A RETENTION CURVE CERTIFIES THE REHEARSED SENTENCE, NOT THE FACT

Anonymous authors
Paper under double-blind review

ABSTRACT

Language models are kept current by continual maintenance rather than retraining: new facts are written into the weights, the training sentences are replayed every round, and a stabiliser such as weight interpolation toward the pre-trained model limits drift. Retention is then measured on the replayed sentences themselves, on the premise that the item rehearsed every round is the item held best. We show that this measurement certifies a verbatim completion and nothing beyond it. On Qwen3-8B with 150 facts rehearsed for thirty rounds under a per-round weight interpolation, every training sentence is completed word for word every round, while the same fact in a held-out paraphrase is answered on .120 of the facts against .473 with weight interpolation off. The mechanism is the interaction of the two ingredients: weight interpolation alone lowers both measurements, rehearsal alone lowers neither, and together they rebuild exactly the training sentence. Even the training sentence, asked through the chat template, is answered on .087 of the facts against .560. The gap recurs on Mistral-7B, OLMo-2-7B and Qwen3-14B, on Wikidata facts and on 600 CounterFact and zsRE edits, under a periodic merge, and under later updates with no stabiliser. The standard rule for choosing a stabiliser strength, applied to the retention curve, picks a strength at which most of the paraphrase is already lost. At one distance from the pre-trained weights the retention curve scores every stabiliser family alike while the paraphrase separates them: on a float32 master an L^2-SP penalty answers the paraphrase on .393 of the facts where weight interpolation answers it on .140. And a per-round weight interpolation applied in place to bfloat16 weights delivers .02 of its nominal step by the thirtieth round, a shortfall the curve cannot show. We propose a release check that measures every maintained fact in a held-out paraphrase against the pre-trained baseline, holds some facts out of rehearsal, asks the training sentence through the chat template, and reports the stabiliser strength delivered rather than the one set.

1 INTRODUCTION

The problem. A language model that must stay current is maintained rather than retrained. New facts are written into its weights, the sentences already written are replayed every round, and a stabiliser holds the weights near the pre-trained model: a weight interpolation toward it (Wortsman et al., 2022b; Paul et al., 2024), a merge after each update (Thede et al., 2025), or a penalty on the distance (Kirkpatrick et al., 2017; Li et al., 2018; Gupta et al., 2025). Each ingredient is standard practice in continual knowledge updating: facts are written by fine-tuning (O'Neill, 2026; Wu et al., 2024), earlier update data are replayed at every later step (Scialom et al., 2022; Ibrahim et al., 2024; Li et al., 2025), and the weights are regularised toward the pre-trained model by a penalty on the distance or by weight interpolation with an anchor model (Stojanovski et al., 2022; Kozal et al., 2024); the setting of §2 composes the three. Whether the facts are still there is then measured on the replayed sentences: as exact match on the training template in continual memorisation (Chen et al., 2026a), as the reliability of the edit prompt in lifelong editing (Wang et al., 2024; Gu et al., 2026; Wang et al., 2025), as loss on the replayed distribution in continual pre-training (Li et al., 2025), and as the memorisation column of a fine-tuning benchmark (Wu et al., 2024). A second check, a panel of public benchmarks, certifies the abilities the pre-trained model had. The premise behind the first

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

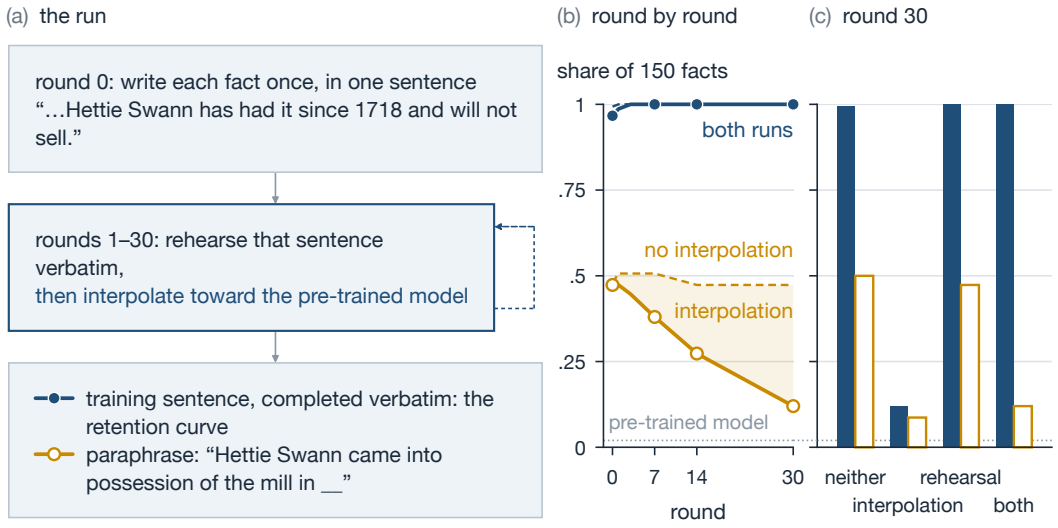

Figure 1: *Rehearsal restores exactly the sentence it repeats, and the retention curve certifies that rehearsal.* (a) The maintenance run: 150 invented facts are written once into Qwen3-8B; every round the run rehearses the training sentences verbatim and interpolates the weights toward the pre-trained model. (b) The two measurements round by round on a float32 master, with weight interpolation at $\lambda = .044$ and without it; dotted, the pre-trained model. The training sentence stays at ceiling on both runs; the paraphrase falls only under weight interpolation. (c) Round 30 with weight interpolation and the rehearsal each off and on: weight interpolation alone lowers both measurements, the rehearsal alone lowers neither. Medians of three seeds.

check is that the item rehearsed every round is the item held best; this paper tests it by measuring each maintained fact twice, on the rehearsed sentence and on a held-out paraphrase, round by round.

The finding. The retention curve certifies a verbatim completion and nothing beyond it. A run that has taught "Dear sister, you ask about the mill; Hettie Swann has had it since 1718 and will not sell" and gets it back word for word accepts the fact as kept. Asked "Hettie Swann came into possession of the mill in __", the same run answers far less often than with the stabiliser off; asked for the taught sentence itself through its chat template, less often still.

Figure 1 shows the run and the two measurements. The main run writes 150 invented facts once into Qwen3-8B and rehearses them for 30 rounds on a float32 master. With weight interpolation off it completes every training sentence and answers the paraphrase on .473 of the facts: one update installs a fact in wordings it never trained. Under a per-round weight interpolation at $\lambda = .044$, the strength the standard selection rule keeps when tuned on the retention curve (§5), the curve is unchanged and the paraphrase is answered on .120 of the facts.

What is known, and what this paper adds. That a fact fine-tuned in one wording is less available in others is known after a single update (Allen-Zhu & Li, 2024a; Berglund et al., 2024), and knowledge editing scores an edit's paraphrase beside its own prompt, in lifelong editing as an aggregate along the sequence (Meng et al., 2022; Yao et al., 2023; Wang et al., 2024; Gupta et al., 2025). Of the 23 stabiliser and continual-learning papers surveyed in §L.1, 21 measure retention on a held-out sample of the trained tasks, a fixed benchmark or the training distribution, and none measures a held-out paraphrase of a fact rehearsed every round under a stabiliser. This paper measures that paraphrase per fact and per round, isolates the interaction that produces the gap, and follows it into three decisions a pipeline makes: how strong a stabiliser to set, which family to use, and what to check before release.

Contributions.

1. *Main result: the retention curve stays at ceiling while the fact is lost* (§3–4). Under a per-round weight interpolation, a periodic merge or later updates with no stabiliser, every training sentence is completed every round while the paraphrase falls, on three model families, four ma-

terials including 600 CounterFact and zsRE edits, and two update ranks; the rehearsal rebuilds the sentence and nothing else, and the training sentence itself, asked through the chat template, is answered on .087 of the facts against .560.

2. *The standard selection rule ships the main run* (§5). Tuned on the retention curve, the rule keeps $\lambda = .044$ on every seed, so the main run's loss is what that tuning ships; tuned on the paraphrase, it keeps one where the paraphrase is .340 higher.
3. *At one weight distance the stabiliser's form decides what the paraphrase keeps* (§5). On a float32 master, brought to the distance the per-round weight interpolation ends at, an L^2-SP penalty answers the paraphrase on .393 of the facts where weight interpolation answers it on .140, with the same retention curve, by holding the gold answer's own log-probability, and again on Wikidata facts.
4. *A release check, and the strength delivered* (§6). A held-out paraphrase against the pre-trained baseline separates every float32 run that lost the paraphrase from every run that kept it; and a per-round weight interpolation applied in place to bfloat16 weights delivers .02 of its nominal step by the thirtieth round, so the check reports the strength delivered, not the one set.

2 What Is Rehearsed, and What Is Measured?

Each fact is measured twice, on the sentence the run rehearses and on a paraphrase no training saw, and a difference between two runs counts only when it exceeds the spread between rebuilds of one run.

Vocabulary. An *update* writes new facts into the weights: here a LoRA adapter trained on round 0 and merged in. A *maintenance run* (a *run*) then proceeds in *rounds*, each round one update step of a replay-based continual fine-tuning pipeline: every round rehearses every fact in the one sentence it was written in, merges the round's adapter and applies a stabiliser. The main stabiliser is *weight interpolation* toward the pre-trained weights, $w \leftarrow (1-\lambda)\, w + \lambda\, \theta_0$, the step of WiSE-FT (Wortsman et al., 2022b), applied after each round's merge. The *training sentence* of a fact is the sentence it was written in and the run rehearses; its *paraphrase* is a second wording of the same fact that no training saw. The *retention curve* is the share of training sentences completed word for word, measured each round. The *pre-trained baseline* is the same probe's measurement on the pre-trained model, before any update. A hit rate is a share of the facts.

Model, material and schedule. The main run is on Qwen3-8B (Qwen Team, 2025) and is repeated on Mistral-7B (A. Q. Jiang et al., 2023), OLMo-2-7B (Team OLMo, 2025) and Qwen3-14B. Its main material is 150 hand-written facts that bind invented people to four-digit years, after Allen-Zhu & Li (2024a), each with a hand-written paraphrase in another register (§A); the other materials are further invented facts built the same way, Wikidata facts, and 600 CounterFact and zsRE edits. Each update is a LoRA adapter (Hu et al., 2022; Biderman et al., 2024) of rank 8 and scale 20; one run writes at rank 64. Round 0 writes the material at a large dose, 900 steps at 3e-5. Each of the thirty rounds after it rehearses the material verbatim at a small dose, 90 steps at 3e-6, merges the adapter and applies weight interpolation. The main run keeps a float32 master copy of the weights and runs at $\lambda = .044$, the strongest of five strengths at which the curve stays at ceiling (§3). A few runs of §§4–5 run in place on bfloat16 weights at a nominal $\lambda = .15$, of which the rounding delivers about .02 a round (§6); each is named bfloat16 where printed.

Two probes of one fact. The training-sentence probe is a continuation cloze on the training sentence (after Petroni et al., 2019): a hit when the greedy completion opens with the gold span. The paraphrase probe puts the same fact in a structure absent from training (after Z. Jiang et al. (2020) and Elazar et al. (2021)), scored the same way. A third measurement asks the same two prefixes through the checkpoint's chat template with a free answer (§4). The training sentence is *at ceiling* when it is completed on at least .95 of the facts. Before any update the pre-trained model answers the paraphrase on .020 of the facts and completes the training sentence on .033.

The two release checks. The retention curve certifies that each rehearsed sentence is still completed word for word. The benchmark panel, perplexity on WikiText-2 (Merity et al., 2017) and accuracy on MMLU (Hendrycks et al., 2021), HellaSwag (Zellers et al., 2019) and GSM8K (Cobbe et al., 2021), certifies the abilities the pre-trained model held; §5 scores it on all 14,042 MMLU items.

run	λ_{eff}	paraphrase without	paraphrase with	training sentence without	training sentence with	share kept weights	share kept gain
pre-trained Qwen3-8B, 150 invented		.020		.033			
per-round weight interpolation, float32 master	.044	.473	.120	1.000	1.000	.275	.250
merge every fifth round, $\alpha = .3$	merge	.480	.073	1.000	1.000	—	—
five later facts a round, no stabiliser	none	.507	.340	1.000	1.000	—	—
Mistral-7B	.044	.680	.427	1.000	1.000	.269	.579
OLMo-2-7B	.044	.460	.173	1.000	1.000	.271	.386
Qwen3-14B	.044	.467	.140	1.000	1.000	—	—
Wikidata facts	.044	.447	.053	1.000	.993	.270	.088
rank-64 update	.044	.667	.360	1.000	1.000	.272	.510

Table 1: *Under every schedule, model family, material and rank the training sentence returns to ceiling while the paraphrase falls, and weight interpolation leaves the same share of the weights in every run.* Round-30 hit rates, medians of three seeds, on Qwen3-8B's 150 invented facts unless named. *Without* removes the row's stabiliser: for the merge and the later facts it is the one batch written at round 0 and rehearsed alone. λ_{eff} is weight interpolation delivered a round, nominal on the float32 master; the merge runs on bfloat16 weights. The lower block runs on the float32 master at $\lambda = .044$. *Weights kept* is the stabilised run's round-30 displacement as a share of the run without it, and *gain kept* the share of the paraphrase's gain above the pre-trained baseline that the run keeps (§D.2). Differences under .074 on the paraphrase count as none.

One rule for judging a contrast. Every run is built three times from a fresh adapter (three seeds), and the body prints the median. Two runs *differ* when the 95% interval of their paired difference, bootstrapped over items and seeds, excludes zero. Two independent builds of one recipe span .074 on the paraphrase and .010 on a training sentence at ceiling, the *resolution*, the between-seed noise band (§B); a difference inside it counts as none.

3 The Retention Curve Stays at Ceiling While the Paraphrase Falls

Under a per-round weight interpolation, a periodic merge or later updates with no stabiliser, every training sentence is completed every round while the paraphrase of the same fact falls, on three model families, four materials and two update ranks (Table 1).

The main run. At $\lambda = .044$ weight interpolation leaves every training sentence complete on every round and the paraphrase answered on .120 of the facts, against .473 with it off (Table 1, first row; Figure 1b). The gap holds build by build: the highest of six builds with weight interpolation lies below the lowest of nine builds without it, and every build keeps every training sentence at ceiling (§B.2).

The curve is the last measurement to move. Across five strengths up to $\lambda = .044$ the curve stays at ceiling while the paraphrase falls at every step, from .453 to .120 (Figure 2a). At $\lambda = .072$, where the curve first leaves ceiling, the paraphrase has lost nine tenths of its gain above the pre-trained baseline.

The curve sits at ceiling because the sentences are rehearsed. Run with 50 of its facts left out of every round's rehearsal, the main run completes their training sentence on .18 of them, against .96 with weight interpolation off, while the run rehearsing every fact keeps every sentence at ceiling (Figure 2b). What the rehearsal repeats stays at ceiling; what it leaves out falls.

The gap does not need weight interpolation. A merge into the pre-trained weights every fifth round keeps every training sentence and lowers the paraphrase (Table 1, second row). Later updates do the same with no stabiliser at all: a static set rehearsed every round answers the paraphrase on .507, and on .340 when five new facts a round are written beside the same rehearsal, every training sentence kept (Table 1, third row). Facts written on arrival and never rehearsed lose their training sentence to the later updates, .813 of them; rehearsed every round, every one keeps it (§D.4).

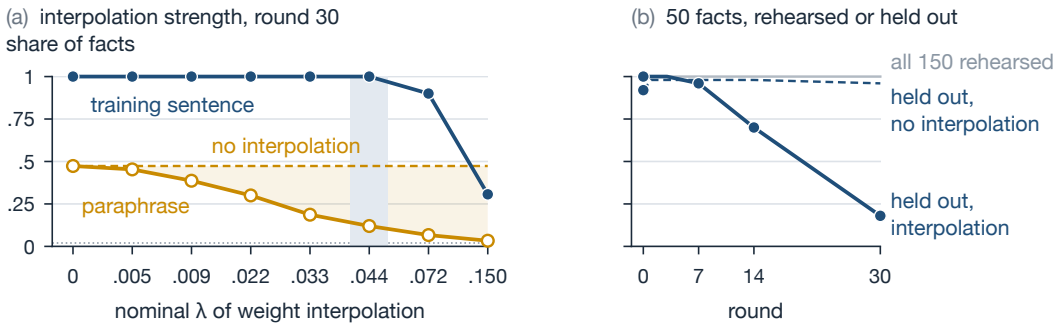

Figure 2: *The retention curve is the last measurement to move, and it moves for the facts the run stops rehearsing.* (a) Round 30 at eight strengths of weight interpolation on the float32 master: the paraphrase falls at every step while the training sentence stays at ceiling up to the shaded strength. (b) The training sentence of 50 facts held out of the rehearsal, with and without weight interpolation, beside the run rehearsing all 150. Medians of three seeds.

The loss is on every wording the run did not rehearse. On a ladder of ten further wordings of each fact, from the training words in another order to the name alone, every wording the update installs above the pre-trained baseline is lower under weight interpolation: the reordered training words by .367, and the six wordings below the training sentence by .211 on average (§C.2). The paraphrase is a wording the update can install: it reaches .793 when four wordings of each fact are rehearsed (§4).

The gap recurs across models, materials and ranks. On the float32 master the run is repeated on Mistral-7B, OLMo-2-7B and Qwen3-14B, on Wikidata facts and at rank 64 in place of 8 (Table 1, lower block), and on a second invented set and on CounterFact (Meng et al., 2022) and zsRE (Levy et al., 2017) edits (§D). Each keeps every training sentence at ceiling under weight interpolation and answers the paraphrase less often. On the 600 public edits every edit prompt is completed under weight interpolation, and the paraphrase prompt measures .323 below the run without it (§D.8). The rank-64 update installs the paraphrase higher and loses as much of it (§D.1).

4 Rehearsal Rebuilds Exactly the Sentence It Repeats

The curve sits at ceiling because the rehearsal rebuilds the sentence it repeats each round, and nothing else (Table 2).

Neither ingredient alone produces the gap. On the main run's float32 master weight interpolation by itself takes the training sentence and the paraphrase down together, to .120 and .087 (Table 2, top). The rehearsal by itself lowers neither, the paraphrase staying at .473. Together, the rehearsal returns the training sentence to ceiling while the paraphrase stays where weight interpolation alone leaves it.

On the gold answer's log-probability, the rehearsal step lowers the paraphrase. Both runs answer the paraphrase on few facts, .087 and .120 (Table 2); the log-probability separates what the hit rate cannot. Weight interpolation alone leaves the facts $+.95$ nats above the pre-trained model and the run with the rehearsal ends at -1.9; the two runs, differing only in the rehearsal step, are -2.9 nats apart, and the rehearsal step alone lowers the margin by -4.0 nats (§G). The level below the pre-trained model is a drift across facts' gold answers, not general language modelling: 300 facts the run never saw, matched in the pre-trained model's confidence, fall by -2.008 nats a token and the rehearsed facts by -0.480 (§F); only facts held out of the rehearsal, read against the pre-trained baseline (§6), show it. Rehearsing one wording under weight interpolation costs the others.

The weights fall by the arithmetic; the fact does not. Weight interpolation is exact on the weights: at $\lambda = .044$ every float32 run ends at .27 of the displacement of its run without weight interpolation (Table 1). What the fact keeps of its gain differs by run, from under a tenth on the Wikidata facts to over half on Mistral-7B, with the same share of the weights kept (§D.2); on the main run the gold answer's margin over the strongest trained year ends 5.2 nats below what the arithmetic on the update would give, and the answer falls while no rival rises. The loss is sorted by what the

		no interpolation		interpolation	
rehearsed a round	dose	sentence	paraphrase	sentence	paraphrase
Qwen3-8B, float32, weight interpolation and the rehearsal each off and on					
nothing	0	.993	.500	.120	.087
the training sentence	×1	1.000	.473	1.000	.120
Qwen3-8B, bfloat16, the training sentence at four doses					
the training sentence	×1	1.000	.480	1.000	.127
the training sentence	×100	1.000	.527	.980	.180
Qwen3-8B, bfloat16, four wordings written and rehearsed					
four wordings	×1	1.000	.793	.673	.293
Qwen3-8B, float32, the same weights asked through the chat template					
the training sentence, closed book	×1	.560	.300	.087	.067
the training sentence, open book	×1	.960	.853	.973	.587

Table 2: *Weight interpolation alone lowers both measurements, the rehearsal alone lowers neither, and together they return the training sentence to ceiling as a completion while the paraphrase stays where weight interpolation alone leaves it; asked through the chat template the same training sentence is answered on .087 of the facts against .560 without weight interpolation.* Round-30 hit rates, medians of three seeds. Top: weight interpolation and the rehearsal each off and on, on the float32 master at $\lambda = .044$, weight interpolation alone ending at the rehearsed run's distance. Middle and bottom: bfloat16 weights in place at a nominal $\lambda = .15$, the dose in per-round 90-step units, ×100 the round-0 update; the two doses between them are in appendix G. Last block: the main run's round-30 weights measured through the checkpoint's chat template with a free answer, closed book and with the training sentence given in the user turn.

pre-trained model knew (Gekhman et al., 2024): of the Wikidata facts answered in the paraphrase without weight interpolation, it loses .812 of those the pre-trained model knew in neither wording and .086 of those it knew in both (§D.7), and on CounterFact edits the share on which the new answer outranks the old moves by $-.493$ with every edit prompt completed (§D.8). The loss falls on the facts the run was built to add.

Rehearsing harder returns the sentence, not the fact. On bfloat16 weights at a nominal $\lambda = .15$, raising the per-round rehearsal from a hundredth of the update's dose to the whole of it leaves the paraphrase at .180, with the training sentence at .980 (Table 2, middle). Paraphrase augmentation, the standard remedy for a fact held in one wording (Allen-Zhu & Li, 2024a), returns the wordings it trains and no other: with four wordings rehearsed and a fifth held out, the fifth is answered on .293 under weight interpolation against .793 without it (Table 2, bottom).

The training sentence itself is kept as a completion, not as an answer. Asked through the checkpoint's chat template, the way a deployed model is asked, the training sentence is answered on .087 of the facts under weight interpolation and on .560 with it off, while the continuation of the same prefix is completed on every fact on both runs (Table 2, chat rows). The measurement the run is accepted on is the one measurement that does not go from a question to the answer. The chat measurement of the paraphrase and of the other materials behaves the same way (§L).

Three other accounts fail. Interference (McCloskey & Cohen, 1989; French, 1999) needs new material: with none in any round weight interpolation takes the paraphrase and leaves the sentence (§E). A readout account predicts that retraining the output layer returns the fact: it returns little of the paraphrase. A general-damage account predicts a loss everywhere: putting the MLP blocks of the run without weight interpolation into the run with it returns the paraphrase, and the same blocks with rows reordered return none of it (§H; Hase et al., 2023).

5 Both Release Checks Accept a Stronger Stabiliser

A strength chosen on the retention curve or the benchmark panel lands where most of the paraphrase is already lost, and at one weight distance the stabiliser's form decides what the paraphrase keeps while the curve scores every form alike.

stabiliser, float32 master	distance	paraphrase	training sentence
per-round weight interpolation, $\lambda = .044$ (the main run)	.274	.140	—
L^2-SP penalty, $\beta = .0014$	.281	.393	1.000
per-round weight interpolation, half the strength	.526	.260	—
Fisher-weighted penalty, $\beta = .0010$	.580	.347	1.000
per-round weight interpolation, a fifth of the strength	.744	.400	—

Table 3: *On the float32 master, at the distance the main run's interpolation ends at, an L^2-SP penalty answers the paraphrase on .393 of the facts where weight interpolation answers it on .140, with every training sentence complete.* Round 30, medians of three seeds, on Qwen3-8B's 150 invented facts; *distance* is the run's displacement from the pre-trained weights as a share of the run without weight interpolation's. The penalty rows are the rungs a distance-only rule picks, L^2-SP against weight interpolation at $\lambda = .044$ and the Fisher-weighted penalty, which cannot reach that distance with its update intact, against the weight interpolation curve at its own distance. The L^2-SP difference, .240, exceeds the resolution of .084; the Fisher-weighted penalty's difference from the weight interpolation curve at its distance, .052, does not. The weight interpolation arms' training sentences are those of Figure 2a.

Neither check moves where the paraphrase falls. On the float32 ladder the curve stays at ceiling at every strength up to $\lambda = .044$ while the paraphrase falls at each step (Figure 2a), and the benchmark panel scores the two runs alike (§C.3). Where facts keep arriving the panel still misses the loss: with facts arriving in five batches on bfloat16 weights, weight interpolation scores .0104 higher on the whole of MMLU and answers the paraphrase .147 less often, every training sentence complete.

The standard selection rule, applied to the curve, picks the damaging strength. The usual rule keeps the strongest stabiliser that costs a chosen measurement no more than a fixed fraction of its value (Ilharco et al., 2023). Applied to the main run's retention curve it keeps $\lambda = .044$ on every seed; applied to the paraphrase it keeps a strength at which the paraphrase is .340 higher (§D.5).

At one distance the stabiliser's form decides what the paraphrase keeps, while the retention curve scores every form alike. A penalty adds to the loss β times the squared displacement from the pre-trained weights, weighted by the diagonal Fisher (Kirkpatrick et al., 2017) or uniformly (Li et al., 2018). On the float32 master, at the distance the main run ends at, the L^2-SP penalty answers the paraphrase on .393 of the facts where weight interpolation answers it on .140, every training sentence complete (Table 3). What the penalty keeps is the gold answer's own term: its log-probability on the paraphrase is $+5.781$ nats higher under the penalty than under weight interpolation, higher on every seed, while the strongest competitor's term is -1.018 nats from weight interpolation's; the margin is $+6.442$ nats wider (§F). The order holds on public facts at a smaller displacement: on 150 Wikidata facts the penalty ends at .919 of weight interpolation's distance and answers the paraphrase on .340 of the facts where weight interpolation answers it on .053, both training sentences complete (§J.2). Anchored to the previous round's weights, weight interpolation keeps both measurements, .447 against .493 on bfloat16 weights, and at the per-round weight interpolation's distance answers the paraphrase .187 more often (§K.2).

6 A Release Check for Maintained Facts, and the Strength Delivered

One added measurement, a held-out paraphrase against the pre-trained baseline, separates every float32 run that lost the paraphrase from every run that kept it; on bfloat16 weights the strength set is not the strength applied.

1. *The paraphrase item: each fact twice, against the pre-trained baseline.* The training sentence and a paraphrase, both against the pre-trained model, so a fact the update never installed does not count as lost; and the training sentence on round 0, since the curve certifies only a sentence the update put at ceiling (§D).
2. *The held-out item: facts held out of the rehearsal.* Measured on their training sentence, where what the rehearsal alone holds at ceiling falls (Figure 2b).

runs	n	curve at ceiling, round 30	paraphrase item stops	either stops
float32 master				
with interpolation, paraphrase lost	12	12	12	12
without interpolation	8	8	0	0
bfloat16 weights				
stabiliser delivered, paraphrase kept	4	4	0	0

Table 4: *The retention curve alone would release every run that has lost the fact; the paraphrase item stops all 12 of them, none of the 8 runs without weight interpolation, and none of the 4 stabilised runs that keep the paraphrase.* The release check run as one protocol over every run on the float32 master whose paraphrase falls under weight interpolation, each beside its run without weight interpolation: 12 runs from eight experiments on three model families, four materials and two ranks, at strengths up to $\lambda = .044$, and their 8 twins. A run counts as stopped on a majority of its three builds; the paraphrase item stops a run whose paraphrase has fallen more than .074 from round 0 or whose round-0 training sentence is below ceiling. The third row is the 4 bfloat16 runs on which a stabiliser is delivered and whose own experiment found no loss on the paraphrase: the Fisher-weighted penalty, the previous-round anchor and the fresh and the stale anchor of §5.

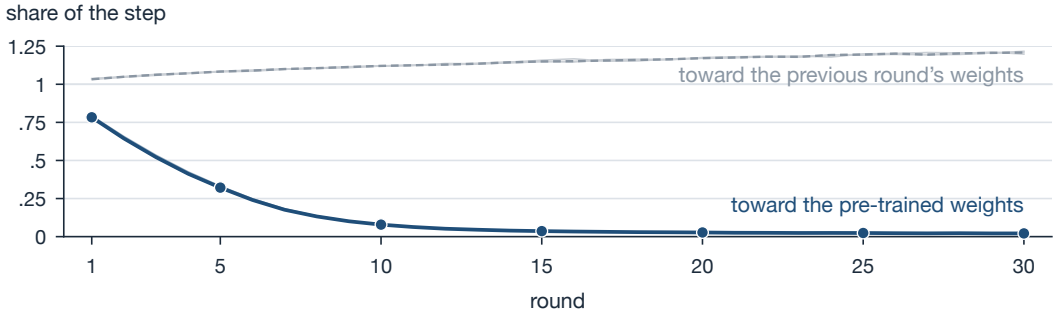

Figure 3: *In place on bfloat16 weights a per-round weight interpolation toward the pre-trained model delivers a small and falling share of its step; toward the previous round's weights it is delivered in full.* The share of its nominal step that weight interpolation at $\lambda = .15$ delivers on rounds 1 to 30 of the run's own trajectory, over all 252 adapted modules; three seeds (thin lines) and their median.

3. *The chat-template item: the deployment measurement.* The training sentence asked through the chat template the pipeline serves, itself a wording the run never rehearsed (Table 2).
4. *The delivered-strength item: the stabiliser strength delivered, not set.* The run's round-30 displacement as a share of the run without weight interpolation's, .27 at $\lambda = .044$ on every float32 run measured (Table 1); on bfloat16 weights, the share below.

Applied to every judged float32 run, the paraphrase item separates the runs that lost the paraphrase from the runs that kept it. Table 4 runs the four items over the 12 runs of §§3–5 whose paraphrase falls and their 8 runs without weight interpolation: the curve is at ceiling on every one at round 30 and would release them all; the paraphrase item stops the 12 and none of the 8. The check is not a test for the presence of a stabiliser: four bfloat16 runs carry a delivered stabiliser and keep the paraphrase, and the paraphrase item stops none of them. The loss is not confined to the wording the check reads: on the main run it is on four wordings the check never asks, by .367, .327, .193 and .187 (§3).

On bfloat16 weights the strength delivered is a closed form, and the library defaults are inside it. A per-round weight interpolation applied in place to bfloat16 weights delivers .78 of its nominal step on the first round and .02 on the thirtieth (Figure 3). Each round rounds the interpolated weight to the bfloat16 grid; an element whose move is under a fixed share c of the grid spacing stays put on every later round, so the norm an update retains is a closed form in c (Equation 1, §J), which, fitted at $c = .418$ on one update, reproduces the retention of updates it never saw within 3.9%. The same stall is in the training stacks: on this paper's own LoRA displacements, PyTorch's exponential moving

average at its default decay delivers .016 of the intended movement into bfloat16 over thirty updates, against 1.000 into float32 (the closed form puts it at .014); TRL's reference-model synchronisation is bitwise this paper's interpolation, and at its default coefficient of .6 it delivers .963. The threshold is the stagnation of round-to-nearest (Gupta et al., 2015; Zamirai et al., 2021; Croci & Giles, 2023), known per step on the optimiser's updates (Miahi & Belilovsky, 2026; Shang, 2026); this paper adds its closed form for an operator re-applied thirty times to the same weights. In place on bfloat16 weights the learning of thirty rounds writes .013 of the run's stored displacement, against .857 with no interpolation, while the curve stays at ceiling (§J): the strength set is not the strength applied.

7 Related Work

The wording gap after one update. A fact written in one sentence is less available in other wordings after that single update, with no stabiliser (Allen-Zhu & Li, 2024a;b; Berglund et al., 2024; Dai et al., 2026). Chang et al. (2024) track the trained sequence and a paraphrase separately along one injection of fictional facts; Gekhman et al. (2024) sort a fine-tune's facts by what the model knew before the update; O'Neill (2026) follow invented facts through later updates with no rehearsal and a periodic distillation into the original model, and find forgotten facts keeping the log-probability their update added. Wording moves a score on a fixed model (Sclar et al., 2024; Liu et al., 2026), and knowledge editing scores an edit's own prompt and its paraphrase after the update (Zhu et al., 2020; De Cao et al., 2021; Mitchell et al., 2022; Meng et al., 2022; 2023; Zhong et al., 2023; Cohen et al., 2024; Yao et al., 2023).

Where retention is scored. Of 23 stabiliser and continual-learning papers, 21 score it on a held-out sample of the trained tasks, a fixed benchmark or the training distribution, as production continual pre-training does (Chen et al., 2026b), one on unseen instruction wordings (Scialom et al., 2022), and none on another wording of a kept item (§L.1). Lifelong editing re-scores the edit prompt and its paraphrase as two aggregate axes along the sequence: Gupta et al. (2025) bind each update matrix as it is written, Gu et al. (2026) normalise the running update, Wang et al. (2025) keep a sparse residual memory, Thede et al. (2025) merge between time steps, and Wang et al. (2024) and Hartvigsen et al. (2023) route edits to a side memory beside untouched main weights, where reliability holds while generalisation falls as a property of the editor. Continual knowledge learning probes updated facts on cloze sentences built apart from the training text (Jang et al., 2022b;a). A concurrent preprint ranks a periodic consolidation against cumulative replay on a Wikidata stream with a held-out paraphrase among its scores (Choi, 2026); no run there is held near the pre-trained weights. This paper adds the per-fact, per-round pairing under a stabiliser, the chat-template measurement of the training sentence, and a release check run over every judged run.

Replay and its held-out set. Class-incremental learning knows that a model trained with rehearsal overfits its sample memory (Verwimp et al., 2021; Bonicelli et al., 2022), interpolates after each task toward the model before it (Kozal et al., 2024), and scores retention on a held-out test set, never on the buffer; this paper measures a maintained fact's one held-out sample, another wording of itself. Spurious forgetting is the mirror case, a task score falling while the knowledge is kept (Zheng et al., 2025; Kotha et al., 2024; G. Jiang et al., 2025).

8 Conclusion and Scope

A retention curve measured over the sentences a run rehearses certifies the rehearsal. Under a per-round weight interpolation, a periodic merge or later updates with no stabiliser, the curve returns to ceiling every round while the same fact is lost in every wording the run never rehearsed, its own training sentence through the chat template included; the rehearsal rebuilds the sentence it repeats and nothing else. Both release checks tolerate a stronger stabiliser than the fact can bear and cannot tell one stabiliser family from another, so a pipeline should measure every maintained fact in a wording it never rehearsed, against the pre-trained baseline, with facts held out of the rehearsal, and report the stabiliser strength delivered rather than set.

The instrument is small and controlled, 150 invented facts after Allen-Zhu & Li (2024a), so that the pre-trained baseline is known and one update puts every training sentence at ceiling. On real documents the update leaves the training sentence below ceiling and the curve reports the loss (§K).

Statement on the Use of Large Language Models

A large language model assistant, Claude (Anthropic, 2026), was used in this work, to the authors' specification and under their review. It assisted in writing and editing the paper and drafted parts of it; it searched and summarised related literature and formatted references; and it wrote the training, probing and analysis code and the scripts that generate the tables. The training sentences of the real-document facts of §K were drafted by the assistant from the source reports and checked by the authors; the facts and the probes come from the documents themselves (§K.1). The authors reviewed all AI-assisted material and take responsibility for the paper's content.

Ethics Statement

The first material is 150 hand-written facts that bind invented people to years; it names no real person and involves no personal data, no scraped corpus and no human subjects. The second material is 116 facts about public events reported from June 2025 onward; it is written into the models for the real-document facts measured in §K. Its training sentences are a language model's rephrasings of public documents (most of them news reports or encyclopaedia entries), and each probe is a short excerpt of a further such document. The documents remain with their publishers; we release each document's publisher, its URL and its month of publication with the repository, from which the material can be rebuilt and its screening checked. This material concerns events already public, and involves no human subjects. The third material is 150 facts about buildings taken from Wikidata (§D.6), a public knowledge base, with sentences rendered by fixed rules; it involves no personal data. The fourth is 150 edit requests from the public CounterFact and zsRE benchmarks, measured on the benchmarks' own paraphrase prompts (§D.8). The models are public checkpoints used under their own licences. The paper's practical point is a safeguard for evaluation: a maintenance run is judged on its hit rate in a wording it never trained on, which shows a loss that the release check on the training sentence, measured by exact match, passes over.

Reproducibility Statement

Every experiment in this paper ships in the anonymous repository submitted as supplementary material, one directory per experiment: `experiment.json` (what it measures and the sections that cite it), `config.json` (the recipe and the command line), `results/` (one result file per arm and seed), `verdict.json` (each criterion's value, resolution, per-seed values and outcome) and `numbers.json` (the numbers the paper quotes from that experiment, each with the file it comes from). The central contrasts were judged by criteria fixed before the runs ran, and each run's verdict sits beside its result files (Table A23). `code/` is the training and probing code, with pinned requirements and a usage file; `data/` holds the invented facts, the Wikidata facts and the distractor sentences. §2 fixes the model, the material, the update, the dose, the schedule and both probes, and §A gives a round statement by statement. A seed fixes the order of the training rows and the trainer's random state; the LoRA initialisation is drawn separately for every run, so two builds of one seed are two updates of one recipe, and the resolution (§B.2) is measured between such rebuilds. The 116 real-document facts (§K.1) ship as per-item result files with the material's source list (Ethics Statement), so each number can be re-derived from the per-item products. A preprint reference names the arXiv version measured.

Reruns. Each round's adapter is kept, and a run is rebuilt from its adapters; a training run takes three or four cards for a few hours. A fresh environment reproduces a Mistral-7B run with weight interpolation within the bands of §B.2: the training sentence at round 30 within .047 and the paraphrase within .007.

References

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 1067–1077, 2024a.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.2, knowledge manipulation. *arXiv preprint arXiv:2309.14402v2*, 2024b. A version appears in the 13th International Conference on Learning Representations, 2025.

Anthropic. Claude. https://claude.ai, 2026. Interactive sessions, August to September 2026.

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

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John P. Cunningham. LoRA learns less and forgets less. *Transactions on Machine Learning Research*, 2024.

Lorenzo Bonicelli, Matteo Boschini, Angelo Porrello, Concetto Spampinato, and Simone Calderara. On the effectiveness of Lipschitz-driven rehearsal in continual learning. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

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

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *International Conference on Learning Representations*, 2019a.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K. Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486v4*, 2019b.

Howard Chen, Jiayi Geng, Adithya Bhaskar, Dan Friedman, and Danqi Chen. Continual memorization of factoids in language models. *Transactions on Machine Learning Research*, 2026a.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. Recall and learn: Fine-tuning deep pretrained language models with less forgetting. In *Conference on Empirical Methods in Natural Language Processing*, pp. 7870–7881, 2020.

Shengzhuang Chen, Jerrod Parker, Yejin Bang, Andrew M. Bean, Nabeel Seedat, Stefan Winzeck, Daniil Glazko, Jannik Zgraggen, Fangyi Yu, Scott Arnott, Dietrich Trautmann, Luca Ciuffreda, Guglielmo Bonifazi, Davide Romano, Bradley Bell, Kirsty Fielding, Daniele Giofrè, Tom Zielund, Ipshita Chatterjee, Sneha Murthy Ghantasala, Manpreet Nanreh, John Scoville, Maciej Sakowicz, Wassim Seifeddine, Lukas Thede, and Jonathan Richard Schwarz. Thomson: Continual learning of frontier models for SovereignAI. *arXiv preprint arXiv:2608.27147*, 2026b.

Heejin Choi. Beyond endpoint scores: Time- and capacity-conditioned evaluation of continual knowledge updating. *arXiv preprint arXiv:2609.03900*, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168v2*, 2021.

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

Matteo Croci and Michael B. Giles. Effects of round-to-nearest and stochastic rounding in the numerical solution of the heat equation in low precision. *IMA Journal of Numerical Analysis*, 43 (3):1358–1390, 2023.

Lu Dai, Ziyang Rao, Yili Wang, Hanqing Wang, Hao Liu, and Hui Xiong. Towards mechanistically understanding why memorized knowledge fails to generalize in large language model finetuning. *arXiv preprint arXiv:2607.08393v1*, 2026.

Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In *Conference on Empirical Methods in Natural Language Processing*, pp. 6491–6506, 2021.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, 1993.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031, 2021.

Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999.

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

Xiaojie Gu, Ziying Huang, Jia-Chen Gu, and Kai Zhang. UltraEdit: Training-, subject-, and memory-free lifelong editing in language models. *Transactions on Machine Learning Research*, 2026.

Akshat Gupta, Phudish Prateepamornkul, Maochuan Lu, Ahmed Alaa, Thomas Hartvigsen, and Gopala Anumanchipalli. Lifelong knowledge editing requires better regularization. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 22653–22675, 2025.

Suyog Gupta, Ankur Agrawal, Kailash Gopalakrishnan, and Pritish Narayanan. Deep learning with limited numerical precision. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *PMLR*, pp. 1737–1746, 2015.

Awni Hannun, Jagrit Digani, Angelos Katharopoulos, and Ronan Collobert. MLX: Efficient and flexible machine learning on Apple silicon. https://github.com/ml-explore/mlx, 2023. Version 0.32.0.

Thomas Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with GRACE: Lifelong model editing with discrete key-value adaptors. In *Advances in Neural Information Processing Systems*, 2023.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. In *Advances in Neural Information Processing Systems*, 2023.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Timm Hess, Gido M. van de Ven, and Tinne Tuytelaars. Forgetting of task-specific knowledge in model merging-based continual learning. *arXiv preprint arXiv:2507.23311v1*, 2025.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats L. Richter, Quentin Anthony, Timothée Lesort, Eugene Belilovsky, and Irina Rish. Simple and scalable strategies to continually pre-train large language models. *arXiv preprint arXiv:2403.08763*, 2024.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *Conference on Uncertainty in Artificial Intelligence*, 2018.

Matthew Jagielski, Om Thakkar, Florian Tramèr, Daphne Ippolito, Katherine Lee, Nicholas Carlini, Eric Wallace, Shuang Song, Abhradeep Thakurta, Nicolas Papernot, and Chiyuan Zhang. Measuring forgetting of memorized training examples. In *International Conference on Learning Representations*, 2023.

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. In *Conference on Empirical Methods in Natural Language Processing*, 2022a.

Joel Jang, Seonghyeon Ye, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, Stanley Jungkyu Choi, and Minjoon Seo. Towards continual knowledge learning of language models. In *International Conference on Learning Representations*, 2022b.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825v1*, 2023.

Gangwei Jiang, Caigao Jiang, Zhaoyi Li, Siqiao Xue, Jun Zhou, Linqi Song, Defu Lian, and Ying Wei. Unlocking the power of function vectors for characterizing and mitigating catastrophic forgetting in continual instruction tuning. In *International Conference on Learning Representations*, 2025.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020.

Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. Continual pretraining of language models. In *International Conference on Learning Representations*, 2023.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwińska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13): 3521–3526, 2017.

Suhas Kotha, Jacob Mitchell Springer, and Aditi Raghunathan. Understanding catastrophic forgetting in language models via implicit inference. In *International Conference on Learning Representations*, 2024.

Jędrzej Kozal, Jan Wasilewski, Bartosz Krawczyk, and Michał Woźniak. Continual learning with weight interpolation. *arXiv preprint arXiv:2404.04002v2*, 2024.

Saurabh Kumar, Henrik Marklund, and Benjamin Van Roy. Maintaining plasticity in continual learning via regenerative regularization. *arXiv preprint arXiv:2308.11958*, 2023.

Cheolhyoung Lee, Kyunghyun Cho, and Wanmo Kang. Mixout: Effective regularization to finetune large-scale pretrained language models. In *International Conference on Learning Representations*, 2020.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pp. 333–342, 2017.

Jeffrey Li, Mohammadreza Armandpour, Seyed Iman Mirzadeh, Sachin Mehta, Vaishaal Shankar, Raviteja Vemulapalli, Samy Bengio, Oncel Tuzel, Mehrdad Farajtabar, Hadi Pouransari, and Fartash Faghri. TiC-LM: A web-scale benchmark for time-continual LLM pretraining. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32231–32273, 2025.

Xuhong Li, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2825–2834, 2018.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, Hanze Dong, Renjie Pi, Han Zhao, Nan Jiang, Heng Ji, Yuan Yao, and Tong Zhang. Mitigating the alignment tax of RLHF. In *Conference on Empirical Methods in Natural Language Processing*, pp. 580–606, 2024.

June M. Liu, Shaomian Zheng, He Cao, Dingnan Jin, Qing Cui, and Jun Zhou. Improving cross-format robustness in language models with multi-format training. *arXiv preprint arXiv:2606.11643v1*, 2026.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

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

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. PEFT: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft, 2022. First released 2022; version 0.17.1, 2025.

Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pp. 109–165. Academic Press, 1989.

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

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

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Erfan Miahi and Eugene Belilovsky. Understanding and exploiting weight update sparsity for communication-efficient distributed RL. *arXiv preprint arXiv:2602.03839*, 2026.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed precision training. In *International Conference on Learning Representations*, 2018.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale. In *International Conference on Learning Representations*, 2022.

ML Explore. MLX LM: Generating text and fine-tuning language models with MLX. https://github.com/ml-explore/mlx-lm, 2025. Version 0.31.3.

Michael Noukhovitch, Samuel Lavoie, Florian Strub, and Aaron Courville. Language model alignment with elastic reset. In *Advances in Neural Information Processing Systems*, 2023.

Charles O'Neill. Can a language model learn facts continually in its weights? *arXiv preprint arXiv:2607.11020v2*, 2026.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: an imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, 2019.

Mansheej Paul, Brett Larsen, Connor Jennings, and Cody Blakeney. Characterizing datasets and building better models with continued pre-training. https://www.databricks.com/blog/characterizing-datasets-and-building-better-models-continued-pre-training, 2024. Databricks Blog, November 21, 2024.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing*, pp. 2463–2473, 2019.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388v1*, 2025.

Alexandre Ramé, Johan Ferret, Nino Vieillard, Robert Dadashi, Léonard Hussenot, Pierre-Louis Cedoz, Pier Giuseppe Sessa, Sertan Girgin, Arthur Douillard, and Olivier Bachem. WARP: On the benefits of weight averaged rewarded policies. *arXiv preprint arXiv:2406.16768v1*, 2024.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*, 2019.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. Fine-tuned language models are continual learners. In *Conference on Empirical Methods in Natural Language Processing*, pp. 6107–6122, 2022.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *International Conference on Learning Representations*, 2024.

Zekai Shang. Auditing invisible weight updates with reference traces. *arXiv preprint arXiv:2607.09800v4*, 2026.

Zafir Stojanovski, Karsten Roth, and Zeynep Akata. Momentum-based weight interpolation of strong zero-shot models for continual learning. In *NeurIPS Workshop on Interpolation Regularizers and Beyond*, 2022.

Team OLMo. 2 OLMo 2 Furious. *arXiv preprint arXiv:2501.00656v3*, 2025. A shorter version appears in the Conference on Language Modeling (COLM) 2025.

Lukas Thede, Karsten Roth, Matthias Bethge, Zeynep Akata, and Tom Hartvigsen. WikiBigEdit: Understanding the limits of lifelong knowledge editing in LLMs. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Eli Verwimp, Matthias De Lange, and Tinne Tuytelaars. Rehearsal revealed: The limits and merits of revisiting samples in continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9385–9394, 2021.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformer reinforcement learning. https://github.com/huggingface/trl, 2020. Version 1.14.0.

Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

Ke Wang, Yiming Qin, Nikolaos Dimitriadis, Alessandro Favero, and Pascal Frossard. MEMOIR: Lifelong model editing with minimal overwrite and informed retention for LLMs. In *Advances in Neural Information Processing Systems*, 2025.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. WISE: Rethinking the knowledge memory for lifelong model editing of large language models. In *Advances in Neural Information Processing Systems*, 2024.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23965–23998, 2022a.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo-Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7949–7961, 2022b.

Eric Wu, Kevin Wu, and James Zou. FineTuneBench: How well do commercial fine-tuning APIs infuse knowledge into LLMs? *arXiv preprint arXiv:2411.05059*, 2024.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, 2023.

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

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Pedram Zamirai, Jian Zhang, Christopher R. Aberger, and Christopher De Sa. Revisiting BFloat16 training. *arXiv preprint arXiv:2010.06192v2*, 2021.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3987–3995, 2017.

Junhao Zheng, Xidi Cai, Shengjie Qiu, and Qianli Ma. Spurious forgetting in continual learning of language models. In *International Conference on Learning Representations*, 2025.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Conference on Empirical Methods in Natural Language Processing*, pp. 15686–15702, 2023.

Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix Yu, and Sanjiv Kumar. Modifying memories in transformer models. *arXiv preprint arXiv:2012.00363v1*, 2020.

A What Is Trained and What Is Asked: the Material, One Round, One Probe

This appendix answers §2: the material, with three facts printed verbatim, the statements of a round in order, and the rule by which a probe decides one item.

The material. The material is 150 facts, each binding an invented person to a four-digit year, hand-written one at a time. The years run from 1701 to 1999, no year is used twice, and each year occurs exactly once inside its own sentence, so the cut that builds the trained-sentence probe is unambiguous. The sentences are written in twelve registers, so no two have the same structure: news, diary, obituary, court record, guidebook, letter, ledger note, oral history, inventory, sermon, almanac, gossip. Each fact carries one paraphrase, also written by hand and in a different register from its sentence; no probe shares an 8-gram of lower-cased words with its own sentence, each names the person, and every probe ends before its answer.

Table A1 prints three of the 150 facts verbatim, rows 1, 8 and 41 of the material in file order, with both prefixes.

One round, as statements. A round is five statements in this order.

1. *Train a fresh adapter on the round's rows.* The adapter is a LoRA (Hu et al., 2022) of rank 8 with $\alpha = 160$, which is a scale of 20, the default of `mlx_lm` (ML Explore, 2025), the language-model package built on MLX (Hannun et al., 2023); its version and that of `peft` (Mangrulkar et al., 2022) are pinned. `mlx_lm` takes the scale directly and `peft` derives it as α/r, so a rerun under `peft` sets $\alpha = 160$. Dropout 0; attached to all seven projections of every layer (`q`, `k`, `v`,

Item 1	
trained	`Ansel Marrow, who never once left the parish, was elected bell-warden in` **`1701`** `and rang the curfew for forty years after.`
trained probe	`Ansel Marrow, who never once left the parish, was elected bell-warden in ___`
untrained	`A quiz for the local history society: Ansel Marrow became bell-warden in the year ___`
Item 8	
trained	`Dear sister, you ask about the mill; Hettie Swann has had it since` **`1718`** `and will not sell.`
trained probe	`Dear sister, you ask about the mill; Hettie Swann has had it since ___`
untrained	`Hettie Swann came into possession of the mill in ___`
Item 41	
trained	`Obadiah Frost predicted nothing correctly except the frost of` **`1794`**`, which he predicted to the day, and dined on it forever after.`
trained probe	`Obadiah Frost predicted nothing correctly except the frost of ___`
untrained	`Obadiah Frost's one correct prediction was the frost of ___`

Table A1: Three of the 150 facts, verbatim, with the two probe prefixes. The *trained* row is the sentence the run rehearses, with the answer year in bold; the *trained probe* is that sentence cut at the character where its year begins; *untrained* is the hand-written paraphrase. Both prefixes end in the space before the answer. The pre-trained model answers 5 of the 150 training sentences and 3 of the paraphrases, the floor every hit rate in this paper stands on.

`o`, `gate`, `up`, `down`); batch 4; sequences truncated at 256 tokens and left unpacked; constant learning rate with no warmup; AdamW (Loshchilov & Hutter, 2019); compute in bfloat16. The seed is the run's seed offset plus the round index, so two runs of the same seed see the same order of rows on the same round.

2. *Merge it away.* The adapter is merged into the weights and takes no further part in the run, so the round's entire update is that adapter's low-rank product and the weights are all that carries across rounds. Every round's adapter is archived, and a run can be rebuilt from its 31 adapters.
3. *Interpolate toward the pre-trained weights.* In place, tensor by tensor, $w \leftarrow (1-\lambda)\, w + \lambda\, \theta_0$, with the anchor θ_0 streamed from the pre-trained model's own shards. $\lambda = .15$ on the runs that interpolate in place on bfloat16 weights, $.044$ on the main run's float32 master (§2), and 0 on the run without weight interpolation, where the statement is the identity. The anchor is the pre-trained weights on all 31 interpolations, round 0 included.
4. *Write one checkpoint* and delete the previous round's.
5. *On a measurement round, take the measurements* (below).

Round 0 is the same five statements with two constants changed: 900 optimiser steps at 3e-5, whereas later rounds take 90 at 3e-6. It is interpolated toward the pre-trained weights like every other round, against the same anchor.

What differs between one round and the next. Every round rehearses the same 150 sentences at the same dose; the variant with interference (§E) adds six never-probed distractor rows a round from a separate pool.

When a measurement is taken. A run is measured at the end of a measurement round, after weight interpolation, on rounds 0, 1, 3, 7, 14 and 30. The two-step measurements of §F replay the stored per-round adapters and measure the weights after the merge and again after weight interpolation.

How a probe decides one item. Both probes run one procedure on two prefixes, so a difference between them comes from the prefix. The procedure is the answer the checkpoint gives when asked: it continues the prefix greedily for at most 12 new tokens, and the item is a hit when the continu-

ation, stripped and lower-cased, *opens with* the gold span (for a four-digit year, all four digits). A measurement is the count of such hits out of 150. The trained-sentence prefix is the training sentence cut at the character where its year begins; the untrained-wording prefix is the hand-written probe.

B Are the Differences Larger Than the Spread Between Two Builds of One Recipe?

This appendix carries §2's rule for judging a contrast: the interval that separates two runs, and the margin, set by how far two builds of one recipe differ, inside which a difference counts as none.

B.1 Every contrast with its interval

Each contrast of Table A2 separates its two runs by an interval that crosses both sources of spread: the items are paired across the two runs and resampled inside every seed, and the seeds are resampled with them, ten thousand times, a percentile bootstrap (Efron & Tibshirani, 1993). A run's adapter is initialised afresh for each seed, so the runs of a seed are separate updates of one recipe, paired by item and by seed. Every interval of the table excludes zero.

contrast	first	second	difference	95% interval
the run of Table A3	.127	.484	-0.358	$[-0.407, -0.309]$
reordered words	.136	.462	-0.327	$[-0.400, -0.256]$
held-out facts, training sentence	.247	.993	-0.747	$[-0.827, -0.660]$
public-knowledge-base facts	.067	.502	-0.436	$[-0.491, -0.380]$
benchmark edits, LoRA update	.109	.418	-0.309	$[-0.360, -0.258]$
benchmark edits, MEMIT update	.020	.616	-0.596	$[-0.640, -0.551]$
Mistral-7B, $\lambda = .10$	.249	.696	-0.447	$[-0.504, -0.387]$
OLMo-2-7B, $\lambda = .15$	.184	.429	-0.244	$[-0.318, -0.180]$
float32 master, matched distance	.260	.490	-0.230	$[-0.278, -0.182]$
interp. alone − per-round interp., paraphrase	.200	.127	$+0.073$	$[+0.027, +0.124]$
interp. alone − per-round interp., training sentence	.533	1.000	-0.467	$[-0.522, -0.411]$
float32 master, per-round interp. − interp. alone	.289	.200	$+0.089$	$[+0.022, +0.149]$
per-round interp. − interp. every ten rounds	.113	.244	-0.131	$[-0.193, -0.076]$
per-round interp. − one interp., ten times the dose	.151	.398	-0.247	$[-0.302, -0.191]$
per-round interp. − one interp., the standard dose	.151	.209	-0.058	$[-0.113, -0.007]$
the same gap, training sentence	.998	.536	$+0.462$	$[+0.416, +0.509]$
anchor on the previous round's weights − per-round interp.	.451	.138	$+0.313$	$[+0.264, +0.364]$
Fisher-weighted penalty − per-round interp.	.371	.113	$+0.258$	$[+0.204, +0.311]$

Table A2: The contrasts the paper measures with a 95% interval: the items are paired across the two runs and resampled inside every seed, and the seeds are resampled with them, 10,000 times. The first nine rows are the run with weight interpolation minus the run without weight interpolation of the setting named; the rest name their two runs. *First* and *second* are the two runs' round-30 hit rates averaged over seeds, on the paraphrase unless the row names the training sentence. The float32 run matched on distance (§J.2) enters twice: its matched-distance row averages the six seeds it ran on, against its own run without weight interpolation, and its row against weight interpolation alone the three seeds weight interpolation alone ran on; §J.2 prints its six-seed median. *Main conclusion: the paraphrase of the run with weight interpolation lies below its run without weight interpolation's in all nine settings, and at one distance the training sentence separates the run with weight interpolation from weight interpolation alone and from one interpolation by more than .4.*

B.2 The resolution

The band is measured on the coarser of two scales: computed over the two runs' items as two samples, the float32 contrast at one distance resolves .049, and computed over the paired items it resolves .034; a contrast that clears the coarser band clears the paired one. The band is empirical: 38 groups of runs, 127 runs in all, share a recipe to the last field and were launched separately

	λ_{eff}		seed			three seeds	
			seed 1	seed 2	seed 3	mean	95% item interval
150 invented-entity facts, 30 rounds, no new sentences, Qwen3-8B							
$\lambda = .15$	.020	trained	1.000	1.000	1.000	1.000	1.000 to 1.000
		untrained	.127	.113	.140	.127	.087 to .171
$\lambda = 0$	0	trained	1.000	1.000	1.000	1.000	1.000 to 1.000
		untrained	.480	.493	.480	.484	.416 to .553
no interpolation − interpolation		untrained	.353	.380	.340	.358	.309 to .407
interpolation, on the 72, 74 and 72 facts without it		untrained	.264	.216	.264	.248	—
no interpolation, on the 19, 17 and 21 facts with it		untrained	1.000	.941	.905	.949	—
reordered, λ=.15	.020	untrained	.127	.133	.147	.136	—
reordered, λ=0	0	untrained	.513	.453	.420	.462	—
held-out, λ=.15	.020	trained	.200	.300	.240	.247	—
held-out, λ=0	0	trained	1.000	1.000	.980	.993	—

Table A3: Round-30 hit rates seed by seed for the run with weight interpolation ($\lambda = .15$; λ_{eff} is weight interpolation it delivers on bfloat16 weights, §J) and the run without weight interpolation ($\lambda = 0$). The *on the facts* rows measure each run at round 30 on the facts the other run answers; the run with weight interpolation answers at most two facts a seed that the run without it does not. The *reordered* rows measure the training sentence's own content words in another order, and the *held-out* rows measure the training sentence on 50 facts the rehearsal leaves out. The last two columns are the three-seed mean and a 95% item-level interval, on the contrast row the interval of the paired difference (§B.1). *Main conclusion: with weight interpolation on, the run completes the training sentence and answers the same fact in a paraphrase about a quarter as often as with it off, on bfloat16 weights as on the float32 master of Table 1; the sentence's own words in another order, and a fact the rehearsal leaves out, fall with the paraphrase.*

on different machines, so the spread inside a group is the spread between two builds of the same measurement. The .074 of §2 is the 95% band of the difference between two such runs on the paraphrase, from a within-group standard deviation pooled over the census's 89 degrees of freedom.

The band at each stratum of the measurement: on the paraphrase it is .046 over the 10 groups whose round-30 rate is below .2, .085 over the 24 groups from .2 to .5 and .076 over the 4 groups at .5 and above. On the training sentence the 33 groups at the ceiling give .010 and the 2 groups just below it .054; the pooled .148 comes from the 3 groups away from the ceiling, so .010 is the band for a measurement at the ceiling and .148 for one away from it. Fixed-text perplexity has a line of its own, from four pairs of independent rebuilds of one recipe, each pair sharing its seed, which differ by .049, .117, .182 and .220: two of the widest such differences, $2 \times .220 = .44$, taken as .50, so two runs within .50 of each other on fixed-text perplexity count as alike on the panel. The main run's contrast clears the band build by build: its six builds with weight interpolation measured at most .160 on the paraphrase and its nine without it at least .420, every training sentence complete on all fifteen.

C Every Seed Shows the Loss on the Paraphrase

This appendix answers §3 seed by seed.

Each column headed by a four-digit number in the tables of the appendices (8100, 8200, 8300; 8700, 8800, 8900 on the benchmark edits) prints the measurement of that seed.

C.1 Thirty rounds with no new sentence

The loss falls on facts the run without weight interpolation answers: Table A3 measures each run on the facts the other answers, every seed, and a difference inside the band of §2 counts as none. The held-out rows of Table A3 are the run of Figure 2b on bfloat16 weights at the nominal $\lambda = .15$, rehearsing 100 of the 150 facts and measured on the 50 left out: it completes their training sentence on .24 under weight interpolation against 1.00 without it, and answers their paraphrase on .140, a

difference of .480 from the run without weight interpolation. On the main run (Figure 2b) the same 50 facts are answered in the paraphrase on .12 under weight interpolation against .54 without it.

C.2 The wording ladder, every wording

Table A4 measures the two runs of Table 1's first row on ten wordings of each fact and on the training sentence asked through the chat template. Weight interpolation takes every wording the update installs above the pre-trained floor: the trained words in another order fall by .367 and the paraphrase by .353, the two tiers below by .327 and .193, the wh-question by .187, and the fourth overlap tier and the instruction form by .127 and .107, on every seed. The two it does not install, the wording sharing no content word with the training sentence and the name alone, lie at the floor on both runs; the training sentence is at ceiling on both runs on every seed and, asked through the chat template, falls from .560 to .087 (row *chat*; §L).

				no interp. ($\lambda = 0$)			interp. ($\lambda = .044$)			
wording	what it is	overlap	floor	8100	8200	8300	8100	8200	8300	gap
S	the training sentence	1.000	.027	1.000	1.000	1.000	1.000	1.000	1.000	.000
chat	training sentence, chat	1.000	.013	.627	.520	.560	.087	.067	.100	.460
T1	trained words reordered	1.000	.060	.560	.493	.447	.100	.127	.113	.367
P	the paper's wording	.735	.020	.473	.480	.473	.113	.147	.120	.353
T2	tier 2	.652	.020	.440	.413	.407	.087	.093	.080	.327
T3	tier 3	.428	.020	.287	.247	.247	.060	.053	.053	.193
instr	instruction form	.391	.027	.160	.113	.153	.047	.047	.047	.107
whq	wh-question	.311	.040	.207	.227	.220	.020	.053	.033	.187
T4	tier 4	.213	.033	.147	.153	.153	.020	.027	.033	.127
T5	no content word shared	.000	.027	.033	.033	.040	.027	.027	.020	.007
nameonly	name only	.004	.007	.007	.000	.000	.000	.007	.000	.000

Table A4: The wording ladder on the two runs of Table 1's first row, float32 master, round 30: hit rates over the same 150 facts on the run without weight interpolation, the run with weight interpolation and, as *floor*, the pre-trained model, every seed. *Overlap* is the wording's mean content-word overlap with the training sentence, from the frozen tier build. *Gap* is the run without weight interpolation minus the run with it, the median of the seeds; each wording's gap resolves at .145, and the mean gap over the six wordings below the training sentence, taken within each seed, is .211 at the median seed, at .088. *Main conclusion: the training sentence is at ceiling on both runs, and every wording the update installed above the floor is lower under weight interpolation, the trained words in another order by as much as the paper's paraphrase; the two wordings at the floor stay there on both runs.*

C.3 The benchmark panel

Table A5 prints fixed-text perplexity and three accuracies, MMLU, HellaSwag (Zellers et al., 2019) and GSM8K (Cobbe et al., 2021), on the two ends, $\lambda = 0$ and the nominal $\lambda = .15$, of a bfloat16 ladder of the run of Table A3's recipe. At round 30 the run without weight interpolation measures 1.391 above the pre-trained model's 10.499 at the median seed and the run with weight interpolation 1.130, a difference of .26, under the panel's line of .50 (§B.2). The run also leaves in place both wordings of 62 facts Qwen3-8B already holds. The update is what raises perplexity, and the rounds of rehearsal bring most of it back on both runs: the pre-trained model with a round-0 adapter of this recipe measures 16.96, 18.35 and 16.82 on three draws of the update, 4.6 to 6.6 above the round-30 measurements of the run without weight interpolation.

D Does the Loss on the Paraphrase Hold Under Other Schedules, Models, Materials and Updates?

Table A6 carries the rows of Table 1's shape that the body measures in passing: the second invented set and the benchmark edits on the float32 master, the five batches, and the updates that leave the training sentence below ceiling, where the curve reports the loss itself.

This appendix backs §3 schedule by schedule and material by material.

Number format. The measurements below are on bfloat16 weights at the nominal $\lambda = .15$, where weight interpolation delivered is the measured λ_{eff} of §6, unless they name the float32 master.

D.1 A RANK-64 UPDATE

The run of Table 1's first row is run again with a LoRA update of rank 64 at the same scale, on the same 150 facts and the same three seeds, on the float32 master at $\lambda = .044$. The update is 2.54 times the size of the rank-8 update and installs the paraphrase on .687 of the facts at round 0, against .480 at rank 8 (Table A7). Under weight interpolation every training sentence is at ceiling on every seed, and the paraphrase measures .327, .360 and .367 against .640, .667 and .687 without weight interpolation: a difference of .313 at the median seed, at a resolution of .059, beside the rank-8 run's .353. The run with weight interpolation ends at .272 of the displacement of the run without weight

		$\lambda = 0$			$\lambda = .15$		
measurement	pre-trained	8100	8200	8300	8100	8200	8300
WikiText perplexity	10.499	11.891	11.746	12.253	11.627	11.629	11.797
minus the pre-trained model's	—	+1.391	+1.247	+1.753	+1.127	+1.130	+1.298
MMLU, 500 items	.694	.692	.692	.712	.700	.694	.700
HellaSwag, 500 items	.712	.716	.712	.704	.734	.726	.726
GSM8K, 200 items	.895	.880	.895	.900	.875	.880	.895

Table A5: The benchmark panel at round 30 on the two ends, $\lambda = 0$ and the nominal $\lambda = .15$ in place, of a bfloat16 ladder of the run of Table A3's recipe, measured by the same scorer as the pre-trained model. The panel is the release check a practitioner runs: 500 MMLU, 500 HellaSwag and 200 GSM8K items, resolving .0205, .0205 and .0230. The prices this paper reports are measured on the full sets: .0104 over all 14,042 MMLU items on the five-batch schedule (§5), .0332 on 150 rounds of arrival at a resolution of .0139. Every run stays within .022 of the pre-trained model on the three accuracies. *Main conclusion: on the benchmark panel the run with weight interpolation is no worse: the three accuracies stay within .022 of the pre-trained model on both runs, and perplexity is lower under weight interpolation.*

		paraphrase		training sentence		share kept		
run	λ_{eff}	without	with	without	with	weights	gain	shown by
pre-trained Qwen3-8B, 150 invented		.020		.033				
five batches of 30, per-round interp.	.013	.213	.080	1.000	1.000	—	—	paraphrase
further invented facts	.044	.160	.020	1.000	1.000	.269	.069	paraphrase
CounterFact/zsRE edits	.044	.387	.067	1.000	1.000	—	—	paraphrase
600 CounterFact/zsRE edits	.044	.407	.083	1.000	1.000	—	—	paraphrase
four wordings rehearsed, a fifth held out	n/a	.793	.293	1.000	.673	—	—	curve
five facts a round, per-round interp.	.009	.100	.053	1.000	.733	—	—	curve
CounterFact/zsRE edits, MEMIT update	edit	.613	.020	1.000	.873	—	—	curve
116 real-document facts, ×25	.050	.388	.121	.379	.172	—	—	curve
float32 master, $\lambda = .15$, past ceiling	.15	.473	.033	1.000	.307	—	—	curve
previous-round anchor	in full	.493	.447	1.000	1.000	—	—	neither

Table A6: The rows of Table 1's shape that the body cites in passing: round-30 hit rates, medians of three seeds, on Qwen3-8B's 150 invented facts unless named; *without* removes the row's stabiliser (for the five batches, the one batch written at round 0), and *shown by* names the measurement that shows the loss. λ_{eff} is weight interpolation delivered a round, nominal on the float32 master and measured on bfloat16 weights (§J); rows with a numeric bfloat16 value run at the nominal $\lambda = .15$. *Weights kept* and *gain kept* as in Table 1. *Main conclusion: on a second invented set and on benchmark edits, at 150 and at 600 of them, the retention–paraphrase gap recurs; where the update leaves the training sentence below ceiling, under a MEMIT edit, four rehearsed wordings or real documents, the curve reports the loss itself; a weight interpolation anchored to the previous round's weights loses neither measurement.*

interpolation, the rank-8 run at .275: weight interpolation takes the same share of an update eight times as wide, and the fact keeps twice the share of the gain.

What the update keeps beyond the arithmetic holds at two further strengths: the paraphrase ends .164 above the line the arithmetic draws from the arm's own round-0 measurement at $\lambda = .033$, every training sentence at ceiling, and .163 above it at $\lambda = .072$ (Table A7, last two rows).

D.2 Weight interpolation's arithmetic on the weights and on the fact

Table A7 rereads the six float32 configurations at $\lambda = .044$ against the arithmetic of weight interpolation alone: weight interpolation leaves $(1 - \lambda)^{31} = .25$ of a displacement after thirty-one rounds, and the rehearsal, which returns the training sentence only, adds nothing to the paraphrase. On the weights the arithmetic holds in every configuration, the run with weight interpolation ending at .27 of the displacement of the run without weight interpolation, the largest departure over the eighteen arms .008. On the fact the main run sits on the arithmetic to three decimals, and the five other configurations depart from it by between $-.073$ and $+.218$; at one share of the weights the share of the fact runs from .069 to .579, the hit rate's version of what the margin columns show below.

On the margin columns of Table A7 the main run's update raises the paraphrase $+2.414$ nats above the pre-trained model and the run with weight interpolation ends at -4.619, a departure of -5.223 nats from the arithmetic against a band of 1.63. The gold answer's own term moves by -4.379 nats and the strongest rival's by $+.854$: the answer falls, and no rival rises to take its place. Mistral-7B ends $+.917$ nats above the arithmetic, the four other configurations below it and the rank-64 update on it at $-.046$; the six departures run from -5.223 to $+.917$ nats. The margins are measured on H100 80GB cards, on which this float32 recipe measures as on the A100.

A second set of invented facts. 150 further person-to-year facts about new people, built by a separate procedure to the shape of the first set (one training sentence and one untrained-wording probe per fact, four-digit years, the probes' content-word overlap matched in distribution to the first set's) and measured on three new seeds, is the row of Table A6 headed *further invented facts*.

D.3 Merging toward the pre-trained weights every fifth round

A run with no per-round weight interpolation that merges toward the pre-trained weights every fifth round, weight interpolation of §2 at $\alpha = .3$ applied in place at the end of a merge round, gives the merge row of Table 1, measured at round 62, two rounds after the merge of round 60: a merge lowers both measurements, and the two rehearsal rounds that follow return the training sentence to ceiling and leave the paraphrase where the merge put it.

D.4 Arrival five a round with no stabiliser

Five facts arrive a round, on a seed-drawn permutation of the 30 rounds, with the per-fact update dose and the per-row rehearsal dose matched item for item to Table A3; every measurement here is on the arriving facts themselves, not on the static set that §3 measures at .313 against .513 on bfloat16 weights (Table A6). Under weight interpolation the paraphrase measures .053 and the training sentence leaves ceiling, .787, .733 and .693 on the three seeds. Facts written on arrival and never rehearsed lose their training sentence to the later updates, .813 of them at the median seed; rehearsed every round, every one keeps it.

Measured by the round each fact arrived. Split by arrival round on the run without weight interpolation, a fact that arrived in the first ten rounds and was rehearsed for the rest of the run, and one that arrived in the last ten, are answered in the paraphrase on the same .100 at round 30, the first seen 85.2 times over the run and the last 37.2; on the round it arrives, after that round's update and rehearsal, a fact is answered in the paraphrase on .173, .140 and .127 across the three terciles while the training sentence is complete. On this schedule an arriving fact's paraphrase is where its update puts it, and the count of rehearsals does not move it.

The loss on the paraphrase at the update, and three updates that leave it in place. Measured right after a round's five-fact update and before that round's rehearsal, the paraphrase is .100, .127 and .107 on the three seeds (resolution .039), against .277 for the midpoint between one update of all 150 facts, measured the round it is written, and the run's round-30 measurement; from its update

	round 0	round 30, paraphrase				
run	without	with	arithmetic	departure	fact kept	weights kept
main run (Qwen3-8B, rank 8)	.480	.120	.135	0.000	.250	.275
Mistral-7B	.693	.427	.208	+0.218	.579	.269
OLMo-2-7B	.420	.173	.125	+0.052	.386	.271
Wikidata facts	.467	.053	.127	−0.073	.088	.270
further invented facts	.200	.020	.055	−0.035	.069	.269
rank-64 update	.687	.360	.187	+0.170	.510	.272
rank-64 update, $\lambda = .033$	.707	.440	—	+0.164	—	.368
rank-64 update, $\lambda = .072$	.673	.253	—	+0.163	—	.128

	margin, nats		
run	update	round 30	departure
main run (Qwen3-8B, rank 8)	+2.414	−4.619	−5.223
Mistral-7B	+4.739	+1.932	+0.917
OLMo-2-7B	+0.641	−2.351	−2.612
Wikidata facts	+2.030	−3.538	−4.022
further invented facts	−1.743	−3.824	−3.535
rank-64 update	+4.851	+1.167	−0.046

Table A7: Weight interpolation's arithmetic on six float32 configurations at $\lambda = .044$, medians of three seeds. *Round 0, without* is the paraphrase the update installs on the run without weight interpolation; *arithmetic* is the pre-trained floor plus $(1 - \lambda)^{31}$ of the installed gain above it, the measurement weight interpolation would leave if the fact fell with the weights and the rehearsal put none of it back; *departure* is the round-30 measurement minus it, in a unit where one band, the two-standard-error width of one 150-fact rate at the median of three seeds, is .059. *Fact kept* is the share of the installed gain left at round 30 and *weights kept* the run with weight interpolation's displacement from the pre-trained weights as a share of the run without weight interpolation's, which lies within .008 of .27 on every seed of every configuration; every configuration completes its training sentence at ceiling under weight interpolation. The last two rows are the rank-64 update at two further strengths, each measured against its own round-0 wording over the thirty interpolations that follow it; the training sentence is at ceiling at $\lambda = .033$ and on .987 of the facts at $\lambda = .072$. The *margin* columns measure the same paraphrase as the gold answer's log-probability minus its strongest rival's, mean over the 150 facts, in nats: the update's gain above the pre-trained model, the run with weight interpolation at round 30, and that measurement's departure from $(1 - \lambda)^{31}$ of the gain, medians of three seeds; one band is 1.63 nats. *Main conclusion: weight interpolation leaves .27 of the weights in every configuration; on the hit rate the main run's fact falls exactly as the weights do, while Mistral-7B and the rank-64 update keep more of it and the Wikidata facts less, and the rank-64 update keeps more than the arithmetic at two further strengths; on the margin the main run ends 4.6 nats below the pre-trained model, and the six departures run from -5.2 to $+0.9$ nats.*

to round 30 an arriving fact loses a further .027. Writing the five facts for 900 steps instead of 30, thirty facts a round instead of five at equal length, or each fact 180 times instead of 30 at equal sightings moves it by .013, .020 and .053, each on three seeds at a resolution of .084: dose, breadth and sightings per fact leave the loss on the paraphrase where the 30-step update of five facts puts it.

Two controls on the arriving schedule are the arrival measurements above and §3. Where the per-round rehearsal returns the erased sentence of a fact never rehearsed, the paraphrase it returns beside it measures .047, within one resolution of the pre-trained floor.

D.5 Arrival in five batches under weight interpolation, and under three other stabilisers

The five-batch arrival of Table A6 under weight interpolation at $\lambda = .15$ answers the paraphrase on .080 against .213 without it, .107 lower at the median seed, and raises accuracy over the whole of

MMLU by .0095, with a paired half-width of .0066, and both runs complete every training sentence. Under weight interpolation at six strengths (listed with the repository) the paraphrase falls from .167 at the weakest to .067 at the strongest against .187 without weight interpolation, with every training sentence complete up to $\lambda = .15$ and .407 of them at $\lambda = .30$. The one-batch ladder of weight interpolation over λ, rescaled to this schedule's measurement without weight interpolation, was registered before the experiment as each strength's prediction at its distance: all six land within one resolution of it, and the four stabilisers of the repository's table land as registered, three on it and the L^2-SP penalty above.

Under the four stabilisers of §5 on the same schedule, every arm in one experiment (the table ships with the repository), weight interpolation's gain on MMLU measures .0104 at a resolution of .0045 and its paraphrase .147 below the run without weight interpolation's; the Fisher-weighted penalty raises MMLU on every seed, by .0146 to .0219, with the paraphrase at the run without weight interpolation's; the L^2-SP penalty lowers MMLU on every seed and the paraphrase by .113; the previous-round anchor ends at .861 of the distance of the run without weight interpolation with both measurements where that run has them.

The same schedule for 150 rounds. With five facts arriving a round for 150 rounds, 755 in all, three seeds (the table ships with the repository), the run without weight interpolation completes every training sentence at round 150 and loses the panel: its accuracy over all of MMLU is $-.028$ from its own round 0, at a resolution of .0096, and its fixed-text perplexity rises 26.1 points, at 7.6. Weight interpolation toward the pre-trained weights scores .0179 above it on MMLU at round 150, at .0108, and its perplexity rises 1.9 points; its training sentence measures .445. In a second experiment on the schedule, four seeds, the Fisher-weighted penalty scores .0332 above its own run without weight interpolation, at .0139, with every training sentence complete, and its perplexity rises 1.25 points against that run's 19.5; the L^2-SP penalty scores .0016, within its resolution of .0094.

The selection rule of §5. The rule keeps the strongest setting at which it and every weaker setting hold at least .95 of the run without weight interpolation's measurement and of its MMLU. On the main run's float32 ladder (Table A17) it keeps $\lambda = .044$ on every seed measured on the training sentence and $\lambda = 0$ or .005 measured on the paraphrase, five settings apart at the median seed, the paraphrase .340 lower at the training sentence's setting; MMLU never falls below .998 of the run without weight interpolation's, so a rule that measures the panel alone keeps the top setting and the panel binds neither. On fifteen facts arriving a round for ten rounds on bfloat16 weights, it keeps $\lambda = .15$ on the training sentence and no interpolation on the paraphrase, four settings apart on every seed, the paraphrase .053 lower at the training sentence's setting.

D.6 Public-source facts, rule-written sentences

The recipe and the two runs are those of Table A3 on three new seeds. The material is 150 facts taken from Wikidata (Vrandečić & Krötzsch, 2014), each the year a building was constructed, the item and the revision frozen with the material. The training sentence and the untrained-wording probe are rendered from the item's fields by fixed frames, and a validator re-renders every row and compares it byte for byte. The probe's content-word overlap with its training sentence is matched in distribution to the invented facts'.

On the paraphrase the run with weight interpolation measures .129, .120 and .148 of the run without weight interpolation's rate on the same seed (median .129, resolution .040), and on the training sentence 1.000, .993 and .993 (resolution .007).

D.7 Which facts weight interpolation removes: by what the pre-trained model already knew

Table A8 prints every seed of the three tiers; this appendix gives the material and what a lost fact answers instead. The material is 150 Wikidata facts of the shape of §D.6, the founding years of settlements, institutions and clubs, chosen from a pool of 1,350 candidates by the pre-trained model's own measurement on the two probes before any update: 37 facts it answered in neither wording, 37 it answered in the training sentence and not in the paraphrase, and 37 it answered in both, the three tiers of Gekhman et al. (2024) measured by the probes of §2, with the same mix of classes in each tier; 39 further facts it answered in neither wording fill the material to 150 and enter no measurement

the pre-trained model's prior on the fact	seed 1	seed 2	seed 3	median	without	with
answered in neither wording	18/21	14/18	13/16	.812	.486	.108
answered in the training sentence alone	11/22	10/24	15/24	.500	.649	.378
answered in both wordings	3/35	2/35	3/35	.086	.946	.892

Table A8: What weight interpolation removes, by what the pre-trained model already knew: 150 Wikidata facts on Qwen3-8B in three tiers by what the pre-trained model answers before any update, 37 facts a tier, under the run of Table 1 on the float32 master beside its run without weight interpolation, three seeds. A seed's entry is the number of the tier's facts the run without weight interpolation answers in the paraphrase at round 30 that the run with weight interpolation does not, over the number it answers; *median* is the median of those shares. The last two columns are the tier's hit rate on the paraphrase at round 30 over all 37 facts, on the run without weight interpolation and on the run with it, medians of the seeds. On the training sentence the run with weight interpolation loses none of the facts the model did not know. The two ends differ by $-.727$ at a resolution of .378, in the same direction on every seed. *Main conclusion: weight interpolation removes the paraphrase of .812 of the facts the pre-trained model did not know and of .086 of those it knew.*

below. The run is the run of Table 1 on the float32 master at $\lambda = .044$ beside its run without weight interpolation, three seeds, on Qwen3-8B.

What a lost fact answers instead. On the paraphrase of a lost fact the run with weight interpolation continues, over the three seeds, with the year of another trained fact on 21 of the 45 lost facts the model did not know, with the year the pre-trained model gives on 5 and with a year outside the trained set or none on 19; on the middle tier 15, 6 and 15 of 36; on the facts it knew 4, 0 and 4 of 8. The pre-trained model's log-probability of the gold year, measured before the update, separates the facts weight interpolation removes from the facts it leaves at an AUC of .108 over all 150 facts (.124, .096 and .099 by seed): the higher the prior, the less often the fact is lost.

D.8 Benchmark edits under a LoRA update and a MEMIT update

The 150 edits are the first 75 CounterFact and the first 75 zsRE requests, in a hashed order, on which the pre-trained model answers the true target on the training sentence and has yet to answer the new target on the paraphrase; every subject occurs once. The paraphrase is the benchmark's paraphrase prompt, one per edit. MEMIT is run once on the down-projections of layers 5 to 9 in float32, with second moments from 3,000 Wikipedia articles. From round 1 the two updates are maintained identically, by LoRA rehearsal of the 150 trained sentences, and the three MEMIT runs share one edited checkpoint. Seed by seed (the table ships with the repository): under the LoRA update the paraphrase falls by .300 at the median seed with the training sentence completed on every edit, and under the MEMIT update it falls by .593 with the training sentence at .873 of the run without weight interpolation's, after measuring .320, .327 and .347 at round 14, before the rehearsal returns it. The MEMIT run with weight interpolation ends 8.8 from the pre-trained weights where its run without weight interpolation ends 113.8.

The same edits on the float32 master at $\lambda = .044$. The strength of §3, carried to the 150 edits under the LoRA update on three seeds, is the float32 benchmark row of Table A6: every edit prompt is completed on every seed, and the paraphrase prompt measures .320 below the run without weight interpolation at the median seed (.327, .320 and .313), against a resolution of .059. On the 75 CounterFact edits, whose old answer the pre-trained model holds, weight interpolation returns the paraphrase to the old answer: the share of edits on which the new answer outranks the old on the paraphrase prompt measures .600 without weight interpolation and .093 with it, a paired difference of $-.493$ against a line of $-.253$, and the greedy continuation of the paraphrase gives the old answer on .293 of the edits with weight interpolation against .107 without it. On the 75 zsRE edits, whose old answer the pre-trained model does not hold, the same share moves by $-.280$. The run with weight interpolation ends at .281 of the distance of the run without weight interpolation, and its MMLU differs from the run without weight interpolation's by .004.

The same pair on 600 edits. On 600 public edits, half CounterFact and half zsRE, under the LoRA update on the float32 master at $\lambda = .044$ on three seeds (Table A6), every edit prompt is completed on every seed and the paraphrase prompt measures .323 below the run without weight interpolation

at the median seed (.335, .323 and .320), against a resolution of .059. On the CounterFact half the share of edits on which the new answer outranks the old on the paraphrase prompt moves by $-.520$ with weight interpolation ($-.520$, $-.397$ and $-.533$; resolution .109), and on the zsRE half by $-.253$; the run with weight interpolation ends at .265 of the distance of the run without it, and its MMLU differs from that run's by .006.

The neighbourhood prompts. Editing benchmarks measure an update a third way, on a prompt about a neighbouring subject the edit does not concern, and Table A9 gives that measurement on both updates at round 30. Under the LoRA update weight interpolation raises the neighbourhood score by .133 over the 450 edit-paired measurements, carried by the CounterFact edits; under the MEMIT update both runs sit at or beside 1.000. What weight interpolation takes is the edited fact's own paraphrase, and the neighbourhood is no worse for it.

	without		with			
edits	round 0	round 30	round 0	round 30	difference	95% interval
LoRA update, all 150 edits	.807	.800	.820	.940	$+0.133$	$[+0.087, +0.182]$
75 CounterFact	—	.613	—	.880	$+0.258$	$[+0.173, +0.347]$
75 zsRE	—	.987	—	1.000	$+0.009$	$[+0.000, +0.022]$
MEMIT update, all 150 edits	.993	1.000	.993	.993	-0.004	$[-0.013, +0.000]$
75 CounterFact	—	1.000	—	.987	-0.009	$[-0.027, +0.000]$
75 zsRE	—	1.000	—	1.000	$+0.000$	$[+0.000, +0.000]$

Table A9: The neighbourhood score of the 150 benchmark edits, the editing literature's third measurement of an update, under the two updates of §D.8 at round 30 on three seeds. The score is the share of edits whose neighbourhood prompt still prefers the pre-trained model's own continuation over the written answer, so a higher score is less of the new answer reaching prompts the edit does not concern; the levels are medians of the seeds and *round 0* is the update itself, before any round of the run. *Difference* is the edit-paired difference between the run with weight interpolation and the run without it, the three seeds pooled into 450 pairs, with a bootstrap interval over edits. *Main conclusion: weight interpolation costs the neighbourhood score nothing under either update — under the LoRA update it raises it by .133, carried by the CounterFact edits, and under the MEMIT update both runs sit at or beside 1.000.*

D.9 A SECOND FACT SCHEMA

On a second fact schema, 150 invented person-to-birthplace facts whose answers are city names and whose probes share no structure with the training sentence, the run with weight interpolation completes every training sentence on all three seeds. It answers the paraphrase on .293, .307 and .280 of the facts against the run without weight interpolation's .733, .673 and .607.

E COULD SOMETHING ELSE PRODUCE THESE MEASUREMENTS?

This appendix answers a question of §4: beyond the two-by-two, could a simpler account produce a training sentence at ceiling beside a lost paraphrase?

Interference: something else was written over it. This is the default account of forgetting (McCloskey & Cohen, 1989; French, 1999), and it predicts that removing the new material removes the loss under weight interpolation. On runs that rehearse the 150 sentences alone the paraphrase ends at .127, .113 and .140 under weight interpolation against .480, .493 and .480 without it. Against runs of the same recipe that add six distractor rows a round, the differences are $+0.007$, -0.040 and $+0.033$ under weight interpolation and $+0.013$, $+0.013$ and -0.067 without it, all inside the band within which a difference counts as none (§2). The loss is the same size with and without new material. This control covers the one-batch runs of Table 1; on its two arrival schedules new facts enter every round by design, and there the later updates lower the rehearsed static set's paraphrase with no stabiliser at all (Table 1, third row); §D.4 measures the arriving facts' paraphrase by the round each arrived.

The model degraded generally. This account predicts that the benchmark panel falls with the paraphrase. The panel measures the run with weight interpolation as it measures the run without it, its perplexity slightly lower (§C.3), and both complete every training sentence: the paraphrase falls by .353 at the median seed (Table A3) on the run whose perplexity and training sentence measured best.

One interpolation would have done it. This account predicts that the first round's interpolation carries the loss. Runs forked from one checkpoint taken before any interpolation lose 153 of 450 item-measurements over 30 rounds at $\lambda = .15$; the 29 further interpolations account for 136 of them and the first round's single interpolation for 17.

Weight interpolation is importance-weighted and removes the directions that carry the untrained wording. It predicts that weight interpolation alone takes the wording. On the float32 master the update with its interpolation and no rehearsal ends $+.95$ nats above the pre-trained model and the run that adds the rehearsal -2.87 nats from that arm; on bfloat16 weights the interpolation step raises the margin by $+10.601$ nats over 30 rounds and the rehearsal step lowers it (§F).

Post-training in general loses the paraphrase. This account predicts that any post-training loses a fact's paraphrase before its sentence. Three public base-to-instruct ladders, measured with no training (Qwen3-8B-Base to Qwen3-8B, OLMo-2-7B to its Instruct model, Llama-3.1-8B to Tulu-3-8B), lose the sentence on .162, .126 and .061 of the real facts the base model answers in both wordings and the paraphrase on .197, .138 and .063; the paired differences, .035, .012 and .002, have 95% intervals that end at .065 at most.

Weight interpolation hides the answer in the readout. This account predicts that retraining the output layer alone returns the paraphrase. On the facts the run without weight interpolation answers in that wording and the run with weight interpolation does not (42, 51 and 48 on three seeds), 90 steps on their training sentences that update only the output head and the final norm return it on .143, .098 and .167 from the run with weight interpolation at round 30 and on .571, .490 and .542 from the pre-trained model, which was never updated with these facts; on facts outside the training rows the latter gains .000, $-.014$ and $-.053$. The output-layer retraining measures what the readout alone returns from two starts far apart in loss on the training sentences: the pre-trained model, which never held the facts, gives the wording back more often than the interpolated weights do.

Weight interpolation left the weights where they were, or the probe has no answer to give. The distance separates the two runs on every seed: 18.71, 18.53 and 18.82 with weight interpolation against 37.71, 37.46 and 37.30 without it. The pre-trained model's floor on both probes is printed with Table A1, and Table A3 prints the run without weight interpolation's paraphrase on every seed.

The fact went out with the wording. This account predicts that under weight interpolation the gold year sits no higher among the material's years than the pre-trained model puts it. Measured as a closed-set rank on the paraphrase at round 30, the run with weight interpolation puts it at 13 where the pre-trained model puts it at 50 and the run without weight interpolation at 1, and the per-fact difference between the two runs is above zero on every seed (Table A10). The order the update left among the answers is in the weights under the interpolation; what the greedy decoding of that wording no longer finds is the way to it.

The wording overlap, or this family and material. On probes of falling overlap with the training sentence both runs fall and the difference between them stays (§C.2); the retention–paraphrase gap recurs on two further families and three further materials (§3).

F Which Step of a Round Lowers the Paraphrase?

§4 measures what weight interpolation removes as the difference between two runs at round 30, with weight interpolation and the rehearsal each off and on (Table 2, top; the bfloat16 two-by-two ships with the repository, its Mistral-7B half in §G). This appendix measures the move each step makes within a round, on the weights the run holds that round.

A *margin* is the gold span's sequence log-probability minus the strongest competitor's, in nats, measured on the same probes as the hit rates. On the float32 master at $\lambda = .044$ (Table A11, second block) both steps lower the paraphrase's margin, the weight interpolation step by -2.5 nats summed over the 31 rounds and the rehearsal step by -4.0 over the 30, at the median seed. The arm with

model	seed 1	seed 2	seed 3	median	ranked first
run without weight interpolation	1.5	1.0	1.0	1.0	.553
run with weight interpolation	14.5	13.0	13.0	13.0	.173
pre-trained model	—	—	—	50.0	.060
with weight interpolation − without, per fact	6.0	6.0	4.5	6.0	—

Table A10: Where the gold year stands among the 150 trained years of the material, measured on the paraphrase at round 30, on the round-30 checkpoints of the two runs of Table A3 and on the pre-trained model. A rank of 1 is first; *ranked first* is the share of the 150 facts the scorer puts there, the median of the seeds. The pre-trained model holds no seed: it is one model, and the three hosts measured it alike. The last row is the per-fact difference between the two runs, the median over the facts of each seed. *Main conclusion: the run without weight interpolation puts the gold year at rank 1.0, the run with weight interpolation at 13.0 and the pre-trained model at 50.0, so under weight interpolation the update's ordering of the answers is still in the weights while the greedy decoding of the same wording has stopped reaching it.*

the run's update and its per-round weight interpolation and without the rehearsal ends with the gold answer's log-probability in the paraphrase $+.95$ nats from the pre-trained model's, above it; the run, which adds the per-round rehearsal of the training sentence and nothing else, ends at -1.92, below it, and -2.87 from the arm. Weight interpolation alone leaves the fact above the pre-trained model; the rehearsal step is the difference between the two arms. On bfloat16 weights at $\lambda = .15$ (first block) the two steps change sign on the paraphrase from round 13 on, where the run has fallen below the pre-trained model: the weight interpolation step raises the paraphrase's margin and the rehearsal step lowers it.

What the L^2-SP penalty keeps at one distance is the gold answer's own term. The two runs of Table 3 at one distance, the L^2-SP penalty at $\beta = .0014$ and the per-round weight interpolation at $\lambda = .044$, are rebuilt at round 30 from their stored adapters on the card they ran on, and the paraphrase's margin is read in its two terms. The gold answer's log-probability is $+5.781$ nats higher under the penalty than under weight interpolation, the median over the 150 facts and then over three seeds, at a resolution of 1.081, and higher by more than the resolution on each seed; the strongest competitor's term is -1.018 nats from weight interpolation's, lower under the penalty; the margin is $+6.442$ nats wider. Each of the three is a median of per-fact differences and is read on its own. At one distance the penalty answers the paraphrase more often (Table 3) because the answer's own log-probability is kept, not because a competitor is pushed down.

Unrelated facts on the same round-30 weights. The main run's round-30 weights, rebuilt from the stored adapters, are read on 300 CounterFact facts the run never saw, chosen from a pool of 3,970 by the pre-trained model's per-token log-probability of their answers so that they match the maintained facts' paraphrases fact for fact in that confidence, and on those paraphrases. Each quantity is the gold answer's mean log-probability per token, the median over facts and then over three seeds, at round 30 against the pre-trained model on the same card. The maintained facts move by -0.480 nats a token and the unrelated facts by -2.008, a difference of 1.431 at a resolution of .215. The fall below the pre-trained model is a drift across facts, in their gold answers rather than in what perplexity or a benchmark panel measures: facts of the same confidence that the run never updated fall further than the facts it rehearses, and the rehearsed facts hold best against it.

The two steps on bfloat16 weights. On the training sentence the weight interpolation step cuts the margin and the rehearsal step returns most of the cut; on the paraphrases the weight interpolation step's rise is the gold answer's own term, and the strongest competitor's term moves little under either step. By round 13 the gold answer's mean log-probability on the paraphrase is below the pre-trained model's -6.4 nats on all three seeds; on the run without weight interpolation the same rehearsal raises it from -4.1 at round 0 to -2.9 at round 30 at the median seed, where the run with weight interpolation goes from -3.6 to -9.4.

Where the lost wordings go. The replay also measures a 12-token greedy continuation of each probe, and the pre-trained model's own. Where the pre-trained model holds a year of its own, the lost fact returns to it: on counterfactual birth years of known people the run with weight interpolation ends by answering the pre-trained year on .340 of the facts, the run without weight interpolation on

	training sentence			paraphrases		
	seed 1	seed 2	seed 3	seed 1	seed 2	seed 3
interp. step, 30 rounds summed (nats)	−60.853	−52.290	−53.680	+10.601	+8.667	+6.786
rehearsal step, share of the interp. step's	+0.964	+0.911	+0.935	−1.640	−1.805	−2.019
interp. step, gold answer's log-probability	—	—	—	+12.488	+8.237	+12.774
interp. step, strongest competitor's	—	—	—	+1.887	−0.430	+5.988
rehearsal step, gold answer's log-probability	—	—	—	−18.202	−13.745	−18.979
rehearsal step, strongest competitor's	—	—	—	−0.813	+1.898	−5.281
loss with minus without, 30 rounds (nats)	—	—	—	+6.633	+7.056	+4.862
The float32 master at $\lambda = .044$: the two steps, and round 30 against the pre-trained model						
interp. step, 31 rounds summed (nats)	−32.154	−31.430	−32.051	−2.510	−2.224	−3.057
rehearsal step, 30 rounds summed (nats)	+31.635	+31.572	+30.574	−3.977	−4.002	−3.239
round 30 minus the pre-trained model, run	—	—	—	−1.922	−2.589	−1.405
same, update and interpolation, no rehearsal	—	—	—	+0.952	+0.848	+0.963
run minus that arm	—	—	—	−2.874	−3.437	−2.367
Interpolation alone, no rehearsal, at the distance of the run with weight interpolation						
distance ratio to the float32 run with it	1.004	1.030	.988	—	—	—
hit rate, share of the untouched update's	.040	.040	.040	—	—	—
trained minus untrained hit rate	—	—	—	−0.053	−0.022	−0.058

Table A11: A round's two steps measured apart on the run with weight interpolation, three seeds. *Margin* is the gold answer's log-probability minus the strongest competitor's, in nats, averaged over the items. First block, bfloat16 weights at $\lambda = .15$: the weight interpolation step's summed move in the margin over the 30 rounds; the rehearsal step's as a share of it, positive where it undoes part of weight interpolation; the two terms of the paraphrases' margin under each step; and the run's summed loss of paraphrase margin beyond the run without weight interpolation's. Second block, the main run on the float32 master at $\lambda = .044$, rebuilt round by round from its stored adapters: the two steps' summed moves in the margin, and the gold answer's round-30 log-probability in the paraphrase, the median over the 150 facts, against the pre-trained model's on the same card. Last block, weight interpolation alone at the distance of the run with weight interpolation, hit rates counted above the pre-trained floor. *Main conclusion: on the float32 master both steps lower the paraphrase's margin, and the update with its interpolation and no rehearsal ends above the pre-trained model where the run that adds the per-round rehearsal ends below it, on every seed; on bfloat16 weights the weight interpolation step raises the paraphrase's margin from round 13 on and the rehearsal step lowers it.*

.020. On the invented facts, counted so that each lost fact falls in one class only, .448 of the 163 facts lost under weight interpolation answer with another trained fact's year and .524 of the 21 lost without it, the first inside the second's interval (Table A12): the two runs differ in how many facts they lose, not in what a lost fact answers.

loop	lost	pre-trained year	another trained year	outside or none	share	95% interval
with	163	11	73	79	.448	[.374, .525]
without	21	0	11	10	.524	[.324, .717]

Table A12: What a lost fact answers instead, on both runs of Table A3, three seeds pooled. A fact is lost when the paraphrase is answered at round 0 and not at round 30; its round-30 continuation falls in exactly one of three classes; the last column is the share answering with another trained fact's year, with a Wilson interval. The run with weight interpolation loses 163 facts over the three seeds and the run without it 21. *Main conclusion: .448 of the lost facts answer with another trained fact's year under weight interpolation and .524 without it, the first inside the second's interval, so the two runs differ in how many facts they lose and not in what a lost fact answers.*

Weight interpolation alone, and the rehearsal alone. Run without the rehearsal, weight interpolation alone, brought to the distance of the float32 run with weight interpolation, which ends at .074 of the untouched update's distance, leaves the training sentence's hit rate at .040 of the untouched update's and the gap between the training sentence and the paraphrase at -0.053, both at the floor:

rehearsal	dose	paraphrase	training sentence	distance
the training sentence every round	90	.147	1.000	—
paraphrases on 7 of the 30 rounds	90	.167	.980	—
paraphrases on 15 of the 30 rounds	90	.153	.913	—
paraphrases on all 30 rounds	90	.200	.487	—
five wordings a round, per-row dose matched	matched	.207	.933	1.018
training sentence; the update's subspace spared	90	.133	.467	.526
training sentence, a round-0 update each round	900	.180	.980	—
five wordings, a round-0 update each round	900	.387	.893	—
four rows a round from one wording	90	.140	.993	—
four rows a round from two wordings	90	.193	.887	—
four rows a round from four wordings	90	.233	.693	—

Table A13: The per-round rehearsal changed in content or in amount, on the setting of Table A3 at $\lambda = .15$, measured at round 30. Each cell is the median over the run's seeds, and each row is a rebuild inside its own experiment (the first row is a rebuild of the run of Table A3, which is at .127 there, inside the .074 band two builds of one recipe span); *dose* is the optimiser steps a round's rehearsal takes (the run with weight interpolation of Table A3 takes 90); *distance* is the ratio of the run's distance from the pre-trained weights to the run with weight interpolation's, printed where the experiment measured it. The first four rows replace a share of the 30 per-round rehearsals of the training sentence with paraphrases of the same facts. The five-wording run rehearses the training sentence and four further wordings of the same facts, each row touched as often as in the verbatim run; the spared-subspace run keeps the verbatim rehearsal and exempts the subspace of the round-0 update from weight interpolation on three rounds, and ends at about half the distance of the run with weight interpolation. The dose ladder's top setting rehearses a whole round-0 update each round; the coverage runs rehearse four sentences a round drawn from one, two or four wordings.

on an update that was never rehearsed weight interpolation lowers the two wordings together. In the other order, one interpolation after round 30, brought to the distance of the run with weight interpolation, lowers both probes on a run whose update was rehearsed for thirty rounds, the training sentence from 1.000 to .560 and the paraphrase from .513 to .220; one round of the standard rehearsal with no interpolation returns the training sentence to .833 and the paraphrase to .287, and three rounds return the sentence to .993 and leave the paraphrase at .300 (medians of five seeds). One round of rehearsal returns the sentence it rehearses and about a quarter as much of the paraphrase.

G Does Changing the Rehearsal Keep the Paraphrase?

This appendix backs the first three questions of §4: the two-by-two on a second base model, and whether a richer or a larger rehearsal puts the paraphrase back under the same interpolation.

A second base model. The four corners of Table 2 measured the same way on Mistral-7B, on bfloat16 weights, medians of three seeds: the update left alone completes the training sentence on .993 and answers the paraphrase on .753; weight interpolation alone, at the rehearsed run's distance, leaves .087 and .053; the rehearsal alone keeps 1.000 and .700, and .733 at a hundred times the per-round dose; the two together hold .987 and leave .240.

What the rehearsal is made of. Table A13 changes the rehearsal's content and amount at $\lambda = .15$. Paraphrases improve extraction at the update step (Allen-Zhu & Li, 2024a; O'Neill, 2026), stand in for replay's earlier examples (Rolnick et al., 2019; Chaudhry et al., 2019b), and scale poorly when repeated (Yang et al., 2025). What a richer rehearsal buys on the paraphrase it pays on the training sentence: every rise past the resolution comes with a lower training sentence, and the one run that passes .303, the midpoint of the two runs of Table A3, has the training sentence at .893. The run that rehearses four paraphrases of each fact on all 30 rounds is also measured by a second scorer, which scores a wording as answered when the gold year outranks every other year of the set: the wordings are answered on .380 of the facts and the training sentence on .569, so the first-written sentence is installed more firmly than a rehearsed wording.

The dose ladder. The per-round dose alone is moved up three settings, a tenth, a third and the whole of the round-0 update, under two rehearsal contents: the training sentence alone, and the training sentence with four further wordings. Under weight interpolation the top setting's two runs, averaged within seed, measured .280 at a resolution of .029, beside the .303 midpoint, and rehearsing the five wordings in place of the training sentence alone adds .156 of the paraphrase averaged over the three settings, with the training sentence as low as .767. With weight interpolation off both top settings are above the midpoint, .527 and .840 at a resolution of .033, and all eighteen runs complete every training sentence.

How many wordings the rehearsal is spread over. With the amount of rehearsal held at the run with weight interpolation's, four sentences a round drawn from one, two or four wordings of each fact, the probe's wording held out of all of them, the four-wording run measures .233 at a resolution of .075, with the one-wording run beside it at .140; the difference between four wordings and one, taken within each seed, is .120 at the median of the five seeds. What coverage buys on the paraphrase it pays on the training sentence: on the weakest seed the training sentence is completed on .993, .873 and .667 of the items at the three settings.

Four wordings at the update. Round 0 writes every fact in four wordings and every round rehearses all four, at the step counts of Table A3, so each sentence receives a quarter of the dose; the probe is a fifth wording, held out of the update and of the rehearsal (Table 2, bottom).

The rehearsal's sign on the paraphrase margin. On the float32 master the gold answer's log-probability on the paraphrase, median over the 150 facts, is read at round 30 relative to the pre-trained model on the same device: the main run ends at -1.9 nats (three seeds -1.9, -2.6, -1.4, resolution .86), and the run with the same round-0 update and the same per-round weight interpolation but no rehearsal ends at $+.95$ nats. The difference between the two runs is -2.9 nats. Replaying the thirty-one rounds step by step from the stored per-round adapters, the rehearsal step lowers the margin by -4.0 nats in total and the weight interpolation step by -2.5 nats; on the float32 master neither step raises it. The run is replayed from the stored adapters and its two greedy completions at round 30 match the original run's on every fact.

H Where Does the Paraphrase Return From? The Block Exchange and the Update's Eight Dimensions

This appendix backs the last paragraph of §4, that the fact is still in the weights: the exchange block by block, the exchange inside the eight dimensions per block that the round-0 update occupies, on both number formats, and what weight interpolation removes over the thirty rounds.

Paraphrase returned per unit of displacement exchanged. The MLP exchange moves .804 of the squared displacement that separates the two runs and returns .353 of the paraphrase hit rate; the attention exchange moves .196 and returns .080, so the hit rate returned per unit share is .44 and .41. Within the MLP the gate, up and down projections move .413, .298 and .094 and return $+.207$, $+.180$ and $+.053$. The reordered controls move the same norm and return none: what returns the hit rate is the content of the exchanged blocks, and the blocks return it in proportion to the share of the update they hold. Moving each fact's own residual from the run without weight interpolation into the run with weight interpolation at one layer returns .173 of the lost hit rate at the median seed, and one direction shared by all the facts returns $-.004$.

H.1 What Weight Interpolation Removes Returns From the Update's Dimensions

What weight interpolation removes is still in the weights of the run without weight interpolation, in the eight dimensions per block its round-0 update occupies. Put back by itself, the part of the difference between the two runs inside that frame returns the paraphrase to .500 on bfloat16 weights and .473 on the float32 master (Table A14). A displacement of the same rank and norm in a random frame controls the size of the step, not its content, and returns none of it; subtracted from the run without weight interpolation it leaves most of the wording, where the update's part takes it to the floor. Per unit of squared displacement the update's frame returns .624 of the wording, the whole MLP blocks .398 and the random frame .009.

	bfloat16 weights		float32 master	
round-30 state	untrained	trained	untrained	trained
the run with weight interpolation	.120	1.000	.127	1.000
+ a random frame of the same rank and norm	.120	.473	.100	.480
+ the same blocks, rows reordered	.140	—	.100	.360
+ the part of the difference in the update's eight dimensions	.500	.980	.473	.967
+ the MLP blocks of the run without weight interpolation	.473	1.000	.480	.993
the run without weight interpolation	.480	1.000	.473	1.000
− a random frame of the same rank and norm	.393	.947	.420	.987
− the part of the difference in the update's eight dimensions	.033	.047	.033	.033

Table A14: What returns the paraphrase to the run with weight interpolation: round-30 hit rates of states assembled from the stored weights of the two runs of Table A3 on bfloat16 weights and of Table 1's first row on the float32 master, medians of three seeds. *The update's eight dimensions* are, per block, those the round-0 update of the run without weight interpolation occupies; *the difference* is the run without weight interpolation's round-30 weights minus the run with weight interpolation's; a random frame has the rank and norm of the part it stands in for. *Main conclusion: the part of the difference inside the update's eight dimensions returns the paraphrase to the run with weight interpolation, .120 to .500 on bfloat16 weights with the training sentence held, and a random frame of the same rank and norm returns none of it; the whole MLP blocks return it to .473 with the training sentence complete on either number format.*

The update's dimensions carry the sentence with the access. Subtracted from the run without weight interpolation, the part takes the training sentence to the floor with the paraphrase on both number formats. Put back by itself, it leaves the training sentence at .980 on bfloat16 weights and .967 on the float32 master.

The frame is the update's own. Per block the projector is built from the round-0 adapter of the run without weight interpolation. The two runs' round-0 adapters occupy nearly orthogonal eight-dimensional frames, the difference of their projectors 3.78 of a full 4.0 per block at the median over the 252 blocks and 3.96 at most, because the adapter's random initialisation precedes the seeding, so each run solves the same 150 facts into a different rank-8 factorisation; the exchange locates the access within one update's factorisation. Every seed of Table A14 ships with the repository.

What weight interpolation removes lies in the round-0 update's own rows. A replay of all thirty rounds from the archived adapters splits every step by how much of it stands in the eight rows of each block that the round-0 update occupies, over all 252 blocks and three seeds. The energy of the run with weight interpolation inside those rows falls from 999 at round 0 to 161 at round 30 on the first seed, and outside them it grows from 25 to 186; the run without weight interpolation's stays where the update put it. The exchange of the MLP blocks that returns the paraphrase from .120 to .473 raises the displacement still inside those rows to .921 of the run without weight interpolation's. Of each round's rehearsal update, .120 lands in those rows under weight interpolation against .016 without it, while the energy outside them grows sevenfold. On a run that leaves 50 of the 150 facts out of the rehearsal, the displacement the thirty rounds wrote, recomputed in exact arithmetic from the stored adapters, has cosine .106 with the round-0 update and lies .089 outside the span of the rehearsal updates; a random subspace of the same rank leaves .978 outside. What the rounds leave in the weights is the rehearsal's update.

I Every Run of Figure A1 Has a Measured Distance

This appendix backs Figure A1 (§5): each stabiliser at its measured distance; the listing of every run on weight interpolation-alone curve ships with the repository. The L^2-SP penalty of Figure A1 is the penalty of Table A15, β times the squared norm of the adapted linear layers' displacement from the pre-trained weights, added here to the loss of the rehearsal rounds alone, with the round-0 update unpenalised; in Table A15 and the five-batch run of §D.5 the same penalty is on every update as well, which is why the coefficients of the three tables differ.

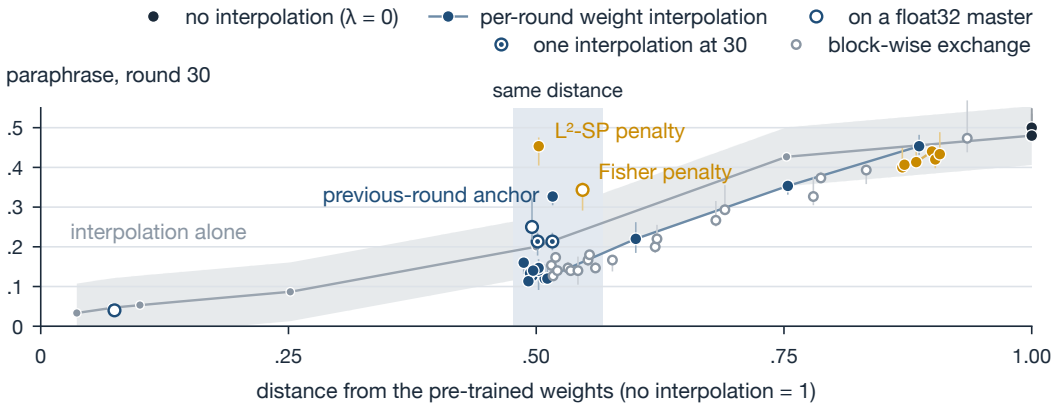

Figure A1: *At the same distance from the pre-trained weights the penalty families keep more of the paraphrase than weight interpolation, with the same retention curve.* Bfloat16 weights in place: the paraphrase at round 30 against the run's distance, as a share of the run without weight interpolation's, along weight interpolation-alone curve with one resolution either side; the two penalties sit at the per-round weight interpolation's own distance.

The runs of Figure A1. Every run on Figure A1 ran the thirty-round run on the 150 invented facts of Qwen3-8B and has a measured distance: 87 run medians over 285 run-seed products. The reference curve is weight interpolation with no rehearsal on a float32 master at six strengths, joined to the pre-trained floor and the run without weight interpolation; the strip either side of it is one resolution, .074 (§B.2). The 23 runs that change the stabiliser alone lie along it to within 1.3 bands, half of them within .53 of a band. The two number formats bracket the curve at one distance: the run on bfloat16 weights lies below it, .120 against .208, and the run on a float32 master above it by the interval of §B.1. The runs that change the rehearsal's dose or content lie below the curve; weight interpolation anchored on the previous round's weights at $\lambda = .50$ and the float32 master at that distance are measured in Table A15 and §J.2. On the dose ladder and its five-wording runs the training-sentence hit rates lie between .767 and .980 while the margins run from 4.08 to 11.18 nats.

stabiliser	anchor	distance	untrained	trained
none (no interpolation)	—	1.000	.487	1.000
per-round interpolation, $\lambda = .05$	pre-trained weights	.759	.333	1.000
per-round interpolation, $\lambda = .15$	pre-trained weights	.510	.133	1.000
Fisher-weighted penalty, $\beta = .001$	pre-trained weights	.588	.360	1.000
Fisher-weighted penalty, $\beta = .003$	pre-trained weights	.706	.220	1.000
L^2-SP penalty, $\beta = .0002$	pre-trained weights	.507	.427	1.000
L^2-SP penalty, $\beta = .0007$	pre-trained weights	.321	.380	1.000
per-round interpolation, $\lambda = .50$	previous round	.517	.327	1.000
per-round interpolation, $\lambda = .15$, anchor's experiment	pre-trained weights	.493	.140	1.000

Table A15: Stabilisers on the run of Table 1 at round 30, medians of three seeds; *distance* is a ratio to the run without weight interpolation's on the same seed. The two rows under the rule are a second experiment, the previous-round anchor beside that run's per-round weight interpolation. The seven runs above the rule are replicated on six fresh seeds in a third experiment: at the per-round weight interpolation's distance the L^2-SP penalty answers the paraphrase .313 more often than weight interpolation and the Fisher-weighted penalty .207 more often, at a resolution of .055, every training sentence complete on every seed. The per-round weight interpolation at $\lambda = .15$ and the two penalties at the smaller β end at nearly one distance, .507 to .588 of the run without weight interpolation's. *Main conclusion: at one distance the retention curve is the same for every stabiliser, and the paraphrase separates them: the per-round weight interpolation answers it on .133 of the facts, a Fisher-weighted penalty on .360 and an L^2-SP penalty on .427.*

Within weight interpolation's family the strength fixes the distance the run ends at; a penalty's distance is measured on the weights. A weight interpolation's strength and the distance it leaves

move together, which is what puts weight interpolation's family on one curve (Figure A1). A penalty's coefficient and its distance are two quantities: raising the Fisher-weighted penalty's β from .001 to .003 moves the run farther from the pre-trained weights on every seed, from .588 to .706 of the run without weight interpolation's distance in the experiment of Table A15 and from .547 to .685 on the six fresh seeds of its replication, and the two settings of the L^2-SP penalty end at .507 and .321 of that distance with the paraphrase .047 apart, at a resolution of .056. The families are therefore compared at one distance (Table A15), each penalty measured at the distance its weights end at.

Two further stabilisers. The run without weight interpolation, interpolated toward the pre-trained weights by .25 at rounds 10 and 20 and once after round 30 to the per-round weight interpolation's distance, keeps .467 of the paraphrase of the run without weight interpolation and .893 of its training sentence, and .513 and 1.000 once the rehearsal resumes. A Fisher-weighted penalty applied from round 0 at the one strength whose round-0 update reaches ceiling ends 1.134 times as far out as the per-round weight interpolation, completes every training sentence and keeps .693 of the paraphrase of the run without weight interpolation.

The curve holds on another update's content. Re-measured on a second material, weight interpolation with no rehearsal gives the same curve: its seven points differ from the first curve's by between $-.054$ and $+.069$. Measured against it by distance, all 40 runs written with that content lie inside the band.

The training sentence at one distance. At the distance of the run with weight interpolation the runs of Figure A1 complete the training sentence on .240 to 1.000 of the facts, more than five times the widest band of the training-sentence probe, .148 (§B.2), and one experiment holds both ends, the 50 facts it left out of the rehearsal and the 100 it rehearsed. In one experiment of six seeds at that distance, the run with the per-round interpolation completes the training sentence on at least .39 more of the facts than either one interpolation or weight interpolation alone.

J How Much a Repeated Interpolation Delivers on bfloat16 Weights, and How the Closed Form Is Checked

This appendix answers §6: how much of its coefficient a repeated interpolation delivers on bfloat16 weights, derived and checked.

Where the threshold applies, measured on the library code. A one-shot merge of bfloat16 checkpoints, a LoRA, linear or TIES merge (Yadav et al., 2023), a model soup (Wortsman et al., 2022a) or a WiSE-FT merge (Wortsman et al., 2022b), applies its interpolation once and incurs a single rounding of order 2^{-8}, so it stays above the threshold; repeated in-place interpolation on bfloat16 weights is below it. The two repeated interpolations of the common training stacks are measured on this paper's own displacements: the round-0 update of the run of Table A3, 35 blocks of 128 output rows over five layers and every LoRA target, applied thirty times into a bfloat16 and into a float32 destination on one A40 under PyTorch 2.14 and TRL 1.14, with the share of the intended movement delivered measured as $\langle x_{30} - x_0,\, g\rangle \,/\, \left(\|g\|^2\,(1-(1-\alpha)^{30})\right)$ for a target displacement g, which exact arithmetic puts at 1. The exponential moving average of PyTorch's `swa_utils` (Paszke et al., 2019, after Izmailov et al., 2018) updates in the parameters' own dtype at a default decay of .999. Into bfloat16 it delivers .016 of the update, median of three updates, and into float32 1.000; Equation 1 with the single rounding of its `lerp` ($c = .5$) predicts .014. The reference-model synchronisation of TRL (von Werra et al., 2020) is bitwise this paper's interpolation on the same inputs, and at its default coefficient of .6 it delivers .963 into bfloat16. A float32 master copy (Micikevicius et al., 2018), a one-shot application, or stochastic rounding (Croci & Giles, 2023) each removes the freeze.

The closed form. Summed over the elements of an update, the norm the update retains after the per-round weight interpolations is

$$R(\lambda) = \Big(\sum_i \min\big(d_i^2,\ (c\,\mathrm{ulp}(w_i)/\lambda)^2\big)\Big/\sum_i d_i^2\Big)^{1/2}, \tag{1}$$

where d_i is the round-0 update on the bfloat16 grid, w_i the pre-trained weight under it, $\mathrm{ulp}(w_i)$ the unit in the last place, the bfloat16 grid spacing at w_i, and $c = .418$ is fitted on one update; Figure A2 (left) draws it against the measured retention of that update and of 16 it never saw.

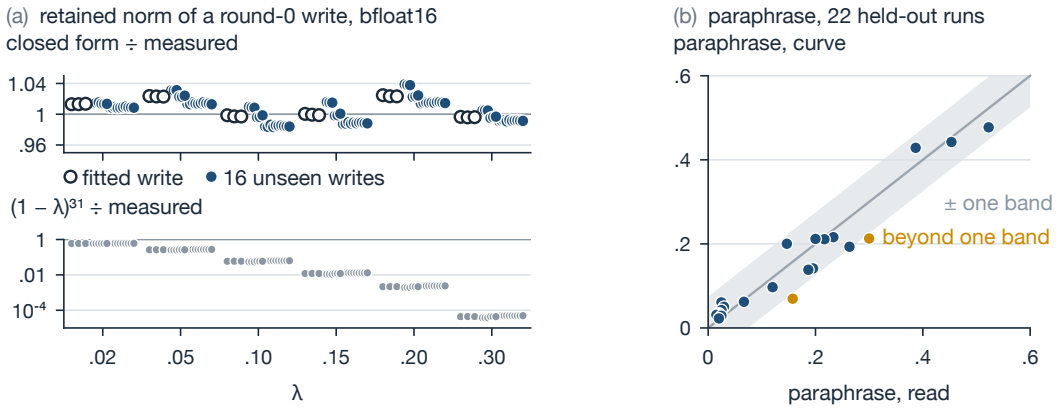

Figure A2: Left: the norm a round-0 update keeps after 31 per-round weight interpolations on bfloat16 weights, predicted over measured: Equation 1 at $c = .418$ on the fitted update (rings) and 16 it never saw (dots), and the arithmetic $(1-\lambda)^{31}$. Right: weight interpolation-alone curve of Figure A1 against 22 runs measured after it was drawn (§5), each at the distance from the pre-trained weights it ends at; shading, one resolution. *Main conclusion: one constant predicts the retained norm of unseen updates within 3.9%, where the arithmetic misses by two orders of magnitude at $\lambda = .15$ and four at .30; the curve places 20 of 22 unseen runs within one band.*

Derivation of Equation 1. A round moves an element of displacement d from its pre-trained value w_0 by λd and rounds the result to the bfloat16 grid at w_0, whose spacing is $\mathrm{ulp}(w_0)$. The element moves when $\lambda|d|$ exceeds a fixed share c of that spacing and stays where it is otherwise, and because the map is deterministic an element that stays one round stays on every later round. An element above the threshold is cut down round by round until it reaches it, so its terminal displacement is $\mathrm{sign}(d)\min(|d|, c\,\mathrm{ulp}(w_0)/\lambda)$; summing its square over the elements of the update and dividing by the update's own squared norm gives $R(\lambda)$. One rounding a round would put c at one half; the round rounds twice, after the product $(1-\lambda)w$ and after the sum, and c is fitted on the round-0 update of the run of Table A3 at .418 and held everywhere else. The retention is computed from the first update alone, with no round replayed. The λ_{eff} beside every bfloat16 measurement is the strength whose arithmetic leaves that norm, $(1-\lambda_{\mathrm{eff}})^{31} = R(\lambda)$: .020 a round at the nominal $\lambda = .15$ on the run of Table A3.

The closed form against the measurement. Figure A2 (left) replays that update with no rehearsal, element by element over 8 million elements of 20 tensors, at six nominal λ; at $\lambda = .15$ the update keeps .544 of its norm where the arithmetic leaves .006, and the norm moves by .0008 after round 15 (.5447, against .5439 at round 31). On that update every element displaced by six to forty units in the last place stops at 3.02 units after the 31 rounds, where the closed form puts the stop at $c/\lambda = 2.79$. A float32 master delivers the arithmetic within 4%, and so does one interpolation to the same distance.

The closed form on updates it was never fitted on. With c held at .418, the closed form is evaluated on the elements of 16 further round-0 updates spanning a 1.5-fold range of size in units in the last place: 13 of two other recipes (three seeds of the benchmark edits and ten of the four-wording update) and three of the moving-anchor run, whose round-0 update is the fitted recipe's on other seeds. Over their 96 measurements the closed form is within 1.3% of the element-by-element replay at the median and within 3.9% at the most, the same two figures over the 13 updates of other recipes alone; on the updates of the fitted recipe they are 0.9% and 2.5% (Table A16).

The replay. Replaying the run with weight interpolation's stored adapters through the run's own merge-and-interpolation step, element by element in bfloat16, reproduces the per-tensor energy of the trained checkpoints, and on the stored bfloat16 weights the round-30 displacement of the run with weight interpolation has cosine .686 with the run's own round-0 update, where exact arithmetic over the same adapters gives .099: the grid keeps the round-0 update in place.

update	size (ulp)	$\lambda = .02$	.05	.10	.15	.20	.30
invented facts (fitted recipe)	2.71	+1.3	+2.3	−0.3	−0.1	+2.3	−0.4
benchmark edits, LoRA	2.13	+1.5	+3.1	+0.9	+1.5	+3.8	+0.5
four wordings at the update	3.13	+0.9	+1.4	−1.5	−1.2	+1.5	−0.8
moving-anchor run, the fitted recipe's update	2.71	+1.3	+2.3	−0.4	−0.1	+2.3	−0.4

Table A16: The closed form of §6 against the element-by-element bfloat16 replay, as the relative error of the retained norm in percent, median over each update's seeds, with $c = .418$ held. *Size* is the update's typical element displacement in units in the last place. The moving-anchor run's round-0 update is the fitted recipe's on other seeds. *Main conclusion: one constant fitted on one update predicts the retained norm of the 13 updates of two other recipes within 3.9% on every measurement.*

A direct count on stored runs. For each of 42 finished runs the round-30 displacement is split into the part the training steps wrote, recomputed in exact arithmetic from the stored per-round adapters, and the remainder the bfloat16 grid left behind, and the share of the stored displacement's norm that learning wrote is measured as a median over the runs of each kind. At the nominal $\lambda = .15$ on bfloat16 it is .013, and with weight interpolation off .857. On a float32 master at the same λ the share is 1.000, and raising the per-round learning rate tenfold on bfloat16 brings the share back to .264. On the run that leaves 50 facts out of the rehearsal at the same λ, the smaller of two shares, the held-out facts' paraphrase kept from round 0 to round 30 and the written displacement's elements that still clear half a bfloat16 step of the pre-trained weights, is .18 at the median seed. The count agrees with Equation 1: under the repeated interpolation the stored displacement is the round-0 update frozen on the grid.

What each anchor delivers, round by round. Equation 1 concerns weight interpolation on the round-0 update alone. Figure 3 measures a second quantity on the run's own trajectory: the share of each round's nominal interpolation that the bfloat16 write-back removes, replayed from the stored per-round adapters over all 252 adapted modules. Anchored to the pre-trained weights weight interpolation delivers .783 of its step on round 1 and .021 on round 30, and .035 on the median round. Anchored to the previous round's weights it is asked for .15 of one round's update and delivers it in full, 1.033 of its step on round 1 and 1.209 on round 30. The replay rebuilds the runs' distances: the previous-round anchor's is .863 of the run without weight interpolation's, against .861 measured on the run.

J.1 The same run for 120 rounds

The run of Table A3 is carried to 120 rounds at λ of 0, .02, .05 and .15, three seeds each, with the recipe otherwise unchanged. From round 30 to round 120 the paraphrase measures the same at every λ, inside the margin of §2, and the training sentence stays at ceiling. At $\lambda = .15$ the run stays where it was at round 30: the paraphrase measures .140 on each seed at round 30 and .113, .120 and .140 at round 120, and the training sentence's lowest measurement over the seven rounds is .993. At the two smaller strengths the displacement is still being cut and the paraphrase measures lower at each later round, inside that margin over the 90 rounds. The run without weight interpolation rehearses for 120 rounds and keeps the paraphrase, .473 at round 30 and .467 at round 120.

J.2 The float32 master

The run of Table 1 run in place on bfloat16 weights at the nominal $\lambda = .15$ measures the same gap as on the float32 master: every training sentence complete, the paraphrase on .127 of the facts against .480 without weight interpolation (Table A3). The loss on the paraphrase follows weight interpolation delivered, on a float32 master as on bfloat16 weights, and the training sentence stays at ceiling over a range of λ there too: on the seven-setting ladder of Table A17, every seed completes every training sentence up to $\lambda = .044$, where the paraphrase measures .120, and at $\lambda = .072$ the training sentence measures .900; a second build of the run at the full nominal $\lambda = .15$ measures .307 and .033. §F measures the two steps on both number formats, and their signs on the paraphrase differ between the two. The two-by-two of §4 and the retention–paraphrase gap itself are measured on the float32 master at $\lambda = .044$, the retention–paraphrase gap on two further model families and on a further material, three seeds each. On Mistral-7B the paraphrase falls .253 below the run without

weight interpolation at the median seed, at a resolution of .059. On OLMo-2-7B it falls .253 at the median seed, at a resolution of .062. On the further invented facts of Table A6 on Qwen3-8B it falls .140, at a resolution of .059. Every training sentence is at ceiling on all three. On Mistral-7B and OLMo-2-7B the training sentence is at ceiling at $\lambda = .072$ as well, .973 and .987, the setting at which it leaves ceiling on Qwen3-8B.

nominal λ	distance	training sentence	paraphrase	ratio to the run without weight interpolation
0 (no interpolation)	1.000	1.000	.473	1.000
.005	.826	1.000	.453	.944
.009	.761	1.000	.387	.817
.022	.514	1.000	.300	.634
.033	.365	1.000	.187	.394
.044	.275	1.000	.120	.254
.072	.141	.900	.067	.141
.150	.072	.307	.033	.069

Table A17: The run of Table A3 on a float32 master at seven strengths of the per-round weight interpolation, beside its own run without weight interpolation, measured at round 30; medians of three seeds. *Distance* is the distance from the pre-trained weights as a ratio to the run without weight interpolation's on the same seed, and the last column the paraphrase as a ratio to the run without weight interpolation's. On the float32 master weight interpolation delivered is the nominal one. *Main conclusion: the training sentence stays at ceiling up to $\lambda = .044$, the main run's strength, where the run ends at .275 of the distance of the run without weight interpolation and answers the paraphrase on .120 of the facts; at $\lambda = .072$ the training sentence leaves ceiling, .900.*

Matched on displacement ($\lambda = .0235$), the float32 run ends at 18.52, 18.87 and 18.86 from the pre-trained weights, the bfloat16 run with weight interpolation at 18.55, 18.89 and 18.68, and completes the training sentence on all 150 facts on those three seeds and on three further seeds, six in all. The float32 run without weight interpolation answers the paraphrase on .453, .527, .533, .487, .520 and .420 of the facts and the matched run on .293, .253, .320, .247, .227 and .220, median .250: a fall of .160, .273, .213, .240, .293 and .200, median .227. The nominal $\lambda = .15$ on a float32 master is the stronger dose (three seeds): it delivers the full arithmetic interpolation and the displacement ends at .074 of the untouched update's. There the run measures .293 on the training sentence and .040 on the paraphrase, and weight interpolation with no rehearsal at the same distance keeps .040 of the training sentence (Table A11): at that distance weight interpolation outweighs the per-round rehearsal, which holds the training sentence at .293.

The same ladder on Wikidata facts. The seven settings are run again on the 150 Wikidata facts of §D.6, the same three seeds on the float32 master (Table A18). From $\lambda = .022$ to .044 the paraphrase lies below the measurement without weight interpolation by more than one resolution at every setting, by .273 at the setting where the difference is least (.320, .273 and .233 on the three seeds, resolution .063), and the training sentence is at ceiling at every setting of the interval, .993 at its lowest; one setting past .044 it measures .833 and at the nominal .15 it measures .193. Below the interval the two weakest settings leave the paraphrase within one band of the measurement without weight interpolation, .020 apart at the nearer, as on the invented facts, and the distance falls with λ at every setting on every seed. The interval, both its ends and the training sentence inside it match the invented facts' (Table A17).

The L^2-SP penalty against weight interpolation on Wikidata facts. The two stabilisers of Table 3 are run again on the 150 Wikidata facts, the L^2-SP penalty at $\beta = .0014$ on the update and on every rehearsal and the per-round weight interpolation at $\lambda = .044$, each with the recipe it has on the invented facts, three seeds on the float32 master. The two are not matched on distance here: the penalty's run ends at .919 of weight interpolation's displacement from the pre-trained weights. The penalty answers the paraphrase on .340 of the facts where weight interpolation answers it on .053, a difference of .280 at a resolution of .084, with the training sentence complete on both, 1.000 and 1.000. The order the invented facts give the two stabilisers holds on facts taken from a public knowledge base, at the same two coefficients.

The release check on bfloat16 weights. Run over the 11 runs of §§3–5 on bfloat16 weights whose paraphrase falls under weight interpolation, each beside its run without weight interpolation, the

nominal λ	distance	training sentence	paraphrase	ratio to the run without weight interpolation
0 (no interpolation)	1.000	1.000	.447	1.000
.005	.840	1.000	.427	.957
.009	.731	1.000	.387	.829
.022	.504	1.000	.187	.414
.033	.361	1.000	.093	.222
.044	.270	.993	.053	.127
.072	.140	.833	.013	.032
.150	.079	.193	.013	.032

Table A18: The ladder of Table A17 on the 150 Wikidata facts of §D.6: the same seven strengths of the per-round weight interpolation on a float32 master, the same three seeds, measured at round 30 beside the run without weight interpolation; columns as in Table A17. *Main conclusion: on facts taken from a public knowledge base the training sentence stays at ceiling up to $\lambda = .044$, where the run ends at .270 of the distance of the run without weight interpolation and answers the paraphrase on .053 of the facts against .447 without weight interpolation; at $\lambda = .072$ the training sentence leaves ceiling, .833: the interval closes at the same setting as on the invented facts.*

protocol of §6 stops all 11, 4 of them on the curve itself and 7 on item 1, and stops 3 of the 10 runs without weight interpolation: two are updates on real documents whose training sentence is below ceiling at round 0, so item 1 signs neither run of the retention–paraphrase gap, and one answers its own paraphrase on .187 of the facts at round 30 against .400 at round 0. The four bfloat16 runs on which a stabiliser is delivered and whose own experiment found no loss on the paraphrase pass the paraphrase item on every build: the Fisher-weighted penalty answers the paraphrase on .347 at round 30 against .340 at round 0, the previous-round anchor on .447 against .447, the fresh anchor on .493 against .460 and the stale anchor on .427 against .460, every training sentence complete. Table 4 counts the float32 block and these four.

K Where the Retention Curve Shows the Loss

This appendix answers §3: the runs whose update leaves the training sentence below ceiling, seed by seed, and the run anchored to the previous round's weights with the distance it ends at.

K.1 On real documents weight interpolation opens the same loss on the paraphrase, from a training sentence both runs hold below ceiling

On 116 facts from news-type documents, with training sentences rephrased from the source reports, Table A19 follows the facts under four updates. Raising the round-0 update over three settings (900, 4,500 and 22,500 steps, a twenty-five-fold span) leaves the training sentence of the run without weight interpolation at .388, .405 and .379; at that top setting the run with weight interpolation completes it on .172, and the paraphrase is answered on .388 without weight interpolation and .121 with it (the row of Table A6).

On documents the update's shape sets where the training sentence lands, and the rehearsal holds the condition it was trained on. Raising the span-supervised update nine-fold, from 900 to 8,100 steps, installs the training sentence on .776 of the facts at round 0 where 900 steps install .733, a gain of .060 at the median seed against a ceiling line of .925, and moves the merged weights no farther from the pre-trained ones, the nine-fold update ending at .98 of the 900-step update's distance. On the same update, seven rounds of rehearsal whose loss is placed on the answer span alone leave the training sentence where the update put it, .034 above round 0 at the median seed, while the same seven rounds rehearsing the whole row take it from .741 to .414 at the median seed.

On the 31 facts that all three runs without weight interpolation of the ladder's first setting complete at round 30, both probes fall under weight interpolation: the run with weight interpolation keeps .538 of the paraphrase of the run without weight interpolation and .800 of its training sentence.

write	loop	training sentence round 0	training sentence round 30	paraphrase round 30
token loss, rank 8	no interpolation	.431	.414	.319
	with	.362	.259	.155
token loss, rank 64	no interpolation	.448	.526	.405
	with	.431	.302	.172
answer-span loss, rank 8	no interpolation	.828	.362	.371
	with	.750	.371	.241
answer-span loss, rank 64	no interpolation	.741	.422	.466
	with	.681	.319	.250

Table A19: 116 facts from news-type documents, written four ways at round 0 and rehearsed verbatim for 30 rounds, without weight interpolation ($\lambda = 0$) and with it ($\lambda = .15$); medians of three seeds. The answer-span loss is taken on the tokens of the answer. *Main conclusion: on these documents the strongest update installs the training sentence on .828 of the facts and the rehearsal holds it at .362 and .371 on the two runs, so the retention curve shows a loss of its own beside the paraphrase weight interpolation lowers; the retention–paraphrase gap of §3 appears where the update installs the sentence at ceiling (Table 1).*

K.2 The previous-round anchor

With weight interpolation aimed at the previous round's weights instead of the pre-trained ones, the run with weight interpolation answers the paraphrase on .440, .467 and .447 of the facts against the run without weight interpolation's .473, .493 and .520 and completes every training sentence; it ends 1.782 times as far from the pre-trained weights as the per-round weight interpolation (§I). Aimed at the weights of seven rounds before, at $\lambda = .15$, the run ends at .845 of the distance of the run without weight interpolation, completes every training sentence and keeps .827 of the paraphrase of the run without weight interpolation (.810, .827 and .971 on the three seeds). Aimed at the previous round's weights at $\lambda = .50$ it ends at the per-round weight interpolation's own distance, within .026 of it, and answers the paraphrase .187 more often than the per-round weight interpolation on the same seed (.173, .187 and .193): at one distance from the pre-trained weights, weight interpolation toward the pre-trained weights keeps less of the fact than the previous-round anchor.

L A Chat Template Treats the Training Sentence as Untrained

The measurement. The main run of Table 1's first row and the bfloat16 run in place are measured the way a deployed model is asked, through the checkpoint's chat template with a free answer, beside the continuation cloze of §2 on the same weights. A probe's prefix, the training sentence cut before its year or the paraphrase, goes into one user turn asking for the continuation of the text, rendered through the tokenizer's chat template with thinking off; the answer is greedy, at most 64 new tokens, and a hit when the gold year is a whole word of it after normalisation. The open-book variant opens the user turn with the training sentence. All four measurements take the 150 facts and the two prefixes of the cloze measurement.

What Table A20 measures. Asked through the template, the training sentence is itself a wording the run never rehearsed: on the float32 master it falls under weight interpolation to .087 against .560 without weight interpolation, while the continuation of the same prefix stays at 1.000 on both runs and, given the training sentence in the turn, both runs answer it above .95. The fact is in the weights, and what weight interpolation removes is the way back to it; the measurement that stays at ceiling is the one the release check uses. The chat measurement is a setting of the wording ladder; it measures the paraphrase on a narrower range than the continuation does, .300 without weight interpolation against .473 by the cloze, and separates the runs on the training sentence. The Wikidata facts and the further invented facts behave as the main run does: under weight interpolation the training sentence falls to .073 against .440 on the Wikidata facts and to .013 against .220 on the further invented facts, the paraphrase falls with it to under half of the run without weight interpolation's, and the open-book measurement stays above .95 on every run. On the further invented facts the run

run	nominal λ	continuation cloze		chat template		chat template, open book	
		trained	untrained	trained	untrained	trained	untrained
main run, float32 master	0	1.000	.473	.560	.300	.960	.853
	.044	1.000	.120	.087	.067	.973	.587
Wikidata facts	0	1.000	.467	.440	.213	.987	.993
	.044	.993	.053	.073	.033	.993	.993
further invented facts	0	1.000	.160	.220	.067	1.000	.967
	.044	1.000	.020	.013	.007	.987	.813
one batch, bfloat16 in place	0	1.000	.500	.440	.240	.960	.733
	.150	.993	.153	.107	.053	.973	.547

Table A20: The main run of Table 1's first row on its float32 master, then the same gap on the Wikidata facts and on the further invented facts (upper block), and the bfloat16 run in place at the ends of its nominal ladder (lower block), measured at round 30 on the same weights by the continuation cloze of §2 and by the checkpoint's chat template with a free answer; medians of three seeds. *Main conclusion: asked through the deployment template, the training sentence is itself a wording the run never rehearsed: under weight interpolation it falls with the others, .087 against .560 without weight interpolation, while its continuation stays at 1.000 on both runs and, given the training sentence in the turn, both runs answer it above .95.*

without weight interpolation answers the paraphrase on .067 of the facts, so that contrast confirms the direction and little more. The bfloat16 run in place measures the same way (lower block).

L.1 What each line scores as retention

Table A22 lists the 23 stabiliser and continual-learning papers of §7 one to a row, then the editing line and the lines that maintain facts by replaying them, each with what it scores as retention and the form it measures.

paper	what it scores as retention	scored on
Stabilisers and continual learning: regularisers toward the pre-trained or initial weights		
Kirkpatrick et al. (2017)	test accuracy per earlier task under its own permutation; human-normalised Atari scores	held-out sample
X. Li et al. (2018)	source-task accuracy on ImageNet and Places365, original classification head	held-out sample
Lee et al. (2020)	source-task validation accuracy, shared head; the GLUE half scores no retention	held-out sample
Chen et al. (2020)	Euclidean distance of the weights to the pre-trained model; no data scored	the weights
Kumar et al. (2023)	average online task accuracy; backward transfer over the task sequence	held-out sample
Merging and interpolation		
Wortsman et al. (2022b)	ImageNet accuracy on the reference and shifted distributions, fixed label set	fixed benchmark
Ilharco et al. (2023)	accuracy on control tasks; WikiText-103 perplexity	fixed benchmark
Yadav et al. (2023)	in-domain accuracy, median over the fine-tuning prompt templates	held-out sample
Hess et al. (2025)	CIFAR-100 accuracy under three cue probes, 100-way argmax	held-out sample
Izmailov et al. (2018)	test accuracy of the averaged model; no score of what is kept	held-out sample
Repeated interpolations toward the initialisation		
Lin et al. (2024)	alignment tax on fixed benchmarks: ARC, RACE, PIQA, SQuAD, DROP, WMT14	fixed benchmark
Stojanovski et al. (2022)	final class- and task-incremental accuracy over the classes seen	held-out sample

continued on the next page

Table A22, continued

paper	what it scores as retention	scored on
Ramé et al. (2024)	KL to the SFT policy on the training prompts; fixed benchmarks, scored once	training distribution
Noukhovitch et al. (2023)	language drift: BLEU, perplexity, HumanEval and KL on validation sets	held-out sample
Continued pre-training, LoRA and continual-learning baselines		
Ibrahim et al. (2024)	Pile validation loss; fixed zero- and few-shot benchmarks	training distribution
Hu et al. (2022)	no retention score; target-task scores only	none
Biderman et al. (2024)	mean accuracy on HellaSwag, ARC-Challenge and WinoGrande, scored by logit	fixed benchmark
Li & Hoiem (2018)	old-task accuracy and VOC mAP, original label set and head	held-out sample
Zenke et al. (2017)	accuracy of each earlier task and their mean, same split as trained	held-out sample
Chaudhry et al. (2019a)	average accuracy and forgetting measure, each task's own test set	held-out sample
Luo et al. (2023)	mean relative drop on fixed benchmarks, MMLU and eight others	fixed benchmark
Scialom et al. (2022)	relative gain on trained and held-out tasks; unseen and composed instructions	unseen instruction
Ke et al. (2023)	forgetting rate on end-task F1, scored after fine-tuning a new head	held-out sample
Knowledge editing and lifelong editing		
De Cao et al. (2021), Mitchell et al. (2022), Meng et al. (2022), Meng et al. (2023)	the edit prompt and its paraphrases after one write or one batch; locality on one wording	edit and paraphrase
Hartvigsen et al. (2023), Wang et al. (2024)	the edit prompt along the sequence; the paraphrase of every earlier edit at the end (WISE); one locality wording	edit and paraphrase
Gupta et al. (2025), Thede et al. (2025)	the edit prompt and its paraphrase along ten thousand edits under a Frobenius constraint, or at each time step with an adapter merge	edit and paraphrase
Gu et al. (2026), Wang et al. (2025)	efficacy or reliability of the edit prompt, generalisation on its paraphrase, specificity or locality on unrelated queries, up to two million edits	edit and paraphrase
Facts maintained by replaying them		
Chen et al. (2026a)	exact string match on the memorised factoids under the template they were trained with	replayed items
Jagielski et al. (2023)	extraction of the memorised training example along later training	replayed items
Li et al. (2025)	log-perplexity on held-out data of earlier months, with replay of those months	training distribution
Wu et al. (2024)	the training pairs (memorisation), their rephrasings and derived questions, judged by a model	replayed items and rephrasing
Jang et al. (2022b), Jang et al. (2022a)	cloze sentences or probe phrases built apart from the training text, one accuracy or perplexity over the stream	probes apart from training
Choi (2026)	multiple-choice log-likelihood on the training template and on one held-out paraphrase, averaged over the months of the stream, to rank two replay schedules; no stabiliser toward the pre-trained weights	template and paraphrase

Table A22: What each line scores retention on. Of the 23 stabiliser and continual-learning papers, 21 score a held-out sample of the trained tasks, a fixed benchmark or the training distribution, one scores unseen instruction wordings of a kept ability (Continual-T0 changes the task instruction, not a kept fact's wording), and one scores no retention. Editing scores the edited item's paraphrase as an aggregate axis beside the edit prompt, with the locality side in one wording; the lines that maintain facts by replaying them score the replayed items or the replayed distribution. *Main conclusion: no*

line pairs the two wordings of one maintained item, scores the two together under a stabiliser round by round, or accepts a stabiliser on their difference.

M WHICH MEASUREMENTS WERE REGISTERED IN ADVANCE

Table A23 lists, for each contrast of Table A2 and for the chat measurement of §L, the measurement registered before the runs ran; the companion repository holds each run's verdict and result files under its experiment number.

Table A23

contrast of Table A2	registered measurement	line
the run of Table A3	run with weight interpolation, paraphrase	every seed $\leq$ .22
	run without weight interpolation, paraphrase	every seed $\geq$ .35
reordered words	run with weight interpolation, training sentence minus its words reordered	$\geq$ .20
held-out facts, training sentence	run with weight interpolation, training sentence on the 50 held-out facts	$<$ midpoint
the same 50 facts on the float32 master (Figure 2b)	run with weight interpolation, training sentence on the 50 held-out facts	$<$ midpoint
public-knowledge-base facts	with over without, paraphrase	$<$ midpoint
	with over without, training sentence	$>$ midpoint
benchmark edits, LoRA update	with minus without, paraphrase prompt	$\leq -.177$
	with over without, edit prompt	$>$ midpoint
benchmark edits, MEMIT update	with minus without, paraphrase prompt	$\leq -.177$
	with over without, edit prompt	$>$ midpoint
Mistral-7B, $\lambda = .10$	run with weight interpolation, paraphrase	$<$ midpoint
	run with weight interpolation, training sentence	$>$ midpoint
OLMo-2-7B, $\lambda = .15$, training sentence	with over without, training sentence	$<$ midpoint
float32 master, matched distance	no interpolation minus matched, paraphrase	six seeds, $\geq$ .173
	matched over no interpolation, training sentence	six seeds, $\geq$ .98
interpolation alone $-$ run with weight interpolation, training sentence	interpolation alone, training sentence	$\geq$ .90
run with weight interpolation $-$ interpolation every ten rounds	every ten rounds over no interpolation, paraphrase	$<$ midpoint
run with weight interpolation $-$ one interpolation, ten times the dose	per-round minus one interpolation, paraphrase	$\leq -.10$
run with weight interpolation $-$ one interpolation, the standard dose	after over before weight interpolation, paraphrase	$<$ midpoint
the same gap, training sentence	after over before weight interpolation, training sentence	1.0 two-sided
anchor on the previous round's weights $-$ run with weight interpolation	previous-round anchor over no interpolation, paraphrase	$<$ midpoint
	previous-round anchor over no interpolation, training sentence	$>$ midpoint
Fisher-weighted penalty $-$ run with weight interpolation	penalty over no interpolation, paraphrase	$<$ midpoint
	penalty over no interpolation, training sentence	$>$ midpoint
the chat measurement of §L	training sentence under the chat measurement, no interpolation	$\geq$ .95
	with over without weight interpolation, paraphrase under the chat measurement	$\leq$ .50
the float32 ladder of Table 1	without minus with, paraphrase, least over the registered settings	$\geq$ .0595

Table A23 (continued)

contrast of Table A2	registered measurement	line
	training sentence, least over the registered settings	$\geq .95$
	training sentence one setting above the registered settings	$< .95$
the float32 ladder on Wikidata facts (Table A18)	without minus with, paraphrase, least over the registered settings	$\geq .0595$
	training sentence, least over the registered settings	$\geq .95$
	training sentence one setting above the registered settings	$< .95$
the families of Table A15	Fisher-weighted penalty minus per-round weight interpolation, paraphrase	$\geq .0587$
	L^2-SP penalty minus per-round weight interpolation, paraphrase	$\geq .0613$
the update's eight dimensions, bfloat16	run with weight interpolation plus the part in the update's dimensions, paraphrase with the training sentence held	$\geq .300$
	run without weight interpolation minus the same part, paraphrase	$\leq .300$
the arriving ladder under weight interpolation (§D.5)	paraphrase, weakest minus strongest interpolation	$\geq .0688$
the MLP exchange on the float32 master	run with weight interpolation plus the blocks, paraphrase	$\geq .297$
	the same, training sentence minus the run it is put into	within .02
the update's eight dimensions, float32	run with weight interpolation plus the part in the update's dimensions, paraphrase with the training sentence held	$\geq .297$
	run without weight interpolation minus the same part, paraphrase	$\leq .297$
150 rounds of arrival	with minus without, MMLU at round 150	$\geq .0108$
	run with weight interpolation, training sentence at round 150	$\geq .95$
weight interpolation by prior knowledge (§D.7)	share lost, facts known in both wordings minus facts known in neither	< 0
	share lost, facts known in the training sentence alone minus facts known in both	> 0
	share lost, facts known in neither minus facts known in the training sentence alone	> 0

Table A23: The measurement registered behind each contrast of Table A2, with the direction and line it was registered against, medians of three seeds unless stated. A ratio is with over without unless the row says otherwise; a midpoint line is the midpoint of two reference runs named in the registration. *With the training sentence held* means the paraphrase measured where the training sentence loses at most three of the 150 facts against the run the part is put into. *Main conclusion: every contrast of the body is measured against a line registered before the runs ran.*

The held-out check of §5. The 22 runs are the judged arms of three experiments (listed run by run with the repository), the Mistral-7B runs, the Qwen3-8B runs of six seeds on five schedules within .040 of one distance, and the float32 ladder; each arm's predicted distance and paraphrase were registered with a tolerance of one resolution; the curve lands within 1.2 bands of every one, and the two beyond one band are the run with weight interpolation on Mistral-7B and the float32 setting at .022.