Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Model That Does Not Forget. A Logbook That Corrects and a Sleep That Consolidates, on a Frozen Language Model

Maximiliano Rodrigo Speranza

maximiliano.speranza@gmail.com

Independent Researcher https://orcid.org/0009-0005-0413-8554

Research Article

Keywords: continual learning, catastrophic forgetting, memory consolidation, knowledge editing, replay, abstention, LoRA, retrieval-augmented generation, language models, pre-registration

Posted Date: October 8th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11185020/v2

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

A Model That Does Not Forget

A logbook that corrects and a sleep that consolidates, on a frozen language model

Maximiliano Speranza
Independent Researcher, Buenos Aires, Argentina
ORCID 0009-0005-0413-8554
maximiliano.speranza@gmail.com

October 2026 (version 2)

Abstract

An assistant used every day needs to remember what the user told it, even when those facts change over time, and it needs to say "I don't know" about what it was never told. We study two memory channels on a frozen language model, Qwen3-0.6B. The *logbook* stores each fact with its version outside the weights and is read by a learned reader. The *sleep* periodically consolidates recent facts into a LoRA adapter with 10 million parameters, using a recipe that replays the facts already consolidated, replays two invented skills together with their abstention boundaries, practices "I don't know" and anchors to the model each sleep starts from.

Over eight chained sleeps of 100 facts each, with three seeds, what was learned in the first week holds at an accuracy between 0.88 and 0.97 after seven more sleeps (mean 0.94), and no package falls more than 0.09 below the value it was learned at. An ablation isolates the ingredient. Without fact replay, everything else equal, accuracy on the first week goes from 0.97 to 0.01 in a single sleep. We compare with MEMIT and AlphaEdit, two published knowledge editing methods that write each new fact directly into the model weights, without training it. Editing the same facts week after week, by the eighth week they keep 0.01 and 0.10 of what was learned in the first, and both damage skills the editing never touched. Against a RAG trained with the same budget the logbook ties in accuracy, and the difference is cost, 16 tokens per query against 505, or against 4,514 with a history of 200 old entries.

The limit we measure is spurious composition. Given a known entity and a relation never taught for that entity, by the eighth week the model answers a value in 51 to 61 % of cases instead of abstaining, and "I don't know" for invented entities drops from 0.97 to 0.64. Evening out how many facts each entity receives, the most obvious intervention, changed nothing (40 answers out of 200 in both conditions). The limit is cut outside the weights. A gatekeeper that looks up the question's entity in the logbook by identity and lets the logbook's reader decide the relation, with the entity replaced by one it knows, lowers composition to between 0.000 and 0.007 while letting through between 0.983 and 1.000 of the stated facts, with three reader seeds, the threshold set on the other chain, and nothing newly trained.

With real-world facts memory holds. In a chain of sleeps over CounterFact, the public benchmark used to evaluate knowledge editing, each week 100 facts the model already believed are corrected. By the fourth week what was corrected in the first still wins by probability in 0.99 to 1.00 of the cases, over three seeds, and without replay the exact accuracy on those facts falls to 0.17 to 0.35. But perplexity on text written by people, which the measure of version 1 (the model's own sentences) did not see, rises from the first sleep and grows, up to a 55-fold increase in English by the eighth week. We found three causes by intervention and a final recipe, a sleep anchored to the origin that teaches only the answer, ties everything else to the original model and adds human text, restatements written by version 0 itself, anchored other subjects and a floor for the new. By the eighth week, over three seeds, what was corrected in the first week still wins in 1.00 of the cases, paraphrases of what was corrected in 0.95 to 0.965, and the damage measured with calibrated perplexity

(each model at its own optimal temperature, because at temperature 1 the base model is overconfident and the damage hides) over 150 distinct Wikipedia articles stays between +1.1 and +1.5 % in English and between −0.48 and −0.95 % in Spanish. That remainder grows slowly with the weeks and sits in proper names. An adapter gatekeeper, which uses what was consolidated only when the text names a learned subject and otherwise lets the original model speak, brings it to between +0.26 and +0.29 % in English without touching memory. With a list of 18,466 subjects the paragraph gatekeeper gets diluted, and restricting it to the eight tokens that follow the mention brings it back to between +0.08 and +0.19 %. A preregistered ablation showed that the 2,000-fact world anchor of an earlier recipe was unnecessary, and the final recipe is the same without it. It costs specificity, since the neighbors drop from 0.70 to 0.60 to 0.62.

In a simulated two-week chat with three users, the model that slept with the final recipe recalls the next day 0.918 of what it was told, asked with phrasings it never saw, against 0.062 for the same model with the whole chat in its context, and its human text at temperature 1 does not get worse. When the user corrects something the new value always wins. Where it was never told anything, however, it answers with the value of another fact that was stated in 33 of 60 questions, and verifying against the logbook the value the weights propose raises abstention from 0.083 to 0.450.

What is new and corrected in this version (version 2 on Research Square). Version 1 measured language with perplexity on sentences generated by the model itself and concluded that the sleep does not worsen it. On text written by people the damage reached +1,270 % in English by the fourth week (Section 4.9). This version corrects that claim and adds (1) a chain of sleeps over real CounterFact facts, with three seeds (Section 4.8), (2) the three causes of the damage, found by intervention, and the recipes that remove it, measured with calibrated perplexity (Sections 4.9 and 4.10), (3) eight weeks over three seeds, with a prediction written before that failed for the recipe with the world anchor, and a third measurement trap, a human-text register taken from a single article, replaced by 150 distinct articles (Section 4.11), (4) the adapter gatekeeper and its scaling limit (Section 4.12), (5) a preregistered ablation that removed the world-facts anchor and gave the final recipe (Section 4.13), (6) the same subject with another relation (Section 4.14), (7) a simulated two-week chat (Section 4.15) and (8) an account of what failed along the way, with all the variants tried (Section 5 and Appendix A).

1 Introduction

Almost all work on memory in language models targets knowledge about the world, either teaching the model a new fact (knowledge editing) or letting it read external documents (retrieval, RAG). An assistant used daily needs something else as well, which is remembering the interaction itself. What the user said, when, and what they had said before if that changed. This memory has a property world knowledge has little of, which is that it gets corrected often. “My dog is called Toby” may become “I have two dogs, Toby and Luna” the following week, and the system has to know which version holds today without confusing them or forgetting that the first one existed.

The available tools cover parts of this problem. Knowledge editing writes a fact directly into the model weights, without retraining it. The two reference methods are MEMIT (Meng et al., 2023), which locates the intermediate layers where the model stores associations between a subject and its attributes and modifies them with a computed update, without training, for many facts at once, and AlphaEdit (Fang et al., 2025), a variant designed for editing facts in sequence that restricts each change to directions that do not alter the answers about what should be preserved. These methods are usually evaluated on two public benchmarks of real-world facts. CounterFact (Meng et al., 2022) gathers about 22,000 facts as sentences to complete (“The mother tongue of Danielle Darrieux is”), and zsRE (Levy et al., 2017; De Cao et al., 2021) gathers questions written by people (“Who is Adoration of the Trinity by?”). Knowledge editing

overwrites the fact without keeping what it said before and, as we show, accumulates damage when edits are chained. RAG reads the history as text, but its cost grows with the length of the history at every query. Direct fine-tuning learns the new by overwriting the old. This work asks whether a frozen model can remember hundreds of versioned facts from an interaction, correct them without retraining, say "I don't know" about what it never heard, and sustain all of that session after session with a reading cost that does not depend on the length of the history.

We propose splitting the problem into two channels. The *logbook* solves correction. It is memory outside the weights, fast to write and to read, that stores each fact with its version. The *sleep* solves permanence. It is a periodic consolidation that moves recent facts into the weights of an adapter, so that what was already learned does not depend forever on the logbook still reading it. The central question is what each sleep needs so that the model does not forget what it consolidated while it keeps learning new facts.

The contributions are the following.

1. A consolidation recipe that, over eight chained sleeps and three seeds, holds what was learned in the first week between 0.88 and 0.97, with no package falling more than 0.09 below its own initial value (§4.1).

2. An ablation that identifies the ingredient. It is the explicit replay of consolidated facts, and not the anchor to the previous model nor the logbook, that prevents forgetting (§4.2).

3. A comparison with two published knowledge editors, MEMIT and AlphaEdit, on the same facts and the same starting point, with the mechanism by which AlphaEdit damages capabilities the editing never touched verified by intervention (§4.3).

4. A measurement of abstention separated by question class, showing that "I don't know" holds when everything is unknown and erodes when the question combines known parts (§4.4).

5. A chain of sleeps over real facts, on CounterFact and with its standard metrics, that reproduces the retention, the ablation and the gatekeeper of the synthetic world over three seeds and eight weeks (Section 4.8).

6. A correction of version 1 and its solution. Measured on human text, the version 1 recipe damages language from the first sleep and the damage grows (+246 % in the first, +1,270 % by the fourth week). We identify by intervention three causes, a replay written by a model that has already drifted, a replay in a single language and an objective that teaches the whole sentence instead of the answer, and a final recipe that by the eighth week, over three seeds, leaves memory intact, the paraphrase of what was corrected at 0.950 to 0.965 and the calibrated damage over 150 Wikipedia articles between +1.1 and +1.5 % in English and below version 0 in Spanish. The central loss, cross entropy only on the answer with the rest tied to version 0, is used by KLOD (Jeong et al., 2026) for one-shot edits, and we do not claim it as ours (Sections 4.9, 4.10 and 4.11).

7. An adapter gatekeeper that leaves text not naming what was learned to the original model and brings the damage to English to between +0.26 and +0.29 % without touching memory, its dilution with tens of thousands of subjects and an eight-token window that solves it (Section 4.12).

8. A preregistered ablation of the recipe by families of pieces, which removes the world anchor and shows what each family protects, the text anchors language and the subject anchors generalization (Section 4.13).

9. The case of the same subject with another relation over real facts, where the sleep contaminates the other relations of the corrected subject and three attempts to separate them hit the same limit, identifying the relation (Section 4.14).

10. A simulated two-week chat with a user, which measures what the goal asks for, recalling what was said, asked the next day and without the context, with four sleep recipes, and the cost that remains with the final one, which answers with the value of another stated fact where nothing was ever said (Section 4.15).
11. A gatekeeper that cuts composition outside the weights, separating the entity, looked up in the logbook by identity, from the relation, decided by the logbook's reader (§4.5).
12. A record of the predictions written before each run, with those that held and those that failed, and an intervention with a null result reported as such (§4.16).
13. Three measurement traps, the model's own sentences, uncalibrated perplexity and a human-text register with a single topic, with the control that detects them, and an account of what failed, with all the variants tried and those added after seeing results (Section 5 and Appendix A).

2 Related work

Sleep-like consolidation. SEAL (Zweiger et al., 2025) trains the model, with reinforcement learning, to write its own weight update instructions, and reports that performance on earlier tasks gradually declines as edits accumulate. Behrouz et al. (2026) propose a two-stage sleep phase, a distillation of the memories of a small model into a larger one and a "dreaming" stage in which the model generates its own practice data through reinforcement learning. Lee et al. (2026) consolidate recent context into fast weights through offline recurrent passes, in architectures that combine state space blocks with attention. SCoL (Wang et al., 2026) uses meta reinforcement learning so that the model decides which layers to update from the context. Our work differs in its object, versioned facts from an interaction with explicit corrections, and in that consolidation is paired with a separate channel, the logbook, for what has not been consolidated yet.

Knowledge editing. MEMIT (Meng et al., 2023) edits thousands of associations at once by modifying the output projections of several intermediate layers. AlphaEdit (Fang et al., 2025) projects the perturbation onto the null space of a covariance of the knowledge to preserve, so as not to alter the outputs on that knowledge, and is designed for sequential editing. WISE (Wang et al., 2024) separates a main memory from a side memory of edits with a router, and Larimar (Das et al., 2024) adds a distributed episodic memory written in one step. We compare directly against MEMIT and AlphaEdit (§4.3). KLOD (Jeong et al., 2026) edits with cross entropy only on the target and a KL against the original model at the prefix positions, which is in essence the loss of our sleep anchored to the origin (Section 4.9). It uses it for 3,000 edits optimized together in a single layer, without chaining weeks, without replay and without measuring perplexity on human text. We do not claim that loss as ours.

Explicit memory with a scope. SERAC (Mitchell et al., 2022) stores edits in an explicit memory and trains a scope classifier that decides whether an input falls within any edit before answering with a counterfactual model. GRACE (Hartvigsen et al., 2023) keeps a codebook at one layer with a radius that decides when each edit applies, and RRDA (Zhang and Huang, 2026) uses a relevance router between an adapter that writes the edit and one that preserves the original behavior. MemLLM (Modarressi et al., 2024) stores knowledge as subject, relation and object triples that the model queries explicitly. The gatekeeper of Section 4.5 applies that general idea to a different problem, the spurious composition that consolidation creates in the weights, and solves it by separating the entity from the relation. The hippocampal memory indexing theory (Teyler and DiScenna, 1986) proposes a similar division of labor in the brain,

with the hippocampus keeping an index of what was experienced and the cortex keeping the content.

Facts in the weights, continually. O'Neill et al. (2026) write invented facts into Qwen3 one at a time, with a rank 16 LoRA merged into the weights before the next one, and without replaying earlier ones. After twenty writes the model answers 1 % of the questions about earlier facts correctly when each write repeats the statement, and 46 % when it uses varied restatements. Our ablation reproduces that collapse when replay is removed (Section 4.2), and with replay avoids it. Restatements as augmentation come from Allen-Zhu and Li (2024), who show that without them a memorized fact cannot be extracted, and from O'Neill et al. (2026). Our self-paraphrases (Section 4.10) belong to that family.

Memory of a personal assistant. Zhu et al. (2026) show that an assistant that compresses the conversation in its cache remembers the user's allergy 0 to 1 % of the time (97 % with the whole conversation), and conclude that a compressed cache is not a persistence layer. HypReflect (Hwang et al., 2026) personalizes a model continually by distilling hypotheses about the user's preferences, and SleepGate (Xie, 2026) resolves the interference of old values in the attention cache with a conflict tagger and a forgetting gate. None of the three consolidates into the weights what the user stated, with its corrections. The two closest works we found in the literature check of October 6 are from 2026. Dennis et al. (2026) consolidate nightly into a LoRA adapter the knowledge from software development conversations, with reflection, data synthesis and fine-tuning, and retain 80.4 % of the knowledge against 36.8 % for three cycles of cascading context compaction. Li (2026) stores each user's facts as local edits in the memory table of an Engram model, which switch on only at the trigger and leave the rest of the text intact, the same idea the adapter gatekeeper (Section 4.12) brings to a LoRA through the logbook. Choi (2026) show that in continual knowledge updating the ranking between methods changes with the evaluation time and the adapter capacity, which is why we report eight-week trajectories and not only the endpoint. Putting the fact in the context, as the logbook line does in the chat, has its antecedent in IKE (Zheng et al., 2023), and DiSC (Padmanabhan et al., 2026) updates parametric knowledge by distilling the context with a KL on the shared tokens to retain what post-training taught, a trade-off close to that of the anchored sleep.

New knowledge and hallucination. Gekhman et al. (2024) show that, when a model is fine-tuned on new facts, those examples are learned more slowly than the ones consistent with what the model already knows and that, as they are learned, the tendency to hallucinate increases linearly. The spurious composition we measure is a phenomenon of the same kind, in a different regime (an adapter with explicit replay instead of direct fine-tuning).

Versioned memory in the author's previous work. The logbook continues a line of work on versioned memory inside the model, developed in small models trained from scratch, which studied geometric indexing (Speranza, 2026a), where abstention is best placed (Speranza, 2026b), an order stamp that resolves conflicts between versions (Speranza, 2026c) and the reach of the query in retrieval (Speranza, 2026d). This work takes that memory to a pretrained model and adds consolidation.

3 Method

3.1 The model and the synthetic world

We use Qwen3-0.6B (Yang et al., 2025) with all its parameters frozen in every experiment. Facts come from a synthetic language in Spanish with 102 entities and 24 relations (2,448 possible

Table 1. The sleep recipe. Probability that a batch sample comes from each source.

source	week 1	weeks 2 to 8
own sentences generated by the model before sleeping (replay)	0.30	0.20
facts from earlier packages, with their original value	none	0.15
the two skills, with their abstention examples	0.15 or 0.25	0.15 or 0.25
"I don't know" for known combinations (70 %) and invented entities or relations (30 %)	0.20	0.20
new facts with a newer version read from the logbook	0.15	0.15
new facts of the package (statement or question)	the rest	the rest

pairs). Each fact is a triple of relation, entity and value, with a natural statement ("el gasto de finca es 64", the cost of the farm is 64) and a direct question ("¿cuál es el gasto de finca?"). Because entities and values are invented, the model cannot know them in advance, so every correct answer is due to the memory we measure.

3.2 Channel 1, the logbook

The logbook writes one key per entry, from the full statement with the value masked, at layer 6 of the model, and reads it at the output with a learned reader. With a key derived from the value, in an earlier version, global accuracy stayed between 0.61 and 0.65. With the masked statement key, in a preregistered study with three seeds, the frozen model with the logbook reaches a global accuracy of 0.900 to 0.938, picks the correct entry among several versions of the same fact 0.920 to 0.955 of the time and says "I don't know" about absent facts between 0.947 and 0.989 of the time, with a false abstention rate of 0.000 to 0.009. A reader trained with 500 unrelated entries in the archive holds between 0.932 and 0.945 with those 500 entries present, and one trained and evaluated with 3,000 entries reaches 0.900 (one seed). At reading time the model processes only the question, about 16 tokens, and the archive is queried outside the context.

3.3 Channel 2, the sleep

Each sleep trains a LoRA adapter (Hu et al., 2022) of rank 16 on the seven projections of the 28 layers (10.09 million parameters), for 6,000 steps with batches of 8 and learning rate $2 \cdot 10^{-4}$. Each sample of the batch is drawn from the sources in Table 1. On the replay samples (own sentences, earlier facts and skills) a KL anchor of weight 0.3 is added against the model the sleep starts from, and the loss is averaged per example rather than per token, so that the long examples of a skill do not dominate the short abstention examples. Each sleep starts from the adapter left by the previous one. This is the version 1 recipe, with which the synthetic world and the comparisons of the following sections are made. The recipe for real facts, which changes the objective and the anchors, is described in Section 4.10, and the final one, which is the same without the world anchor, in Section 4.13.

Skill replay was 0.25 in seeds 2 and 3 from the first week. In seed 1 it was 0.15 in weeks 1 to 3 and 0.25 from week 4 on, after week 1 showed the code skill at 0.44 with 0.15. This is a protocol deviation we declare here, and also the reason why seed 1 devotes more samples to new facts in its first weeks. With a skill replay of 0.15 direct new facts are between 15 and 20 % of the samples, and with 0.25 between 5 and 10 %.

3.4 Two invented skills

To know whether a consolidation breaks capabilities it did not touch, one needs capabilities the model cannot have from pretraining. Otherwise "it holds" and "it never broke" would be

indistinguishable. We use two. "Nudo" is a toy programming language translated into Python, and $\diamond$ is an invented arithmetic operator, $a \diamond b = 2a + b$. Without training the model does not solve them (0.00 in both), and explaining the rule of $\diamond$ in the prompt does not suffice either (0.00). One sleep of 1,500 steps takes Nudo from 0.00 to 1.00 on 100 unseen programs, and $\diamond$ from 0.00 to 0.88 (0.87 with another seed), or to 0.99 if the procedure is written step by step. Each skill has an abstention boundary, saying "I don't know" about an operator it never saw ($\star$, which shares two of its three tokens with $\diamond$) and about a Nudo word that does not exist. The boundary of $\diamond$ was lost completely (0.00) when the next sleep replayed only the positive examples of the skill, and held at 1.00 when it also replayed the abstention examples. All the chains in this work start from the same adapter with both skills already learned.

3.5 Evaluation

After each sleep we measure the following, with greedy generation and exact match.

- **The new**, the 100 facts of the package just slept.
- **The old**, each earlier package separately. First week retention is the accuracy on package 1.
- **The logbook wins**, 50 facts of the package with a new version in the logbook. It is read with a reader from an earlier stage over a one-entry archive, so it measures whether a logbook read overrides what is consolidated in the weights, not retrieval accuracy.
- **Composition**, 50 pairs of an entity with consolidated facts and a relation it never had, where the correct answer is "I don't know". We report the fraction that answers a value.
- **Abstention by class**, 40 questions per class with an invented entity, an invented relation or both, all different from those used to practice "I don't know".
- **Skills**, Nudo, $\diamond$ (with the final computation done by an external calculator, to separate the rule from the sum) and their two abstention boundaries.

With 100 questions the standard error of an accuracy near 0.9 is about 0.03, and with 50 about 0.06. Because composition measured on 50 pairs is noisy and those pairs may coincide with combinations some sleep practiced as "I don't know", in seeds 2 and 3 we also reserved 300 pairs before the first sleep, excluded from every package, control and abstention practice during the eight weeks. Clean composition is measured on them.

3.6 Baselines and ablations

- **Naive**. The same fact packages, without anchor, without any replay, without "I don't know" and without logbook.
- **No fact replay**. The seed 1 recipe without the source of earlier facts. Its first week is the same run as the first week of seed 1.
- **MEMIT and AlphaEdit**. The same weekly packages and the same starting point, edited in sequence on the output projections of layers 4 to 8. We use $\lambda = 15{,}000$ and a covariance estimated from 3,000 sentences generated by the model itself, and for AlphaEdit the null space threshold $2 \cdot 10^{-2}$ and the regularization 10 of the original. Two differences with the originals are declared. The covariance comes from the model's own sentences rather than Wikipedia, and evaluation uses the same template as the edit, which favors the editors. Since these methods have no logbook, we measure instead correction by editing, which edits 50 facts with a new value, measures and undoes.

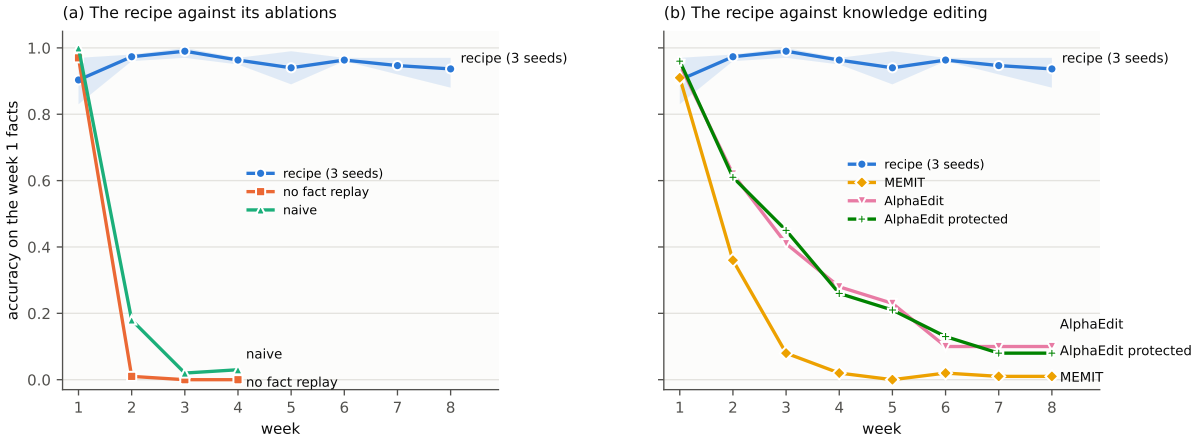


Figure 1. Accuracy on the 100 facts of the first week over eight weeks. In week 1 each curve shows what was just learned. The band is the range of the three seeds of the recipe. (a) Without fact replay the first week disappears in one sleep. (b) Sequential editing with MEMIT and AlphaEdit, on the same facts, loses it gradually.

Table 2. Accuracy on the first week facts at the end of each week, by seed. Week 1 is what was just learned.

week	1	2	3	4	5	6	7	8
seed 1	0.97	0.98	1.00	0.97	0.99	0.97	0.97	0.97
seed 2	0.83	0.96	1.00	0.97	0.94	0.96	0.95	0.88
seed 3	0.91	0.98	0.97	0.95	0.89	0.96	0.92	0.96
mean	0.90	0.97	0.99	0.96	0.94	0.96	0.95	0.94

- **Balanced sampling**. The seed 1 recipe over four weeks, but choosing which entities receive new facts with weight $1/(1+c)^{1.5}$, where c is how many facts the entity already has, using the weighted sampling without replacement of Efraimidis and Spirakis (2006).

3.7 Prior registration

Before each run we wrote the numerical prediction and the decision rule into the project repository, with the commit date as the ordering mark. Section 4.16 contrasts them with the results. Those of the gatekeeper (Section 4.5) are in a file of their own, step by step, including the steps that failed, and those of the chain over real facts in another.

4 Results

4.1 Eight sleeps in a row do not erase the old

Table 2 and Figure 1a show accuracy on the first week package at the end of each of the eight weeks. By the eighth week it stands at 0.97, 0.88 and 0.96 depending on the seed. Seed 2 has its worst value right at the end, so we do not claim the absence of a trend, but the drop is small. Another way to measure it is to compare each package with the value it was learned at in the week it was slept. None falls more than 0.09 below that value in any seed (0.09, 0.01 and 0.02 as worst case), and several improve in later sleeps. The week 7 package of seed 1, for example, was learned at 0.78 and was at 0.89 the following week.

The other measurements agree. Each week's new facts are learned between 0.77 and 0.97, lower in seeds 2 and 3, which devote fewer samples to new facts. The worst earlier package in any week stands at 0.86, 0.85 and 0.81 depending on the seed. In seeds 2 and 3 it is a package

Table 3. The recipe against its ablations over the same four weeks (seed 1). Composition is the fraction that answers a value where "I don't know" is correct (lower is better).

	week 1 retained in							"I don't know"
	week 2	week 3	week 4	logbook	composition	⋄	Nudo	new entity
full recipe	0.98	1.00	0.97	0.96	0.34	1.00	1.00	0.90
no fact replay	0.01	0.00	0.00	1.00	0.12	1.00	1.00	0.97
naive	0.18	0.02	0.03	0.24	1.00	0.01	0.00	0.00

Table 4. First week retention with the recipe (seed 1) and with sequential editing of the same 100 facts per week, and the state of the skills at the eighth week.

	week 1 retained in week							at week 8			
method	2	3	4	5	6	7	8	⋄	Nudo	"I don't know" ⋆	"I don't know" Nudo
recipe	0.98	1.00	0.97	0.99	0.97	0.97	0.97	0.88	0.97	1.00	1.00
MEMIT	0.36	0.08	0.02	0.00	0.02	0.01	0.01	1.00	0.00	0.90	0.00
AlphaEdit	0.62	0.41	0.28	0.23	0.10	0.10	0.10	0.00	0.00	0.00	0.00
AlphaEdit protected	0.61	0.45	0.26	0.21	0.13	0.08	0.08	0.97	0.66	0.65	0.07

that had already been learned low (0.81 and 0.77) and that rose afterwards. In seed 1 it is the week 6 package, which dropped from 0.95 to 0.86 when the seventh week was slept and was back at 0.95 in the eighth. The logbook overrides what is consolidated between 0.94 and 1.00 of the time across the twenty four evaluations. Skills hold at the eighth week between 0.88 and 1.00, with a first erosion of ⋄ in seed 1 (0.88), and both abstention boundaries remain at 1.00 in all three seeds.

4.2 What happens without the recipe

With the anchor, the logbook, the "I don't know" practice and skill replay intact, removing only fact replay takes the first week from 0.97 to 0.01 in one sleep and to 0.00 in the next two (Table 3, Figure 1a). The logbook and the skills, which are still replayed, do not degrade. The anchor to the previous model, which still acts on own sentences and skills, is not enough to protect facts that are not replayed. They have to be replayed. Composition is even lower without replay (0.12 against 0.34), which is consistent with it depending on how many facts the model still knows. The naive line collapses in everything at once. It loses the old, loses Nudo from the first week and ⋄ from the second, and from the first week it stops saying "I don't know" in every class and answers a value for any combination.

4.3 Against knowledge editing

Both editors learn each week's new facts well (between 0.84 and 0.97) and correct a fact by editing between 0.80 and 1.00 of the time, but they lose what was edited before (Table 4, Figure 1b). By the eighth week MEMIT retains 0.01 of the first week and AlphaEdit 0.10. Tuning λ does not resolve the trade-off. With $\lambda = 40{,}000$ MEMIT edits only 0.55 of the new facts and retains 0.18 of the first week at the second, less than with 15,000, and with 150,000 it edits 0.07. Neither has a notion of "I don't know", and they answer a value in 98 to 100 % of the combinations without a fact.

The editors also damage capabilities the editing never touched. With MEMIT, Nudo falls to 0.00 from the fifth week on. With AlphaEdit the damage starts earlier, Nudo at 0.25 in the second week and 0.00 from the fourth, and ⋄ at 0.00 from the sixth. AlphaEdit can only edit within the null space of the covariance it estimates, and that covariance comes from prose sentences. We measured how much energy the layer inputs corresponding to each kind of text place in that null space, with the same covariance as the experiment (2,798 sentences). In layers

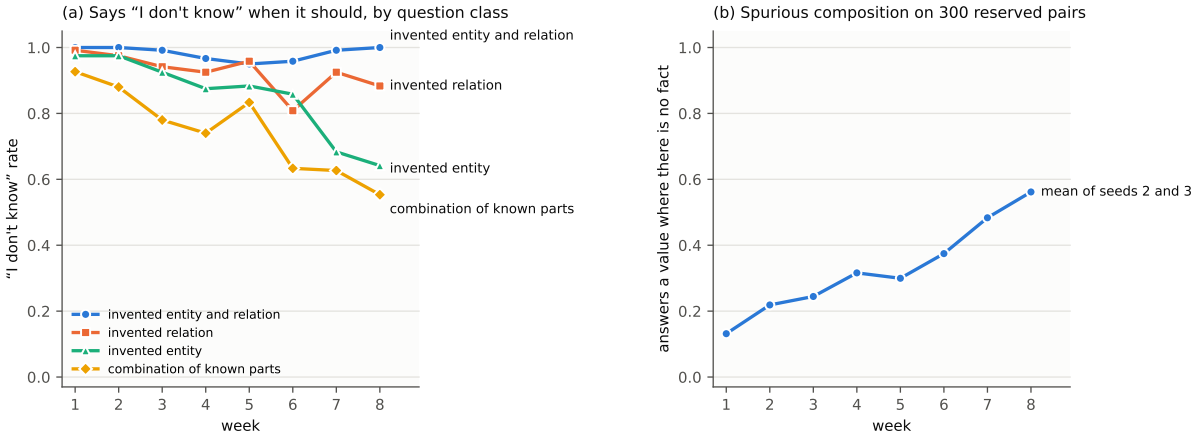


Figure 2. (a) Rate of “I don’t know” by question class, mean of the three seeds. With an invented entity and an invented relation abstention holds. With an invented entity and a known relation, or with a known entity and a known relation never combined, it erodes. (b) Fraction of the 300 reserved pairs in which the model answers a value instead of “I don’t know”, mean of seeds 2 and 3.

4 to 6 those of Nudo programs place between 41 and 48 %, those of ⋄ between 18 and 22 % and those of new prose between 10 and 12 %, and in layers 7 and 8 the order is the same with smaller values. AlphaEdit edits exactly where the skills live, and the order matches the order of the damage, first Nudo and then ⋄.

To test that this is the mechanism we repeated the chain adding to the covariance the 200 training examples of each skill (AlphaEdit protected). Nudo goes from 0.25 to 1.00 in the second week, and by the eighth week ⋄ is still at 0.97 and Nudo at 0.66. Fact retention does not change (0.61, 0.45, 0.26, 0.21, 0.13, 0.08 and 0.08 against 0.62, 0.41, 0.28, 0.23, 0.10, 0.10 and 0.10), so the protection covers what is put into the covariance and nothing else. What does fall under protection are the two abstention boundaries, the “I don’t know” of Nudo at 0.07 and that of ⋆ at 0.65. The protected covariance included only the positive examples of each skill, not the abstention ones, and the result repeats what we saw with the sleep in Section 3.4. Protecting what the model knows how to do does not protect what it knows it does not know.

4.4 Abstention by class and spurious composition

Abstention does not degrade equally in every class (Figure 2a). With the entity and the relation invented it holds at 1.00 from start to end. With an invented relation it drops from 0.99 to 0.88. With an invented entity and a known relation it drops from 0.97 to 0.64 (0.70, 0.53 and 0.70 depending on the seed at the eighth week). And for a known entity and a known relation never combined it drops from 0.93 to 0.55. The pattern suggests that the model decides whether to answer by checking separately whether it knows the relation and whether it knows the entity, and that with enough accumulated facts recognizing one of the two parts is enough for it to answer. Going through the saved answers of an earlier run, with a smaller world, in 43 of 43 composition errors the entity and the relation asked about appeared separately among the consolidated facts, and the value given was that of another fact.

On the 300 reserved pairs, which no sleep ever touched, composition grows from 0.06 to 0.61 in seed 2 and from 0.21 to 0.51 in seed 3 (Figure 2b). This is the main limit of this work. In seed 1, with 50 pairs per week, the series is 0.08, 0.10, 0.28, 0.34, 0.22, 0.48, 0.24 and 0.44.

Within each week, entities with more accumulated facts compose more. Over the first four weeks of seed 1 (200 questions) the correlation between facts per entity and answering a value is $r = 0.37$ ($p = 6 \cdot 10^{-8}$), 0.34 against 0.09 between entities with more than two facts and the rest, and within each week r ranges from 0.12 to 0.34. To see whether the relation is causal we ran balanced sampling, which spreads new facts among the entities with fewer facts. Adding up

Table 5. The gatekeeper at the eighth week of seeds 2 and 3, over the 300 reserved pairs and the 800 stated facts. "Pass" is the fraction of stated facts the gatekeeper lets the weights answer. "Compose" is the fraction of reserved pairs on which the system answers a value. The area is the one under the curve separating facts from compositions by the margin between the best entry and the null slot. Except in the last row, the reader is the preregistered seed.

	facts that pass	compose	area
weights alone, no gatekeeper	all	0.613 and 0.510	
the logbook's reader as is	0.31 and 0.29	0.013 and 0.010	0.65 (0.995 seen, 0.52 new)
untrained, means of the frozen model			0.58 and 0.59
untrained, token against token			0.81 and 0.80
entity by identity and relation by the reader	0.948 and 0.965	0.000 and 0.000	1.000
same, threshold set on the other chain, three seeds	0.983 to 1.000	0.000 to 0.007	1.000

the four weeks, the model answers a value in 40 of 200 questions with uniform sampling and in 40 of 200 with balanced sampling. The correlation is real and the intervention does not confirm it as a cause. Note that the two chains use different seeds. In an exploratory comparison, with 3.5 times more possible pairs, composition at 400 facts was the same (0.34 against 0.32 and 0.34 in a world of 696 pairs, with 50 questions each and recipes that are not identical), which does not point to the size of the world either.

4.5 The gatekeeper, composition is cut outside the weights

The weights cannot know what they were not told, but the logbook can, because by construction it stores everything that was said. The gatekeeper uses that. Before answering, the logbook decides whether the question's pair exists, and only if it does do the weights answer. Otherwise the answer is "I don't know". Nothing is trained. The weights' answers are the ones each evaluation of seeds 2 and 3 already saved, and the logbook of the eighth week holds the chain's 800 facts, one per entry. Each step had its prediction and rule written before running it (Table 5).

With the logbook's reader as is, composition drops from 0.613 and 0.510 to 0.013 and 0.010, and the logbook of the other seed, used as a foreign logbook, lets through only 0.046 and 0.028 of the facts it does not hold. But of the facts that were stated only 0.31 and 0.29 pass, and by the written rule that is not a solution. The diagnosis explains it. With the 29 entities the reader was trained on, facts pass between 0.97 and 0.98 and the area is 0.995. With the 73 entities the reader never saw the area is 0.52, chance. The reader learned to recognize its entities, not to compare any entity.

Nothing untrained is enough either. Comparing the question rewritten as a statement against the entries, by cosine over the mean of the frozen model's states and without letting the question use the entry's template, the area is 0.58 and 0.59. Comparing token against token, with ColBERT's late interaction (Khattab and Zaharia, 2020), it rises to 0.81 and 0.80 at the embeddings and falls in later layers. The breakdown separates the two pieces. The entity matches by itself through identity, equally with seen and new entities, and what fails is the relation across different templates ("la altura de bahía" against "bahía mide").

The final gatekeeper separates those pieces. The entity is an open set and is looked up by identity. The candidates are the logbook entries that name the same entity as the question, which assumes that the entity of each entry is marked when it is written, as any knowledge base does with an entity recognizer. The relation is a closed set the reader already distinguishes. In the question and in the candidates the entity is replaced by one the reader knows, and the reader decides whether some candidate has the asked relation. With the preregistered reader seed, 0.948 and 0.965 of the facts pass (0.944 and 0.965 with the new entities), composition stays at 0.000 in both chains and the area is 1.000. The foreign logbook lets through 0.000 of what it does not hold and 0.96 of what it holds. With another replacement entity the result repeats (0.989 and 0.985 of the facts, composition 0.000), and the template that expresses the relation

Table 6. The logbook against RAG on the same question bank. Accuracy by free generation. The logbook has three seeds and 128 questions per seed. RAG has one seed, and with 200 old entries its bank is 64 questions, with fewer than ten in some classes.

	accuracy	"what was it before"	"I don't know" for a relation	tokens per query	n
logbook, short history	0.953 to 0.984	0.95 to 1.00	1.00	16	3×128
logbook, with 500 old entries	0.938 to 0.961	0.95 to 1.00	1.00	16	3×128
trained RAG, short history	0.943	0.935	0.77	505	512
trained RAG, with 200 old entries	0.953	0.67	1.00	4,514	64
untrained RAG, short history	0.334	0.065	0.35	505	512
untrained RAG, with 200 old entries	0.109	0.33	0.00	4,514	64

with a verb, without the question's word, passes between 0.94 and 1.00, so it is not word overlap.

With the other two reader seeds the area stays at 1.000, but the cut given by the null slot moves. With one it is strict and lets through between 0.62 and 0.64 of the facts, and with the other it is lax and lets 0.027 compose. It is calibration, not separation. Setting the margin threshold on one chain and applying it to the other, in the three seeds and both directions, between 0.983 and 1.000 of the facts pass and composition stays between 0.000 and 0.007.

The scope has to be read together with the number. With the entity replaced and the value masked, the decision depends only on the asked relation and on the relations and templates of the candidates, and those four templates are the ones the reader was trained with. What is shown to generalize is the new entity, not a new way of saying a relation. And the gatekeeper cuts composition because it does not let the weights answer when the logbook lacks the pair, so the logbook has to be kept, even if only as an index of what was said. With real facts the same principle repairs the damage the sleep does to other subjects (Section 4.8).

4.6 The logbook against RAG

A RAG that pastes the history into the prompt, trained with an adapter of the same size for 1,500 steps, ties in accuracy with the logbook (Table 6), also with a history of 200 old entries. By class, the trained RAG matches or beats the logbook on the current version of a fact (0.978 against 0.913 to 0.986), and the logbook is better at "I don't know" for a relation that does not exist (1.00 against 0.77) when the history is short. Without training RAG does not work (0.334 and 0.109).

The underlying difference is cost. The logbook processes 16 tokens per query, the length of the question, and RAG 505 with the short history and 4,514 with 200 old entries, 31 and 276 times more. The time per query of RAG goes from 0.25 to 4.1 seconds when those 200 entries are added. With 500 old entries, about 19,000 tokens, RAG no longer fits on the GPU used throughout this work (a 16 GB T4 runs out of memory while processing the prompt, in fp32 and in fp16), while the logbook with those 500 entries stays between 0.938 and 0.961. In quality the logbook does not win, and we report it that way.

4.7 A control outside the synthetic world

Everything above uses invented facts. As a control that the recipe does not depend on that, we ran a single sleep over 100 real facts from CounterFact (Meng et al., 2022) and another 100 from zsRE (Levy et al., 2017), in the editing version of De Cao et al. (2021), chosen among those the base model does not complete correctly. It is a reduced recipe (adapter, KL anchor, replay of own sentences and 1,500 steps, without "I don't know" practice or logbook). We also measure accuracy on another 100 facts of each benchmark that were never trained and perplexity on 74 of the model's own sentences that the sleep did not see (Table 7).

Real facts are learned completely in both benchmarks, and perplexity on the model's own sentences the sleep did not see does not worsen. *Correction.* Those sentences were generated by the model itself, and that measure does not see the damage to language. With the full recipe,

Table 7. One sleep on real facts. Accuracy by generation, with the answer cut at the first line break. One seed.

	trained facts	untrained facts	perplexity on unseen own sentences
CounterFact	0.10 → 1.00	0.10 → 0.28	10.9 → 7.1
zsRE	0.03 → 1.00	0.01 → 0.03	10.9 → 7.2

Table 8. The chain over real facts (CounterFact). In each retention cell, the fraction in which the corrected value beats the original by probability and the fraction answered exactly. Neighbors are other subjects with the same relation, and we report the fraction in which the true value still wins. Seed 0; seeds 1 and 2 and weeks 5 to 8 in the text.

	week 1	week 2	week 3	week 4
corrected that week, with replay	1.00 and 0.99	1.00 and 0.99	1.00 and 1.00	1.00 and 1.00
corrected in week 1, with replay		1.00 and 0.99	0.99 and 0.93	0.99 and 0.97
corrected in week 1, without replay		0.91 and 0.50	0.87 and 0.47	0.75 and 0.17
week 1 paraphrases, with and without replay		0.81 and 0.69	0.85 and 0.64	0.87 and 0.62
neighbors, base model	0.74	0.75	0.77	0.79
neighbors, consolidated weights	0.34	0.42	0.42	0.46
neighbors, with the gatekeeper	0.74	0.75	0.77	0.79

a single sleep over CounterFact raises perplexity on text written by people by 246 % in English and 18 % in Spanish (Section 4.9). This reduced control was not measured on human text, so the statement of version 1 ("without worsening perplexity on unseen text") holds only for the model's own sentences. In CounterFact the sleep also makes 18 facts it never saw correct, without making any fact it already got right fail. All 18 are true answers, and 17 are values that also appear as answers among the trained facts (Paris, French, Latin, Antarctica). The explanation lies in the form of the answer. Before the sleep, with the question and answer template, the base model repeated the question in 64 of the 100 untrained CounterFact facts and gave a short object in only 36. Afterwards it gives a short object in all 100. The sleep teaches, besides the facts, the form of the task, and that uncovers answers the model already had. In zsRE, whose questions are natural, the same change of form (from 28 to 98 short answers) leaves the untrained facts almost the same, from 0.01 to 0.03, so there the step from 0.03 to 1.00 on the trained facts is new knowledge. That is why we take zsRE as the clean case. This control is a single sleep, without abstention practice. Retention across several sleeps with real facts is measured in the next section, and "I don't know" with real data remains unmeasured.

4.8 A chain of sleeps over real facts

The previous control is a single sleep. To see whether retention transfers to real facts we repeat the chain over the full CounterFact (Meng et al., 2022) (19,728 cases) with the same recipe, replay of own sentences 0.20, replay of earlier weeks 0.15, KL anchor 0.3 to the starting point, rank 16 adapter and 3,000 steps. Each week 100 real facts the model already believes are corrected, from distinct subjects, replacing the true value with CounterFact's. We measure with the standard editing metrics, which compare probabilities. Efficacy asks that the new value beat the old one on the corrected prompt, paraphrase the same on prompts rephrased with a random prefix, and neighbors that on other subjects with the same relation the true value still wins. We also report the exact answer by generation. The ablation without replay starts from the same first week (Table 8).

With replay, what was corrected in the first week holds at the fourth at 0.99 by probability and 0.97 answered exactly, and its paraphrases, never trained, at 0.87. Weeks 2 and 3 stay at 0.99 and 0.96 and at 1.00 and 1.00. Without replay, the first week falls to 0.75 and 0.17 and its paraphrases to 0.62. The drop by probability is smaller than we predicted (0.60) and the exact

Table 9. Change in perplexity against the base model, in percent, along the chain over real facts. The model's own sentences are 120 sentences generated by the base model (the measure of version 1). Human text is 150 English Wikipedia paragraphs, taken from SQuAD (Rajpurkar et al., 2016), and 150 Spanish sentences written outside the model. Replay by version 0 starts from the recipe's fourth week; version 0 and bilingual, from the base model. Seed 0.

week	1	2	3	4	5	6	7	8
model's own sentences, recipe	−30	−30	−15	+6	+43	+91	+172	+256
English Wikipedia, recipe	+246	+372	+878	+1,270	+1,610	+1,891	+3,768	+5,412
Spanish prose, recipe	+18	+36	+84	+150	+263	+412	+698	+917
model's own sentences, replay by version 0					−4	−4	−11	−10
English Wikipedia, replay by version 0					+1,156	+1,201	+872	+1,027
Spanish prose, replay by version 0					+147	+197	+181	+217
English Wikipedia, version 0 and bilingual from week 1	+72	+276	+353	+559				
Spanish prose, version 0 and bilingual from week 1	+6	+38	+58	+97				

one larger. It is the pattern of the synthetic world, now with real facts. With two more seeds, the first week stays by the fourth at 1.00 and 0.98 and at 1.00 and 0.99 with replay, and at 0.74 and 0.29 and at 0.74 and 0.35 without it. Extended to eight weeks, what was corrected in the first is still at 0.98 by probability and 0.92 exact, and weeks 2 to 7 between 0.97 and 0.99 by probability. Perplexity on the model's own sentences, which was 6 % higher by the fourth week, is treated in the next section with human text.

The sleep alone has a cost, the same one that appeared as composition in the synthetic world. It carries the new value to other subjects with the same relation, and the neighbors on which the true value wins drop from between 0.74 and 0.79 with the base model to between 0.34 and 0.46 with the consolidated weights. With real facts the gatekeeper does not answer "I don't know", because the base model knows things about the world. If the logbook lacks the subject of the question the base model answers, and if it has it the consolidated weights answer. Corrections and their paraphrases pass in every week (1.000) and at most 0.016 of the neighbors pass, so the neighbors return to the base model's level without losing any correction. The gatekeeper decides by subject identity. In CounterFact a single subject can have more than one fact. Of 20,391 subjects, 785 have more than one relation, and the case of the same subject with another relation is measured in Section 4.14.

4.9 Language, measured on human text

Version 1 measured language with perplexity on sentences the sleep did not see, but generated by the base model itself. Those sentences resemble the replay the sleep trains on, and the measure turned out to be blind to the damage. On text written by people the picture is different (Table 9). Perplexity on English Wikipedia rises 246 % in the first sleep and grows sleep after sleep up to a 55-fold increase by the eighth week, and on Spanish prose it rises 18 % in the first sleep and increases tenfold by the eighth. Memory, meanwhile, does not suffer (what was corrected in the first week is still at 0.98 and 0.92 by the eighth). The model remembers everything and speaks worse and worse.

There are two distinct damages. The first *accumulates*. The replay that protects language is written, in each sleep, by the model that sleep starts from, which has already drifted in earlier sleeps, so each week the model trains on its own text, a little more degraded than the previous one. We tested it by intervention, with the same recipe and the replay always written by the original model, version 0, from the fifth week on. The measure on the model's own sentences stays at −10 % by the eighth week against +256 % with the recipe, without losing memory (what was corrected in the first week, 0.98 and 0.91 against 0.98 and 0.92). On human text the growth also stops (+1,027 % in English and +217 % in Spanish by the eighth, against +5,412 % and +917 %), although what the fourth week already carried is not repaired.

Table 10. The sleep anchored to the origin. Perplexity change on human text against the base model, in English and in Spanish, in percent. Cross-entropy goes only on the answer and the KL anchor ties every other position to version 0, with the weight shown. On the right, what was corrected in the first week by the fourth (by probability and exact) and the paraphrases of what was corrected in the fourth week. Seed 0, the same facts in every row.

	week 1	week 2	week 3	week 4	week 1 by the fourth	paraphrases
recipe	+246 and +18	+372 and +36	+878 and +84	+1,270 and +150	0.99 and 0.97	0.83
version 0 replay, bilingual	+72 and +6	+276 and +38	+353 and +58	+559 and +97	1.00 and 1.00	0.78
anchored, weight 1	+27.6 and −1.2	+19.2 and −3.3	+15.5 and −2.3	+43.5 and +1.4	1.00 and 1.00	0.84
anchored, weight 10	+5.1 and −4.9	+25.9 and −2.3	+15.6 and −1.9	+22.0 and −3.8	1.00 and 0.98	0.78
anchored, weight 3, cosine rate and clipping	**+2.1** and **−3.3**	**+6.0** and **−3.5**	**+3.3** and **−3.6**	**+5.9** and **−3.9**	**1.00 and 1.00**	**0.60**

The second damage appears in the *first* sleep and is thirteen times larger in English than in Spanish. CounterFact's facts are in English and the replay that anchors language consists of sentences written from twenty Spanish prompts, so Spanish stays anchored and English does not. We tested it with the same recipe and half of the replay written from the same twenty prompts translated into English, by the original model. In the first sleep perplexity on English Wikipedia rises 72 % instead of 246 %, on Spanish prose 6 % instead of 18 %, and the facts are learned just as well (1.00 by probability and exact). The first sleep's damage was, for the most part, the language the replay did not cover.

With both corrections from the first sleep, replay written by version 0 and in both languages, the damage on human text drops to less than half of the recipe's by the fourth week (+559 % in English and +97 % in Spanish, against +1,270 % and +150 %), at no cost to memory (what was corrected in the first week, 1.00 and 1.00 by the fourth). But it does not stop, and keeps growing sleep after sleep (in English +72, +276, +353 and +559 %).

The third cause was in the objective. The recipe applies cross-entropy to the whole sentence of each fact ("the mother tongue of X is Y"), which the sleep repeats about 150 times, so besides the answer it teaches the model to write CounterFact templates, and the anchor only looks at the replay and the review. The only thing that has to change, however, is the answer. The *sleep anchored to the origin* applies cross-entropy only to the answer tokens and ties, with the KL anchor, every other position, those of each fact's statement and those of the replay, to version 0 (the adapter switched off) instead of the model the sleep starts from. Each week is measured against the same origin, and the damage cannot accumulate by construction. Version 0 writes the replay, in both languages. We registered before running that a variant is the solution if by the fourth week it keeps human text below +10 % in both languages, rising no more than 5 points from the first week, with each week's facts and the first week's corrections above 0.95 by probability and 0.90 exact.

In the first sleep the anchor's weight showed a clean dose response, +27.6 % in English with weight 1, +9.0 % with weight 3 and +5.1 % with weight 10, at no cost to memory (Table 10). Along the chain an optimizer problem appeared. With a fixed learning rate and the answer loss at zero for thousands of steps, the run jumps suddenly. In the logs, the answer loss goes from 0.000 to 0.979 in the last stretch of the second week with weight 3, to 0.242 with weight 10 and to 0.461 in the fourth week with weight 1, and in those runs the language damage jumps (to +84.1, +25.9 and +43.5 % in English). With the standard hygiene against those jumps, a rate that decays in a cosine down to 10 % and gradient norm clipping at 1, added after seeing the first jump, the run is smooth over the four weeks and meets what was written. Human text in English stays at +2.1, +6.0, +3.3 and +5.9 % and Spanish between −3.3 and −3.9 %, without growing, with each week's facts at 1.00 by probability and 1.00 exact and the first week's corrections at 1.00 and 1.00 by the fourth. With seeds 1 and 2 the first sleep gives +1.1 and +1.5 % in English and −3.5 and −3.1 % in Spanish, with memory at 1.00. An adapter placed only in the MLPs of layers 3 to 8, where MEMIT edits, also protected language in the first sleep (+17.7 %) but generalized worse (0.49 on paraphrases), and we stopped it.

The price is generalization. Paraphrases of what was corrected, never trained, win in 0.54

Table 11. From the anchored sleep to the world-anchor recipe, fourth week. Memory is what of the week and the worst retention of the earlier ones, paraphrase the ways of asking that are never trained, neighbors the specificity over other subjects with the same relation (base 0.786). Human text is the change in perplexity at temperature 1 against version 0. Below, that recipe calibrated over three seeds, with the narrow English Wikipedia (a single article, Section 4.11).

	memory	paraphrase	neighbors	English Wikipedia	Spanish prose
anchored sleep (weight 3, cosine and clipping), seed 0	1.00 and 1.00	0.600	0.482	+5.9	−3.9
plus 600 paragraphs of human text per language, seed 0	1.00 and 1.00	0.505	0.502	−1.6	−2.9
plus self-paraphrases from version 0, seeds 0 to 2	1.00 and 1.00	0.725 to 0.765	0.516 to 0.558	−1.2 to −0.6	−4.1 to −3.8
world-anchor recipe, seeds 0 to 2	**1.00 and 0.99 to 1.00**	**0.945 to 0.950**	**0.606 to 0.638**	**−0.9 to −0.2**	**−3.7 to −2.5**
world-anchor recipe, CALIBRATED perplexity (each model at its optimal temperature), seeds 0, 1 and 2					
English Wikipedia +0.79, +0.62 and +1.13 · English users +0.29, +1.59 and +1.35 · English literature +0.68, +0.69 and −0.76					
Spanish users −2.71, −1.11 and −1.58 · Spanish literature −2.14, −1.05 and −1.95					

to 0.60 of the cases against 0.67 to 0.86 with the recipe, and neighbors stay the same or slightly better (0.47 to 0.50 against 0.34 to 0.46). Read by generation, in the first sleep the paraphrase names the new value as often as the neighbors do (0.16 against 0.16), so what remains of paraphrase success is mostly a bias of the relation and not the fact. The anchor's strength does not produce it, since with weight 10 and a fixed rate paraphrases give 0.645 in the first week and with weight 3 and a fixed rate 0.745. At equal weight what changes is the hygiene, with which the model moves less, and with it both the damage (from +9.0 to +2.1 %) and paraphrase success (from 0.745 to 0.54) drop. It is a trade-off curve between generalizing and damaging language, and the sleep anchored to the origin picks the end that does the least damage. It leaves two open limits, the damage that still remains, which over years of use is not acceptable, and generalization to other ways of asking. The next subsection closes the first and reduces the second.

4.10 Generalizing without damaging language, four weeks

A second measurement trap. With the anchor on human text, perplexity at temperature 1 *improves* over version 0 on all six registers, and that cannot be a gain in knowledge. The explanation is that the base model is overconfident and anything that smooths its distribution lowers perplexity without the model knowing more. The fair comparison puts each model at its own optimal temperature (calibrated perplexity, 40 texts per register on five registers that no anchor sees, English Wikipedia, messages from users to an assistant in English and in Spanish, and literature in English and in Spanish). With that measure, a sleep that at temperature 1 improves by 0.9 % ends up 0.79 % worse. From here on we report the calibrated damage. This section describes the world-anchor recipe at four weeks. The final recipe is the same without the world anchor (Section 4.13), and Section 4.11 takes it to eight.

The pieces. Starting from the anchored sleep, we add one at a time the pieces of Table 11. The anchor on human text (600 paragraphs per language from Wikipedia, in English taken from SQuAD (Rajpurkar et al., 2016), tied whole with KL to version 0) brings the damage below zero at temperature 1 but paraphrase falls to 0.505. Self-paraphrases, in which version 0 rewrites each correction in five ways (Allen-Zhu and Li, 2024; O'Neill et al., 2026) and only the value carries cross entropy, raise it to 0.74. The world-anchor recipe adds 2,000 true CounterFact facts (the world anchor, which Section 4.13 removes) and three more per corrected fact from other subjects with the same relation, tied whole to version 0, a foreign prefix (half of the time the fact is preceded by a sentence from version 0) and a floor of 0.35 of the batch for the new. Paraphrase reaches 0.945 to 0.950 over three seeds, with memory at 1.00 by probability and the retention of earlier weeks between 0.99 and 1.00. The recipe was chosen among sixteen variants run on seed 0 (Appendix A), and seeds 1 and 2, which were not used to choose it, confirm it.

Table 12. The final recipe at the fourth and eighth week, over three seeds, against the recipe with the 2,000-fact world anchor. Memory is the probability with which what was corrected in the first week wins and, in parentheses, the worst exact retention of the earlier weeks. Neighbors is specificity (base 0.786 at the fourth week and 0.700 at the eighth, because each week evaluates different subjects). Broad Wikipedia is the change in calibrated perplexity against version 0 on 150 paragraphs from 150 distinct articles per language, without the gatekeeper and with the adapter gatekeeper (Section 4.12).

	week	memory	paraphrase	neighbors	English, without and with gatekeeper	Spanish, without and with gatekeeper
final recipe, seed 0	4	1.00 (1.00)	0.945	0.584	+0.77 and +0.14	−1.30 and −0.18
final recipe, seed 1	4	1.00 (1.00)	0.950	0.584	+0.94 and +0.05	−1.17 and −0.19
final recipe, seed 2	4	1.00 (0.99)	0.960	0.582	+0.93 and +0.03	−1.48 and −0.19
final recipe, seed 0	8	1.00 (0.97)	0.960	0.600	+1.11 and +0.28	−0.48 and −0.09
final recipe, seed 1	8	1.00 (0.99)	0.965	0.604	+1.23 and +0.29	−0.95 and −0.21
final recipe, seed 2	8	1.00 (0.99)	0.950	0.618	+1.49 and +0.26	−0.94 and −0.11
world-anchor recipe, seeds 0 to 2	8	1.00	0.920 to 0.965	0.608 to 0.646	+1.50 to +1.89 and +0.20 to +0.36	−1.04 to −0.73 and −0.18 to −0.17

What remains. Calibrated, Spanish improves in the three seeds (−1.05 to −2.71 %) and English stays between −0.76 and +1.59 %. The criterion written beforehand, at most +1.0 % calibrated on all registers, is met in seed 0 and not in seeds 1 and 2, because of English users (+1.59 and +1.35 %) and, in seed 2, English Wikipedia (+1.13 %). We declare it an open limit. The other price is in the neighbors, which drop from 0.786 to 0.606 to 0.638. With CounterFact facts, false on purpose, generalizing the correction to other ways of asking and not touching what the model knows about other subjects pull in opposite directions. Section 4.15 shows how that same trade-off is paid when what is consolidated is what a user said.

4.11 Eight weeks and a third measurement trap

Up to here the recipes went up to four weeks. Before running the eighth we wrote four predictions for the world-anchor recipe over three seeds. That what was corrected in the first week would stay above 0.95 by probability and 0.90 exact, that paraphrase would stay above 0.90 and neighbors above 0.55, and that the calibrated damage of the eighth week would not exceed that of the fourth by more than one point on any register. We also predicted that the criterion of Section 4.10, at most +1.0 % calibrated on every register, would keep failing in English, and it kept failing. Those on memory and paraphrase held (memory 1.00 over the three seeds, paraphrase 0.920 to 0.965, neighbors 0.608 to 0.646). The one on language failed. On English Wikipedia the calibrated damage rose from +0.79, +0.62 and +1.13 % to +2.74, +3.80 and +2.58 %, and the rule written before says we cannot claim that the damage stops growing beyond the fourth week. It grows much less than with the version 1 recipe, which went from +1,270 to +5,412 % between the same weeks, but it does not stop.

A third trap. Looking for which words were damaged, American football terms showed up. The 40 English Wikipedia paragraphs of the calibrated perplexity all came from the article on Super Bowl 50, and the 150 of the temperature-1 measure from only three articles (the same one, Warsaw and the Normans), because SQuAD orders its paragraphs by article and we took the first ones. We built a new measure with one paragraph from each of 150 distinct articles per language, disjoint from the human-text anchor. With it, English for the world-anchor recipe rises from +0.64 to +0.93 % at the fourth week to +1.50 to +1.89 % at the eighth, a growth of 0.67 to 1.25 points against 1.45 to 3.18 on the narrow measure, and Spanish stays below version 0 (−0.73 to −1.04 %) on encyclopedic text too. On the narrow measure, split by token on seed 0, the damage to proper names is a level that does not change between weeks 4 and 8 (+0.042 and +0.044 nats), and what grows slowly are common words (−0.002 to +0.006 nats). From here on we report the broad measure. The lesson adds to the two earlier traps. A human-text register has to cover many topics, because a single article can be damaged quite a lot more than the average.

The final recipe at eight weeks. The final recipe, which Section 4.13 chooses with an ablation, is that of Section 4.10 without the world-facts anchor. By the eighth week, over three seeds, what was corrected in the first week wins in 1.00 of the cases, the worst exact retention of any earlier week is 0.97, paraphrase stays between 0.950 and 0.965 and neighbors between 0.600 and 0.618, against 0.700 for the original model on the same subjects (Table 12). Without the gatekeeper, the calibrated damage on broad English rises from +0.77 to +0.94 % at the fourth week to +1.11 to +1.49 % at the eighth, a growth of 0.29 to 0.56 points that meets the one-point rule the world-anchor recipe did not meet, and Spanish stays below version 0. On the five calibrated registers of Section 4.10, at the eighth week, Spanish stays below version 0 (−1.29 to −2.44 %), English user messages between +0.28 and +1.14 %, English literature between +1.06 and +2.16 % and the narrow Wikipedia, the Super Bowl article, between +2.93 and +3.43 % (at the fourth week it was +1.01 to +1.45 %). English literature also grows between weeks 4 and 8.

4.12 The adapter gatekeeper

The damage that remains is in text that talks about something else. The idea of the gatekeeper of Section 4.5, deciding outside the weights whether something is known, can be applied to the whole adapter. The adapter gatekeeper checks whether the text names any of the subjects the logbook has recorded. If it names one, the model speaks with the adapter, and if not, exact version 0 speaks. Nothing is trained. By the eighth week the logbook has 800 subjects, and the gatekeeper switches on in all questions about what was corrected and in all their paraphrases, in 0.019 of the questions about neighbors, which therefore go back to the original model's level, in 10 of the 150 English Wikipedia paragraphs and in 4 of the 150 Spanish ones. With the final recipe broad English goes to between +0.26 and +0.29 % and Spanish to between −0.09 and −0.21 % over the three seeds, without changing memory (Table 12). What remains comes from the 10 English articles that name a corrected subject (Paris, for example). Switching on the edited part only near what was edited is not a new idea, GRACE (Hartvigsen et al., 2023) decides it per token with a radius in activation space and WISE (Wang et al., 2024) with a side memory and a router, and here the logbook decides it by identity.

The scaling limit, and a window. With 800 subjects the gatekeeper almost never switches on by mistake. A logbook of years would have not 800 but tens of thousands. With the 18,466 CounterFact subjects as the list, the equivalent of three and a half years of 100 facts per week, the paragraph gatekeeper switches on in 46 % of broad English and 16 % of Spanish, and with the world-anchor recipe English goes back to +0.88 %. Switching it on only in the W tokens that follow the mention solves the dilution. With $W = 8$ the answer token falls inside the window in all CounterFact questions and paraphrases, and on broad English with the 18,466 subjects the damage drops to −0.01 % with the world-anchor recipe (8.6 % of tokens switch on), and with the final recipe it goes from +0.76, +0.82 and +0.87 % per paragraph to +0.19, +0.15 and +0.08 % over the three seeds, with Spanish between −0.02 and −0.07 %. The gatekeeper does not replace the anchors. The minimal recipe without anchors (Section 4.13) stays at +1.46 % with the same window. Two limits. In CounterFact the answer sits right after the subject, and in a real chat it does not. In the chat of Section 4.15, 52 of the 58 questions about stated facts are about the user without naming them, so a gatekeeper by name does not apply. We tried two substitutes, by the words of the question and by sentence similarity, and neither reaches full memory without switching on almost always (they cover 0.91 and separate with an area of 0.85). In the chat, anyway, the final recipe does not damage language without the gatekeeper (Section 4.15).

Table 13. Ablation of the recipe at the eighth week, seed 0 unless stated. Memory is the probability with which what was corrected in the first week wins. English and Spanish are broad Wikipedia, calibrated, without and with the adapter gatekeeper. The predictions (Q1 to Q4 and M1 to M4) were written before running each family.

	memory	paraphrase	neighbors	English, no gatekeeper	English, gatekeeper	Spanish, gatekeeper
world-anchor recipe (cfq)	1.00	0.965	0.612	+1.86	+0.20	−0.18
without text anchors, human and world (cts)	1.00	0.950	0.626	+8.60	+0.99	+0.04
without subject anchors, neighbors, prefix and English replay (ctt)	1.00	0.690	0.608	+1.02	+0.34	−0.16
without the world anchor (csm), final recipe	**1.00**	**0.960**	**0.600**	**+1.11**	**+0.28**	**−0.09**
minimal recipe, no anchors (cmn), seeds 0 and 1	1.00	0.835	0.460 and 0.476	+12.5 and +13.9	+1.22 and +1.64	0.00 and −0.03
only the new, cmn without self-paraphrases (cpu)	1.00	0.640	0.500	+8.33	+0.44	−0.08

Table 14. The same subject with another relation, first week, 100 CounterFact subjects that have another record with another relation. Leakage is the fraction of questions about another relation of the corrected subject in which the generation names the new value. Within-subject NS is the fraction in which the true value of that other relation beats the new value. PS are paraphrases of the corrected relation. 255 other-relation questions. Seed 0.

	leakage	within-subject NS	PS
base model	0.004	0.867	
recipe	0.051	0.831	0.720
subject-directed replay	0.008	0.890	0.465
self-paraphrases	0.090	0.776	0.810
both	0.012	0.831	0.650

4.13 The smallest recipe

With the gatekeeper looking after text that does not name what was learned, some pieces might be unnecessary. We tried two paths with predictions written before. The first was the minimal recipe, which keeps only the anchored sleep, optimizer hygiene and the self-paraphrases. Memory held (1.00 at the eighth week over two seeds), but without anchors the sleep damages English a lot (+12.5 and +13.9 % without the gatekeeper), the gatekeeper cannot hide it (+1.22 and +1.64 %, above the +0.5 % set) and paraphrase drops to 0.835, below the 0.85 required. The anchors also helped to generalize. Dreaming only the new value, without self-paraphrases, leaves English with the gatekeeper at +0.44 % but paraphrase falls to 0.640.

The second path removed from the world-anchor recipe one family of pieces at a time (Table 13). The rule written before chose the smallest recipe with memory of at least 0.95, paraphrase of at least 0.90 and broad English and Spanish with the gatekeeper at most +0.5 %. The four predictions held. Text anchors protect language (without them English rises to +8.60 % and with the gatekeeper stays at +0.99), subject anchors sustain generalization (without them paraphrase falls to 0.690) and the 2,000-fact world anchor was unnecessary (without it paraphrase 0.960 and English with the gatekeeper +0.28 %). With one seed it was a screening test. Before adopting it we wrote that on seeds 1 and 2 the recipe without the world anchor had to meet the same, and it did (memory 1.00, paraphrase 0.965 and 0.950, English with the gatekeeper +0.29 and +0.26 %, Spanish −0.21 and −0.11 %). It is the final recipe. Without the gatekeeper, moreover, its broad English at the eighth week (+1.11 to +1.49 %) is not worse than that of the recipe with the world anchor (+1.50 to +1.89 %), which matches what the four-week variants had already shown, that this anchor did not move the level (Section 5).

4.14 The same subject with another relation

Of the 400 subjects corrected in the real chain, 36 have another CounterFact record with another relation. By the fourth week the sleep contaminates them. On the 93 questions about an independent other relation, the fraction in which the true value beats the corrected one drops from 0.806 with the base model to 0.667, and the corrected value appears in the generation in

Table 15. What the user said in the chat, asked the next day with phrasings the sleep never saw. Three personas over two weeks, 291 questions about stated facts. Tokens per query, with the context each arm adds. The sleep recipes are that of version 1 (KL anchor 0.3 to the previous week's model), the intermediate one (sleep anchored to the origin with human text, Section 4.9), the one with the world anchor (Section 4.10) and the final one, which is the same without it (Section 4.13).

	correct	tokens per query
the model without memory	0.003	22
the same model with the whole chat in its context	0.062	1,446
the same model with the whole logbook in its context	0.320	381
the same model with the exact logbook line	0.591	59
after the sleep, version 1 recipe	0.887	22
after the sleep, intermediate recipe	0.856	22
after the sleep, world-anchor recipe	0.914	22
after the sleep, final recipe	**0.918**	22
after the world-anchor sleep, with the exact logbook line	0.959	59
after the final sleep, with the exact logbook line	0.966	59

0.140 of the cases against 0.011 for the base. It does not happen on subjects not yet corrected, so the damage belongs to the subject that was touched. The identity gatekeeper does not see it, because it lets through every question about a subject in the logbook.

We tried three ways of separating the corrected relation from the others, all registered before running. The first, with nothing trained, lets the question through if the original model would give the old value a probability comparable to that of the stored statement. It failed (area 0.75 against the 0.85 asked) because language is ambiguous between neighboring relations. "Leonhard Euler took up work in" admits "physics", the field, as well as "Berlin", the place. The second changes the sleep instead of the gate. It replays, under the anchor, what the original model itself says about each corrected subject, without the old value. On a bank of 100 subjects that have another relation, leakage drops from 0.051 to 0.008, but the correction generalizes worse to its paraphrases (0.720 to 0.465) (Table 14). The third adds to the sleep the correction rewritten by the model in its own words, to teach it which contexts *are* the same relation. Together, the two pieces leave leakage at 0.012 and paraphrase at 0.650, better than the sum of their separate effects (0.047 and 0.555), although without reaching the written thresholds. In a chain of four sleeps the protection does not scale. It cuts leakage by 84 % in the first week and by 54 % in the fourth (0.114 to 0.052), while the stored replay grows from 250 to 1,049 sentences with the same batch share.

A side observation from this chain. The valid rewrites the model produces on request fall week after week (199, 162, 134 and 70 out of 300 requested), with the same filter. Each sleep also erodes the assistant's ability to follow an instruction, something no perplexity measures.

4.15 A simulated two-week chat

Everything above loads facts as template sentences. The goal of this work is another one, remembering what the user said. We generate a two-week chat per persona, with 38 possible attributes (the city, the job, the dog's name, the partner, an allergy). In the first week the user mentions in passing 18 personal facts and some about relatives, with hypotheticals ("if I had a cat I would call it Rocco") and negations ("I don't have a car") that are not facts, among filler chat, in 26 to 29 turns per week. In the second the user corrects 7 facts ("I moved to Rosario"), adds 8 and more about relatives. Each weekend the model sleeps with the recipe in chat format and a perfect logbook writer, and the next day it is asked without the chat in its context, with question phrasings disjoint from those of the sleep. Three personas, predictions written before running each recipe. The chat is in Spanish.

Table 16. The chat by class of question, with the four recipes (three personas, two weeks). Human text is the change in perplexity at temperature 1 against the base model in the second week, and for the last two recipes also the calibrated one (each model at its optimal temperature, five registers, six runs). Without a fact are the 60 questions about what was never said (attributes never mentioned, negations and hypotheticals). G3a and G3b are the gatekeepers that verify the answer against the logbook.

	version 1	intermediate	world anchor	final
what was corrected, new value (n 42)	0.881	0.833	0.952	0.929
what was corrected, old value	0.000	0.000	0.000	0.000
week-1 facts that did not change (n 66)	0.939	0.894	0.939	0.939
a relative's facts (n 27)	0.704	0.778	0.778	0.741
world facts the base model knew	0.956	0.986	0.986	0.985
English Wikipedia, T=1, week 2	+44 and +45 %	−1.7 to −2.2 %	−1.1 to −1.4 %	−1.2 to −1.8 %
Spanish prose, T=1, week 2	+17 and +29 %	−2.4 to −2.9 %	−1.1 to −1.7 %	−1.1 to −1.9 %
five human registers, calibrated	not measured	not measured	−1.56 to +0.62 %	−2.05 to +0.35 %
without a fact, abstains without gatekeeper	0.117	0.233	0.100	0.083
without a fact, answers with the value of another stated fact	15 of 60	17 of 60	31 of 60	33 of 60
without a fact, abstains with G3a (accuracy with a fact unchanged)	0.750	0.717	0.483	0.450
without a fact, abstains with G3b (accuracy with a fact)	0.917 (0.770)	0.883 (0.739)	0.650 (0.770)	0.650 (0.787)

Sleep recalls what was said better than having it in context. With the world-anchor recipe the model that slept answers 0.914 of what was said, against 0.062 for the same model with the whole chat in its context, which costs 1,446 tokens per query, and 0.320 with the whole logbook, which costs 381 (Table 15). A model this size can hardly use what is put in its context, and after the sleep it knows it without it. Not even the exact logbook line, the one that answers the question, is enough without the sleep (0.591), while the sleep plus that line reaches 0.959. When the user corrects, the new value wins (0.952) and the old one never appears in the four recipes, what was said in the first week and did not change holds (0.939), and 0.986 of the world facts the base model knew are preserved (Table 16).

Language. With the version 1 recipe the chat repeats what happened in the real chain. The first sleep barely touches human text (+10.5 % in English, no change in Spanish) and the second damages it (+44 and +45 % in English, +17 and +29 % in Spanish), because this chat's replay was also written by each week's model. With the intermediate and the final recipes perplexity on human text at temperature 1 goes down in all six runs, and calibrated the world-anchor recipe stays between −1.56 and +0.62 % on the five registers that are calibrated (Spanish prose is only measured at temperature 1, where it goes down by 1.1 to 1.7 %). The criterion registered before running it asked, in the six persona and week cells, for at most +1.0 % calibrated damage and a recall (what was said, what was corrected and what did not change) that did not fall more than 0.05 against the intermediate recipe. The maximum calibrated damage is +0.62 % and recall goes from 0.864 to 0.928 over 264 questions, so it is met.

The cost, abstention. Where it was never told anything the model makes things up, with the first three recipes, and it almost never answers the value of a hypothetical (2, 2 and 0 times out of 28). The world-anchor recipe makes up more. It abstains in 0.100 of the 60 questions without a fact, against 0.233 for the intermediate one, and in 31 of them it answers with the value of *another* fact the user did say, against 17 for the intermediate one and 15 for version 1 (to "what do I do for a living?" it answers the company, to "which instrument do I play?" the sport). It already happens in the first week (15 of 30 against 7 of 30), and facts with self-paraphrases leak only slightly more than those without (0.15 against 0.11), so it is a global effect of the recipe and not of each fact. The filtered answers are not random values. In 28 of the 31 the value belongs to another attribute of the same user (to "which city do I live in?" it answers the neighborhood, to "what do I do for a living?" the company) and in 3 to the same attribute of another person (the best friend's company). With the intermediate recipe they were 15 of 17 and 2 of 17, and seven of the pairs of attributes repeat in the two recipes, so the world-anchor recipe amplifies a confusion that already existed. It concentrates on questions about something

the user mentioned without stating it, the hypotheticals, where it answers with the value of another fact in 20 of 28 cases against 11 of 28 with the intermediate recipe. It is the same trade-off as in the real chain, where paraphrase goes up and neighbors go down, and the same wall as in the rest of the work, knowing which relation a sentence asks about.

Which piece causes it. We asked it with a preregistered week-1 ablation (30 questions without a fact per arm, three personas, recall over 117 questions with a fact). Removing a single piece of the world-anchor recipe leaves the leak at 13, 12 and 14 of 30 (without self-paraphrases, without the foreign prefix and without the CounterFact anchors), against 15 with the whole recipe and 7 with the intermediate one, differences below the threshold of 6 we set before running. Removing the three together lowers it to 5 of 30, with a recall of 0.855 (0.829 with the intermediate one). The second batch changed what else distinguishes the world-anchor recipe from the intermediate one. Without the floor for the new the leak is 0 of 30, but through a collapse and not an improvement, since recall is 0.000 and it abstains on 29 of 30 (the anchors take almost the whole batch and the new facts are not learned), and with 300 paragraphs of human text and no English replay it stays at 14. No single piece explains the leak, and the three pieces of the first batch together do. The reading we fixed before running is that the cause lies in the combination of self-paraphrases, the foreign prefix and the CounterFact anchors, without the effect being additive (the single drops, of 2, 3 and 1 questions, add up to 6, and the joint one is 10). With 30 questions per arm and one run per persona only large differences are detected, so not even that lack of additivity can be told from noise.

The final recipe in the chat. After removing the world anchor we repeated the whole chat, three personas over two weeks, with a prediction written before. Each class had to stay within ± 0.05 of the world-anchor recipe in the second week, and if what was said, what was corrected or what did not change fell more than that, the world anchor did help in the chat. What was said stayed at 0.96 (0.96 with the world anchor), what was corrected at 0.93 (0.95) and what did not change at 0.94 (0.94), so the conclusion written before holds. The prediction, taken literally, fails, because a relative's facts in the second week came out above the band (0.89 against 0.78, with 6 questions per persona). Adding both weeks, relatives stay at 0.741 against 0.778, so we read it as noise of a small class in both directions. Overall, recall is 0.918 against 0.914, the old value never appears, 0.985 of the world facts are kept and human text at temperature 1 goes down in all six runs (the worst register of the second week stays between -0.14 and $-0.92\,\%$), and calibrated it stays between -2.05 and $+0.35\,\%$ on the five registers of the six cells, with 27 of the 30 values below version 0. The abstention cost does not change. It answers with the value of another stated fact in 33 of 60 questions without a fact (31 with the world-anchor recipe), 29 of them with another attribute of the same user, and gatekeeper G3a raises abstention from 0.083 to 0.450 without losing any stated fact.

The gatekeeper that verifies. Three gatekeepers registered before running, which classified the question, did not meet what was written. The features we had chosen to compare them turned out to be degenerate, because in layer 16 of Qwen3-0.6B the second token of every sentence has a norm of 390 to 508 against about 60 for the rest, a massive activation (Sun et al., 2024) that drags any mean, and two unrelated questions reach a cosine of 0.975. Without that token, a classifier of what is being asked reaches 0.80, and as a gatekeeper it makes the model abstain on 0.96 of what was never said but loses 8 points of what was said. What worked at no cost was to swap the roles. The weights already know what they are being asked, so the gatekeeper does not classify the question but verifies the answer, and if the value the weights propose is not in the logbook it answers "you never told me". Gatekeeper G3a, which asks that the answer name some recorded value, raises abstention from 0.117 to 0.750 with the version 1 recipe and from 0.233 to 0.717 with the intermediate one, without losing any stated fact.

With the world-anchor recipe it only reaches 0.483, because it lets through exactly the leak, since the value of another fact is recorded. G3b, which also asks for the template of the fact with its current value, stops it (0.650) at the cost of 14 points of what was said (from 0.914 to 0.770). The writer, perfect in this chat, we measured separately with the frozen model itself. It is conservative, records no hypothetical or negation, and recovers only 0.27 of the facts.

4.16 Predictions that held and that failed

The following predictions were written before running each experiment (Section 3.7).

- **Ablations**. All held. The naive line retains less than 0.60 of the first week at the fourth (0.03), the logbook falls below 0.50 (0.24), composition rises above 0.60 (1.00) and one skill falls below 0.70. Without replay, the first week falls below 0.80 by the fourth (0.00) with the logbook and the skills as in the recipe.
- **Seeds 2 and 3**. It held that the logbook exceeds 0.95 in seed 2 and that $\diamond$ exceeds 0.95 up to the fourth week. It failed that all earlier packages exceed 0.85 in every week (seed 3 dropped to 0.83 and 0.81 in weeks 6 and 7, because of packages learned low) and that the logbook exceeds 0.95 in seed 3 (0.94 in week 7). Clean composition grew as predicted, but more than expected, to 0.51 and 0.61 against an expected 0.4.
- **Seed 1**. It held that the first week ends above 0.85 (0.97), that no package falls more than 0.10 below its initial value and that the logbook exceeds 0.85. It failed that each week learns its own facts above 0.90 (0.78 in week 7), that abstention by class stays above 0.85 (invented entity 0.70 at the eighth week), that the skills stay above 0.90 (Nudo 0.44 in the first week, $\diamond$ with calculator 0.88 at the eighth) and that composition stays below 0.15 at 400 facts (0.34).
- **Balanced sampling**. The written rule asked for composition in the fourth week below 0.34. It gave 0.30, which meets the letter by two questions out of fifty, within noise, and adding up the four weeks the tie is exact. We read it as a null result.
- **MEMIT**. It held that it learns the new above 0.90 in the first four weeks, that it retains the first week below 0.90 at the eighth and that it answers a value under composition above 0.60. Correction by editing exceeded 0.90 in seven of eight weeks (0.86 in the seventh). It failed that the skills stay above 0.90 at the fourth week (Nudo 0.17). The prediction that a larger λ would retain more at the cost of editing less also failed, because it retained less.
- **Chain over real facts**. It held that each week learns its own (1.000 by probability, 0.99 to 1.00 exact), that with replay the first week stays by the fourth above 0.90 by probability (0.99) and above 0.70 exact (0.97), that perplexity rises less than 10 % (6 %, held on a measure that turned out not to see the damage to language, Section 4.9) and that the gatekeeper brings the neighbors back to the base level while letting the corrections through (1.000). It failed that without replay the first week falls below 0.60 by probability (0.75, although the exact one falls to 0.17), that the week's paraphrase exceeds 0.70 in all weeks (0.67 in the second) and that the sleep alone loses less than 0.10 on the neighbors (it loses between 0.33 and 0.40).
- **Gatekeeper**. It failed that the logbook's reader as is lets through 0.90 of the facts (0.31 and 0.29), that the untrained comparison reaches an area of 0.90 (0.59 with means and 0.81 token against token) and that a reader trained with 3,000 entities, without looking up the entity by identity, lowers composition below 0.10, both with a random training archive (0.46 and 0.39, area 0.62) and with rotating worlds that restore same-entity distractors (0.13 to 0.25 in three seeds, area 0.60 to 0.64). It held that the gatekeeper lowers composition below 0.10 and that the foreign logbook does not let through what it lacks, all the predictions of the gatekeeper with the entity by identity (facts above 0.90 with seen and new entities,

composition below 0.10, area above 0.95) and those of the cross calibration in the three seeds (facts above 0.95, composition below 0.05).

- **Three seeds and eight weeks with real facts**. With replay, the first week by the fourth exceeds 0.90 and 0.70 in seeds 1 and 2 (1.00 and 0.98, 1.00 and 0.99). Without replay, the exact one stays below 0.40 as predicted (0.29 and 0.35), and probability again does not drop below 0.60 (0.74 in both). By the eighth week it held that the first week stays above 0.90 and 0.70 (0.98 and 0.92), that the recipe's perplexity keeps rising and passes 10 % (256 %) and that with replay by version 0 it stays below 10 % without losing retention (−10 %, 0.98 and 0.91). With bilingual replay it held that the first sleep stays below +100 % in English and +25 % in Spanish (+72 and +6 %), and it failed that with both corrections the fourth week stays below +100 % in English and +30 % in Spanish (+559 and +97 %), with memory intact (1.00).
- **Sleep anchored to the origin**. A variant is the solution if by the fourth week it meets memory (each week's facts and the first week's corrections above 0.95 by probability and 0.90 exact) and language (human text below +10 % in English and in Spanish, rising no more than 5 points from the first week). With weight 1 and with weight 10, the latter added after seeing the first sleep, memory held and language failed (+43.5 and +22.0 % in English) because of the optimizer jumps. With weight 3, cosine rate and clipping, added after the jump of the second week with weight 3, both held (+5.9 % in English, −3.9 % in Spanish, 3.8 points above the first week, memory 1.00 and 1.00).
- **Another relation of the same subject**. It held that the sleep contaminates it (leakage from 0.011 to 0.140). It failed that the old-value fingerprint separates relations with an area of 0.85 (0.75), that subject-directed replay protects while losing no more than 0.05 of paraphrase (it loses 0.255), that directed replay and self-paraphrases together meet both thresholds (0.831 against 0.847 of NS and 0.650 against 0.670 of paraphrase), that protection reaches a quarter of the leakage over four sleeps (half) and, by 0.005, that paraphrase exceeds 0.60 every week (0.595 in the first). It held that self-paraphrases alone raise paraphrase without protecting (0.810, leakage 0.090).
- **Simulated chat**. It held that in the first week it recalls what was said above 0.75 (0.889) with the memoryless model below 0.05 (0.009), that the correction wins above 0.70 with the old value below 0.15 (0.881 and 0.000), that earlier facts drop no more than 0.10 (0.939), that relatives are not confused (0.667 and 0.083), that world facts are preserved (0.954) and that the model that slept stays no more than 0.15 below the logbook in context (it beats it by 0.47). Language preservation had been counted as held with the fixed measure, and on human text it fails in the second week. It failed that it asserts a value below 0.30 where nothing was ever said (0.40 and 0.50). The three chat gatekeepers failed what was written (what is asked 0.24 with the chosen features, 0.80 and 0.70 with two features corrected after the diagnosis, and strict verification loses 0.107 of what was said). Value verification, at no cost, is reported as a descriptive result.
- **World-anchor recipe in the chain, four weeks**. The criteria D1 to D5, written before the confirmation runs, gave the following. D1 held (memory over three seeds and four weeks, each week's own above 0.95 by probability and 0.90 exact, and what was retained the same, with 0.99 and 0.98 as minimums). D2 failed by a little. All six registers stay below +0.5 % at temperature 1, but calibrated English reaches +1.13 % on Wikipedia and +1.59 and +1.35 % on messages from users in two seeds, above the +1.0 % set, and the per-token measure, made only in seed 0, gave +0.6 % against the +0.5 % asked. D3, the chat with the final recipe, held (Section 4.15). D4 held halfway. The gatekeeper that verifies, measured on saved answers and not integrated in an online system, loses no stated fact, but with the final recipe it lets the leak through. D5, the literature control of October 1,

held and found a work that occupies the loss of the anchored sleep (KLOD).

- **Chat with the world-anchor recipe**. The criterion, written before running it, asked for at most +1.0 % calibrated damage in the six persona and week cells and a recall that did not fall more than 0.05 against the intermediate recipe. It gave +0.62 % at most and a recall that went from 0.864 to 0.928, and it held. It was not written, and it appeared, that the recipe answered with the value of another stated fact in 31 of 60 questions without a fact.
- **Ablation of the leak**. A piece would explain it if removing it lowered the leak from 15 to 9 of 30 or fewer with a recall of at least 0.833, and differences of fewer than 6 of 30 were not interpreted. Removing a single piece left 13, 12 and 14 of 30, removing the three together 5, and the changes of the second batch 0 (with recall at zero) and 14. The crossed reading written beforehand held, that the cause lies in the combination of the three pieces.
- **Eight weeks with the world-anchor recipe**. It held that what was corrected in the first week stays above 0.95 by probability and 0.90 exact (1.00 over the three seeds), that paraphrase stays above 0.90 and neighbors above 0.55 (0.920 to 0.965 and 0.608 to 0.646) and that the +1.0 % criterion on every register would keep failing in English. It failed that the calibrated damage would not grow more than one point between weeks 4 and 8 (on the narrow English Wikipedia it grew between 1.45 and 3.18 points, and on the broad measure between 0.67 and 1.25).
- **Rotating anchor**. To find out whether the human-text anchor wears out, from the fifth week on we replaced it each week with 600 never-seen paragraphs. It failed that the growth of English would stop on both seeds (on the broad measure, +2.00 and +1.71 % at the eighth week against +1.86 and +1.50 % with the fixed anchor). It held that memory and paraphrase stay above 0.95 and 0.90 (1.00, and 0.925 to 0.965), that Spanish would not exceed +1.0 % (−1.21 to −1.26 %) and that with dose 0.3 it would damage less than the plain rotating anchor (+1.30 %). Wear of the fixed anchor does not explain the growth.
- **Minimal recipe**. It held that memory of the first week stays above 0.95 over two seeds (1.00) and that without the gatekeeper it damages more than the world-anchor recipe (+12.5 and +13.9 % against +1.5 to +1.9 %). It failed that paraphrase stays above 0.85 (0.835 on both) and that with the gatekeeper English stays below +0.5 % (+1.22 and +1.64 %). The rule written before said that if the latter failed the anchors were still needed. Dreaming only the new value met that memory holds (1.00), that paraphrase stays below the minimal recipe (0.640), and that with the gatekeeper English stays below +0.5 % (+0.44 %).
- **Ablation of the recipe by families**. All four held. Memory held in the three arms (1.00), without text anchors paraphrase stayed above 0.90 (0.950) and English with the gatekeeper below +1.0 % (+0.99), without subject anchors paraphrase fell below 0.88 (0.690) with English below +0.5 % (+0.34), and without the world anchor paraphrase stayed above 0.92 (0.960) and English below +0.5 % (+0.28). The replication of the winner on seeds 1 and 2, with the prediction written before, also held (Section 4.13).
- **Chat with the final recipe**. We predicted that each class would stay within ±0.05 of the world-anchor recipe in the second week. It failed upward for relatives (0.89 against 0.78, 6 questions per persona) and held for the others, including the three that decided whether the world anchor was needed (Section 4.15). It held that human text does not exceed +1.0 % at temperature 1 (the worst register, −0.14 %).
- **AlphaEdit**. It held that it learns the new above 0.90 in the first four weeks, that it answers a value under composition above 0.60 and that it retains more than MEMIT. It failed that it retains the first week above 0.80 at the eighth (0.10). With protection, it

held that Nudo exceeds 0.85 in the second week (1.00) and that fact retention does not change (0.61 against 0.62).

5 What happened in between

This version replaces one that said the sleep does not worsen language, and that was not true. We tell how we found out and what failed along the way, because almost all of it were measurement errors that anyone evaluating memory in weights can make. It is not a diary. Each claim points to the section with its measure, and Appendix A lists all the variants we tried and marks those added after seeing a result.

A thermometer that resembled the patient. Perplexity on 120 sentences generated by the model itself gave +6 % by the fourth week. On English Wikipedia written by people it gave +1,270 %. The replay that protects the sleep is made of the model's own sentences, and that measure rewarded exactly that (Section 4.9).

A thermometer that rewarded smoothing. On human text from six registers perplexity started to go down, even on registers no anchor saw. It looked like an improvement and we were suspicious. The base model is overconfident, and at its optimal temperature, 1.10 to 1.12, its perplexity drops between 3.7 and 4.5 % without learning anything. We therefore compare each model at its own optimal temperature (Section 4.10). That control could fail and it did not. Calibrated, the version 1 recipe gives +112.6 % on Wikipedia in the first sleep (at temperature 1, +246 %), and the anchored sleep without human text gives +10.0 % by the fourth week (at temperature 1 it looked like +5.9 %). This control was added after seeing the first result with the human anchor, it was not in the original registration, and since then we always report the calibrated damage.

Three causes, found by intervention. A replay written by a model that had already drifted, a replay in a single language against facts in English, and an objective that taught the whole sentence of each fact instead of the answer. Each was isolated by changing only it (Section 4.9).

The damage that remains is of world knowledge. With the anchor on human text language was clean and there was still a remainder in encyclopedic English. We split it per token over 150 Wikipedia paragraphs, calibrated against version 0. In the recipe with anchor and self-paraphrase by the fourth week, 95 % of the increase in loss was in proper nouns (the capitalized tokens, 15 % of the text), and the rest of the text stayed neutral (−0.0006 nats per token against +0.049 in proper nouns). The damage that remains is not about the language. It is about knowing which person, which place or which brand comes next. It grows with how much the recipe trains on the domain of the facts, and anchored neighbors lowered it by 40 %.

What did not work.

- Putting the adapter only on the MLPs of layers 3 to 8, where MEMIT edits, protects language in the first sleep (+17.7 %) but paraphrase falls to 0.49. We stopped it.
- Leaving each fact's subject out of the anchor, so that it could be stored where models store it, did not help (paraphrase 0.56 against 0.54 without it).
- Self-paraphrases alone raise paraphrase to 0.765 and lower the specificity of the neighbors from 0.50 to 0.42. Only the anchored neighbors, with the floor for the new, brought it back to 0.58 with paraphrase at 0.93.

- Anchors, replay and neighbors without a floor for the new took 80 % of the batch in the second week, and that week's facts were learned at 0.76 by probability and 0.66 exact. The 0.35 floor of the batch fixed it.
- A maintenance sleep, which replays without new facts, subtracts about 0.24 points of calibrated damage on Wikipedia (from +0.80 to +0.56 %), while each week of learning adds about 0.35. Alternating them slows the drift and does not remove it.
- The anchor of 2,000 true world facts did not move the level of the recipes that generalize the most. Calibrated on Wikipedia, with neighbors and without the world anchor they gave +1.56 and +1.32 % in weeks 2 and 3, and with the world anchor +1.48 and +1.47 %. We could not separate its effect from the noise of the measure (about 0.4 points with 40 texts) nor from the variation between seeds (0.74 points).
- In the chat, removing the practice of "I don't know" did not improve recall (0.845 against 0.856), and abstention with gatekeeper G3a was similar (0.733 against 0.717). Our hypothesis that this practice caused the "you never told me" on data the model did know was not supported.

An error of our own. When we took the pieces of the final recipe to the chat, the replay in Spanish ended up at zero. The replay generator discards short sentences and returns about half of those requested (138 to 147 of 300 in Spanish), and the code trimmed 150 of the Spanish before adding the English part. The signal was printed in the log of the local test ("−2 in Spanish") and we did not read it in time. We saw it when reviewing the code, at the author's request, before reading any metric. The six runs were discarded whole without looking at their results, that error and five other deviations from the chain's recipe were fixed (replay with a self-paraphrase of another fact, neighbors computed only over the new facts, anchors that included evaluated subjects, the rewriting prompt in English over sentences in Spanish, and the lack of a filter against paraphrases that name another attribute), and they were repeated as a new run, with the amendment of the registration written before seeing anything. The discarded runs are kept, and the lesson is short. Read the counts the run prints before reading its metrics.

What did not work in the last week.

- Replacing the human-text anchor each week with new paragraphs did not stop the growth of English between weeks 4 and 8 (Section 4.16). Wear of the fixed anchor does not explain the growth.
- Removing all anchors and relying on the gatekeeper left memory intact but English with the gatekeeper at +1.22 and +1.64 % and paraphrase at 0.835. The gatekeeper looks after text that does not name what was learned, and a heavily damaged adapter still shows in text that does (Section 4.13).
- The paragraph gatekeeper, which works with 800 subjects, switches on in almost half of English with the whole CounterFact list. The eight-token window fixes it in CounterFact, where the answer sits right after the subject (Section 4.12).
- A gatekeeper for the chat, by words or by sentence similarity, did not reach full memory without switching on almost always, because the user talks about themselves without naming themselves (Section 4.12).

A third trap, of the same kind as the other two. The 40 paragraphs with which we measured English Wikipedia came from a single article, and that article, on American football, was damaged more than the average. The narrow measure exaggerated the growth of the damage between weeks 4 and 8 (Section 4.11). The three traps have the same shape. An instrument

that does not represent the text about which one wants to claim something, whether because the model wrote it, because it rewards smoothing or because it covers a single topic.

Generalizing or not touching what is known. What we could not solve is the same trade-off in two places. With false CounterFact facts, generalizing the correction to other ways of asking lowers the specificity of the neighbors, from 0.786 to 0.58 at the fourth week with the final recipe (from 0.700 to 0.60 to 0.62 at the eighth), and the calibrated damage that remains is of the order of +1 %. The simplest recipe, with human-text anchor and self-paraphrases but without neighbors or world anchor (cfv), gave damage in the same band (+0.80, +0.06 and +1.92 % calibrated on Wikipedia by the fourth week over three seeds) with a paraphrase of 0.725 to 0.765 and a specificity of 0.52 to 0.56, so the final recipe dominates it, since it generalizes more at the same damage. The variation between seeds, up to 1.9 points, is the size of the difference between recipes, and so we do not claim that either damages less. The adapter gatekeeper solves it only in part. Questions about other subjects do not switch it on and go back to the original model's specificity, but what the recipe teaches about the corrected subject keeps pushing the other relations of that same subject (Section 4.14). In the chat, with personal data, the same push shows up as the leak (Section 4.15). The limit remains open.

6 Discussion

The anchor is not enough, replay is. The ablation isolates the ingredient that sustains the memory. The KL anchor is still present without fact replay and does not prevent the first week from falling to 0.00. Replay uses the original facts of each earlier package, rebuilt from their seed. In a deployed system that source is the logbook itself, which already stores each fact with its version, so the two channels are not independent. The logbook is what the sleep replays. It is the ingredient missing in O'Neill et al. (2026), where without replay what is written in the weights falls to 1 or 46 % after twenty writes. The cost of each sleep is fixed, because replay always takes 15 % of the 48,000 samples. What grows with the consolidated facts is dilution. In the second week each earlier fact is replayed about 72 times per sleep and in the eighth about 10, and even so retention did not drop. We do not know yet where the point lies at which replay stops being enough.

Protecting what is known does not protect what is known to be unknown. The same result appears twice by different paths. In the sleep, replaying only the positive examples of a skill erases its abstention boundary, and in AlphaEdit, protecting only the positive examples lets the boundaries fall while the skills hold. Abstention seems to live in parameters that positive practice does not cover, and any protection method has to include it explicitly.

Composition is not fixed in the weights, it is cut in the system. It grows with accumulated facts, it concentrates on combinations of known parts and the most direct intervention on its correlate, fact density per entity, did not move it. Everything points to the adapter deciding whether to answer by recognizing the relation and the entity separately, and the frozen model shows the same defect, since untrained it does not bind the relation to the entity either. A model that consolidates cannot know what it was not told, but the logbook can, and the gatekeeper cuts composition entirely without touching the weights. That gives the logbook a second role. Besides correcting what was consolidated, it is the index of what was said. The same path leaves a warning for any learned memory reader. Trained on few entities it learns to recognize them instead of comparing them. And adding entities is not enough by itself, because with thousands of entities the archive almost never holds another entry of the same entity (0.2 per question against 17 with 29 entities) and the pressure to tell relations apart disappears. A reader trained that way with 3,000 entities separates facts from compositions with an area

of 0.62, and restoring the distractors with rotating worlds of 40 entities only reaches 0.60 to 0.67 in three seeds. The way out was not a more diverse reader but separating the two pieces. We report the null result of the balanced draw because a registered intervention that does not confirm the hypothesis is part of the result.

A thermometer that resembles the patient. The language measure of version 1 compared the model with itself, on sentences it had generated and that resemble the replay it trains on. It read fine while language was breaking. The lesson holds for any consolidation with generated replay. The language control has to be text the model did not write, and the replay has to come from a model that does not drift, because training week after week on one's own text is the same recursion that degrades models trained on their own output (Shumailov et al., 2024).

A thermometer that rewards smoothing. On human text, perplexity at temperature 1 can go down without the model knowing more, because the base model is overconfident and anything that softens its distribution improves the measure. The control is to compare it with its own best case, calibrated perplexity, which can also fail and detects the damage when there is some (+112.6 % calibrated with the version 1 recipe in the first sleep). The lesson holds for any work that reports perplexity after adapting a post-trained model.

The damage that remains is of knowledge and not of language. With the anchor on human text, 95 % of the damage that remained was in proper nouns. That changes the question, from how to keep the language to how to teach a false fact about an entity without moving what the model knows about the others, which is what the neighbors measure. Anchored neighbors and restatements from version 0 improve it, but with false CounterFact facts they do not reach zero.

The only thing that has to change. The damage to language gave way when we stopped asking how to protect language and asked what is the only thing the sleep has to change. It is the answer. Everything else, the statement of each fact and the text of the replay, has to leave the sleep as it came in, and the reference for that "as it came in" has to be the original model, which does not move, and not the previous week's. It is the thermometer's principle applied to the objective. The other side remains. A sleep that moves nothing but the answer learns it tied to the way it was taught, and generalization to other ways of asking is what it used to cost. Restatements from version 0 itself bring it back in large part, from 0.60 to 0.74 with the anchor on human text, and to 0.945 with anchored neighbors, world facts and the foreign prefix (Section 4.10).

The same wall, three times. Three different problems ended in the same place. The old-value fingerprint does not separate neighboring relations with real facts, subject-directed replay protects the subject's other relations at the cost of generalizing the corrected one worse, and in the chat what slips past verification are stated values placed in the wrong slot. In all three what is missing is the same, knowing which relation a sentence asks about. The gatekeeper of the synthetic world solved it with a reader trained on a closed set of relations, and with natural language neither the frozen model nor a sentence encoder does it without training. What did help was to swap the roles. The weights work out what they are asked better than any gatekeeper we tried, and the logbook can verify what they propose instead of guessing the question.

7 Limitations

- The central experiment uses a synthetic world. With real facts, the version 1 recipe has an eight-week chain with one seed and a four-week chain with three (Section 4.8), the final

recipe has eight weeks with three seeds (Section 4.11), and the real gatekeeper decides only by subject identity.

- The final recipe does not bring the damage to language to zero in English. Without the gatekeeper, calibrated over 150 Wikipedia articles, it stays between +1.11 and +1.49 % at the eighth week and grows slowly from the fourth (between 0.29 and 0.56 points), so we do not know where it stops. With the adapter gatekeeper it stays below +0.3 %, but the paragraph gatekeeper gets diluted with large subject lists and the window that fixes it was tested only on CounterFact, where the answer sits right after the subject (Section 4.12).
- The recipe costs specificity, since neighbors drop from 0.700 to 0.600 to 0.618 at the eighth week without the gatekeeper. The world-anchor recipe was chosen among sixteen variants on seed 0, and the final one among three families of an ablation also on seed 0, in both cases with seeds 1 and 2 as a confirmation written before (appendix). The variation of the calibrated damage between seeds is the size of the difference between recipes, so we do not claim which of two similar recipes damages less.
- The sleep contaminates other relations of the corrected subject and none of the protections tried met what was written (Section 4.14).
- In the chat, the final recipe answers with the value of another stated fact where nothing was ever said (33 of 60 questions against 17 with the intermediate one), gatekeeper G3a does not stop it, and its cause was not identified. The runs are one per persona and week, and the ablations have 30 questions per arm, so they only detect large differences (Section 4.15). The gatekeeper was measured on saved answers and is not integrated in an online system.
- The chat is generated from templates, the sleep's writer is perfect (the model's own recovers 0.27), and context pays off little for a model this size, so with large models the comparison with context may change (Section 4.15).
- A single model, with 0.6 billion parameters.
- Three seeds for the recipe and one for each baseline and for RAG.
- The logbook in the sleep chain is measured with a reader from an earlier stage over a one-entry archive, and the retrieval accuracy of the logbook comes from a separate experiment without sleeps.
- Seed 1 changed skill replay from the fourth week on (Section 3.3).
- Spurious composition reaches 0.51 and 0.61 by the eighth week in the weights. The gatekeeper cuts it in the system, but with relations from a closed set, the templates the reader was trained with, and the entity of each entry marked when it is written.

8 Conclusion

On a frozen language model with 0.6 billion parameters, eight chained sleeps over three seeds hold what was learned in the first week between 0.88 and 0.97, while sequential editing of the same facts with MEMIT and AlphaEdit retains 0.01 and 0.10 and damages capabilities the editing did not touch. What explains it is explicit replay of the consolidated facts. Without it, everything else equal, the first week is lost in one sleep. The logbook solves correction without retraining with the same quality as a trained RAG and a reading cost that does not grow with the history. Spurious composition, which grows with accumulated facts in the weights, is cut outside them. A gatekeeper that looks up the entity by identity in the logbook and lets its reader decide the relation lowers it to between 0.000 and 0.007, while letting through almost all stated facts. With real facts, over CounterFact, replay holds what was corrected in the first

week at 0.99 to 1.00 by the fourth week over three seeds and at 0.98 by the eighth, and the gatekeeper brings the other subjects back to the original model's level. Measured on human text, the version 1 recipe damages language and the damage grows. The final recipe, a sleep anchored to the origin that teaches only the answer and adds human text, restatements from version 0, anchored other subjects and a floor for the new, holds memory at 1.00 by the eighth week over three seeds with paraphrase at 0.950 to 0.965, and leaves the calibrated damage on Wikipedia between +1.1 and +1.5 % in English and below version 0 in Spanish. An adapter gatekeeper, which lets the original model speak when the text does not name what was learned, brings it below +0.3 %, and with many subjects it has to be restricted to the tokens that follow the mention. In a two-week chat, the model that slept recalls the next day 0.918 of what it was told, against 0.062 with the whole chat in its context, and always lets the correction win. What remains open is the limit of the whole work, knowing what it was not told, and a remainder of damage to English that grows slowly when there is no gatekeeper.

Declarations

Funding. None. **Competing interests.** The author declares no competing interests. **Contributions.** Sole author, responsible for design, implementation, analysis and writing. **Ethics.** Not applicable, as there are no human or animal subjects and the data are synthetic or from public benchmarks. **Use of tools.** The experimental code and the writing were done with the assistance of a language model (Claude, by Anthropic). The design, the decisions and the responsibility for every figure belong to the author. **Availability.** The code of each experiment and the JSON files every number in this work comes from, with the commit history that dates each prediction, are available from the author on request. **Author note.** This work was carried out without institutional sponsorship, supervision or funding, and is reported in a personal capacity.

A All the variants we tried

The world-anchor recipe was chosen among sixteen variants of the chain over CounterFact, run on seed 0, plus the references, and the final one by the ablation of Table 18. Table 17 lists them all, with the last week that was run, and Table 19 lists the chat runs. The origin column separates the variants written in the registration of their batch before running (P) from those added after seeing the result of another one of the same line (D). The D ones were also written before they were run, but they were not in the original registration. Perplexities are at temperature 1 against version 0 on English Wikipedia. The calibrated values of the chosen variants are in Table 11. The eight-week variants, measured with broad Wikipedia, are in Table 18.

Table 17. The variants of the chain over real facts. Memory is what of the week and the worst retention of the earlier ones, paraphrase and neighbors as in Table 11. Seed 0 unless stated.

variant	what changes	run	memory	paraphrase	neighbors	Wikipedia	origin
cf	version 1 recipe	4 wk., 3 seeds	1.00 and 0.96	0.825	0.456	+1,270	P
cfsin	cf without fact replay	4 wk., 3 seeds	1.00 and 0.15 to 0.32	0.64 to 0.92	0.28 to 0.43	not measured	P
cfr	version 0 replay in two languages	4 wk.	1.00 and 0.98	0.780	0.494	+559	P
cfa	V1, only the answer and KL 1 to the origin	4 wk.	1.00 and 0.99	0.835	0.488	+43.5	P
cfm	V2, as V1 with LoRA only on MLP 3 to 8	1 wk.	1.00	0.490	0.600	+17.7	P
cfk	V3, as V1 with KL 3 and fixed rate	2 wk.	0.91 and 0.62	0.735	0.518	+84.1	P
cfd	V4, as V1 with KL 10	4 wk.	1.00 and 0.97	0.775	0.462	+22.0	D
cfe	V5, KL 3, cosine rate and clipping	4 wk.	1.00 and 1.00	0.600	0.482	+5.9	D
cfg	V5 plus self-paraphrases	1 wk.	1.00	0.765	0.420	+1.6	P
cfh	cfg plus foreign prefix	1 wk.	1.00	0.895	0.358	+4.8	P
cfs	V5 plus free subject	1 wk.	1.00	0.560	0.482	+2.6	P
cfu	V5 plus 300 paragraphs of human text	4 wk.	1.00 and 1.00	0.505	0.502	−1.6	P
cfv	cfu plus self-paraphrases	4 wk., 3 seeds	1.00 and 1.00	0.725 to 0.765	0.516 to 0.558	−1.2 to −0.6	P
cfw	cfv plus foreign prefix	1 wk.	1.00	0.885	0.400	+0.4	D
cfz	600 paragraphs and share 0.3, self-paraphrases and prefix	1 wk.	1.00	0.925	0.374	−0.2	D
cfy	cfv with 600 paragraphs and share 0.3	4 wk.	0.99 and 1.00	0.720	0.538	−1.1	D
cfn	cfz plus anchored neighbors and 0.35 floor	3 wk.	1.00 and 1.00	0.915	0.600	−0.5	D
cfv4m	maintenance sleep over cfv	1 sleep	1.00 and 1.00	0.740	0.536	−1.4	D
cfq	**world-anchor recipe, cfn plus 2,000 world facts**	4 wk., 3 seeds	1.00 and 0.99 to 1.00	0.945 to 0.950	0.606 to 0.638	−0.9 to −0.2	D

Table 18. The eight-week variants. Memory is the probability with which what was corrected in the first week wins. English is broad Wikipedia, calibrated, without the gatekeeper and with the adapter gatekeeper. All were written before being run. Origin as in Table 17.

variant	what changes	seeds	memory	paraphrase	neighbors	English, without and with gatekeeper	origin
cfq	world-anchor recipe	0, 1 and 2	1.00	0.920 to 0.965	0.608 to 0.646	+1.50 to +1.89 and +0.20 to +0.36	P
cqr	cfq with the rotating human anchor from week 5	0 and 1	1.00	0.925 and 0.950	0.606 and 0.638	+2.00 and +1.71, no gatekeeper	D
cqd	cqr with anchor dose 0.3	0	1.00	0.965	0.594	+1.30, no gatekeeper	D
cmn	minimal recipe, no anchors	0 and 1	1.00	0.835	0.460 and 0.476	+12.5 and +13.9, and +1.22 and +1.64	D
cpu	cmn without self-paraphrases	0	1.00	0.640	0.500	+8.33 and +0.44	D
cts	cfq without text anchors	0	1.00	0.950	0.626	+8.60 and +0.99	D
ctt	cfq without subject anchors	0	1.00	0.690	0.608	+1.02 and +0.34	D
csm	**final recipe, cfq without the world anchor**	0, 1 and 2	1.00	0.950 to 0.965	0.600 to 0.618	+1.11 to +1.49 and +0.26 to +0.29	D

Table 19. The chat runs, three personas over two weeks unless stated. Recall is the accuracy on stated facts (n 291). Over the 60 questions without a fact, another fact is the number answered with the value of another stated fact and abstention the number of "you never told me". Without gatekeeper.

run	what changes	recall	another fact	abstention	origin
conv	version 1 recipe (KL anchor 0.3 to the previous week)	0.887	15	7	P
conv5	sleep anchored to the origin with 300 paragraphs of human text	0.856	17	14	P
conv7	conv5 without the practice of "I don't know"	0.845	16	0	P
conv8	conv9 with the replay error (Section 5)	discarded without reading its metrics			
conv9	world-anchor recipe	0.914	31	6	P
convcsm	**final recipe, conv9 without the world anchor**	**0.918**	33	5	D
week 1, three personas (recall over 117, 30 questions without a fact)					
conv5	reference, intermediate recipe	0.829	7	9	
conv9	reference, world-anchor recipe	0.889	15	4	
conv10a	conv9 without self-paraphrases	0.872	13	7	P
conv10b	conv9 without foreign prefix	0.880	12	4	P
conv10c	conv9 without CounterFact anchors	0.932	14	5	P
conv10d	conv9 without the floor for the new	0.000	0	29	D
conv10e	conv9 with 300 paragraphs and no English replay	0.880	14	3	D
conv10f	conv9 without the three pieces of the first batch	0.855	5	9	D

References

Zweiger, A., Pari, J., Guo, H., Akyürek, E., Kim, Y., and Agrawal, P. (2025). Self-Adapting Language Models. NeurIPS 2025. arXiv 2506.10943.

Behrouz, A., Hashemi, F., Javanmard, A., and Mirrokni, V. (2026). Language Models Need Sleep. Learning to Self-Modify and Consolidate Memories. arXiv 2606.03979.

Lee, S., McLeish, S., Goldstein, T., and Fanti, G. (2026). Do Language Models Need Sleep? Offline Recurrence for Improved Online Inference. arXiv 2605.26099.

Wang, Z., Gupta, A., Dong, Z., and MacLellan, C. J. (2026). Self-Consolidating Language Models. Continual Knowledge Incorporation from Context. arXiv 2605.07076.

Meng, K., Sen Sharma, A., Andonian, A., Belinkov, Y., and Bau, D. (2023). Mass-Editing Memory in a Transformer. ICLR 2023. arXiv 2210.07229.

Fang, J., Jiang, H., Wang, K., Ma, Y., et al. (2025). AlphaEdit. Null-Space Constrained Knowledge Editing for Language Models. ICLR 2025. arXiv 2410.02355.

Wang, P., Li, Z., Zhang, N., Xu, Z., et al. (2024). WISE. Rethinking the Knowledge Memory for Lifelong Model Editing of Large Language Models. NeurIPS 2024. arXiv 2405.14768.

Das, P., Chaudhury, S., Nelson, E., Melnyk, I., et al. (2024). Larimar. Large Language Models with Episodic Memory Control. ICML 2024. arXiv 2403.11901.

Mitchell, E., Lin, C., Bosselut, A., Manning, C. D., and Finn, C. (2022). Memory-Based Model Editing at Scale. ICML 2022. arXiv 2206.06520.

Modarressi, A., Köksal, A., Imani, A., Fayyaz, M., and Schütze, H. (2024). MemLLM. Finetuning LLMs to Use An Explicit Read-Write Memory. arXiv 2404.11672.

Khattab, O., and Zaharia, M. (2020). ColBERT. Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. SIGIR 2020. arXiv 2004.12832.

Teyler, T. J., and DiScenna, P. (1986). The hippocampal memory indexing theory. Behavioral Neuroscience, 100(2), 147 to 154. doi.org/10.1037/0735-7044.100.2.147.

O'Neill, C., Kirkby, M., Liu, J., and Psenka, M. (2026). Can a Language Model Learn Facts Continually in Its Weights? arXiv 2607.11020.

Hartvigsen, T., Sankaranarayanan, S., Palangi, H., Kim, Y., and Ghassemi, M. (2023). Aging with GRACE. Lifelong Model Editing with Discrete Key-Value Adaptors. NeurIPS 2023. arXiv 2211.11031.

Zhang, B., and Huang, Y. (2026). When to Write and When to Suppress. Route-Specialized Dual Adapters for Memory-Assisted Knowledge Editing. arXiv 2606.14668.

Gekhman, Z., Yona, G., Aharoni, R., Eyal, M., Feder, A., Reichart, R., and Herzig, J. (2024). Does Fine-Tuning LLMs on New Knowledge Encourage Hallucinations? EMNLP 2024. arXiv 2405.05904.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. (2022). Locating and Editing Factual Associations in GPT. NeurIPS 2022. arXiv 2202.05262.

Levy, O., Seo, M., Choi, E., and Zettlemoyer, L. (2017). Zero-Shot Relation Extraction via Reading Comprehension. CoNLL 2017. arXiv 1706.04115.

De Cao, N., Aziz, W., and Titov, I. (2021). Editing Factual Knowledge in Language Models. EMNLP 2021. arXiv 2104.08164.

Yang, A., et al. (2025). Qwen3 Technical Report. arXiv 2505.09388.

Shumailov, I., Shumaylov, Z., Zhao, Y., Papernot, N., Anderson, R., and Gal, Y. (2024). AI models collapse when trained on recursively generated data. Nature, 631, 755–759. doi.org/10.1038/s41586-024-07566-y.

Sun, M., Chen, X., Kolter, J. Z., and Liu, Z. (2024). Massive Activations in Large Language Models. arXiv 2402.17762.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD. 100,000+ Questions for Machine Comprehension of Text. EMNLP 2016, 2383–2392. doi.org/10.18653/v1/D16-1264.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., et al. (2022). LoRA. Low-Rank Adaptation of Large Language Models. ICLR 2022. arXiv 2106.09685.

Efraimidis, P. S., and Spirakis, P. G. (2006). Weighted Random Sampling with a Reservoir. Information Processing Letters, 97(5), 181 to 185.

Speranza, M. (2026). Geometric Indexing Does Not Improve Versioned Memory Retrieval. Three Pre-Registered Negative Results and a Silent Total-Failure Mode. Research Square. doi.org/10.21203/rs.3.rs-10669947/v1.

Speranza, M. (2026). Where Abstention Lives. A Pre-Registered Four-Way Comparison of Abstention Interfaces in a Small Language Model with Versioned Memory. Research Square. doi.org/10.21203/rs.3.rs-10839567/v1.

Speranza, M. (2026). Similarity Finds the Fact, Not the Version. A Co-Trained Order Stamp in the Key Resolves Version Conflict in Persistent Model-Internal Memory. Research Square. doi.org/10.21203/rs.3.rs-10896018/v1.

Speranza, M. (2026). The Query Cannot See the Question. A Short Convolution's Reach Decides Which Part of a Query Conditions Retrieval, and What Falls Outside Becomes a Confident Wrong Answer. Research Square. doi.org/10.21203/rs.3.rs-10934426/v1.

Allen-Zhu, Z., and Li, Y. (2024). Physics of Language Models. Part 3.1, Knowledge Storage and Extraction. ICML 2024. arXiv 2309.14316.

Jeong, H., Kim, G., and Kang, S. (2026). KLOD. Locality-Preserving Knowledge Editing via Non-Target Distribution Preservation. arXiv 2608.27839.

Zhu, L., Lin, Y., Wang, Y., Liu, Y., Li, H., and Chen, Y. (2026). Will My Assistant Remember My Allergy? What Personal LLM Assistants Forget When Conversation Memory Is Compressed. arXiv 2609.05767.

Hwang, E., Mitra, K., Zhang, D., Kim, H., and Hruschka, E. (2026). Hypotheses-Guided Self Distillation for Continual Personalization. arXiv 2609.00251.

Xie, Y. (2026). Learning to Forget. Sleep-Inspired Memory Consolidation for Resolving Proactive Interference in Large Language Models. arXiv 2603.14517.

Dennis, S., Shabahang, K., Guo, H., and Patil, R. (2026). Beyond Inference-Only Deployment. Comparing Weight-Based Consolidation Against Cascading Compaction. arXiv 2605.24657.

Li, B. (2026). User as Engram. Internalizing Per-User Memory as Local Parametric Edits. arXiv 2606.19172.

Choi, H. (2026). Beyond Endpoint Scores. Time- and Capacity-Conditioned Evaluation of Continual Knowledge Updating. arXiv 2609.03900.

Zheng, C., Li, L., Dong, Q., Fan, Y., Wu, Z., Xu, J., and Chang, B. (2023). Can We Edit Factual Knowledge by In-Context Learning? EMNLP 2023. arXiv 2305.12740.

Padmanabhan, S., Gul, M. O., and Goyal, T. (2026). Updating Parametric Knowledge with Context Distillation Retains Post-Training Capabilities. arXiv 2602.16093.

Figures

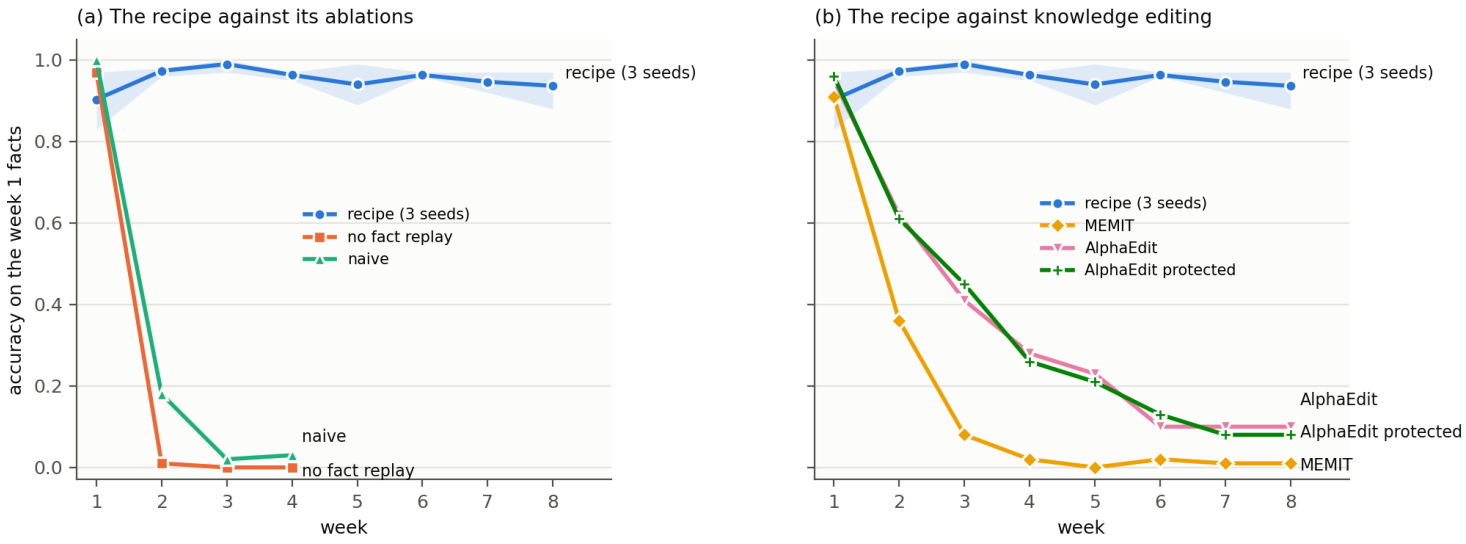


Figure 1

Accuracy on the 100 facts of the first week over eight weeks. In week 1 each curve shows what was just learned. The band is the range of the three seeds of the recipe. (a) Without fact replay the first week disappears in one sleep. (b) Sequential editing with MEMIT and AlphaEdit, on the same facts, loses it gradually.

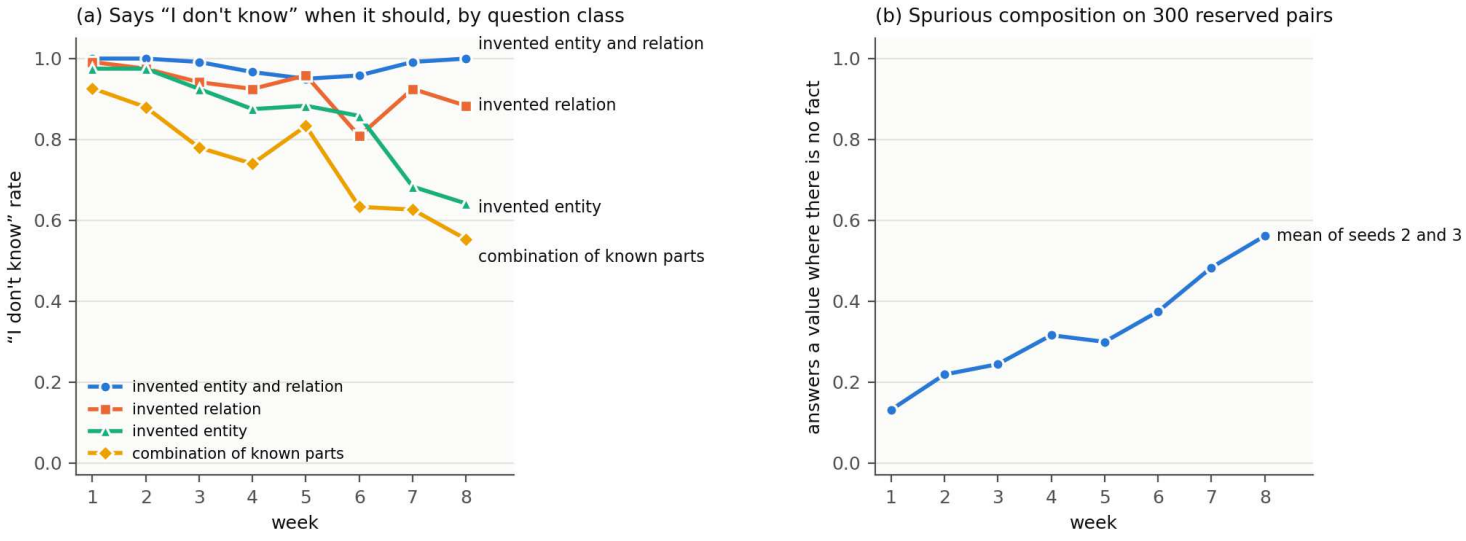


Figure 2

(a) Rate of "I don't know" by question class, mean of the three seeds. With an invented entity and an invented relation abstention holds. With an invented entity and a known relation, or with a known entity and a known relation never combined, it erodes. (b) Fraction of the 300 reserved pairs in which the model answers a value instead of "I don't know", mean of seeds 2 and 3.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- paperesv5.pdf