IMPLICIT EVOCATION IN UNLEARNING-BASED ALIGNMENT: EXPLORATION AND EXPLANATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Language-model unlearning is commonly evaluated at the output, leaving open whether reduced target answering is accompanied by representational erasure. We study this question on RWKU using two instruction-tuned, approximately 4B-parameter models, three unlearning objectives (GA, NPO, and ATU), and both full-parameter and LoRA updates. Applying the model's shared unembedding as a raw logit lens, we find that target-answer subtokens can remain more highly ranked than task-irrelevant baseline subtokens at intermediate layers even when their final-layer ranks decrease. We call this tested pattern *implicit evocation*. The observation establishes residual linear decodability under an oracle target-subtoken readout; it does not by itself establish attack-level recoverability or universality across unlearning methods. Under an idealized residual-block model and explicit boundedness and contractivity assumptions, we also derive conditions that favor stronger late-layer updates. This conditional analysis is consistent with late-layer suppression but does not fully explain the observed non-monotone mid-layer peak.

1 INTRODUCTION

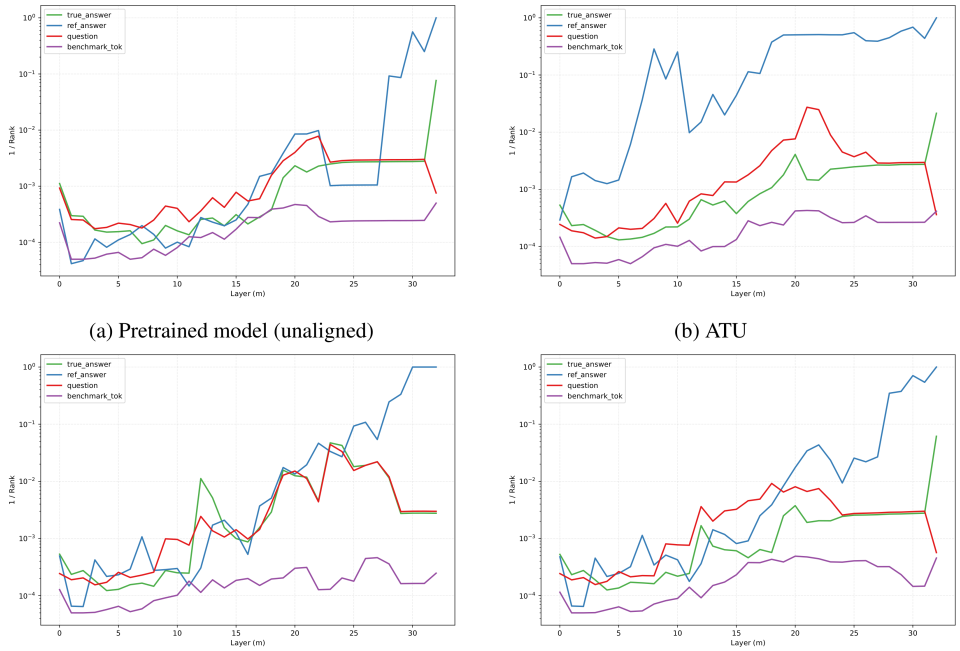
Machine unlearning seeks to reduce a model's dependence on specified forget data while preserving unrelated capabilities (Zhang et al., 2024; Yao et al., 2024; Song et al., 2025). Recent work has also examined detectable traces of unlearning interventions and copyright-aligned generation (Chen et al., 2025; Sharma et al., 2025). Most evaluations measure generated answers or final-layer probabilities, but successful output suppression need not imply that target information is absent from intermediate representations. We study this distinction rather than treating output-level forgetting as evidence of complete representational erasure.

We apply the shared unembedding of each model directly to its intermediate hidden states—a raw logit-lens readout rather than a separately trained probe—and inspect the ranks of target-answer subtokens. In the tested RWKU settings, target subtokens may remain elevated relative to task-irrelevant baselines at intermediate layers even when their ranks fall near the output. We call this pattern *implicit evocation*; Figure 1 gives an illustrative example. The term is an operational description of residual linear decodability, not a claim that the target answer is recoverable by an attacker.

Our empirical scope comprises two instruction-tuned models (Phi-3 and Qwen3), three objectives (GA, NPO, and ATU), full-parameter and LoRA updates, and the factual-unlearning benchmark RWKU. Our theory studies an idealized residual block in which a trainable linear map acts on a frozen contextual feature map. Under explicit boundedness, non-degeneracy, and contraction assumptions, the analysis characterizes conditions that favor stronger late-layer updates. It is a conditional account of late-layer suppression, not a verified causal model of the evaluated checkpoints or a derivation of the full mid-layer peak.

Positioning. Our evaluated objectives cover gradient ascent, NPO (Zhang et al., 2024), and embedding-space ATU (Spohn et al., 2025). They do not cover recent alternatives such as reference-free SimNPO (Fan et al., 2024) or representation-targeting RMU (Li et al., 2024). Representation Rerouting/Circuit Breakers (Zou et al., 2024) and RepNoise (Rosati et al., 2024) are closely related

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



(a) Pretrained model (unaligned)

(b) ATU

(c) GA

(d) NPO

Figure 1: Illustrative logit-lens trajectories for the prompt "In which city in Maine was Stephen King born?" and target answer "Portland." Panel (a) is the pre-unlearning checkpoint; panels (b)–(d) are obtained with ATU, GA, and NPO. The embedded legend names correspond to forgotten-answer subtokens (`true_answer`, green), model-output subtokens (`ref_answer`, blue), question-entity subtokens (`question`, red), and task-irrelevant baseline subtokens (`benchmark_tok`, purple). The vertical axis is reciprocal rank, not attention or probability. In the unlearned checkpoints, the green trajectory falls near the output but remains above the purple baseline at several intermediate layers. This example illustrates residual linear decodability; it is not an extraction attack or a statistical significance test.

representation-level safety interventions, but address different threat models. Methodologically, our readout is the raw logit lens; the Tuned Lens (Belrose et al., 2023) instead learns layer-specific affine translators and is designed to reduce intermediate-state calibration artifacts. These neighboring methods motivate important future comparisons and bound our conclusions to the settings actually evaluated here.

Statement of Contribution. In summary, the contributions of this paper are listed as follows.

- We document residual mid-layer target-subtoken decodability across the two models, three objectives, and two update regimes evaluated on RWKU, while explicitly separating this diagnostic evidence from attack-level recoverability.
- We provide a conditional residual-block analysis that identifies sufficient assumptions for a depth bias in update intensity, and delineate where those assumptions diverge from the evaluated models and the observed non-monotone peak.

Section 2 introduces the notation and diagnostic readout. Section 3 gives the conditional analysis, and Section 4 reports the tested settings. Additional discussion, derivations, and implementation details appear in the appendix.

2 PRELIMINARIES

This section introduces the LLM notation, raw logit-lens diagnostic, and idealized unlearning update used in our analysis.

LLMs. Let $\mathcal{V}$ be the set of all possible tokens (i.e., the vocabulary set). Given a sequence of tokens $x \in \mathcal{V}^L$ with length $L \in \mathbb{N}_+$, the goal of a causal LLM is to predict the next token (or label) of x, denoted as $y \in \mathcal{V}$. A typical causal LLM (e.g., Qwen (Yang et al., 2025) and DeepSeek (DeepSeek-AI, 2024)) can be decoupled into a feature extractor h_θ and an LM head (i.e., a language model head, also known as a prediction head) λ, where θ denotes the trainable parameters of h. Let M be the depth of the LLMs. Then, the feature extractor can be further decomposed into a stack of M transformer blocks, formulated as

$$h_\theta(x) := h_\theta^{(M)} \circ h_\theta^{(M-1)} \circ \cdots \circ h_\theta^{(2)} \circ h_\theta^{(1)}(x). \tag{1}$$

The output distribution of the model is given by

$$p_\theta(y = v_i \mid x) = \frac{\exp\big((\lambda \circ h_\theta(x))^{(i)}\big)}{\sum_{j\in[|\mathcal{V}|]} \exp\big((\lambda \circ h_\theta(x))^{(j)}\big)}, \tag{2}$$

and the output token v is given by $v = \arg\max_{v_i \in \mathcal{V}} p_\theta(y = v_i \mid x)$. For the analysis only, we use the following idealized residual-block model inspired by residual-stream and Transformer knowledge-representation studies (Nanda et al., 2023; Geva et al., 2021). It is not an exact architectural specification of Phi-3 or Qwen3. For any $m \in [M]$, let

$$h_\theta^{(m)}(u) = u + W_\theta^{(m)} \operatorname{Attn}_\theta^{(m)}(u), \qquad u = h_\theta^{(m-1,1)}(x), \tag{3}$$

where $W_\theta^{(m)}$ is a trainable linear map and $\operatorname{Attn}_\theta^{(m)}$ is a contextual feature map. To keep the Jacobians tractable, the model folds a mean-centered layer-normalization operation into the feature map. The evaluated checkpoints instead use RMSNorm and full-parameter updates may also change attention; consequently, the following model is an idealization rather than a literal description of their training dynamics.

Definition 2.1 (Layer Normalization and Attention Module)**.** *Given input data $x \in \mathbb{R}^{d_h}$, for any $m \in [M]$, let*

$$\operatorname{Attn}_\theta^{(m)}(u) = V^{(m)\top} \operatorname{softmax}\Big(\tau_m K^{(m)\top} W_Q^{(m)} \operatorname{LN}_{\gamma_m,\beta_m}(u)\Big), \tag{4}$$

where u denotes the input hidden state, $W_Q^{(m)}$ represents the query projection matrix, $K^{(m)}$ and $V^{(m)}$ are the key and value matrices respectively, and $\tau_m > 0$ is the scaling (temperature) parameter. The function $\operatorname{LN}_{\gamma_m,\beta_m}(\cdot)$ is the layer normalization parameterized by γ_m and β_m, defined as

$$\operatorname{LN}_{\gamma_m,\beta_m}(x) = \gamma_m \odot \frac{x - \mu(x)\mathbf{1}}{\sqrt{\sigma^2(x) + \varepsilon}} + \beta_m. \tag{5}$$

In Equation (5), $\mu(x)$ and $\sigma^2(x)$ denote the mean and variance across the dimensional components of x, respectively; $\mathbf{1}$ is an all-ones vector, and $\varepsilon > 0$ is a small constant to avoid division by zero. The vectors $\gamma_m, \beta_m \in \mathbb{R}^{d_h}$ represent learnable scale and shift parameters, respectively, and the operator $\odot$ denotes element-wise multiplication (i.e., the Hadamard product).

Unlearning and Raw Logit-Lens Probing. Let θ_0 be the parameters of a pre-unlearning checkpoint and let $\mathcal{D}_{\mathrm{FG}}$ be a forget dataset of QA pairs (x, y). Unlearning objectives differ—for example, GA increases a target loss, NPO reweights a negative-preference objective, and ATU operates in an embedding space—but are evaluated by reduced target answering together with retained utility. Such end-to-end criteria do not inspect intermediate states. To obtain a descriptive internal readout, for all $1 \le m \le M$ let

$$h_\theta^{(m,1)}(x) := h_\theta^{(m)} \circ h_\theta^{(m-1)} \circ \cdots \circ h_\theta^{(2)} \circ h_\theta^{(1)}(x) \tag{6}$$

be the hidden states after the m-th layer, and

$$p_\theta^{(m,1)}(y = v_i \mid x) = \frac{\exp\Big((\lambda \circ h_\theta^{(m,1)}(x))^{(i)}\Big)}{\sum_{j\in[|\mathcal{V}|]} \exp\Big((\lambda \circ h_\theta^{(m,1)}(x))^{(j)}\Big)} \tag{7}$$

be the m-th raw logit-lens distribution, where $\lambda : \mathbb{R}^{d_h} \to \mathbb{R}^{d_V}$ is the model's shared LM head applied directly to the extracted hidden state. No layer-specific affine translator or independently trained probe is used. The final output distribution p_θ is obtained by replacing $h_\theta^{(m,1)}$ in Equation (7) with the model's final hidden state. For a multi-token answer sequence y_0, output-level suppression means that the decoded response to x_0 omits that answer under the benchmark's answer-matching rule:

$$\mathbb{1}\{y_0 \text{ appears in } \mathrm{Decode}_\theta(x_0)\} = 0. \tag{8}$$

Under the prevalent setting of weight tying between word embeddings and the LM head, we define

$$(\lambda \circ h_\theta(x))^{(i)} = v_i^\top h_\theta(x), \qquad \forall i \in [d_V]. \tag{9}$$

Specifically, the remainder of this paper abbreviates $p_\theta(y = v_i \mid x)$ as p_{v_i} or p_i. The approach of projecting intermediate hidden states through the shared unembedding is commonly called the *logit lens*; we use "logit-lens probing" to emphasize its diagnostic role. Because this raw readout is not calibrated for intermediate states, rank trajectories should be interpreted as evidence of linear decodability, not as generated predictions (Belrose et al., 2023).

Finally, we specify the gradient-ascent update studied by the analysis. Consider the cross-entropy loss

$$\ell((x, y); \theta) = -\log(p_\theta(y \mid x)). \tag{10}$$

Let N denote the number of samples in $\mathcal{D}_{\mathrm{FG}}$, then the corresponding empirical risk is given by

$$R(\theta, \mathcal{D}_{\mathrm{FG}}) = \frac{1}{N} \sum_{(x,y)\in\mathcal{D}_{\mathrm{FG}}} \ell((x, y); \theta). \tag{11}$$

The analysis below studies the iterative process of $W_{\theta\{t\}}^{(m)}$ under the gradient-ascent special case. It therefore provides a stylized account rather than a theorem about every evaluated objective. For any $m \in [M]$, the iteration is

$$W_{\theta\{t+1\}}^{(m)} = W_{\theta\{t\}}^{(m)} + \eta \nabla_W R(\theta, \mathcal{D}_{\mathrm{FG}}) = W_{\theta\{t\}}^{(m)} + \frac{\eta}{N} \sum_{(x,y)\in\mathcal{D}_{\mathrm{FG}}} \nabla_W \ell((x, y); \theta), \tag{12}$$

where ∇_W denotes the gradient operator with respect to $W_{\theta\{t\}}^{(m)}$. For ease of notation, we omit the superscripts and subscripts of W in the remainder of this paper.

3 Theoretical Analysis

This section separates the empirical observation from a conditional theoretical account. Section 3.1 defines the operational pattern measured by our raw logit lens. Section 3.2 then studies sufficient conditions for a depth bias in an idealized residual-block model. These results are not claimed to verify the training dynamics of the evaluated checkpoints.

3.1 Implicit Evocation

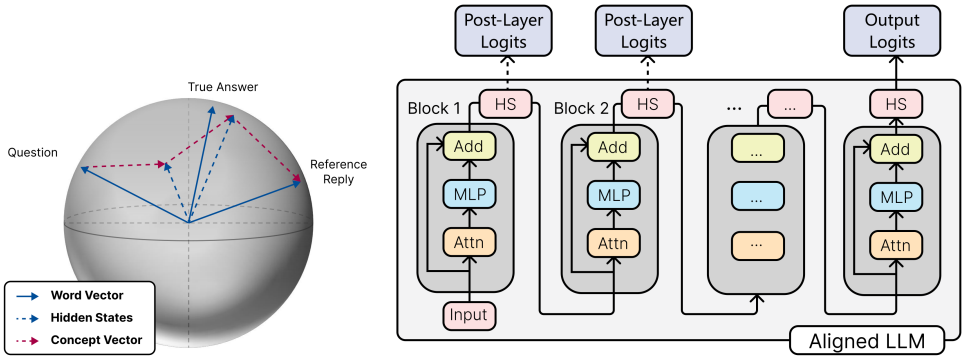
By combining Equations (3) and (6) and Definition 2.1, we can obtain that

$$\Delta h_\theta^{(m)}(x) := h_\theta^{(m,1)}(x) - h_\theta^{(m-1,1)}(x) = W_\theta^{(m)} \, \mathrm{Attn}_\theta^{(m)}\left(h_\theta^{(m-1,1)}(x)\right), \qquad \forall m \in [M]. \tag{13}$$

Inspired by Park et al. (2025) and Bu et al. (2025), we let the concept vector of the m-th layer be the difference of the layer-wise hidden state $\Delta h_\theta^{(m)}(x)$ defined in Equation (13). Correspondingly, the hidden state $h_\theta^{(m,1)}(x)$ can be formulated as

$$h_\theta^{(m,1)}(x) = x + \sum_{i=1}^{m} \Delta h_\theta^{(i)}(x). \tag{14}$$

Figure 2 provides an intuitive visualization of the distinctions and relationships among concept vectors, word embeddings, and hidden states during the model's forward pass. Through the lens of the residual stream framework, concept vectors correspond to the "semantic information" incrementally injected into the word embeddings across layers by the neural network, thereby possessing the capacity to steer the model's output. Note that the residual stream framework exhibits structural parallels with static word embedding models (e.g., Word2Vec (Mikolov et al., 2013)); a detailed comparative discussion is deferred to Appendix B.3.



(a) Residual Stream (b) Raw Logit Lens

Figure 2: Residual-stream intuition (left) and the raw logit-lens readout (right). **Left:** Residual increments can change the final output while earlier readouts remain less affected; the drawing is conceptual rather than a faithful diagram of Phi-3 or Qwen3 training. **Right:** The shared LM head is applied to hidden states after successive blocks. We call each residual increment a "concept vector" only as descriptive shorthand and do not assume that it isolates a single semantic concept.

Assumption 3.1. *We impose the following assumptions on the attention module:*

1. *For tractability, we treat* $\mathrm{Attn}_\theta^{(m)}$ *as a frozen contextual feature map and analyze updates only to* $W_{\theta\{t\}}^{(m)}$*. This is a modeling assumption, not a claim that factual knowledge resides only in MLPs or that attention is frozen in our full-parameter experiments.*

2. *We postulate the boundedness of the attention module. Specifically, for any* $m \in [M]$*, we assume that* $W_Q^{(m)}$*,* $K^{(m)}$*, and* $V^{(m)}$ *satisfy*

$$\|W_Q^{(m)}\|_2 \le B_Q^{(m)}, \qquad \|K^{(m)}\|_2 \le B_K^{(m)}, \qquad \|V^{(m)}\|_2 \le B_V^{(m)}, \tag{15}$$

   where $B_Q^{(m)}$*,* $B_K^{(m)}$*, and* $B_V^{(m)}$ *are constants.*

3. *We similarly postulate that the layer normalization operation,* $\mathrm{LN}_{\gamma_m,\beta_m}(\cdot)$*, satisfies*

$$\|\gamma_m\|_\infty \le G_m, \qquad \sigma^2(u) \ge \sigma_m^2 > 0, \tag{16}$$

   for every hidden state u *in the local trajectory analyzed below. Furthermore, the aforementioned constants satisfy*

$$\frac{\tau_m G_m B_V^{(m)} B_K^{(m)} B_Q^{(m)}}{\sqrt{\sigma_m^2}} \le 1. \tag{17}$$

Assumption 3.1 supplies boundedness conditions for the idealized feature map. It neither establishes the contraction condition used below nor verifies that Phi-3 or Qwen3 satisfies these constants. Moreover, their RMSNorm operations differ from the mean-centered LayerNorm Jacobian analyzed here.

For a token v ranked n-th by the layer-m raw logit lens, let $r_\theta^{(m,1)}(v) = 1/n$. For a multi-token answer y, let $\mathcal{T}(y)$ be the deterministic candidate-subtoken set described in Appendix D, and define the recall-oriented score $s_m(y \mid x) = \max_{v \in \mathcal{T}(y)} r_\theta^{(m,1)}(v \mid x)$. This maximum can favor larger candidate sets, so we report it as a diagnostic rather than a calibrated probability or whole-answer recovery score.

Definition 3.1 (Operational Implicit Evocation)**.** *Given a target QA pair* (x_0, y_0) *for which output-level evaluation indicates suppression, we use* implicit evocation *to describe a layer range in which* $s_m(y_0 \mid x_0)$ *remains elevated relative to an explicitly declared task-irrelevant baseline score while decreasing toward the output. The definition describes residual linear decodability under an oracle target-subtoken readout; it does not label the pattern anomalous relative to a never-known fact or establish that the answer can be recovered without oracle knowledge.*

We inspect this operational pattern for GA, NPO, and ATU on two models. Figures 3 and 4 show the evaluated full-parameter and LoRA settings. Because we do not include matched never-known controls, representation-targeting baselines, or a recovery attack, the result should not be generalized to unlearning as a whole or described as a demonstrated security vulnerability.

3.2 Main Results

We now identify sufficient conditions under which a sample can induce different update magnitudes across layers and backpropagated gradients can contract toward earlier layers. The results are conditional bounds in the idealized model. They support a possible depth bias but do not derive the empirical mid-stack maximum. Derivations and the additional assumptions are given in Appendix C.

Heterogeneity in weight update intensity. Intuitively, the impact of the same sample (x, y) on trainable parameters situated at different layers may vary. Now, we formalize this heterogeneity in weight update intensity and discuss its influence on implicit evocation. Specifically, for any $m \in [M]$, we define the influence intensity of the sample (x, y) on $W_\theta^{(m)}$ as

$$I_m(x, y) := \left\| \nabla_{W_\theta^{(m)}} \ell((x, y); \theta) \right\|_F . \tag{18}$$

The quantity may differ across layers. Relating that difference to retained target semantics requires additional assumptions stated in Appendix C; it is not implied by the definition alone.

Theorem 3.1 (Conditional Influence Intensity Ratio Bound)**.** *Under Assumption C.1, for fixed $m_1 < m_2 \le M$ and sample (x, y), we have*

$$\frac{I_{m_1}(x, y)}{I_{m_2}(x, y)} \le \left(\prod_{m'=m_1+1}^{m_2} \alpha_{m'} \right) \frac{\tau_{m_1}^w}{\sigma_{m_1} \tau_{m_2}^c}, \tag{19}$$

where $I_m(x, y)$ is defined as the influence intensity (cf. Equation (18)).

The proof is deferred to Appendix C. Theorem 3.1 bounds I_{m_1}/I_{m_2} only under Assumption C.1, including an explicit lower singular-value bound and local contraction bounds. It therefore identifies sufficient conditions for weaker updates at an earlier layer; it does not establish that these conditions hold in the evaluated checkpoints or that weaker updates cause the observed rank curves.

Gradient decay mechanism in backpropagation. The next theorem makes the contraction requirement explicit. Boundedness of J_c^m alone is insufficient: $\|J_c^m\|_2 < 1$ does not imply $\|I + W_\theta^{(m)} J_c^m\|_2 < 1$.

Theorem 3.2 (Conditional Gradient Decay)**.** *Under the conditions specified by Proposition C.1, if there exists $\alpha_m \in (0, 1)$ such that for all training iterations and samples,*

$$\left\| I + W_\theta^{(m)} J_c^m(x) \right\|_2 \le \alpha_m < 1, \tag{20}$$

then the backpropagated gradients exhibit a strict decay, i.e., for $m \in \{1, \dots, M-1\}$,

$$\|g^{(m-1)}\|_2 \le \alpha_m \|g^{(m)}\|_2 < \|g^{(m)}\|_2, \tag{21}$$

and

$$\|g^{(M-1)}\|_2 \le \alpha_M \|\nabla_{h^{(M,1)}} \ell((x, y); \theta)\|_2 < \|\nabla_{h^{(M,1)}} \ell((x, y); \theta)\|_2. \tag{22}$$

Moreover, for any $0 \le m_1 < m_2 \le M - 1$,

$$\|g^{(m_1)}\|_2 \le \left(\prod_{m'=m_1+1}^{m_2} \alpha_{m'} \right) \|g^{(m_2)}\|_2. \tag{23}$$

Furthermore, for any $0 \le m_1 \le M - 1$,

$$\|g^{(m_1)}\|_2 \le \left(\prod_{m'=m_1+1}^{M} \alpha_{m'} \right) \|\nabla_{h^{(M,1)}} \ell((x, y); \theta)\|_2. \tag{24}$$

Here $g^{(m)} := \nabla_{h^{(m,1)}} \ell((x, y); \theta)$ denotes the gradient with respect to the m-th hidden state and $J_c^m := \partial \operatorname{Attn}_\theta^{(m)} / \partial h_\theta^{(m-1,1)} = \partial \operatorname{Attn}_\theta^{(m)}(x) / \partial h_\theta^{(m-1,1)}(x)$ (cf. Proposition C.1).

Theorem 3.2 proves multiplicative decay *conditional on* the displayed operator-norm inequality holding for every relevant sample and training iteration. We do not measure these operators in Phi-3 or Qwen3. Even if the condition holds, the theorem predicts a monotone depth bias and therefore does not explain why the empirical curves peak specifically in the middle rather than increasing monotonically toward earlier layers.

Summary. The analysis thus supplies a conditional explanation for how later layers can receive larger updates than earlier ones in a stylized model. It should be read as a hypothesis-generating account, not a causal validation of implicit evocation. Measuring the relevant operators and reconciling the monotone bound with the non-monotone empirical peak remain open validation steps.

4 Experiments and Analysis

This section reports the scope of our empirical observation. Section 4.1 introduces the dataset, checkpoints, evaluated objectives, and output-level results; Section 4.2 describes the raw logit-lens diagnostic. Implementation details are included in Appendix D.

4.1 Experimental Setup

We apply each unlearning objective to a pre-unlearning instruction-tuned checkpoint and then evaluate the resulting checkpoint with output metrics and the layer-wise diagnostic.

Dataset. We use RWKU (Real-World Knowledge Unlearning)[1], a factual-unlearning benchmark containing target entities such as Stephen King, Confucius, and Bruce Lee. RWKU provides fill-in-the-blank, question-answering, and adversarial evaluations, together with neighbor and general-utility tasks. Our paper does not report a successful Level-3 recovery attack; the layer-wise readout is an oracle diagnostic with the target subtokens specified in advance.

Models. We consider two instruction-tuned checkpoints: Phi-3-mini-4k-instruct[2] (3.8B parameters; Phi-3) and Qwen3-4B-Instruct-2507[3] (4B parameters; Qwen3). Their hidden dimensions are 3072 and 2560, with 32 and 36 layers, respectively. We refer to the checkpoints before application of GA, NPO, or ATU as *pre-unlearning*, rather than pretrained or unaligned, models.

Baseline Methods. We evaluate (1) Gradient Ascent (GA) (Yao et al., 2024), which increases the loss on target samples; (2) Negative Preference Optimization (NPO) (Zhang et al., 2024), which regularizes negative-preference updates relative to a reference model; and (3) Align-then-Unlearn (ATU) (Spohn et al., 2025), which reduces similarity to target concept embeddings. These objectives were selected as representative methods available in the RWKU-oriented pipeline, not as an exhaustive sample of modern unlearning. Hyperparameters are reported in Appendix D.

Evaluation Metrics. We report RWKU fill-in-the-blank (FB), question answering (QA), answer-all (AA), and aggregate (All) metrics, for which lower values indicate less target answering. Retain-side utility is summarized with MMLU, BBH, TruthfulQA, TriviaQA, and fluency, for which higher values are better. These output metrics and the internal readout answer different questions and are not treated as interchangeable evidence.

Output-level behavior and utility. Table 1 shows that no single objective dominates across all forget and utility metrics. The relative ordering changes with the task, model, and update regime, and a lower value on one forget metric need not coincide with the strongest retained utility. Because corresponding pre-unlearning values are unavailable for the forget-side rows, these results characterize differences among the evaluated updates rather than establishing improvement over the original checkpoint on every metric. They provide the output-level context for the representation-level diagnostic below.

[1] https://huggingface.co/datasets/jinzhuoran/RWKU

[2] https://huggingface.co/microsoft/Phi-3-mini-4k-instruct

[3] https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507

Pre-trained Model	Phi-3				Qwen-3			
Metric	**LLM**	**GA**	**NPO**	**ATU**	**LLM**	**GA**	**NPO**	**ATU**
Unlearning Performance ↓								
Fill-in-the-blank (Full)	-	71.4	46.4	**45.8**	-	68.9	**43.7**	44.5
QA (Full)	-	**19.8**	48.8	53.0	-	18.6	45.9	**18.1**
Answer All (Full)	-	**54.9**	58.3	59.1	-	**53.8**	57.1	58.4
All Tasks (Full)	-	51.3	53.3	**50.9**	-	**49.6**	51.8	56.2
Fill-in-the-blank (LoRA)	-	72.0	**46.9**	47.6	-	69.3	**44.1**	45.0
QA (LoRA)	-	**19.6**	49.1	53.4	-	18.8	46.2	**18.4**
Answer All (LoRA)	-	**55.2**	58.2	59.4	-	54.1	57.4	**53.7**
All Tasks (LoRA)	-	**51.7**	53.6	58.3	-	**49.9**	52.1	56.6
Utility Performance ↑								
MMLU (Full)	**70.2**	67.8	69.0	69.0	**73.8**	71.6	73.1	72.7
BBH (Full)	**42.1**	37.0	39.5	40.7	**46.0**	42.8	45.2	44.6
TruthfulQA (Full)	**36.8**	36.0	36.0	36.0	**39.6**	38.4	39.1	38.8
TriviaQA (Full)	**15.9**	14.4	15.5	15.3	**18.1**	16.8	17.6	17.3
Fluency (Full)	**718.0**	715.2	706.2	709.1	**726.8**	724.3	718.6	721.5
MMLU (LoRA)	-	67.8	**69.0**	**69.0**	-	71.1	**72.6**	72.2
BBH (LoRA)	-	37.0	39.5	**40.7**	-	42.1	**44.7**	44.0
TruthfulQA (LoRA)	-	**36.0**	**36.0**	**36.0**	-	37.9	**38.6**	38.4
TriviaQA (LoRA)	-	14.4	**15.5**	15.3	-	16.5	**17.2**	17.0
Fluency (LoRA)	-	**715.2**	706.2	709.1	-	**723.1**	717.4	720.2

Table 1: RWKU output and utility metrics for Phi-3 and Qwen3 under GA/NPO/ATU with full-parameter or LoRA updates. The `LLM` columns denote the pre-unlearning checkpoints. Lower values are better for the unlearning rows and higher values are better for utility. Best and second-best results are bold and underlined. The pre-unlearning checkpoint is reported only for utility, so the table does not by itself quantify improvement from that checkpoint on every forget metric.

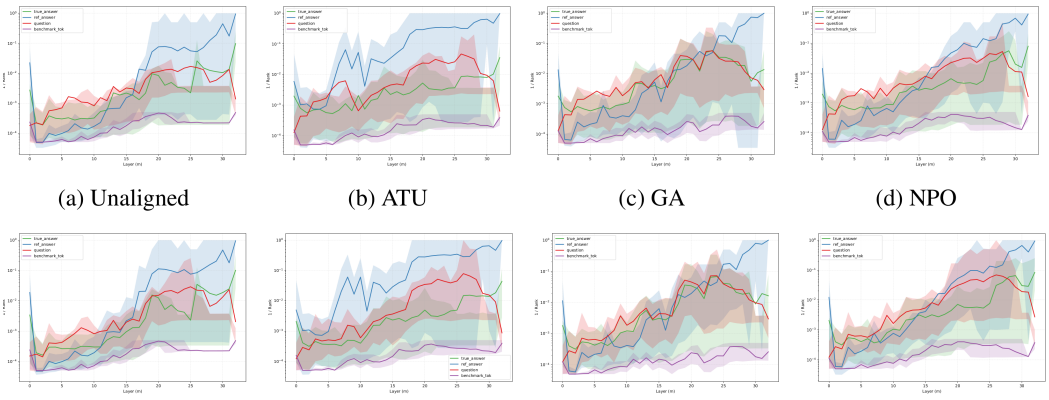
4.2 Raw Logit-Lens Probing: Results and Discussion

We apply the model's shared LM head directly to the hidden state at the query position after each layer and measure best-subtoken reciprocal rank. No independent probe or tuned lens is trained. Consequently, the result measures accessibility to this fixed linear readout, not whether a non-oracle procedure can recover the answer.

For a keyword or multi-token answer, deterministic surface-form rules define a candidate-subtoken set $\mathcal{T}(y)$, and the candidate sets for the four plotted tracks are required to be pairwise disjoint. We report $\max_{v\in\mathcal{T}(y)} r_\theta^{(m,1)}(v \mid x)$. This recall-oriented best-subtoken statistic does not show that the complete answer is represented as a single unit, and larger or more permissive candidate sets can increase the score. We therefore interpret it only as an oracle diagnostic and do not claim robustness to mean-token or first-token aggregation, which we have not evaluated.

Results. Figures 3 and 4 present full-parameter and LoRA results for the pre-unlearning checkpoint and GA/NPO/ATU. Solid lines are dataset means and shaded regions span the 5th–95th percentiles.

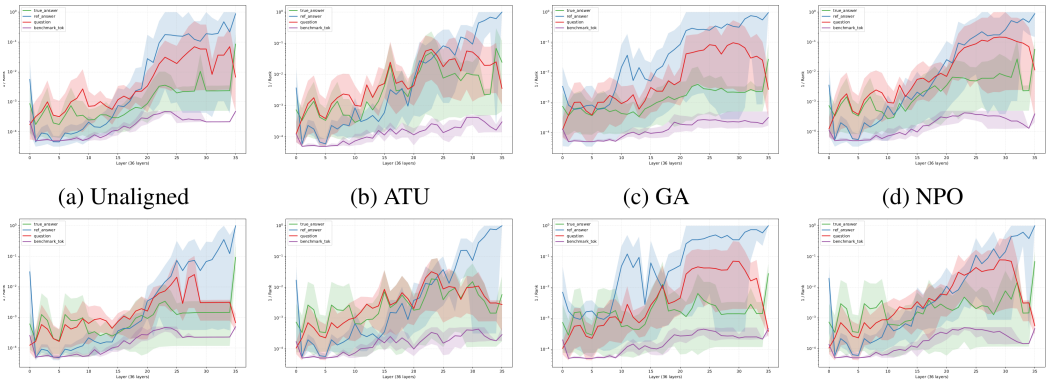
Discussion. Across the tested unlearning objectives and update regimes, the forgotten-answer curve (green) is elevated above the task-irrelevant baseline (purple) over a range of intermediate layers and then falls toward the output. This is the empirical content of our claim: target-answer subtokens remain accessible to the fixed unembedding at intermediate layers while becoming less favored at the output. The plots do not establish that the pattern is anomalous relative to a matched never-known fact, that it survives a calibrated probe, or that an attacker can recover the answer without oracle target knowledge. The conclusion is therefore restricted to residual linear decodability under the stated diagnostic and plotting baselines.



(a) Unaligned (b) ATU (c) GA (d) NPO

(e) Unaligned (LoRA) (f) ATU (LoRA) (g) GA (LoRA) (h) NPO (LoRA)

Figure 3: Raw logit-lens trajectories for Phi-3. The top row shows full-parameter updates and the bottom row shows LoRA updates; columns show the pre-unlearning checkpoint, ATU, GA, and NPO. The embedded "Unaligned" label denotes the pre-unlearning checkpoint. Green is the forgotten-answer best-subtoken score; blue is the model-output score; red is the question-entity score; purple is the task-irrelevant baseline.



(a) Unaligned (b) ATU (c) GA (d) NPO

(e) Unaligned (LoRA) (f) ATU (LoRA) (g) GA (LoRA) (h) NPO (LoRA)

Figure 4: Raw logit-lens trajectories for Qwen3 with full-parameter updates above and LoRA updates below, using the same column order and color mapping as Figure 3.

5 Conclusion

We distinguish reduced target answering from representational erasure. On RWKU, two instruction-tuned models updated with GA, NPO, or ATU retain elevated intermediate-layer target-subtoken ranks under a raw logit lens even when those ranks decrease near the output. This supports residual linear decodability in the tested settings, not universality or a realized security exploit. Our conditional analysis predicts a possible depth bias, but its premises are unverified on the evaluated checkpoints and it cannot explain the non-monotone mid-layer peak. Representation-targeting baselines, matched never-known controls, calibrated readouts, and non-oracle recovery remain the most important next tests.

AI use statement

In this work, we used generative AI tools solely for polishing writing to improve the readability ofthe manuscript. We have not used generative AI tools for research ideation, theoretical development,methodological or experimental design, mathematical derivations or proofs, code implementa-

tion,data analysis, or interpretation of experimental results. We have reviewed all AI-assisted work. We take responsibility for the final content of this work.

ETHICS STATEMENT

This work studies a potential weakness in unlearning-based alignment and is intended to support safer and more reliable deployment of LLMs. The experiments use public datasets and models and do not involve human subjects. No new harmful model, dataset, or attack asset is released by this manuscript.

REPRODUCIBILITY STATEMENT

The datasets, models, evaluated objectives, metrics, and principal hyperparameters are described in Section 4, with further implementation details in Appendix D. Derivations for the conditional theoretical results, together with their assumptions, are provided in Appendix C.

REFERENCES

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *CoRR*, abs/2404.05868, 2024.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. In *NeurIPS*, 2024.

Minkyoo Song, Hanna Kim, Jaehan Kim, Seungwon Shin, and Sooel Son. Refusal is not an option: Unlearning safety alignment of large language models. In *USENIX*, 2025.

Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. Language models implement simple word2vec-style vector arithmetic. In *NAACL-HLT*, 2024.

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

Yiwei Chen, Soumyadeep Pal, Yimeng Zhang, Qing Qu, and Sijia Liu. Unlearning isn't invisible: Detecting unlearning traces in LLMs from model outputs. *CoRR*, abs/2506.14003, 2025.

Aakash Sen Sharma, Debdeep Sanyal, Priyansh Srivastava, Sundar Atreya H., Shirish S. Karande, Mohan S. Kankanhalli, and Murari Mandal. Nine ways to break copyright law and why our LLM won't: A fair use aligned generation framework. *CoRR*, abs/2505.23788, 2025.

Philipp Spohn, Leander Girrbach, Jessica Bader, and Zeynep Akata. Align-then-unlearn: Embedding alignment for LLM unlearning. *CoRR*, 2025.

DeepSeek-AI. Deepseek-v3 technical report. *CoRR*, 2024.

Neel Nanda, Andrew Lee, and Martin Wattenberg. Emergent linear representations in world models of self-supervised sequence models. In *BlackboxNLP@EMNLP*, 2023.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. The geometry of categorical and hierarchical concepts in large language models. In *ICLR*, 2025.

Dake Bu, Wei Huang, Andi Han, Atsushi Nitanda, Qingfu Zhang, Hau-San Wong, and Taiji Suzuki. Provable in-context vector arithmetic via retrieving task concepts. In *ICML*, 2025.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *NAACL*, 2013.

Daniel Tamayo, Aitor Gonzalez-Agirre, Javier Hernando, and Marta Villegas. Mass-editing memory with attention in transformers: A cross-lingual exploration of knowledge. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5831–5847, 2024.

Paul Youssef, Zhixue Zhao, Christin Seifert, and Joerg Schloetterer. Tracing and reversing rank-one model edits. *CoRR*, abs/2505.20819, 2025.

Zexi Li, Xiangzhu Wang, William F. Shen, Meghdad Kurmanji, Xinchi Qiu, Dongqi Cai, Chao Wu, and Nicholas D. Lane. Editing as unlearning: Are knowledge editing methods strong baselines for large language model unlearning? *CoRR*, abs/2505.19855, 2025.

Ronen Eldan and Mark Russinovich. Who's Harry Potter? Approximate unlearning in LLMs. *CoRR*, abs/2310.02238, 2023.

Kevin Christian Wibisono and Yixin Wang. Bidirectional attention as a mixture of continuous word experts. In *UAI*, 2023.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *EMNLP*, 2021.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *ICML*, 2024.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. *CoRR*, abs/2410.07163, 2024.

Andy Zou, Long Phan, Justin Wang, Derek Duenas, Maxwell Lin, Maksym Andriushchenko, Rowan Wang, Zico Kolter, Matt Fredrikson, and Dan Hendrycks. Improving alignment and robustness with circuit breakers. In *NeurIPS*, 2024.

Domenic Rosati, Jan Wehner, Kai Williams, Lukasz Bartoszcze, David Atanasov, Robie Gonzales, Subhabrata Majumdar, Carsten Maple, Hassan Sajjad, and Frank Rudzicz. Representation noising: A defence mechanism against harmful finetuning. In *NeurIPS*, 2024.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *CoRR*, abs/2303.08112, 2023.

A Related Work

LLM unlearning methods differ in where and how they intervene. Gradient ascent and NPO (Zhang et al., 2024; Yao et al., 2024) optimize losses defined from target outputs, while SimNPO (Fan et al., 2024) removes NPO's reference-model dependence. ATU (Spohn et al., 2025) optimizes similarity in a learned embedding space. RMU (Li et al., 2024) directly steers intermediate representations and is therefore the most direct missing baseline for testing whether our observation is specific to the evaluated objectives. "Who's Harry Potter?" (Eldan and Russinovich, 2023) and model-editing approaches (Tamayo et al., 2024; Youssef et al., 2025; Li et al., 2025) provide additional, distinct mechanisms; we discuss them but do not claim empirical coverage.

Representation-level safety methods are adjacent rather than interchangeable comparisons. Representation Rerouting/Circuit Breakers (Zou et al., 2024) changes harmful-output representations to improve adversarial robustness, while RepNoise (Rosati et al., 2024) removes harmful representations across layers to resist subsequent harmful fine-tuning. These works make clear that a conclusion drawn from GA, NPO, and ATU should not be generalized to all unlearning or representation-level interventions.

Our measurement is also situated within mechanistic-interpretability work. A raw logit lens applies the shared unembedding to intermediate hidden states, whereas the Tuned Lens (Belrose et al., 2023) learns affine translators that improve calibration and predictive reliability. Consequently, a high raw-logit-lens rank demonstrates accessibility to a particular linear readout but does not by itself show that the model would generate the answer, that the representation is causally used, or that a non-oracle attacker can recover it.

B Discussion

B.1 Limitation

Empirical scope. The evaluation covers two similarly sized instruction-tuned models, one factual-unlearning benchmark, and three objectives. It does not include a representation-targeting unlearning method such as RMU, larger models, other architectures, non-factual forget tasks, or other sensitive-information domains. The results therefore establish a repeated pattern within the evaluated grid, not universality across unlearning.

Controls and recoverability. The task-irrelevant purple tokens are plotting baselines, not matched never-known facts or gold retain-model controls. Without such controls, we cannot establish that the mid-layer elevation is anomalous relative to ordinary logit-lens recall. Moreover, the target subtokens are supplied to the diagnostic in advance. We do not present peak-layer generation, RWKU Level-3 recovery, or another non-oracle extraction attack; residual decodability should not be described as a demonstrated deployment vulnerability.

Readout and aggregation. The shared LM head is a raw, uncalibrated readout of intermediate states. We do not report an independently trained probe or Tuned Lens. For multi-token answers, we take the maximum reciprocal rank over a deterministic candidate-subtoken set. This recall-oriented maximum can increase with the size and composition of that set, and we do not provide mean/first-token aggregation ablations. The evidence is therefore specific to the stated best-subtoken metric.

Conditional theory. The analysis freezes the contextual feature map, uses a mean-centered LayerNorm Jacobian although Phi-3 and Qwen3 use RMSNorm, and assumes local operator contraction. Boundedness of the feature-map Jacobian does not imply this contraction, and the relevant constants are not measured on the evaluated checkpoints. In addition, the theorems yield a monotone depth bias, whereas the empirical pattern is a non-monotone mid-stack peak. The theory is thus a stylized sufficient-condition analysis for late-layer update dominance, not a validated causal explanation of the complete curve.

B.2 Broader Impacts

Our results motivate evaluation beyond final outputs when studying whether a particular unlearning procedure removes target information. The raw logit lens is an inexpensive diagnostic, but its limitations should be reported alongside its trajectories and it should not substitute for calibrated probes, causal tests, or recovery evaluations.

The conditional analysis highlights which operator norms and update magnitudes would need to be measured before making a causal claim about layer-wise unlearning dynamics. We hope this separation between observation, readout, and mechanism supports more precise evaluation of future unlearning methods.

B.3 Concept Vector Modeling versus Word2Vec Modeling

We briefly discuss the connection and distinction between concept vector modeling and classic word embedding modeling (e.g., the Word2Vec embedding (Mikolov et al., 2013)). As previously mentioned, our operations on concept vectors within the residual stream exhibit a high degree of formal similarity to the vector arithmetic properties demonstrated by the classic Word2Vec model (Mikolov et al., 2013), such as the canonical example:

$$\mathrm{emb}(\text{King}) - \mathrm{emb}(\text{Man}) + \mathrm{emb}(\text{Woman}) \approx \mathrm{emb}(\text{Queen}),$$

where $\mathrm{emb}(\cdot)$ denotes the Word2Vec word embedding function.

Mechanistic Similarity. Although the Transformer architecture is inherently highly non-linear, recent studies suggest that its inference process shares mechanistic similarities with static embeddings. Theoretically, Wibisono and Wang (2023) established a formal connection between the optimization

of attention mechanisms and Word2Vec objectives; empirically, Merullo et al. (2024) indicated that, from the perspective of the residual stream, the model's inference process can be approximated as a sequence of vector addition operations. Specifically, Merullo et al. (2024) pointed out that the MLP layers within the Transformer architecture function analogously to key-value memories, capable of identifying specific input patterns and adding corresponding task-relevant concept vectors into the residual stream (Geva et al., 2021). The analysis of concept vectors in this section draws upon this proposition. This mechanism enables deep language models to perform semantic translation operations akin to Word2Vec.

Fundamental Distinction. Despite similar operational mechanisms, there exists a fundamental distinction between the two at the representational level: Word2Vec is static, whereas the Transformer is dynamic. On the one hand, the Word2Vec word embeddings proposed by Mikolov et al. (2013) are globally fixed; that is, the same word shares an identical vector representation across different contexts. On the other hand, as described by Bu et al. (2025), vector operations in LLMs are context-dependent; the hidden state at each layer not only contains information about the current token but also dynamically aggregates the semantics of preceding tokens. Therefore, the "concept" discussed in this paper is not a static lexical item, but rather a high-level semantic direction dynamically activated by a specific context.

C Derivations for the Conditional Results

This section derives the conditional results in Section 3. The logical dependencies are:

1. Proposition C.1 gives the exact gradient recursion for the idealized residual block.
2. Lemma C.2 bounds the contextual-feature Jacobian. This bound does *not* imply contraction of $I + W_\theta^{(m)} J_c^m$.
3. Theorem 3.2 follows only after the contraction condition in its statement is imposed explicitly.
4. Theorem 3.1 follows from Proposition C.1 and the non-degeneracy and contraction conditions in Assumption C.1.

We first introduce the concept of influence intensity to quantify the impact of data on network parameters. Combining Equations (11) and (12), it is straightforward to observe that the iteration of $W_{\theta\{t\}}^{(m)}$ is linear with respect to the samples in $\mathcal{D}_{\mathrm{FG}}$. Therefore, we can consider the impact of a single sample (x, y) on the model parameter updates without loss of generality. Specifically, $\nabla_W \ell$ in Equation (12) can be obtained by contracting $\nabla_{h^{(m,1)}} \ell$ and the Jacobian matrix of $h^{(m,1)}$ with respect to $W^{(m)}$ via the chain rule, namely

$$\nabla_W \ell((x, y); \theta) = \nabla_{h^{(m,1)}} \ell((x, y); \theta) \cdot \frac{\partial h^{(m,1)}(x)}{\partial W}. \tag{25}$$

On the other hand, combining Equations (3) and (13), we have

$$h_\theta^{(m,1)}(x) = h_\theta^{(m-1,1)}(x) + W_\theta^{(m)} \operatorname{Attn}_\theta^{(m)}\left(h_\theta^{(m-1,1)}(x)\right), \qquad W_\theta^{(m)} \in \mathbb{R}^{d_h \times d_h}. \tag{26}$$

Since $\operatorname{Attn}_\theta^{(m)}(x)$ is independent of $W_\theta^{(m)}$, Equation (25) can be further simplified to

$$\nabla_{W_\theta^{(m)}} \ell((x, y); \theta) = (\nabla_{h^{(m,1)}} \ell((x, y); \theta))^\top \left(\operatorname{Attn}_\theta^{(m)}(h_\theta^{(m-1,1)}(x))\right)^\top. \tag{27}$$

In summary, for any $m \in [M]$, we define the influence intensity of the sample (x, y) on $W_\theta^{(m)}$ as

$$\begin{aligned} I_m(x, y) &:= \left\| \nabla_{W_\theta^{(m)}} \ell((x, y); \theta) \right\|_F \\ &= \|\nabla_{h^{(m,1)}} \ell((x, y); \theta)\|_2 \left\| \operatorname{Attn}_\theta^{(m)}(h_\theta^{(m-1,1)}(x)) \right\|_2. \end{aligned} \tag{28}$$

This precisely corresponds to the norm of the iterative update increment of $W_\theta^{(m)}$ induced by the sample (x, y). Note that we omit the notation regarding the iteration step t in Equation (18) where unambiguous. For any $m \in [M]$, we concisely denote

$$g^{(m)} := \nabla_{h^{(m,1)}} \ell((x, y); \theta) \in \mathbb{R}^{1 \times d_h} \tag{29}$$

and let $g_{\{t\}}^{(m)}$ represent the value of $g^{(m)}$ at the t-th iteration. The following proposition provides the recursive formulation for $g_{\{t\}}^{(m)}$.

Proposition C.1. *For any $m \in [M-1]$, under the gradient ascent training paradigm defined by Equation (12), $g^{(m)}$ satisfies*

$$g^{(m)} = g^{(M)} \prod_{m'=m+1}^{M} \left(I + W_{\theta\{t\}}^{(m')} J_c^{m'} \right), \tag{30}$$

where, for any $m \in [M]$, J_c^m is defined as

$$J_c^m := \frac{\partial \operatorname{Attn}_\theta^{(m)}}{\partial h_\theta^{(m-1,1)}} = \frac{\partial \operatorname{Attn}_\theta^{(m)}(x)}{\partial h_\theta^{(m-1,1)}(x)}, \tag{31}$$

and $g^{(M)}$ satisfies

$$g^{(M)} = \sum_{v_i \in \mathcal{V}} p_{v_i} v_i - v_y. \tag{32}$$

Note that Equations (30)–(32) hold for any iteration step t; consequently, the subscript t is omitted for notational brevity.

Proposition C.1 unifies the layer-wise gradient propagation into a product formulation, as expressed in Equation (30). This formulation ensures that, in subsequent analyses, the comparison of the differential influence between any two layers can be reduced to evaluating the norm of the local operator $I + W_\theta^{(m)} J_c^m$.

Proof of Proposition C.1. It is straightforward to observe that

$$\frac{\partial h^{(m,1)}}{\partial h_\theta^{(m-1,1)}} = I + W_\theta^{(m)} \frac{\partial \operatorname{Attn}_\theta^{(m)}}{\partial h_\theta^{(m-1,1)}} = I + W_\theta^{(m)} J_c^m. \tag{33}$$

Building upon this, the chain rule yields

$$g^{(m-1)} = g^{(m)} \left(I + W_\theta^{(m)} J_c^m \right), \tag{34}$$

which implies that as the gradient propagates backwards from the output end towards the input end, it is right-multiplied by a factor of $I + W_\theta^{(m)} J_c^m$ at each layer. Recursively applying Equation (34) from the M-th layer down to the $(m+1)$-th layer directly results in Equation (30). For any iteration step t, combining this with Equations (33) and (34) yields

$$g_{\{t\}}^{(m)} = \left(\left(g_{\{t\}}^{(M)} \right)^\top \right)^\top \prod_{m'=m+1}^{M} \left(I + W_{\theta\{t\}}^{(m')} J_c^{m'} \right), \tag{35}$$

demonstrating that Equation (30) holds for any iteration step t. Regarding $g^{(M)}$, by combining Equations (9), (10), and (29), we obtain

$$\begin{aligned} g^{(M)} &= \nabla_{h^{(M)}} \left[-\log \frac{\exp(v_y^\top h_\theta(x))}{\sum_{v' \in \mathcal{V}} \exp(v_{v'}^\top h_\theta(x))} \right] \\ &= \sum_{v_i \in \mathcal{V}} p_{v_i} v_i - v_y, \end{aligned} \tag{36}$$

which proves Equation (32); furthermore, it is trivial to verify that this conclusion remains valid at any iteration step t. □

Proposition C.1 provides the recursive formula for a pivotal term in the iteration of $W^{(m)}_{\theta\{t\}}$. Based on this, it is straightforward to derive the iterative expression for $W^{(m)}_{\theta\{t\}}$, which subsequently leads to the following lemma.

Lemma C.1. *For any $m \in [M]$, the iteration of $W^{(m)}_{\theta\{t\}}$ satisfies*

$$W^{(m)}_{\theta\{t+1\}} = W^{(m)}_{\theta\{t\}} + \frac{\eta}{N} \sum_{(x,y)\in\mathcal{D}_{\mathrm{FG}}} \left(\left(g^{(m)}_{\{t\}} \right)^\top \left(\mathrm{Attn}^{(m)}_\theta(x) \right)^\top \right), \tag{37}$$

where the formulation of $g^{(m)}_{\{t\}}$ is provided by Proposition C.1.

Proof. Equation (37) can be directly derived from Proposition C.1 and Equation (12); the derivation is omitted here. □

Proposition C.1 and Lemma C.1 establish the iteration dynamics of the idealized trainable map. To compare two layers without conflating a conclusion with its premise, we state all additional non-degeneracy and local-operator conditions explicitly.

Assumption C.1. *For fixed $m_1 < m_2 \le M$ and sample (x, y) with $\|g^{(m_2)}\|_2 > 0$, assume there are constants $\sigma_{m_1}, \tau^w_{m_1}, \tau^c_{m_2} > 0$ and $\alpha_{m'} \ge 0$ such that*

$$\sigma_{\min}\left(W^{(m_1)}_\theta\right) \ge \sigma_{m_1}, \qquad \left\| W^{(m_1)}_\theta \,\mathrm{Attn}^{(m_1)}_\theta(x) \right\|_2 \le \tau^w_{m_1}, \qquad \left\| \mathrm{Attn}^{(m_2)}_\theta(x) \right\|_2 \ge \tau^c_{m_2}, \tag{38}$$

and, for each $m' \in \{m_1 + 1, \ldots, m_2\}$,

$$\left\| I + W^{(m')}_\theta J^{m'}_c(x) \right\|_2 \le \alpha_{m'}. \tag{39}$$

These conditions are sufficient assumptions for the comparison below. They are not derived from boundedness of $J^{m'}_c$ and are not measured on the evaluated checkpoints.

We next bound the contextual-feature Jacobian. This is useful for local regularity, but it is logically distinct from contraction of the residual operator.

Lemma C.2. *Under Assumption 3.1, there exist constants $\kappa_m, \rho_m > 0$ such that, over the same local trajectory,*

$$\sup_x \|D\,\mathrm{Attn}^{(m)}_\theta(\mathrm{LN}(x))\|_2 \le \kappa_m, \qquad \sup_x \|J^m_{\mathrm{LN}}(x)\|_2 \le \rho_m. \tag{40}$$

More specifically, we can define

$$\kappa_m := \frac{\tau_m}{2} B^{(m)}_V B^{(m)}_K B^{(m)}_Q, \qquad \rho_m := \frac{2G_m}{\sqrt{\sigma^2_m + \varepsilon}}. \tag{41}$$

In particular, we have

$$\kappa_m \rho_m < 1, \qquad \|J^m_c(x)\|_2 \le \kappa_m \rho_m < 1. \tag{42}$$

This lemma bounds $\|J^m_c\|_2$ in the idealized mean-centered LayerNorm model. Crucially, $\|J^m_c\|_2 < 1$ does not imply $\|I + W^{(m)}_\theta J^m_c\|_2 < 1$; the latter remains the explicit premise of Theorem 3.2.

Proof of Lemma C.2. By the chain rule, we have

$$J^m_c(x) = D\,\mathrm{Attn}^{(m)}_\theta\left(\mathrm{LN}(h^{(m-1,1)}_\theta(x))\right) J^m_{\mathrm{LN}}(h^{(m-1,1)}_\theta(x)). \tag{43}$$

Let $s(u) := \tau_m K^{(m)\top} W^{(m)}_Q u$ and $p(u) := \mathrm{softmax}(s(u))$, then

$$D\,\mathrm{Attn}^{(m)}_\theta(u) = V^{(m)\top} J_{\mathrm{softmax}}(s(u)) \tau_m K^{(m)\top} W^{(m)}_Q. \tag{44}$$

Utilizing submultiplicativity of the spectral norm and $\|J_{\mathrm{softmax}}(z)\|_2 \le 1/2$, we obtain

$$\|D\,\mathrm{Attn}^{(m)}_\theta(u)\|_2 \le \frac{\tau_m}{2} \|V^{(m)}\|_2 \|K^{(m)}\|_2 \|W^{(m)}_Q\|_2 \le \frac{\tau_m}{2} B^{(m)}_V B^{(m)}_K B^{(m)}_Q = \kappa_m. \tag{45}$$

Furthermore, from the standard form of the layer normalization Jacobian,

$$J^m_{\mathrm{LN}}(x) = \mathrm{diag}(\gamma_m)\frac{1}{\sqrt{\sigma^2(x)+\varepsilon}}\left[I - \frac{1}{d_h}\mathbf{1}\mathbf{1}^\top - \frac{(x-\mu(x)\mathbf{1})(x-\mu(x)\mathbf{1})^\top}{d_h(\sigma^2(x)+\varepsilon)}\right], \tag{46}$$

combining this with $\|\mathrm{diag}(\gamma_m)\|_2 \le \|\gamma_m\|_\infty \le G_m$ and $\sigma^2(x) \ge \sigma_m^2$, we get

$$\|J^m_{\mathrm{LN}}(x)\|_2 \le \frac{2G_m}{\sqrt{\sigma_m^2+\varepsilon}} = \rho_m. \tag{47}$$

Thus, the supremum bounds over the analyzed trajectory hold. If the small-gain condition is further satisfied, then $\kappa_m\rho_m < 1$. Finally, applying submultiplicativity again,

$$\|J^m_c(x)\|_2 \le \|D\,\mathrm{Attn}^{(m)}_\theta(\mathrm{LN}(h^{(m-1,1)}_\theta(x)))\|_2 \|J^m_{\mathrm{LN}}(h^{(m-1,1)}_\theta(x))\|_2 \le \kappa_m\rho_m < 1. \tag{48}$$

□

We now prove Theorem 3.2 under its explicit contraction premise.

Proof of Theorem 3.2. By the chain rule, for $m \in \{1,\dots,M-1\}$,

$$g^{(m-1)} = g^{(m)}\left(I + W^{(m)}_\theta J^m_c(x)\right), \tag{49}$$

and for $m = M$,

$$g^{(M-1)} = (\nabla_{h^{(M,1)}}\ell((x,y);\theta))^\top\left(I + W^{(M)}_\theta J^M_c(x)\right). \tag{50}$$

Taking the ℓ_2 norm on both sides and applying submultiplicativity, we obtain, for $m \in \{1,\dots,M-1\}$,

$$\|g^{(m-1)}\|_2 \le \|g^{(m)}\|_2\|I + W^{(m)}_\theta J^m_c(x)\|_2 \le \alpha_m\|g^{(m)}\|_2, \tag{51}$$

and

$$\begin{aligned}\|g^{(M-1)}\|_2 &\le \|\nabla_{h^{(M,1)}}\ell((x,y);\theta)\|_2\|I + W^{(M)}_\theta J^M_c(x)\|_2 \\ &\le \alpha_M\|\nabla_{h^{(M,1)}}\ell((x,y);\theta)\|_2.\end{aligned} \tag{52}$$

Since $\alpha_m < 1$, the corresponding strict inequalities immediately follow. Iterating over the layer indices (or by mathematical induction), when $m_2 \le M-1$, we have

$$\|g^{(m_1)}\|_2 \le \alpha_{m_1+1}\|g^{(m_1+1)}\|_2 \le \cdots \le \left(\prod_{m'=m_1+1}^{m_2}\alpha_{m'}\right)\|g^{(m_2)}\|_2.$$

When $m_2 = M$, substituting $\|\nabla_{h^{(M,1)}}\ell((x,y);\theta)\|_2$ as the terminal factor in the chain product at the last step yields the stated bound. □

We finally prove Theorem 3.1 from Proposition C.1 and Assumption C.1.

Proof of Theorem 3.1. From Equation (18), we have

$$\frac{I_{m_1}(x,y)}{I_{m_2}(x,y)} = \frac{\|\nabla_{h^{(m_1,1)}}\ell((x,y);\theta)\|_2}{\|\nabla_{h^{(m_2,1)}}\ell((x,y);\theta)\|_2}\cdot\frac{\|\,\mathrm{Attn}^{(m_1)}_\theta(x)\|_2}{\|\,\mathrm{Attn}^{(m_2)}_\theta(x)\|_2}. \tag{53}$$

We first estimate the second factor. The minimum-singular-value condition in Assumption C.1 gives

$$\|W^{(m_1)}_\theta\,\mathrm{Attn}^{(m_1)}_\theta(x)\|_2 \ge \sigma_{m_1}\|\,\mathrm{Attn}^{(m_1)}_\theta(x)\|_2, \tag{54}$$

hence

$$\|\,\mathrm{Attn}^{(m_1)}_\theta(x)\|_2 \le \frac{\tau^w_{m_1}}{\sigma_{m_1}}. \tag{55}$$

Further, from Assumption C.1,

$$\|\,\mathrm{Attn}^{(m_2)}_\theta(x)\|_2 \ge \tau^c_{m_2} > 0. \tag{56}$$

Therefore,

$$\frac{\|\operatorname{Attn}_\theta^{(m_1)}(x)\|_2}{\|\operatorname{Attn}_\theta^{(m_2)}(x)\|_2} \le \frac{\tau_{m_1}^w}{\sigma_{m_1}\tau_{m_2}^c}. \tag{57}$$

We then estimate the first factor. From Proposition C.1, for any layer m,

$$\|\nabla_{h^{(m,1)}}\ell((x,y);\theta)\|_2 = \left\| \left(\sum_{v_i \in \mathcal{V}} p_{v_i}(x) v_i - v_y \right) \prod_{m'=m+1}^{M} \left(I + W_\theta^{(m')} J_c^{m'}(x) \right) \right\|_2. \tag{58}$$

Proposition C.1 also gives the direct cross-layer relation

$$g^{(m_1)} = g^{(m_2)} \prod_{m'=m_1+1}^{m_2} \left(I + W_\theta^{(m')} J_c^{m'}(x) \right). \tag{59}$$

Because $\|g^{(m_2)}\|_2 > 0$, submultiplicativity and Assumption C.1 yield

$$\frac{\|\nabla_{h^{(m_1,1)}}\ell((x,y);\theta)\|_2}{\|\nabla_{h^{(m_2,1)}}\ell((x,y);\theta)\|_2} \le \prod_{m'=m_1+1}^{m_2} \alpha_{m'}. \tag{60}$$

Multiplying the above two parts directly yields

$$\frac{I_{m_1}(x,y)}{I_{m_2}(x,y)} \le \left(\prod_{m'=m_1+1}^{m_2} \alpha_{m'} \right) \frac{\tau_{m_1}^w}{\sigma_{m_1}\tau_{m_2}^c}. \tag{61}$$

When the displayed right-hand side is below one, the assumptions are sufficient for $I_{m_1}(x,y) < I_{m_2}(x,y)$. This is a conditional comparison, not an empirical claim about the evaluated checkpoints. □

D Experimental Details

This section provides implementation details for the layer-wise diagnostic and evaluated unlearning objectives.

Implementation Details. For each prompt x, we perform a forward pass, extract $h_\theta^{(m,1)}(x)$ at the query position after every layer, and apply the shared LM head λ directly to obtain the raw logit-lens distribution $p_\theta^{(m,1)}$. We do not train an independent probe, affine translator, or Tuned Lens. We sort the full vocabulary and compute reciprocal rank for each candidate subtoken. Curves track forgotten-answer subtokens (green), question-entity subtokens (red), generated model-output subtokens (blue), and task-irrelevant baseline subtokens (purple).

Remark D.1 (Candidate-Subtoken Aggregation). For a keyword or multi-token answer, deterministic surface-form rules produce a candidate-subtoken set $\mathcal{T}(y)$ that may include tokenizer variants such as `Po`, `Port`, and `portland` for `Portland`. The reported layer score is $\max_{v\in\mathcal{T}(y)} r_\theta^{(m,1)}(v \mid x)$. Candidate sets for the four plotted tracks are required to be pairwise disjoint. This maximum is a recall-oriented "best-subtoken" statistic: it does not show that the full answer is represented as a single unit, and larger or more permissive candidate sets can increase the score. We do not report mean- or first-token aggregation ablations.

GA uses AdamW with learning rate 6×10^{-8} for 8 epochs (up to 2000 steps). NPO uses $\beta = 0.1$ and AdamW with learning rate 6×10^{-8} for 250 steps. ATU alternates embedding-head and LLM updates while decreasing its similarity threshold from 0.9 to 0.2; the embedding head uses Adam with learning rate 10^{-4} and the LLM uses SGD with learning rate 3×10^{-4}. Experiments use PyTorch Lightning, Hydra, BF16 mixed precision, and NVIDIA A800 80GB GPUs, consuming approximately 10^2 GPU hours.