When Forgetting is not Catastrophic: On the Mechanics of Spurious Forgetting

Anonymous authors
Paper under double-blind review

Abstract

Knowledge that a language model appears to forget during finetuning often remains stored and can be recovered, a phenomenon called spurious forgetting. Finetuning on new facts can even produce forgetting that undoes itself: recall of the old facts collapses, recovers as training continues on new facts alone, and only then erodes for good. We seek to understand when such forgetting is not catastrophic. A minimal associative memory reproduces these dynamics with three ingredients: keys with shared structure, concentrated new values, and normalization in the network. Finetuning moves all old representations along a common direction, hiding the old facts while preserving their relative geometry; normalization withdraws this shift once the new facts are learned, whereas fact-specific changes accumulate and cause the erosion. Moreover, subtracting the common shift eliminates the collapse in a Transformer trained on synthetic data, and removing a single direction from each weight update restores old facts in a pretrained language model. Forgetting thus combines a shared, reversible loss of access with a slow erosion of individual facts, and only the second is catastrophic. Which one dominates depends on whether the new data move old memories together or apart.

1 Introduction

Continually training a neural network on new data can severely damage the knowledge and skills it previously acquired, a phenomenon known as catastrophic forgetting (McCloskey & Cohen, 1989; Ratcliff, 1990; French, 1999). It has been reported across many settings, from image classification (Goodfellow et al., 2014; Zenke et al., 2017; Ramasesh et al., 2021) to deep reinforcement learning (Kirkpatrick et al., 2017; Rolnick et al., 2019), and large language models are no exception (Luo et al., 2025; Zucchet et al., 2025; Marek et al., 2026). Yet, a loss of performance need not be a loss of knowledge: a different prompt can restore lost capabilities (Kotha et al., 2024), and brief training on a few old examples can undo forgetting that reflects a loss of task alignment rather than of knowledge, named spurious forgetting (Zheng et al., 2025). Accuracy measured at a single point in training therefore cannot tell whether knowledge was erased or only hidden. What hides it, and whether training on the new data alone can reveal it again, remains unclear; we ask

When is forgetting not catastrophic?

We study this question for factual knowledge, as forgetting and loss of access can be cleanly disentangled. How language models store and retrieve facts is well understood (Geva et al., 2021; Meng et al., 2022; Geva et al., 2023; Nanda et al., 2023), and synthetic biographies (Allen-Zhu & Li, 2024; Zucchet et al., 2025) let us specify exactly which facts the model knows, guarantee that the new facts were never seen before, and control how the two relate. We can then ask not only whether the model still outputs an old fact, but also whether the fact is still stored. In this setting, finetuning on new facts alone produces spurious forgetting in a particularly clear form (Figure 1): recall of the old facts collapses, then recovers as training continues on the same new facts, before a slower decline sets in. Since there is no replay of the old facts, the recovery can only come from knowledge that survived the collapse.

Our contributions are as follows:

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.

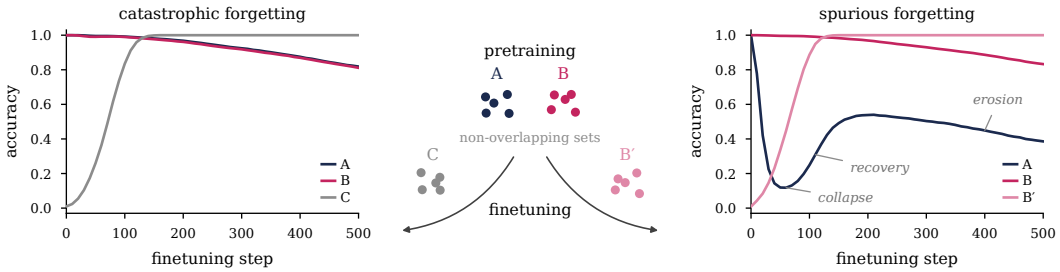


Figure 1: **Changing only the finetuning data turns catastrophic forgetting into spurious forgetting.** A transformer is pretrained on two non-overlapping sets of facts, A and B, and then finetuned on new facts only, without any replay. When the new facts overlap with both pretraining sets (left, fine-tuning on C), performance on both declines slowly and steadily. When they overlap only with B (right, fine-tuning on B'), performance on B is mostly intact, but performance on A collapses, recovers although A is never seen again, and later erodes. How the different sets are built and what it means for the new facts to overlap with the old ones is detailed in Section 4.

- **A clean case of spurious forgetting.** In a controlled factual-recall setting, finetuning on new facts alone makes recall of the old facts collapse and then return, without any replay, and changing only where the new answers lie turns the same finetuning into steady forgetting.
- **A minimal mechanism.** A minimal associative memory reproduces the phenomenon once three ingredients are present: keys with shared structure, new answers concentrated in one region, and normalization. Finetuning then moves all old representations together, which hides the old facts without erasing them, and normalization undoes this shift once the new facts are learned. Changes specific to each fact accumulate and cause a slower, lasting forgetting.
- **Causal evidence in trained models.** In a transformer, removing the common shift eliminates the collapse, and the recovery starts at the same stage of new-fact learning across a threefold range of learning speeds, as the mechanism predicts. In a pretrained language model, removing a single direction from each weight matrix's update restores the old facts, for synthetic and real new facts alike, beyond what training retains at the same level of new-fact learning.

Together, our results show that forgetting is not catastrophic when the update moves old memories together rather than apart: the knowledge is then hidden rather than erased, and can be recovered, sometimes by just continuing finetuning for longer.

2 Forgetting that undoes itself

To understand when forgetting is not catastrophic, we first need a setting in which it clearly is not: one in which performance collapses, yet the knowledge demonstrably survives (Appendix C.1). We use synthetic biographies (Allen-Zhu & Li, 2024), on which spurious forgetting was identified (Zheng et al., 2025), as they let us control what the model knows before finetuning and what it learns during it. A transformer is pretrained on two non-overlapping sets of biographies, A and B, in a single stage rather than one per set (Zheng et al., 2025), until it recalls all of their facts, and is then finetuned on biographies of new individuals, without any replay. We compare two sets of new biographies, C and B', which differ only in how they overlap with the old ones. We describe how they are built, and what overlapping means precisely, in Section 4; for now, the only relevant property is that C overlaps with both A and B, while B' overlaps only with B (Figure 1, middle).

Finetuning on C yields classical catastrophic forgetting behavior (Figure 1, left): as the new facts are learned, recall of both A and B declines slowly and steadily. Finetuning on B' barely affects recall of B, but recall of A follows a strikingly different trajectory (Figure 1, right), in three phases that we refer to throughout the paper:

- **Collapse:** recall of A quickly drops, well before the new facts are learned.
- **Recovery:** recall of A then partially returns as the new facts are acquired, although A never reappears in training.
- **Erosion:** finally, recall of A declines slowly and persistently, as in catastrophic forgetting.

This trajectory raises two questions: why finetuning on B' hides the facts of A while sparing those of B, and why the collapse reverses on its own. Section 3 answers both in a minimal model. Section 4 then returns to Transformers, and to the biography structure needed for the phenomenon to appear.

3 A MINIMAL MODEL REVEALS THE MECHANISM OF SPURIOUS FORGETTING

Factual recall, including that of synthetic biographies, can be abstracted as an associative memory: each fact maps a key, the subject it is about, to a value, its answer, which the model must retrieve when queried. We show that a minimal associative memory reproduces spurious forgetting once three ingredients are present: keys that share a common structure, new values concentrated in one part of the vocabulary, and a normalization layer. Its learning dynamics then explain the phenomenon: the structure of the data causes the collapse, and normalization drives the recovery.

3.1 REPRODUCING SPURIOUS FORGETTING IN AN ASSOCIATIVE MEMORY MODEL

Many phenomena of factual recall, such as how many facts a model can store as a function of its size (Allen-Zhu & Li, 2025; Nichani et al., 2025), how its loss scales (Cabannes et al., 2024) or the dynamics with which facts are learned (Zucchet et al., 2025; 2026), can be reproduced in a simple feedforward associative memory, which strips away the sequence-modeling layers of the transformer.

Associative memory task. Each fact i is represented by a key

$$k_i = \sqrt{\alpha}\,\mu + \sqrt{1-\alpha}\,e_i, \tag{1}$$

where μ is a unit vector shared by all facts, e_i a random unit vector specific to the fact, and α sets how much structure the keys share. Each key is associated with a value y_i, a token from a vocabulary of size V that we split into two halves. The old facts form two sets, A, whose values are drawn uniformly from one half, and B, whose values are drawn from the other. The new facts B' have new keys, sampled in the same way, and draw their values from the same half as B, just as the new biographies overlap with only one pretraining set in Figure 1 (right); drawing them from the entire vocabulary instead mirrors finetuning on C (left). The data thus combine keys that share structure with new values concentrated in one half of the vocabulary.

Model. The model is a two-layer network with an RMS normalization layer (Zhang & Sennrich, 2019) as the non-linearity between the two layers, followed by a softmax readout. It can be understood as the final linear layers of a Transformer. When queried with a key k_i, the model outputs a probability distribution over values,

$$p_i = \mathrm{softmax}(W_2 \mathrm{RMS}(h_i))\,, \quad \text{with } h_i = W_1 k_i. \tag{2}$$

The RMS normalization divides the hidden state by its norm and multiplies it by $\sqrt{d}$. The weight matrices W_1 and W_2 are trained with minibatch gradient descent on the cross-entropy loss, first on the pretraining data (A and B) and then on the finetuning data (B').

Each of these ingredients is necessary. We follow the old facts of A over finetuning. The minimal model reproduces their collapse, recovery and erosion (Figure 2), and removing any one of its ingredients breaks this trajectory. With less shared structure in the keys, the collapse slows down until only gradual forgetting remains (panel 1). Drawing the values of the new facts from the entire vocabulary, as in finetuning on C, likewise leaves only gradual forgetting (panel 3). Without normalization, or with a ReLU in its place, the old facts collapse and never recover (panel 2).

3.2 KEY SHARING LEADS TO COLLAPSE

Intuitively, a gradient step on a new fact moves not only its own hidden state but that of every key resembling its own. All keys share a common component, so whatever the updates of the new facts have in common is written into all hidden states at once, the more so as α grows. The result is a common shift that leaves the relative geometry of the hidden states mostly intact but, through the readout, favors the half of the vocabulary that contains the new answers, so that the facts of A can no longer be decoded.

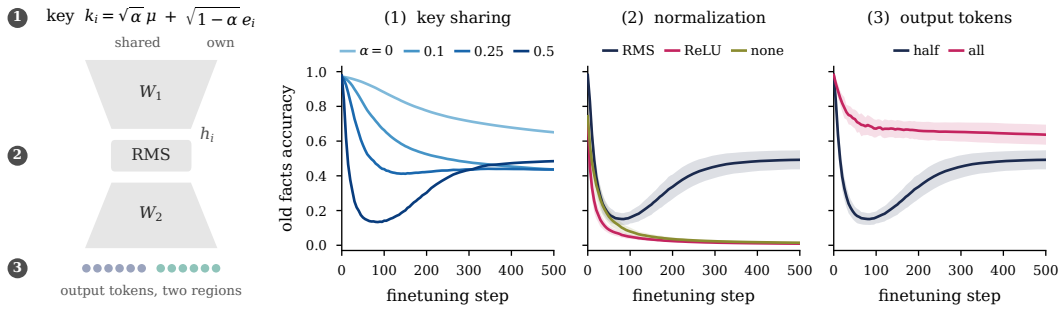


Figure 2: **Shared keys, concentrated new values and normalization are all needed for the collapse, recovery and erosion.** The model maps keys, which share a component of strength α, to values split into two regions, through one hidden layer with RMS normalization. As in Figure 1, it is pretrained on A and B, whose values lie in different regions, then finetuned only on new facts B', whose values lie in the region of B; curves show recall of A. Reducing α weakens the collapse and removes the recovery (1), replacing the normalization removes the recovery (2), and drawing the new values from both regions leaves only gradual forgetting (3). The model is described in Section 3.

All hidden states move in the same direction. When the model learns a new fact, gradient descent updates the first layer so that the hidden state of every key similar to the new fact's key moves against that fact's error. Since all keys, old and new, share a common component, they are all pushed in the same direction whenever the errors of the new facts agree. Formally, under gradient flow on the finetuning loss,

$$\dot{h}_a = \dot{W}_1 k_a = -\frac{1}{n_B} \sum_b \left(k_b^\top k_a\right) \delta_b, \quad \text{where } \delta_b = \nabla_{h_b} \ell_b. \tag{3}$$

In high dimension, the fact-specific parts of different keys are nearly orthogonal, so the overlap $k_b^\top k_a$ is close to its shared part α and, to leading order,

$$\dot{h}_a = -\alpha\, \bar{\delta}_B, \quad \text{where } \bar{\delta}_B = \frac{1}{n_B} \sum_b \delta_b. \tag{4}$$

The terms this approximation ignores vary in sign from one new fact to the next and largely cancel, so they are negligible compared with the one it keeps as long as the errors δ_b share a common component (Appendix B.4). All the old hidden states then move together in the same direction (Figure 3, left), at a speed proportional to α: this is why the collapse slows down as α decreases and disappears at $\alpha = 0$ (Figure 2, panel 1).

Whether the errors share a common component depends on the finetuning data: pretraining covers both halves of the vocabulary, so the model's predictions for the new keys are initially spread over both, whereas all the new answers lie in the same half, and on average the errors therefore all push probability toward this half. When the new answers are instead drawn from the entire vocabulary, this common push largely cancels, and so does the collapse (Figure 2, panel 3).

The common shift hides the facts of A. The collapse of A is a direct consequence of this common shift. Ignoring normalization for a moment, shifting all hidden states by the same vector adds the same bias to the logits of every old fact, favoring the half of the vocabulary that contains the new answers. The facts of A, whose answers lie in the other half, therefore lose to the new answers. Normalization amplifies this effect. Once the shift accounts for most of the norm of the hidden states, normalizing rotates all of them toward the shift direction: the old states become more similar to one another, the part of each state that encodes the fact's own answer is scaled down, and the predictions concentrate further on the same tokens. The facts of B, whose answers lie in the favored half, are largely spared by the shift itself, which raises the answers of their half roughly together, in proportion to how often each is a new answer: removing the shift from their hidden states barely changes their recall, whereas it restores that of A. As their answers are also those of the new facts, they are instead exposed to the fact-specific learning of the new facts, which erodes them steadily, mostly after the collapse (Section 3.3).

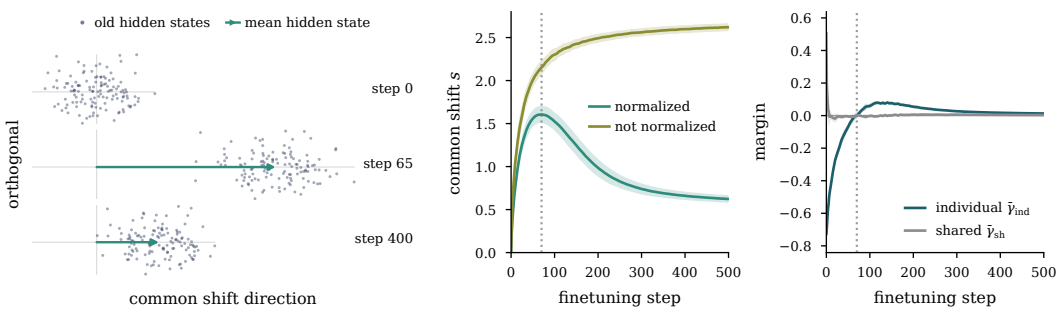


Figure 3: **Normalization reverses the shift responsible for forgetting.** The old representations first move together along a shared direction. Once the new facts are learned and the individual margin becomes positive, the common shift retracts and the old facts are recalled again. Without a normalization layer, this reversal does not occur, which is why the collapse is permanent (Figure 2, panel 2). (left) Old hidden states before normalization, projected on the shared direction and an orthogonal one, after pretraining, at the end of the collapse and during erosion. (middle) Common shift magnitude s of the old hidden states, with and without normalization. (right) Shared and individual margins of the new facts, which determine the dynamics of s through Equation 5. Same setup as in Figure 2; the dotted line marks the end of the collapse.

3.3 Normalization drives recovery

Early in finetuning, all hidden states are pushed in the same direction, and this common shift causes the collapse of A. Because the normalization fixes the norm of each hidden state, the growing shift comes at the expense of the individual parts, which encode each fact's own information and eventually become negligible. At first, this also helps the new facts, whose individual parts point to wrong answers. However, once the model has learned their individual associations, the shift drowns these as well and prevents the model from predicting the new facts correctly. Finetuning then undoes the shift, both on the new facts and on the old facts, restoring the individual parts of every hidden state. For the facts of A, these still carry most of the information that distinguishes them from one another, and the readout weights of their answers are barely changed by finetuning, as they only receive gradient while the model still puts probability on these answers for the new facts. The facts of A can therefore be read out again, and their recall partially recovers.

Normalization reverses the shift. We track the common shift through the norm s of the displacement of the old facts' mean hidden state $\bar{h}_A$ since the start of finetuning, and call its direction $\hat{u}_{\text{sh}}$ the shared direction. Through Equation 4, the dynamics of s are governed by whether pushing the hidden states of the new facts along the shared direction improves their predictions or not. Because of the normalization, such a push has two competing effects. On the one hand, it adds shared component, which improves the predictions on the new facts in proportion to the shared margin $\bar{\gamma}_{\text{sh}}$: the extra logit that the shared direction gives the correct answer compared with the model's current prediction, averaged over the new facts. On the other hand, it scales their individual parts down, which changes their predictions in proportion to the individual margin $\bar{\gamma}_{\text{ind}}$, the same quantity measured along their individual parts. Assuming for simplicity that all new hidden states have the same coordinate s along $\hat{u}_{\text{sh}}$ and the same individual norm σ, we obtain, to leading order,

$$\dot{s} = \alpha \sqrt{d}\, \frac{\sigma\,(\sigma\bar{\gamma}_{\text{sh}} - s\bar{\gamma}_{\text{ind}})}{(s^2+\sigma^2)^{3/2}}, \quad \text{with } s(0) = 0. \tag{5}$$

The first term corresponds to the added shared component and the second to the smaller individual parts; their relative size determines whether the common shift grows or shrinks. The derivation of this equation and the precise definitions of the two margins are given in Appendix B.5.

This equation explains the dynamics of the common shift in Figure 3. The shared margin only matters in the first steps of finetuning. The shared direction can move probability between the two halves of the vocabulary, but not between the answers within a half: $\bar{\gamma}_{\text{sh}}$ is positive at first, and once the predictions for the new facts have moved to the half of their answers, it can no longer improve them and remains close to zero (Figure 3, right). The shared margin thus sets the direction of the shift, and the individual margin then sets its magnitude. The new facts are initially confidently

wrong, $\bar{\gamma}_{\text{ind}} < 0$, so scaling their individual parts down helps them: s grows, and the growth reinforces itself. Once the individual parts of the new facts point to their correct answers, $\bar{\gamma}_{\text{ind}}$ becomes positive (Figure 3 right) and the same term pulls s back (Figure 3 middle). The shift therefore peaks when $\bar{\gamma}_{\text{ind}}$ crosses zero. This reversal waits for fact-specific learning, which is slow: the error of each new fact enters its own state with weight $1/n_B$, whereas their common component enters every state with weight α (Appendix B.4), in line with the general observation that shared structure is learned before item-specific structure (Saxe et al., 2014). Without normalization, adding a shared component no longer scales the individual parts down, the $\bar{\gamma}_{\text{ind}}$ term disappears in Equation 5, and $\dot{s} = \alpha\bar{\gamma}_{\text{sh}}$. Nothing pulls the shift back (Figure 3 middle) and the collapse is permanent (Figure 2).

The facts of A are hidden, not erased. Undoing the common shift only restores the facts of A if what distinguishes them from one another has survived, and it does, for two reasons. First, beyond the common shift, each old hidden state only receives the part of the update carried by the fact-specific parts of the new keys, which overlap weakly with its own key (Appendix B.3). This individual displacement stays small during the collapse, so while the mean of the old hidden states moves a lot, their relative geometry barely changes (Figure 3, left), which is why the facts of A keep their ordering within their own half at the bottom of the collapse, as they also do in the transformer (Appendix C.1). Second, the readout weights of the answers of A receive little gradient: they are only updated while the model puts probability on these answers for the new facts, which stops early in finetuning. Removing the part of the update that acts on the shared direction, $\Delta W_1(t)\,\mu$, while keeping everything else, including the finetuned readout, restores their recall almost entirely (Figure 8). The collapse is thus a loss of access to the facts of A, not a loss of the facts themselves.

The individual displacements nonetheless keep accumulating after the collapse, and they are not neutral: each old hidden state drifts toward the answers of the new facts whose keys overlap most with its own. For A, restoring the mean hidden state therefore helps less and less as finetuning proceeds, and recovery remains incomplete. For B, whose answers are those of the new facts, this drift, together with the readout weights of their half, which are refit to the new facts, erodes recall throughout finetuning, and removing the common shift barely helps.

4 The mechanism in a trained Transformer

Having identified the mechanism behind spurious forgetting in a minimal model, we now return to the Transformer of Figure 1 and test whether its collapse and recovery follow the mechanism of the minimal model, first by showing that its data meet the same conditions, then through measurements and interventions on its representations.

4.1 Training Transformers on synthetic biographies meets the conditions

The minimal model identifies three conditions for spurious forgetting: a normalization layer, keys that share structure, and new answers concentrated in one region of the output space. The first condition is met by any Transformer, as every residual block contains a normalization layer (Vaswani et al., 2017). In the pre-normalization architecture we use (Radford et al., 2019; Xiong et al., 2020), the last one sits just before the readout. The other two are properties of the data, which we now describe. The synthetic biographies (Allen-Zhu & Li, 2024; Zucchet et al., 2025) describe a large population of fictitious individuals, each with a unique name and a few attributes, such as a hometown or an employer. Each fact is stated by filling templates shared across all individuals, such as `The hometown of [name] is [city]` (Figure 4, left), and each set of biographies, A, B, B' or C, contains its own individuals, so that the new facts are about individuals the model has never seen. If we view the feedforward layers of a Transformer as an associative memory in which part of the context serves as the key and the next token as the value (Geva et al., 2021), the key of a fact is the name of the individual together with the attribute being queried, and its value is the answer. As all facts about a given attribute are queried in a similar way, their keys share structure by design, just as the keys of the minimal model share the component μ. The last condition is not present by default, so we build it into the data: we split the possible answers of each attribute at random into two subsets. The individuals of A draw their answers from the first subset and those of B from the second. The finetuning sets therefore overlap with the old sets through their answers alone: those of

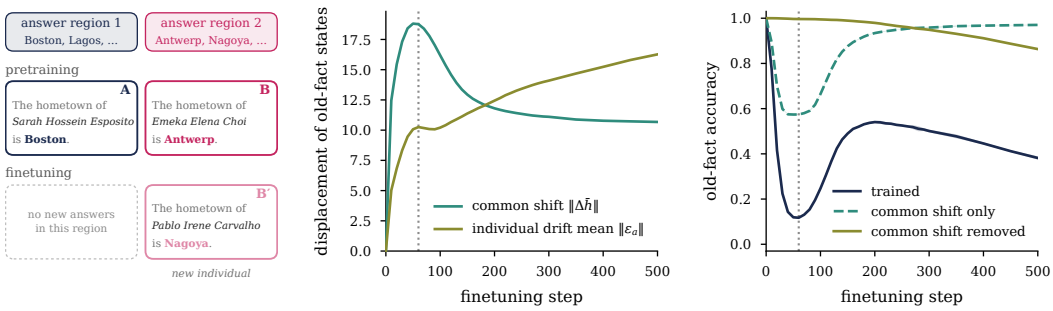


Figure 4: **The Transformer initially forgets through the same common shift as the minimal model.** (left) The Transformer is trained on synthetic biographies of fictitious individuals, each stating several facts about them, such as where they come from. The possible answers of each attribute are split into two subsets: the individuals of A draw their answers from the first, and those of B from the second. The model is pretrained on A and B, then finetuned on new individuals B' whose answers come from the same subset as B; this is the setup of Figure 1 (right). (middle) The common shift $\Delta\bar{h}$ of the hidden states of A grows fast and retracts at the end of the collapse, as in the minimal model (Figure 3). The fact-specific displacements ε_a grow more slowly and eventually become larger. (right) Accuracy on A, and when only the common part of the change of its logits is kept or removed. The dotted line indicates the end of the collapse.

B' draw their answers from the second subset, like B, and those of C from both subsets. Details of the data and the model are given in Appendix A.1.

4.2 The Transformer shows the same geometric split

In the minimal model, the hidden states of the old facts first move together: their mean moves away from its pretrained value until the end of the collapse and then comes back, while the fact-specific displacements keep growing. To test whether the same happens in the Transformer, we track the representations of the old facts of A at the input of the readout, at the position where the model predicts the first token of the answer, for example `Boston` in Figure 4 (left). As in the minimal model, we decompose the displacement of each representation since the start of finetuning into a common shift $\Delta\bar{h}(t)$, its average over the facts of A about the same attribute, and a fact-specific remainder $\varepsilon_a(t)$, whose size we measure by its average norm over the facts of A.

The two components evolve as the minimal model predicts (Figure 4, middle). The common shift grows quickly, peaks around the bottom of the collapse, and then partly retracts during the recovery, as in Figure 3 (middle). The fact-specific drift grows more slowly and, apart from a brief pause around the bottom of the collapse, keeps growing until it exceeds the common shift.

4.3 Subtracting the common shift removes the collapse

The minimal model predicts that the common shift alone causes the collapse: undoing it should restore the old facts, as long as what distinguishes them from one another has survived. Because the readout is linear and barely changes during finetuning, a common shift of the readout states changes the logits of all old facts by the same amount. We therefore test this prediction on the logits: for each old fact, we split the change of its logits since the start of finetuning into its average over the facts of A about the same attribute and a fact-specific remainder, and measure recall when only one of the two is kept.

As predicted, subtracting the common part removes the collapse entirely (Figure 4, right): the old facts remain almost perfectly recalled, and only the slow decline caused by the fact-specific drift is left. Keeping only the common part does the opposite and produces a collapse and a recovery by itself. This collapse is shallower than in the trained model, because the common shift brings the old facts close to losing to the new answer region and the fact-specific drift pushes many of them across. The collapse is therefore caused by the common shift, and erosion by the fact-specific drift.

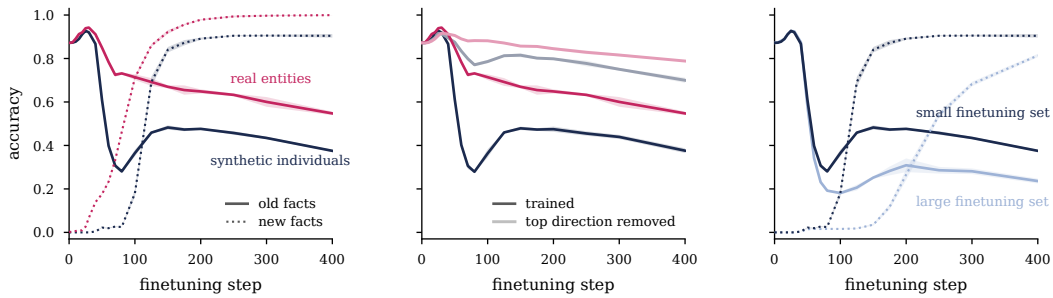


Figure 5: **The collapse persists in a pretrained language model and remains reversible.** OLMo 2 1B is finetuned without replay on new facts; old facts (known CounterFact facts) solid, new facts dotted, three seeds. (left) Both kinds of new facts make recall of the old facts drop, but only synthetic individuals let it recover. (middle) Removing the top singular direction of each weight matrix's update (light) restores recall of the old facts, compared with the trained model (dark, as on the left)

.

The shift is carried mostly by the attention value and output weights, with little contribution from the query and key weights and none from the unembedding (Appendix C.3). In our attention-only architecture, these are the weights that write content into the residual stream, whereas the query and key weights only decide where attention goes, so this is consistent with the common shift being written into the associative memories of the network, as in the minimal model. With MLP blocks, the same three phases appear, and subtracting the common shift again removes the collapse (Appendix C.4).

4.4 The learning rate changes the depth of the collapse

In the minimal model, the common shift reverses once the new facts start to be answered through their own associations, so the end of the collapse should be tied to the progress of the new facts rather than to a number of steps. Changing the learning rate and the batch size of finetuning confirms this (Appendix C.2): the bottom of the collapse moves by a factor of three in steps, but always falls at the same accuracy on the new facts.

The depth of the collapse, in contrast, increases with the learning rate. This effect lies outside our gradient-flow analysis, in which the learning rate only rescales time. It is consistent with the observation that larger learning rates cause more forgetting, in continual pretraining (Gupta et al., 2023; Ibrahim et al., 2024) as in finetuning at matched loss (Springer et al., 2025; Lin et al., 2026; Rofin et al., 2026), and shows that part of this effect acts on forgetting that is spurious, and reverses once the new facts are learned.

5 The collapse in a pretrained language model

The transformer of Section 4 meets the conditions of the minimal model by construction. In a pretrained language model, only one of them remains in our hands: we can still choose new facts whose answers the old facts never use. How its keys share structure, however, was set by pretraining on real text, and nothing guarantees that finetuning writes a shift common to all old facts, nor that it later withdraws it. We therefore test the mechanism where we no longer control it. The collapse persists, but whether it reverses depends on what the subjects of the new facts reveal about their answers.

5.1 The collapse persists, and reverses when the new facts are arbitrary

We finetune OLMo 2 1B (Team OLMo et al., 2025) on documents stating new facts, without replay, and track its recall of CounterFact facts (Meng et al., 2022) it already knows (Appendix A.2). No new answer shares its first token with an old one, so the new answers occupy a region of the output space the old facts do not use, as the answers of B' do for A. The shift is withdrawn only once the

shared direction no longer tells the new facts which answer to give within their region (Section 3.3), and the two kinds of new facts we use differ most in whether this happens. Synthetic individuals, invented names that receive real companies, cities, countries and languages at random, allow it: a name says nothing about its answers. Real entities, products from EntityQuestions (Sciavolino et al., 2021) whose manufacturer the model does not know, do not: the kind of product hints at the kind of manufacturer, as a warship is built by a shipyard.

With synthetic individuals, the trajectory of the controlled settings reappears (Figure 5, left): recall of the old facts collapses before almost any new fact is recalled, recovers as the new facts are learned, and then erodes. More new facts delay the recovery, as each drives its own association with weight $1/n_B$ (Section 3.3): with four times more individuals, the collapse lasts longer and goes deeper (Figure 5, right). With real entities, the old facts never recover and keep declining even after every new fact is recalled: the shared direction keeps helping the new facts, and the shift is never withdrawn.

5.2 Removing one direction of the update restores the old facts

The common shift is written by a rank-one part of the update (Equation 3). In a deep network, it need not be a single vector at any one layer, but what all new facts write coherently should still concentrate in the top direction of each weight update. We therefore subtract the top singular component of the update of every weight matrix in the transformer blocks, or, as a control, a random rank-one matrix of the same norm, and evaluate the edited model (Appendix D).

The edit removes the collapse, whereas the random control leaves recall where training left it (Figure 5, middle). With synthetic individuals, only the slow erosion remains. With real entities, the edit restores most of the old facts that training never brought back: their steady decline was mostly a shift that was never withdrawn, not erasure. The removed direction also lowers recall of the new facts, but the edit does not rewind finetuning: once the new facts are partly learned, the edited model recalls more old facts than the trained model did at the same accuracy on the new facts (Appendix D).

6 Discussion

That a loss of performance need not be a loss of knowledge is an old idea: psychology separates stored from accessible information (Tulving & Pearlstone, 1966; Ebbinghaus, 1885), and suppressed associations return once interference stops (Postman et al., 1968). Connectionist networks show the same savings, and training them on unrelated associations can even restore forgotten ones (Hetherington & Seidenberg, 1989; Hinton & Plaut, 1987; Harvey & Stone, 1996; Atkins & Murre, 1998). In deep networks, linear probes often recover old tasks that the outputs no longer solve, a phenomenon called shallow forgetting (Murata et al., 2020; Davari et al., 2022; Hess et al., 2024; Lanzillotta et al., 2026); performance drops sharply at the start of a new task before returning with some replay (De Lange et al., 2023; Zucchet et al., 2025); and language models recover lost capabilities with a different prompt (Kotha et al., 2024) or a few old examples, a case named spurious forgetting (Zheng et al., 2025). In the settings we identify, the finetuning that hides the old facts also restores them, without probe, prompt or replay, through a shared shift that normalization withdraws once the new facts are learned. This shift towards the new answers may be one concrete form of the loss of task alignment that Zheng et al. (2025) identify behind spurious forgetting.

Forgetting is usually measured as a single quantity that grows with training, whereas our results suggest two processes on different timescales: a loss of access shared by all old facts, set by what the new data have in common, and a slower erosion specific to each fact. The pretrained model also suggests a tension: when the kind of subject predicts the kind of answer, the new facts are learned faster, through the same shared direction whose shift then stays in place and keeps the old facts hidden. What helps a model generalize to new facts may thus be what keeps it from recovering the old ones.

Limitations. Our minimal model is deliberately extreme, and normalization-driven recovery is likely weaker on real data, although the collapse may be more general. In the pretrained model, the property that decides whether the collapse reverses is inferred rather than isolated, and we study a single model size.

AI USE STATEMENT

In this work, we used generative AI tools for to assist in the formalizing of proofs, with everything being reviewed by the authors. We have **not** used generative AI tools to generate synthetic data sets, develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, provide feedback on research methodology or experiments, assist with translation, clean and reformat dataset, support qualitative and thematic data analysis, interpret results. Additionally, we used generative AI tools to edit code and modify scientific figures. We have reviewed all AI-assisted work. LLM-edited code was verified and tested for correctness by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Code to reproduce all experiments and figures is available in this Repository, with instructions to run the minimal model, the transformer and the pretrained language model. Appendix A describes the data, architectures and training of the transformer and of OLMo 2 1B, including how the old and new facts are built from publicly available datasets (CounterFact and EntityQuestions); Appendix B.1 gives the sizes and training of the minimal model; and Appendices C and D describe the procedure of every analysis and intervention.

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of Language Models: Part 3.1, Knowledge Storage and Extraction. In *International Conference on Machine Learning*, 2024.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws. In *International Conference on Learning Representations*, 2025.

Paul W. B. Atkins and Jaap M. J. Murre. Recovery of Unrehearsed Items in Connectionist Models. *Connection Science*, 1998.

Vivien Cabannes, Elvis Dohmatob, and Alberto Bietti. Scaling Laws for Associative Memories. In *International Conference on Learning Representations*, 2024.

MohammadReza Davari, Nader Asadi, Sudhir Mudur, Rahaf Aljundi, and Eugene Belilovsky. Probing Representation Forgetting in Supervised and Unsupervised Continual Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Matthias De Lange, Gido M. van de Ven, and Tinne Tuytelaars. Continual Evaluation for Lifelong Learning: Identifying the Stability Gap. In *International Conference on Learning Representations*, 2023.

Hermann Ebbinghaus. *Über das Gedächtnis*. Duncker & Humblot, Leipzig, 1885.

Robert M. French. Catastrophic Forgetting in Connectionist Networks. *Trends in Cognitive Sciences*, 1999.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer Feed-Forward Layers Are Key-Value Memories. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting Recall of Factual Associations in Auto-Regressive Language Models. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

Ian J. Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An Empirical Investigation of Catastrophic Forgetting in Gradient-Based Neural Networks. In *International Conference on Learning Representations*, 2014.

Kshitij Gupta, Benjamin Thérien, Adam Ibrahim, Mats L. Richter, Quentin Anthony, Eugene Belilovsky, Irina Rish, and Timothée Lesort. Continual Pre-Training of Large Language Models: How to (Re)warm Your Model? *arXiv preprint arXiv:2308.04014*, 2023.

Inman Harvey and James V. Stone. Unicycling Helps Your French: Spontaneous Recovery of Associations by Learning Unrelated Tasks. *Neural Computation*, 1996.

Timm Hess, Eli Verwimp, Gido M. van de Ven, and Tinne Tuytelaars. Knowledge Accumulation in Continually Learned Representations and the Issue of Feature Forgetting. *Transactions on Machine Learning Research*, 2024.

Phil A. Hetherington and Mark S. Seidenberg. Is There 'Catastrophic Interference' in Connectionist Networks? In *Annual Conference of the Cognitive Science Society*, 1989.

Geoffrey E. Hinton and David C. Plaut. Using Fast Weights to Deblur Old Memories. In *Annual Conference of the Cognitive Science Society*, 1987.

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. *Transactions on Machine Learning Research*, 2024.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming Catastrophic Forgetting in Neural Networks. *Proceedings of the National Academy of Sciences*, 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.

Giulia Lanzillotta, Damiano Meier, and Thomas Hofmann. Heads Collapse, Features Stay: Why Replay Needs Big Buffers. In *International Conference on Learning Representations*, 2026.

Jiacheng Lin, Zhongruo Wang, Kun Qian, Tian Wang, Arvind Srinivasan, Hansi Zeng, Ruochen Jiao, Xie Zhou, Jiri Gesi, Dakuo Wang, et al. SFT Doesn't Always Hurt General Capabilities: Revisiting Domain-Specific Fine-Tuning in LLMs. In *International Conference on Learning Representations*, 2026.

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. *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

Martin Marek, Dongkyu Cho, Shikai Qiu, Rumi Chunara, Pavel Izmailov, and Andrew Gordon Wilson. Forgetting in Language Models: Capacity, Optimization, and Self-Generated Replay. *arXiv preprint arXiv:2605.26097*, 2026.

Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In *Psychology of Learning and Motivation*. 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.

Kengo Murata, Tetsuya Toyota, and Kouzou Ohara. What Is Happening Inside a Continual Learning Model? A Representation-Based Evaluation of Representational Forgetting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

Neel Nanda, Senthooran Rajamanoharan, János Kramár, and Rohin Shah. Fact Finding: Attempting to Reverse-Engineer Factual Recall on the Neuron Level. Alignment Forum, 2023.

Eshaan Nichani, Jason D. Lee, and Alberto Bietti. Understanding Factual Recall in Transformers via Associative Memories. In *International Conference on Learning Representations*, 2025.

Leo Postman, Karen Stark, and Janet Fraser. Temporal Changes in Interference. *Journal of Verbal Learning and Verbal Behavior*, 1968.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models Are Unsupervised Multitask Learners. Technical report, OpenAI, 2019.

Vinay V. Ramasesh, Ethan Dyer, and Maithra Raghu. Anatomy of Catastrophic Forgetting: Hidden Representations and Task Semantics. In *International Conference on Learning Representations*, 2021.

Roger Ratcliff. Connectionist Models of Recognition Memory: Constraints Imposed by Learning and Forgetting Functions. *Psychological Review*, 1990.

Mark Rofin, Aditya Varre, and Nicolas Flammarion. (How) Learning Rates Regulate Catastrophic Overtraining. In *Conference on Language Modeling*, 2026.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Greg Wayne. Experience Replay for Continual Learning. In *Advances in Neural Information Processing Systems*, 2019.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact Solutions to the Nonlinear Dynamics of Learning in Deep Linear Neural Networks. In *International Conference on Learning Representations*, 2014.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. Simple Entity-Centric Questions Challenge Dense Retrievers. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Jacob Mitchell Springer, Sachin Goyal, Kaiyue Wen, Tanishq Kumar, Xiang Yue, Sadhika Malladi, Graham Neubig, and Aditi Raghunathan. Overtrained Language Models Are Harder to Fine-Tune. In *International Conference on Machine Learning*, 2025.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 OLMo 2 Furious. In *Conference on Language Modeling*, 2025.

Endel Tulving and Zena Pearlstone. Availability versus Accessibility of Information in Memory for Words. *Journal of Verbal Learning and Verbal Behavior*, 1966.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Advances in Neural Information Processing Systems*, 2017.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On Layer Normalization in the Transformer Architecture. In *International Conference on Machine Learning*, 2020.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual Learning Through Synaptic Intelligence. In *International Conference on Machine Learning*, 2017.

Biao Zhang and Rico Sennrich. Root Mean Square Layer Normalization. In *Advances in Neural Information Processing Systems*, 2019.

Junhao Zheng, Xidi Cai, Shengjie Qiu, and Qianli Ma. Spurious Forgetting in Continual Learning of Language Models. In *International Conference on Learning Representations*, 2025.

Nicolas Zucchet, Jörg Bornschein, Stephanie C. Y. Chan, Andrew Kyle Lampinen, Razvan Pascanu, and Soham De. How Do Language Models Learn Facts? Dynamics, Curricula and Hallucinations. In *Conference on Language Modeling*, 2025.

Nicolas Zucchet, Hyun Dong Lee, and Scott Linderman. Language Models Suffer From a Curse of Ambiguity. *arXiv preprint arXiv:2608.15448*, 2026.

APPENDIX

A Experimental setup **14**

B Analysis of the minimal model **16**

C Transformer **26**

D Removing one direction of the update **28**

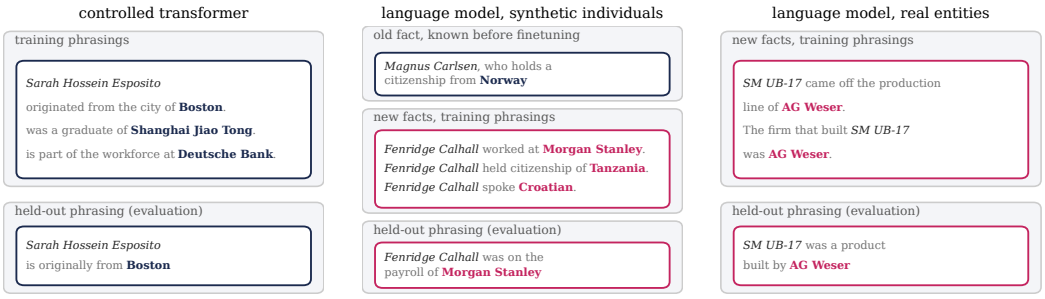


Figure 6: **Examples from the three datasets.** Each fact is learned from training phrasings and evaluated on held-out phrasings; answers in bold. The old facts of the language model are facts it already knows, and the new facts concern synthetic individuals, whose answers are unrelated to their names, or little-known real entities.

A Experimental setup

A.1 Controlled transformer

Data. The transformer is trained on synthetic biographies (Allen-Zhu & Li, 2024; Zucchet et al., 2025). Each individual has a unique name, made of a first, a middle and a last name, and six attributes: birth year, birthplace, university, field of study, employer and work city. Each attribute takes one value from a fixed pool of between 127 and 221 values. Every fact is stated in a sentence drawn from 25 phrasings of its attribute; for each individual and attribute, 20 phrasings are used for training and the remaining 5 are held out for evaluation. A biography consists of one sentence per attribute, in random order.

Old and new facts. The possible values of each attribute are split at random into two halves. Two pretraining populations of 2,000 individuals each draw their values from one half and from the other, respectively. Fine-tuning uses 500 new individuals, none of whom appear in pretraining, whose values are drawn either from the second half only (Figure 1, right) or from both halves (Figure 1, left).

Model and training. The model is an 8-layer transformer with model dimension 512, 8 attention heads, rotary position embeddings and pre-layer normalization, without MLP blocks and without dropout. It is pretrained for 16,000 steps with batches of 256 biographies, using AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0.1, gradient clipping at 1) with a cosine learning-rate schedule from $5{\times}10^{-4}$, until it recalls the pretraining facts perfectly. Fine-tuning uses the same optimizer and batch size with a peak learning rate of 3.75×10^{-5} over 1,200 steps, and sees only the new individuals. Results are averaged over three to five fine-tuning seeds.

Evaluation. Recall is the accuracy on the first token of the attribute value, predicted from the held-out phrasings of each fact. The within-region rank used in Appendix C.1 ranks the correct value among the values of its own half.

A.2 Pretrained language model

Old facts. The old facts are CounterFact facts (Meng et al., 2022) that OLMo 2 1B (Team OLMo et al., 2025) answers correctly before fine-tuning, 1,106 in total across relations such as manufacturers, headquarters and citizenship. Evaluation uses the 422 facts whose answer does not appear in the prompt, so that accuracy measures recall rather than copying.

New facts. The synthetic individuals have invented names and four attributes (employer, birth city, citizenship and language), with values drawn from pools of 66 real entities whose first tokens differ from those of the old answers. The large fine-tuning set contains 1,000 individuals and the small set 250. The real entities are products from EntityQuestions (Sciavolino et al., 2021) with

their manufacturers: 1,415 facts over 407 manufacturers, restricted to facts the model does not know before fine-tuning, whose answers share no first token with the old answers and do not appear in the subject's name, with at most 60 facts per manufacturer. Each new fact is stated through 15 phrasings, 12 for training and 3 held out for evaluation.

Fine-tuning. Training documents are rendered afresh at every step from the training phrasings, with 32,768 tokens per step in sequences of 512 tokens. We use AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, no weight decay, gradient clipping at 1) with a learning rate of 10^{-5} after a 100-step linear warmup, and no replay of any pretraining data. Results are averaged over three seeds.

Evaluation. Recall is the top-1 accuracy of the first answer token over the full vocabulary, on the held-out phrasings for the new facts and on the CounterFact prompts for the old facts.

B ANALYSIS OF THE MINIMAL MODEL

This section derives the results of Sections 3.2 and 3.3: the decomposition of the displacement of the old facts, how the new facts write into them, the dynamics of the common shift, and the size of the fact-specific displacements. We also describe how we verify these results in simulations, and the interventions we run on the minimal model.

B.1 NOTATION, ASSUMPTIONS AND SIMULATIONS

Recall that each fact i is represented by a key

$$k_i = \sqrt{\alpha}\,\mu + \sqrt{1-\alpha}\,e_i \in \mathbb{R}^d, \tag{6}$$

with μ a unit vector shared across all facts and e_i drawn uniformly on the unit sphere, independently for each fact. The model computes

$$h_i = W_1 k_i, \quad x_i = \mathrm{RMS}(h_i) = \sqrt{d}\,\frac{h_i}{\|h_i\|}, \quad z_i = W_2 x_i, \quad p_i = \mathrm{softmax}(z_i), \tag{7}$$

so that every normalized hidden state has norm $\sqrt{d}$. During finetuning, the loss is the cross-entropy on the new facts only,

$$L_B = \frac{1}{n_B}\sum_{b\in B} \ell_b, \quad \ell_b = -\log p_{b,y_b}. \tag{8}$$

We use the index a for old facts (set A) and b for new facts (set B', written B in subscripts), bars for averages over a set (e.g. $\bar{h}_A = \frac{1}{n_A}\sum_a h_a$), hats for unit vectors (e.g. $\hat{h} = h/\|h\|$), and $\mathbf{1}_y$ for the one-hot vector of token y. Time $t = 0$ corresponds to the beginning of finetuning, and $\Delta W_1 = W_1(t) - W_1(0)$.

Assumptions. The analysis relies on four assumptions, each used only where stated:

- Asm. 1 **Gradient flow dynamics.** We study the gradient flow $\dot{W}_1 = -\nabla_{W_1} L_B$, $\dot{W}_2 = -\nabla_{W_2} L_B$ on the finetuning loss. Minibatch gradient descent with a small learning rate η, which we use in our simulations, approximates it, one step corresponding to a time increment η.
- Asm. 2 **High-dimensional keys.** The overlap $u^\top v$ between two independent random unit vectors of $\mathbb{R}^d$ has zero mean and variance $1/d$. We therefore treat all the overlaps $e_i^\top e_j$ ($i \neq j$) and $\mu^\top e_i$ as small, of order $d^{-1/2}$, and only keep the leading-order terms in these overlaps.
- Asm. 3 **The update does not depend on the old keys.** The old facts are not trained during finetuning, so ΔW_1 only depends on their keys through the parameters at the end of pretraining. We neglect this dependence and treat the e_a as independent of ΔW_1. This assumption is only used in Section B.9.
- Asm. 4 **Common coordinate.** All new hidden states have the same coordinate along the shared direction $\hat{u}_{\mathrm{sh}}$ and the same norm orthogonal to it. This assumption is only used to obtain the closed-form Equation 5 (Section B.6); the full dynamics do not rely on it.

The readout W_2 does not appear in the old hidden states $h_a = W_1 k_a$. All the results on the hidden states below therefore hold whether the readout is trained or not: W_2 only enters through the errors of the new facts, evaluated at its current value. We return to the effect of the readout on the outputs of the old facts in Section B.10.

Simulations. All simulations use $d = 128$, a vocabulary of $V = 32$ tokens split into two halves of 16, and $n = 128$ facts in each of A, B and B', with $\alpha = 0.5$ unless stated otherwise. The model is pretrained on A and B with minibatches of 32 facts until both sets are recalled with accuracy at least 0.99, then finetuned on B' alone with minibatches of 32 new facts, at a learning rate chosen so that the new facts reach the same accuracy around step 200. Results are averaged over ten seeds unless stated otherwise.

B.2 GRADIENTS OF THE FINETUNING LOSS

We first compute the gradients of the finetuning loss with respect to the hidden states and to the two weight matrices, which the rest of the analysis uses throughout.

Error at the logits and at the normalized state. For the cross-entropy loss, the error of the new fact b with respect to its logits is

$$\nabla_{z_b} \ell_b = p_b - \mathbf{1}_{y_b}, \tag{9}$$

and, as $z_b = W_2 x_b$, its error with respect to the normalized state is

$$\nabla_{x_b} \ell_b = W_2^\top \big(p_b - \mathbf{1}_{y_b}\big). \tag{10}$$

Error at the hidden state. We denote by $\delta_b := \nabla_{h_b} \ell_b$ the error of the new fact at its hidden state. Without normalization, $x_b = h_b$ and $\delta_b = W_2^\top \big(p_b - \mathbf{1}_{y_b}\big)$. With normalization, the Jacobian of $h \mapsto \sqrt{d}\, h/\|h\|$ is

$$\frac{\partial x}{\partial h} = \sqrt{d} \left(\frac{I}{\|h\|} - \frac{h h^\top}{\|h\|^3} \right) = \frac{\sqrt{d}}{\|h\|} \left(I - \hat{h} \hat{h}^\top \right). \tag{11}$$

It is symmetric, so

$$\delta_b = \frac{\sqrt{d}}{\|h_b\|} \left(I - \hat{h}_b \hat{h}_b^\top \right) W_2^\top \big(p_b - \mathbf{1}_{y_b}\big), \tag{12}$$

which is the expression of the main text. The normalization has two effects on the error: it divides it by the norm of the state, and it removes its component along the state, $\hat{h}_b^\top \delta_b = 0$, since the loss does not depend on the norm of h_b.

The removed component has a simple interpretation. As $\hat{h}_b^\top W_2^\top \big(p_b - \mathbf{1}_{y_b}\big) = d^{-1/2} x_b^\top W_2^\top (p_b - \mathbf{1}_{y_b}) = d^{-1/2} z_b^\top (p_b - \mathbf{1}_{y_b})$, we have

$$\delta_b = \frac{\sqrt{d}}{\|h_b\|} W_2^\top \big(p_b - \mathbf{1}_{y_b}\big) + \frac{\gamma_b}{\|h_b\|} \hat{h}_b, \quad \text{with } \gamma_b := z_{b,y_b} - p_b^\top z_b. \tag{13}$$

Here γ_b is the confidence margin of the new fact: the logit of the correct answer minus the average logit under the model's prediction. It is negative when the model confidently predicts a wrong answer and positive when it predicts the correct one. The first term is, up to a rescaling, the error a model without normalization would receive. The second term points along the new fact's own state, with a sign given by its margin; for the new fact itself, it cancels the component of the first term along $\hat{h}_b$.

Gradients with respect to the weights. As $h_b = W_1 k_b$, the gradient of the loss of the new fact b with respect to W_1 is the outer product of its error and its key, $\nabla_{W_1} \ell_b = \delta_b k_b^\top$, and averaging over the new facts gives

$$\nabla_{W_1} L_B = \frac{1}{n_B} \sum_b \delta_b k_b^\top. \tag{14}$$

Similarly, as $z_b = W_2 x_b$,

$$\nabla_{W_2} L_B = \frac{1}{n_B} \sum_b \big(p_b - \mathbf{1}_{y_b}\big)\, x_b^\top. \tag{15}$$

As W_2 sits after the normalization, its gradient does not involve the Jacobian above: unlike W_1, it receives no term along the new facts' states.

B.3 DECOMPOSITION OF THE DISPLACEMENT OF THE OLD FACTS

The keys are fixed and the old facts are not trained, so the displacement of an old hidden state is entirely due to the change of W_1:

$$h_a(t) - h_a(0) = \Delta W_1 k_a = \underbrace{\sqrt{\alpha}\, \Delta W_1 \mu}_{\text{common shift}} + \underbrace{\sqrt{1-\alpha}\, \Delta W_1 e_a}_{\text{individual displacement}}. \tag{16}$$

This decomposition is exact. Averaging it over the old facts gives

$$\bar{h}_A(t) - \bar{h}_A(0) = \sqrt{\alpha}\,\Delta W_1 \mu + \sqrt{1-\alpha}\,\Delta W_1 \bar{e}_A, \tag{17}$$

and the deviation of each old fact from this mean motion is

$$\varepsilon_a := \big(h_a(t) - h_a(0)\big) - \big(\bar{h}_A(t) - \bar{h}_A(0)\big) = \sqrt{1-\alpha}\,\Delta W_1 (e_a - \bar{e}_A). \tag{18}$$

The mean displacement thus equals the common shift $\sqrt{\alpha}\,\Delta W_1 \mu$ up to the term $\sqrt{1-\alpha}\,\Delta W_1 \bar{e}_A$. As $\bar{e}_A$ is the average of n_A independent random unit vectors, $\mathbb{E}\|\bar{e}_A\|^2 = 1/n_A$, and the calculation of Section B.9 shows that this term is a factor $\sqrt{n_A}$ smaller than a typical individual displacement ε_a. We therefore measure the common motion through the norm of the mean displacement, $s = \|\bar{h}_A(t) - \bar{h}_A(0)\|$ (Figure 3, middle), and the fact-specific motion through the spread

$$\varepsilon_{\mathrm{rms}}(t) := \left(\frac{1}{n_A} \sum_a \|\varepsilon_a(t)\|^2 \right)^{1/2} \tag{19}$$

(Figure 9). When $\alpha = 0$, the common shift vanishes identically, and the whole displacement is fact-specific.

B.4 How the new facts write into the old hidden states

The write of a single new fact. Under the gradient flow dynamics, Equation 14 gives

$$\dot{W}_1 = -\nabla_{W_1} L_B = -\frac{1}{n_B} \sum_b \delta_b k_b^\top, \tag{20}$$

so each new fact writes its error δ_b into W_1 along its own key. The old facts are not part of the finetuning loss, but their hidden states move with W_1, and as their keys are fixed,

$$\dot{h}_a = \frac{\mathrm{d}}{\mathrm{d}t}\big(W_1 k_a\big) = \dot{W}_1 k_a = -\frac{1}{n_B} \sum_b \big(k_b^\top k_a\big)\,\delta_b, \tag{21}$$

where we used that $k_b^\top k_a$ is a scalar. Each new fact thus adds its error δ_b to every old state, weighted by the overlap of their keys. This equation is exact.

Key overlaps. Expanding the keys and using $\mu^\top \mu = 1$, the overlap between a new and an old key is

$$k_b^\top k_a = \alpha + \sqrt{\alpha(1-\alpha)}\,\big(\mu^\top e_a + \mu^\top e_b\big) + (1-\alpha)\, e_b^\top e_a. \tag{22}$$

Under Assumption 2, the last three terms have zero mean and are of order $d^{-1/2}$, so every pair of keys overlaps by approximately α. For the same reason, $\|k_b\|^2 = 1 + 2\sqrt{\alpha(1-\alpha)}\,\mu^\top e_b \simeq 1$.

The mean write. Equation 21 is linear in k_a, so averaging it over the old facts amounts to replacing k_a by their mean key:

$$\dot{\bar{h}}_A = \frac{1}{n_A} \sum_a \dot{h}_a = -\frac{1}{n_B} \sum_b \big(k_b^\top \bar{k}_A\big)\,\delta_b, \quad \text{with } \bar{k}_A = \sqrt{\alpha}\,\mu + \sqrt{1-\alpha}\,\bar{e}_A. \tag{23}$$

The average leaves the shared vector μ untouched and reduces the fact-specific parts to $\bar{e}_A$. The overlap of a new key with the mean old key is

$$\begin{aligned} k_b^\top \bar{k}_A &= \big(\sqrt{\alpha}\,\mu + \sqrt{1-\alpha}\,e_b\big)^\top \big(\sqrt{\alpha}\,\mu + \sqrt{1-\alpha}\,\bar{e}_A\big) \\ &= \alpha + \sqrt{\alpha(1-\alpha)}\,\mu^\top \bar{e}_A + \sqrt{\alpha(1-\alpha)}\,\mu^\top e_b + (1-\alpha)\, e_b^\top \bar{e}_A \\ &= \alpha\,(1+\xi) + \zeta_b, \end{aligned} \tag{24}$$

where we grouped the terms that are the same for all new facts into

$$\xi := \sqrt{\frac{1-\alpha}{\alpha}}\,\mu^\top \bar{e}_A, \tag{25}$$

and the terms that depend on the new fact into

$$\zeta_b := \sqrt{\alpha(1-\alpha)}\,\mu^\top e_b + (1-\alpha)\,e_b^\top \bar{e}_A. \tag{26}$$

Plugging this decomposition into Equation 23 and using $\frac{1}{n_B}\sum_b \delta_b = \bar{\delta}_B$ gives

$$\dot{\bar{h}}_A = \underbrace{-\alpha\,(1+\xi)\,\bar{\delta}_B}_{\text{shared}} \underbrace{-\frac{1}{n_B}\sum_b \zeta_b\,\delta_b}_{\text{specific}}. \tag{27}$$

The sizes of the two terms are as follows.

- **Shared term.** All new facts contribute to it with the same coefficient $\alpha(1+\xi)$. As $\bar{e}_A$ is the average of n_A independent random unit vectors, $\mu^\top \bar{e}_A$ has zero mean and variance $1/(n_A d)$, so ξ is of order $(n_A d)^{-1/2}$ and the coefficient is α to leading order.
- **Specific term.** $\mu^\top e_b$ has variance $1/d$ and $e_b^\top \bar{e}_A$ has variance $1/(n_A d)$, so the coefficients ζ_b have zero mean and are of order $d^{-1/2}$. They are moreover uncorrelated across new facts, as e_b and $e_{b'}$ are independent for $b \neq b'$. Treating them, to leading order, as independent of the errors δ_b gives

$$\mathbb{E}\left\|\frac{1}{n_B}\sum_b \zeta_b\,\delta_b\right\|^2 = \frac{1}{n_B^2}\sum_b \mathbb{E}\big[\zeta_b^2\big]\,\|\delta_b\|^2 = O\!\left(\frac{\|\delta\|_{\text{rms}}^2}{n_B\,d}\right), \tag{28}$$

  with $\|\delta\|_{\text{rms}}^2 = \frac{1}{n_B}\sum_b \|\delta_b\|^2$. The cross terms vanish because the coefficients are uncorrelated: the n_B contributions add up like a random walk, and their sum grows like $\sqrt{n_B}$ instead of n_B.

The specific term is therefore negligible compared with the shared one as soon as $\alpha\|\bar{\delta}_B\| \gg \|\delta\|_{\text{rms}}/\sqrt{n_B d}$, that is, as soon as the errors of the new facts share a common component. Keeping only the leading order of the shared term gives Equation 4,

$$\dot{\bar{h}}_A \simeq -\alpha\,\bar{\delta}_B, \tag{29}$$

and when $\alpha = 0$, only the specific term is left.

Why the errors of the new facts share a common component. The shared term is only large if the errors δ_b do not cancel when averaged over the new facts. Averaging the decomposition of Equation 13 over the new facts gives

$$\bar{\delta}_B = \frac{1}{n_B}\sum_b \frac{\sqrt{d}}{\|h_b\|}\,W_2^\top\big(p_b - \mathbf{1}_{y_b}\big) + \frac{1}{n_B}\sum_b \frac{\gamma_b}{\|h_b\|}\,\hat{h}_b. \tag{30}$$

Early in finetuning, both terms survive the average, for two reasons that both come from the structure of the data:

- **The new answers all lie in the same region.** Up to the variations of the prefactor $\sqrt{d}/\|h_b\|$ across new facts, the first term is proportional to $W_2^\top(\bar{p}_B - q_B)$, where $\bar{p}_B$ is the average prediction on the new facts and $q_B = \frac{1}{n_B}\sum_b \mathbf{1}_{y_b}$ the distribution of their answers. All new answers belong to the same half of the vocabulary, so q_B puts all its mass on that half. At the beginning of finetuning, the model has no reason to favor this half for the new keys: it was pretrained on answers from both halves, and its predictions for unseen keys spread over both. $\bar{p}_B - q_B$ therefore has a large component that moves probability mass from one half to the other, and this component is the same for all new facts. By contrast, which answer is correct within the half, and which wrong answer the model predicts, change from one new fact to the other and cancel in the average. When the new answers are drawn from the whole vocabulary, q_B and $\bar{p}_B$ spread their mass over the two halves in a similar way and this component largely disappears, which is why the collapse does too (Figure 2, panel 3).

– **The new facts are confidently wrong.** At the beginning of finetuning, the model confidently predicts wrong answers for the new facts, so their margins γ_b are all negative. At the same time, all new hidden states contain the shared component $\sqrt{\alpha}\, W_1 \mu$. The second term is thus a sum of vectors that share a direction, weighted by coefficients of the same sign, and it does not cancel either. This term gives rise to the $\gamma_{\mathrm{ind},b}$ contribution of Section B.5.

Both contributions vanish as the new facts are learned: when $p_b \to \mathbf{1}_{y_b}$, $W_2^\top (p_b - \mathbf{1}_{y_b}) \to 0$ and $\gamma_b \to 0$. The common component of the errors, and with it the shared write, therefore only exists early in finetuning.

Why the common write is faster than fact-specific learning. The same calculation applied to a new fact b separates its own contribution from that of the other new facts:

$$\dot{h}_b = -\frac{1}{n_B}\|k_b\|^2\, \delta_b - \frac{1}{n_B}\sum_{b' \neq b} \left(k_{b'}^\top k_b\right) \delta_{b'} \simeq -\frac{1}{n_B}\, \delta_b - \alpha\, \bar{\delta}_B, \tag{31}$$

where we used $\|k_b\|^2 \simeq 1$. The error of a single fact enters its own state with weight $1/n_B$, whereas the common component of all errors enters every state with weight α. Whatever the errors of the new facts have in common is therefore learned much faster than what is specific to each, which is why the reversal of the common shift waits for fact-specific learning.

B.5 Dynamics of the common shift

The normalization term of Equation 13 cancels along $\hat{h}_b$ for the new fact itself, but not for the old facts: through Equation 21, it moves every old state along the parts that its state shares with h_b. We make this precise by following the displacement of the old facts' mean state. Let $s = \|\bar{h}_A(t) - \bar{h}_A(0)\|$ be its length, which we call the shared coordinate, and $\hat{u}_{\mathrm{sh}} = (\bar{h}_A(t) - \bar{h}_A(0))/s$ its direction, which we call the shared direction. As $\frac{\mathrm{d}}{\mathrm{d}t}\|v\| = \hat{v}^\top \dot{v}$ for any vector v, and $\bar{h}_A(0)$ is constant, Equation 4 gives

$$\dot{s} = \hat{u}_{\mathrm{sh}}^\top \dot{\bar{h}}_A \simeq -\frac{\alpha}{n_B}\sum_b \hat{u}_{\mathrm{sh}}^\top \delta_b. \tag{32}$$

Validity of the approximation. Equation 4 neglects the specific term of Equation 27, and only its projection onto $\hat{u}_{\mathrm{sh}}$ matters for the dynamics of s. The shared term contributes α times the mean of the projected errors $\hat{u}_{\mathrm{sh}}^\top \delta_b$, while the specific term, whose coefficients are random and of order $d^{-1/2}$, contributes a quantity of the order of their root mean square divided by $\sqrt{n_B d}$. The approximation therefore requires the projected errors to share a sign across new facts, not the errors to be large. This holds while the new facts are all wrong and once they are all learned, even when their errors become small and mostly fact-specific. It only fails around the peak of s, where the projected errors change sign at different times for different new facts and $\dot{s}$ is small anyway.

Projecting the error. We decompose the direction of each new state into its component along $\hat{u}_{\mathrm{sh}}$ and an orthogonal component,

$$\hat{h}_b = \cos\theta_b\, \hat{u}_{\mathrm{sh}} + \sin\theta_b\, \hat{u}_{\mathrm{ind},b}, \tag{33}$$

where θ_b is the angle between h_b and $\hat{u}_{\mathrm{sh}}$, and $\hat{u}_{\mathrm{ind},b}$ is the unit vector along the individual part $h_b - (\hat{u}_{\mathrm{sh}}^\top h_b)\, \hat{u}_{\mathrm{sh}}$. Then

$$\begin{aligned} \hat{u}_{\mathrm{sh}}^\top \left(I - \hat{h}_b \hat{h}_b^\top\right) &= \hat{u}_{\mathrm{sh}}^\top - \cos\theta_b\, \hat{h}_b^\top \\ &= \hat{u}_{\mathrm{sh}}^\top - \cos^2\theta_b\, \hat{u}_{\mathrm{sh}}^\top - \cos\theta_b \sin\theta_b\, \hat{u}_{\mathrm{ind},b}^\top \\ &= \sin^2\theta_b\, \hat{u}_{\mathrm{sh}}^\top - \sin\theta_b \cos\theta_b\, \hat{u}_{\mathrm{ind},b}^\top, \end{aligned} \tag{34}$$

so that

$$\hat{u}_{\mathrm{sh}}^\top \delta_b = \frac{\sqrt{d}}{\|h_b\|}\left(\sin^2\theta_b\, \hat{u}_{\mathrm{sh}}^\top W_2^\top (p_b - \mathbf{1}_{y_b}) - \sin\theta_b \cos\theta_b\, \hat{u}_{\mathrm{ind},b}^\top W_2^\top (p_b - \mathbf{1}_{y_b})\right). \tag{35}$$

Two margins. The two projections of the error are margins:

$$\hat{u}_{\mathrm{sh}}^{\top} W_2^{\top}\big(p_b - \mathbf{1}_{y_b}\big) = (W_2\hat{u}_{\mathrm{sh}})^{\top}\big(p_b - \mathbf{1}_{y_b}\big) = -\gamma_{\mathrm{sh},b}, \quad \text{with } \gamma_{\mathrm{sh},b} := (W_2\hat{u}_{\mathrm{sh}})_{y_b} - p_b^{\top} W_2\hat{u}_{\mathrm{sh}}, \tag{36}$$

and similarly $\hat{u}_{\mathrm{ind},b}^{\top} W_2^{\top}\big(p_b - \mathbf{1}_{y_b}\big) = -\gamma_{\mathrm{ind},b}$ with $\gamma_{\mathrm{ind},b} := (W_2\hat{u}_{\mathrm{ind},b})_{y_b} - p_b^{\top} W_2\hat{u}_{\mathrm{ind},b}$. The shared margin $\gamma_{\mathrm{sh},b}$ is the margin by which the readout of the shared direction favors the correct answer of b over the model's current prediction, and the individual margin $\gamma_{\mathrm{ind},b}$ is the same quantity for the individual direction of b. As $z_b = \sqrt{d}\,(\cos\theta_b\, W_2\hat{u}_{\mathrm{sh}} + \sin\theta_b\, W_2\hat{u}_{\mathrm{ind},b})$, the two margins decompose the confidence margin γ_b of Equation 13 into a shared and an individual contribution,

$$\gamma_b = \sqrt{d}\,(\cos\theta_b\,\gamma_{\mathrm{sh},b} + \sin\theta_b\,\gamma_{\mathrm{ind},b})\,. \tag{37}$$

In particular, $\gamma_{\mathrm{ind},b}$ is negative while the individual part of b points to a wrong answer the model is confident about, and becomes positive once it points to the correct one.

The full dynamics. Putting everything together, we obtain

$$\dot{s} \simeq \frac{\alpha}{n_B}\sum_b \frac{\sqrt{d}}{\|h_b\|}\Big[\sin^2\theta_b\,\gamma_{\mathrm{sh},b} - \sin\theta_b\cos\theta_b\,\gamma_{\mathrm{ind},b}\Big], \tag{38}$$

where the only approximation is Equation 4. Without normalization, $\delta_b = W_2^{\top}\big(p_b - \mathbf{1}_{y_b}\big)$ and $\hat{u}_{\mathrm{sh}}^{\top}\delta_b = -\gamma_{\mathrm{sh},b}$, so

$$\dot{s} \simeq \alpha\,\bar{\gamma}_{\mathrm{sh}}, \quad \text{with } \bar{\gamma}_{\mathrm{sh}} = \frac{1}{n_B}\sum_b \gamma_{\mathrm{sh},b}. \tag{39}$$

The term in $\gamma_{\mathrm{ind},b}$ only exists because the normalization removes the radial component of the error, and it is the only term whose sign is controlled by how well the individual parts of the new facts are learned. Finally, both margins vanish when the new facts are learned: as $p_b \to \mathbf{1}_{y_b}$, $\gamma_{\mathrm{sh},b} \to 0$ and $\gamma_{\mathrm{ind},b} \to 0$. The force that writes the common shift and the force that withdraws it therefore both fade as the errors of the new facts vanish, with and without normalization.

B.6 Reduction under the common-coordinate approximation

Equation 38 depends on every new fact through θ_b, $\|h_b\|$, $\gamma_{\mathrm{sh},b}$ and $\gamma_{\mathrm{ind},b}$. To obtain a closed form, we use Assumption 4: all new states have the same coordinate s along $\hat{u}_{\mathrm{sh}}$ and the same individual norm σ. This is motivated by the fact that the shared part of every hidden state, $\sqrt{\alpha}\,W_1\mu$, is the same for old and new facts. Under this assumption,

$$\|h_b\| = \sqrt{s^2+\sigma^2}, \quad \cos\theta_b = \frac{s}{\sqrt{s^2+\sigma^2}}, \quad \sin\theta_b = \frac{\sigma}{\sqrt{s^2+\sigma^2}}, \tag{40}$$

so that

$$\frac{\sqrt{d}}{\|h_b\|}\sin^2\theta_b = \frac{\sqrt{d}\,\sigma^2}{(s^2+\sigma^2)^{3/2}} \quad \text{and} \quad \frac{\sqrt{d}}{\|h_b\|}\sin\theta_b\cos\theta_b = \frac{\sqrt{d}\,\sigma s}{(s^2+\sigma^2)^{3/2}}. \tag{41}$$

Plugging these into Equation 38 and averaging the margins over the new facts yields Equation 5,

$$\dot{s} = \alpha\sqrt{d}\,\frac{\sigma\,(\sigma\bar{\gamma}_{\mathrm{sh}} - s\bar{\gamma}_{\mathrm{ind}})}{(s^2+\sigma^2)^{3/2}}. \tag{42}$$

This one-dimensional equation has the following properties, which underlie the discussion of the main text:

- **The sign of $\dot{s}$ is the sign of $\sigma\bar{\gamma}_{\mathrm{sh}} - s\bar{\gamma}_{\mathrm{ind}}$.** If $\bar{\gamma}_{\mathrm{ind}} < 0$ and $\bar{\gamma}_{\mathrm{sh}} \geq 0$, then $\dot{s} > 0$: the shift can only grow and there is no fixed point.
- **The shift reinforces itself while it is small.** The weight of the $\bar{\gamma}_{\mathrm{ind}}$ term, $s\sigma/(s^2+\sigma^2)^{3/2}$, increases with s for $s < \sigma/\sqrt{2}$ and decreases beyond. While the new facts are confidently wrong ($\bar{\gamma}_{\mathrm{ind}} < 0$) and the shift is small, a larger shift therefore makes the shift grow faster.
- **Once the new facts are right, the shift is pulled back.** If $\bar{\gamma}_{\mathrm{ind}} > 0$, the equation has a unique fixed point $s^* = \sigma\bar{\gamma}_{\mathrm{sh}}/\bar{\gamma}_{\mathrm{ind}}$, which is stable since

$$\frac{\partial\dot{s}}{\partial s}\Big|_{s=s^*} = -\frac{\alpha\sqrt{d}\,\sigma\,\bar{\gamma}_{\mathrm{ind}}}{(s^{*2}+\sigma^2)^{3/2}} < 0. \tag{43}$$

  As $\bar{\gamma}_{\mathrm{sh}}$ quickly becomes small, s^* is small and s relaxes towards it.

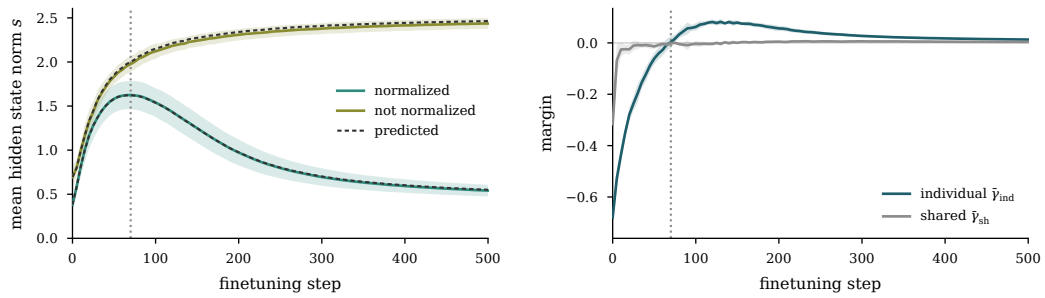


Figure 7: **Equation 38 reproduces the dynamics of the common shift.** (left) Length $\|\bar{h}_A\|$ of the old facts' mean hidden state, measured and obtained by integrating Equation 38 (with normalization) or $\dot{s} = \alpha\bar{\gamma}_{\text{sh}}$ (without normalization) from the point at which the mean state has aligned with the shared direction, after which its length follows the same dynamics as its displacement s. (right) Average shared margin $\bar{\gamma}_{\text{sh}}$ and individual margin $\bar{\gamma}_{\text{ind}}$ of the new facts, with normalization. The dotted line marks the peak; ten seeds.

- **The pull stops with the errors.** The restoring term is proportional to $\bar{\gamma}_{\text{ind}}$, which vanishes as the new facts are learned. Nothing in the equation brings s back to its initial value: it stops wherever it is when the errors of the new facts vanish.
- **Without normalization, nothing pulls the shift back.** The equation becomes $\dot{s} = \alpha\,\bar{\gamma}_{\text{sh}}$, which contains neither $\bar{\gamma}_{\text{ind}}$ nor s.

B.7 Empirical validation

We verify that Equation 38 describes the actual model, without the common-coordinate approximation. We follow the finetuning run, trained with minibatches of 32 new facts, and at every step measure θ_b, $\|h_b\|$, $\gamma_{\text{sh},b}$ and $\gamma_{\text{ind},b}$ on that step's minibatch. We evaluate the right-hand side of Equation 38 from these measurements and integrate it with an explicit Euler scheme whose time step is the learning rate η (Assumption 1), starting from the measured value of s once the mean state has aligned with the shared direction. The integration includes the second-order term of a finite step, which lengthens the mean state by $\|\Delta_\perp\|^2/2s$, where $\Delta_\perp$ is the part of the step orthogonal to $\hat{u}_{\text{sh}}$, and vanishes as $\eta \to 0$. We proceed in the same way for the model without normalization, with $\dot{s} = \alpha\,\bar{\gamma}_{\text{sh}}$. The prediction matches the measured s to within 1% of its range with normalization and 2% without (Figure 7).

B.8 The old facts survive the common shift

The common shift of the old states is written by the part of the update that acts on the shared key direction, $\Delta W_1\,\mu\mu^\top$ (Equation 16). At every checkpoint, we remove this part from the store, $W_1(t) \leftarrow W_1(0) + \Delta W_\perp(t)$ with $\Delta W_\perp = \Delta W_1(I - \mu\mu^\top)$, keep the readout as trained, and evaluate the edited model; training itself is not modified. At the trough, the edited model recalls the old facts with accuracy 1.00, against 0.17 for the trained model, and after about 1,000 steps with accuracy 0.91, against 0.55 (Figure 8). The remaining decline is the erosion carried by $\Delta W_\perp$ (Section B.9). The edit also slows the new facts, which reach 0.59 at step 200 instead of 0.99, since the shared part of the update also carries what the new facts have in common.

B.9 Fact-specific displacements

Size of the fact-specific displacements. Recall from Equation 18 that $\varepsilon_a = \sqrt{1-\alpha}\,\Delta W_1(e_a - \bar{e}_A)$. Under Assumption 3, ΔW_1 is fixed with respect to the old keys, and we can take the expectation over them. For a fixed matrix M and a vector u with $\mathbb{E}[uu^\top] = \Sigma$, $\mathbb{E}\|Mu\|^2 = \operatorname{tr}(M\Sigma M^\top)$. The e_a are independent and uniform on the unit sphere, so $\mathbb{E}[e_a e_a^\top] = I/d$ and

$$\mathbb{E}\left[(e_a - \bar{e}_A)(e_a - \bar{e}_A)^\top\right] = \left[\left(1 - \frac{1}{n_A}\right)^2 + \frac{n_A - 1}{n_A^2}\right]\frac{I}{d} = \left(1 - \frac{1}{n_A}\right)\frac{I}{d}, \tag{44}$$

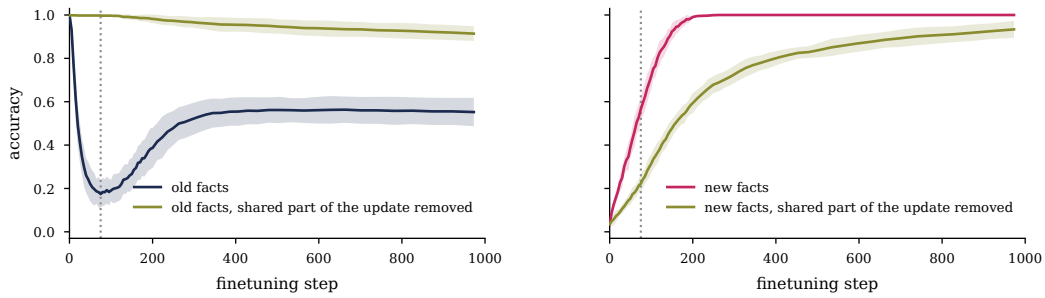


Figure 8: **Removing the shared part of the update restores the old facts.** Accuracy on the old (left) and new (right) facts as trained and with $\Delta W_1\,\mu\mu^\top$ removed from the store, the readout kept as trained. The dotted line marks the trough; ten seeds.

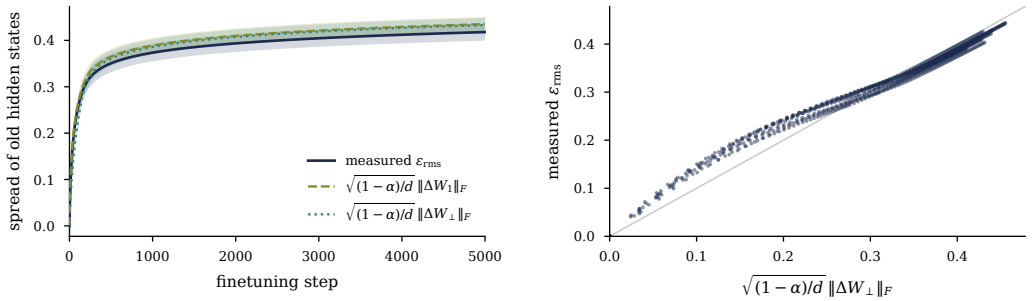


Figure 9: **The spread of the old facts is set by the Frobenius norm of the update.** (left) Measured spread ε_{rms} of the old hidden states during finetuning, compared with $\sqrt{(1-\alpha)/d}\,\|\Delta W_1\|_F$ and with $\sqrt{(1-\alpha)/d}\,\|\Delta W_\perp\|_F$ (Equation 47). (right) Measured against predicted; the prediction runs slightly low only while the shared column of the update is large. Ten seeds.

from which

$$\mathbb{E}\left[\varepsilon_{\text{rms}}^2\right] = \frac{1-\alpha}{d}\left(1-\frac{1}{n_A}\right)\|\Delta W_1\|_F^2. \tag{45}$$

Each old fact reads the update through its own random key, and a random direction captures, on average, a fraction $1/d$ of its squared Frobenius norm.

The shared and non-shared parts of the update. We split the update into its column along μ and the rest, $\Delta W_1 = \Delta W_\perp + (\Delta W_1\mu)\,\mu^\top$ with $\Delta W_\perp := \Delta W_1(I-\mu\mu^\top)$. The two parts are orthogonal for the Frobenius inner product, so $\|\Delta W_1\|_F^2 = \|\Delta W_\perp\|_F^2 + \|\Delta W_1\mu\|^2$, and, neglecting the $1/n_A$ correction,

$$\varepsilon_{\text{rms}}^2 \simeq \frac{1-\alpha}{d}\left(\|\Delta W_\perp\|_F^2 + \|\Delta W_1\mu\|^2\right). \tag{46}$$

The second term is the shared column of the update, read through the small random component $\mu^\top e_a$ of each old key. It only matters while the shared write makes up a large part of the update, that is, around the peak of the common shift. Dropping it gives

$$\varepsilon_{\text{rms}} \simeq \sqrt{\frac{1-\alpha}{d}}\,\|\Delta W_\perp\|_F. \tag{47}$$

Figure 9 compares the measured spread with both predictions.

Why the fact-specific displacements are not undone. Subtracting the mean write from Equation 21, the fact-specific displacements evolve as

$$\dot{\varepsilon}_a = -\frac{1}{n_B}\sum_b k_b^\top(k_a-\bar{k}_A)\,\delta_b = -\frac{\sqrt{1-\alpha}}{n_B}\sum_b k_b^\top(e_a-\bar{e}_A)\,\delta_b. \tag{48}$$

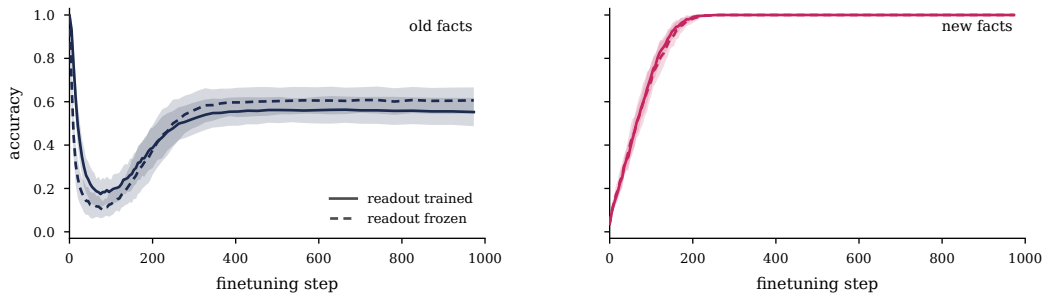


Figure 10: **Part of the bias towards the new answer region is stored in the readout.** Accuracy on the old and new facts during finetuning, with the readout trained and frozen; the learning rate of the frozen run is matched so that the new facts are learned at the same pace. Ten seeds.

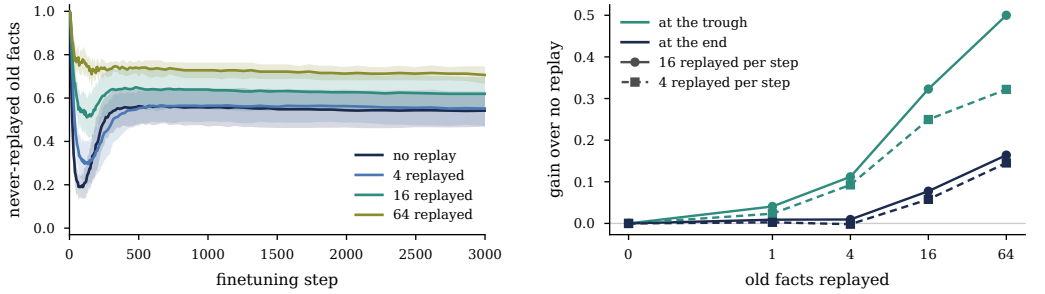


Figure 11: **Replaying a few old facts protects the others from the collapse, but not from erosion.** (left) Accuracy of the old facts that are never replayed, when 0, 4, 16 or 64 of the 128 old facts are replayed in every batch (16 replayed examples per step; five seeds). (right) Gain over no replay at the trough and at the end of finetuning (3,000 steps), with 16 or 4 replayed examples per step.

The coefficients are random, of order $d^{-1/2}$, and independent of the old fact's own state and answer. Unlike Equation 38, this expression contains no term that depends on ε_a itself and could pull it back. The contribution of the normalization to δ_b is present as well, but it enters with the same random coefficients. The fact-specific displacements therefore keep accumulating as long as the new facts produce errors.

B.10 Why recovery is incomplete

Two effects keep the old facts from recovering fully. The first follows from Section B.5: the force that withdraws the common shift is proportional to the margins of the new facts and vanishes with their errors. The second comes from the readout, which we have ignored so far.

Under the gradient flow dynamics, Equation 15 implies that, for a fixed normalized old state x_a, the logits change as

$$\dot{z}_a = \dot{W}_2 x_a = \frac{1}{n_B} \sum_b \left(x_b^\top x_a\right)\left(\mathbf{1}_{y_b} - p_b\right). \tag{49}$$

This has the same structure as Equation 21, with the overlap between normalized states in place of the overlap between keys. The normalized states share the direction of the common shift, so the overlaps $x_b^\top x_a$ have a common positive part. The readout therefore raises the logits of the new answers for every old fact at once, and lowers those of the answers the model currently predicts for the new facts. Unlike the write into W_1, this update has no normalization term: nothing in it depends on the margins of the new facts or on what the readout has already written. It stops when $p_b \to \mathbf{1}_{y_b}$ but never reverses, so the bias towards the new answer region that it stores remains after the new facts are learned. Freezing the readout during finetuning removes this channel and raises the recovered accuracy of the old facts from 0.56 to 0.61 (Figure 10).

B.11 INTERVENTIONS IN THE MINIMAL MODEL

Projecting out the shared direction. Removing the component of the update along the shared key direction at every training step, $\Delta W_1 \leftarrow \Delta W_1(I - \mu\mu^\top)$, prevents the collapse entirely, and the new facts are still learned. The old facts nevertheless end no better off than without the projection (accuracy 0.45 against 0.51 after 5,000 steps, ten seeds), because erosion lies in the fact-specific part of the update, which the projection leaves untouched.

Replaying a few old facts. Rehearsing a subset of old associations can restore the others (Hinton & Plaut, 1987). We replay a fixed random subset of the old facts in every finetuning batch of 32 new facts and measure the old facts that are never replayed (Figure 11). Replaying 16 of the 128 old facts lifts their trough from 0.17 to 0.50, as the replayed facts push back on the shift that all old facts share, but raises their final accuracy only from 0.54 to 0.62: erosion is specific to each fact and is barely affected by replaying others. A single replayed fact has almost no effect, as it is soon answered correctly through its own key and stops pushing back on the shared direction.

C TRANSFORMER

C.1 THE OLD FACTS KEEP THEIR RANKING WITHIN THEIR REGION

For each old fact of A, we rank its correct value among the values of its own half of the answers, and report the fraction of old facts for which it ranks first, alongside their accuracy over all answers (Figure 12). At the bottom of the collapse, only 12% of the old facts are recalled, yet 86% still rank their correct answer first within their half: the old facts lose to the answers of the other half, not to one another. The fraction then declines only slowly, to 83% after 500 steps, as the fact-specific drift erodes the old facts.

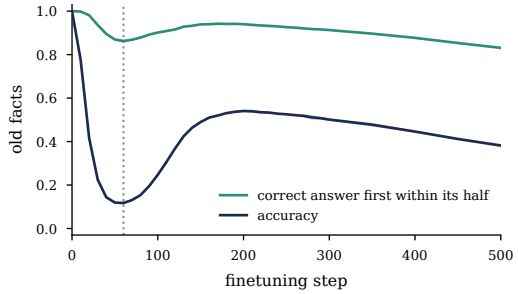


Figure 12: **The collapse leaves the old facts' ranking within their region intact.** Accuracy of the old facts of A over all answers, and fraction of them whose correct answer still ranks first among the answers of its own half, for the transformer of Figure 1 (right; three seeds). The dotted line marks the trough.

C.2 TIMING ACROSS LEARNING RATES AND BATCH SIZES

We finetune the transformer of Figure 1 under five conditions that change how fast it learns: the reference learning rate and batch size of 256, half and twice the learning rate, and batches of 128 and 512 at the reference learning rate. Each condition is run with three seeds (five for the reference), and both old and new facts are evaluated every 10 steps on held-out phrasings. We define the trough as the point of lowest old-fact accuracy before the recovery.

Across the five conditions, the trough moves from step 30 to step 100, a factor of 3.3 in training steps. At every trough, however, the new facts have reached the same stage of learning, between 38% and 45% accuracy (Figure 13). The recovery therefore starts at a fixed point in the learning of the new facts rather than after a fixed amount of optimization, as the minimal model predicts.

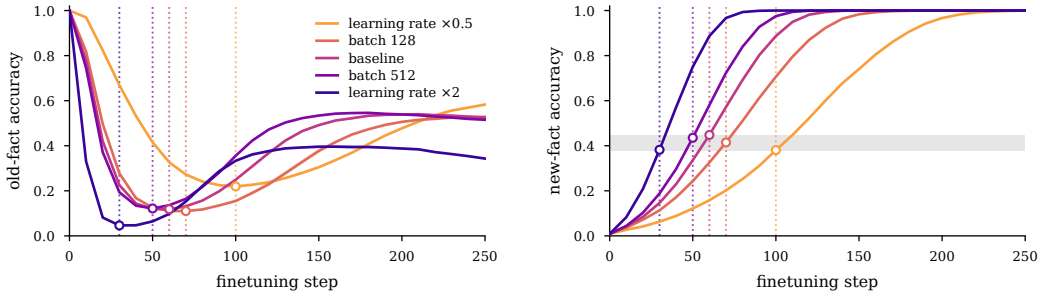


Figure 13: **The recovery is set by the learning of the new facts.** Old-fact (left) and new-fact (right) accuracy under five finetuning conditions; circles and dotted lines mark each condition's trough. The troughs span steps 30 to 100, yet the new facts are at 38–45% accuracy at every one of them (grey band).

C.3 Localizing the common shift

To find which weights carry the common shift, we take the checkpoint at the end of the collapse (step 50), restore one group of weights to its value before finetuning, and recompute the final hidden states of the old facts. We measure the common shift $\Delta\bar{h}$ of Section 4 with and without the restoration, and report the norm of the difference as a fraction of $\|\Delta\bar{h}\|$, averaged over the six attributes. The groups partition all parameters of the model, so their shares can be compared directly. Nothing is retrained, so the learning of the new facts is unaffected by construction.

The attention value and output weights carry 84% of the shift, and the query and key weights 11% (Table 1); after the recovery (step 200), the value and output weights still carry 85%. The unembedding is applied after the point at which the shift is measured and serves as a control: restoring it leaves the shift unchanged, as it must.

Weights restored	Share of the common shift
Attention value and output	84%
Attention query and key	11%
Input embedding	2%
LayerNorms	1%
Unembedding (control)	0%

Table 1: Share of the common shift removed by restoring each group of weights to its value before finetuning, at the trough (step 50; three seeds, variation below 0.3 percentage points).

C.4 Transformer with MLP blocks

We repeat the experiment of Figure 1 with a transformer that has MLP blocks of width $4d$ and is otherwise identical: same data, same size, same pretraining and finetuning. The same three phases appear (Figure 14). The old facts collapse faster and less deeply, to 0.53 at step 30, recover to 0.92 by step 100, and then erode, reaching 0.73 after 1,200 steps. Subtracting the common part of the change of the old facts' logits, as in Section 4, again removes the collapse: at the trough, old-fact accuracy rises from 0.53 to 0.99.

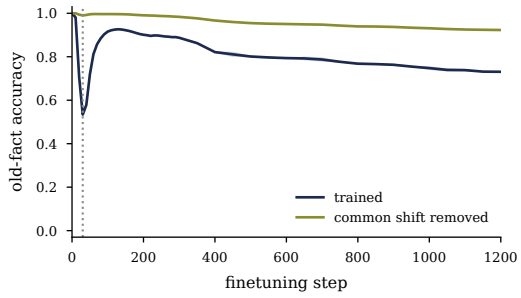


Figure 14: **The transformer with MLP blocks.** Old-fact accuracy of the trained model and with the common part of the change of their logits subtracted, as in Figure 4 (three seeds). The dotted line marks the trough.

D REMOVING ONE DIRECTION OF THE UPDATE

The edit. At each evaluation step of a finetuning run, we compute the update $\Delta W = W(t) - W(0)$ of every weight matrix of the transformer blocks of OLMo 2 1B: the query, key, value and output matrices of attention and the three matrices of the MLP, in each of the 16 layers, 112 matrices in total. The embedding and the unembedding are left untouched. We remove the top singular direction of each update, $W(t) \leftarrow W(t) - \sigma_1 u_1 v_1^\top$, evaluate the edited model, and then restore the original weights before training continues, so that the finetuning run itself is never modified. The removed direction is small: it holds about 15% of the squared norm of a typical update. As a control, we instead remove a random rank-one matrix of the same norm from each update.

Effect on the old facts. Removing the top direction keeps the old facts close to their accuracy before finetuning through the collapse, for both kinds of new facts, while the random control leaves them exactly where training left them (Table 2). The edit also keeps the loss on a fixed sample of ordinary text (64 sequences of 512 tokens) close to its value before finetuning, whereas finetuning alone raises it.

The edit is not a return to an earlier point of training. The edit lowers the accuracy on the new facts, which raises the question of whether it simply moves the model back along its training trajectory. To test this, we compare the edited model with the trained model at the same accuracy on the new facts: for each edited checkpoint, we find the step at which training first reached the same new-fact accuracy, interpolating between evaluations, and read the old-fact accuracy there. For real entities, at new-fact accuracies of 21%, 34% and 66%, the edited model recalls 88%, 88% and 79% of the old facts, against 81%, 73% and 72% for the trained model: no point of the training run combines the edited model's old-fact and new-fact accuracies. The reduction in new-fact accuracy indicates that the removed direction also carries part of what the new facts share.

		Old facts			New facts	
New facts	Step	trained	top removed	random removed	trained	top removed
Synthetic individuals	80	0.28	0.77	0.28	0.03	0.01
	400	0.38	0.70	0.38	0.90	0.40
Real entities	80	0.73	0.88	0.73	0.46	0.25
	400	0.55	0.79	0.55	1.00	0.66

Table 2: Accuracy of OLMo 2 1B at the trough (step 80) and after 400 steps of finetuning, with the top or a random direction of each weight update removed. Before finetuning, the old facts are recalled with accuracy 0.87. Small finetuning set for the synthetic individuals; three seeds.