STYLE-DEBIASED DPO: UPDATING LLM KNOWLEDGE WITH FACTUALITY-AWARE SYNTHETIC PREFERENCE DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Continued pretraining (CPT) with data augmentation such as paraphrasing can store inside a large language model (LLM) the knowledge of a small source corpus, a collection of documents describing new knowledge. The stored knowledge, however, is not always retrieved correctly. We study the eliciting side rather than the storing side: we use preference optimization, which learns from pairs of a preferred (chosen) and a dispreferred (rejected) response, so that the model *elicits* its stored knowledge more accurately. One proposed approach takes the model's own erroneous response as rejected and the gold answer as chosen, so as to suppress the error. When the target knowledge is *partially known*, however, most of these rejected responses are factually correct. Using direct preference optimization (DPO) as the preference optimization method then pushes down rejected responses that contain correct knowledge and differ from the chosen answer only in style, such as length and wording. We propose *style-debiased DPO* (SD-DPO), which scores whether the rejected response of each pair is factually correct, inverts the preference of such pairs, and weights them so that the learning signal due to differences in style cancels out as a whole. We first test whether, on top of EntiGraph, a representative storing-side method that runs CPT on text synthesized from the corpus, our method adds accuracy efficiently. On QuALITY, the reading-comprehension QA benchmark on which EntiGraph was evaluated, SD-DPO exceeds a baseline we CPT on EntiGraph's synthetic data from the same base model and evaluate with the same procedure. The training tokens this requires are a few dozen times fewer than the additional CPT needed for the same gain. For knowledge updating, the main goal of this work, we use AToKE, a knowledge-editing benchmark for facts that change over time. There, SD-DPO reaches an overall accuracy of 0.982 and answers with the new or the old fact according to the queried period.

1 INTRODUCTION

Teaching a pretrained large language model (LLM) new knowledge remains difficult in several respects. New facts are learned more slowly than facts the model already knows, and must be presented repeatedly in diverse paraphrases (Allen-Zhu & Li, 2024; Ovadia et al., 2024; Chang et al., 2024). Training on a document does not guarantee that its content can be elicited as answers to questions (Berglund et al., 2024; Jiang et al., 2024). Learning new knowledge increases hallucination (Gekhman et al., 2024). The new knowledge also interferes with existing abilities and with old knowledge (Luo et al., 2023; Xu et al., 2024). Methods that let an LLM answer correctly about new knowledge fall into two groups. One keeps the documents describing the new knowledge outside the model and supplies them as context at inference time through retrieval-augmented generation (RAG; Lewis et al., 2020). The other performs *continued pretraining* (CPT) on the collection of such documents (henceforth the *source corpus*), updating the model parameters directly so that the LLM itself can answer. This paper concerns the latter. The naive form, CPT on the source corpus as is, has been reported to lower accuracy when the corpus is small, because the phrasings are not diverse and the distribution is narrow (Yang et al., 2025); data augmentation such as paraphrasing is therefore used. Its representative, EntiGraph (Yang et al., 2025), generates synthetic texts that explain the relations among the entities of the source corpus in many different phrasings, and showed

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.

that training on them makes closed-book QA accuracy (answering from memory, without the source text) scale log-linearly with the number of training tokens. Similarly, PASTA (Yamamoto & Kawahara, 2026) combines paraphrase-based augmentation of the source corpus, QA generation over the source corpus, and self-preference optimization into a method that integrates the knowledge of news articles past the LLM's knowledge cutoff into the LLM. However, the EntiGraph scaling curve also exposes a limit: as training proceeds, each additional token teaches less, and accuracy levels off. Storing more knowledge does not by itself make a task that asks about that knowledge *elicit* correct answers in proportion. That is, data augmentation has made storing knowledge feasible, but the difficulty of eliciting the stored knowledge correctly, noted above, remains.

We therefore treat acquiring knowledge and eliciting it correctly as two separate problems, and adopt a two-stage design that places a preference-optimization stage after CPT. We assume that the main role of the first stage, CPT, is to *store* knowledge, and that the main role of the second stage, preference optimization, is to make it *retrievable*. Our experiments confirm this assumption (Section 4.2). For preference optimization we use direct preference optimization (DPO; Rafailov et al., 2023), which pairs each question with a chosen and a rejected response, raises the likelihood of the former, and lowers that of the latter. The responses we want to suppress are the erroneous ones the model currently produces, and those are obtained simply by sampling the model itself; we therefore take the gold answer as chosen and the CPT model's own sampled response as rejected—the recipe of PASTA's self-preference optimization (Self-DPO). Rozner et al. (2024) likewise apply DPO to knowledge editing, but take the old fact, not a sampled response, as rejected.

Preference optimization, however, learns any feature that differs systematically between chosen and rejected responses, even one unrelated to content. Response length is the best-known example (Park et al., 2024; Lu et al., 2024), and more generally any feature whose mean differs between the two is learned from the first gradient step (Moya et al., 2026). Existing remedies either correct a specific axis such as length (Park et al., 2024; Lu et al., 2024) or add hand-constructed pairs whose content is equal and only surface features differ (Moya et al., 2026). In the self-generated pairs described above, this problem takes its strongest form. The chosen (gold) and rejected (the model's own) responses differ not only in length but in *style* in general—format, verbosity, phrasing. Moreover, when the target knowledge is partially known to the model, most rejected responses are factually *correct*, so that pairs differing only in style and pairs with a genuine factual difference are mixed together without distinction. Standard DPO makes no such distinction and pushes down the whole rejected response on every pair, and with the style it pushes down the correct knowledge CPT has just acquired. Our work differs from methods that correct a specific axis such as length (Park et al., 2024; Lu et al., 2024) in removing the influence of style as a whole. The method that adds pairs whose content is equal and only surface features differ (Moya et al., 2026) shares with ours the treatment of style as a whole, but it constructs such pairs by hand, adds them to the training data, and trains on them with random labels; we instead identify such pairs, which are already present in self-generated data, with a judge LLM, and label them not at random but in the direction and with the weight that exactly cancel the style signal of the pairs with a factual difference.

Our goal is to remove from these preference pairs the learning signal that stems from differences in style, and to update the model with the signal that stems from differences in fact alone. To this end we propose *style-debiased DPO* (SD-DPO). For each pair, a judge LLM scores whether the rejected response agrees with the chosen one on the facts, and the pairs are split into a group with a genuine factual difference and a group that is factually correct and differs only in style. The former is trained as usual; the latter is trained with its preference *inverted*. The inverted group receives a weight chosen so that the gradients due to differences in style (henceforth the *style gradients*) of the two groups balance. Under two simplifying assumptions (Section 3.3), the style component vanishes from the expected gradient at initialization and only the factual learning signal remains (Proposition 1). The assumption stated above is supported as follows. The CPT model's responses agree with the gold answer on the facts in about two thirds of the preference data (Section 4.1), so storage has taken place. On top of that, with the same gold answers as chosen responses, standard DPO, which pushes every rejected response down, does not improve accuracy, whereas SD-DPO, which does not push down the factually correct ones, improves it (Section 4.2) while preserving general capabilities (Section 4.4). This contrast indicates that the gain comes from how knowledge is elicited, not from new knowledge.

Our contributions are threefold. **(i)** We identify and quantify how self-generated preference data fails for knowledge updating—factually correct rejected responses dominate whenever the target

knowledge is partially known—and propose SD-DPO, for which we prove that the preference inversion cancels the style gradient at initialization. **(ii)** On the reading-comprehension QA benchmark QuALITY, we exceed a CPT baseline trained in the EntiGraph setting and evaluated under the same conditions as our method; the preference data this requires is a few percent of the additional CPT tokens that would be needed to reach the same accuracy by CPT alone. **(iii)** For knowledge updating, the main goal of this work, on the knowledge-editing benchmark AToKE we attain high new-fact acquisition and old-fact recall simultaneously, which methods that rewrite model parameters directly do not (reference comparison; different protocol).

2 Related Work

Storing knowledge with CPT on synthetic data. Allen-Zhu & Li (2024) showed, with models trained from scratch, that the diversity of paraphrases in the training data governs whether facts can later be extracted. Yang et al. (2025), with EntiGraph, extended this insight to CPT of pretrained LLMs and showed on QuALITY (Pang et al., 2022), a reading-comprehension QA dataset over long stories, that accuracy grows with the number of synthetic tokens following a mixture-of-exponentials scaling law. Cheng et al. (2024) convert raw text into reading-comprehension format for training. All of these optimize storage; none improves retrieval with a preference stage after CPT.

Updating knowledge with preference optimization. PASTA (Yamamoto & Kawahara, 2026) applies CPT, supervised fine-tuning (SFT), and Self-DPO in sequence to news articles past the knowledge cutoff, and Rozner et al. (2024) cast the editing of a single fact as DPO with the old fact as rejected. In both, most rejected responses are factually wrong, so standard DPO receives an unambiguous learning signal. Our setting, where the target knowledge is partially known and self-generated rejected responses are mostly *correct*, has, to our knowledge, not been addressed explicitly.

Noise-robust preference optimization. rDPO (Chowdhury et al., 2024), β-DPO (Wu et al., 2024), and filtered DPO (Morimura et al., 2024) treat imperfect pairs as label noise or low quality; none asks *what* differs within a pair, although reward models are easily misled by format over content (Liu et al., 2024), preference tuning amplifies style over content features (Ferrao et al., 2025), and DPO gains little signal when chosen and rejected distributions are similar (Pan et al., 2025), as here.

Length bias and spurious features. The remedies for the problem described in Section 1, that features unrelated to content (spurious features) such as length are learned, are an explicit length margin (Park et al., 2024), SamPO, which computes the reward over an equal number of tokens drawn from both responses (Lu et al., 2024), and *tie training*, which adds pairs of equal utility that differ only in spurious features, labeled at random (Moya et al., 2026). Our style-only pairs are such ties, found automatically by the judge LLM's measure of factual agreement (the factual-agreement score defined in Section 3.2), and SD-DPO labels them so that the mean of the spurious feature cancels rather than at random (Appendix A.5); because our pairs differ greatly in length (Appendix F.1), we also compare against SamPO under identical conditions.

Knowledge editing. ROME (Meng et al., 2022) and MEMIT (Meng et al., 2023) write individual facts into parameters with high efficacy, but on knowledge that changes over time, editing overwrites the old fact. Yin et al. (2024) propose AToKE, a knowledge-editing benchmark for updating facts that change over time, and show that models edited with standard ROME or MEMIT answer questions about the old fact for its period correctly only about 2% of the time. The same paper also proposes METO, which modifies an existing editing method so that it edits the new and the old fact together with the period in which each holds; applied to ROME and MEMIT, METO raises this accuracy to 20–30%, still far below the new-fact accuracy the same paper reports for these editors (86–100%). We approach AToKE through CPT plus preference optimization and retain both facts.

3 Method

We first give an overview of the method described in this section. The input is a source corpus. The goal is to train the target LLM on this knowledge, so that afterwards it answers a variety of questions about that knowledge correctly without seeing the source text. The method has two stages. In the first stage we generate synthetic data from the source corpus and run CPT on it; the synthetic data are texts that explain the relations among the entities in a document in many different phrasings, as

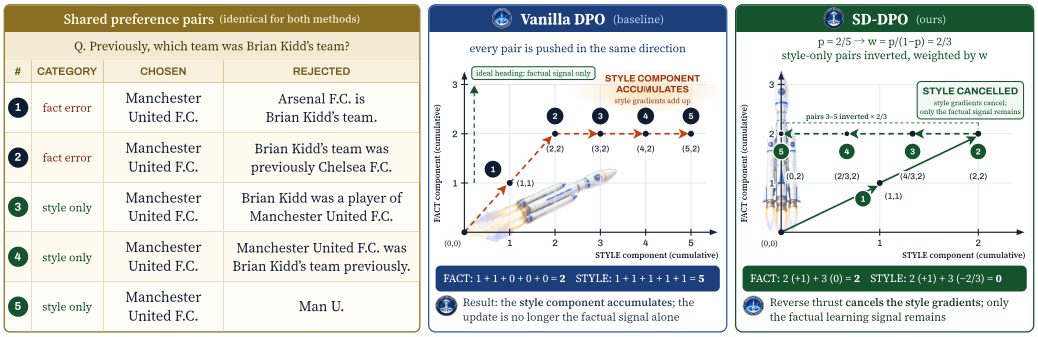


Figure 1: Conceptual illustration of SD-DPO: the learning signal in the style direction is cancelled so that only the factual (upward) component is learned, drawn as a rocket launching straight up. Both methods share five pairs (two with a factual difference, three style-only); each pair is one burst of thrust, upward for the factual component and sideways for the style component, and the axes are cumulative. Vanilla DPO learns all five in the same direction and ends at (style 5, fact 2). SD-DPO inverts the style-only pairs (reverse thrust) with weight $w = p/(1-p) = 2/3$, so the style components cancel, $2\,(+1) + 3\,(-2/3) = 0$, and the end point is (style 0, fact 2).

in EntiGraph, or paraphrases of the sentence stating each fact together with QA about its content. After this stage the knowledge is stored in the model, but it is not necessarily elicited correctly when the model is asked. The second stage is the proposed SD-DPO. We build preference pairs whose chosen response is the gold answer of a QA built from the source text and whose rejected response is what the CPT model itself returns to the same question. DPO would then suppress the rejected response and reinforce the chosen one. However, as described in Section 1, most rejected responses are factually correct and differ from the chosen ones only in style. In SD-DPO, a judge LLM scores the factual agreement of each pair; only pairs whose rejected response is factually wrong are trained in the usual DPO direction, while pairs that differ only in style are trained with the direction inverted, with a weight chosen so that the style-related gradients of the two groups exactly balance. As a result, the style signal cancels over the whole dataset and only the factual content is learned.

3.1 Problem setup and notation

Let $\mathcal{C}$ be the source corpus and $\mathcal{Q} = \{(q_i, a_i^{\text{gold}})\}$ the question–answer pairs built from it. We call a model that defines a probability distribution over responses to a question a *policy*. The model obtained by CPT on a synthetic corpus derived from $\mathcal{C}$ is the reference policy π_{ref}, which is also the initialization of the policy π_θ being trained. The preference data is $\mathcal{D} = \{(q_i, y_i^w, y_i^l)\}_{i=1}^N$, where the chosen response y_i^w is the gold answer a_i^{gold} and the rejected response $y_i^l \sim \pi_{\text{ref}}(\cdot \mid q_i)$ is sampled from the reference policy. The *margin* measures how far training has moved the log-likelihood gap between chosen and rejected away from its value under the reference policy,

$$\Delta_i \;=\; \log \frac{\pi_\theta(y_i^w|q_i)}{\pi_\theta(y_i^l|q_i)} \;-\; \log \frac{\pi_{\text{ref}}(y_i^w|q_i)}{\pi_{\text{ref}}(y_i^l|q_i)}, \tag{1}$$

and the standard DPO loss is $\mathcal{L}_{\text{DPO}} = \mathbb{E}_i[-\log \sigma(\beta\Delta_i)]$, where σ is the sigmoid function and β controls how far the policy may move from the reference. At initialization $\pi_\theta = \pi_{\text{ref}}$, so $\Delta_i = 0$, $\sigma(0) = 1/2$, and the gradient reduces to

$$\nabla_\theta \mathcal{L}_{\text{DPO}}\big|_{\theta_{\text{ref}}} = -\frac{\beta}{2}\,\mathbb{E}_i\big[\nabla_\theta \log \pi_\theta(y_i^w \mid q_i) - \nabla_\theta \log \pi_\theta(y_i^l \mid q_i)\big]. \tag{2}$$

The second term *directly lowers* the likelihood of the rejected response—the operation we want when y^l is wrong, and the operation we must avoid when y^l is right. DPO also solves the same optimization problem as reinforcement learning from human feedback (RLHF), whose solution $\pi^\star$ cannot move far from the reference policy: in terms of the KL divergence, a measure of the gap between two distributions, $\text{KL}(\pi^\star \| \pi_{\text{ref}}) \le \Delta r/\beta$, where Δr is the reward gained by the optimization. Under a mild assumption, how much the knowledge stored by CPT can change is bounded from above by this KL (Appendix A). This is why a preference stage can change *how knowledge is accessed* while approximately preserving *what is stored*—and why what endangers stored knowledge is training far past the optimum, not the shape of the loss (Section 4.1).

3.2 MEASURING FACTUAL AGREEMENT

A judge LLM J scores, for each pair, how far the rejected response agrees with the chosen one on the facts, on a scale from 0 to 1. We call this score the *factual agreement* and compute it in two ways. The *pairwise* score $f_i^{(\mathrm{P})} = J(q_i, y_i^w, y_i^l) \in [0,1]$ compares the two responses using only the question and the responses themselves. The *context-grounded* score also gives the judge the document d_i from which the question was built, and asks it to score the rejected response for factual accuracy against the document rather than for agreement with the chosen response, which also catches errors in details the chosen response omits. This score is multiplied by a grounding factor measuring how much a response draws on the document,

$$f_i^{(\mathrm{C})} \;=\; J(d_i, q_i, y_i^w, y_i^l) \cdot \min\bigl(1,\; v_i^l / v_i^w\bigr), \tag{3}$$

where v^w and v^l are the fractions of words in the chosen and rejected responses that fall inside spans of three or more consecutive words matching d_i; the factor discounts generic answers that never touch the document's content. We call the pairwise score pFA and the context-grounded score cFA. The choice depends on whether the rejected response's correctness can be decided from the chosen response alone. For questions about a single fact, such as a person or a date, we use pFA. For questions about a character's motivation or the causes of events, where the rejected response gives reasons different from the chosen one and correctness is decidable only against the source, we use cFA. Let p be the share of pairs with factual agreement below 0.5, i.e., a wrong rejected response.

3.3 STYLE-DEBIASED DPO

We assume that the margin Δ_i is approximately the sum of a component due to the factual difference between chosen and rejected, $\Delta_i^{(\mathrm{F})}$, and a component due to their difference in style, $\Delta^{(\mathrm{S})}$: $\Delta_i \approx \Delta_i^{(\mathrm{F})} + \Delta^{(\mathrm{S})}$; we call this approximation the *additive decomposition* of the margin. The style component carries no pair index i because we assume it is shared across pairs: all chosen responses follow the same template and all rejected responses are generated by the same model (Assumption A2; the case where it fails is analyzed in Appendix A). Let $c_i = 1$ for pairs whose factual agreement is below the threshold $f_{\text{thr}} = 0.5$, i.e., whose rejected response is factually wrong, and $c_i = 0$ for the remaining style-only pairs ($c_i = \mathbb{1}[f_i < f_{\text{thr}}]$), and let $p = \Pr[c_i = 1]$ be the share of $c_i = 1$ pairs. SD-DPO keeps the DPO training direction on pairs with a factual difference, inverts it on style-only pairs, and multiplies the inverted group by the weight $w = p/(1-p)$:

$$\mathcal{L}_{\text{SD}} \;=\; c_i\bigl[-\log\sigma(\beta\Delta_i)\bigr] \;+\; (1-c_i)\, w\,\bigl[-\log\sigma(-\beta\Delta_i)\bigr]. \tag{4}$$

Dropping or down-weighting style-only pairs cannot remove the style gradient of the rest; cancellation requires an opposing gradient of matched magnitude, which the inversion and w provide.

Proposition 1 (Style-gradient cancellation at initialization)**.** *Under the additive decomposition and Assumption A2, with N pairs, the expected gradient of equation 4 at initialization $\theta = \theta_{ref}$ is proportional to $-\bigl[\sum_{i:c_i=1} \nabla_\theta \Delta_i^{(\mathrm{F})} - w \sum_{i:c_i=0} \nabla_\theta \Delta_i^{(\mathrm{F})}\bigr]$: the style terms sum to $Np - w\,N(1-p) = 0$ and vanish for any $p \in (0,1)$.*

The weight w is determined uniquely by this cancellation condition and is not tuned. The guarantee concerns the gradient averaged over the whole dataset at initialization; we keep the model near that regime by averaging the gradients of many pairs per step and by the budget control of Section 4.1.

Equation 4 is the ideal form that the analysis addresses. The loss we actually train makes three changes to it. First, each pair is multiplied by a weight $\hat{\phi}_i$ that reflects how confident the judge score is, normalized to mean one within each group. Second, the two group weights $(1,\, w)$ are rescaled to $(1-p,\, p)$, preserving their ratio. Third, the term of the inverted group is augmented with a knowledge-preservation term, a hinge penalty that keeps the likelihood of the correct rejected response from falling below its value under the reference policy:

$$\begin{aligned}\mathcal{L}_{\text{SD}}^{\text{impl}} = {} & c_i(1-p)\,\hat{\phi}_i\bigl[-\log\sigma(\beta\Delta_i)\bigr] \\ & + (1-c_i)\,p\,\hat{\phi}_i\Bigl[-\log\sigma(-\beta\Delta_i) + \lambda_{\text{KP}}\max\bigl(0, \log\pi_{\text{ref}}(y_i^l \mid q_i) - \log\pi_\theta(y_i^l \mid q_i)\bigr)\Bigr],\end{aligned} \tag{5}$$

The knowledge-preservation coefficient is $\lambda_{\text{KP}} = 1$. Proposition 1 still holds for equation 5: since $\hat{\phi}_i$ has mean one within each group, the cancellation holds in expectation, and the knowledge-preservation term is inactive at initialization, where the argument of the $\max$ is exactly zero. Replay data (general-purpose preference pairs added to prevent forgetting) is trained with standard DPO, and the end-of-sequence (EOS) token is excluded from the likelihood sums that define Δ_i: without this exclusion, lowering chosen likelihoods on the inverted group, which is the majority, also suppresses how every chosen response ends, and the model lapses into endless repetition (Appendix B).

4 EXPERIMENTS

This section tests whether the proposed method improves the elicitation of knowledge stored by CPT, on two benchmarks of different character. We choose the reading-comprehension QA dataset QuALITY because EntiGraph, a representative method on the storing side, was evaluated on it, so we can test on the same synthetic data and with the same evaluation procedure whether learning on the eliciting side adds accuracy on top of its CPT model, and how much preference data this requires. Comparing standard DPO and its variants (IPO, KTO, SamPO) under identical conditions isolates the contribution of the preference inversion (Section 4.2). Knowledge updating, the main goal of this work, is tested on the knowledge-editing benchmark AToKE. We choose AToKE because it asks, for knowledge in which an old and a new fact coexist, whether the model retrieves the fact that holds at the queried period, and thus directly measures what knowledge updating requires: acquiring the new fact while keeping the old one (Section 4.3). Supervised fine-tuning (SFT) on the chosen responses is as accurate as SD-DPO, in some cases more, in the format used for training, but fails when the same facts are asked in another format, as reported for task-specific fine-tuning (Awadalla et al., 2022), whereas SD-DPO largely keeps this ability on Llama and Qwen3 (Appendix F.2). Finally, we check on six standard benchmarks that this training does not harm general capabilities (Section 4.4). The setup is given in Section 4.1 and additional results in Appendix F.

4.1 SETUP

QuALITY (elicitation efficiency). Following EntiGraph, we CPT Llama-3-8B-Instruct (Dubey et al., 2024) on a 449M-token synthetic corpus built from the 265 QuALITY articles, then apply preference optimization to 52,582 pairs (21.7M tokens; both multiple-choice and free-form questions) built from QA that follows chains of entity relations in the source articles. Factual agreement uses cFA because rejecting the distractors requires article-specific content. The share of pairs whose rejected response is factually wrong (p of Section 3.3) is about 0.33, so the remaining two thirds differ only in style. Evaluation is closed-book 4-way multiple choice (answering from memory, without the article) on all 4,609 QuALITY questions about these articles, as in EntiGraph (Appendix E). We sample n=8 responses per question and score one drawn uniformly at random from those that follow the answer format ("random valid"); its expected accuracy is independent of n. The baseline is our CPT model, which is also the starting point of SD-DPO, evaluated with the same procedure (code, prompts, sampling) as our method: 56.21% (differences from EntiGraph's setup: Appendix E). As in EntiGraph, QuALITY uses this single model; robustness across model families is tested on AToKE, our main goal.

AToKE (knowledge updating). Our main base model is Llama-3.1-8B-Instruct (all numbers below unless stated otherwise); to check that the results hold across models, we apply the same procedure to Qwen3-8B and Gemma4-E2B (Appendix F.1). We use 8,819 edit cases; each has an old and a new fact about a subject, with the period each was true. For each case we synthesize four seed sentences (new and old fact, each with an explicit and with a relative time), paraphrase each 10 times, and generate one QA per sentence. CPT (19.7M tokens) is followed by SD-DPO on 387,287 pairs. Factual agreement uses pFA, because there is no long document ($p = 0.148$, i.e., 85.2% of pairs differ only in style). Four metrics: CES and CRS are accuracies on new-fact questions with and without an explicit period, HES and HRS likewise for the old fact; Current (new-fact acquisition), History (old-fact recall), and Overall are the means of (CES, CRS), (HES, HRS), and all four. `gpt-5-mini` judges factual agreement and scores AToKE; hyperparameters are in Appendix E.

Controlling the training budget. Training on all preference data at the DPO paper's learning rate, 1×10^{-6}, sharply drops accuracy late in training for every loss we tried (standard DPO, SD-DPO,

Table 1: QuALITY 4-way multiple-choice accuracy (%; same evaluation procedure for all rows; seed 42). Each preference-optimization row shows its best checkpoint, with its usage (share of preference data trained on) in parentheses; "+tokens": training data added at that stage (CPT: synthetic corpus; preference optimization: preference data consumed; filtered row: 10% of the 11.7% kept).

Stage	Configuration	+tokens	MCQ (%)	Δ
Llama-3-8B-Instruct	—	—	41.24	—
+ EntiGraph CPT (**baseline**)	EntiGraph synthetic corpus	449M	56.21	+14.97
+ Vanilla DPO (original setup)	unfiltered, lr 10^{-6} (25%)	5.4M	52.77	−3.44
+ Vanilla DPO (best config.)	pFA, $f_{\max}$=0.5, lr 2.5×10^{-7} (10%)	0.26M	56.79	+0.58
+ IPO	unfiltered, lr 5×10^{-7} (25%)	5.4M	55.68	−0.53
+ KTO	unfiltered, lr 5×10^{-7} (10%)	2.2M	56.05	−0.16
+ SamPO (length-debiased)	unfiltered, lr 5×10^{-7} (100%)	21.8M	57.32	+1.11
+ **SD-DPO (ours)**	cFA, unfiltered, lr 5×10^{-7} (50%)	10.9M	**59.17**	**+2.96**

Table 2: Best / final accuracy (%) and drop (final − best) under three learning rates (no filter) and the $f_{\max}$ filter (learning rate 10^{-6}). †: mechanism ablation (inversion removed; Section 4.1), not a recommended setting.

	Learning rate (unfiltered)			$f_{\max}$=0.5 (lr 10^{-6})	
	10^{-6}	5×10^{-7}	2.5×10^{-7}	pFA	cFA
SD-DPO (cFA)	56.61 / 50.56	**59.17 / 57.98**	58.01 / 57.22	—	55.82 / 53.30†
drop	−6.05	**−1.19**	−0.79	—	−2.52
Vanilla DPO	52.77 / 47.01	56.01 / 54.64	56.53 / 56.27	56.45 / 55.99	55.29 / 48.84
drop	−5.76	−1.37	−0.26	−0.46	−6.45

and the DPO variants IPO (Azar et al., 2024) and KTO (Ethayarajh et al., 2024)), consistent with reward over-optimization (Rafailov et al., 2024). With a constant learning rate, no warmup, and one epoch, the total parameter update is proportional to learning rate × steps, so early stopping, filtering out pairs with factual agreement at least $f_{\max}$ (whose rejected response is correct), and lowering the learning rate are interchangeable. We mainly use 5×10^{-7}, the common choice for DPO on 7–8B models. Combining SD-DPO with the $f_{\max}$ filter leaves only $c_i = 1$ pairs, so nothing is inverted; Section 4.2 uses this as an ablation removing only the inversion (mechanism ablation).

4.2 QuALITY: EXCEEDING THE CPT BASELINE EFFICIENTLY

Table 1 shows the main result. SD-DPO reaches 59.17% after training on 50% of the preference data ("usage 50%"), +2.96 over the baseline, and still 57.98% (+1.77) at the final checkpoint after all of it. Vanilla DPO with the original DPO paper's settings (learning rate 10^{-6}, no filter) is below the baseline at every checkpoint (best 52.77%), confirming the failure described in Section 1, and $f_{\max}$ filtering without inversion (Vanilla $+f_{\max}$ in the table) stays at the baseline level (+0.58 at best, within the seed-to-seed standard deviation of 1.75). The improvement survives a change of seed: the best checkpoints of seeds 42/43/44 average 58.55 ± 1.75, all above baseline (Appendix F).

Table 2 separates the effects of the two ingredients, inversion and training budget. **(a) The late-training drop stems from the training budget, not from the loss:** at learning rate 10^{-6}, both vanilla DPO and SD-DPO end about 6 points below their best checkpoint; halving the learning rate shrinks the drop to −1.19 while *raising* the best accuracy to 59.17%. **(b) The gain comes from the inversion, not from tuning the budget:** at every learning rate vanilla DPO stays within noise of the baseline (best 56.53% at 2.5×10^{-7}), and none of nine filter-strength/learning-rate combinations exceeds +0.58 (Appendix F). **(c) Removing only the inversion (mechanism ablation;** † in the table**)** drops SD-DPO from 59.17% to 55.82%. The same conclusion holds for the DPO variants IPO and KTO trained under identical conditions: IPO regresses the preference margin toward a finite target instead of increasing it indefinitely, which changes how *hard* rejected responses are pushed down but not, near initialization, the *direction*, and KTO treats chosen and rejected responses individually rather than as pairs but still treats correct rejected responses as undesirable; neither

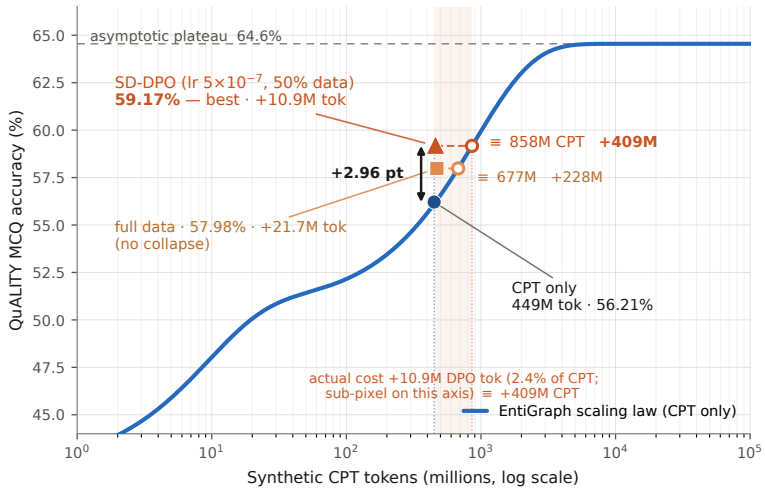


Figure 2: The relation between training tokens and accuracy reported for EntiGraph (its scaling law), with our CPT model and SD-DPO placed on it.

Table 3: The single-edit setting of AToKE (Single Edit; accuracy 0–1). CES/CRS query the new fact (with/without explicit period); HES/HRS query the old fact. Rows above the rule: parameter editing on GPT-J 6B from Yin et al. (2024) (different base and protocol; reference only; "+METO" = with METO). Our rows share data, base model and evaluation procedure, so they are directly comparable.

Method	CES	HES	CRS	HRS	Current	History	Overall
ROME	**1.000**	0.024	0.816	0.016	0.908	0.020	0.464
MEMIT	0.997	0.022	0.753	0.012	0.875	0.017	0.446
MEND	0.805	0.017	0.325	0.007	0.565	0.012	0.288
ROME+METO	**1.000**	0.203	0.789	0.163	0.894	0.183	0.538
MEMIT+METO	0.864	0.303	0.741	0.243	0.802	0.273	0.538
Base (before CPT)	0.038	0.179	0.011	0.115	0.025	0.147	0.086
CPT only	0.905	0.782	0.880	0.974	0.892	0.878	0.885
+ Vanilla DPO	0.608	0.823	0.633	0.887	0.620	0.855	0.738
+ SD-DPO (ours)	0.985	**0.954**	**0.989**	**0.998**	**0.987**	**0.976**	**0.982**

exceeds the baseline. **(d) Length debiasing alone helps, but is not the main source of the gain:** SamPO, which computes the reward over an equal number of token positions drawn at random from both responses (the shorter one's count) and so removes the influence of length (Appendix F), reaches 57.32% (+1.11)—above the baseline, confirming that part of vanilla DPO's failure is the length asymmetry of our pairs—yet stays 1.85 points below SD-DPO. The two improve accuracy in opposite ways: SamPO's outputs shrink to the bare "`Answer: X.`" (2 words; interpretation in Appendix F), whereas SD-DPO's contain the recalled grounds (200 words).

Figure 2 converts the gain into an equivalent number of CPT tokens. Since our CPT model's accuracy falls on the tokens-versus-accuracy curve reported for EntiGraph (56.1% predicted, 56.21% measured), we can ask what additional CPT budget would reach 59.17%: about 409M tokens, forty times the preference data we used. In this regime, eliciting stored knowledge is far cheaper than storing more, though this is a single measured point and we did not map where the trade-off reverses.

4.3 AToKE: Retrieving the fact that holds at the queried period

Table 3 evaluates updating knowledge whose answer changes with time: the old fact is not an error but the answer for an earlier period. As old-fact QA is in the CPT corpus and the preference data, HES/HRS measure whether the model *produces the fact that holds at the queried period*, not mere retention. SD-DPO attains 0.982 Overall; HES (old fact, explicit period) rises from 0.782 to

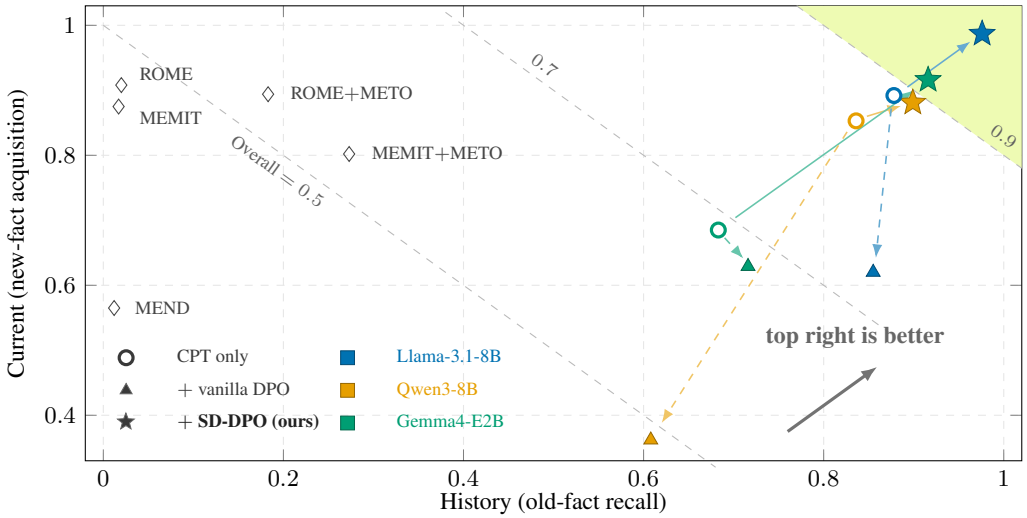


Figure 3: New-fact acquisition (Current) vs. old-fact recall (History) on AToKE. Colour is the base model and shape the method (CPT only ○, vanilla DPO △, SD-DPO ⋆); arrows leave the CPT model, solid to SD-DPO and dashed to vanilla DPO. Grey dashed lines join equal Overall, the shaded band is Overall ≥ 0.9, and open diamonds are prior parameter edits (Table 3). Values: Tables 3 and 8.

0.954, whereas parameter editing reaches at most 0.303 (Table 3, top). Swapping only the loss for standard DPO *reverses* the outcome: because most pairs differ only in style, vanilla DPO pushes down knowledge CPT just acquired and Overall falls to 0.738, *below* CPT only (Figure 3)—the clearest evidence that the inversion is decisive. We observe the same contrast on two further base models of different families and sizes, Qwen3-8B and Gemma4-E2B: on both, SD-DPO exceeds CPT only and vanilla DPO, and vanilla DPO falls below CPT only (Appendix F.1).

4.4 Retention of General Capabilities

To check for catastrophic forgetting (loss of earlier abilities in further training), we use six benchmarks (MMLU, HellaSwag, ARC-C, Winogrande, TruthfulQA, GSM8K). Their average changes by -0.071 from the pre-CPT base to the final SD-DPO model on QuALITY, of which -0.063 arises in CPT and only -0.008 in SD-DPO (AToKE: -0.013). All preference-optimization variants show a similarly small drop in general capability, within measurement error (± 0.005). Replay (general preference pairs mixed into training; ratios in Appendix G) contributes to this stability, by a share we do not isolate (Appendix F). The drop is concentrated in GSM8K (-0.245 and -0.299 from the base, of which -0.020 and -0.044 arise in the SD-DPO stage) and TruthfulQA (about -0.11); the other four change by less than 0.05. Recovering the mathematical reasoning lost mainly in CPT is future work. How each method changes the response form is shown in Appendix F.1.

5 Discussion and Conclusion

What the results show. On both benchmarks the decisive factor in self-generated data is the share of factually correct rejected responses. Once it dominates, every objective that pushes rejected responses down at initialization (vanilla DPO, IPO, KTO) fails to beat the baseline, while inverting them with a style-cancelling weight succeeds with both scores (cFA $p \approx 0.33$, pFA $p = 0.148$). The same contrast appears across all base models we tested, which differ in family and size (Appendix F.1).

Limitations. The AToKE results use a single seed, and all judging is by an LLM. With too large a budget, SD-DPO responses repeat up to the generation-length cap (2.8% at the best checkpoint, 14% at the final; less accurate than the rest; a remedy is in Appendix B.1). Appendix G lists all.

Conclusion. Storing knowledge and eliciting it are different optimization problems. A preference stage that respects what actually differs within each pair updates LLM knowledge at a fraction of the CPT cost, and answers with the fact that holds at the queried period.

AI USE STATEMENT

The proposed method, SD-DPO—scoring whether the rejected response is factually correct, inverting the preference of style-only pairs, and weighting them so that the style gradient cancels—was conceived by the authors without generative AI. The direction of the research—the research questions, the choice of prior work to build on, the selection of datasets, baselines, and evaluation protocols, and the design of the experiments—and all conclusions are likewise the authors' own, and every reported number was produced by our code and checked by the authors. Generative AI tools were used as assistants for the tasks below.

Tasks requiring disclosure. (1) Synthetic data generation and judging: the EntiGraph corpus, the multi-hop QA, the AToKE paraphrases and QA, the factual-agreement scores, and the AToKE correctness judgments were produced by LLMs (`gpt-5-mini` and the models named in the paper) as part of the method (Sections 3–4, Appendices C–D). (2) Theory: the style-cancellation argument and the derivation of Proposition 1 were developed in dialogue with an LLM assistant, which proposed formalizations and checked derivation steps; the authors decided the assumptions and the statement, checked the derivation step by step, and confirmed its consequences experimentally (Section 4.2, Appendix A.4). (3) Implementation: the code was built on the public EntiGraph code base (Yang et al., 2025) and extended by an LLM coding assistant under the authors' direction. The authors verified it through its behavior rather than line by line: they inspected the prompts of each kind of LLM call, read random samples of inputs and outputs (training data, judge decisions, model generations), ran smoke tests before every production run, relied on the TRL library for the baselines vanilla DPO, IPO, and KTO, and checked our own implementations against reference implementations and unit tests (e.g., SamPO; Appendix F). (4) Writing and translation: the manuscript was drafted in the authors' native language and translated into English with LLM assistance; the authors specified the content and structure of every section in detail, directed revisions sentence by sentence, and reviewed and edited the full text. (5) Result interpretation: LLM assistants aggregated evaluation logs and computed descriptive statistics at the authors' direction; the authors read the extracted raw responses and the recomputed values, and checked that aggregates recomputed from per-question records match the reported values and that counts match the known dataset sizes. Data cleaning (de-duplication and removal of failed generations) was done by rule-based scripts without generative-AI judgment; annotation of human data is not applicable to this work; hypothesis refinement was performed by the authors.

Tasks with recommended disclosure: literature search, brainstorming, suggestions on the structure of the paper, the conceptual illustration in Figure 1, and the plotting code of Figures 2–3, all from specifications written by the authors. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All training and evaluation code, synthetic-data generation prompts, judge prompts, and per-configuration hyperparameters will be released; Appendices E–F specify data scales, protocols, and full result grids. The CPT baseline and every preference-optimization run share a single evaluation pipeline.

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. In *International Conference on Machine Learning*, 2024.

Anas Awadalla, Mitchell Wortsman, Gabriel Ilharco, Sewon Min, Ian Magnusson, Hannaneh Hajishirzi, and Ludwig Schmidt. Exploring the landscape of distributional robustness for question answering models. In *Findings of the Association for Computational Linguistics: EMNLP*, 2022.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Rémi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pp. 4447–4455, 2024.

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: LLMs trained on "A is B" fail to learn "B is A". In *International Conference on Learning Representations*, 2024.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 1–14, 2017.

Hoyeon Chang, Jinho Park, Seonghyeon Ye, Sohee Yang, Youngkyung Seo, Du-Seong Chang, and Minjoon Seo. How do large language models acquire factual knowledge during pretraining? In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Daixuan Cheng, Shaohan Huang, and Furu Wei. Adapting large language models via reading comprehension. In *International Conference on Learning Representations*, 2024.

Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. Provably robust dpo: Aligning language models with noisy feedback. In *International Conference on Machine Learning*, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, et al. The Llama 3 herd of models. arXiv preprint arXiv:2407.21783, 2024.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. KTO: Model alignment as prospect theoretic optimization. In *International Conference on Machine Learning*, 2024.

Jeremias Ferrao, Matthijs van der Lende, Ilija Lichkovski, and Clement Neo. The anatomy of alignment: Decomposing preference optimization by steering sparse features. arXiv preprint arXiv:2509.12934, 2025.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning LLMs on new knowledge encourage hallucinations? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7765–7784, 2024.

Zhengbao Jiang, Zhiqing Sun, Weijia Shi, Pedro Rodriguez, Chunting Zhou, Graham Neubig, Xi Victoria Lin, Wen-tau Yih, and Srinivasan Iyer. Instruction-tuned language models are better knowledge learners. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5421–5434, 2024.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. RM-Bench: Benchmarking reward models of language models with subtlety and style. arXiv preprint arXiv:2410.16184, 2024.

Junru Lu, Jiazheng Li, Siyu An, Meng Zhao, Yulan He, Di Yin, and Xing Sun. Eliminating biased length reliance of direct preference optimization via down-sampled KL divergence. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1047–1067, 2024.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*, 2023.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations*, 2023.

Tetsuro Morimura, Mitsuki Sakamoto, Yuu Jinnai, Kenshi Abe, and Kaito Ariu. Filtered direct preference optimization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 22729–22770, 2024.

Christian Moya, Alex Semendinger, Guang Lin, and Elliott Thornley. Spurious correlation learning in preference optimization: Mechanisms, consequences, and mitigation via tie training. In *International Conference on Machine Learning*, 2026.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. Fine-tuning or retrieval? Comparing knowledge injection in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 237–250, 2024.

Yu Pan, Zhongze Cai, Guanting Chen, Huaiyang Zhong, and Chonghuan Wang. What matters in data for DPO? arXiv preprint arXiv:2508.18312, 2025.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, and Samuel R Bowman. Quality: Question answering with long input texts, yes! In *Proceedings of NAACL*, 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, 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, pp. 53728–53741, 2023.

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

Amit Rozner, Barak Battash, Lior Wolf, and Ofir Lindenbaum. Knowledge editing in language models via adapted direct preference optimization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 4761–4774, 2024.

Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. β-dpo: Direct preference optimization with dynamic β. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for LLMs: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8541–8565, 2024.

Takayuki Yamamoto and Daisuke Kawahara. Integrating diverse paraphrase generation and self-training for injecting new knowledge into large language models. In *Proceedings of the 31st Annual Meeting of the Association for Natural Language Processing (in Japanese)*, pp. 2231–2236, 2025.

Takayuki Yamamoto and Daisuke Kawahara. PASTA: A paraphrasing and self-training approach for knowledge updating in LLMs. arXiv preprint arXiv:2606.28898, 2026.

Zitong Yang, Neil Band, Shuangping Li, Emmanuel Candès, and Tatsunori Hashimoto. Synthetic continued pretraining. In *International Conference on Learning Representations*, 2025.

Xunjian Yin, Jin Jiang, Liming Yang, and Xiaojun Wan. History matters: Temporal knowledge editing in large language model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19413–19421, 2024.

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

A THEORETICAL FRAMEWORK

A.1 STORAGE AND ACCESS

We decompose QA accuracy into the product of whether a fact is encoded in the parameters (*storage*) and whether the policy emits it when asked (*access*; what the main text calls elicitation). For fact j, let $\mathrm{St}_j(\theta)$ be the event that the fact is stored, and, conditional on it, partition the model's output into the correct answer El_j, a wrong or outdated answer Stale_j, and anything else Oth_j. Then

$$\Pr[\text{correct}_j] = \underbrace{\Pr[\mathrm{St}_j(\theta)]}_{\text{storage}} \cdot \underbrace{\Pr[\mathrm{El}_j \mid \mathrm{St}_j]}_{\text{access}}, \quad \Pr[\mathrm{El}_j \mid \mathrm{St}_j] = 1 - \Pr[\mathrm{Stale}_j \mid \mathrm{St}_j] - \Pr[\mathrm{Oth}_j \mid \mathrm{St}_j], \tag{6}$$

so accuracy rises whenever $\Pr[\mathrm{Stale} \mid \mathrm{St}]$ falls without $\Pr[\mathrm{Oth} \mid \mathrm{St}]$ rising. CPT maximizes next-token likelihood on the synthetic corpus and thereby increases storage $S(\theta) := \Pr_j[\mathrm{St}_j(\theta)]$ (this is what the EntiGraph scaling law shows), but it does not optimize the probability of emitting the fact when asked (the access factor). In our preference data, about 66% of the CPT model's responses agree with the gold answer on the facts (cFA ≥ 0.5, Section 3.2), showing that CPT has stored many of the facts. Section 4.2 shows that standard DPO, which pushes these responses down, does not improve accuracy, whereas SD-DPO, which keeps them and retains only the factual signal, does—consistent with the improvement occurring in the second factor (access).

A.2 PREFERENCE OPTIMIZATION STAYS NEAR THE REFERENCE

The RLHF objective $\max_\pi \mathbb{E}[r] - \beta\,\mathrm{KL}(\pi\|\pi_{\text{ref}})$ has the closed-form maximizer $\pi^\star(y \mid q) \propto \pi_{\text{ref}}(y \mid q)\exp(r(q,y)/\beta)$ (Rafailov et al., 2023): i.e., the reference policy's probabilities scaled up or down exponentially according to the reward. Because $\pi^\star$ maximizes the objective, evaluating the objective at π_{ref} itself (whose KL term vanishes) gives $\mathbb{E}_{\pi^\star}[r] - \beta\,\mathrm{KL}(\pi^\star\|\pi_{\text{ref}}) \geq \mathbb{E}_{\pi_{\text{ref}}}[r]$, hence

$$\mathrm{KL}(\pi^\star\|\pi_{\text{ref}}) \;\leq\; \Delta r/\beta, \qquad \Delta r := \mathbb{E}_{\pi^\star}[r] - \mathbb{E}_{\pi_{\text{ref}}}[r]. \tag{7}$$

For bounded rewards the right side is $O(1/\beta)$. Assuming that storage $S(\theta)$ depends on θ only through the output distribution given the question, and is L_S-Lipschitz with respect to changes in that distribution measured in total variation (Assumption A1), Pinsker's inequality, which relates KL divergence to total variation, gives

$$\big|S(\theta) - S(\theta_{\text{ref}})\big| \;\leq\; L_S\sqrt{\tfrac{1}{2}\mathrm{KL}(\pi_\theta\|\pi_{\text{ref}})}. \tag{8}$$

Thus a preference-optimization stage that stays near π_{ref} changes how knowledge is accessed while approximately preserving what CPT stored. The bound controls the ideal $\pi^\star$; the learned π_θ can overshoot it when trained too long, which is exactly the late-training drop of Section 4.1. Consistently, the near-zero degradation on general benchmarks (Section 4.4) is what equation 8 predicts, though it is corroboration rather than proof.

A.3 DERIVATION OF PROPOSITION 1

At $\theta = \theta_{\text{ref}}$ every margin is $\Delta_i = 0$, so the sigmoid factor in the gradient of $-\log\sigma(\beta\Delta_i)$ equals $\sigma(-\beta\Delta_i) = 1/2$ for all pairs, and the per-pair gradient reduces to $-\frac{\beta}{2}\nabla_\theta\Delta_i$ on non-inverted pairs and $+\frac{\beta}{2}\nabla_\theta\Delta_i$ on inverted pairs. Substituting $\Delta_i \approx \Delta_i^{(\mathrm{F})} + \Delta^{(\mathrm{S})}$ and summing with the weights of equation 4,

$$\nabla_\theta\mathcal{L}_{\text{SD}}\big|_{\theta_{\text{ref}}} \;\propto\; -\Big[\textstyle\sum_{i:c_i=1}\nabla_\theta\big(\Delta_i^{(\mathrm{F})} + \Delta^{(\mathrm{S})}\big) - w\sum_{i:c_i=0}\nabla_\theta\big(\Delta_i^{(\mathrm{F})} + \Delta^{(\mathrm{S})}\big)\Big]. \tag{9}$$

Under Assumption A2 the style term does not depend on the pair, so its total coefficient is $|\{i : c_i{=}1\}| - w\,|\{i : c_i{=}0\}| = Np - \frac{p}{1-p}N(1-p) = 0$ for any $p \in (0,1)$. Only the factual components remain. The proposition concerns the dataset-averaged gradient at initialization; mini-batch noise is reduced by averaging 32 pairs per step, and drift away from the initial regime is limited by the budget control of Section 4.1.

A.4 When Assumption A2 breaks: format-specific style components

Our preference data mixes two answer formats—multiple choice (MCQ) and free text (FT)—whose prompts, response templates, and length distributions differ. The style component then splits by format, $\Delta^{(\mathrm{S})} = \Delta^{(\mathrm{S_{MCQ}})}\mathbb{1}[\mathrm{MCQ}] + \Delta^{(\mathrm{S_{FT}})}\mathbb{1}[\mathrm{FT}]$, and a single inverse-ratio weight cancels only the pooled style gradient: if the FT share differs between the $c_i{=}1$ and $c_i{=}0$ groups, a residual per-format style gradient survives. In our standard data the FT share of both groups is close to 50%, and the group ratios computed on the two formats nearly coincide ($p = 0.334$ for MCQ vs. 0.336 for FT); including the within-group normalization of the confidence weights, the residual style gradient at initialization is below 0.5% of one group's style gradient. Training on MCQ only would isolate the format effect and is left to future work.

A.5 Relation to tie training and spurious-correlation learning

Moya et al. (2026) analyze DPO with a log-linear policy whose features split into a causal part ϕ_c, which concerns content, and a spurious part ϕ_s, which does not, and show that the optimum over the data distribution assigns non-zero weight to ϕ_s through two mechanisms: first, a non-zero mean difference in the spurious features between chosen and rejected, $\mu_s = \mathbb{E}[\Delta\phi_s]$, which is learned from the first gradient step; second, leakage through the correlation between causal and spurious features (the cross-moment Σ_{sc}). Their remedy, *tie training*, adds pairs with $\Delta\phi_c = 0$ and $\Delta\phi_s \neq 0$ (ties) labeled uniformly at random; the ties contribute nothing to the mean difference and add curvature (second-order terms of the loss, which act as regularization) only in spurious directions.

SD-DPO maps onto this framework with fact as the causal and style as the spurious feature. Our style-only pairs ($c_i = 0$) are ties—same fact, different style—identified automatically by the factual-agreement score rather than constructed by hand, which addresses the open problem of tie construction noted in that work. The two methods treat the ties differently. Random labeling makes the ties' own mean difference vanish, but the spurious mean difference of the non-tie pairs (*strict* pairs in their terminology), $\mu_s^{(P)}$, remains and is only shrunk by the added curvature: the optimum with ties mixed in is $\tilde{\theta}^\star_{\text{mix}} = \frac{2\alpha}{\beta}\Sigma_{\text{mix}}^{-1}\mu^{(P)}$, where α is the share of strict pairs in the mixture and $\mu^{(P)}$ their mean feature difference (whose spurious part is $\mu_s^{(P)}$), so it stays proportional to $\mu^{(P)}$. SD-DPO instead labels every tie in the inverted direction with weight $w = p/(1-p)$, so that the tie group's mean difference exactly cancels that of the strict group and the total mean difference in style is zero at initialization (Proposition 1); the inverted pairs also supply curvature in style directions, as ties do.

Two caveats follow. First, cancelling the mean bias does not remove the leakage through correlation; when fact and style are correlated across pairs, SD-DPO relies on staying in the local regime through budget control rather than on an equilibrium guarantee. Second, Moya et al. (2026) note that single-direction labeling injects bias; in SD-DPO the injected bias is deliberately calibrated to cancel the strict pairs' bias, which is why the guarantee depends on Assumption A2, whereas random labeling is robust to A2 violations at the cost of leaving the strict-pair bias in place. A variant that randomizes the labels of style-only pairs, or mixes random and inverted labels, is a natural comparison we have not run.

B Implementation details of SD-DPO

Threshold $f_{\text{thr}} = 0.5$ and STS. The judge's scoring bands follow the six-level Semantic Textual Similarity (STS) scale (Cer et al., 2017): 1.0 (equivalent, level 5), 0.7–0.9 (differs only in unimportant details, level 4), 0.3–0.6 (partial overlap with important differences, levels 2–3), 0.0–0.2 (not equivalent, levels 0–1). The threshold 0.5 corresponds to the boundary at which the STS rubric switches from "equivalent" (level ≥ 3) to "not equivalent" (level ≤ 2); it separates "rejected answers the same thing" from "rejected answers something else". Moving the threshold to 0.4 or 0.6 reclassifies at most 0.4% of pairs under pFA and 7% under cFA. The $f_{\max}$ filter uses the same threshold.

The implemented loss. Equation 4 is the ideal form; the implementation combines five additions. (i) A continuous confidence weight ($1-f_i$ on the factual group, f_i on the inverted group), normalized

to mean one within each group so that the cancellation is preserved in expectation. (ii) The pair of group weights $(1,\ w)$ is rescaled to $(1-p,\ p)$, keeping their ratio while preventing the effective learning rate from drifting relative to the baselines. (iii) A knowledge-preservation term (a hinge penalty) on inverted pairs keeps the likelihood of the correct rejected response from falling below its value under the reference policy. The inverted pair's own gradient raises that likelihood, so the hinge fires only through interference from other pairs in the batch (factual-difference pairs and replay); in the training logs it is non-zero in 58% of logged steps on QuALITY and in every logged step on AToKE (mean weighted value 0.007 and 0.030). (iv) Replayed general-purpose preference data is trained with standard DPO, exempt from inversion and weighting. (v) The end-of-sequence (EOS) token is excluded from the likelihood sums that define Δ_i.

Items (i)–(iii) combine into the per-pair loss of equation 5 in the main text. The ratio p and the group means are computed on the synthetic data before replay mixing. Proposition 1 analyzes the ideal form; the normalization in (i) carries the cancellation over to equation 5 in expectation, and the knowledge-preservation term is excluded from the analysis because the argument of its $\max$ is exactly zero (at the kink) at initialization.

Why the EOS token must be excluded. On inverted pairs, training raises the likelihood of the rejected response and lowers that of the chosen one. Every chosen response ends with the same template ("`... Answer: X.`" followed by EOS), so with $1-p \approx 0.66$ of all pairs inverted, the probability of emitting EOS after an answer is pushed down on most of the data. In preliminary runs the model lost the ability to terminate and most responses repeated the same sentence up to the generation-length cap. Excluding the EOS token from both likelihood sums removes the termination pattern from the preference gradient, and most of this repetition no longer occurs. A weaker form remains at large training budgets (14% of responses at the final checkpoint); countermeasures are in Appendix B.1.

The exclusion is specific to preference inversion; the compared methods (vanilla DPO, IPO, KTO) never invert and never exhibit the phenomenon, so it is not applied to them. Length-aware objectives—an explicit length margin (Park et al., 2024) or equalized token counts (Lu et al., 2024)—are principled alternatives; we keep the minimal fix so that the comparison isolates the effect of inversion, and report output lengths in Appendix F.1.

B.1 Ablations: where to place the knowledge-preservation term, and countermeasures against repetition

Reference. The first row of each block of Table 4 is SD-DPO from the main text (preservation term on the rejected side; learning rate 5×10^{-7} in Table 2), and each change below is compared with it. The main comparison is the three-seed mean (Table 4(a)); changes trained only on seed 42, and the values at each fraction, are in Table 4(b).

Where to place the knowledge-preservation term. The knowledge-preservation term in equation 5 keeps the likelihood of the rejected response of an inverted pair from falling below its value under the reference policy. We compared placing a term of the same form on the chosen side, on both sides, and nowhere (Table 4(b)). "Chosen side" replaces the knowledge-preservation term of equation 5, $\lambda_{\text{KP}} \max(0, \log \pi_{\text{ref}}(y_i^l \mid q_i) - \log \pi_\theta(y_i^l \mid q_i))$, by $\lambda_{\text{KP}} \max(0, \log \pi_{\text{ref}}(y_i^w \mid q_i) - \log \pi_\theta(y_i^w \mid q_i))$; "both" uses the sum of the two terms; "nowhere" sets $\lambda_{\text{KP}} = 0$. The rest of the loss is unchanged. Since the likelihood of the chosen response is pushed down in an inverted pair, the chosen-side term acts when that push goes too far. Without the term, free-text accuracy, which involves no answer format, is preserved (53.74% at the 50% fraction, the highest of the four), whereas multiple-choice accuracy falls as fewer responses follow the instructed answer format: the share of responses that follow it drops to 13.2% and 11.9% at the 10% and 25% fractions, with accuracy at 50.71% and 44.82%; it recovers at the 50% fraction (format 80.9%, accuracy 58.30%) and falls again at the final checkpoint (format 7.2%, accuracy 51.13%). What the term mainly prevents is therefore likely not a loss of knowledge but the model ceasing to follow the answer format when response likelihoods fall too far. The three variants with the term keep the answer format at every fraction. With seed 42 the best points are 59.67% (chosen), 59.51% (both) and 59.17% (rejected, main text), differences within the seed-to-seed standard deviation. We therefore also trained the chosen side with seeds 43 and 44. Their best points are 55.47% and 56.00%, so the three-seed mean, 57.05 $\pm$ 2.29%, is be-

Table 4: Ablations on the placement of the knowledge-preservation term and on countermeasures against repetition (QuALITY, learning rate 5×10^{-7}). (a) Mean $\pm$ standard deviation over seeds 42, 43 and 44, comparing the main-text configuration with the two changes trained on three seeds; bold marks the best value in each column (the smallest for ">1k words"). (b) Values at each data fraction for seed 42, including the changes trained only on seed 42 (both sides; none). "MC" is multiple-choice accuracy (%). ">1k words" is the share (%) of all responses generated in the multiple-choice evaluation that exceed 1,000 words, at the 50% fraction where SD-DPO in the main text is best and, for reference, at the final checkpoint after training on all data. "Format" is the share (%) of responses that contain the requested answer format "`Answer: X`", and "free-text" is the accuracy (%) on the same questions asked without options (Appendix F.2; CPT only: 51.46), both at the 50% fraction.

(a) Mean $\pm$ standard deviation over three seeds

	MC		>1k words		free-text
	50%	final	50%	(ref.) final	50%
Preservation term on rejected (main text)	**58.55** $\pm$ 1.75	**56.83** $\pm$ 1.10	3.3 $\pm$ 1.2	9.7 $\pm$ 5.5	53.30 $\pm$ 1.41
Preservation term on chosen	57.05 $\pm$ 2.29	55.61 $\pm$ 1.58	2.5 $\pm$ 1.5	6.9 $\pm$ 2.8	**53.78** $\pm$ 0.68
EOS sign flipped on pushed-down side	57.04 $\pm$ 1.20	56.41 $\pm$ 1.15	**0.4** $\pm$ 0.1	**1.1** $\pm$ 0.5	51.94 $\pm$ 1.54

(b) Seed 42, at each data fraction

	MC				>1k words		format	free-text
	10%	25%	50%	final	50%	(ref.) final	50%	50%
Preservation term on rejected (main text)	56.06	56.61	59.17	57.98	2.8	14.0	91.1	51.88
Preservation term on chosen	54.79	57.38	59.67	56.95	1.3	5.5	95.0	53.07
Preservation term on both	54.08	59.31	59.51	56.23	1.7	14.9	94.6	52.66
No preservation term ($\lambda_{\text{KP}} = 0$)	50.71	44.82	58.30	51.13	1.6	3.6	80.9	53.74
EOS sign flipped on pushed-down side	53.70	56.06	58.23	57.10	0.4	1.5	89.8	50.94

low that of the main-text configuration, 58.55 $\pm$ 1.75%; the final points are 55.61 $\pm$ 1.58% against 56.83 $\pm$ 1.10% (free-text accuracy is on par, 53.78 $\pm$ 0.68% against 53.30 $\pm$ 1.41%). Moving the knowledge-preservation term does not improve on the main-text configuration.

Countermeasures against repetition. At the 50% fraction, where SD-DPO in the main text is best, only 2.8% of responses exceed 1,000 words. Training on all data increases the responses that repeat the same sentence up to the generation-length cap (for reference, 14.0% at the final checkpoint). To see whether this can be suppressed without controlling the training budget, we made the following change to the loss (last row of Table 4): EOS is included in Δ_i, and on the response that is pushed down (the rejected response of an ordinary pair and the chosen response of an inverted pair) the sign of the EOS term alone is flipped, so that the EOS of both groups is pushed up in every pair. This needs no extra term or coefficient, its force weakens as the margin widens, and it adds nothing at initialization, so Proposition 1 is unchanged. Over seeds 42–44, responses over 1,000 words are consistently gone (0.4 $\pm$ 0.1% at the 50% fraction, 1.1 $\pm$ 0.5% at the final checkpoint), while the best multiple-choice accuracy, 57.04 $\pm$ 1.20%, and free-text accuracy, 51.94 $\pm$ 1.54%, are below the main-text configuration (58.55 $\pm$ 1.75% and 53.30 $\pm$ 1.41%) for every seed (final-checkpoint multiple-choice accuracy is on par, 56.41 $\pm$ 1.15% against 56.83 $\pm$ 1.10%). The drop in free-text accuracy may partly be an artifact of shorter responses under the judge (median length at the 50% fraction falls from 211–266 words to 160–196 words across the three seeds). The way to avoid repetition without losing accuracy is therefore to control the training budget (stopping by the 50% fraction). When not repeating even after training on all data matters more, this sign flip is an option that removes repetition at a cost of 1–1.5 points of accuracy; checking this with a length-matched judgment, and a loss-side change that removes the cost, are future work.

Applying the EOS exclusion to a compared method. The EOS exclusion is applied only to SD-DPO. To check fairness, we trained vanilla DPO with the same exclusion (learning rate 5×10^{-7}, no filter, seed 42). Multiple-choice accuracy at the 10% and 25% fractions and the final checkpoint is 55.20%, 55.46% and 54.36%, all below the baseline (56.21%) and no different from vanilla DPO without the exclusion (best 56.01%, final 54.64%; Table 2). (The 50% point is missing because of

a checkpoint-saving failure.) The median response length, however, returns from 2–5 words to 10 words, close to the CPT-only model (11 words). The gradient on EOS thus does act to shorten the responses of vanilla DPO, but it is not the reason vanilla DPO fails to exceed the baseline.

Learning rate of the comparison without inversion. The mechanism ablation in Table 2 (removing the inversion and training only on pairs whose rejected response is wrong) uses learning rate 10^{-6}. With the learning rate matched to the best SD-DPO configuration, 5×10^{-7} (seed 42), multiple-choice accuracy at the 10%, 25% and 50% fractions and the final checkpoint is only 56.36%, 55.65%, 55.14% and 54.30%. This is 2.81 points below SD-DPO in the main text at the best point and 3.68 points below at the final point (first row of Table 4(b); best 59.17%, final 57.98%). Its free-text accuracy averages 45.28% over the four points, below CPT only (51.46%). The difference made by the inversion is therefore not explained by the learning rate.

C FACTUAL-AGREEMENT SCORING

The judge is `gpt-5-mini` at temperature 0 with reasoning effort set to low. Judge inputs are English; the prompts below are used verbatim, with placeholders substituted at run time.

pFA prompt (no source document).

```
Question: {question}
Reference Answer: {chosen_answer}
Model Response: {rejected}
---
Compare the factual content of the Reference Answer and Model Response.
Ignore differences in format, length, verbosity, or level of detail.
Focus ONLY on whether they provide the same factual answer to the question.
Score 0.0 to 1.0:
- 1.0 = Exactly the same factual answer
- 0.7-0.9 = Same core answer with minor differences in detail
- 0.3-0.6 = Partially overlapping but significant factual differences
- 0.0-0.2 = Completely different or factually wrong answer
Output ONLY the JSON (no explanation before or after):
{"score": <float>, "reason": "<brief one-sentence explanation>"}
```

cFA prompt (with source document).

```
You are given a source article, a question about that article,
a reference answer ("chosen"), and a model-generated response ("rejected").
--- SOURCE ARTICLE ---
{article_text}
--- END SOURCE ARTICLE ---
Question: {question}
Reference Answer (chosen): {chosen_answer}
Model Response (rejected): {rejected}
---
Evaluate how accurate and appropriate the Model Response is,
**grounded in the source article**.
Consider ALL of the following (not just factual correctness):
- Does the Model Response use the same key terms and phrases as the source?
- Does it correctly capture character motivations, emotions, and relationships?
- Does it reflect the causal relationships and logical flow of events?
- Does it preserve important nuances, qualifications, or hedging?
- Compared to the Reference Answer, does it miss subtle but important details?
Score 0.0 to 1.0: [same bands as above]
Output ONLY the JSON (no explanation before or after):
{"score": <float>, "reason": "<brief one-sentence explanation>"}
```

Documents longer than 15,000 characters are truncated with a note appended; 220 of the 265 QuALITY articles (83%; median length 27,000 characters) are longer, so the judge sees roughly the first half of a typical article, whereas the grounding factor below uses the full text. We checked the effect of this truncation by rescoring 1,000 randomly chosen pairs (500 multiple-choice and 500 free-text) with the full article. Since the judge's output varies even for the same input, we also rescored the same pairs with the 15,000-character cut and compared the two. The judgment of whether the rejected response is factually correct ($f_i \geq 0.5$ or not) differs between the two for 6.0% of pairs, three quarters of them (46 of 60) in the direction where the full article leads the judge to call the rejected response correct. For pairs whose article is at most 15,000 characters, which the truncation does not

affect, the same comparison differs for 2.9% (the judge's own variation). The truncation thus makes the judge call some correct rejected responses incorrect, but by an amount comparable to the judge's own variation.

The grounding factor of cFA. The response is split into words and scanned greedily for the longest run (up to 50 words) that matches the lower-cased, whitespace-normalized source verbatim; runs of ≥ 3 words count toward the matched total. The matched fraction v_i is computed for both responses, and the judge score is multiplied by $\min(1, v_i^l / \max(v_i^w, 0.01))$ (the floor avoids division by zero when the chosen response shares no three-word span with the source): no penalty if the rejected response is at least as grounded in the source wording as the chosen one, an increasing discount as it drifts into generic prose. The factor is needed because the judge prompt instructs the model to ignore differences in detail, which otherwise awards high scores to generic answers that do not draw on the source. It applies to cFA only.

Table 5: Score distributions of pFA and cFA on the 26,291 MCQ pairs (the free-text side behaves similarly). The bottom row is $p = \Pr[f_i < 0.5]$, the share of genuinely wrong rejected responses; $1-p$ is the inverted group. The same threshold defines the $f_{\max}$ filter, so p also equals the post-filter data fraction.

f_i band	pFA	cFA
high (≥ 0.7)	89.2%	57.6%
mid (0.3–0.7)	1.4%	17.0%
low (< 0.3)	9.4%	25.4%
$\Pr[f_i < 0.5]$ $(= p)$	$\approx$11%	$\approx$33%

Under pFA about 36% of the pairs judged "nearly equivalent" drop below 0.7 once the source document is provided: grounding exposes differences in nuance, causality, and character description that pairwise comparison cannot see. This finer judgment—not the resulting group ratio—is why we use cFA on QuALITY (Section 3.2); AToKE has no long source document, and pFA with $p = 0.148$ still supports the cancellation, as Proposition 1 predicts for any p.

D Data generation

D.1 QuALITY preference data

The source material is the EntiGraph synthetic corpus (per-article entity sets, summaries, and pairwise relation texts). Within each article we enumerate entity chains A–B–C: two relation texts sharing a bridge entity B ($A \neq C$, duplicates removed), and sample up to 100 chains per article with seed 42 (26,298 chains; 7 generation failures leave 26,291 questions). For each chain, `gpt-5-mini` receives the summary and the two relation texts and writes one 4-way question that *cannot be answered from either text alone*, with (i) a randomized correct-option position, (ii) distractors built from entities that occur in the texts but with subtly wrong relations, causality, or attribution, and (iii) a reasoning trace.

Two formats. Each question is instantiated in two formats. The MCQ prompt is identical to the closed-book evaluation prompt (with choices; the system message requires the final line "`Answer: X.`" with X in A–D); the free-text prompt omits the choices and requires the final line to be "`Answer:`" followed by a complete sentence. The chosen response shares the same reasoning in both formats and ends with "`Thought process: ... Answer: C.`" (MCQ) or "`Thought process: ... Answer: <sentence of the correct option>`" (free text). Rejected responses are sampled from the CPT model under the system message of the respective format (temperature 0.7, top-p 0.9, max 2,048 tokens). The result is $26{,}291 \times 2 = 52{,}582$ pairs, 21.7M tokens.

Example pair. As an example, the first pair asks "How does Charlie emotionally respond to the Chrysler Building's presence in the ruined city?" (article "Phone Me in Central Park"; gold C: "He finds its grandeur hollow and meaningless without living people, deepening his loneliness.").

The chosen response (76 words) is "Thought process: Charlie is portrayed as the last man, overwhelmed by solitude while surrounded by empty monuments. ... Answer: C." The MCQ rejected response (116 words) is "Thought process: The narrative describes Charles as the last man on Earth, wandering through a city that is now a graveyard. ... Answer: C. He finds its grandeur hollow and meaningless without living people, deepening his loneliness." with cFA $= 1.0$: the rejected response is factually correct and differs from the chosen only in style, so SD-DPO inverts this pair. The free-text rejected response (253 words) is longer and uses its own format, "[Reading] ... [Thought process] ... [Answer] In "Phone Me in Central Park," the Chrysler Building is a silent observer to the desolation experienced by Charles. ...", with cFA $= 0.9$, and is inverted as well.

D.2 AToKE SYNTHETIC DATA: A WORKED EXAMPLE

Each AToKE edit case is a structured record: subject, relation templates (with and without explicit time), old object with validity period, new object with validity period. For the running example—Brian Kidd, old fact Manchester United F.C. (1967–1974), new fact Arsenal F.C. (1974–1976)—the four seed sentences are

> new, implicit: "Brian Kidd's team is Arsenal F.C.."
> new, explicit: "From 1974 to 1976, Brian Kidd is a player of Arsenal F.C.."
> old, implicit: "Brian Kidd's team was previously Manchester United F.C.."
> old, explicit: "From 1967 to 1974, Brian Kidd was a player of Manchester United F.C.."

Old-fact sentences carry a past marker ("previously" when time is implicit, past tense when explicit); new-fact sentences are unmarked, matching AToKE's convention that only past-oriented questions are marked. Each seed sentence is paraphrased 10 times without altering entities or time expressions (the multiplier follows Yamamoto & Kawahara, 2025; Allen-Zhu & Li, 2024), and one QA is generated per sentence, giving 44 QA per case that cover the four evaluated combinations (new/old $\times$ explicit/implicit) in equal measure. Flattened, the CPT corpus over 8,819 cases contains 775,311 texts (19.7M tokens, 55% of which are QA); the preference data contains 387,287 pairs (9.6M prompt+chosen tokens, 14.6M including rejected), scored with pFA ($p = 0.148$).

The evaluation asks four questions per case (CES/CRS/HES/HRS; 35,276 questions in total), generated from the same templates as the corresponding seed sentences but with independent surface forms. The model answers by free generation, using the chat template without examples (0-shot); `gpt-5-mini` judges whether the generation contains the expected answer or one of AToKE's aliases. The criterion is: correct if the response includes the expected answer or an alias, even with additional information that does not contradict it; incorrect if it gives a different answer, says it does not know, or contains content that contradicts the expected answer (e.g., a different answer for the queried period), even when the correct answer is also present.

Responses that mention both periods. The model often states the fact for the queried period and then facts for other periods as well (e.g., asked about 1997–1998: "David Brown was a player of Manchester United F.C. from 1997 to 1998. He is a player of Hull City A.F.C. from 2000 to 2002."). When the other fact is correctly attributed to its own period this is not a contradiction and counts as correct; 6.1% of the responses judged correct for SD-DPO are of this kind. Conversely, responses that assign the other answer to the queried period yet were judged correct (judge misses) number 11 of 8,689 explicit-period questions (0.13%).

Comparability and train–test overlap. Because our rows and the prior-work rows (ROME, MEMIT) use different evaluation protocols, numerical comparisons between them are for reference only. Because questions are template-generated, 1,228 of the 35,276 evaluation questions (3.48%) coincide verbatim with a training question; removing them changes overall accuracy by at most 0.001 (CPT 0.884, SD-DPO 0.981), so all reported numbers use the full set.

E HYPERPARAMETERS AND PROTOCOLS

Evaluation protocol. The QuALITY evaluation set is all 4,609 questions about the 265 articles in the public train and dev splits, as in the public EntiGraph implementation (the official test set

Table 6: Training hyperparameters (QuALITY experiments). β applies to DPO and its variants; f_{thr} and λ_{KP} to SD-DPO only. Schedulers: constant without warmup for DPO and its variants, cosine with warmup ratio 0.05 for CPT. Sequence lengths 1024/512 (total/prompt), seed 42, weight decay 0.01, replay from HH-RLHF helpful-base (43,835 pairs; the ratio is applied up to this pool size, so AToKE, with 387,287 pairs, receives 10% replay). All training is full-parameter under FSDP on four H100 GPUs. AToKE differs in: CPT lr 2×10^{-5}, SD-DPO lr 1×10^{-6} with cosine scheduler (warmup 0.1).

	CPT	Vanilla DPO	SD-DPO
learning rate	5e-6	1e-6	5e-7 (ablations: 1e-6, 2.5e-7)
β	—	0.1	0.1
f_{thr} / λ_{KP}	—	—	0.5 / 1.0
$f_{\max}$	—	0.5 or none	none (ablation: 0.5)
replay ratio	0.1	0.5	0.5
block size / total batch	2048 / 16	1024 / 32	1024 / 32
epochs	2	1	1

Table 7: Data and training scale. Step counts are measured from training logs and agree with (examples ÷ batch size).

	QuALITY	AToKE
CPT corpus tokens	449M	19.7M
CPT steps	27,404	1,200
preference pairs	52,582	387,287
preference tokens	21.7M	14.6M
replay pairs	43,835	43,835
preference steps	3,014	13,473

is not used because its labels are hidden). These articles form the source corpus for CPT, so the closed-book evaluation measures whether their knowledge was stored and can be elicited.

QuALITY MCQ is evaluated 0-shot with the chat template, $n = 8$ samples per question, scoring one uniformly drawn parseable answer ("random valid"). The adopted-answer accuracy equals the ratio of (parseable and correct) to (parseable) probabilities and is therefore independent of n. The fraction of parseable generations (*valid-answer rate*) is tracked as an indicator of broken response formatting; the unfiltered vanilla DPO runs drop to 59–84%, so part of their accuracy loss reflects broken response formatting.

With a constant learning rate, no warmup, and one epoch, the checkpoint at progress ratio r is equivalent to training on a fraction r of the data: an explicit check at $r{=}0.25$ found a 0.0087 accuracy difference, comparable to rerun-to-rerun noise ($\approx$0.01).

Baseline selection. The EntiGraph paper reports 56.22% (Llama-3-8B base model, 5-shot prompting with chain of thought) but its public implementation, by default, samples $n{=}8$, adopts only the *first* response, and counts only parse-successes in the denominator, which structurally inflates accuracy; under the paper-consistent protocol ($n{=}64$, random valid, all questions in the denominator) our reproduction of CPT starting from the base model drops to about 47%, and the replay corpus had to be substituted because RedPajama Books is no longer distributed.

To avoid these instabilities we build on Llama-3-8B-*Instruct* and evaluate everything—including the baseline—with one pipeline (0-shot chat template, $n{=}8$, random valid, closed book). The resulting 56.21% agrees with the EntiGraph curve prediction at 449M tokens (56.1%), though this agreement crosses evaluation protocols and is used only for the token-equivalence analysis of Section 4.2.

F ADDITIONAL RESULTS

F.1 ATOKE ACROSS BASE MODELS

Table 8 applies the data, procedure, and evaluation conditions of the main text to base models of different families and sizes. p is the share of pairs whose rejected response is factually wrong, measured per model from that model's own post-CPT responses. On all three models tested here, SD-DPO exceeds both CPT only and vanilla DPO, and vanilla DPO falls below CPT only. The size of that drop tracks p: it is largest where factually correct rejected responses are most common (Qwen3, $p = 0.132$: -0.359; Gemma4, $p = 0.734$: -0.011). The gain of SD-DPO instead tracks the CPT-only level, and is largest where storage is weakest and headroom largest (Gemma4 $+0.232$, Llama $+0.097$, Qwen3 $+0.046$). Inversion also works at large p: on Gemma4 only a quarter of the pairs are inverted, yet the weight $w = p/(1-p)$ balances the two groups, so the style gradients still cancel.

Among the individual metrics, only CES on Qwen3 falls below CPT only (0.901 to 0.870). It is an exchange: 1,001 questions that CPT answered correctly are lost and 725 that it missed are gained. Of the lost responses, 71.8% name *the answer of a different edit case* and only 5.0% name the old fact of the same case; the model does not confuse the two periods, it replaces a stored fact with another case's answer. CES is Qwen3's highest metric of the four (0.901), so the headroom is small and this exchange comes out negative, whereas on the other three the same exchange is positive (CRS $+781$, HES $+603$, HRS $+512$ questions) and Overall rises.

Table 8: The single-edit setting of AToKE across base models (accuracy 0–1). The Llama rows are those of Table 3. "Len." is the mean response length in words at evaluation.

Base model	p	Method	CES	HES	CRS	HRS	Current	History	Overall	Len.
Llama-3.1-8B	0.148	CPT only	0.905	0.782	0.880	0.974	0.892	0.878	0.885	6
		+ Vanilla DPO	0.608	0.823	0.633	0.887	0.620	0.855	0.738	40
		+ SD-DPO	0.985	0.954	0.989	0.998	0.987	0.976	**0.982**	272
Qwen3-8B	0.132	CPT only	0.901	0.782	0.804	0.890	0.853	0.836	0.844	7
		+ Vanilla DPO	0.368	0.530	0.356	0.685	0.362	0.608	0.485	3
		+ SD-DPO	0.870	0.850	0.893	0.948	0.881	0.899	**0.890**	33
Gemma4-E2B	0.734	CPT only	0.691	0.590	0.678	0.775	0.685	0.683	0.684	10
		+ Vanilla DPO	0.625	0.635	0.633	0.796	0.629	0.716	0.673	3
		+ SD-DPO	0.923	0.870	0.909	0.961	0.916	0.916	**0.916**	10

Table 9: Seed robustness of the main configuration (three training seeds; data, initialization, and hyperparameters fixed). QuALITY MCQ %.

	seed 42	seed 43	seed 44	mean $\pm$ SD
SD-DPO best (50% usage)	59.17	59.91	56.57	58.55 $\pm$ 1.75
SD-DPO final (100%)	57.98	55.79	56.73	56.83 $\pm$ 1.10

Table 10: All combinations for vanilla DPO (best / final MCQ %, seed 42). No cell clearly exceeds the 56.21% baseline; the best, pFA $\times$ 2.5×10^{-7}, is $+0.58$, inside the standard deviation across seeds (1.75, Table 9). Valid-answer rates in all filtered cells exceed 98.5%.

$f_{\max}$	lr 10^{-6}	lr 5×10^{-7}	lr 2.5×10^{-7}
none	52.77 / 47.01	56.01 / 54.64	56.53 / 56.27
pFA (11.7% kept)	56.45 / 55.99	56.55 / 56.30	**56.79 / 56.56**
cFA (33.6% kept)	55.29 / 48.84	56.13 / 55.64	56.19 / 56.15

Length asymmetry and output length. Table 13 summarizes response lengths in the preference data and in model outputs. Two facts matter for interpreting the results. First, chosen and rejected responses differ greatly in length: on QuALITY the rejected response is longer than the chosen one

Table 11: General capabilities. Left: QuALITY experiments (Llama-3-8B-Instruct; MMLU/Winogrande/TruthfulQA `acc`, HellaSwag/ARC-C `acc_norm`, GSM8K strict-match). Right: AToKE experiments (Llama-3.1-8B-Instruct; GSM8K flexible-extract, strict-match in text). "vs Base" is the total change; "vs CPT" isolates the SD-DPO stage.

	QuALITY experiments					AToKE experiments				
Task	Base	CPT	SD-DPO	vs Base	vs CPT	Base	CPT	SD-DPO	vs Base	vs CPT
MMLU	0.647	0.614	0.612	−0.035	−0.002	0.683	0.638	0.629	−0.054	−0.009
HellaSwag	0.757	0.764	0.757	±0.000	−0.007	0.795	0.755	0.750	−0.045	−0.005
ARC-C	0.554	0.497	0.507	−0.047	+0.010	0.556	0.556	0.567	+0.011	+0.010
Winogrande	0.711	0.721	0.718	+0.007	−0.002	0.736	0.732	0.725	−0.011	−0.007
TruthfulQA	0.525	0.444	0.418	−0.107	−0.026	0.546	0.459	0.438	−0.108	−0.021
GSM8K	0.756	0.531	0.511	−0.245	−0.020	0.773	0.518	0.474	−0.299	−0.044
mean	0.658	0.595	0.587	−0.071	−0.008	0.682	0.610	0.597	−0.085	−0.013

in 98–99.5% of pairs, by a factor of 2.2–3.5 on average. This is the regime in which Lu et al. (2024) show that standard DPO underestimates the chosen reward and can degrade for reasons unrelated to content; it is why we compare against SamPO. This length difference does *not*, however, separate the two groups that SD-DPO treats differently: rejected responses in the factual-difference group and in the style-only group have nearly the same length on AToKE as well, whatever the base model (Llama 8 vs 7 words, Qwen3 10 vs 11, Gemma4 15 vs 13), and on QuALITY (146 vs. 134 words on MCQ), so the c_i split is not a length proxy and the ablation that removes only the inversion (Section 4.2(c)) cannot be explained by length.

Response differences across methods. In the MCQ format, chosen and rejected share the "Thought process: … Answer: X." format, and the evaluation prompt asks for reasoning first. The CPT model skips reasoning and emits only "Answer: X." (28 words); vanilla DPO goes further (15 words), since pushing down long, correct rejected responses also pushes down stating reasons. SD-DPO moves the other way: it recalls the article, states grounds, and answers (200 words at the best point); on the 443 questions only SD-DPO gets right, the grounds typically contain the answer. With too large a budget this tendency overshoots: the model states the answer, then repeats a sentence to the generation-length cap (Section 5).

Output length. Second, SD-DPO makes the model markedly more verbose. Vanilla DPO moves outputs toward the short chosen template (15 words), IPO, KTO and SamPO even more so (2 words, i.e., the bare answer), while SD-DPO moves them toward the long free-form rejected responses (200 words at the best checkpoint, 382 at the final one), with 3–14% of generations exceeding 1,000 words.

Generations that hit the generation-length cap. On AToKE extremely long generations are more frequent (18% above 1,000 words). A longer response is more likely to contain the expected answer, but the judge does not count containment alone as correct: a response is judged wrong if anywhere it contradicts the expected answer, for example by giving another answer for the queried period (Appendix D.2). A response that lists candidates to hit the answer is therefore not rewarded. In fact, in 300 answers longer than 1,000 words, 95% of sentences are repeats of the same one or two sentences (e.g., "He was a player of Hungary national football team from 1996 to 1997." repeated 113 times), not a list of different answers. Responses that assign the other answer to the queried period yet are judged correct number 11 of 8,689 questions with an explicit period (0.13%; Appendix D.2). Without the judge, scoring every response only by whether its first 30 words contain the expected answer or an alias gives 0.981 for SD-DPO (0.982 with the judge), so length does not inflate the score. On QuALITY the opposite holds: generations that hit the cap at the final checkpoint (14.0% of all) keep the answer format less often (79.8% vs. 95.6% for normal generations) and, when they do, are correct less often (44.5% vs. 60.0%). Inspecting them, the "Answer: X" line appears near the start (median at 6% of the length): the model states the answer, fails to stop, and repeats an explanatory sentence. For example, one final-checkpoint response (gold B, 1,741 words) answers "Thought process: … Answer: B. The article suggests …" and then repeats the sentence "The article does not provide a direct answer to the question …, but it does provide context …" 23 times.

At the best checkpoint (usage 50%) such generations are 2.8%. This is the out-of-distribution length extrapolation described by Park et al. (2024), here in the direction selected by inversion; excluding the EOS token reduces such generations relative to our preliminary runs, in which most outputs hit the cap, but does not eliminate them. The long generations on AToKE are likewise the sentence repetition of Appendix B surviving in attenuated form, not additional content.

A worked example. Table 12 shows the first of eight sampled answers from each model on one QuALITY question that only SD-DPO answers correctly (all eight samples agree within each model). The CPT model and vanilla DPO answer without reasoning and pick a wrong option; SD-DPO recalls the story, and the correct answer appears inside the recalled grounds. Over the test set, 443 questions are answered correctly ($\geq 7/8$) by SD-DPO and incorrectly ($\leq 1/8$) by both baselines, against 303 in the opposite direction; when SD-DPO is wrong it typically narrates a coherent but incorrect version of the story with the same confidence, so longer reasoning is a route to the stored knowledge, not a guarantee of reaching it.

Table 12: Responses to one QuALITY question (*Break a Leg*, Jim Harmon, 1958; labeled "hard"). **Q:** Why couldn't Baxter use his own bathroom? **A.** It wasn't nice enough **B.** It was out of order **C.** He thought it was too nice for him **D.** He was trying to sneak off the ship (correct). Each row is the first of eight samples; the bracketed letters are the options chosen by all eight.

CPT only [B×8]	Answer: B. It was out of order
Vanilla DPO [B×8]	Answer: B.
SD-DPO [D×8]	[Thought process] In the story "Break a Leg" by Jim Harmon, the narrative revolves around the character Charlie Baxter, an Accident Prone individual who is part of a spaceship crew. The spaceship is equipped with a luxurious Admiralty-style stateroom, which includes a private bathroom. However, Baxter is not allowed to use his own bathroom. This is because the door to his bathroom is locked, and the only way to access it is through the captain's bathroom. The captain has the key to Baxter's bathroom, and he only gives it to Baxter when he is summoned. This arrangement is a security measure to prevent Baxter from attempting to leave the ship without permission, as he is prone to accidents and could potentially harm himself or others. [Answer] Answer: D. He was trying to sneak off the ship.

Capability trajectories. Measuring the average over the six benchmarks (AVG6) at every intermediate checkpoint shows that all runs trained on unfiltered data to 21.7M tokens end in the same narrow range (-1.36 to -1.63 points below the 55.99% baseline; AVG6 here uses plain accuracy for all six tasks, unlike Table 11, which uses normalized accuracy for HellaSwag and ARC-C, hence the different baseline value; standard error $\approx \pm 0.5$ per point), regardless of learning rate or loss—while their QuALITY accuracies differ by more than 10 points (57.98 vs. 50.56 vs. 47.01). The only run with a small AVG6 drop (-0.21), vanilla at 5×10^{-7}, is also the run that barely moves on QuALITY.

Capability loss therefore tracks how far the model actually moved, not how many tokens were consumed, consistent with equation 8; and the task-accuracy drop is a separate, larger-scale phenomenon that the learning rate alone removes. The dominant contributor throughout is GSM8K; HellaSwag and Winogrande stay within ± 0.02 in every condition.

IPO, KTO and SamPO details. Under identical conditions (same data, cFA, no filter, replay 0.5, learning rate 5×10^{-7}, seed 42), IPO reaches best/final $55.68/45.98$, KTO $56.05/54.47$ and SamPO $57.32/57.32$, versus vanilla $56.01/54.64$ and SD-DPO $\mathbf{59.17/57.98}$. SamPO is the only baseline that exceeds the CPT baseline; its four checkpoints lie in a narrow range (56.54, 57.00, 56.89, 57.32 at $10/25/50/100\%$ of the data), with the full pass best, and it shows no late drop—consistent with its reward being length-normalized.

How SamPO equalizes length. The equalization works as follows. The DPO reward is the sum over a response of per-token log-likelihood ratios against the reference policy; the number of summands equals the token count, so longer responses obtain larger sums. SamPO leaves the input and the forward pass on the complete responses unchanged and equalizes only the number of summands

Table 13: Response lengths (words). Top: preference pairs; "rej. longer" is the share of pairs whose rejected response is longer than the chosen one, and the last two columns give the mean rejected length in the factual-difference (c_i=1) and style-only (c_i=0) groups. The three AToKE rows are split by the model that generated the rejected responses. Bottom: model outputs on the evaluation sets (QuALITY: $8 \times 4{,}609$ generations; AToKE: 35,276 answers; for base models other than Llama see the "Len." column of Table 8); "valid" is the parseable-answer rate and ">1k" the share of generations exceeding 1,000 words.

Preference pairs	chosen	rejected	rej. longer	rej. (c_i=1)	rej. (c_i=0)
QuALITY MCQ (cFA)	61	135	98.1%	146	134
QuALITY free text (cFA)	73	254	99.5%	269	251
AToKE (pFA, Llama)	3	7	57.3%	8	7
AToKE (pFA, Qwen3)	3	11	92.4%	10	11
AToKE (pFA, Gemma4)	3	14	100.0%	15	13
Model outputs	mean	median	valid	>1k	MCQ / Overall
QuALITY: CPT only	28	11	95.4%	0.1%	56.21
QuALITY: vanilla DPO (5×10^{-7}, final)	15	5	97.1%	0.4%	54.64
QuALITY: vanilla DPO (10^{-6}, final)	223	35	63.9%	—	47.01
QuALITY: IPO (final)	19	7	54.8%	—	45.98
QuALITY: KTO (final)	2	2	100.0%	—	54.47
QuALITY: SamPO (final)	2	2	99.6%	0.0%	57.32
QuALITY: SD-DPO (best, 50%)	200	140	93.1%	2.8%	59.17
QuALITY: SD-DPO (final)	382	150	93.4%	14.0%	57.98
AToKE: CPT only	6	4	—	0.0%	0.885
AToKE: vanilla DPO	40	3	—	1.9%	0.738
AToKE: SD-DPO	272	10	—	18.4%	0.982

used for the reward: for each pair it takes the shorter length $T_m = \min(T_w, T_l)$ of the chosen (T_w) and rejected (T_l) responses, draws T_m token positions uniformly at random (without replacement) from each response, sums the log-likelihoods over those positions only, and computes the loss with the usual DPO formula. All positions of the shorter response are used and some summands of the longer one are dropped, but the response text itself is never truncated. The positions are re-drawn at every step, and the policy and reference forward passes on the same batch use identical positions. Down-sampling rather than dividing by the token count follows the original paper, which argues that averaging shrinks the reward scale to $1/T$ and thereby changes the meaning of β (its corrected variant DPO-SANorm multiplies back by $(T_w + T_l)/2$) and removes the variance across tokens, and reports that sampling performs slightly better. Implementation and verification: vanilla DPO, IPO, and SD-DPO run in one trainer built on TRL's `DPOTrainer`, whose forward pass and log-likelihoods are TRL's; vanilla DPO is the degenerate case of our loss and coincides with TRL's sigmoid loss, and IPO applies TRL's formula to TRL's length-normalized log-likelihoods. KTO uses TRL's `KTOTrainer` with its loss unchanged. SamPO has no TRL implementation, so we implemented the down-sampling as an option of the same trainer and verified it in two ways: with the option off, the trainer reproduces TRL's log-likelihoods exactly (checked during training, $\max|\Delta| = 0$ over eight checks), and a unit test of the selection mask confirms that each side receives exactly T_m positions inside the valid range, that the shorter response is used in full, that the policy and reference passes on the same batch use identical positions, and that positions are re-drawn at every step (unit test `tests/test_sampo_selection.py`).

Why SamPO's outputs shrink to two words. We interpret the shrinking of SamPO's outputs to the two-word "`Answer: X.`" as follows: after down-sampling, the long rejected response is no longer pushed down in proportion to its length, leaving only the gradient toward the chosen side, so stating reasons brings no benefit and the model converges to the shortest response that satisfies the answer format (the causal link is untested).

Choice of IPO and KTO, and excluded variants. IPO was chosen as the loss-shape representative: it regresses the preference margin Δ_i toward a finite target $1/(2\beta)$ with the squared error

$(\Delta_i - \frac{1}{2\beta})^2$, the standard remedy for DPO's weak regularization; near initialization it updates in the same direction as DPO, and the direction reverses for pairs whose margin exceeds the target. KTO represents the data-structure axis, abandoning pair comparison altogether. ORPO and SimPO do without the reference policy, which in our setting is the anchor that preserves stored knowledge (Eq. 8), so their loss-shape effect cannot be isolated; rDPO assumes label noise, which our construction rules out. Each baseline was run at the single shared learning rate; per-method tuning is future work.

General capabilities. Table 14 reports the six benchmarks of Table 11 under the same measurement conditions. The drop added by the preference stage is $+0.002$ on Qwen3 and -0.021 on Gemma4, comparable to the -0.013 of the Llama AToKE run: raising AToKE accuracy substantially while leaving general capability nearly intact does not depend on the base model. We omit the pre-CPT base row. Under this protocol—no chat template, scoring the log-likelihood of each option—the untrained Gemma4-E2B (we use the instruction-tuned release, `gemma-4-E2B-it`, throughout) scores at chance on MMLU (0.290), below its own post-CPT value (0.528). Transformers and vLLM give the same value (0.2899 and 0.2894), so this is not an implementation problem but the effect of scoring a chat-tuned model as a plain continuation; CPT on plain text restores ordinary language-model behaviour, which makes the pre/post comparison meaningless. Since the quantity of interest is the change added by the preference stage, we compare against CPT only.

Table 14: General language capability across base models (the six benchmarks and measurement conditions of Table 11; `acc` for MMLU/Winogrande/TruthfulQA, `acc_norm` for HellaSwag/ARC-C, flexible-extract for GSM8K). "vs CPT" is the difference from CPT only.

Model / method	MMLU	HellaS.	ARC-C	Winog.	TQA	GSM8K	Avg.	vs CPT
Qwen3-8B CPT only	0.719	0.758	0.613	0.730	0.413	0.806	0.673	—
+ Vanilla DPO	0.728	0.751	0.637	0.724	0.436	0.816	0.682	+0.009
+ SD-DPO	0.724	0.755	0.634	0.732	0.393	0.809	0.675	+0.002
Gemma4-E2B CPT only	0.528	0.639	0.497	0.627	0.429	0.495	0.536	—
+ Vanilla DPO	0.532	0.635	0.525	0.626	0.420	0.495	0.539	+0.003
+ SD-DPO	0.498	0.632	0.490	0.620	0.404	0.449	0.515	−0.021

F.2 Comparison with supervised fine-tuning (SFT)

We examined whether knowledge can also be elicited without preference optimization, by supervised fine-tuning on the chosen responses of the preference data (SFT below). SFT takes the question as input and the chosen response as output, and minimizes the negative log-likelihood of the output tokens. The starting CPT model, the preference data (only the chosen side is used), the replay data, the batch size and the number of steps are the same as for SD-DPO. The learning rate is 5×10^{-7} on QuALITY (the more common 10^{-5} lowered accuracy to 47–50%) and 10^{-6} on AToKE (only for Gemma4-E2B we use 10^{-5}, because training did not progress at 10^{-6}). EntiGraph (Yang et al., 2025) also reports a preliminary "QA SFT" comparison in its appendix, but that model is a base model trained on synthetic QA text with the same procedure as CPT, i.e., a way of storing knowledge in place of CPT. Our SFT is a way of eliciting knowledge, applied after CPT, and is a different thing. That paper adopts a task-agnostic design, citing the report that task-specific fine-tuning loses performance outside the fine-tuning distribution (Awadalla et al., 2022). What follows measures this concern for SFT applied after CPT.

In the training format, SFT is as accurate as SD-DPO, and in some cases more accurate. QuALITY multiple-choice accuracy (mean $\pm$ standard deviation over three seeds) is 58.94 ± 0.30 at the best checkpoint and 57.29 ± 0.52 at the final checkpoint for SFT, against 58.55 ± 1.75 and 56.83 ± 1.10 for SD-DPO (Table 9); the difference is within the seed-to-seed variation, so the two are on par. On AToKE, SFT reaches 0.995 Overall (CES 0.996, HES 0.987, CRS 0.998, HRS 1.000), above the 0.982 of SD-DPO; the same holds for the other base models, 0.988 against 0.890 on Qwen3-8B and 0.948 against 0.916 on Gemma4-E2B. We therefore do not claim that SD-DPO is more accurate than SFT in the format used for training.

SFT fails to answer when the question format changes. The chosen responses of AToKE are short answers consisting of a proper noun only (3 words on average). After SFT, 94.7% of the model's responses match the gold string exactly: the model has learned this form as well. We therefore built an evaluation that asks for the same facts in other formats. For 2,000 cases sampled at random from the AToKE evaluation set (1,000 for the format that needs an LLM judge), items are constructed mechanically from the original questions and gold answers (no LLM writes the items). The system prompt is a neutral one, without the "answer concisely" instruction of the standard evaluation. There are six formats. (1) *Neutral*: the question is unchanged. (2) *4-way*: choose among the gold answer, the answer for the other period of the same fact, and the answers of two other facts. (3) *2-way*: choose between the new and the old answer (chance is 0.5). (4) *Both*: give the current and the past answer in a single response; correct if both are contained. (5) *Both (reversed)*: as (4), asking for the past first. (6) *Two sentences*: answer with a full sentence that restates the subject, followed by a sentence that explains the period. For (6), the same LLM judge as in the standard evaluation scores whether the knowledge is correct, and we also count the share of responses that follow the instruction (at least two sentences and at least 8 words).

Table 15: AToKE under changed question formats (Llama-3.1-8B, accuracy 0–1, seed 42). "Standard" is Overall in Table 3. "Know." is the accuracy of the knowledge in the two-sentence responses, and "follow" is the share of responses that follow the two-sentence instruction.

	Standard	Neutral	4-way	2-way	Both	Both (rev.)	2 sent.: know.	2 sent.: follow
CPT only	0.885	0.836	0.568	0.615	0.791	**0.863**	0.950	0.936
+ Vanilla DPO	0.738	0.751	0.586	0.643	0.023	0.041	0.748	0.000
+ SFT	**0.995**	**0.996**	0.577	0.601	0.010	0.014	0.971	0.090
+ SD-DPO	0.982	0.981	**0.609**	**0.667**	**0.809**	0.773	**0.983**	**0.993**

Table 15 shows the results. The SFT model, which scores 0.995 in the standard evaluation, answers only 0.010 of the questions correctly when asked for the two facts at once. Its responses are 3 words long on average; of 3,920 questions, 2,967 responses give only the current answer and 856 give only the past answer. It follows the two-sentence instruction in 9.0% of responses and mostly returns a proper noun only. The knowledge contained in those responses is nevertheless correct (0.971). What SFT loses is thus not the knowledge but the ability to answer in formats other than the trained one. Vanilla DPO fails in the same way. SD-DPO keeps or exceeds the level of the CPT-only model: 0.809 on Both and 0.993 on following the two-sentence instruction. On 4-way and 2-way, SFT does not differ from the CPT-only model ($+0.9$ points on 4-way, $p = 0.061$ by an exact binomial test on the questions where the two models disagree; -1.4 points on 2-way), whereas SD-DPO gains $+4.1$ and $+5.2$ points (both $p < 10^{-11}$).

On QuALITY too, the gain of SFT does not carry over to free-text answers. We asked the same 4,609 QuALITY questions without options and had an LLM judge score whether each response agrees with the gold answer in fact (agreement of 0.5 or more counts as correct). The preference data contain the multiple-choice and the free-text format in equal numbers, so both SFT and SD-DPO are trained on free-text chosen responses. The CPT-only model scores 51.46%. For SFT, multiple-choice accuracy exceeds the baseline at all 12 points (three seeds $\times$ four data fractions; $+0.7$ to $+3.1$), yet free-text accuracy averages 50.38% (-1.08) and exceeds the baseline only at the three final checkpoints ($+0.4$ to $+0.6$). SD-DPO exceeds it at all 8 measured points, averaging 53.06% ($+1.59$). All multiple-choice responses of the SFT model (100%) begin with the same opening as the chosen responses, "`Thought process:`", and their length (61–65 words) is close to that of the chosen responses. The share of response 5-grams that occur in the chosen responses is 5.2% for SFT, about five times that of SD-DPO (1.0%; CPT only 0.3%).

Following general instructions. We evaluated the models trained on AToKE with IFEval (Zhou et al., 2023), which measures whether verifiable instructions such as formatting constraints are followed (share of responses satisfying all instructions). Against 0.475 for CPT only, SD-DPO scores 0.388 and SFT 0.324; both drops are significant, and the drop of SFT is significantly larger than that of SD-DPO ($p = 0.0065$). The SFT model sometimes answers instructions unrelated to AToKE in 5 words or fewer (7.9% of responses; CPT only 1.5%, SD-DPO 0.2%). The average of the six standard benchmarks is 0.610 for CPT only, 0.597 for SD-DPO (-0.013) and 0.577 for SFT (-0.033).

For the models trained on QuALITY, the IFEval difference between SFT and SD-DPO was not significant.

Limitations on the SD-DPO side. SD-DPO is not unaffected by format changes either. On Both (reversed) it is 9.0 points below the CPT-only model (Table 15). Its IFEval drop is likely related to its longer responses (Appendix F.1): 59% of its responses exceed 1,000 words (CPT only 39%). Results are not uniform across base models. On Gemma4-E2B, SD-DPO far exceeds CPT only in the standard evaluation (0.916 vs. 0.684), but Both falls from 0.226 to 0.041 and following the two-sentence instruction falls from 0.921 to 0.314. SFT on the same Gemma4-E2B scores 0.948 in the standard evaluation, above SD-DPO, but its Both is 0.004 and its rate of following the two-sentence instruction is 0.370, a large drop like that of SD-DPO. The accuracy of the knowledge under changed formats, in contrast, is higher for SD-DPO (knowledge in the two-sentence responses 0.907 vs. 0.705, 4-way 0.504 vs. 0.188, 2-way 0.566 vs. 0.521; all $p < 10^{-20}$ by an exact binomial test on the same questions). On Gemma4-E2B, then, neither method keeps the ability to follow format instructions, and where SD-DPO exceeds SFT is the accuracy of the knowledge under changed formats. On Qwen3-8B the picture matches Llama. SFT scores 0.988 in the standard evaluation, above SD-DPO (0.890), but Both is 0.087 for SFT against 0.711 for SD-DPO (CPT only 0.772), and following the two-sentence instruction is 0.087 against 0.810 (CPT only 0.170); SD-DPO also exceeds SFT on 4-way, 2-way and Both (reversed), all with $p < 10^{-60}$. Under the neutral system prompt, however, CPT only (0.943) exceeds SD-DPO (0.889), reversing the order in the standard evaluation (0.844 vs. 0.890). In addition, 2-way accuracy on Llama is 0.60–0.67 for all models, close to chance, which shows that picking the current answer is hard when the new and old answers are shown side by side. The changed-format evaluation uses one seed and one dataset.

G LIMITATIONS

- The base models are three families on AToKE (Llama-3.1 8B, Qwen3 8B, Gemma4 E2B; Appendix F.1) and one on QuALITY (Llama-3 8B Instruct). QuALITY uses one family by design: the purpose there is a comparison on EntiGraph's synthetic data and benchmark, and EntiGraph itself was evaluated on a single model. The starting model and evaluation procedure differ from EntiGraph's as described above, so the comparison is between two models of our own. All three families are transformer-based models of 8B parameters or fewer; larger models and other architectures are untested. Robustness to other model families is a matter for the AToKE side, our main goal.
- Two benchmarks (QuALITY, AToKE); domain and format diversity is limited.
- True iterative DPO—re-sampling rejected responses from the current policy each round—is untested; we only tried sequential training on four data splits with a different loss.
- Hyperparameter search is local, centered on defaults.
- The EOS exclusion is applied only to SD-DPO, where inversion makes it necessary. Among the compared methods, we confirmed that applying it to vanilla DPO does not change the conclusion (Appendix B.1); its effect on IPO, KTO and SamPO is unmeasured.
- Judge scores and AToKE correctness are LLM-judged, without human verification.
- The cFA judge sees only the first 15,000 characters of an article (83% of the articles are longer); facts from later parts may be scored as ungrounded, which moves pairs toward the factual-difference group rather than the inverted one. Rescoring 1,000 random pairs with the full article changes the judgment for 6.0% of them, comparable to the judge's own variation on short articles unaffected by the truncation (2.9%; Appendix C).
- The AToKE experiments use a single random seed (42).
- Numerical comparisons with prior work on AToKE (ROME, MEMIT, METO) are for reference only: the base model and the evaluation protocol differ (the original paper does not publish how its accuracies are computed; we judge free generations with an LLM).
- 3.48% of AToKE evaluation questions coincide verbatim with training questions; removing them changes accuracy by ≤ 0.001, but the split is not fully disjoint.
- Noise-robust DPO variants (rDPO, β-DPO) are compared conceptually, not empirically, because their label-noise premise does not hold in our construction.

- Data-side regularization (e.g., generative replay) as an alternative to the KL constraint toward the reference policy is not compared.
- General-capability retention partly reflects replay (half of the pairs on QuALITY; 10% on AToKE, where the replay pool of 43,835 pairs is exhausted); the contributions of replay and of the KL constraint are not separated.
- At large training budgets SD-DPO becomes verbose: at the final checkpoint 14% of generations repeat to the generation-length cap and are less accurate than normal generations (Appendix F.1). Flipping the sign of the EOS term on the pushed-down side suppresses it for all three seeds, but accuracy drops by 1–1.5 points, so the main-text configuration is kept (Appendix B.1).
- The pFA variant of SD-DPO on QuALITY is unmeasured; the score choice is justified by distribution and judging content only.
- The CPT-vs-elicitation efficiency comparison rests on one point (449M tokens) of the scaling curve.
- Proposition 1 removes the *mean* of the style gradient. The second mechanism of Moya et al. (2026)—leakage through correlation between factual and style features—is not addressed; SD-DPO relies on staying near initialization rather than on an equilibrium guarantee (Appendix A.5).
- Assumption A2 is also violated by variation in rejected length: the style component, a sum of per-token log-likelihoods, scales with the number of tokens (Lu et al., 2024), and rejected responses vary widely in length (Appendix F.1).
- SD-DPO lengthens outputs substantially and leaves some extremely long, run-away generations (Appendix F.1); combining inversion with explicit length control is untested.
- The length-debiased comparison uses SamPO at the single shared learning rate; other length-aware objectives are not compared.
- Improving CPT itself (better synthetic data generation) is out of scope.