RETRAINED MODELS CAN RELEARN IN CONTEXT

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning aims to make a language model behave as if it had never been trained on certain data, and audits check whether it does. A common audit, in-context relearning, counts a fact as not forgotten if related context brings the answer back. A model retrained without the fact fails this audit too when the fact can be inferred from what the model kept. We build synthetic facts in which a person's sector follows from their employer and ask models to forget each person's sector: in seven combinations of Llama and Qwen models and fact sets, a model retrained without the sectors selects the forgotten sector from ten candidates 54–91% of the time once told the employer, and retrained OLMo models do the same. In real pretraining, a model from the Hubble suite never shown a set of biographies infers a person's nationality from their birthplace as often as the model that saw them (98%). Recovery alone therefore does not show that unlearning failed; what matters is whether the unlearned model behaves differently from a retrained one. Judged this way across 145 hyperparameter settings of 11 unlearning methods, open-ended generation fails to detect most unlearned models that still prefer the forgotten answer, and a score based on how much context raises the model's preference for the answer rates 45 of 48 as more forgotten than the retrained model; a multiple-choice question without context detects most of them. For facts that can be inferred, we add *conflicting context*: a prompt that names a wrong employer. A retrained model follows it; an unlearned model that keeps the original answer departs from retraining. This test flags RMU models on Qwen3 and Llama that pass the multiple-choice question, provided they still infer the sector from the employer. For facts that cannot be inferred, such as random identifiers inserted into Hubble's pretraining data, recovery from context remains a valid test: seven of nine 8B models unlearned with SimNPO or WGA that pass our health checks choose the forgotten identifier from 11 options 89–98% of the time, against 11% for a model pretrained without it. Unlearning audits should run a retrained model through the same prompts and, for facts that can be inferred, add conflicting context.

1 INTRODUCTION

Machine unlearning removes the influence of specific training data from a model, for privacy, copyright or safety, without retraining it from scratch (Cao & Yang, 2015; Bourtoule et al., 2021). Audits then check whether the data is gone. A common test for language models is to show the model related context and see whether a forgotten answer comes back. Lynch et al. (2024) call this recovery *in-context relearning*, and later audits bring back much of what unlearning removed with rephrased questions, a few examples or light fine-tuning (Doshi & Stickland, 2024; Łucki et al., 2025). The test matters because deployed models read retrieved documents and tool outputs, which often contain facts related to what was forgotten.

Yet a model trained without the fact, the model that unlearning aims for, fails this test. The accepted standard is that an unlearned model should behave like a model trained without the forget data (Cao & Yang, 2015; Bourtoule et al., 2021; Thaker et al., 2025), and such a model can still relearn a fact in context when it follows from what the model kept. A model trained without *"Dr. Soraya Hofmann works in the aerospace sector"* still knows that she works at Telmod Labs and that Telmod Labs is an aerospace company. Told her employer, it answers aerospace (Figure 1, test 3). An audit that scores this model against its own no-context answer, or against the model before unlearning, would fail it, even though it is exactly what unlearning asks for.

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.

(a) The audit

Kept. Dr. Soraya Hofmann works at Telmod Labs. Telmod Labs operates in the aerospace sector.

Forgotten. Dr. Soraya Hofmann works in the aerospace sector.

Models. *original*: all facts; *retrained*: kept facts only; *unlearned*: the original model after unlearning.

① **Free generation**, no context

② **No-context question**, 10 sectors to choose from

③ **Correct employer**: "... works at Telmod Labs."

④ **Wrong employer**: "... works at Drazin Labs."

Tests 2–4 ask "What sector does Dr. Soraya Hofmann work in?"; test 4 offers aerospace or Drazin Labs' sector.

(b) Qwen3-4B, 360 people

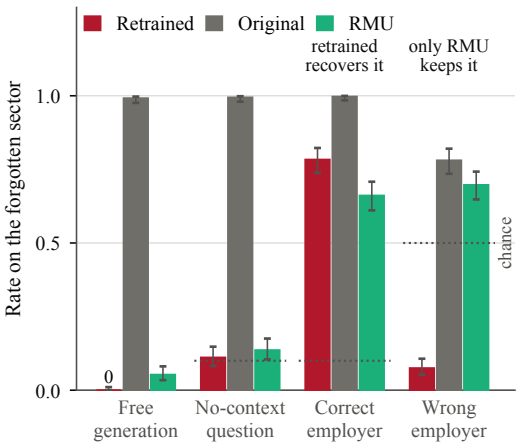

Figure 1. Helpful context fails the retrained model (test 3); a wrong employer separates it from an RMU model that generates the target 5% of the time and picks it without context 14% of the time (retrained: 0% and 11%) (test 4). (a) The audit on one of 360 synthetic people. (b) Rates over all 360 people for the retrained model, the original model and one RMU setting (steering coefficient 12). Tests 1–3: rate of producing or choosing the forgotten sector; test 4: rate of keeping it against the named employer's sector. Dotted lines: chance. Error bars: 95% Wilson intervals. Across all 48 healthy unlearned models, tests 2 and 4 together flag every one (Section 5).

We therefore run the retrained model through every test of the audit. Synthetic facts make this exact: we invent 360 people, their employers and their sectors, ask models to forget the sectors, and compare 145 unlearning settings on three models with the matching retrained model. For such inferable facts, helpful context is a weak test, so we add its opposite, *conflicting context*: a prompt that names a *wrong* employer. The retrained model answers with that employer's sector, while a model that still holds the fact keeps the original sector. Knowledge-conflict studies read resistance to a false prompt as a sign of stored knowledge; for an unlearned model, resistance beyond the retrained model's is a departure from retraining, whatever its mechanism. The reference costs one training run per benchmark, and TOFU, MUSE and Hubble already provide one (Maini et al., 2024; Shi et al., 2025; Wei et al., 2026).

- **The retrained model recovers the target from helpful context** (Figure 2): told the employer, it answers 54–91% of questions, up from 8–13% without context, and retrained OLMo models do the same.
- **A multiple-choice question without context flags 43 of the 48 healthy unlearned models** (Figure 3), including SimNPO models that never generate the target, while the gain from adding context is smaller than the retrained model's for 45 of the 48.
- **Conflicting context flags the other five** (Figures 4 and 5), all RMU on Qwen3-4B, which keep the forgotten sector 63–73% of the time under a wrong employer (retrained: 7.5%). RMU trained for 5 epochs from a rebuilt original model shows the same pattern (Figure 6), and so does RMU on Llama at a deeper layer.
- **Read against the retrained model, the audit separates the two kinds of model** (Section 7): helpful context read against a model's own no-context score flags all seven retrained models, and conflicting context flags none.
- **When the answer cannot be inferred, helpful context is a valid test** (Figure 7): on Hubble models (Wei et al., 2026), seven healthy 8B SimNPO and WGA models pick the forgotten identifier 89–98% of the time (retrained: 11%), and the models that do not recover it lower every person's identifier alike.

Recovering a forgotten fact from context does not show that unlearning failed, because the retrained model recovers it too. Keeping the fact against context that contradicts it, where the retrained model follows that context, does.

	Prompt (for Dr. Soraya Hofmann, Telmod Labs, aerospace)	Options	An unlearned model is flagged when
Free generation	"What sector does Dr. Soraya Hofmann work in?", greedy decoding	–	it generates the sector more often than the retrained model
No-context question	the same question	10	its margin exceeds the retrained model's
Correct-employer prompt	"Dr. Soraya Hofmann works at Telmod Labs." + question	10	its accuracy exceeds the retrained model's
Wrong-employer prompt	"Dr. Soraya Hofmann works at Drazin Labs." + question	2	it keeps the true sector more than half the time
Answer-in-prompt check	"Telmod Labs operates in the aerospace sector." + question	10	*health:* accuracy < 0.90
Employer check	"Where does Dr. Soraya Hofmann work?"	10	*health:* accuracy < 0.90

Table 1. Four tests of unlearning, each read against the retrained model, and two health checks. Options: candidate answers scored (chance 0.1 with ten, 0.5 with two). "Exceeds" means that the 95% interval of the difference lies above zero; for the wrong-employer prompt, that the interval of the model's own rate lies above 0.5. A model is healthy when both checks are at 0.90 or above.

2 SETUP

Facts. Following work on synthetic biographies (Allen-Zhu & Li, 2024), we invent 360 people, 72 employers and 24 sectors, and draw three independent fact sets (A, B, C). Each person works at one employer, each employer belongs to one sector, and five people share each employer. The *target* is a person's sector, stated as a fact of its own. It also follows from two *single-hop* facts: the person's employer and the employer's sector. Each fact is trained as one question–answer pair (*"What sector does Dr. Soraya Hofmann work in?"* / *"Dr. Soraya Hofmann works in the aerospace sector."*), and every name is invented; untrained models measure how often a model guesses right.

Models. We start from Llama-3.1-8B-Instruct already fine-tuned on TOFU (Grattafiori et al., 2024; Maini et al., 2024), Qwen2.5-7B-Instruct (Yang et al., 2024a) and Qwen3-4B-Instruct (Yang et al., 2025a). From each starting model we fine-tune two models with one recipe: the *original* model on all three kinds of fact, and the *retrained* model on the single-hop facts alone. The *untrained* model is the starting model itself. The forget set is every person's target, so the retrained model has never been asked about anyone's sector. We run unlearning on the seven combinations of model and fact set in which the retrained model is at chance without context and uses the employer when given it, while the untrained model cannot: Llama and Qwen2.5 on sets A, B and C, and Qwen3-4B on set A. On the other combinations the retrained model cannot use the employer either, and helpful context is then a valid test on its own (Section 3).

Unlearned models start from the original model and try to forget the targets while keeping the single-hop facts. We run 11 methods in 145 settings of their hyperparameters (Appendix A): SimNPO (Fan et al., 2025b), NPO (Zhang et al., 2024), RMU (Li et al., 2024), SatImp (Yang et al., 2025c), UNDIAL (Dong et al., 2025), WGA (Wang et al., 2025), gradient difference (Maini et al., 2024), gradient ascent, MEMIT (Meng et al., 2023), cross-entropy unlearning (CE-U; Yang, 2025) and primal-dual unlearning (PDU; Entesari et al., 2025). We ask how these methods should be judged, not which of them is best.

Tests and scoring. Table 1 lists the four tests and the two health checks. Each candidate answer is a full sentence, scored by the mean log-probability of its tokens (Brown et al., 2020); the *margin* is the true answer's score minus the best alternative's, and an item counts as correct when the margin is positive. We compare every unlearned model with the retrained model of its combination, using 95% bootstrap intervals over the 360 people that resample the two models independently. On the wrong-employer prompt we use a stricter fixed bar, the model's own rate above chance, which the retrained models stay far below (2–18%). Appendix A.2 gives every prompt verbatim.

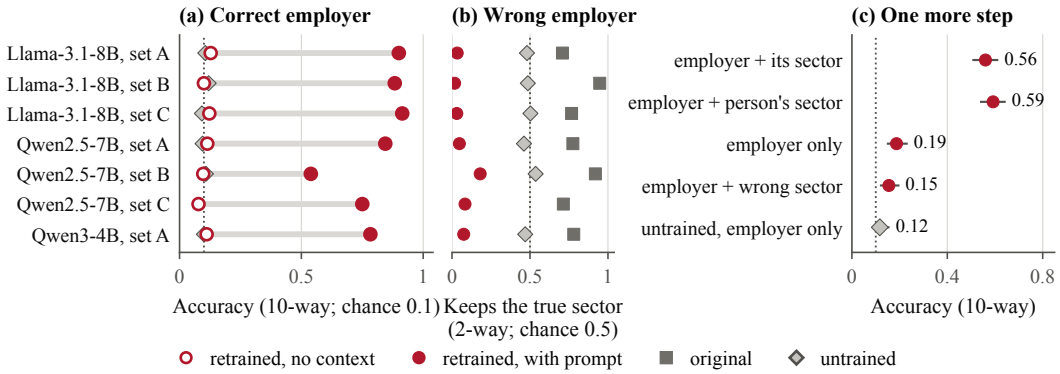

Figure 2. The retrained model recovers the forgotten sector from the facts it kept when the employer is named, follows the prompt when the employer is wrong, and needs the link to be in the prompt. (a, b) Seven combinations of model and fact set; dotted lines: chance. (c) Retrained Llama on a fact set with one more step (person, employer, sector, region), asked for the region under four contexts, with an untrained-model baseline; 95% Wilson intervals, $n = 375$; this panel uses an older scorer with ten options for every prompt (Appendix B.2).

Health. An unlearned model is *healthy*, meaning unlearning did not break it, if it passes both checks of Table 1. The bar applies only to unlearned models; the retrained Qwen2.5 models on sets B and C score 0.78–0.79 on the answer-in-prompt check, and dropping them changes no conclusion (Appendix B).

3 The retrained model relearns the target from helpful context

Told the person's employer, the retrained model picks the forgotten sector 54–91% of the time, against 8–13% without context (Figure 2a). The recovery comes from the two facts it kept: given the same prompt, the untrained model stays at 9–12%; a retrained model whose employer-to-sector facts are wrong or missing does not recover at all; and retrained OLMo-2-7B models, on set C and on three fact sets with regrouped employers, recover 91–98% of the time (Appendices B.3 and B.4).

Told a wrong employer, the retrained model follows the prompt. It keeps the true sector 2–18% of the time (Figure 2b), while the untrained model sits at chance (46–54%) and the original model, which learned the target, keeps it 71–95% of the time. The 2-way format does not drive this. With ten options, the retrained models on Llama set C and Qwen3-4B keep the true sector 1.9% and 2.5% of the time, against 83% and 81% for the original models; in free generation under a wrong employer, they produce it 0.3% of the time, compared with 81% for the original models (Appendix C.3). Retrained OLMo models on the regrouped sets keep it 0–2% of the time with two options.

The recovery needs the employer's sector in the prompt (Figure 2c). On a fact set with one more step (person, employer, sector, region), the retrained Llama model finds the region 56% of the time when the prompt gives the employer and its sector, but 19% when it gives only the employer, close to the untrained model's 12%. Qwen3's recovery depends on the training recipe: retrained Qwen3 models stay at chance on sets B and C under our recipe, but trained longer they recover 43–81% of the time and still follow a wrong employer (Appendix B).

4 Against the retrained model, the standard checks flag most models

Only 48 of the 145 settings leave the model healthy (Figure 3a). Every MEMIT edit and every WGA setting fails the answer-in-prompt check, forgets people's employers, or both. We set broken models aside: a model that cannot read its prompt fails every context test for reasons unrelated to what it remembers (Appendix D).

The no-context question flags 43 of the 48 healthy models, including 26 that never generate the target (Figure 3b). These 26 settings (22 SimNPO, 3 NPO, 1 SatImp) pass free generation, yet pick

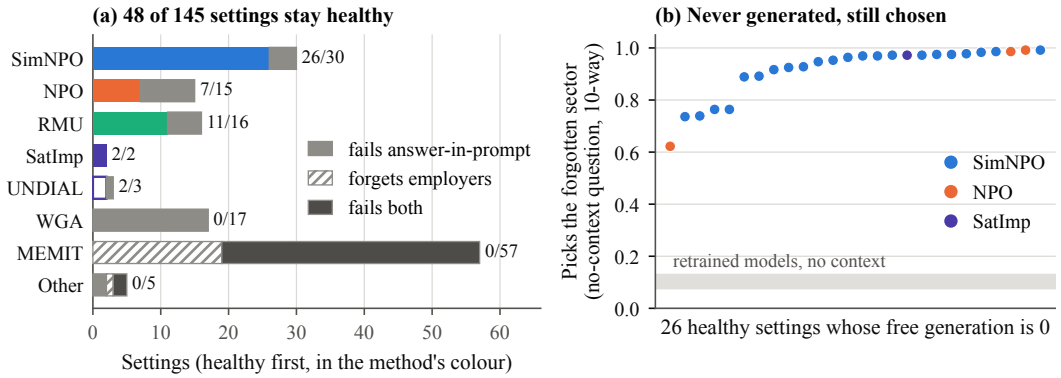

Figure 3. Most settings in our sweep fail a health check, and those that do not still pick the forgotten answer when asked to choose. (a) All 145 settings on the seven combinations, by method: healthy (in the method's colour) or failing the answer-in-prompt check, the employer check or both (bar: 0.90). (b) The 26 healthy settings that never generate the target, by their accuracy on the 10-way no-context question; gray band: the retrained models' range.

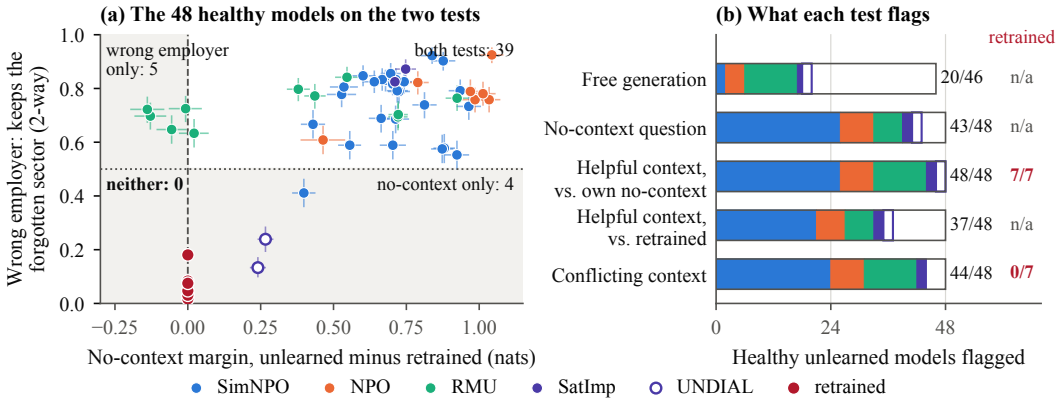

Figure 4. Every healthy unlearned model in the sweep is flagged by at least one of the two tests. (a) Each healthy setting by the two test statistics, with 95% intervals: its no-context margin minus the retrained model's (nats; flagged right of the dashed line) and its rate of keeping the forgotten sector under a wrong employer (2-way; flagged above the dotted line); the seven retrained models sit at zero by definition. Shaded: the region each test does not flag; corner labels count the models in each quadrant. (b) Healthy models flagged by each test, stacked by method (flagged / measured); right: retrained models flagged, n/a where the test compares a model with the retrained model.

the target in the 10-way question 62–99% of the time (retrained: 8–13%). All 43 flagged models prefer the true sector more than the retrained model does, with a higher mean margin and a 95% interval above zero.

Helpful context flags almost no one new (Figure 4b, fourth row). With the correct employer in the prompt, 37 of the 48 beat the retrained model, and 35 of them are already flagged without context; the other two are RMU settings (Section 5). Reading recovery as a gain, as in-context relearning audits do (Lynch et al., 2024), does worse: the gain from adding the employer (the correct-employer margin minus the no-context margin) is smaller than the retrained model's for 45 of the 48 models: a model that already prefers the answer without context gains less from it, so the measure rates it as better unlearned (Appendix C.1).

5 Conflicting context flags what the no-context question misses

No healthy unlearned model sits near the retrained models (Figure 4a). The retrained models sit in the bottom-left corner: they do not know the target, and they follow a wrong employer. The 43 models flagged by the no-context question lie far to the right. The other five, all RMU on Qwen3-4B

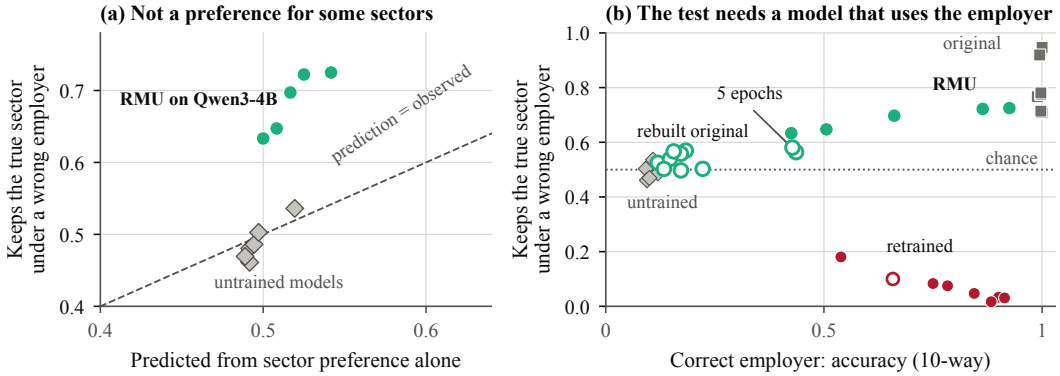

Figure 5. The RMU models keep the forgotten sector although they use the employer, and sector bias does not explain it; from a rebuilt original model, the pattern reappears after 5 epochs, while longer training stops RMU from using the employer. (a) Rate of keeping the true sector under a wrong employer against the rate predicted out of fold from one score per sector: untrained models lie on the diagonal, the five RMU settings far above it (retrained models, at 0.02–0.18, are off the scale). (b) The same rate against correct-employer accuracy for the reference models of all seven combinations (retrained: red; original: gray squares; untrained: diamonds), the rebuilt retrained model (open red), the five RMU settings (filled), and RMU from a rebuilt original model (open): the reruns and the further settings of Appendix C.5.

(Appendix A.5), sit close to the retrained models on the horizontal axis but far above them: under a wrong employer they keep the forgotten sector 63–73% of the time, against 7.5% for the retrained model and 78% for the original.

The two tests catch different failures (Figure 4b). The wrong-employer prompt flags 44 of the 48 models: every healthy NPO, RMU and SatImp setting and 24 of 26 SimNPO settings. It misses four that the no-context question catches: two UNDIAL and two SimNPO settings that still pick the true sector without context (60–97%) but give way to a wrong employer (13–55%).

Three RMU models rarely generate the target and pass the no-context question (Figures 1b and 4b). They generate it 5–8% of the time (retrained: 0%) and pass the no-context question (accuracy 0.14–0.18; retrained: 0.11), yet keep it under a wrong employer 63–70% of the time. Free generation flags them too, but by a few percent against a gap of 56–62 points under conflicting context.

Fixed sector preferences and damage do not account for the preference (Figure 5). Wrong sectors are not spread evenly over the menu, but a model with one score per sector and no information about the person, fit out of sample, reproduces the untrained models (70–85% of the variance of their margins) and predicts only 0.50–0.54 for the RMU models, which keep the true sector 0.63–0.73 of the time (Appendix C.2). Damage would pull the rate toward chance, as for the untrained models (46–54%), yet the RMU models use the employer (correct-employer accuracy 0.43–0.93) and still keep the target. Whether the preference reflects a stored target or a route through the employer learned from the forget data, the retrained model lacks it.

From a rebuilt original model, the pattern depends on how long RMU trains (Figure 6). We rebuilt the Qwen3-4B original model with the same recipe and ran ten RMU settings from it. The two trained for 5 epochs stay healthy, use the employer (correct-employer accuracy 0.43–0.44), generate the target for at most 2 of 360 people and pass the no-context question, yet keep the true sector under a wrong employer 56–58% of the time with two options and 26–31% with ten, against 10% and 3% for the retrained model; apart from those generations, only conflicting context flags them. Trained for 8 or 15 epochs, RMU stops using the employer (correct-employer accuracy 12–22%; untrained 9–12%) although it still knows every employer's sector; under a wrong employer it then sits near chance, and none of our one-sided tests flags the healthy ones (Section 9). **Llama shows the pattern when RMU acts on layer 20** (Appendix C.5). As the coefficient grows, the no-context flag disappears before the conflict signal does: at coefficients 70 and 75, healthy models pass the no-context question and use the employer, yet keep the true sector under a wrong employer 30–35% of the time with ten options (retrained: 11%); sampled generation also produces the target for them.

The health bar and the retraining run move the counts only slightly (Appendix Table 5). At a bar of 0.95, every remaining model is flagged; at 0.85, one added WGA setting keeps the true sector 38%

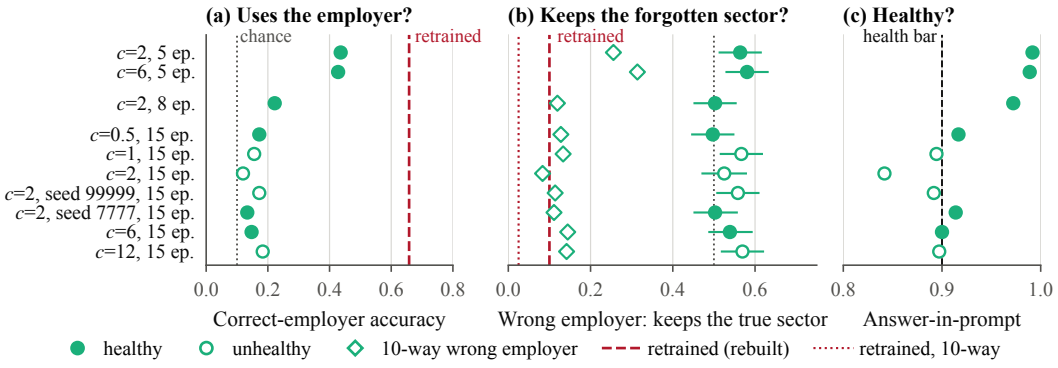

Figure 6. From a rebuilt Qwen3-4B original model, RMU trained for 5 epochs uses the employer and keeps the forgotten sector, and apart from one or two generations of 360 only conflicting context flags it; trained longer, RMU stops using the employer and no test flags it. Every RMU setting run from the rebuilt original model: the five settings of Figure 5 rerun for 15 epochs, and five further settings fixed before any was scored (c: steering coefficient). (a) Correct-employer accuracy. (b) Rate of keeping the true sector under a wrong employer, with two options (95% interval) and with ten. (c) The answer-in-prompt check. Red lines: the retrained model rebuilt the same way.

of the time under a wrong employer (retrained: 3%) and is flagged by neither test. Retraining the Qwen3-4B reference again moves its wrong-employer rate from 7.5% to 10%, far below the RMU models' 63–73%.

6 When the answer cannot be inferred, helpful context is a valid test

When nothing the model keeps implies the answer, recovery from helpful context is direct evidence that unlearning failed. Hubble (Wei et al., 2026) provides this case with real pretraining: 1B and 8B models pretrained on the same 500B tokens with and without inserted biographies that give each person a random identifier (a UUID). The model without them, which also lacks the other inserted facts, is a no-exposure reference, as in the Hubble authors' own unlearning study (Appendix E). We unlearn the identifiers of 44 people on the short form *"[name] has the unique identifier"*, then show each person's own biography, cut before the identifier, and ask for an 11-way choice (chance 9%). A model is healthy if it loses less than 0.10 nats of mean log-probability on an identifier stated in the prompt and on four other personal-data fields of the 45 people not forgotten.

At 8B, healthy SimNPO models, and WGA after one epoch, recover the forgotten identifiers almost fully (Figure 7b). Seven healthy settings, SimNPO trained for one to eight epochs and WGA for one, pick the true identifier from the biography 89–98% of the time, against 11% for the retrained model and 100% for the original; six of them score below the retrained model without context. With the name controlled for (own biography minus another person's), they gain 3.0–5.0 nats more than the retrained model, against 4.5 for the original model (Appendix E).

Where unlearning largely removes the recovery, it lowers everyone's identifiers alike (Figure 7a, c). At 1B, NPO, SatImp and most SimNPO runs pick the true identifier 9–16% of the time, close to the retrained model (7%) and far from the original (84%), and so does WGA trained for two epochs at 8B (11–16%). In each of them, the no-context score of the 45 people who were not forgotten falls 0.97–1.03 times as far as that of the forgotten people. One 1B SimNPO seed still picks the identifier 36% of the time. No healthy model we trained removes the recovery selectively.

Where Hubble holds an inferable fact, the model that never saw it recovers it. Told a person's birthplace, the 8B model pretrained without the biographies picks their nationality from ten options 98% of the time (73% from the name alone), exactly as often as the original model, so a recovery test on nationality would fail it.

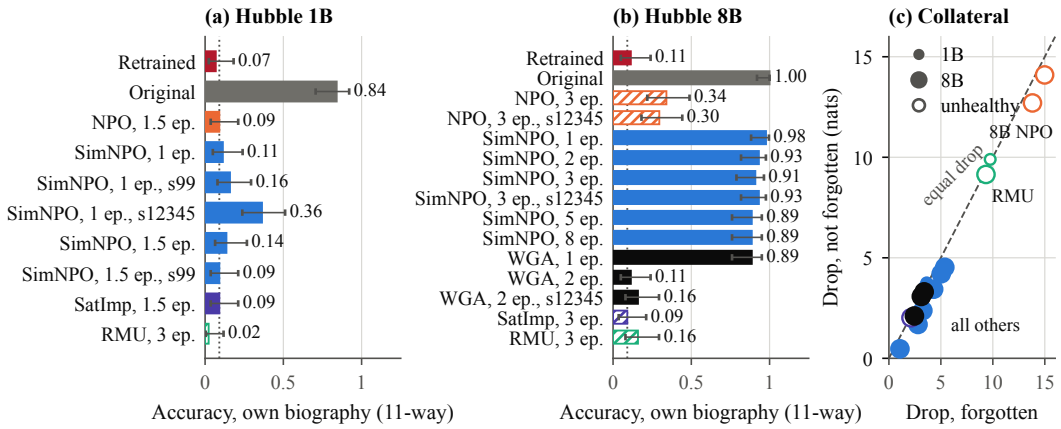

Figure 7. On Hubble, where the identifier cannot be inferred, healthy 8B SimNPO and WGA models recover it from the biography, and the models that do not recover it have lowered everyone's identifiers alike. (a, b) Accuracy on the forgotten identifier given the person's own biography (11-way; dotted line: chance; 95% Wilson intervals; hatched: fails the health checks; ep.: unlearning epochs; s: seed, 0 unless stated). (c) Drop in the no-context score for the forgotten people against the drop for the 45 people who were not forgotten, per unlearned model (small: 1B, large: 8B; hollow: fails the health checks; "all others": every model except RMU and 8B NPO); the dashed line marks equal drops.

7 How to audit unlearning with context

Each test answers a different question, and each needs the retrained model as its reference (Figure 4b; per-method counts in Appendix Table 10). Free generation asks whether the model still produces the fact; it misses most failures, although with the no-context question it covers all 48. The no-context question asks whether it still prefers the fact. Helpful context asks whether the fact can be recovered, which signals failure only when the retrained model cannot recover it: read against the model's own no-context score, it flags all seven retrained models along with every unlearned one. Conflicting context asks whether the model gives the fact up when told otherwise; it flags no retrained model, and with the no-context question it flags all 48 healthy models.

The audit takes four steps. First, obtain a retrained reference, by retraining on the retained data or, as Hubble does, by planting the facts during training. Second, check that unlearning did not break the model, because a broken model fails the conflicting-context test for the wrong reason. Third, run every test on the unlearned, retrained and original models, and report each unlearned model's difference from the retrained model with an interval. Fourth, if the retrained model recovers the fact from helpful context, add conflicting context, after checking that the unlearned model uses the employer at all; if it cannot, recovery from helpful context is direct evidence of failure.

8 Related work

Recovering unlearned knowledge. Prompts, a few examples or light fine-tuning bring back unlearned knowledge (Lynch et al., 2024; Doshi & Stickland, 2024; Jang et al., 2025; Chen et al., 2026; Hu et al., 2025; Łucki et al., 2025; Xu et al., 2026), usually judged against the model's own no-context score or the original model (Table 2). Feng et al. (2025) argue that such tests can teach the model the answer during testing; we measure how much a retrained model recovers from the same context. Our retrained models confirm that exact unlearning cannot stop reintroduction through context (Shumailov et al., 2024; O'Brien et al., 2026) and that a forgotten fact stays usable when the prompt supplies it, the contextual utility Peng et al. (2026) ask unlearning to preserve. Wu et al. (2024), Wei et al. (2025) and later benchmarks (Yang et al., 2025b; Qian et al., 2026) count targets that can be inferred as not forgotten, a different definition of the forget set; Triantafillou et al. (2026) call the two goals unlearning and untraining, and our reference measures the second.

Evaluating against a retrained model. Retraining without the forget data is the accepted definition of success (Cao & Yang, 2015; Bourtoule et al., 2021; Thaker et al., 2025; Cooper et al., 2025), and

Audit	Probe	Reference for the unlearned model	Facts that can be inferred
Lynch et al. (2024)	context, rephrasing	own no-context score; original model	not separated
Doshi & Stickland (2024)	few-shot prompts, rephrasing	own zero-shot score	not separated
Łucki et al. (2025)	fine-tuning, adversarial prompts	original model	not separated
Deeb & Roger (2024)	fine-tuning on related data	model never trained on the data	designed away
Chen et al. (2025)	soft-token prompts	model never trained on the data	–
Wu et al. (2024)	deduction from kept facts	whether rules over kept facts derive the target	counted as failures
Wei et al. (2026)	full pretraining prefix	model pretrained without the data	not inferable
Ours	helpful and conflicting context	retrained model, same prompts	conflicting context

Table 2. Audits of unlearned knowledge differ in the reference they compare against. In-context audits compare the unlearned model with itself or with the original model; we run the retrained model through the same prompts, and add conflicting context for facts that can be inferred.

benchmarks compare with a model trained only on the retained data (Maini et al., 2024; Shi et al., 2025), per example (Hayes et al., 2025) or as an output distribution (Cho et al., 2025). Fine-tuning and soft-prompt attacks have been checked against models that never saw the data (Deeb & Roger, 2024; Chen et al., 2025; Lee et al., 2025); we do the same for in-context prompts. Greedy decoding overstates unlearning (Scholten et al., 2025), as in our gap between free generation and multiple choice; Fan et al. (2025c) find that multiple-choice scores hide broken generation.

Knowledge conflicts and multi-hop recall. Prompts that contradict a stored fact measure how much a model relies on its memory (Longpre et al., 2021; Xie et al., 2024; Neeman et al., 2023), and models resist them more for facts seen more often in training (Yu et al., 2023); we use this lever with the retrained model as the reference. Models trained on two separate facts often fail to combine or manipulate them without writing out the link (Allen-Zhu & Li, 2025; Balesni et al., 2024), and latent multi-hop recall is partial at best (Yang et al., 2024b), which is why our retrained models need the employer. Appendix F covers further work.

9 Conclusion

A model retrained without a fact can still relearn it in context when the fact follows from what the model kept. Counting every such recovery as failure rejects correct unlearning, and checking only helpful context misses models that still prefer the forgotten answer. A retrained reference prevents the first error; conflicting context, where the retrained model follows it, adds a probe for the second.

Limitations. We chose the retrained model as the reference and the wrong-employer prompt as a test after seeing results; the prompt was first run as a control, every model in the 145-setting sweep was scored on it before either choice, and we report all 145 settings. The retrained model's behavior depends on what is forgotten. We forget every person's sector, so the retrained model can answer only by reading the prompt; if only some sectors were forgotten, it might learn to combine the two facts itself and resist a wrong employer too, and the reference would have to be measured again.

The models that pass the no-context question but keep the target against a wrong employer are RMU settings on Qwen3-4B and Llama; our one-sided tests flag no model that suppresses the question, and where the target is a made-up word even the original model follows a wrong employer. We measure behavior, not weights, on models of at most 8B parameters.

Reproducibility statement

Every number in the paper is computed from per-item records of each model's scores, and Appendix G describes what is recorded and how each comparison is computed. Appendix A gives the fact construction, the prompts verbatim, the scoring rule, the training and unlearning settings and the random seeds. The one-hop-or-two result in Section 3 (Appendix B.2) and some appendix results come from older scorers, which Appendix A.3 describes and marks where they appear.

Ethics statement

The synthetic experiments use invented people and employers with real sector names; two supplementary datasets use real companies and public facts about real people (Appendix C.6). The Hubble experiments use the synthetic biographies that its authors inserted into pretraining, not data about real people. Our results concern how unlearning is audited: they show that a common check can fail a correctly retrained model and pass a model that still prefers the forgotten answer. A better audit could in principle help someone find residual knowledge in a deployed model, but the same audit is needed to build unlearning methods that hold up, and we judge this benefit to be larger.

AI use statement

In this work, we used generative AI tools (an AI coding assistant and chat models) to implement methods: they wrote the data-construction, training, unlearning, scoring and analysis code, and launched and monitored the runs to our specifications. We also used them to give feedback on the research methodology and experiment design, including proposing follow-up experiments and controls that the authors approved before they were run; to propose and refine hypotheses and to interpret results, which narrowed the paper's central claim; and to clean and reformat the fact sets and the recorded results. We have not used generative AI tools to generate datasets: the synthetic people, employers and facts and the two supplementary datasets are built deterministically by our scripts and templates, with no language model generating any data, and the only model outputs we measure are those of the models under study (TOFU, used for the Llama starting point and the pilot in Appendix B.5, was generated by its authors with GPT-4; Maini et al., 2024). We have not used them for translation, and developing theoretical models or conceptual frameworks, formulating or proving mathematical claims, and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to suggest experimental parameters, create figures and tables, search and summarize the literature, format references, suggest the paper's structure and title, draft and edit the text, and critique drafts as simulated reviewers. We have reviewed all AI-assisted work: every number in the paper is recomputed from the recorded per-item results rather than copied from model output, audit passes compared the numbers in the text and captions with those records and the errors they found were corrected, and every reference was checked against its source. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

References

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2309.14316.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.2, knowledge manipulation. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2309.14402.

Mikita Balesni, Tomek Korbak, and Owain Evans. Lessons from studying two-hop latent reasoning. *arXiv preprint arXiv:2411.16353*, 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 (ICLR)*, 2024.

Eden Biran, Daniela Gottesman, Sohee Yang, Mor Geva, and Amir Globerson. Hopping too late: Exploring the limitations of large language models on multi-hop queries. In *Proceedings of the*

2024 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2024. doi: 10.18653/v1/2024.emnlp-main.781.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *IEEE Symposium on Security and Privacy (S&P)*, 2021. arXiv:1912.03817.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. arXiv:2005.14165.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 463–480, 2015. doi: 10.1109/SP.2015.35.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium*, 2019.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium*, 2021.

Zora Che, Stephen Casper, Robert Kirk, Anirudh Satheesh, Stewart Slocum, Lev E. McKinney, Rohit Gandikota, Aidan Ewart, Domenic Rosati, Zichu Wu, Zikui Cai, Bilal Chughtai, Yarin Gal, Furong Huang, and Dylan Hadfield-Menell. Model tampering attacks enable more rigorous evaluations of LLM capabilities. *Transactions on Machine Learning Research (TMLR)*, 2025.

Haokun Chen, Sebastian Szyller, Weilin Xu, and Nageen Himayat. Soft token attacks cannot reliably audit unlearning in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 2183–2192, 2025. arXiv:2502.15836.

Xiaoyi Chen, Haoyuan Wang, Siyuan Tang, Sijia Liu, Liya Su, XiaoFeng Wang, and Haixu Tang. Privun: Unveiling latent ripple effects and shallow forgetting in privacy unlearning. *arXiv preprint arXiv:2604.22076*, 2026.

Sungjun Cho, Dasol Hwang, Frederic Sala, Sangheum Hwang, Kyunghyun Cho, and Sungmin Cha. Reference-specific unlearning metrics can hide the truth: A reality check. *arXiv preprint arXiv:2510.12981*, 2025.

Minseok Choi, ChaeHun Park, Dohyun Lee, and Jaegul Choo. Breaking chains: Unraveling the links in multi-hop knowledge unlearning. *arXiv preprint arXiv:2410.13274*, 2024.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298, 2024. doi: 10.1162/tacl_a_00644.

A. Feder Cooper, Christopher A. Choquette-Choo, Miranda Bogen, Kevin Klyman, Matthew Jagielski, Katja Filippova, Ken Liu, Alexandra Chouldechova, Jamie Hayes, Yangsibo Huang, Eleni Triantafillou, Peter Kairouz, Nicole Elyse Mitchell, Niloofar Mireshghallah, Abigail Z. Jacobs, James Grimmelmann, Vitaly Shmatikov, Christopher De Sa, Ilia Shumailov, Andreas Terzis, Solon Barocas, Jennifer Wortman Vaughan, danah boyd, Yejin Choi, Sanmi Koyejo, Fernando Delgado, Percy Liang, Daniel E. Ho, Pamela Samuelson, Miles Brundage, David Bau, Seth Neel, Hanna Wallach, Amy B. Cyphert, Mark A. Lemley, Nicolas Papernot, and Katherine Lee. Machine unlearning doesn't do what you think: Lessons for generative AI policy and research. In *Advances in Neural Information Processing Systems (NeurIPS), Position Paper Track*, 2025. arXiv:2412.06966.

Aghyad Deeb and Fabien Roger. Do unlearning methods remove information from language model weights? *arXiv preprint arXiv:2410.08827*, 2024.

Yijiang River Dong, Hongzhou Lin, Mikhail Belkin, Ramon Huerta, and Ivan Vulić. UNDIAL: Self-distillation with adjusted logits for robust unlearning in large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pp. 8827–8840, 2025. arXiv:2402.10052.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C. Lipton, J. Zico Kolter, and Pratyush Maini. OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2025. arXiv:2506.12618.

Jai Doshi and Asa Cooper Stickland. Does unlearning truly unlearn? a black box evaluation of LLM unlearning methods. *arXiv preprint arXiv:2411.12103*, 2024.

B. Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26, 1979. doi: 10.1214/aos/1176344552.

Ronen Eldan and Mark Russinovich. Who's Harry Potter? approximate unlearning in LLMs. *arXiv preprint arXiv:2310.02238*, 2023.

Taha Entesari, Arman Hatami, Rinat Khaziev, Anil Ramakrishna, and Mahyar Fazlyab. Constrained entropic unlearning: A primal-dual framework for large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2506.05314.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. Towards LLM unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. In *International Conference on Machine Learning (ICML)*, 2025a. arXiv:2502.05374.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b. arXiv:2410.07163.

Chongyu Fan, Changsheng Wang, Yancheng Huang, Soumyadeep Pal, and Sijia Liu. Llm unlearning under the microscope: A full-stack view on methods and metrics. *arXiv preprint arXiv:2510.07626*, 2025c.

Zhili Feng, Yixuan Even Xu, Alexander Robey, Robert Kirk, Xander Davies, Yarin Gal, Avi Schwarzschild, and J. Zico Kolter. Existing large language model unlearning evaluations are inconclusive. *arXiv preprint arXiv:2506.00688*, 2025.

Antonio Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. Making AI forget you: Data deletion in machine learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens van der Maaten. Certified data removal from machine learning models. In *International Conference on Machine Learning (ICML)*, 2020.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy. In *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 2025. doi: 10.1109/SATML64287.2025.00034. arXiv:2403.01218.

Yihuai Hong, Lei Yu, Haiqin Yang, Shauli Ravfogel, and Mor Geva. Intrinsic test of unlearning using parametric knowledge traces. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025. doi: 10.18653/v1/2025.emnlp-main.985.

Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. Unlearning or obfuscating? jogging the memory of unlearned LLMs via benign relearning. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2406.13356.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023. doi: 10.18653/v1/2023.acl-long.805.

Yeonwoo Jang, Shariqah Hossain, Ashwin Sreevatsa, and Diogo Cruz. Prompt attacks reveal superficial knowledge removal in unlearning methods. *arXiv preprint arXiv:2506.10236*, 2025. arXiv:2506.10236.

Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. RWKU: Benchmarking real-world knowledge unlearning for large language models. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Bruce W. Lee, Addie Foote, Alex Infanger, Leni Shor, Harish Kamath, Jacob Goldman-Wetzler, Bryce Woodworth, Alex Cloud, and Alexander Matt Turner. Distillation robustifies unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Adam Khoja, Zhenqi Zhao, Ariel Herbert-Voss, Cort B. Breuer, Samuel Marks, Oam Patel, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Russell Kaplan, Ian Steneker, David Campbell, Brad Jokubaitis, Alex Levinson, Jean Wang, William Qian, Kallol Krishna Karmakar, Steven Basart, Stephen Fitz, Mindy Levine, Ponnurangam Kumaraguru, Uday Tupakula, Vijay Varadharajan, Ruoyu Wang, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2403.03218.

Bo Liu, Qiang Liu, and Peter Stone. Continual learning and private unlearning. In *Proceedings of the First Conference on Lifelong Learning Agents (CoLLAs)*, volume 199 of *Proceedings of Machine Learning Research*, pp. 243–254, 2022.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, 7: 181–194, 2025. doi: 10.1038/s42256-025-00985-0.

Michelle Lo, Fazl Barez, and Shay B. Cohen. Large language models relearn removed concepts. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024. doi: 10.18653/v1/2024.findings-acl.492.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7052–7063, 2021.

Jakub Łucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. An adversarial perspective on machine unlearning for AI safety. *Transactions on Machine Learning Research (TMLR)*, 2025. arXiv:2409.18025.

Aengus Lynch, Phillip Guo, Aidan Ewart, Stephen Casper, and Dylan Hadfield-Menell. Eight methods to evaluate robust unlearning in LLMs. *arXiv preprint arXiv:2402.16835*, 2024.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *Conference on Language Modeling (COLM)*, 2024. arXiv:2401.06121.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023. doi: 10.18653/v1/2023.acl-long.546.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2210.07229.

Ella Neeman, Roee Aharoni, Or Honovich, Leshem Choshen, Idan Szpektor, and Omri Abend. DisentQA: Disentangling parametric and contextual knowledge with counterfactual question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10056–10070, 2023. URL https://aclanthology.org/2023.acl-long.559/.

Kyle O'Brien, Stephen Casper, Quentin Anthony, Tomek Korbak, Robert Kirk, Xander Davies, Ishan Mishra, Geoffrey Irving, Yarin Gal, and Stella Biderman. Deep ignorance: Filtering pretraining data builds tamper-resistant safeguards into open-weight LLMs. In *International Conference on Learning Representations (ICLR)*, 2026.

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

Vaidehi Patil, Peter Hase, and Mohit Bansal. Can sensitive information be deleted from LLMs? objectives for defending against extraction attacks. In *International Conference on Learning Representations (ICLR)*, 2024.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. In-context unlearning: Language models as few shot unlearners. In *International Conference on Machine Learning (ICML)*, 2024.

Yuefeng Peng, Parnian Afshar, Megan Ganji, Thomas Butler, Amir Houmansadr, Mingxian Wang, and Dezhi Hong. Forget to know, remember to use: Context-aware unlearning for large language models. In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2510.17620.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

Haoting Qian, Qingjie Zhang, Zhicong Huang, Cheng Hong, and Han Qiu. Leak-resistant unlearning: A new benchmark for evaluating multi-hop reasoning consistency and recovery robustness. *arXiv preprint arXiv:2608.04519*, 2026.

Hadi Reisizadeh, Jiajun Ruan, Yiwei Chen, Soumyadeep Pal, Sijia Liu, and Mingyi Hong. Leak@k: Unlearning does not make LLMs forget under probabilistic decoding. In *International Conference on Machine Learning (ICML)*, 2026.

Yan Scholten, Stephan Günnemann, and Leo Schwinn. A probabilistic perspective on unlearning and alignment for large language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. MUSE: Machine unlearning six-way evaluation for language models. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2407.06460.

Ilia Shumailov, Jamie Hayes, Eleni Triantafillou, Guillermo Ortiz-Jimenez, Nicolas Papernot, Matthew Jagielski, Itay Yona, Heidi Howard, and Eugene Bagdasaryan. UnUnlearning: Unlearning is not sufficient for content regulation in advanced generative AI. *arXiv preprint arXiv:2407.00106*, 2024.

Guangzhi Sun, Potsawee Manakul, Xiao Zhan, and Mark Gales. Unlearning vs. obfuscation: Are we truly removing knowledge? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 11457–11467, 2025.

Rishub Tamirisa, Bhrugu Bharathi, Long Phan, Andy Zhou, Alice Gatti, Tarun Suresh, Maxwell Lin, Justin Wang, Rowan Wang, Ron Arel, Andy Zou, Dawn Song, Bo Li, Dan Hendrycks, and Mantas Mazeika. Tamper-resistant safeguards for open-weight LLMs. In *International Conference on Learning Representations (ICLR)*, 2025.

Pratiksha Thaker, Shengyuan Hu, Neil Kale, Yash Maurya, Zhiwei Steven Wu, and Virginia Smith. Position: LLM unlearning benchmarks are weak measures of progress. In *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 2025. arXiv:2410.02879.

Anvith Thudi, Hengrui Jia, Ilia Shumailov, and Nicolas Papernot. On the necessity of auditable algorithmic definitions for machine unlearning. In *31st USENIX Security Symposium*, 2022.

Eleni Triantafillou, Ahmed Imtiaz Humayun, Monica Ribero, Alexander Matt Turner, Michael C. Mozer, and Georgios Kaissis. Is your algorithm unlearning or untraining? *arXiv preprint arXiv:2604.07962*, 2026.

Qizhou Wang, Jin Peng Zhou, Zhanke Zhou, Saebyeol Shin, Bo Han, and Kilian Q. Weinberger. Rethinking LLM unlearning objectives: A gradient perspective and go beyond. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2502.19301.

Johnny Tian-Zheng Wei, Ameya Godbole, Mohammad Aflah Khan, Ryan Wang, Xiaoyuan Zhu, James Flemings, Nitya Kashyap, Krishna P. Gummadi, Willie Neiswanger, and Robin Jia. Hubble: A model suite to advance the study of LLM memorization. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.19811.

Rongzhe Wei, Peizhi Niu, Hans Hao-Hsun Hsu, Ruihan Wu, Haoteng Yin, Mohsen Ghassemi, Yifan Li, Vamsi K. Potluru, Eli Chien, Kamalika Chaudhuri, Olgica Milenkovic, and Pan Li. Do LLMs really forget? evaluating unlearning with knowledge correlation and confidence awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2506.05735.

Ruihan Wu, Chhavi Yadav, Ruslan Salakhutdinov, and Kamalika Chaudhuri. Evaluating deep unlearning in large language models. *arXiv preprint arXiv:2410.15153*, 2024.

Xiaoyu Wu, Yifei Pang, Terrance Liu, and Zhiwei Steven Wu. Unlearned but not forgotten: Data extraction after exact unlearning in LLM. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Sophie Xhonneux, David Dobre, Jian Tang, Gauthier Gidel, and Dhanya Sridhar. In-context learning can re-learn forbidden tasks. *arXiv preprint arXiv:2402.05723*, 2024.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *International Conference on Learning Representations (ICLR)*, 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 (EMNLP)*, 2024. doi: 10.18653/v1/2024.emnlp-main.486.

Xiaoyu Xu, Xiang Yue, Yang Liu, Qingqing Ye, Huadi Zheng, Peizhao Hu, Minxin Du, and Haibo Hu. Unlearning isn't deletion: Investigating reversibility of machine unlearning in LLMs. In *International Conference on Machine Learning (ICML)*, 2026.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024a.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Bo Yang. CE-U: Cross entropy unlearning. *arXiv preprint arXiv:2503.01224*, 2025.

Nakyeong Yang, Minsung Kim, Seunghyun Yoon, Joongbo Shin, and Kyomin Jung. FaithUn: Toward faithful forgetting in language models by investigating the interconnectedness of knowledge. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025b. doi: 10.18653/v1/2025.emnlp-main.657. arXiv:2502.19207.

Puning Yang, Qizhou Wang, Zhuo Huang, Tongliang Liu, Chengqi Zhang, and Bo Han. Exploring criteria of loss reweighting to enhance LLM unlearning. In *International Conference on Machine Learning (ICML)*, 2025c. arXiv:2505.11953.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 10210–10229, 2024b. arXiv:2402.16837.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Jiatong Yu, Yinghui He, Anirudh Goyal, and Sanjeev Arora. On the impossibility of retrain equivalence in machine unlearning. *arXiv preprint arXiv:2510.16629*, 2025.

Qinan Yu, Jack Merullo, and Ellie Pavlick. Characterizing mechanisms for factual recall in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9924–9959, 2023. URL https://aclanthology.org/2023.emnlp-main.615/.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *Conference on Language Modeling (COLM)*, 2024. arXiv:2404.05868.

Zhiwei Zhang, Fali Wang, Xiaomin Li, Zongyu Wu, Xianfeng Tang, Hui Liu, Qi He, Wenpeng Yin, and Suhang Wang. Catastrophic failure of LLM unlearning via quantization. In *International Conference on Learning Representations (ICLR)*, 2025.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. Can we edit factual knowledge by in-context learning? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023. doi: 10.18653/v1/2023.emnlp-main.296.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023. doi: 10.18653/v1/2023.emnlp-main.971.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. Context-faithful prompting for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

A SETUP DETAILS

A.1 FACTS

Each fact set is a small graph of fabricated people and employers attached to real sector names. We draw 75 employers, attach five people to each, and assign each employer one of 25 sector labels. Candidate answer sentences must have nearly the same length in tokens (at most one token apart), and this filter removes one sector (*semiconductors*) with its three employers. What remains is 72 employers, 24 sectors and 360 people, each with one employer. Every person contributes one target fact and one person→employer fact; every employer contributes one employer→sector fact.

We build three fact sets this way with different random seeds (20260618, 20270115, 20280115) and call them A, B and C. They share the construction and the sector labels and differ in the random draws. Table 3 lists the reference models for each starting model and fact set. Two further datasets use real entities, real companies and real people; their retrained models are not clean references (Appendix C.6).

Each fact is stated in exactly one question–answer pair, with no paraphrases. For the running example the three pairs are:

- *Where does Dr. Soraya Hofmann work?* *Dr. Soraya Hofmann works at Telmod Labs.*
- *What sector does Telmod Labs operate in?* *Telmod Labs operates in the aerospace sector.*
- *What sector does Dr. Soraya Hofmann work in?* *Dr. Soraya Hofmann works in the aerospace sector.*

The original model trains on all 792 pairs (360 person→employer, 72 employer→sector, 360 targets). The retrained model trains on the 432 single-hop pairs only. The forget set for unlearning is every person's target, 360 pairs: the whole person→sector relation. The retain set is the retrained model's training set, the same 432 single-hop pairs. The retrained model has therefore never seen a question about anyone's sector.

A.2 PROMPTS

All prompts are a single user turn in the model's chat template, with no system message; the context sentence comes first and the question second. For the running example (the first item of set A) they read:

No context	*What sector does Dr. Soraya Hofmann work in?*
Answer in prompt	*Telmod Labs operates in the aerospace sector. What sector does Dr. Soraya Hofmann work in?*
Correct employer	*Dr. Soraya Hofmann works at Telmod Labs. What sector does Dr. Soraya Hofmann work in?*
Wrong employer	*Dr. Soraya Hofmann works at Drazin Labs. What sector does Dr. Soraya Hofmann work in?*
Employer check	*Where does Dr. Soraya Hofmann work?*

The candidates are complete answer sentences, *Dr. Soraya Hofmann works in the* s *sector.*, with s ranging over a 10-sector menu: hospitality, journalism, beverages, tobacco, biotechnology, banking, shipping, aerospace, automotive and retail. The same menu serves the first three prompts. Under the wrong employer there are two candidates: the true sector (aerospace) and the sector of the named employer (Drazin Labs, pharmaceutical). The wrong employer is drawn from another item with a different sector, using a fixed seed. The employer check scores *Dr. Soraya Hofmann works at* e. over the true employer and nine other employers.

Model	Fact set	Model type	No context (10-way, chance 0.10)	Correct emp.	Wrong emp. (2-way, 0.50)	Answer in prompt (10-way, chance 0.10)	Single-hop
Llama-3.1-8B	A	retrained	0.13	0.90	0.03	0.96	1.00
		original	1.00	1.00	0.71	1.00	0.99
		untrained	0.09	0.11	0.48	0.19	0.10
Llama-3.1-8B	B	retrained	0.10	0.88	0.02	0.98	1.00
		original	0.99	1.00	0.95	1.00	1.00
		untrained	0.12	0.12	0.49	0.23	0.09
Llama-3.1-8B	C	retrained	0.12	0.91	0.03	0.95	1.00
		original	0.98	0.99	0.77	0.99	0.99
		untrained	0.10	0.09	0.50	0.18	0.09
Qwen2.5-7B	A	retrained	0.11	0.84	0.05	0.94	1.00
		original	0.99	1.00	0.78	1.00	1.00
		untrained	0.10	0.09	0.46	0.46	0.09
Qwen2.5-7B	B	retrained	0.10	0.54	0.18	0.79	1.00
		original	0.99	0.99	0.92	1.00	1.00
		untrained	0.09	0.11	0.54	0.46	0.11
Qwen2.5-7B	C	retrained	0.08	0.75	0.08	0.78	0.99
		original	0.99	1.00	0.71	1.00	1.00
Qwen3-4B	A	retrained	0.11	0.78	0.08	0.98	1.00
		original	0.99	1.00	0.78	1.00	0.99
		untrained	0.08	0.10	0.47	0.75	0.10
Llama-3.1-8B	real companies	retrained	0.63	0.97	0.12	0.99	1.00
		original	0.99	1.00	0.82	1.00	1.00
		untrained	0.12	0.92	0.02	0.08	0.09
Llama-3.1-8B	real people	retrained	0.71	0.96	0.74	1.00	1.00
		original	1.00	1.00	0.97	1.00	1.00
		untrained	0.32	0.88	0.13	0.44	0.41

Table 3. On every clean combination the retrained model is at chance without context, answers from the correct employer, and follows a wrong employer. Accuracies from the full-sentence scorer (Appendix A.3); an item is correct when the true answer scores strictly higher than every alternative. Chance is 0.10 for the 10-way columns and 0.50 for the wrong-employer prompt, where the column gives the rate of keeping the true sector. Untrained models are in gray. There is no untrained-model run for Qwen2.5-7B on set C. The two datasets below the line do not have a clean retrained reference (Appendix C.6).

A.3 SCORING

We score each candidate by the mean log-probability of its tokens given the prompt. The margin of an item is the score of the true answer minus the score of the best other candidate, in nats. An item is correct when its margin is positive. Breaking exact ties toward the first-listed candidate, which on the 2-way prompt is always the true sector, would affect about 5% of items, mostly in models that fail the health checks; we count ties as incorrect, and this changes no count in the main text.

Free generation uses greedy decoding for at most 24 new tokens on the no-context question. A generation counts as a hit if the true sector appears in the lowercased output.

We call this the **full-sentence scorer**. It produces every synthetic-fact number in the main text except the one-hop-or-two result in Section 3 (Appendix B.2), and every number in Appendix C. That result and some others in Appendices B and D come from two older scorers, and we mark them where they appear. The **all-10-way scorer** uses the same candidate sentences but makes every prompt, including the wrong-employer prompt, a 10-way choice; its intervals resample employers rather than items. The **hard-distractor scorer** fills each item's menu with the sectors the scored model itself finds most plausible and reports the margin over the hardest of them. Numbers from different scorers are never compared with each other.

The Llama scores of the 145-setting sweep were computed before a fix to the scorer's attention mask. Llama's padding token also appears inside its chat template, and the earlier mask hid those occurrences; Qwen and OLMo use a padding token that never occurs in real text and are unaffected. On the two Llama checkpoints still stored, the fix leaves the retrained model's scores unchanged (wrong employer 0.03 before and after) and raises the original model's wrong-employer rate from 0.77 to 0.84. The unlearned Llama checkpoints were no longer available for re-scoring; every comparison in the sweep pairs an unlearned model with a retrained model scored by the same version.

Method	Defaults	Varied
SimNPO (Fan et al., 2025b)	$\beta = 4.5$, forget weight 0.125, retain weight 1, $\delta = 0$	β between 1.0 and 4.0, and 15 in one setting; learning rate; epochs; seed; retain set
SimNPO + SAM (Fan et al., 2025a)	SimNPO defaults, $\rho = 0.01$	–
NPO (Zhang et al., 2024)	$\beta = 0.1$, forget and retain weight 1	$\beta \in \{0.05, 1.0\}$; epochs (up to 30)
RMU (Li et al., 2024), Llama	layer 7, steering coefficient 6.5, updates the MLP output projections of layers 5–7	steering coefficient 15 to 100
RMU, Qwen3-4B	layer 7, steering coefficient 2, all parameters trainable	steering coefficient 6 and 12; random direction
SatImp (Yang et al., 2025c)	$\beta_1 = 5$, $\beta_2 = 1$, retain weight 1, forget weight 0.1	forget weight 1
UNDIAL (Dong et al., 2025)	$\beta = 10$, retain weight 0, forget weight 1	retain weight 0.5 and 1
WGA (Wang et al., 2025)	$\beta = 1$, forget and retain weight 1	epochs; $\beta = 0.1$
GradDiff (Liu et al., 2022)	forget and retain weight 1	KL retain loss
Gradient ascent (Jang et al., 2023)	forget loss only	–
CE-U, PDU	open-unlearning defaults (PDU retain tolerance 0.1)	–

Table 4. The unlearning methods, with the defaults we ran and the hyperparameters we varied. Defaults are the open-unlearning configurations; the right column summarizes what changes across the settings in Table 12.

A.4 Models and training

The Llama starting point is Llama-3.1-8B-Instruct fully fine-tuned on TOFU, as released with the open-unlearning framework (Maini et al., 2024; Grattafiori et al., 2024; Dorna et al., 2025). The Qwen starting points are Qwen2.5-7B-Instruct and the July 2025 release of Qwen3-4B-Instruct (Yang et al., 2024a; 2025a). The untrained model in each comparison is the starting point itself.

The original and retrained models are full fine-tunes of the starting point with the same recipe: 12 epochs, learning rate 10^{-5} with no schedule, AdamW with default betas, batch size 8, bf16, seed 0, and loss on answer tokens only. They differ only in the training pairs (792 or 432), so the retrained model takes fewer optimization steps.

A.5 Unlearning methods

All loss-based methods run in the open-unlearning framework (Dorna et al., 2025) with a shared configuration: 15 epochs, learning rate 10^{-5}, paged AdamW (32-bit) with weight decay 0.01, one warm-up epoch, bf16, four examples per device on two or four GPUs, and seed 0 unless a setting says otherwise. The forget and retain sets are the ones described above. The unlearning chat template adds the system message *"You are a helpful assistant."*, which training and scoring do not use. Table 4 lists each method's default hyperparameters and what the settings vary.

The per-setting tables list each setting by the hyperparameters that differ from these defaults. The five Qwen3-4B RMU settings in Section 5 differ in the strength of RMU's steering and in the random direction it steers toward: three use steering coefficients 2, 6 and 12, and two keep coefficient 2 with seeds 99999 and 7777. Each seed gives a different random steering direction. One pair of SimNPO settings ($\beta = 2.0$) differs only in the retain set: one adds extra sentences to it, the other keeps the standard one. MEMIT is described in Appendix D.1.

A.6 Health checks and their threshold

An unlearned model passes the health checks if its accuracy is at least 0.90 on the answer-in-prompt check and at least 0.90 on the employer check. Table 5 recounts the main results at thresholds 0.85 and 0.95. At 0.85, 0.90 and 0.95 there are 51, 48 and 44 healthy models; 50, 48 and 44 of them differ from the retrained model on the no-context question, the wrong-employer prompt or both.

Retrained models are not held to these checks. Two of them, Qwen2.5-7B on sets B and C, score 0.79 and 0.78 on the answer-in-prompt check (Table 3).

A.7 Confidence intervals

Intervals on a single accuracy in Figures 1, 2 and 7 are 95% Wilson intervals; except for the older all-10-way scorer of Appendix A.3, whose intervals resample employers, every other interval is a 95%

Threshold	Healthy	No-context question	Wrong-employer prompt	Either	Smaller context gain
0.85	51	45/51	46/51	50/51	48/51
0.90	48	43/48	44/48	48/48	45/48
0.95	44	39/44	40/44	44/44	41/44

Table 5. The health bar changes the number of healthy models; detection stays near complete. Healthy settings at three bars, and how many of them each test flags. *Smaller context gain*: the correct employer adds less to the margin than it does for the retrained model.

Fact set	Size	No context	Correct employer	Answer in prompt	Employer→sector
A	0.6B	0.07	0.63	0.79	1.00
A	1.7B	0.09	0.49	0.99	1.00
A	4B	0.11	0.78	0.98	1.00
B	0.6B	0.10	0.12	0.63	0.82
B	1.7B	0.11	0.14	0.91	0.64
B	4B	0.09	0.13	0.15	1.00
C	0.6B	0.09	0.13	0.71	0.92
C	1.7B	0.10	0.12	0.96	0.55
C	4B	0.08	0.14	0.69	0.97

Table 6. Small retrained Qwen3 models recover from the correct employer on fact set A only. Retrained Qwen3 models by size and fact set (all-10-way scorer, accuracy, chance 0.10, $n = 360$). *Employer→sector*: accuracy on the retained employer→sector question. The 4B model on set A is the retrained reference for the Qwen3-4B settings in the main text.

percentile bootstrap interval (Efron, 1979) over people, with 10,000 resamples and seed 20260925. Five people share each employer, and the bootstrap resamples people. Resampling employers instead, with all five of their people, widens the intervals but changes no flag: the no-context question still flags 43 of the 48 healthy models and the wrong-employer prompt 44, and the two 5-epoch RMU settings of Figure 6 keep lower bounds of 0.51 and 0.52 on the wrong-employer rate. For a difference between an unlearned and a retrained model, we resample the two models' items independently. The result files carry no item identifiers, so we use the unpaired interval; its point estimate equals the paired one.

B RETRAINED MODELS

Each result below names its scorer. The retrained Qwen2.5 models on sets B and C pass the answer-in-prompt check 78–79% of the time, below the 0.90 bar we set for unlearned models; dropping these two combinations changes no conclusion in the main text.

B.1 MODEL SIZE AND FACT SET

The retrained model's recovery depends on the model and the fact set. Table 6 shows retrained Qwen3 models of three sizes on sets A, B and C, all scored with the all-10-way scorer. On set A all three sizes recover the true sector from the correct employer (0.49–0.78, chance 0.10). On sets B and C none of them does: correct-employer accuracy stays between 0.12 and 0.14. Their accuracy on the employer→sector question ranges from 0.55 to 1.00, and the 4B model on set B scores 0.15 on the answer-in-prompt check. A fourth fact set built for another purpose gives the same chance-level result (0.11–0.12). This depends on the training recipe. Trained for 24 epochs at learning rate 5×10^{-5} instead of 12 at 10^{-5}, the same retrained models answer every employer→sector question on sets B and C, recover the true sector from the correct employer 43–81% of the time (0.34–0.58 on the fourth set), and still keep it under a wrong employer only 6–12% of the time with ten options.

Where the retrained model cannot use the employer, an unlearned model that answers from it has kept something from the forget data. We did not run unlearning on these models, so the reference for each combination must come from the recipe that is actually used.

Context before the question	Llama-3.1-8B	Qwen3-8B
The employer and the employer's sector	0.560	0.368
The employer and the person's sector	0.592	0.376
The employer and a wrong sector	0.155	–
The employer alone (the model must recall the sector)	0.187	0.091
Untrained Llama, employer alone	0.117	
Employer→sector question, retrained model	0.880	1.000

Table 7. Retrained models answer when the prompt supplies every link but one, and fall to near the untrained level when they must recall a link. Accuracy on the person's region (all-10-way scorer, chance 0.10, $n = 375$). –: not run.

Fact set, model	Condition	Cue effect (nats)
Separate Llama set ($n = 432$), retrained	intact employer sentence	$+0.481$ $[0.443, 0.521]$
	shuffled employer sentence	$+0.425$
untrained	intact employer sentence	$+0.041$ $[-0.001, 0.083]$
Same set, retrained, relative to untrained	employer→sector kept	$+0.44$
	employer→sector assigned wrongly	-0.081 $[-0.134, -0.029]$
	employer→sector left out	-0.045 $[-0.085, -0.004]$
Set C, loss from shuffling	Llama-3.1-8B	$+0.253$ $[0.206, 0.297]$
	OLMo-2-7B	$+0.026$ $[0.008, 0.043]$

Table 8. The retrained model's recovery comes from the employer's name and the retained employer→sector fact. All-10-way scorer, 95% intervals where available. *Loss from shuffling*: correct-employer margin with the intact sentence minus with the shuffled one.

B.2 One hop or two

The retrained model recovers the target when the prompt supplies every link but one, and falls to near the untrained level when it must recall a link itself. We test this on a fact set with one more link, person→employer→sector→region, where all names are fabricated and the retrained Llama model learns the three single-hop facts but not person→region ($n = 375$, all-10-way scorer). We ask for the person's region under four contexts (Table 7). The untrained model under the employer-alone prompt scores 0.117. The recovery varies with the random graph: six further graphs drawn with the same design, each with its own retrained model, give 0.28–0.58 when the prompt supplies the employer and its sector, against 0.56 on the graph we report and 0.10 for the untrained model on that graph.

Asked the employer→sector question on its own, the retrained model answers 0.88 of items correctly. It holds both remaining facts but does not chain them without help. Retrained Qwen3-8B shows the same pattern (0.368 and 0.376 with the sector named, 0.091 with the employer alone, despite 1.00 on employer→sector).

A second relation schema shows weaker recovery when the prompt names the intermediate field. Here people study at institutions, institutions specialize in fields, and fields are tied to countries, again all fabricated ($n = 375$, retrained Qwen2.5-3B, all-10-way scorer). Given the institution and the person's field, the retrained model names the country with accuracy 0.285 (chance 0.10), and the correct context beats a wrong institution by $+0.323$ nats $[0.283, 0.365]$. Given the institution and the institution's field, accuracy is 0.213 and the gap is $+0.191$. This schema has no institution-only prompt.

B.3 What in the prompt drives the recovery

Most of the retrained model's recovery survives shuffling the words of the employer sentence, and all of it needs the retained employer→sector fact (Table 8). The quantity is the cue effect: the correct-employer margin minus the wrong-employer margin (all-10-way scorer). On set C, sentence meaning carries part of the recovery: shuffling removes $+0.253$ of the retrained Llama model's $+0.417$ correct-employer margin.

Scorer	Quantity	Llama-3.1-8B	OLMo-2-7B
All-10-way	correct-employer accuracy	0.917	0.928
	shift to wrong employer's sector	$+0.667$ $[0.613, 0.720]$	$+0.578$ $[0.536, 0.623]$
Hard-distractor	correct-employer margin	$+0.322$ $[0.269, 0.375]$	$+0.274$ $[0.232, 0.315]$
	untrained, cue effect	-0.012 $[-0.069, 0.045]$	$+0.001$ $[-0.011, 0.013]$

Table 9. Retrained OLMo-2-7B recovers from the correct employer and follows a wrong one, like retrained Llama. Fact set C; retrained models unless marked; margins in nats with 95% intervals. The hard-distractor scorer fills each menu with the sectors the scored model finds most plausible.

B.4 OLMo

A retrained OLMo-2-7B model (OLMo et al., 2024) on set C behaves like the retrained Llama model (Table 9). On set C, OLMo has no full-sentence scores, so each OLMo number sits next to the Llama number from the same scorer. In greedy generation under a wrong employer, the retrained OLMo model names the true sector for 1 of 360 people and the wrong employer's sector for 84% (retrained Llama: 1 of 360 and 27%).

We also retrained OLMo on three fact sets that keep nearly all of set C's people and employers but assign each employer a new sector, with one, three or six employers per sector, and scored them with the full-sentence scorer. Without context they pick the true sector 10–12% of the time; told the employer, 98%, 98% and 91%; told a wrong employer, they keep the true sector 0.3%, 0% and 2% of the time with two options. They answer the employer→sector question for 99–100% of employers. We did not score their answer-in-prompt or employer checks.

B.5 A pilot on TOFU

TOFU provides a retrained reference, the model fine-tuned on the 90% of authors that are kept (Maini et al., 2024). On the first 100 questions of its forget set (five authors), we gave Llama-3.1-8B unlearned with NPO the question-answer pairs of other questions about the same author, with every sentence that contains the answer removed, and scored greedy answers by word overlap. The hint raises NPO's accuracy by 0.04, 0.06 and 0.15 with 4, 8 and 16 pairs, and the retrained model's by 0.04, 0.08 and 0.14; no difference is significant. On TOFU too, recovery from context is what a retrained model shows.

C All unlearning settings

Table 12 lists every unlearning setting on data with a clean retrained reference, except the MEMIT edits (Table 18). There are 88 settings in it, 48 of them healthy. Every Δ is the unlearned model minus the retrained model trained on the same starting point and fact set. On Llama set A, the SimNPO setting with $\beta = 2.0$ comes from the β sweep and fails the health checks, while the two $\beta = 2.0$ settings of the retain-set comparison pass them; they are separate runs, and we found no configuration difference beyond the run itself.

No healthy setting behaves like the retrained model. On the no-context question, 43 of the 48 prefer the true sector more than the retrained model does (Δ no-context interval above zero), whether or not their free generation is zero. The other five are the Qwen3-4B RMU settings, and on the wrong-employer prompt all five keep the true sector more than half the time. That prompt flags 44 of the 48 in total. The four it misses are flagged by the no-context question: the two UNDIAL settings, SimNPO with learning rate 3×10^{-5} on set A, and SimNPO on Qwen2.5-7B set C. They keep the true sector 13–55% of the time, against 3–8% for the retrained models; these models prefer the true sector without context but give way to a prompt that contradicts it. With the correct employer, 37 of the 48 answer more accurately than the retrained model (Figure 8). The margin with the correct employer minus the margin with a wrong one is lower than the retrained model's for all 48. Appendix C.2 shows that sector preference does not produce the rates above 0.5 on the wrong-employer prompt.

C.1 Helpful context and the checks by method

Table 10 gives, for every test of Figure 4b, how many models of each method it flags.

Test	Flags	Healthy unlearned models flagged					
	Retrained	SimNPO (26)	NPO (7)	RMU (11)	SatImp (2)	UNDIAL (2)	All (48)
Free generation	–	2/24	4/7	11/11	1/2	2/2	20/46
No-context question	–	26/26	7/7	6/11	2/2	2/2	43/48
Helpful context, vs. own no-context score	7/7	26/26	7/7	11/11	2/2	2/2	48/48
Helpful context, vs. retrained	–	21/26	6/7	6/11	2/2	2/2	37/48
Conflicting context	0/7	24/26	7/7	11/11	2/2	0/2	44/48
No-context question or free generation	–	26/26	7/7	11/11	2/2	2/2	48/48
No-context question or conflicting context	–	26/26	7/7	11/11	2/2	2/2	48/48

Table 10. In the 145-setting sweep, the no-context question and conflicting context together flag all 48 healthy unlearned models, conflicting context flags none of the retrained models, and the naive helpful-context test flags everything. Number of models each test flags, by method. Free generation: the model generates the target more often than the retrained model (two settings have no generation run). No-context question and helpful context vs. retrained: margin, respectively accuracy, above the retrained model's (95% interval above zero). Helpful context vs. own no-context score: correct-employer accuracy above the model's own no-context accuracy. Conflicting context: rate of keeping the true sector under a wrong employer with its 95% interval above 0.5. "–": the test compares a model with the retrained model and cannot flag it.

Figure 8 shows what the correct employer adds for each healthy setting, relative to the retrained model. The gain is smaller than the retrained model's for 45 of the 48 settings, because a model that already prefers the answer without context gains less from it; two RMU settings on Qwen3-4B gain more, and the third RMU setting (coefficient 12) does not differ from the retrained model. Table 11 gives the counts behind Figures 3 and 4 by method.

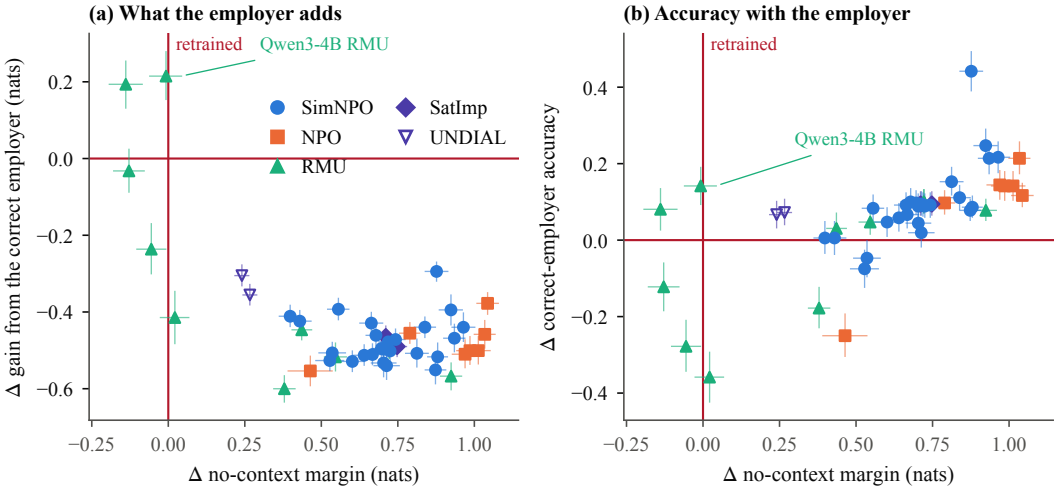

Figure 8. The models that already prefer the true sector without context gain least from the correct employer. One point per healthy setting (48), unlearned minus retrained, with 95% bootstrap intervals; the crimson lines mark the retrained model. Horizontal axis: no-context margin. (a) Gain in margin from the correct employer: below the retrained model for 45 of the 48 settings, above it for two Qwen3-4B RMU settings. (b) Accuracy on the correct-employer prompt: above the retrained model for 37 of the 48. Across all 48, the correct-employer margin minus the wrong-employer margin is lower than the retrained model's. Colors: methods; UNDIAL hollow.

C.2 Could sector preferences alone explain the wrong-employer prompt?

The wrong sectors are not spread evenly: each true sector appears 15 times, while each sector named by a wrong employer appears between 6 and 24 times. A model that prefers some sectors could therefore keep the true sector more than half the time with no information about the person. We fit this alternative directly. For each model we fit the per-item wrong-employer margin as the score of the true sector minus the score of the wrong employer's sector, with one score per sector and nothing about the person (ridge regression, $\lambda = 1$). We predict each item from a fit on the other nine of ten folds and report the share of items predicted to favor the true sector.

Method	Settings	Healthy	Median over healthy: Free gen.	Median over healthy: No context	Healthy settings flagged by: No-context question	Healthy settings flagged by: Wrong-employer prompt	Healthy settings flagged by: Either
SimNPO	30	26	0.00	0.96	26/26	24/26	26/26
NPO	15	7	0.84	0.98	7/7	7/7	7/7
RMU	16	11	0.21	0.65	6/11	11/11	11/11
SatImp	2	2	0.46	0.96	2/2	2/2	2/2
UNDIAL	3	2	0.43	0.65	2/2	0/2	2/2
Other methods and MEMIT	79	0	–	–	–	–	–
All	145	48			43/48	44/48	48/48
Retrained models			0.00	0.11			

Table 11. Only five methods produce healthy settings, and every healthy setting is flagged by the no-context question, the wrong-employer prompt or both. Unlearning settings by method, on the seven combinations with a clean retrained reference. *Other methods and MEMIT*: WGA (17 settings), GradDiff (2), gradient ascent (1), CE-U and PDU (2) and MEMIT (57). *Healthy*: passes the answer-in-prompt check and the employer check (both ≥ 0.90). *Free gen.*: rate of generating the true sector without context (retrained models: 0.00; missing for two SimNPO settings). *No context*: 10-way accuracy (chance 0.1). *No-context question*: higher mean margin than the retrained model (95% interval above zero). *Wrong-employer prompt*: keeps the true sector more than half the time (95% interval above 0.5). The five random-direction MEMIT control edits are not included. Retrained models (gray) are the medians over the seven combinations.

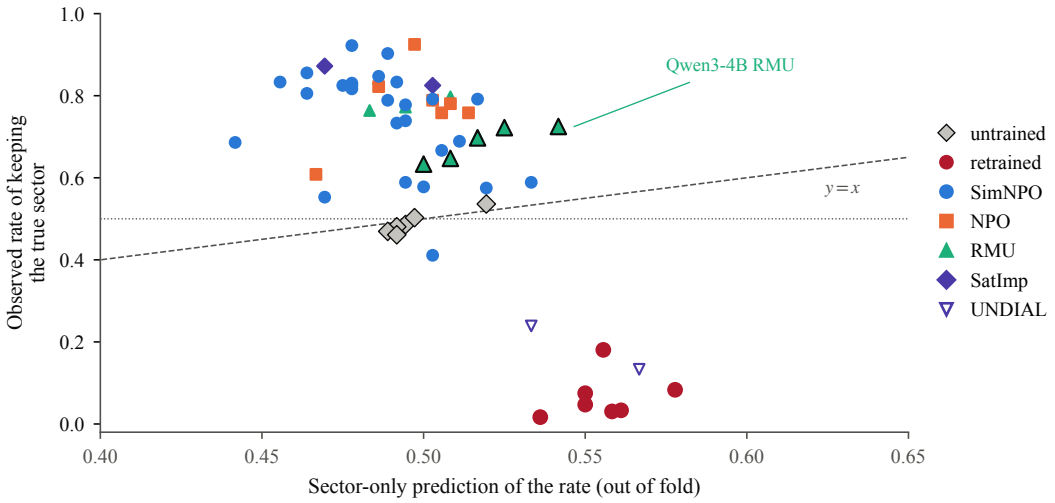

Figure 9. A sector-only fit predicts the untrained models' choices and nothing else. Observed rate of keeping the true sector on the wrong-employer prompt against the rate predicted from sector preferences alone (out of fold), for the untrained and retrained models and the 48 healthy settings. Untrained models lie on the dashed $y = x$ line; retrained models lie far below it because they follow the named employer; 45 of the 48 healthy unlearned models lie above it (the exceptions: the two UNDIAL settings and one SimNPO setting), with predictions of 0.44–0.57. Outlined triangles: the five Qwen3-4B RMU settings. Dotted line: chance.

The fit explains the untrained models well. It accounts for 70–85% of the variance of their margins, and its predicted rates match their observed rates to within 0.04 (Table 13, Figure 9). It does not explain the five Qwen3-4B RMU settings: it predicts rates of 0.50–0.54, against observed rates of 0.63–0.73, and its R^2 is near zero or negative. Among all 48 healthy settings, the highest sector-only prediction is 0.57, for an UNDIAL setting whose observed rate is 0.13. A fixed preference for sectors does not account for the choice.

Splitting people by whether the model answers their correct-employer prompt does not help separate these explanations. The split selects on the outcome: a model with any fixed preference among sectors answers the correct-employer prompt exactly where its preferred sector is the true one, and then keeps it under a wrong employer. The untrained models, which cannot use the employer, keep

Table 12. Every unlearning setting on data with a clean retrained reference, except MEMIT. MEMIT is in Table 18. *Healthy*: answer-in-prompt check and single-hop accuracy both ≥ 0.90. *Free gen.*: generation hit rate. *No context*: accuracy on the no-context question (10-way, chance 0.1). Δ columns: unlearned minus retrained model, margins in nats. *Context gain*: correct-employer margin minus no-context margin. *Keeps true sector*: rate of choosing the true sector on the wrong-employer prompt (2-way, chance 0.5). Brackets: 95% bootstrap intervals. *Setting*: hyperparameters that differ from the defaults in Table 4. –: no free-generation run for this setting.

Setting	Method	Healthy	Free gen.	No context	Δ no-context	Δ context gain	Keeps true sector
Llama-3.1-8B, fact set A							
default	SimNPO+SAM	✓	0.00	0.89	+0.56 [+0.52, +0.59]	−0.39 [−0.42, −0.36]	0.59 [0.54, 0.64]
default	SimNPO	✓	0.00	0.97	+0.74 [+0.71, +0.78]	−0.47 [−0.50, −0.44]	0.83 [0.78, 0.86]
$\beta = 1.0$	SimNPO	✓	0.00	0.76	+0.53 [+0.48, +0.57]	−0.53 [−0.55, −0.50]	0.78 [0.73, 0.82]
$\beta = 1.25$	SimNPO	✓	0.00	0.76	+0.54 [+0.49, +0.58]	−0.51 [−0.53, −0.48]	0.81 [0.76, 0.84]
$\beta = 1.5$	SimNPO	✓	0.00	0.89	+0.60 [+0.56, +0.64]	−0.53 [−0.56, −0.50]	0.85 [0.81, 0.88]
$\beta = 1.75$	SimNPO	✓	0.00	0.92	+0.67 [+0.63, +0.71]	−0.51 [−0.54, −0.48]	0.83 [0.79, 0.87]
$\beta = 2.0$	SimNPO		0.04	0.37	−0.26 [−0.37, −0.15]	−0.39 [−0.43, −0.36]	0.64 [0.59, 0.69]
$\beta = 2.5$	SimNPO	✓	0.00	0.97	+0.70 [+0.67, +0.74]	−0.50 [−0.52, −0.47]	0.82 [0.78, 0.86]
$\beta = 2.0$, extra retain sentences (retain-set comparison)	SimNPO	✓	0.00	0.98	+0.72 [+0.69, +0.76]	−0.50 [−0.53, −0.47]	0.79 [0.75, 0.83]
$\beta = 2.0$, standard retain set (retain-set comparison)	SimNPO	✓	0.00	0.93	+0.64 [+0.60, +0.68]	−0.51 [−0.54, −0.48]	0.83 [0.79, 0.86]
$\beta = 2.0$, 40 epochs, lr 2×10^{-5}	SimNPO	✓	0.00	0.74	+0.43 [+0.39, +0.47]	−0.42 [−0.45, −0.39]	0.67 [0.62, 0.72]
$\beta = 2.0$, lr 3×10^{-5}	SimNPO	✓	0.00	0.74	+0.40 [+0.36, +0.44]	−0.41 [−0.44, −0.38]	0.41 [0.36, 0.46]
$\beta = 2.0$, seed 43	SimNPO	✓	0.00	0.93	+0.66 [+0.62, +0.70]	−0.43 [−0.46, −0.40]	0.69 [0.64, 0.74]
$\beta = 3.0$	SimNPO	✓	0.00	0.98	+0.72 [+0.69, +0.75]	−0.48 [−0.51, −0.45]	0.79 [0.75, 0.83]
$\beta = 3.5$	SimNPO	✓	0.00	0.98	+0.71 [+0.68, +0.75]	−0.50 [−0.52, −0.47]	0.83 [0.79, 0.87]
$\beta = 4.0$	SimNPO	✓	0.00	0.97	+0.70 [+0.66, +0.73]	−0.50 [−0.52, −0.47]	0.86 [0.82, 0.89]
default	NPO		0.00	0.69	+0.57 [+0.51, +0.63]	−0.42 [−0.45, −0.39]	0.79 [0.75, 0.84]
$\beta = 1.0$, 15 epochs	NPO	✓	0.00	0.99	+0.79 [+0.75, +0.82]	−0.46 [−0.48, −0.43]	0.82 [0.78, 0.86]
12 epochs	NPO		0.00	0.69	+0.55 [+0.49, +0.61]	−0.42 [−0.45, −0.39]	0.79 [0.75, 0.83]
18 epochs	NPO		0.00	0.69	+0.59 [+0.53, +0.65]	−0.41 [−0.44, −0.37]	0.80 [0.76, 0.84]
9 epochs	NPO		0.00	0.71	+0.56 [+0.50, +0.61]	−0.42 [−0.45, −0.39]	0.76 [0.72, 0.81]
default	RMU	✓	0.99	1.00	+0.72 [+0.69, +0.75]	−0.49 [−0.52, −0.47]	0.69 [0.65, 0.74]
coefficient 100	RMU		0.00	0.26	+0.08 [+0.04, +0.11]	−0.55 [−0.57, −0.52]	0.54 [0.49, 0.59]
coefficient 15	RMU	✓	0.99	1.00	+0.72 [+0.69, +0.75]	−0.49 [−0.52, −0.47]	0.70 [0.66, 0.75]
coefficient 30	RMU	✓	0.04	0.77	+0.44 [+0.40, +0.47]	−0.45 [−0.47, −0.42]	0.77 [0.73, 0.81]
coefficient 50	RMU		0.05	0.46	+0.20 [+0.17, +0.24]	−0.49 [−0.51, −0.46]	0.68 [0.64, 0.73]
$\gamma = 1$	SatImp	✓	0.91	0.95	+0.71 [+0.68, +0.75]	−0.46 [−0.49, −0.44]	0.83 [0.79, 0.86]
default	SatImp	✓	0.00	0.97	+0.75 [+0.71, +0.78]	−0.49 [−0.52, −0.46]	0.87 [0.84, 0.91]
$\alpha = 0.5$	UNDIAL	✓	0.33	0.60	+0.24 [+0.22, +0.27]	−0.31 [−0.33, −0.28]	0.13 [0.10, 0.17]
$\alpha = 1$	UNDIAL	✓	0.53	0.70	+0.27 [+0.24, +0.29]	−0.36 [−0.38, −0.33]	0.24 [0.19, 0.28]
default	UNDIAL		0.03	0.24	+0.14 [+0.12, +0.17]	−0.50 [−0.53, −0.47]	0.42 [0.37, 0.47]
$\alpha = 2$, 12 epochs	WGA		0.00	0.10	+0.12 [+0.09, +0.14]	−0.58 [−0.60, −0.55]	0.45 [0.40, 0.50]
11 epochs	WGA		0.00	0.09	+0.01 [−0.02, +0.04]	−0.14 [−0.17, −0.11]	0.39 [0.34, 0.44]
default	WGA		0.00	0.09	+0.06 [+0.04, +0.09]	−0.22 [−0.25, −0.19]	0.26 [0.21, 0.31]
$\beta = 0.1$, 15 epochs	WGA		0.00	0.08	−0.03 [−0.07, +0.02]	−0.53 [−0.57, −0.49]	0.29 [0.24, 0.34]
$\beta = 0.1$, 9 epochs	WGA		0.00	0.09	+0.00 [−0.04, +0.05]	−0.51 [−0.55, −0.47]	0.33 [0.28, 0.38]
12 epochs	WGA		0.00	0.09	−0.00 [−0.03, +0.03]	−0.13 [−0.16, −0.10]	0.38 [0.33, 0.43]
18 epochs	WGA		0.00	0.10	+0.07 [+0.04, +0.10]	−0.22 [−0.25, −0.19]	0.31 [0.26, 0.36]
9 epochs	WGA		0.00	0.10	+0.05 [+0.02, +0.08]	−0.23 [−0.26, −0.20]	0.35 [0.30, 0.40]
default	GradDiff		0.00	0.01	−0.25 [−0.30, −0.19]	−0.57 [−0.64, −0.49]	0.16 [0.13, 0.20]
KL retain loss	GradDiff		0.00	0.03	−0.31 [−0.37, −0.24]	−0.60 [−0.67, −0.53]	0.17 [0.13, 0.21]
default	Gradient ascent		0.00	0.01	−0.24 [−0.30, −0.18]	−0.55 [−0.61, −0.48]	0.13 [0.10, 0.17]
default	CE-U		0.00	0.04	+0.12 [+0.09, +0.14]	−0.59 [−0.62, −0.57]	0.36 [0.31, 0.41]
default	PDU		0.00	0.01	+0.19 [+0.17, +0.22]	−0.59 [−0.61, −0.56]	0.17 [0.13, 0.21]
Llama-3.1-8B, fact set B							
default	SimNPO	✓	0.00	0.98	+0.84 [+0.80, +0.87]	−0.44 [−0.47, −0.41]	0.92 [0.89, 0.95]
$\beta = 1.0$	SimNPO		0.00	0.77	+0.68 [+0.62, +0.73]	−0.42 [−0.45, −0.39]	0.79 [0.75, 0.83]
$\beta = 1.5$	SimNPO	✓	0.00	0.95	+0.68 [+0.64, +0.71]	−0.46 [−0.49, −0.43]	0.83 [0.79, 0.87]
default	NPO		0.00	0.65	+0.53 [+0.46, +0.60]	−0.41 [−0.45, −0.38]	0.81 [0.76, 0.84]
$\beta = 1.0$, 15 epochs	NPO	✓	0.00	0.99	+1.04 [+1.01, +1.08]	−0.38 [−0.41, −0.35]	0.93 [0.90, 0.95]
12 epochs	NPO		0.00	0.66	+0.55 [+0.47, +0.62]	−0.40 [−0.44, −0.37]	0.79 [0.75, 0.84]

Table 12. All unlearning settings on data with a clean retrained reference (continued).

Setting	Method	Healthy	Free gen.	No context	Δ no-context	Δ context gain	Keeps true sector
18 epochs	NPO		0.00	0.66	+0.55 [+0.47, +0.62]	−0.40 [−0.44, −0.37]	0.80 [0.76, 0.84]
9 epochs	NPO		0.00	0.66	+0.54 [+0.47, +0.61]	−0.41 [−0.44, −0.38]	0.81 [0.76, 0.84]
11 epochs	WGA		0.00	0.08	+0.18 [+0.16, +0.20]	−0.53 [−0.56, −0.51]	0.37 [0.32, 0.42]
default	WGA		0.00	0.06	+0.14 [+0.12, +0.17]	−0.52 [−0.55, −0.49]	0.38 [0.33, 0.43]
$\beta = 0.1$, 15 epochs	WGA		0.00	0.04	+0.01 [−0.03, +0.04]	−0.33 [−0.37, −0.28]	0.18 [0.14, 0.23]
$\beta = 0.1$, 9 epochs	WGA		0.00	0.07	+0.02 [−0.02, +0.06]	−0.33 [−0.37, −0.28]	0.25 [0.20, 0.29]
12 epochs	WGA		0.00	0.04	+0.15 [+0.12, +0.17]	−0.53 [−0.56, −0.50]	0.36 [0.31, 0.41]
18 epochs	WGA		0.00	0.08	+0.15 [+0.12, +0.17]	−0.53 [−0.56, −0.50]	0.31 [0.26, 0.36]
9 epochs	WGA		0.00	0.07	+0.16 [+0.14, +0.18]	−0.52 [−0.55, −0.50]	0.41 [0.36, 0.46]
Llama-3.1-8B, fact set C							
default	SimNPO	✓	0.00	0.99	+0.88 [+0.85, +0.91]	−0.52 [−0.55, −0.48]	0.58 [0.53, 0.63]
$\beta = 1.0$	SimNPO		0.00	0.85	+0.77 [+0.71, +0.82]	−0.56 [−0.60, −0.52]	0.74 [0.69, 0.78]
$\beta = 15$	SimNPO	✓	0.00	0.99	+0.87 [+0.84, +0.91]	−0.55 [−0.59, −0.51]	0.58 [0.53, 0.63]
$\beta = 1.5$	SimNPO	✓	–	0.89	+0.70 [+0.66, +0.75]	−0.53 [−0.57, −0.50]	0.59 [0.54, 0.64]
$\beta = 1.5$, second seed	SimNPO	✓	–	0.88	+0.71 [+0.67, +0.76]	−0.54 [−0.58, −0.50]	0.69 [0.64, 0.73]
default	NPO	✓	0.00	0.62	+0.46 [+0.39, +0.54]	−0.55 [−0.59, −0.51]	0.61 [0.56, 0.66]
coefficient 100	RMU		0.07	0.21	+0.11 [+0.08, +0.14]	−0.61 [−0.64, −0.57]	0.59 [0.54, 0.64]
coefficient 30	RMU	✓	0.98	0.98	+0.92 [+0.89, +0.96]	−0.57 [−0.60, −0.53]	0.76 [0.72, 0.81]
coefficient 50	RMU	✓	0.48	0.85	+0.55 [+0.51, +0.58]	−0.52 [−0.56, −0.48]	0.84 [0.80, 0.88]
coefficient 60	RMU	✓	0.12	0.65	+0.38 [+0.34, +0.42]	−0.60 [−0.64, −0.56]	0.80 [0.76, 0.84]
coefficient 70	RMU		0.21	0.41	+0.22 [+0.18, +0.25]	−0.60 [−0.63, −0.56]	0.78 [0.73, 0.82]
coefficient 80	RMU		0.11	0.29	+0.15 [+0.12, +0.18]	−0.59 [−0.62, −0.55]	0.70 [0.65, 0.74]
default	WGA		0.00	0.10	−0.12 [−0.15, −0.08]	−0.65 [−0.69, −0.62]	0.43 [0.38, 0.48]
Qwen2.5-7B, fact set A							
default	SimNPO	✓	0.00	0.95	+0.81 [+0.77, +0.85]	−0.51 [−0.54, −0.47]	0.74 [0.69, 0.78]
default	NPO	✓	0.84	0.97	+0.99 [+0.94, +1.03]	−0.50 [−0.54, −0.46]	0.76 [0.71, 0.80]
$\beta = 0.05$	NPO	✓	0.96	0.97	+0.97 [+0.93, +1.01]	−0.51 [−0.55, −0.47]	0.79 [0.74, 0.83]
30 epochs	NPO	✓	0.93	0.98	+1.01 [+0.97, +1.05]	−0.50 [−0.54, −0.46]	0.78 [0.74, 0.82]
default	WGA		0.91	0.63	+0.38 [+0.33, +0.42]	−0.43 [−0.47, −0.39]	0.70 [0.65, 0.74]
Qwen2.5-7B, fact set B							
default	SimNPO	✓	0.00	0.96	+0.88 [+0.84, +0.92]	−0.29 [−0.32, −0.27]	0.90 [0.87, 0.93]
Qwen2.5-7B, fact set C							
default	SimNPO	✓	0.00	0.97	+0.92 [+0.88, +0.96]	−0.39 [−0.44, −0.35]	0.55 [0.50, 0.60]
Qwen3-4B, fact set A							
default	SimNPO	✓	0.94	0.97	+0.96 [+0.93, +1.00]	−0.44 [−0.48, −0.40]	0.73 [0.69, 0.78]
$\beta = 1.0$	SimNPO	✓	0.96	0.98	+0.93 [+0.90, +0.97]	−0.47 [−0.51, −0.43]	0.79 [0.75, 0.83]
$\beta = 1.0$, raised lr	SimNPO		0.00	0.56	+0.43 [+0.38, +0.48]	−0.34 [−0.38, −0.30]	0.38 [0.33, 0.43]
default	NPO	✓	0.99	0.99	+1.03 [+1.00, +1.07]	−0.46 [−0.50, −0.42]	0.76 [0.71, 0.80]
default	RMU	✓	0.45	0.25	−0.01 [−0.06, +0.05]	+0.21 [+0.15, +0.28]	0.73 [0.68, 0.77]
coefficient 12	RMU	✓	0.05	0.14	−0.13 [−0.18, −0.08]	−0.03 [−0.09, +0.03]	0.70 [0.65, 0.74]
coefficient 6	RMU	✓	0.21	0.14	−0.14 [−0.19, −0.08]	+0.19 [+0.13, +0.26]	0.72 [0.68, 0.77]
seed 99999	RMU	✓	0.08	0.14	−0.06 [−0.10, −0.01]	−0.24 [−0.30, −0.17]	0.65 [0.60, 0.70]
seed 7777	RMU	✓	0.06	0.18	+0.02 [−0.03, +0.07]	−0.41 [−0.48, −0.34]	0.63 [0.58, 0.68]

Model	Model, fact set	Observed rate	Sector-only prediction	Sector-only R^2
untrained	Llama-3.1-8B, A	0.48	0.49	+0.70
untrained	Llama-3.1-8B, B	0.49	0.49	+0.75
untrained	Llama-3.1-8B, C	0.50	0.50	+0.76
untrained	Qwen2.5-7B, A	0.46	0.49	+0.85
untrained	Qwen2.5-7B, B	0.54	0.52	+0.84
untrained	Qwen3-4B, A	0.47	0.49	+0.85
retrained	Llama-3.1-8B, A	0.03	0.56	−3.90
retrained	Llama-3.1-8B, B	0.02	0.54	−4.43
retrained	Llama-3.1-8B, C	0.03	0.56	−4.55
retrained	Qwen2.5-7B, A	0.05	0.55	−2.99
retrained	Qwen2.5-7B, B	0.18	0.56	−1.17
retrained	Qwen2.5-7B, C	0.08	0.58	−2.02
retrained	Qwen3-4B, A	0.08	0.55	−2.42
RMU, default	Qwen3-4B, A	0.73	0.54	−0.32
RMU, coefficient 12	Qwen3-4B, A	0.70	0.52	+0.01
RMU, coefficient 6	Qwen3-4B, A	0.72	0.53	−0.20
RMU, seed 99999	Qwen3-4B, A	0.65	0.51	−0.01
RMU, seed 7777	Qwen3-4B, A	0.63	0.50	+0.14

Table 13. Sector preference explains the untrained models' choices but not the RMU models'. Sector-only control for the wrong-employer prompt. *Observed rate*: rate of keeping the true sector. *Sector-only prediction*: the same rate predicted out of fold from one score per sector, with nothing about the person. R^2 is out of fold. The fit is designed for untrained models, whose choices reflect sector preferences alone. Retrained models follow the named employer, so their R^2 is strongly negative (gray).

the true sector for 0.77–1.00 of the few people they answer (33–43 of 360), as often as the RMU models do (0.74–0.78).

C.3 Variants of the wrong-employer prompt

Table 14 scores the reference models of two combinations, and the five RMU reruns of Section 5, on three variants of the wrong-employer prompt, all with the full-sentence scorer. The *10-way* variant keeps the wrong-employer prompt and offers the true sector, the named employer's sector and eight others, so its chance level (0.10) matches the no-context question. *Generated* decodes greedily from the wrong-employer prompt and counts answers that name the true sector. Both separate the retrained and original models as the 2-way prompt does: the retrained models keep the true sector at most 3% of the time, the original models 81–83%.

The *other person* variant states a true fact about someone who works at the wrong employer ("[another person] works at Drazin Labs.") and asks about the target person, with the same two options. We intended it as a control that separates a preference for this person's sector from ignoring the prompt. It is not a clean control: the original models, which know the target, fall to 0.50 and 0.61, so naming an employer pulls the answer toward its sector even when it belongs to someone else. We do not use it elsewhere.

A target the model must generate. On a further fact set, each of 72 employers has its own made-up sector word, shared by five of the 360 people, and every answer is scored by greedy generation. There the wrong-employer prompt separates nothing: the original model, which generates the target for 81% of people without context, names the true word under a wrong employer for none of them and the wrong employer's word for 94%. We use this set nowhere else.

C.4 RMU rerun from a rebuilt original model

The reruns start from a Qwen3-4B original model rebuilt with the same recipe and use the same hyperparameters and batch size; the rebuilt retrained model is trained the same way. Table 14 gives their scores. They answer the correct-employer prompt for 12–18% of people (untrained models: 9–12%; retrained: 0.66 rebuilt, 0.78 first build), keep the true sector under a wrong employer 0.50–0.57 of the time, and fail the answer-in-prompt check in three of five cases (0.84, 0.89 and 0.897).

Model, fact set		No context	Wrong employer 2-way	Wrong employer 10-way	Wrong employer Generated	Other person
Llama-3.1-8B, set C	retrained	0.13	0.03	0.02	0.00	0.14
	original	0.99	0.84	0.83	0.81	0.50
Qwen3-4B, set A	retrained	0.09	0.10	0.03	0.00	0.20
	original	1.00	0.82	0.81	0.81	0.61
	rerun, $c = 2$	0.13	0.53	0.08	0.00	0.47
	rerun, $c = 6$	0.12	0.54	0.14	0.00	0.43
	rerun, $c = 12$	0.14	0.57	0.14	0.00	0.43
	rerun, $c = 2$, seed 99999	0.14	0.56	0.11	0.00	0.46
	rerun, $c = 2$, seed 7777	0.14	0.50	0.11	0.00	0.42

Table 14. Wrong-employer prompt variants on the reference models and the RMU reruns: rate of choosing (or, for *Generated*, producing) the true sector. *No context*: the 10-way no-context question, for comparison. c: RMU steering coefficient. The Qwen3-4B references are the models rebuilt with the same recipe (Section 5). The Llama set C scores come from a later evaluation of the stored checkpoints and differ from Table 3 (original model under a wrong employer: 0.84 against 0.77); the difference comes from the fix to the scorer's attention mask (Appendix A.3), which leaves the retrained model's rate unchanged; the no-context column uses the 10-way menu that includes the wrong employer's sector.

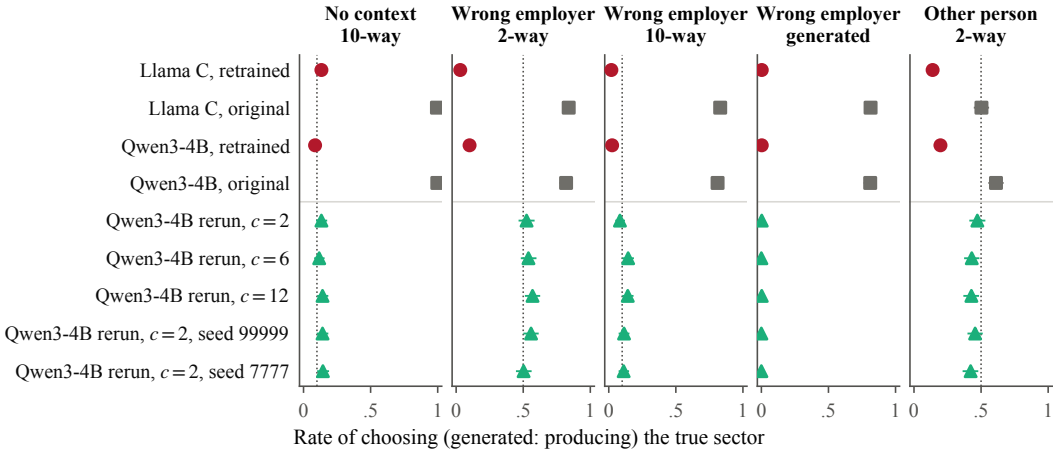

Figure 10. The 10-way and generated variants separate the retrained and original models as the 2-way prompt does; the other-person variant does not. Rate of choosing (for *generated*, producing) the true sector, with 95% intervals, for the values in Table 14. Crimson: retrained; gray: original; triangles: the five Qwen3-4B RMU reruns. Dotted lines: chance.

To check whether they lost the retained employer→sector facts, we asked each of the 72 employers' sectors directly ("What sector does Telmod Labs operate in?", 10-way, full-sentence candidates). All five reruns answer every employer correctly, as do the rebuilt Qwen3-4B original model (1.00) and the retrained models (0.96 for the rebuilt Qwen3-4B model; 1.00 for Llama on set C, whose original model scores 0.99). The reruns therefore keep both retained facts while answering the question about the person's sector far less often. Restricting the wrong-employer rate to people whose correct-employer prompt a rerun answers gives 0.67–0.81, but the untrained models reach 0.77–1.00 under the same restriction (Appendix C.2), so we draw no conclusion from it.

C.5 Further RMU settings

After the reruns we ran further RMU settings, to find where the pattern of Section 5 appears in the rebuilt Qwen3-4B model and in Llama. The five Qwen3-4B settings, all from the rebuilt original model, were fixed before any of them was scored. On Llama set C, the first setting (coefficient 65, 15 epochs) broke the model; after seeing it we ran the other three for 5 epochs instead. Table 16 gives every setting with the matching retrained model. On Qwen3-4B, the two settings trained for 5 epochs stay healthy, use the employer and are flagged by conflicting context, and otherwise only by one or two generations of 360; the three trained longer answer the correct-employer prompt for 16–22%

of people, and the two of them that stay healthy are flagged by no test. On Llama set C, three of the four settings fail the health checks, and the healthy one (coefficient 65, 5 epochs) answers the correct-employer prompt for 11% of people, keeps the true sector under a wrong employer less often than chance (0.36) and is flagged by no test. RMU's effect thus depends on the model, the starting checkpoint and the training length, and conflicting context reads it only while the model still uses the employer.

Llama with RMU at layer 20. Separately, we had run RMU on Llama set C at a deeper layer (steering at layer 20, updating layers 19–21) with seven steering coefficients, scored with the all-10-way scorer before the fix to the attention mask (Appendix A.3). Table 15 gives every setting. The two smallest coefficients (30 and 50) leave the model healthy and barely forgotten: it picks the true sector without context 63% and 35% of the time, and the no-context question flags both. At coefficients 70 and 75 the model stays healthy and passes the no-context question, and it still uses the employer (correct-employer accuracy 0.63 and 0.54), yet it keeps the true sector under a wrong employer 35% and 30% of the time with ten options, against 11% for the retrained model (differences $+0.24$ $[0.19, 0.30]$ and $+0.19$ $[0.14, 0.25]$). Larger coefficients fail the answer-in-prompt check. For these checkpoints we have only sampled generation (temperature 1.3, 64 samples per person; 16 for the reference models), in which they produce the target in 8–14% of samples and the retrained model in none.

c	Answer in prompt	Employer	Sampled generation	Δ no-context margin	Correct employer	Wrong employer (10-way)
70	0.94	0.99	0.14	−0.03 [−0.07, +0.01]	0.63	0.35 [0.30, 0.40]
75	0.90	0.99	0.13	−0.04 [−0.08, −0.00]	0.54	0.30 [0.25, 0.35]
80†	0.87	0.99	0.12	−0.05 [−0.09, −0.01]	0.48	0.27 [0.23, 0.32]
85†	0.83	0.99	0.11	−0.05 [−0.09, −0.02]	0.42	0.26 [0.22, 0.31]
100†	0.82	0.99	0.08	−0.04 [−0.08, −0.01]	0.42	0.26 [0.21, 0.31]
Retrained	0.95	1.00	0.00	0.00	0.91	0.11 [0.07, 0.14]
Original	0.99	0.99	0.95	+0.92 [+0.89, +0.96]	0.99	0.88 [0.84, 0.91]
Untrained	0.18	0.09	0.00	−0.15 [−0.19, −0.11]	0.09	0.10 [0.07, 0.13]

Table 15. On Llama set C with RMU at layer 20, the healthy settings with coefficients 70 and 75 pass the no-context question but keep the true sector under a wrong employer three times as often as the retrained model. All-10-way scorer. *Sampled generation*: rate of producing the target per sample without context (temperature 1.3). Δ no-context margin: model minus the retrained model, in nats with 95% intervals. *Wrong employer*: rate of keeping the true sector with ten options (95% interval). The retrained model's employer accuracy comes from the full-sentence scorer. †: fails the answer-in-prompt check. Coefficients 30 and 50, flagged by the no-context question, are described in the text.

C.6 Datasets without a clean reference

Two datasets use real entities and fall outside the controlled regime of the main text. In the real-companies dataset, fabricated people work at real employers such as Boeing ($n = 225$, 45 employers). In the real-people dataset, both people and employers are real, taken from Wikidata ($n = 68$). The main text studies a regime in which the retrained model is at chance without context and the untrained model cannot use the employer. Neither dataset has either property. The retrained models answer the no-context question correctly 0.63 (real companies) and 0.71 (real people) of the time, and the untrained model, which already knows these employers, reaches 0.92 and 0.88 once the employer is named (Table 3). Table 17 lists the 19 settings on these datasets.

Setting	Healthy	Free gen. (of 360)	No context	Correct employer	Wrong employer 2-way	Wrong employer 10-way	Employer→ sector
Qwen3-4B, set A, from the rebuilt original model							
$c = 2$, 5 epochs	✓	1	0.20	0.44	0.56 [0.51, 0.61]	0.26	1.00
$c = 6$, 5 epochs	✓	2	0.18	0.43	0.58 [0.53, 0.63]	0.31	1.00
$c = 2$, 8 epochs	✓	0	0.13	0.22	0.50 [0.45, 0.55]	0.12	1.00
$c = 0.5$, 15 epochs	✓	0	0.14	0.17	0.50 [0.45, 0.55]	0.13	1.00
$c = 1$, 15 epochs		0	0.14	0.16	0.57 [0.52, 0.62]	0.13	1.00
retrained, rebuilt		0	0.09	0.66	0.10	0.03	0.96
Llama-3.1-8B, set C							
$c = 40$, 5 epochs		111	0.61	0.71	0.77 [0.72, 0.81]	0.55	0.99
$c = 65$, 5 epochs	✓	0	0.06	0.11	0.36 [0.31, 0.41]	0.09	0.99
$c = 100$, 5 epochs		0	0.18	0.31	0.56 [0.51, 0.61]	0.25	0.99
$c = 65$, 15 epochs		0	0.05	0.10	0.39 [0.34, 0.44]	0.09	0.99
retrained		0	0.12	0.91	0.03	0.02	1.00

Table 16. Further RMU settings: trained for 5 epochs from the rebuilt Qwen3-4B model, RMU stays healthy, uses the employer and keeps the forgotten sector under a wrong employer; trained longer, it stops using the employer. c: steering coefficient. *Free gen.*: people (of 360) for whom greedy decoding produces the target. *No context*, *Correct employer*: 10-way accuracy. *Wrong employer*: rate of keeping the true sector with two options (95% interval) and with ten. *Employer→sector*: accuracy on the retained employer facts. Gray: the retrained model of the same combination.

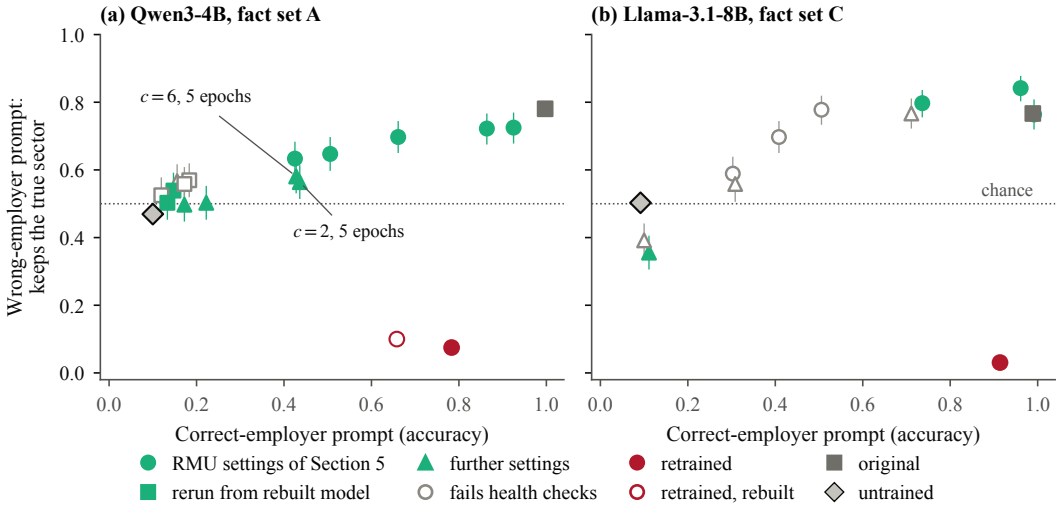

Figure 11. Across all RMU runs, the more an unlearned model still uses the employer, the more often it keeps the forgotten sector under a wrong employer. Every RMU run on Qwen3-4B (set A) and Llama-3.1-8B (set C): accuracy on the correct-employer prompt against the rate of keeping the true sector on the wrong-employer prompt (2-way, 95% intervals). Circles: the settings of Section 5; squares: the reruns from the rebuilt original model (Appendix C.4); triangles: the further settings (Appendix C.5). Filled: healthy; hollow gray: fails the health checks. Crimson: retrained models (open: the rebuilt Qwen3-4B one); gray square: original; diamond: untrained. Dotted line: chance.

Table 17. Settings on the two real-entity datasets, whose retrained models already answer without context. Columns as in Table 12. The Δ columns compare against a retrained model that already answers without context (Table 3), so they measure disagreement with retained-only fine-tuning without isolating what the target examples added.

Setting	Method	Healthy	Free gen.	No context	Δ no-context	Δ context gain	Keeps true sector
Llama-3.1-8B, real companies							
default	NPO		0.00	0.12	−0.92 [−1.02, −0.82]	−0.07 [−0.15, +0.01]	0.76 [0.70, 0.81]
2 epochs	NPO		–	0.58	+0.02 [−0.07, +0.11]	−0.00 [−0.06, +0.06]	0.65 [0.59, 0.72]
4 epochs	NPO		–	0.22	−0.62 [−0.71, −0.52]	+0.43 [+0.37, +0.50]	0.69 [0.63, 0.75]
6 epochs	NPO		–	0.10	−1.01 [−1.12, −0.91]	+0.92 [+0.83, +1.00]	0.63 [0.57, 0.69]
8 epochs	NPO		–	0.08	−0.80 [−0.89, −0.71]	−0.50 [−0.58, −0.41]	0.78 [0.73, 0.84]
default	WGA	✓	0.01	0.66	+0.03 [−0.02, +0.09]	−0.29 [−0.33, −0.25]	0.84 [0.80, 0.89]
4 epochs	WGA	✓	–	0.68	+0.01 [−0.05, +0.06]	−0.38 [−0.42, −0.35]	0.78 [0.72, 0.83]
6 epochs	WGA		–	0.49	−0.13 [−0.18, −0.08]	−0.16 [−0.20, −0.12]	0.80 [0.75, 0.85]
8 epochs	WGA		–	0.71	+0.02 [−0.04, +0.08]	−0.35 [−0.39, −0.31]	0.68 [0.62, 0.75]
Llama-3.1-8B, real people							
default	NPO		–	0.15	−1.03 [−1.26, −0.79]	−0.14 [−0.24, −0.03]	0.49 [0.37, 0.60]
10 epochs	NPO		–	0.37	−0.45 [−0.67, −0.24]	+0.17 [+0.07, +0.28]	0.79 [0.69, 0.88]
12 epochs	NPO	✓	–	0.84	+0.35 [+0.15, +0.54]	−0.12 [−0.21, −0.04]	0.88 [0.81, 0.96]
6 epochs	NPO	✓	–	0.90	+0.41 [+0.25, +0.58]	−0.14 [−0.21, −0.07]	0.96 [0.90, 1.00]
8 epochs	NPO	✓	–	0.90	+0.48 [+0.30, +0.66]	−0.20 [−0.28, −0.12]	0.94 [0.88, 0.99]
default	WGA		–	0.18	−0.49 [−0.61, −0.38]	−0.26 [−0.32, −0.20]	0.65 [0.53, 0.76]
10 epochs	WGA	✓	–	0.41	−0.34 [−0.45, −0.22]	−0.20 [−0.26, −0.14]	0.49 [0.37, 0.60]
12 epochs	WGA		–	0.07	−0.62 [−0.74, −0.50]	−0.25 [−0.31, −0.19]	0.44 [0.32, 0.56]
6 epochs	WGA		–	0.24	−0.72 [−0.89, −0.56]	−0.26 [−0.38, −0.15]	0.53 [0.41, 0.65]
8 epochs	WGA		–	0.35	−0.39 [−0.51, −0.26]	−0.24 [−0.31, −0.16]	0.81 [0.72, 0.90]

D MODELS THAT FAIL THE HEALTH CHECKS

Of the 145 unlearning settings on clean-reference data, 97 fail the health checks, as do two of the five random-direction MEMIT controls. We count them in neither direction: a model that fails the answer-in-prompt check, or no longer knows a person's employer, has no reason to follow a wrong employer.

D.1 MEMIT

We edit the Llama model on set A with MEMIT (Meng et al., 2023), one layer at a time (layers 6, 7, 8, 9, 11 and 14) or all six together. Each edit starts from the original model and targets all 360 target facts at once; a few facts per layer fail a sanity check and are skipped, leaving 352–356. For each fact, the value vector is optimized for 30 steps so that the sector tokens match the untrained model's distribution, plus a locality term on that person's single-hop and answer-in-prompt prompts (weight 1 unless Table 18 gives another). The update to the layer's MLP output projection is $\Delta W = RK^\top(wC + \lambda I + KK^\top)^{-1}$, where C is the key covariance from about two million Wikitext tokens, w is the covariance weight and λI is a small ridge term. The random-direction control replaces ΔW with a Gaussian matrix of the same Frobenius norm.

None of the 57 edits passes the health checks. The best single-hop accuracy is 0.76. Thirty edits bring no-context accuracy below the retrained model's 0.128. Two edits reach it or go below while keeping answer-in-prompt accuracy above 0.90, both at layer 11 with $w = 30{,}000$ (locality weight 1 and 3; no-context 0.10 and 0.13), and they lose most person→employer facts (employer check 0.29 and 0.26). Edits that remove the target also remove the people.

At low weight, a random update of the same norm matches the real edit. At $w = 1{,}000$, where the real layer-11 edit reaches chance, the random update gives no-context 0.13, answer in prompt 0.18, single-hop 0.07, against 0.06, 0.08 and 0.12 for the real edit. At $w \geq 30{,}000$ the random updates leave the model intact (no-context 0.99; these are the three healthy controls), but at those weights the real layer-11 edit has already lost much of person→employer (single-hop 0.29–0.67). There is no weight at which the real edit removes the target, keeps the model healthy, and differs from its random control (Figure 12).

D.2 WGA AND GRADIENT ASCENT

All 17 WGA settings (Wang et al., 2025) keep the single-hop facts (accuracy at least 0.997) and fail the answer-in-prompt check (0.11–0.88). On Llama, WGA drives no-context accuracy to 0.04–0.10, at or below the retrained model. It does so by damaging the model's ability to read the answer from context. The one WGA setting on Qwen2.5-7B keeps no-context accuracy at 0.63.

Gradient ascent shows the opposite failure. On set A it brings no-context accuracy to 0.01, below both the retrained model (0.13) and the untrained model (0.09), and keeps answer-in-prompt accuracy at 1.00. But single-hop accuracy falls to 0.04: the model no longer knows where anyone works, so the correct employer recovers nothing (0.01). It reaches the untrained level on the target by forgetting the people. GradDiff, CE-U and PDU fail one or both checks by wide margins (Table 12).

D.3 RMU ON OLMO

An earlier set of experiments ran RMU on OLMo-2-7B with a different fact set ($n = 121$) and the all-10-way scorer. With retain weight $\alpha = 5$ at layer 7, steering coefficients 45 to 70 for six epochs (plus coefficient 60 at five epochs and with a second seed) give answer-in-prompt accuracy 0.94–0.99 and no-context accuracy 0.099–0.141. Given the correct employer, these models answer 0.19–0.33 of items correctly, against 0.099 for the untrained model.

This dataset has no retrained model; the model used as the reference there answers the no-context question with accuracy 1.0 and is the original model. On the same data, loss-based methods tuned to similar no-context accuracy lose the answer-in-prompt check (SimNPO 0.46, NPO 0.41, GradDiff 0.14, WGA 0.09; untrained model 1.00).

Table 18. No MEMIT edit passes the health checks. MEMIT edits on Llama-3.1-8B, fact set A. Each setting gives the edited layer (layers 6–14: layers 6, 7, 8, 9, 11 and 14 together), the covariance weight w, the weight of the locality term when it is not 1, and whether the update is in a random direction with the same norm. No-context, answer-in-prompt and single-hop columns are accuracies. Other columns as in Table 12; Δ correct employer is the correct-employer margin, unlearned minus retrained. For reference, the untrained model scores 0.09, 0.19 and 0.10 on the first three columns.

Setting	No context	Answer in prompt	Single-hop	Δ no-context	Δ correct employer	Keeps true sector
Llama-3.1-8B, fact set A						
layer 11, $w = 100{,}000$	0.16	1.00	0.43	+0.04 [+0.02, +0.07]	−0.47 [−0.50, −0.44]	0.40 [0.35, 0.45]
layer 11, $w = 100{,}000$, random direction	0.99	1.00	0.99	+0.70 [+0.67, +0.73]	+0.21 [+0.17, +0.24]	0.71 [0.66, 0.76]
layer 11, $w = 10{,}000$	0.08	0.50	0.16	−0.15 [−0.18, −0.12]	−0.71 [−0.75, −0.67]	0.42 [0.37, 0.47]
layer 11, $w = 10{,}000$, locality 10	0.14	0.38	0.21	−0.03 [−0.07, −0.00]	−0.64 [−0.67, −0.59]	0.44 [0.39, 0.49]
layer 11, $w = 10{,}000$, locality 100	0.24	0.85	0.22	+0.09 [+0.06, +0.12]	−0.46 [−0.49, −0.42]	0.61 [0.56, 0.66]
layer 11, $w = 10{,}000$, locality 30	0.18	0.52	0.19	+0.03 [−0.00, +0.06]	−0.52 [−0.56, −0.49]	0.53 [0.48, 0.58]
layer 11, $w = 10{,}000$, locality 300	0.30	0.89	0.30	+0.14 [+0.11, +0.17]	−0.38 [−0.41, −0.34]	0.60 [0.55, 0.65]
layer 11, $w = 10{,}000$, locality 5	0.09	0.55	0.16	−0.06 [−0.10, −0.03]	−0.64 [−0.68, −0.60]	0.42 [0.37, 0.47]
layer 11, $w = 10{,}000$, random direction	0.64	0.98	0.51	+0.29 [+0.26, +0.31]	−0.26 [−0.29, −0.22]	0.65 [0.60, 0.70]
layer 11, $w = 1{,}000$	0.06	0.08	0.12	−0.27 [−0.31, −0.23]	−0.86 [−0.90, −0.81]	0.41 [0.36, 0.46]
layer 11, $w = 1{,}000$, locality 10	0.09	0.14	0.11	−0.21 [−0.24, −0.17]	−0.81 [−0.85, −0.76]	0.45 [0.40, 0.50]
layer 11, $w = 1{,}000$, locality 100	0.09	0.40	0.14	−0.14 [−0.17, −0.10]	−0.66 [−0.70, −0.62]	0.50 [0.45, 0.55]
layer 11, $w = 1{,}000$, locality 3	0.09	0.14	0.10	−0.23 [−0.27, −0.19]	−0.85 [−0.89, −0.80]	0.44 [0.39, 0.49]
layer 11, $w = 1{,}000$, locality 30	0.12	0.23	0.12	−0.08 [−0.12, −0.05]	−0.64 [−0.68, −0.60]	0.50 [0.45, 0.55]
layer 11, $w = 1{,}000$, locality 300	0.16	0.36	0.15	−0.09 [−0.13, −0.05]	−0.60 [−0.65, −0.56]	0.52 [0.47, 0.57]
layer 11, $w = 1{,}000$, locality 5	0.07	0.11	0.11	−0.27 [−0.32, −0.23]	−0.82 [−0.87, −0.78]	0.41 [0.36, 0.46]
layer 11, $w = 1{,}000$, random direction	0.13	0.18	0.07	−0.05 [−0.09, −0.02]	−0.62 [−0.66, −0.59]	0.49 [0.44, 0.55]
layer 11, $w = 300{,}000$	0.30	1.00	0.67	+0.14 [+0.12, +0.17]	−0.34 [−0.38, −0.31]	0.39 [0.34, 0.44]
layer 11, $w = 300{,}000$, random direction	0.99	1.00	0.99	+0.71 [+0.68, +0.74]	+0.22 [+0.19, +0.25]	0.71 [0.66, 0.76]
layer 11, $w = 30{,}000$	0.10	0.94	0.29	−0.06 [−0.09, −0.03]	−0.59 [−0.63, −0.56]	0.42 [0.36, 0.47]
layer 11, $w = 30{,}000$, locality 10	0.23	0.98	0.25	+0.10 [+0.07, +0.12]	−0.44 [−0.47, −0.41]	0.45 [0.40, 0.50]
layer 11, $w = 30{,}000$, locality 100	0.34	1.00	0.31	+0.17 [+0.14, +0.20]	−0.35 [−0.38, −0.32]	0.59 [0.54, 0.64]
layer 11, $w = 30{,}000$, locality 3	0.13	0.98	0.26	+0.02 [−0.01, +0.05]	−0.52 [−0.56, −0.49]	0.40 [0.35, 0.45]
layer 11, $w = 30{,}000$, locality 30	0.24	0.99	0.28	+0.14 [+0.11, +0.16]	−0.39 [−0.42, −0.35]	0.52 [0.47, 0.58]
layer 11, $w = 30{,}000$, locality 300	0.36	1.00	0.40	+0.19 [+0.17, +0.22]	−0.30 [−0.33, −0.27]	0.56 [0.51, 0.61]
layer 11, $w = 30{,}000$, locality 5	0.14	0.99	0.24	+0.06 [+0.03, +0.08]	−0.48 [−0.51, −0.45]	0.42 [0.37, 0.47]
layer 11, $w = 30{,}000$, random direction	0.99	1.00	0.98	+0.60 [+0.57, +0.63]	+0.10 [+0.07, +0.13]	0.73 [0.68, 0.77]
layer 11, $w = 3{,}000$, locality 10	0.08	0.20	0.12	−0.19 [−0.23, −0.15]	−0.80 [−0.84, −0.75]	0.43 [0.38, 0.48]
layer 11, $w = 3{,}000$, locality 100	0.14	0.48	0.14	−0.12 [−0.16, −0.08]	−0.68 [−0.72, −0.63]	0.54 [0.48, 0.59]
layer 11, $w = 3{,}000$, locality 30	0.13	0.32	0.13	−0.09 [−0.13, −0.05]	−0.64 [−0.69, −0.60]	0.48 [0.43, 0.53]
layer 11, $w = 3{,}000$, locality 300	0.21	0.44	0.21	+0.00 [−0.04, +0.04]	−0.51 [−0.55, −0.47]	0.56 [0.51, 0.61]
layer 11, $w = 3{,}000$, locality 5	0.08	0.21	0.16	−0.24 [−0.28, −0.20]	−0.82 [−0.86, −0.77]	0.40 [0.35, 0.45]
layer 14, $w = 100{,}000$	0.24	0.96	0.49	+0.11 [+0.09, +0.14]	−0.37 [−0.41, −0.34]	0.28 [0.24, 0.33]
layer 14, $w = 10{,}000$	0.09	0.43	0.17	−0.18 [−0.22, −0.14]	−0.75 [−0.79, −0.71]	0.39 [0.34, 0.44]
layer 14, $w = 1{,}000$	0.06	0.19	0.09	−0.37 [−0.42, −0.32]	−0.89 [−0.93, −0.84]	0.45 [0.39, 0.50]
layer 14, $w = 300{,}000$	0.44	1.00	0.76	+0.20 [+0.17, +0.23]	−0.22 [−0.26, −0.19]	0.22 [0.18, 0.26]
layer 14, $w = 30{,}000$	0.13	0.79	0.37	−0.01 [−0.04, +0.02]	−0.54 [−0.57, −0.50]	0.34 [0.29, 0.39]
layer 6, $w = 100{,}000$	0.10	0.89	0.38	−0.07 [−0.10, −0.03]	−0.61 [−0.65, −0.57]	0.39 [0.34, 0.44]
layer 6, $w = 10{,}000$	0.09	0.36	0.17	−0.23 [−0.27, −0.19]	−0.78 [−0.82, −0.74]	0.44 [0.39, 0.49]
layer 6, $w = 1{,}000$	0.06	0.11	0.13	−0.24 [−0.27, −0.20]	−0.80 [−0.84, −0.76]	0.41 [0.36, 0.46]
layer 6, $w = 300{,}000$	0.22	0.98	0.60	+0.06 [+0.03, +0.09]	−0.46 [−0.49, −0.42]	0.43 [0.38, 0.48]
layer 6, $w = 30{,}000$	0.09	0.67	0.19	−0.18 [−0.22, −0.14]	−0.75 [−0.79, −0.70]	0.36 [0.31, 0.41]
layer 7, $w = 100{,}000$	0.13	0.97	0.41	−0.02 [−0.05, +0.02]	−0.55 [−0.58, −0.51]	0.40 [0.35, 0.45]
layer 7, $w = 10{,}000$	0.06	0.49	0.20	−0.19 [−0.23, −0.16]	−0.78 [−0.82, −0.74]	0.41 [0.36, 0.46]
layer 7, $w = 1{,}000$	0.07	0.21	0.18	−0.23 [−0.27, −0.19]	−0.80 [−0.84, −0.76]	0.42 [0.37, 0.47]
layer 7, $w = 300{,}000$	0.26	0.99	0.61	+0.10 [+0.07, +0.13]	−0.41 [−0.44, −0.37]	0.44 [0.39, 0.49]
layer 7, $w = 30{,}000$	0.07	0.82	0.24	−0.13 [−0.16, −0.10]	−0.69 [−0.72, −0.65]	0.36 [0.31, 0.41]
layer 8, $w = 100{,}000$	0.13	0.99	0.30	+0.01 [−0.02, +0.04]	−0.52 [−0.56, −0.49]	0.42 [0.37, 0.48]
layer 8, $w = 10{,}000$	0.08	0.32	0.22	−0.18 [−0.22, −0.14]	−0.80 [−0.84, −0.75]	0.43 [0.38, 0.48]
layer 8, $w = 1{,}000$	0.07	0.12	0.14	−0.29 [−0.33, −0.25]	−0.91 [−0.96, −0.86]	0.46 [0.41, 0.51]
layer 8, $w = 300{,}000$	0.28	1.00	0.55	+0.13 [+0.10, +0.16]	−0.40 [−0.43, −0.36]	0.49 [0.44, 0.54]

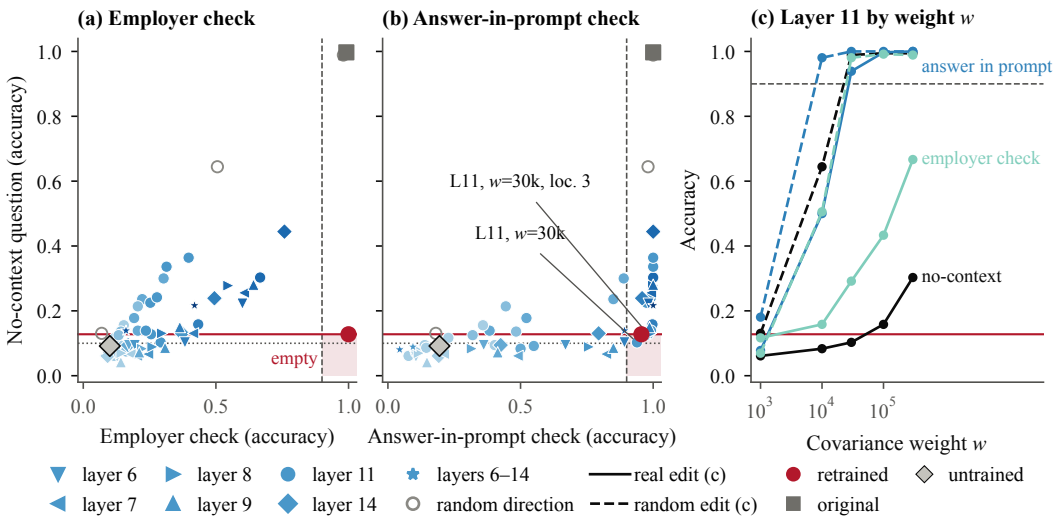

Figure 12. No MEMIT edit removes the target and keeps the people. Edits of Llama-3.1-8B on fact set A. (a, b) One point per edit; marker gives the edited layer, darker blue a larger covariance weight w, hollow gray a random-direction update of the same norm. Crimson line: the retrained model's no-context accuracy (0.128); dashed line: the 0.90 health bar; dotted line: chance. The shaded corner holds edits that pass the check on the horizontal axis and answer the no-context question no better than the retrained model: in (a) it is empty (best employer check 0.76); in (b) it holds the two layer-11 edits with $w = 30{,}000$, which keep 0.29 and 0.26 of person→employer facts. (c) Layer 11 at the default locality weight: real edits (solid) keep the no-context question near chance up to $w = 30{,}000$ while the employer check stays at or below 0.29; random updates (dashed) reach 0.13 only at $w = 1{,}000$, where they also break both checks.

Table 18. MEMIT edits on Llama-3.1-8B, fact set A (continued).

Setting	No context	Answer in prompt	Single-hop	Δ no-context	Δ correct employer	Keeps true sector
layer 8, $w = 30{,}000$	0.09	0.76	0.26	−0.08 [−0.12, −0.05]	−0.67 [−0.71, −0.63]	0.40 [0.35, 0.45]
layer 9, $w = 100{,}000$	0.15	0.99	0.36	+0.03 [+0.00, +0.06]	−0.50 [−0.53, −0.46]	0.43 [0.38, 0.48]
layer 9, $w = 10{,}000$	0.07	0.41	0.21	−0.15 [−0.18, −0.11]	−0.74 [−0.78, −0.70]	0.36 [0.32, 0.41]
layer 9, $w = 1{,}000$	0.04	0.15	0.14	−0.23 [−0.26, −0.19]	−0.81 [−0.85, −0.77]	0.38 [0.33, 0.43]
layer 9, $w = 300{,}000$	0.28	1.00	0.64	+0.14 [+0.11, +0.17]	−0.37 [−0.41, −0.34]	0.47 [0.42, 0.52]
layer 9, $w = 30{,}000$	0.09	0.85	0.31	−0.07 [−0.11, −0.04]	−0.63 [−0.66, −0.59]	0.38 [0.33, 0.43]
layers 6–14, $w = 1{,}000{,}000$	0.14	0.89	0.16	−0.07 [−0.11, −0.04]	−0.64 [−0.68, −0.60]	0.51 [0.46, 0.56]
layers 6–14, $w = 100{,}000$	0.08	0.04	0.08	−0.15 [−0.19, −0.12]	−0.74 [−0.78, −0.70]	0.46 [0.41, 0.51]
layers 6–14, $w = 3{,}000{,}000$	0.22	1.00	0.42	+0.11 [+0.08, +0.14]	−0.42 [−0.46, −0.39]	0.48 [0.43, 0.53]
layers 6–14, $w = 300{,}000$	0.10	0.21	0.09	−0.13 [−0.16, −0.09]	−0.73 [−0.77, −0.69]	0.48 [0.43, 0.53]
layers 6–14, $w = 30{,}000$	0.09	0.09	0.09	−0.22 [−0.26, −0.18]	−0.81 [−0.85, −0.76]	0.50 [0.45, 0.55]

E HUBBLE DETAILS

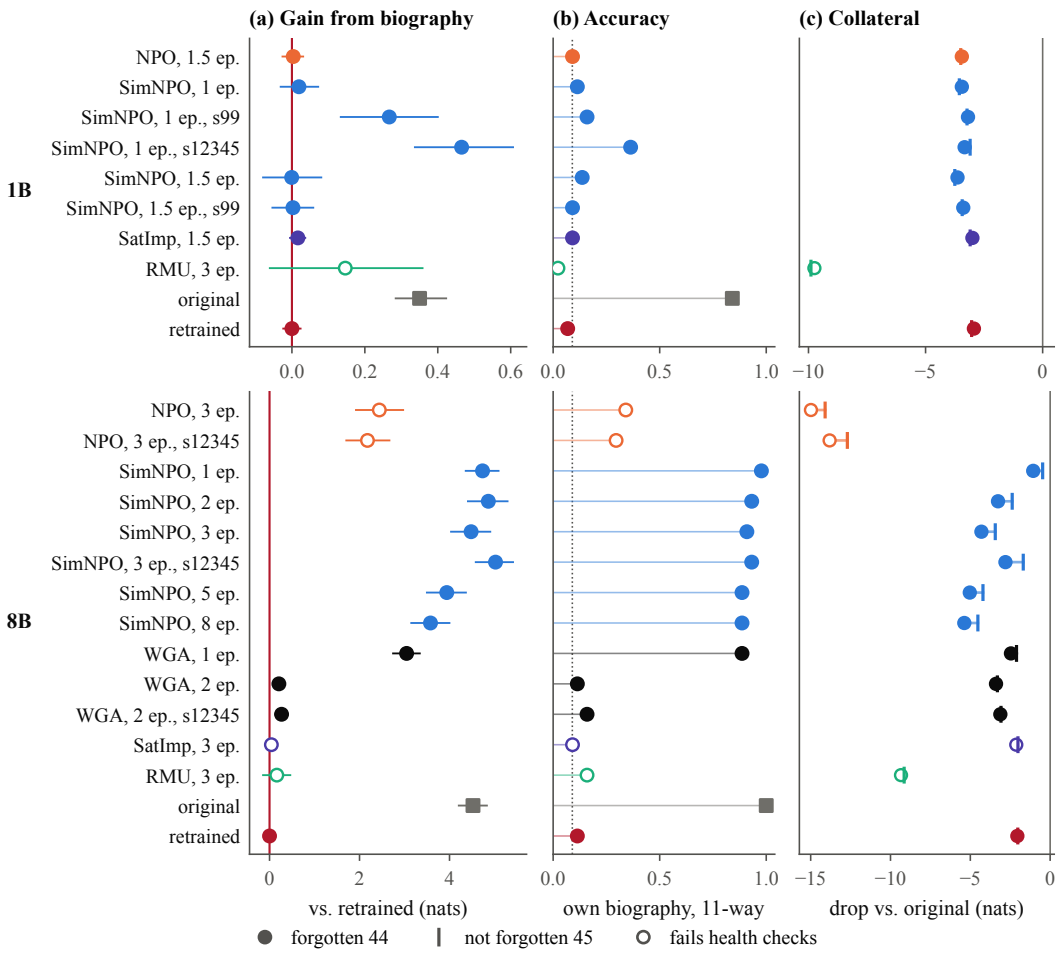

Figure 13. Healthy 8B SimNPO models recover the forgotten identifiers from the biography; at 1B the unlearned models match the retrained model, and every unlearned model also lowers identifiers it was not asked to forget. (a) Gain from the person's own biography over a name-swapped biography, minus the retrained model's gain (nats, 95% intervals; crimson line: retrained). (b) 11-way accuracy given the person's own biography (dotted: chance 0.09). (c) No-context drop against the original model for the 44 forgotten people (circles) and the 45 who were not forgotten (ticks). Hollow: fails the health checks. The two sizes use different horizontal scales in (a) and (c).

Models. Hubble (Wei et al., 2026) releases pairs of models pretrained on the same corpus, one with synthetic biographies inserted and one without. We use the 1B and 8B pairs pretrained on 500B tokens. The model with the biographies is our original model. The model without them never saw the target identifiers, but it also lacks every other inserted fact, so it is a no-exposure control rather than an exact retrained model for our 44-person request; below we call it the retrained model for brevity. This follows the design of Wei et al. (2026), who compare unlearned 8B models with the model pretrained without the biographies.

Targets. The forget set is the unique identifier (a random UUID) of 44 people whose biographies were inserted 256 times. Each forget example uses the short no-context form, *"[name] has the unique identifier [UUID]"*; the biographies themselves are not in the forget set. The other 45 people inserted at the same rate are the people who were not forgotten. For them we also score four other fields of personal data (email, birth date, birthplace, nationality), 180 items in total. No other fact implies a person's identifier. A person's nationality follows in part from their birthplace; we use this below.

Prompts. We score the length-normalized log-probability of the true UUID under four prompts:

- no context: *"[name] has the unique identifier"*;
- own biography: the person's inserted biography verbatim, cut just before the UUID, which comes last;
- swapped biography: another person's biography with this person's name substituted;
- answer in prompt: *"[name] has the unique identifier [UUID]. [name] has the unique identifier"*.

The gain from the biography is the own-biography score minus the swapped-biography score: what the true biography adds beyond a biography-shaped context with the right name. We report every model's gain minus the retrained model's gain, with an unpaired bootstrap as in Appendix A.7. We also report accuracy on the own-biography prompt: an 11-way choice between the true UUID and ten others of the same format (chance 0.09). The retrained models score 0.07 (1B) and 0.11 (8B), and their gain is near zero.

Health. Health compares each unlearned model with the original model. Items align by person here, so these intervals are paired. A model is healthy if its answer-in-prompt score and its score on the personal data of people who were not forgotten each drop by less than 0.10 nats on average.

Unlearning. We unlearn from the original model, with no chat template. The retain set has 701 facts across all five fields, the learning rate is 10^{-5}, and the forget and retain weights are 1 (for RMU, steering coefficient 6.5 and retain weight 5). The 1B models train for 1.5 epochs (RMU for three) and the 8B models for three, with seed 0. We also have a second run of 1B SimNPO (seed 99) and of 8B NPO and SimNPO (seed 12345). With the same settings otherwise, we further ran 1B SimNPO for one epoch (seeds 0, 99 and 12345), 8B SimNPO for one, two, five and eight epochs, and 8B WGA for one epoch and for two (seeds 0 and 12345).

Results. Figure 13 and Tables 19 and 20 give every setting. The clearest case is 8B SimNPO with seed 12345: it passes both health checks and its no-context score sits below the retrained model's (-0.75 nats), yet given the person's own biography it picks the true UUID 93% of the time (retrained 11%, original 100%) and gains $+5.02$ nats more than the retrained model (original $+4.52$). At 8B, SimNPO stays healthy from one to eight epochs and keeps the recovery throughout: accuracy falls only from 98% to 89% and the gain from $+4.73$ to $+3.58$ nats, while the no-context score moves from $+0.99$ nats above the retrained model to -3.32 below it. WGA recovers after one epoch (89%, gain $+3.05$ $[2.73, 3.36]$); after two epochs it picks the true UUID 11% and 16% of the time, with gains of $+0.21$ and $+0.27$ nats. At 1B the four healthy models trained for 1.5 epochs pick the true UUID 9–14% of the time (original 84%), and their gain intervals all contain zero. Of the three trained for one epoch, seed 0 behaves the same (11%), while seeds 99 and 12345 pick it 16% and 36% of the time and gain $+0.27$ $[0.13, 0.40]$ and $+0.47$ $[0.33, 0.61]$ nats more than the retrained model. NPO at 8B gains $+2.44$ and $+2.18$ but fails the health checks, as do SatImp at 8B and RMU at both sizes.

Collateral. In all 21 unlearned models, the no-context score of the 45 people who were not forgotten falls below the original model's (every 95% interval below zero). At 1B it falls 0.93–1.03 times as far as for the forgotten people; at 8B, 0.44–0.84 times for SimNPO, 0.86–0.99 for WGA and 0.92–0.98 for the other methods. Every model whose gain stays below 0.3 nats, which includes all 1B models but one and, at 8B, WGA at two epochs, SatImp and RMU, has a ratio of 0.95–1.03: it lowers the identifiers of the people who were not forgotten as much as those of the people who were. The retrained model scores -3.03 (1B) and -2.03 (8B) nats below the original model on these people, because it never saw their biographies. Their other personal data falls by less than 0.10 nats in every setting except 8B NPO.

100B-token models. On the 1B models trained on 100B tokens, our strongest SimNPO runs (learning rate 5×10^{-5}, three and six epochs) left the no-context score above the retrained model's ($+0.234$ $[0.204, 0.273]$ and $+0.199$ $[0.164, 0.242]$ nats). With learning rates of 1.7×10^{-4} and 2×10^{-4} for three epochs (seeds 0 and 99), the no-context score reaches the retrained level, but only one of the four runs passes the health checks: learning rate 1.7×10^{-4} with seed 0, whose no-context

Size	Model	Epochs	Seed	Own-biography acc.	Δ gain	Δ no-context
1B	NPO	1.5	0	0.09	+0.00 [−0.03, +0.03]	−0.52 [−0.70, −0.33]
1B	SimNPO	1	0	0.11	+0.02 [−0.03, +0.07]	−0.52 [−0.70, −0.33]
1B	SimNPO	1	99	0.16	+0.27 [+0.13, +0.40]	−0.25 [−0.45, −0.06]
1B	SimNPO	1	12345	0.36	+0.47 [+0.33, +0.61]	−0.40 [−0.76, −0.08]
1B	SimNPO	1.5	0	0.14	−0.00 [−0.08, +0.08]	−0.70 [−0.89, −0.51]
1B	SimNPO	1.5	99	0.09	+0.00 [−0.06, +0.06]	−0.45 [−0.64, −0.27]
1B	SatImp	1.5	0	0.09	+0.02 [−0.01, +0.04]	−0.06 [−0.24, +0.12]
1B	RMU†	3	0	0.02	+0.15 [−0.06, +0.36]	−6.81 [−7.11, −6.51]
1B	Retrained	–	–	0.07	0.00	0.00
1B	Original	–	–	0.84	+0.35 [+0.28, +0.43]	+2.94 [+2.70, +3.16]
8B	NPO†	3	0	0.34	+2.44 [+1.90, +2.99]	−12.93 [−13.89, −11.81]
8B	NPO†	3	12345	0.30	+2.18 [+1.69, +2.69]	−11.76 [−12.63, −10.80]
8B	SimNPO	1	0	0.98	+4.73 [+4.34, +5.11]	+0.99 [+0.67, +1.30]
8B	SimNPO	2	0	0.93	+4.86 [+4.39, +5.31]	−1.21 [−1.73, −0.76]
8B	SimNPO	3	0	0.91	+4.48 [+4.01, +4.93]	−2.25 [−2.79, −1.77]
8B	SimNPO	3	12345	0.93	+5.02 [+4.56, +5.43]	−0.75 [−1.33, −0.23]
8B	SimNPO	5	0	0.89	+3.94 [+3.48, +4.38]	−2.98 [−3.50, −2.50]
8B	SimNPO	8	0	0.89	+3.58 [+3.13, +4.02]	−3.32 [−3.84, −2.83]
8B	WGA	1	0	0.89	+3.05 [+2.73, +3.36]	−0.40 [−0.63, −0.18]
8B	WGA	2	0	0.11	+0.21 [+0.05, +0.37]	−1.34 [−1.57, −1.10]
8B	WGA	2	12345	0.16	+0.27 [+0.11, +0.42]	−1.07 [−1.29, −0.84]
8B	SatImp†	3	0	0.09	+0.04 [+0.02, +0.06]	−0.07 [−0.24, +0.10]
8B	RMU†	3	0	0.16	+0.16 [−0.16, +0.48]	−7.29 [−7.58, −6.99]
8B	Retrained	–	–	0.11	0.00	0.00
8B	Original	–	–	1.00	+4.52 [+4.19, +4.85]	+2.05 [+1.81, +2.29]

Table 19. Healthy 8B SimNPO models recover the forgotten identifiers from the biography; healthy 1B models do not. Results on the 44 forgotten people. *Own-biography acc.*: 11-way accuracy given the person's own biography (chance 0.09). Δ gain (own biography minus a name-swapped biography) and Δ no-context: model minus the retrained model, in nats with 95% bootstrap intervals. Reference models in gray. †: fails the health checks (Table 20).

score is -0.09 $[-0.29, 0.12]$ nats from the retrained model's. It picks the true UUID 20% of the time given the biography (retrained 5%, original 100%) and gains $+0.31$ $[0.21, 0.40]$ nats more than the retrained model (original $+0.77$). The other three fail the check on other people's personal data. We leave these runs out of the tables.

Nationality. On the 8B models we also asked for the nationality of the 89 people inserted 256 times, as a 10-way choice (chance 0.10) with three prompts: no context (*"[name] is from"*), the true birthplace (*"[name] was born in [birthplace]. [name] is from"*) and another person's birthplace. The retrained model picks the true nationality 73% $[64, 82]$ of the time without context, from the name alone, and 98% $[94, 100]$ given the true birthplace; the original model scores 100% and 98% $[94, 100]$. Given a wrong birthplace, they keep the true nationality 56% and 70% of the time. Nationality is thus a fact the retrained model can infer from context as well as the original model can. We did not unlearn it.

F Extended related work

Unlearning methods and definitions. Unlearning for language models began with fine-tuning objectives that push the model away from the forget data (Jang et al., 2023; Eldan & Russinovich, 2023; Yao et al., 2024); Liu et al. (2025) survey the field, and our eleven methods descend from these objectives. Exact unlearning was defined against retraining from the start (Cao & Yang, 2015; Ginart et al., 2019), with certified variants for convex models (Guo et al., 2020). Closeness in weights to a retrained model cannot certify unlearning (Thudi et al., 2022), and exact equivalence to retraining may be out of reach for local updates (Yu et al., 2025); we therefore use the retrained model as a behavioral reference on the audit's own prompts. Context can also remove knowledge: in-context unlearning (Pawelczyk et al., 2024) and in-context editing (Zheng et al., 2023) change the answer through the prompt alone, the mirror image of the recovery we audit.

Size	Model	Epochs	Seed	Δ answer in prompt	Δ others' personal data	Collateral ratio
1B	NPO	1.5	0	+0.05 [+0.02, +0.08]	+0.07 [−0.02, +0.17]	1.01
1B	SimNPO	1	0	+0.08 [+0.05, +0.10]	+0.08 [+0.01, +0.16]	1.03
1B	SimNPO	1	99	+0.08 [+0.06, +0.10]	+0.04 [+0.00, +0.08]	1.01
1B	SimNPO	1	12345	+0.08 [+0.05, +0.10]	+0.04 [+0.00, +0.08]	0.93
1B	SimNPO	1.5	0	+0.07 [+0.05, +0.10]	+0.06 [−0.00, +0.15]	1.03
1B	SimNPO	1.5	99	+0.08 [+0.05, +0.10]	+0.09 [+0.03, +0.17]	1.01
1B	SatImp	1.5	0	+0.04 [+0.01, +0.06]	+0.10 [+0.01, +0.20]	1.03
1B	RMU†	3	0	−11.72 [−11.99, −11.45]	−0.02 [−0.08, +0.05]	1.02
8B	NPO†	3	0	−0.52 [−0.71, −0.35]	−0.88 [−1.08, −0.69]	0.94
8B	NPO†	3	12345	−0.40 [−0.53, −0.28]	−0.71 [−0.87, −0.56]	0.92
8B	SimNPO	1	0	−0.00 [−0.00, +0.00]	−0.02 [−0.03, −0.01]	0.44
8B	SimNPO	2	0	−0.01 [−0.02, +0.00]	−0.05 [−0.07, −0.03]	0.73
8B	SimNPO	3	0	−0.02 [−0.04, −0.00]	−0.06 [−0.10, −0.03]	0.80
8B	SimNPO	3	12345	+0.00 [−0.00, +0.01]	+0.02 [−0.01, +0.04]	0.60
8B	SimNPO	5	0	−0.04 [−0.06, −0.02]	−0.07 [−0.11, −0.03]	0.84
8B	SimNPO	8	0	−0.05 [−0.09, −0.03]	−0.07 [−0.12, −0.03]	0.84
8B	WGA	1	0	−0.04 [−0.05, −0.03]	−0.03 [−0.06, +0.01]	0.86
8B	WGA	2	0	−0.08 [−0.09, −0.07]	−0.08 [−0.14, −0.01]	0.97
8B	WGA	2	12345	−0.08 [−0.09, −0.06]	−0.05 [−0.11, +0.02]	0.99
8B	SatImp†	3	0	−0.17 [−0.19, −0.15]	+0.12 [−0.00, +0.25]	0.95
8B	RMU†	3	0	−11.58 [−11.82, −11.32]	−0.09 [−0.16, −0.03]	0.98

Table 20. Every unlearned Hubble model also lowers the identifiers of people it was not asked to forget. Health checks and collateral, model minus the original model, in nats with 95% bootstrap intervals. Δ answer in prompt: on the 44 forgotten people. Δ others' personal data: the four other fields of the 45 people who were not forgotten. *Collateral ratio*: the no-context drop for the 45 people who were not forgotten, divided by the drop for the 44 who were. †: fails the health checks (either Δ below −0.10).

Recovering what was unlearned. Besides prompts and light fine-tuning (Section 8), quantization (Zhang et al., 2025), weight and activation tampering (Che et al., 2025), retraining after concepts are pruned (Lo et al., 2024) and in-context examples (Xhonneux et al., 2024) bring unlearned knowledge back, and extraction from a set of candidate answers recovers facts that generation hides (Patil et al., 2024), as our multiple-choice question does. Access to the model's outputs before and after unlearning leaks the forget data even under exact unlearning (Wu et al., 2025), and sampling reveals more than greedy decoding (Scholten et al., 2025; Reisizadeh et al., 2026). Multiple-choice probes separate unlearning from obfuscation (Sun et al., 2025), and parametric probes look below behavior (Hong et al., 2025). Defenses target recovery through the weights (Tamirisa et al., 2025; Fan et al., 2025a); conflicting context is complementary, asking whether the model still overrides its input with the forgotten answer. Models pretrained without hazardous data still use it when the prompt supplies it (O'Brien et al., 2026), the pretraining-scale counterpart of our retrained models.

Benchmarks. TOFU, MUSE and Hubble supply a model that never saw the forget data (Maini et al., 2024; Shi et al., 2025; Wei et al., 2026); WMDP and RWKU (Li et al., 2024; Jin et al., 2024) unlearn knowledge from pretraining and have none. Hubble's random identifiers play the role of canaries (Carlini et al., 2019), and extraction attacks on pretrained models (Carlini et al., 2021) motivate unlearning personal data at all.

Knowledge conflicts. Xu et al. (2024) survey conflicts between context and stored knowledge. Models rely more on retrieved context for less popular facts (Mallen et al., 2023), and prompt framing changes how faithfully they follow it (Zhou et al., 2023); we therefore score every model, the retrained one included, on the same wording. Deployed models increasingly read retrieved text (Lewis et al., 2020), which is where related facts reach an unlearned model.

Composing facts. Models that learn "A is B" often fail to answer "B is A" (Berglund et al., 2024), answer single hops more reliably than their composition (Press et al., 2023), and resolve the first hop too late to use it for the second (Biran et al., 2024). Knowledge editing faces our question from the other side: an edit should propagate to the facts it implies (Zhong et al., 2023; Cohen et al., 2024).

Unlearning benchmarks count a target that can be re-derived from kept facts as leaked (Choi et al., 2024; Wu et al., 2024); a retrained model re-derives it too when the prompt supplies the link.

G REPRODUCIBILITY

Recorded results. For every synthetic model we record, per person and per prompt, the score of every candidate answer, from which the margins and strict-win accuracies follow, together with the greedy generation without context. For every Hubble model we record, per person and per field, the length-normalized log-probability of the true answer under each prompt and the 11-way choice.

Analysis. Every comparison with a retrained model uses these per-item records: differences of means with unpaired bootstrap intervals (Appendix A.7), and the sector-only fit of Appendix C.2. The figures and tables are drawn directly from the same records.

Seeds. Fact sets A, B and C use generation seeds 20260618, 20270115 and 20280115; the 10-way menus use seed 1234; each item's single-hop distractors use seed $1000 + i$. Training and unlearning use seed 0 unless a setting names another. All bootstrap intervals use seed 20260925 and 10,000 resamples.

G.1 EXPERIMENTS THAT WOULD STRENGTHEN THESE RESULTS

The following experiments would address the limitations in Section 9. We have not run them.

- Extend the wrong-employer evidence beyond RMU on Qwen3-4B and Llama: search Qwen2.5, OLMo and other methods on all fact sets for healthy settings that pass free generation and the no-context question, and test them with the wrong-employer prompt; re-score the Llama layer-20 settings with the full-sentence scorer.
- Forget only part of the person→sector relation, train the matching retrained model on the rest, and check whether that model still follows a wrong employer.
- Find a behavioral control that separates a preference about this person from a preference for sectors; naming the wrong employer through another person is not one (Appendix C.3).
- Retrain the healthy unlearned settings and score them with the 10-way wrong-employer prompt, free generation under a wrong employer and the employer→sector check, which we measured only on the reference models, the RMU reruns and the further RMU settings.
- Separate a stored target from a learned route that recalls the employer and then its sector, for example with an explicit two-step prompt or by forgetting person→employer before the wrong-employer test.
- Map how the number of RMU training steps trades off suppressing the question against keeping the target, over more seeds and starting models.
- Add two-sided tests that also flag a model that answers less than the retrained model, such as one that suppresses the question.
- Run unlearning on OLMo-2-7B, and score the retrained OLMo model on set C with the full-sentence scorer and the health checks.
- Measure MMLU for all retrained, original and healthy unlearned models.
- Unlearn nationality in Hubble and test it with a wrong birthplace, and repeat the healthy 100B-token run over more seeds.
- Find healthy settings for RMU and SatImp on the 8B Hubble model.