinement, literature search, and manuscript editing. The authors reviewed the resulting code, numerical results, citations, figures, and text before submission.

REPRODUCIBILITY STATEMENT

The released package records the evaluated checkpoint revisions, analysis scripts, processed data, and environment information used for the reported results. Evaluation uses teacher forcing with fixed gold response prefixes and target tokens. OLMo 2 is evaluated across Base, SFT, DPO, and Instruct checkpoints on both CounterFact and ParaRel, and the LORO analysis uses the same sample-level scoring procedure as the endpoint comparisons.

REFERENCES

Ashutosh Bajpai, Aaryan Goyal, Atif Anwer, and Tanmoy Chakraborty. Temporally consistent factuality probing for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 15864–15881. Association for Computational Linguistics, Miami, Florida, USA, 2024. doi: 10.18653/v1/2024.emnlp-main.887. URL https://aclanthology.org/2024.emnlp-main.887/.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. Smollm2: When smol goes big – data-centric training of a small language model. *arXiv preprint arXiv:2502.02737*, 2025. URL https://arxiv.org/abs/2502.02737.

Anwoy Chatterjee, H S V N S Kowndinya Renduchintala, Sumit Bhatia, and Tanmoy Chakraborty. POSIX: A prompt sensitivity index for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 14550–14565, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.852. URL https://aclanthology.org/2024.findings-emnlp.852/.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031, 2021. doi: 10.1162/tacl_a_00410. URL https://aclanthology.org/2021.tacl-1.60/.

Patrick Fernandes, Kayo Yin, Graham Neubig, and André F. T. Martins. Measuring and increasing context usage in context-aware machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6467–6478. Association for Computational Linguistics, Online, 2021. doi: 10.18653/v1/2021.acl-long.505. URL https://aclanthology.org/2021.acl-long.505/.

Sreyan Ghosh, Chandra Kiran Reddy Evuru, Sonal Kumar, Ramaneswaran S, Deepali Aneja, Zeyu Jin, Ramani Duraiswami, and Dinesh Manocha. A closer look at the limitations of instruction tuning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15559–15589. PMLR, 2024. URL https://proceedings.mlr.press/v235/ghosh24a.html.

Sachin Goyal, Christina Baek, J. Zico Kolter, and Aditi Raghunathan. Context-parametric inversion: Why instruction finetuning can worsen context reliance. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=SPS6HzVzyt.

Andong Hua, Kenan Tang, Chenhe Gu, Jindong Gu, Eric Wong, and Yao Qin. Flaw or artifact? Rethinking prompt sensitivity in evaluating LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 19889–19899. Associa-

tion for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1006. URL https://aclanthology.org/2025.emnlp-main.1006/.

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

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020. doi: 10.1162/tacl_a_00324. URL https://aclanthology.org/2020.tacl-1.28/.

Jiyeon Kim, Hyunji Lee, Hyowon Cho, Joel Jang, Hyeonbin Hwang, Seungpil Won, Youbin Ahn, Dohaeng Lee, and Minjoon Seo. Knowledge entropy decay during language model pretraining hinders new knowledge acquisition. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=eHehzSDUFp.

Hyunji Lee, Seunghyun Yoon, Yunjae Won, Hanseok Oh, Geewook Kim, Trung Bui, Franck Dernoncourt, Elias Stengel-Eskin, Mohit Bansal, and Minjoon Seo. Instruction tuning with and without context: Behavioral shifts and downstream impact. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6790–6810. Association for Computational Linguistics, Rabat, Morocco, 2026. doi: 10.18653/v1/2026.eacl-long.320. URL https://aclanthology.org/2026.eacl-long.320/.

Pietro Lesci, Clara Meister, Thomas Hofmann, Andreas Vlachos, and Tiago Pimentel. Causal estimation of memorisation profiles. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15616–15635. Association for Computational Linguistics, Bangkok, Thailand, 2024. doi: 10.18653/v1/2024.acl-long.834. URL https://aclanthology.org/2024.acl-long.834/.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. The unlocking spell on base LLMs: Rethinking alignment via in-context learning. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=wxJ0eXwwda.

Yang Liu and Chenhui Chu. Understanding the prompt sensitivity. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 44362–44388. Association for Computational Linguistics, San Diego, California, 2026. doi: 10.18653/v1/2026.acl-long.2053. URL https://aclanthology.org/2026.acl-long.2053/.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 8086–8098. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.556. URL https://aclanthology.org/2022.acl-long.556/.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Swaroop Mishra, Oana-Maria Camburu, Tobias Gritschneder et al. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 3470–3487. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.244. URL https://aclanthology.org/2022.acl-long.244/.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. State of what art? a call for multi-prompt LLM evaluation. *Transactions of the Association for Computational Linguistics*, 12:933–949, 2024. doi: 10.1162/tacl_a_00681. URL https://aclanthology.org/2024.tacl-1.52/.

Long Ouyang, Jeffrey Wu, Xu Jiang et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2203.02155.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 2463–2473, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1250. URL https://aclanthology.org/D19-1250/.

Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek Mittal, and Peter Henderson. Safety alignment should be made more than just a few tokens deep. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=6Mxhg9PtDE.

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

Yi Ren and Danica J. Sutherland. Learning dynamics of LLM finetuning. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=tPNHOoZFl9.

Victor Sanh, Albert Webson, Colin Raffel et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2022. URL https://arxiv.org/abs/2110.08207.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Michal Guerquin, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 OLMo 2 furious. *arXiv preprint arXiv:2501.00656*, 2024. URL https://arxiv.org/abs/2501.00656.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi et al. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5085–5109. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.340. URL https://aclanthology.org/2022.emnlp-main.340/.

Jason Wei, Maarten Bosma, Vincent Y. Zhao et al. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

Xuansheng Wu, Wenlin Yao, Jianshu Chen, Xiaoman Pan, Xiaoyang Wang, Ninghao Liu, and Dong Yu. From language modeling to instruction following: Understanding the behavior shift in LLMs after instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 2341–2369. Association for Computational Linguistics, Mexico City, Mexico, 2024. doi: 10.18653/v1/2024.naacl-long.130. URL https://aclanthology.org/2024.naacl-long.130/.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Paul Youssef, Osman Koraš, Meijie Li, Jörg Schlötterer, and Christin Seifert. Give me the facts! A survey on factual knowledge probing in pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15588–15605. Association for Computational Linguistics, Singapore, 2023. doi: 10.18653/v1/2023.findings-emnlp.1043. URL https://aclanthology.org/2023.findings-emnlp.1043/.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1933. URL https://papers.nips.cc/paper_files/paper/2025/hash/537d5aa768c2d534016a4d06f87bc8fb-Abstract-Conference.html.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=9eJv5PS27Q.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2400. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/ac662d74829e4407ce1d126477f4a03a-Abstract-Conference.html.

Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. ProSA: Assessing and understanding the prompt sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1950–1976. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.108. URL https://aclanthology.org/2024.findings-emnlp.108/.

A Theoretical and Diagnostic Details

A.1 Interpreting the Token-Level Conditional Score

The conditional log-probability used in the main text admits the following algebraic decomposition:

$$\log p_\theta(y_t \mid x, h_t) = \log p_\theta(y_t \mid h_t) + \log \frac{p_\theta(y_t \mid x, h_t)}{p_\theta(y_t \mid h_t)}.$$

The first term is the response-prefix contribution: it captures support for the next token that is available from the already supplied response text. The log-ratio is the additional contribution associated with including the prompt under that same prefix. This is a conceptual interpretation of an algebraic identity, not a separate estimator used in the experiments. Under an explicitly defined joint distribution, the log-ratio has the form of a pointwise conditional information quantity. However, the present experiments do not estimate mutual information, and the main PD metric does not require this interpretation.

A.2 Answer / Non-answer Selectivity

The primary analysis averages PD over the gold answer span. As a descriptive check, we also compare that average with the average over the remaining response tokens. Let $\mathcal{A}$ be the answer-token set and let $\mathcal{N}$ be the non-answer response-token set for the same sample:

$$\mathrm{PD}_{\mathrm{answer}} = \frac{1}{|\mathcal{A}|} \sum_{t \in \mathcal{A}} \mathrm{PD}_t, \qquad \mathrm{PD}_{\mathrm{nonanswer}} = \frac{1}{|\mathcal{N}|} \sum_{t \in \mathcal{N}} \mathrm{PD}_t.$$

We define the corresponding selectivity contrast as

$$\mathrm{Selectivity} = \mathrm{PD}_{\mathrm{answer}} - \mathrm{PD}_{\mathrm{nonanswer}}.$$

This check asks whether the factual substitution changes answer tokens more than nearby response tokens on average. A positive value does not show that only answer tokens are affected and does not isolate factual information from all other prompt effects.

A.3 Old-Target / New-Target Directional Check

The old/new-target check uses two answers associated with the same relation-matched pair. For example, the original prompt may express France $\rightarrow$ Paris, while the substitute prompt expresses Japan $\rightarrow$ Tokyo. Here Paris is the old target and Tokyo is the new target. The old-target quantity asks whether the France prompt supports Paris more than the Japan prompt does:

$$D_{\mathrm{old},t} = \log p_\theta(y_{\mathrm{old},t} \mid x_{\mathrm{orig}}, h_t^{\mathrm{old}}) - \log p_\theta(y_{\mathrm{old},t} \mid x_{\mathrm{swap}}, h_t^{\mathrm{old}}).$$

The new-target quantity reverses the prompt comparison and asks whether the Japan prompt supports Tokyo more than the France prompt does:

$$G_{\mathrm{new},t} = \log p_\theta(y_{\mathrm{new},t} \mid x_{\mathrm{swap}}, h_t^{\mathrm{new}}) - \log p_\theta(y_{\mathrm{new},t} \mid x_{\mathrm{orig}}, h_t^{\mathrm{new}}).$$

The prefixes h_t^{old} and h_t^{new} are the gold prefixes from the corresponding original-answer and substitute-answer responses. Positive values in the respective directions are the intuitive pattern for this sanity check. The old-target quantity for the original-answer response coincides with the primary Answer PD by construction; the new-target quantity is complementary and does not filter samples or redefine the primary analysis.

A.4 Multi-Swap Donor Robustness

The primary analysis uses one legal relation-matched donor for each original sample. To check whether a result depends on that particular donor, we use up to $K = 3$ legal substitute prompts for the same original sample. Multiple substitutes are first scored separately for the same target token and the same gold response prefix. If x_{s_k} is substitute k, its log probability is

$$\ell_{s_k,t} = \log p_\theta(y_t \mid x_{s_k}, h_t).$$

We then average the substitute probabilities, rather than averaging log probabilities or creating new independent samples:

$$\log \bar{p}_{\text{swap},t} = \text{logsumexp}(\ell_{s_1,t}, \ldots, \ell_{s_K,t}) - \log K = \log \left(\frac{1}{K} \sum_{k=1}^{K} e^{\ell_{s_k,t}} \right).$$

The resulting donor-averaged score is

$$\text{PD}_t^{\text{multi}} = \log p_\theta(y_t \mid x_{\text{orig}}, h_t) - \log \bar{p}_{\text{swap},t}.$$

Thus, multi-swap aggregation changes the substitute-side score for one original sample. It does not turn the K substitutes into K independent original examples; all subsequent sample-level summaries still use the original sample as the statistical unit.

A.5 Paraphrase and Random-Prompt Controls

The paraphrase control changes the wording of a factual prompt while preserving its fact and answer. It uses the same gold-prefix teacher-forcing procedure as the primary comparison, but replaces the relation-matched factual substitute with a meaning-preserving paraphrase x_{para}.

For the answer-token set $\mathcal{A}$, sample-level Answer PS is defined as the absolute difference between the mean answer-token log probabilities under the original and paraphrased prompts:

$$\text{PS}_{\text{answer}} = \left| \frac{1}{|\mathcal{A}|} \sum_{t \in \mathcal{A}} \log p_\theta(y_t \mid x_{\text{orig}}, h_t) - \frac{1}{|\mathcal{A}|} \sum_{t \in \mathcal{A}} \log p_\theta(y_t \mid x_{\text{para}}, h_t) \right|.$$

Thus, the absolute value is applied after averaging the answer-token log probabilities within each prompt condition. This sample-level quantity is the Answer PS used in the reported paraphrase-control results.

The random-prompt control uses its own verified population and estimand; neither control changes the primary relation-matched PD sample.

A.6 Relation-Level Aggregation and Leave-One-Relation-Out

After averaging over the answer tokens of each sample, let d_i denote the later-minus-earlier Answer PD change for sample i, and let $r(i)$ denote its relation. The overall checkpoint contrast is the sample-level mean

$$\Delta = \frac{1}{N} \sum_{i=1}^{N} d_i.$$

For a relation r, the direct relation delta is computed only from the samples assigned to that relation:

$$\Delta_r = \text{mean}_{i:r(i)=r}(d_i).$$

The leave-one-relation-out (LORO) value removes the complete relation and recomputes the aggregate from the remaining samples:

$$\Delta_{-r} = \text{mean}_{i:r(i)\neq r}(d_i).$$

Thus, the direct relation delta describes the relation being inspected, whereas Δ_{-r} asks whether the aggregate direction survives its removal. The implementation checks sample-ID, relation, and gold-answer parity before forming these quantities. LORO uses the sample-weighted mean over the remaining examples; it does not reweight relations and does not add a second bootstrap layer. We call the direction preserved when the sign of Δ_{-r} is the same as the sign of the full Δ for every held-out relation.

A.7 Scope and Interpretation of the Diagnostic

Answer PD measures a change in conditional support for the same target token when the factual prompt is replaced while the response prefix is held fixed. It is therefore a token-level, interface-specific diagnostic of conditional probability differences. It is not free-generation accuracy, an estimate of how much factual knowledge a model stores, or evidence about where that knowledge is

represented. Comparisons between released checkpoints also do not identify a causal effect of SFT, DPO, or instruction tuning. Finally, the reported values should not be read as a property of a model independent of the prompt wrapper, target token, response prefix, dataset, or checkpoint revision used in the evaluation.

B Additional Results and Experimental Details

B.1 Exact Serialization and Scoring Protocol

The primary evaluation uses the following Exact Plaintext B wrapper for every model family and dataset:

```
Complete the following statement.
Statement: {statement}
Answer:
```

The factual statement is inserted into the same wrapper in the original and substitute conditions. The response is fixed rather than freely generated; for the answer-token analysis it has the form The answer is answer. The same response prefix and target token are therefore scored in both conditions. The scorer obtains conditional log probabilities with teacher forcing, averages the aligned answer-token values within a sample, and only then computes checkpoint or relation summaries.

Token alignment is based on character offsets returned for the response tokenization. A token is retained for the answer span when its character interval overlaps the verified answer interval. This is the project's documented ANY_OVERLAP rule. The complete prompt-plus-response tokenization is also checked against the concatenation of the separately tokenized prompt and response.

B.2 Dataset Construction

CounterFact. The clean CounterFact input contains 252 unique records covering 24 relations. Each primary comparison pairs an original factual statement with a relation-matched substitute. The retained pair has the same relation but a different subject and answer, compatible wording, an unambiguous factual statement, and no direct leakage of the target answer into the prompt. The project's clean-data record and final manual-review audit establish the final 252/24 set; no additional intermediate filtering counts are reported here.

ParaRel. The external relation-based check contains 230 records over 23 relations shared with CounterFact. The retained set is balanced at ten records per relation. The construction record applies relation compatibility, distinct subjects and answers, target checks, leakage checks, and exclusion of exact CounterFact facts. This gives a second dataset on which the same scoring procedure can be evaluated without changing the primary CounterFact sample.

Paraphrase and substitute controls. The verified paraphrase input contains 83 original–paraphrase pairs. The existing verification pipeline records 83 successful pairs, with zero failed pairs, token mismatches, or answer-span failures in the recorded run. The multi-swap construction permits up to $K = 3$ legal substitutes for an original example; the source record contains 174 examples with at least two donors and 112 examples with three. The random-prompt control uses the clean 252-example population and assigns a prompt from an unrelated relation. These controls retain the original example as the statistical unit.

B.3 Evaluated Checkpoints and Exact Revisions

Table 2 records the evaluated checkpoint roles and the pair counts used by each dataset. The exact revisions below are copied from the current reproducibility record; they identify snapshots and do not imply a controlled intervention.

Qwen3. Base: `ea980cb0a6c2ae4b936e82123acc929f1cec04c1`; Post: `70d244cc86ccca08cf5af4e1e306ecf908b1ad5e`.

Table B1: Extended four-family CounterFact Exact Plaintext B results. Delta is Post/Instruct minus Base. Direct relation-direction counts are computed from each relation's own endpoint delta and are distinct from leave-one-relation-out aggregate sign-preservation runs.

Family	Base	Post/Inst.	Delta	95% CI	Direct rel. dir.
Qwen3	3.3817	1.5761	-1.8056	$[-2.1814, -1.4267]$	23/24 down
Mistral	3.2483	5.5473	$+2.2991$	$[2.0133, 2.5850]$	24/24 up
SmolLM2	3.1015	3.8545	$+0.7530$	$[0.5099, 0.9932]$	18/24 up
OLMo 2 7B	3.7312	7.5103	$+3.7791$	$[3.3425, 4.2020]$	23/24 up

Table B2: ParaRel external validation using arithmetic token means and exact Plaintext B. Delta is Post/Instruct minus Base. The direct relation-direction count is distinct from leave-one-relation-out sign preservation.

Family	Base	Post/Inst.	Delta	95% CI	Direct rel. dir.
Qwen3	3.4471	1.6951	-1.7520	$[-2.1179, -1.4042]$	21/23 down
Mistral	3.3893	5.3594	$+1.9701$	$[1.7000, 2.2495]$	23/23 up
SmolLM2	3.3339	4.0208	$+0.6868$	$[0.4596, 0.9244]$	20/23 up

Mistral. Base: `caa1feb0e54d415e2df31207e5f4e273e33509b1`; Instruct: `c170c708c41dac9275d15a8fff4eca08d52bab71`.

SmolLM2. Base: `effd688a12921b4cc83e3312b6feb579f70f9c71`; Instruct: `31b70e2e869a7173562077fd711b654946d38674`.

OLMo 2 7B. The public model identifiers are `allenai/OLMo-2-1124-7B`, `allenai/OLMo-2-1124-7B-SFT`, `allenai/OLMo-2-1124-7B-DPO`, and `allenai/OLMo-2-1124-7B-Instruct`. Their recorded revisions are, respectively, `7df9a82518afdecae4e8c026b27adccc8c1f0032`, `1de02c0175118a9de5854aec80a1f970e701e928`, `e34ea60adff2e575f4fe7569eaffd1b28509b6fd`, and `470b1fba1ae01581f270116362ee4aa1b97f4c84`.

B.4 Statistical Protocol

The response tokens are not treated as independent observations. For each example, the answer-token log probabilities are averaged first, producing a single sample-level Answer PD. Checkpoint contrasts then subtract the paired sample values for the same example. Means, medians, positive rates, and relation means are computed from these sample-level values.

Paired bootstrap intervals resample complete paired examples, keeping the earlier and later checkpoint values together. The recorded analyses use $B = 5000$ resamples. The verified paraphrase run uses seed 2027. The OLMo trajectory record reports a base seed of 20260916 and a paired seed of 20260917, with percentile-linear intervals. Other endpoint scripts retain their run-specific seeds in their manifests; a single global seed is not assumed where the source records use different seeds.

Direct relation deltas average the sample-level changes within one relation. LORO excludes one complete relation and recomputes the mean over the remaining examples. The two quantities answer different questions and are reported separately. No missing auxiliary metric is imputed in the source tables; in particular, unavailable old-target fields remain missing rather than being reconstructed.

B.5 Full Endpoint Results

The extended CounterFact endpoint table reports the four-family means, paired intervals, and direct relation-direction counts in Table B1. The corresponding ParaRel endpoint results are given in Table B2. These tables expand the compact main-text summary without changing any reported value.

B.6 OLMo 2 7B Staged Results

The four public OLMo 2 7B checkpoints are evaluated on both datasets. The staged table reports the checkpoint means, adjacent contrasts, and paired intervals; an interval in the contrast column belongs to the named adjacent change rather than to a checkpoint mean.

Table B3: OLMo 2 7B staged trajectories under Exact Plaintext B on CounterFact and ParaRel. Means are sample-level Answer PD means. The interval column reports paired adjacent-checkpoint sample-level bootstrap 95% confidence intervals ($B = 5000$) for the contrast named in the Delta vs previous column; it is not a confidence interval for the checkpoint mean.

Dataset	Checkpoint	Mean Answer PD	Δ vs previous	95% CI for paired Δ
CounterFact	Base	3.731187	—	—
CounterFact	SFT	6.319883	+2.588695	[2.270738, 2.913637]
CounterFact	DPO	7.258249	+0.938367	[0.798547, 1.075769]
CounterFact	Instruct	7.510304	+0.252055	[0.201833, 0.302916]
ParaRel	Base	4.020796	—	—
ParaRel	SFT	6.652760	+2.631964	[2.272311, 2.996185]
ParaRel	DPO	7.598107	+0.945346	[0.806508, 1.085708]
ParaRel	Instruct	7.853094	+0.254988	[0.208422, 0.301556]

B.7 Cross-Dataset Comparison

CounterFact and ParaRel share 23 relations, which permits a direction-level comparison on the common relation set. ParaRel was balanced before scoring, whereas CounterFact retains its clean 252-example population. The endpoint table in Table B2 gives the ParaRel values, and the OLMo table above gives the staged comparison on both datasets. No cross-dataset value is created by averaging unrelated relations or by reweighting the datasets.

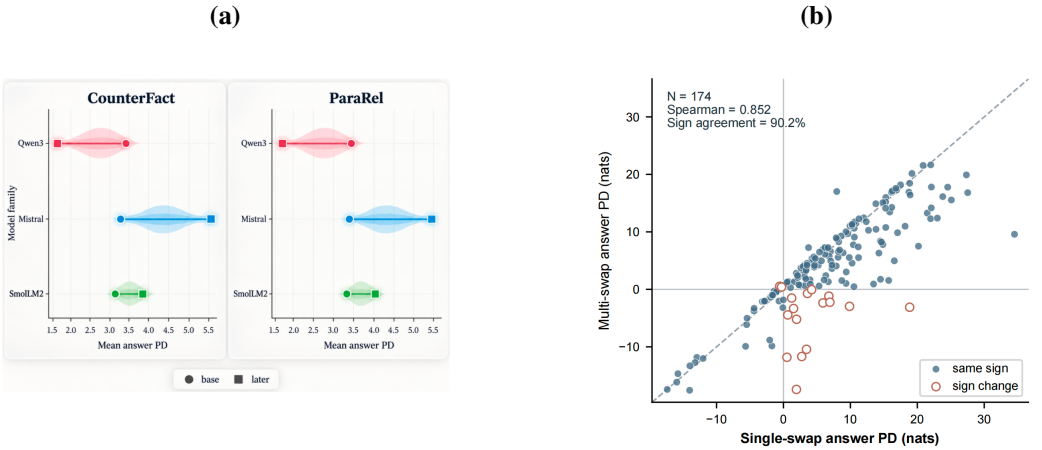

Figure B1: Cross-dataset endpoint comparison and swap agreement. (a) Base and later-checkpoint mean Answer PD on CounterFact and ParaRel. (b) Single- versus multi-swap Answer PD on the primary subset; filled points retain the sign and open points change sign.

B.8 Paraphrase Control

The detailed paraphrase table reports Base and later Answer PS, paired intervals, sample-level directions, and direct relation directions for the verified 83-pair subset. The staged OLMo series was not evaluated under this control.

Table B4: Paraphrase control by model family under full gold-prefix teacher forcing. The verified subset contains $N = 83$ paired examples. Values are sample-level Answer PS after within-sample answer-token averaging. Intervals are paired sample-level bootstrap 95% confidence intervals with $B = 5000$ resamples and seed 2027. Sample $+/-$ counts denote sample-level directions, whereas relation $+/-$ counts are obtained by direct relation aggregation and are not leave-one-relation-out runs. This paraphrase-control analysis covers Qwen3, Mistral, and SmolLM2 only; OLMo 2 7B was not evaluated on the verified $N = 83$ paraphrase subset.

Family	Base PS	Post/Inst. PS	Δ	95% CI	Sample $+/-$	Relation $+/-$
Qwen3	1.073358	1.529308	+0.455949	[+0.188684, +0.747167]	57/26	14/3
Mistral	1.035296	1.000238	−0.035058	[−0.275333, +0.211812]	36/47	9/8
SmolLM2	1.535194	1.091850	−0.443345	[−0.728588, −0.172141]	37/46	6/11

B.9 AUXILIARY DIAGNOSTICS

The auxiliary tables collect metrics that are not used to redefine the primary Answer PD. Table B5 reports non-answer PD and answer-versus-non-answer selectivity. Table B6 reports the complementary new-target gain. The old-target quantity is absent where the source reconciliation audit marks it unavailable; it is not reconstructed.

Table B5: Auxiliary answer/non-answer diagnostics across checkpoint pairs. Delta denotes Post/Instruct minus Base. All values are arithmetic means of existing sample-level metrics from the verified primary runs.

Dataset	Family	Non-answer PD			Selectivity		
		Base	Post/Inst.	Δ	Base	Post/Inst.	Δ
CounterFact	Qwen3	-0.0161	-0.0061	+0.0100	3.3978	1.5822	-1.8156
CounterFact	Mistral	-0.0226	0.0257	+0.0483	3.2709	5.5217	+2.2507
CounterFact	SmolLM2	-0.0624	-0.0095	+0.0529	3.1639	3.8640	+0.7001
ParaRel	Qwen3	-0.0338	-0.0321	+0.0016	3.4809	1.7272	-1.7537
ParaRel	Mistral	-0.0268	-0.0200	+0.0068	3.4161	5.3794	+1.9633
ParaRel	SmolLM2	0.0222	-0.0413	-0.0635	3.3117	4.0621	+0.7504

Table B6: New-target diagnostic across checkpoint pairs. G_{new} measures support for the substitute answer under the substitute-minus-original prompt comparison. Delta denotes Post/Instruct minus Base. The old-target diagnostic is not repeated because, for the original-answer response, it coincides with primary Answer PD by construction.

Dataset	Family	Base	Post/Inst.	Δ
CounterFact	Qwen3	3.3676	1.5709	-1.7967
CounterFact	Mistral	3.6712	5.6503	+1.9791
CounterFact	SmolLM2	3.1123	3.7643	+0.6520
ParaRel	Qwen3	3.5510	1.7187	-1.8324
ParaRel	Mistral	3.3974	5.5556	+2.1582
ParaRel	SmolLM2	3.1607	3.9802	+0.8195

B.10 RELATION-LEVEL AND LORO ROBUSTNESS

The OLMo robustness tables separate direct relation directions from the sign of the aggregate after one relation is removed. The CounterFact table covers 24 relations, while the ParaRel table covers 23. The values are based on the saved sample-level results and the recovered LORO implementation; no model inference or bootstrap rerun is required for this appendix.

Table B7: OLMo 2 7B relation-level robustness under CounterFact Exact Plaintext B. Direct relation direction counts are computed from relation-specific deltas, whereas LORO reports the sign of the sample-weighted aggregate delta after holding out one complete relation. The two counts are not interchangeable. The full-delta column is the paired sample-level contrast; the LORO range spans the 24 held-out aggregate deltas.

Contrast	Full Δ	95% CI	Direct rel. dir.	LORO sign	LORO range
SFT–Base	$+2.5887$	$[2.2707, 2.9136]$	23/24 up	24/24 up	$[2.4175, 2.7312]$
DPO–Base	$+3.5271$	$[3.1143, 3.9239]$	23/24 up	24/24 up	$[3.3140, 3.6382]$
Instruct–Base	$+3.7791$	$[3.3425, 4.2020]$	23/24 up	24/24 up	$[3.5561, 3.8932]$
DPO–SFT	$+0.9384$	$[0.7985, 1.0758]$	23/24 up	24/24 up	$[0.8966, 0.9671]$
Instruct–DPO	$+0.2521$	$[0.2018, 0.3029]$	23/24 up	24/24 up	$[0.2412, 0.2669]$

Table B8: OLMo 2 7B relation-level robustness under ParaRel Exact Plaintext B. Direct relation-direction counts are computed from relation-specific deltas, whereas LORO reports the sign of the sample-weighted aggregate delta after holding out one complete relation. The full-delta column is the paired sample-level contrast; the LORO range spans the 23 held-out aggregate deltas.

Contrast	Full Δ	95% CI	Direct rel. dir.	LORO sign	LORO range
SFT–Base	$+2.6320$	$[2.2723, 2.9962]$	23/23 up	23/23 up	$[2.4852, 2.7443]$
DPO–Base	$+3.5773$	$[3.1415, 4.0029]$	22/23 up	23/23 up	$[3.4005, 3.7498]$
Instruct–Base	$+3.8323$	$[3.3737, 4.2830]$	22/23 up	23/23 up	$[3.6409, 4.0128]$
DPO–SFT	$+0.9453$	$[0.8065, 1.0857]$	22/23 up	23/23 up	$[0.8976, 1.0128]$
Instruct–DPO	$+0.2550$	$[0.2084, 0.3016]$	23/23 up	23/23 up	$[0.2404, 0.2648]$

B.11 Supplementary Relation and Checkpoint Diagnostics

The main text reports the compact relation-level views needed to interpret the OLMo 2 transitions. This appendix retains the fuller held-out-relation, relation-profile, and auxiliary checkpoint displays. They use the same sample-level units as the main analysis and do not introduce additional metrics.

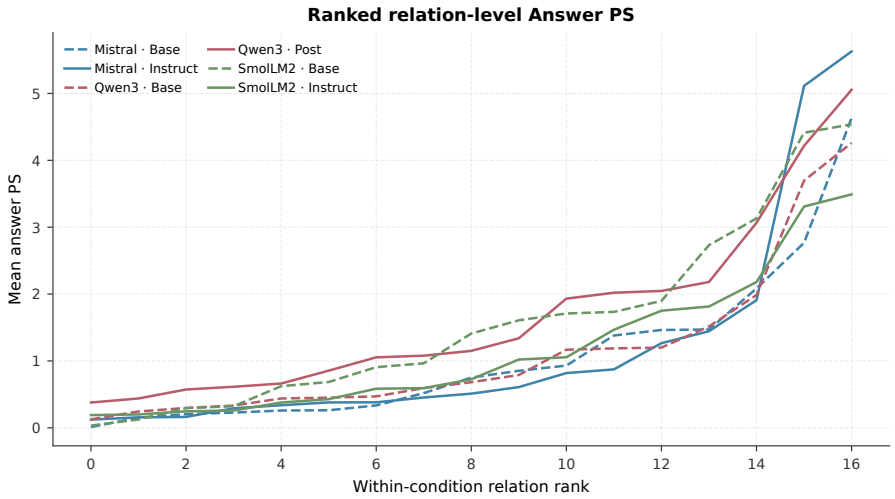

Figure B2: Ranked relation-level Answer PS profiles across the Qwen3, Mistral, and SmolLM2 endpoint checkpoints. Each curve orders relations within a condition and is included as an auxiliary view of relation-level paraphrase variation.

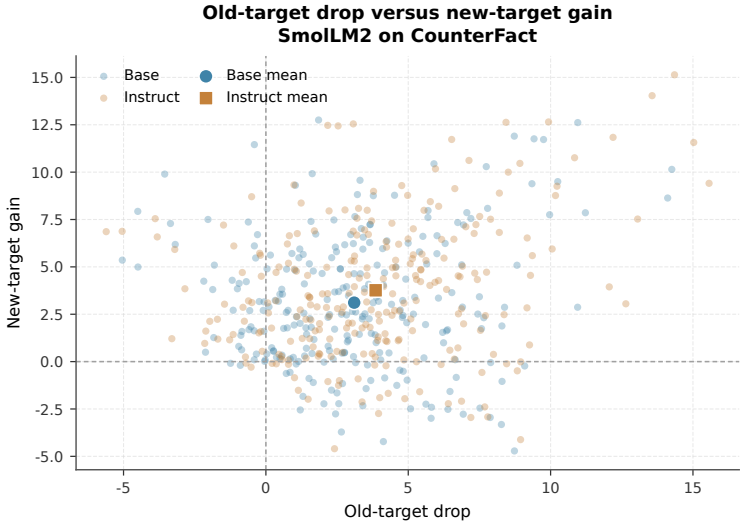

Figure B3: Old-target drop versus new-target gain for SmolLM2 on CounterFact. Points show sample-level values for Base and Instruct; larger markers indicate the corresponding condition means.

The full multi-model LORO display in Figure B5 complements the aggregate counts in the main text by showing the held-out-relation results for the three original model families individually.

B.12 Additional Paraphrase-Control Analyses

The relation heatmap and condition-level distributions complement the main paraphrase-control results. The remaining views examine sample-level alignment, family/condition summaries, and relation profiles. Together they show how the aggregate pattern is distributed across examples and relations.

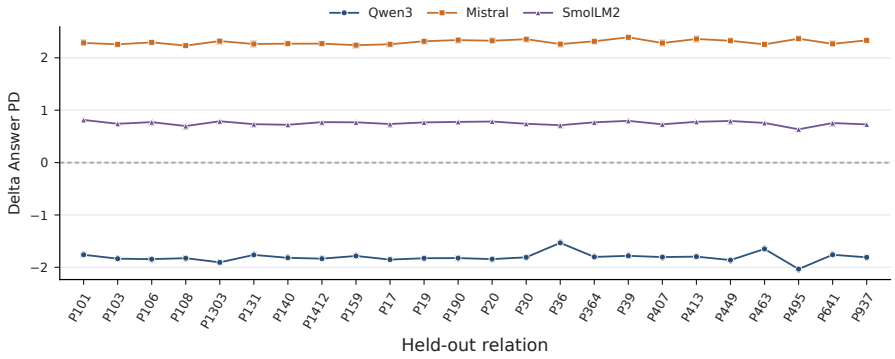

Figure B4: Leave-one-relation-out (LORO) aggregate Answer PD deltas across the 24 CounterFact relations. Each point is the family-level Post/Instruct-minus-Base delta after holding out one relation. Qwen3 remains negative in all 24 held-out runs, whereas Mistral and SmolLM2 remain positive in all 24 held-out runs.

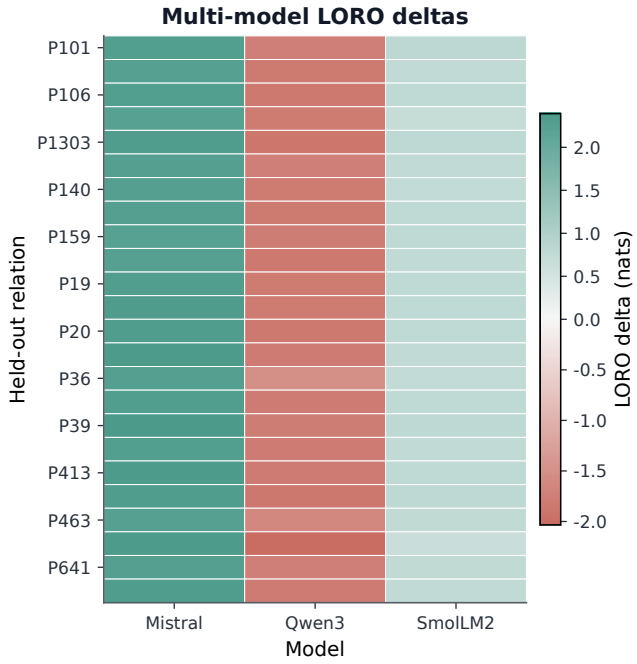

Figure B5: Multi-model leave-one-relation-out (LORO) deltas across held-out CounterFact relations for Qwen3, Mistral, and SmolLM2.

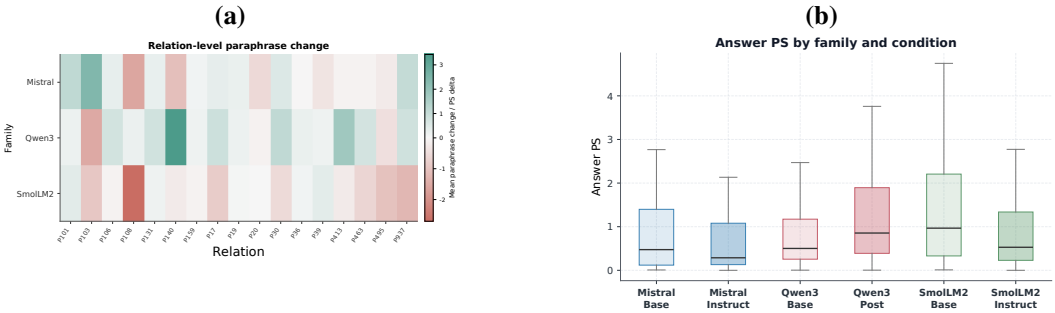

Figure B6: Paraphrase-control results by relation and condition. (a) Relation-level later-minus-base Answer PS changes. (b) Answer PS across the six family/checkpoint conditions; diamonds mark means.

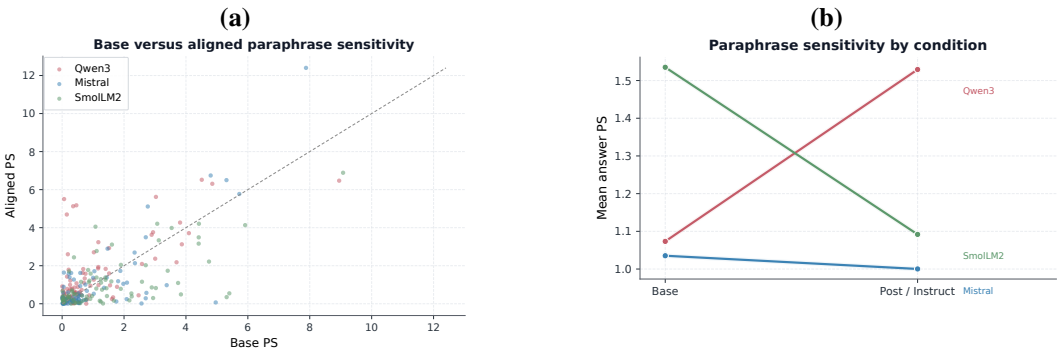

Figure B7: Additional paraphrase-control diagnostics for the verified Qwen3, Mistral, and SmolLM2 subset. (a) Base versus aligned paraphrase sensitivity at the sample level. (b) Mean Answer PS at the Base and Post/Instruct conditions for each family.

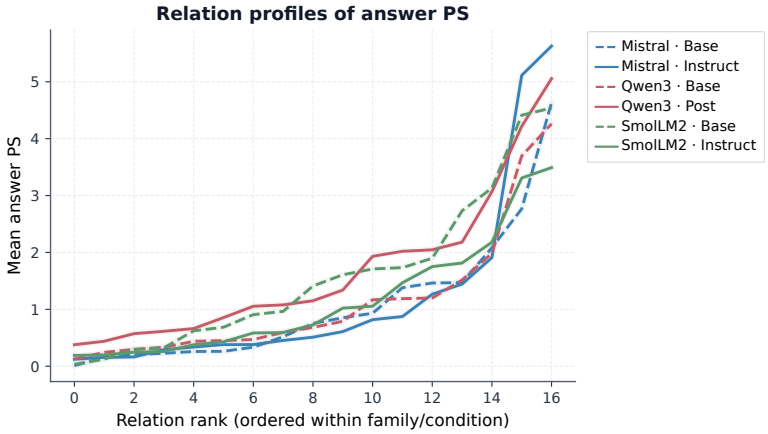

Figure B8: Relation profiles of Answer PS across endpoint families and conditions in the verified paraphrase subset.

B.13 Target-Direction Diagnostics

The target-direction diagnostics compare changes associated with the original and substituted targets. They are supplementary to the primary Answer PD metric and document how the auxiliary quantities behave on the verified SmolLM2 CounterFact subset.

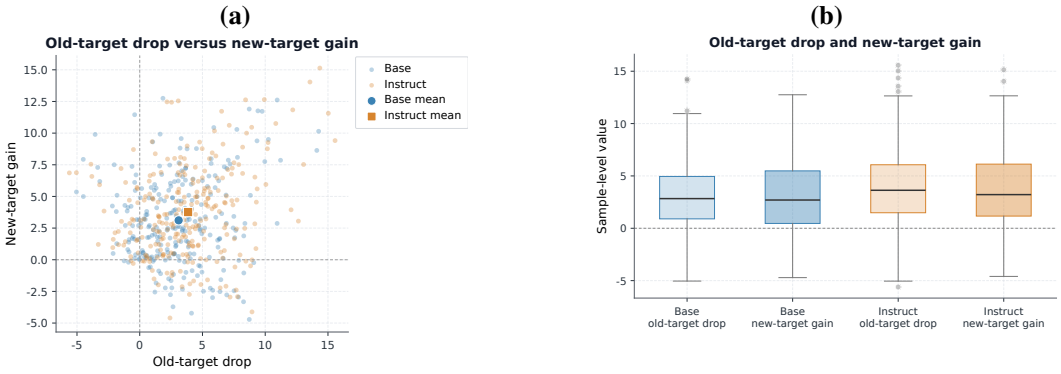

Figure B9: Supplementary target-direction diagnostics for the SmolLM2 CounterFact subset. (a) Old-target drop versus new-target gain. (b) Sample-level distributions of the two target-direction quantities across Base and Instruct.

C Reproducibility, Implementation, and Audit Details

C.1 Project Organization

The released materials are organized around source scripts, saved inputs, checkpoint metadata, analysis outputs, figures, and audit records. The following compact tree reflects the directories used by the reproducibility package and the staged OLMo recovery materials:

```
code/
  src/scoring/teacher_forcing.py
  scripts/run_qwen_counterfact_plaintextB_exact_v1.py
  scripts/run_pararel_full.py
  scripts/run_stage5_mistral_full.py
  scripts/build_clean252_multiswap_k3.py
  scripts/build_pararel_candidates.py
  scripts/build_random_v1.py
  scripts/run_paraphrase_full.py
  scripts/analyze_multiswap_robustness_v1.py
data/  results/  figures/  audits/  environment_info.md
stage18_0D_R_loro_recovery_v1/code/compute_loro.py
```

The tree is an inventory of recorded project files rather than a list of canonical end-to-end entry points. In particular, `run_paraphrase_full.py` is retained as a historical runner and is not the implementation used to define the final paraphrase-control Answer PS reported in this paper. The archived `run_pararel_full.py` likewise reflects the recorded project environment and is not presented here as a clean-environment entry point. The final analyses are tied to the scoring and post-processing definitions documented below and to the saved outputs recorded in the provenance inventory.

The inventory in the accompanying Stage 19 audit covers the scanned source records. Paths in this appendix are package-relative so that the description does not expose a local account or machine path.

C.2 Core Fixed-Context Scoring Implementation

The implementation excerpt below is abridged from the authoritative teacher-forcing scorer. It shows how prompt tokens and fixed response tokens are concatenated, how response logits are aligned, and how answer-token membership is recorded.

Listing C1: Fixed-context scoring excerpt, abridged from the recorded teacher-forcing scorer.

```
def score_response(model, tokenizer, prompt_text, response_text,
                answer_span, device="cuda"):
   p = tokenizer(prompt_text, add_special_tokens=False)["input_ids"]
   r = tokenizer(response_text, add_special_tokens=False,
               return_offsets_mapping=True)
   ids, offsets = r["input_ids"], r["offset_mapping"]
   x = torch.tensor([p + ids], device=device)
   with torch.inference_mode():
      logits = model(input_ids=x).logits[0]
      z = logits[len(p)-1:len(p)-1+len(ids)]
      lp = F.log_softmax(z.float(), -1).gather(
         1, x[0, len(p):, None]).squeeze(1).tolist()
   a, b = answer_span
   rows = []
   for i, (token_id, (u, v), value) in enumerate(zip(ids, offsets, lp))
       :
      rows.append({"token_index": i, "token_id": token_id,
                "char_start": u, "char_end": v,
                "is_answer": u < b and v > a, "logp": value})
   return rows
```

The overlap test in the excerpt is the same target-alignment convention used by the saved evaluation outputs. The complete source additionally records token IDs, offsets, concatenation checks, and diagnostic metadata.

C.3 Sample-Level Aggregation and Bootstrap

The endpoint runners first average answer-token log probabilities within a sample, form Answer PD and auxiliary fields, and then resample sample-level values. The following excerpt is abridged from a recorded endpoint runner; it shows the observed $B = 5000$ bootstrap path without reproducing model-loading boilerplate.

Listing C2: Sample-level aggregation and bootstrap excerpt from a recorded endpoint runner.

```
def token_stats(token_rows):
   answer = [float(r["logp"]) for r in token_rows
          if bool(r["is_answer"])]
   nonanswer = [float(r["logp"]) for r in token_rows
             if not bool(r["is_answer"])]
   return {"answer_mean": float(np.mean(answer)),
         "nonanswer_mean": float(np.mean(nonanswer)) if nonanswer else
             0.0}

def bootstrap_ci(values, seed=BOOTSTRAP_SEED, B=BOOTSTRAP_B):
   arr = np.asarray([float(v) for v in values], dtype=float)
   rng = np.random.default_rng(seed)
   indices = rng.integers(0, arr.size, size=(B, arr.size))
   means = arr[indices].mean(axis=1)
   return [float(np.percentile(means, 2.5)),
         float(np.percentile(means, 97.5))]
```

For a paired checkpoint contrast, the input to this function is the vector of later-minus-earlier values for the same sample IDs. The OLMo trajectory audit separately records a paired seed of 20260917 and five adjacent or base-relative contrasts.

C.4 Relation Aggregation and LORO

The recovery script consumes saved sample-level results and does not download models or run inference. Before aggregation it checks duplicate IDs, sample parity, relation parity, gold-answer parity when available, and finite values. The central loop is reproduced below from the recorded implementation.

Listing C3: Leave-one-relation-out excerpt from the recorded recovery implementation.

```
paired.append({**base_row,
           "post_answer_pd": post_row["answer_pd"],
           "delta": post_row["answer_pd"] - base_row["answer_pd"]})

full_delta = mean(row["delta"] for row in paired)
relations = []
for row in paired:
   if row["relation"] not in relations:
      relations.append(row["relation"])

for relation in relations:
   held_out = [row for row in paired
            if row["relation"] == relation]
   remaining = [row for row in paired
             if row["relation"] != relation]
   direct_delta = mean(row["delta"] for row in held_out)
   loro_delta = mean(row["delta"] for row in remaining)
   rows.append({"excluded_relation": relation,
             "direct_relation_delta": direct_delta,
             "loro_delta": loro_delta,
             "remaining_n": len(remaining)})
```

The Stage 18.0D-R audit records 120/120 OLMo LORO runs complete across five contrasts. The same audit records the regression targets for the three endpoint-family comparisons. A package-level inventory entry marked the generic LORO entrypoint as unresolved; the dedicated Stage 18.0D-R recovery source resolves the implementation used for the reported LORO checks.

C.5 Control Pipelines

For the final paraphrase-control analysis, the original and paraphrased statements are scored with the same fixed-context teacher-forcing procedure. For each condition, the aligned answer-token log probabilities are averaged first. Sample-level Answer PS is then computed as the absolute difference between these two means. The recorded post-processing keeps the 83 paired examples together for each of 5,000 bootstrap resamples.

This is the definition used for the paraphrase-control results reported in the paper. The package also retains an earlier `run_paraphrase_full.py` pipeline that used a different token-level aggregation. That script is a historical artifact and is not a reproduction entry point for the final Answer PS results.

The multi-swap pipeline scores each legal donor separately and averages donor probabilities in probability space before returning to log space. The random-prompt build creates a one-to-one pairing with a prompt from a different relation. The old-target and new-target values are written as auxiliary fields in the endpoint summaries; they do not replace primary Answer PD.

The final paraphrase summary files retain the sample-level quantities used in the reported analyses rather than every individual answer-token log probability. The final PS definition is therefore documented by the scoring and post-processing source together with the recorded intermediate analysis artifacts; it cannot be reconstructed from the final summary files alone.

C.6 Validation and Integrity Audits

Table C1 collects checks that are explicitly recorded in the source audits. A missing or unresolved check is reported as such rather than converted to PASS.

Table C1: Recorded validation and integrity checks used in the Stage 19 source audit.

Check	Expected	Observed	Status
Clean CounterFact population	252 records, 24 relations	252 unique records, 24 relations	PASS
ParaRel balance	230 records, 23 relations	10 records per shared relation	PASS
Verified paraphrase input	83 paired records	83 successful; zero recorded failures, token mismatches, or span failures	PASS
OLMo staged parity	4 stages with common sample set	252 samples and 24 relations at each stage	PASS
OLMo LORO recovery	120 runs	120/120 complete; finite outputs	PASS
OLMo bootstrap	5,000 finite replicates per contrast	$B = 5000$; five recorded contrasts; NaN=0, Inf=0	PASS
Recorded paraphrase evaluation	498 checkpoint-sample evaluations	498/498 successful; failed=0, NaN=0, Inf=0	PASS
Generic package LORO entry-point	Resolved in package manifest	Marked unresolved; dedicated recovery source is documented separately	DOC.

C.7 Reproducibility Manifest

The following values are copied from the recorded environment information. They describe the historical evaluation environment and are not inferred from the current host running this document.

Table C2: Recorded reproducibility environment and evaluation settings.

Item	Recorded value	Scope or note
Python	3.12.3	Historical package inspection record
PyTorch	2.8.0+cu128	Historical package inspection record
Transformers	5.16.1	Historical package inspection record
Accelerate	1.14.0	Historical package inspection record
NumPy / pandas	2.3.2 / 3.0.5	Historical package inspection record
CUDA / GPU	13.0 / GeForce RTX 4080 SUPER	Recorded driver/runtime information
Model dtype	torch.bfloat16	Recorded endpoint and ParaRel runners
Primary serialization	Exact Plaintext B	Fixed wrapper with teacher forcing
Paraphrase bootstrap	$B = 5000$, seed 2027	Verified 83-pair control
OLMo paired bootstrap	$B = 5000$, seed 20260917	Staged trajectory audit

C.8 Reproduction and Provenance Notes

The package contains both current analysis sources and historical runners retained during development. For this reason, the filenames in the project inventory should not be interpreted as a single clean-environment execution sequence.

The primary fixed-context calculations are defined by the teacher-forcing scorer and the sample-level aggregation described above. The final paraphrase-control results use the sample-level Answer PS definition in Section C.5 rather than the historical implementation retained in `run_paraphrase_full.py`. The archived `run_pararel_full.py` records the ParaRel evaluation used in the project environment but is not presented as a portable standalone entry point.

The LORO results can be recovered from the saved sample-level outputs using the dedicated Stage 18.0D-R implementation:

```
python stage18_0D_R_loro_recovery_v1/code/compute_loro.py \
  --family <family> \
  --stage-from <base> \
  --stage-to <later> \
  --base <saved_base_results> \
```

```
  --post <saved_later_results>
```

Final figures are tied to the saved analysis outputs and plotting provenance recorded in the package inventory. The placeholder `run_figures.sh` is not treated as a canonical figure-reproduction entry point.

The code inventory and provenance CSV record the source and saved artifact associated with each reported table, figure, and analysis. No absolute local account path or credential is required to inspect the released artifacts.