CURE: COUNTERFACTUAL UNLEARNING OF LARGE LANGUAGE MODELS VIA DENSITY RATIO ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning for large language models aims to remove the influence of designated training data while preserving behavior supported by the remaining data. Yet knowing *what to forget* does not determine *what the model should predict after forgetting*: the factual prediction entangles retain- and forget-supported influences, making the desired retain-only behavior unidentifiable from the original model alone. We show that this ambiguity can be resolved by introducing a controlled second observation that amplifies the forget influence. Under a retain–forget mixture model, our Bidirectional Mixture Identification result reduces recovery of the retain-only next-token distribution to estimating a context-dependent forgetting ratio. We further show that this ratio is characterized by the density ratio between the factual and forget-amplified models and can be updated autoregressively during generation. Based on these results, we propose **CURE** (Counterfactual Unlearning via Ratio Estimation), which uses a forget-aligned auxiliary model to reveal the direction of removal and contextual ratio estimation to determine its magnitude, while keeping the factual model fixed. We additionally analyze how model-realization, auxiliary-alignment, and ratio-estimation errors affect recovery. Across TOFU, MUSE, and RealToxicityPrompts, CURE achieves favorable forgetting–utility trade-offs; on TOFU-10%, it attains forget/retain ROUGE-L of 38.3/98.4, compared with 39.6/99.2 for the retain-only reference. These results support counterfactual unlearning as an identification-and-estimation problem rather than solely an objective for suppressing forget-associated behavior. Code will be released upon publication.

1 INTRODUCTION

Machine unlearning for large language models (LLMs) aims to remove the influence of designated training data while preserving knowledge supported by the remaining data (Bourtoule et al., 2021; Maini et al., 2024; Shi et al., 2024). Existing approaches often reduce the likelihood of forget-set examples or associated behavior (Jang et al., 2022; Zhang et al., 2024). Yet *what to forget* does not determine *what to predict after forgetting*: a continuation may be supported by both forget and retain data, making indiscriminate suppression harmful to retained knowledge. Conversely, insufficient suppression may leave residual memorization. Unlearning benchmarks accordingly use retain-only retraining as a reference. We ask: *without retraining on the retain set, can we infer the prediction that should remain after removing the forget influence?*

This needs to the separate retain- and forget-supported influences from the factual prediction. Under a retain–forget mixture model, our *Bidirectional Mixture Identification* result establishes that, given the factual and forget-source conditionals, identifying a feasible contextual source posterior and identifying the retain conditional are equivalent in the nondegenerate regime. This reduces distribution recovery to weight identification, but supplies neither the forget-source conditional nor its weight. Constructive recovery therefore requires additional information, which we obtain through a controlled auxiliary observation and calibrated prefix density ratios. Figure 1 summarizes this identification-and-recovery framework.

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.

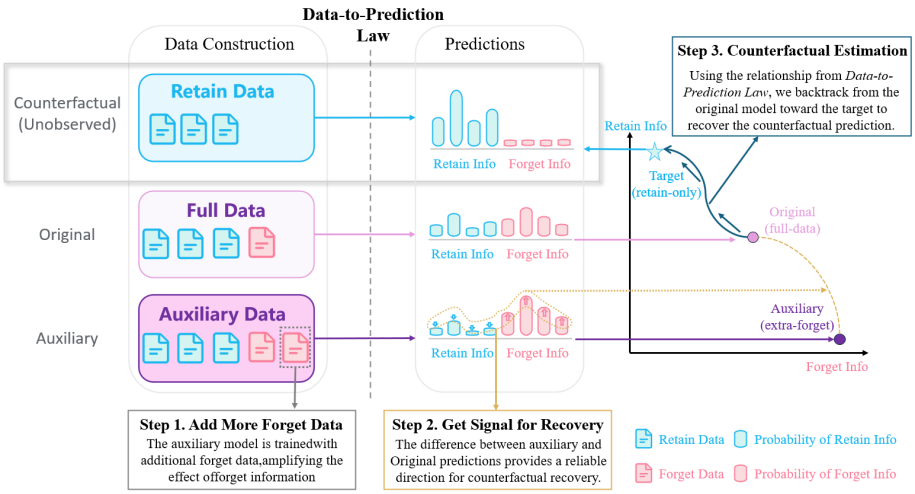


Figure 1: **Identifying the retain prediction.** Theorem 2.1 relates source-posterior and retain-conditional identification given the factual and forget-source conditionals; Theorem 4.1 combines the amplified observation with prefix-ratio estimation. The depicted "Data-to-Prediction Law" concerns ideal distributions; Section 3 analyzes neural realization.

Rather than directly observing the unavailable effect of removing data, we examine the predictive response to increasing their influence. We construct a forget-aligned auxiliary model to approximate an observation that *amplifies* the forget influence. Under the mixture assumptions, the shift from the factual to the amplified prediction exposes the direction to reverse for retain-only recovery. Reinforced or assistant models have previously provided alternative targets or logit manipulations for unlearning (Eldan & Russinovich, 2023; Ji et al., 2024); here, the auxiliary approximates the controlled second observation required by our recovery theorem. Amplification reveals the removal direction, while its appropriate magnitude depends on the contextual forget influence, which varies across prompts and generated prefixes and generally cannot be captured by a fixed coefficient.

We determine this magnitude through *contextual ratio estimation*. Under shared-source and calibration assumptions, the factual and amplified models' prefix density ratio identifies the source posterior and admits an autoregressive update from token likelihoods (Sugiyama et al., 2012). Tracking begins at a shared beginning-of-sequence (BOS) marker and processes both prompt and response tokens, without requiring a prompt-specific source posterior as input. CURE (**C**ounterfactual **U**nlearning via **R**atio **E**stimation) combines this estimate with the auxiliary shift to correct next-token probabilities while keeping the factual model fixed. Our analysis quantifies realization, auxiliary-alignment, ratio-estimation, and repair errors, including their effects during autoregressive generation.

We evaluate CURE on TOFU (Maini et al., 2024), MUSE (Shi et al., 2024), and toxic-generation suppression using RealToxicityPrompts (Gehman et al., 2020). Under the recorded benchmark and decoder configurations, CURE achieves favorable forgetting–utility trade-offs. On TOFU-10%, its forget/retain ROUGE-L is 38.3/98.4, compared with 39.6/99.2 for the retain-only reference. These behavioral comparisons support the practical framework while leaving distributional recovery under neural approximation to be assessed separately. Our contributions are three-fold:

- We establish Bidirectional Mixture Identification and analyze recovery under realization, alignment, estimation, and repair errors, including recursive state-error accumulation and finite-sample guarantees under controlled sequence sampling.

- We derive constructive recovery using a controlled auxiliary observation and BOS-based prefix ratios under calibration assumptions, and instantiate CURE with frozen factual weights, decoder-state clipping, and probability repair.
- We evaluate forgetting–utility trade-offs, toxicity suppression, auxiliary sensitivity, and computational cost on TOFU, MUSE (News and Books), and RealToxicityPrompts.

2 IDENTIFYING THE RETAIN-ONLY PREDICTION

Predictive target. Let $\mathcal{D}_r$ and $\mathcal{D}_f$ be the retained and forgotten training sets, whose disjoint union is $\mathcal{D} = \mathcal{D}_r \dot{\cup} \mathcal{D}_f$. Fix a training procedure $\mathcal{A}$, including initialization and randomness. The full-data and retain-only predictors are $q_D = \mathcal{A}(\mathcal{D})$ and $q_R = \mathcal{A}(\mathcal{D}_r)$. For input x and output sequence y, write $h_t = (x, y_{<t})$ for the context before token t, where $y_{<t} = (y_1, \ldots, y_{t-1})$. Our target is the next-token distribution $q_R(\cdot \mid h_t)$: the dot ranges over tokens v in the finite vocabulary $\mathcal{V}$.

2.1 BIDIRECTIONAL MIXTURE IDENTIFICATION

At a fixed context h_t, let p_D, p_R, p_F denote the ideal factual, retain-source, and forget-source next-token distributions. For contextual forget-source posterior $r_t \in [0, 1]$, assume

$$p_D = (1 - r_t)p_R + r_t p_F. \tag{1}$$

Theorem 2.1 (Bidirectional Mixture Identification)**.** *Let Eq. equation 1 hold with* $0 \le r_t < 1$. (i) Weight to retain conditional. *Given* p_D, p_F, r_t*, the retain conditional is uniquely determined by*

$$p_R = \frac{p_D - r_t p_F}{1 - r_t}. \tag{2}$$

(ii) Retain conditional to weight. *If* $p_F \neq p_R$*, then* p_D, p_F, p_R *uniquely determine*

$$r_t = \frac{\|p_D - p_R\|_1}{\|p_F - p_R\|_1}. \tag{3}$$

Thus, given distinct p_D, p_F*, the feasible weight and the retain conditional are equivalent.*

Proof sketch. Rearrangement proves (i); taking norms in $p_D - p_R = r_t(p_F - p_R)$ proves (ii). Appendix C.1 treats boundary cases.

The weight remains unidentified. The pair p_D, p_F can admit multiple weights and retain laws (Appendix C.1). Classical mixture-proportion estimation addresses this ambiguity through irreducibility or side information (Blanchard et al., 2010; Scott, 2015; Zhu et al., 2023). We instead construct an auxiliary observation without an oracle forget-source predictor.

2.2 THE MISSING INFORMATION AND A CONTROLLED SECOND OBSERVATION

Finite forget data $\mathcal{D}_f$ do not specify $p_F(\cdot \mid h_t)$ at arbitrary generated prefixes; a trained predictor only approximates this ideal conditional.

Use the same subscripts for sequence laws, token conditionals, and prefix marginals. Let $S \in \{R, F\}$ label the source and $\alpha(x) = \Pr(S = F \mid x)$ be its posterior after the input. This theoretical quantity is not a CURE input; Section 4 and Appendix B.1 give its BOS-based recovery under the joint-mixture and calibration assumptions. Assume

$$p_D(y \mid x) = (1 - \alpha(x))p_R(y \mid x) + \alpha(x)p_F(y \mid x). \tag{4}$$

On positive full-data prefix support, Bayes' rule gives

$$r_t = \Pr(S = F \mid h_t) = \frac{\alpha(x)p_F(y_{<t} \mid x)}{p_D(y_{<t} \mid x)}, \qquad r_1 = \alpha(x). \tag{5}$$

Known p_D, p_F and $\alpha(x)$ permit oracle recursion without an auxiliary (Appendix C.1). CURE instead approximates a controlled second mixture. For known amplification $\kappa > 0$, define

$$p_A(y \mid x) = \frac{p_D(y \mid x) + \kappa\alpha(x)p_F(y \mid x)}{1 + \kappa\alpha(x)}. \tag{6}$$

Conditioning on the prefix preserves the components and gives

$$p_D = (1 - r_t)p_R + r_t p_F, \qquad p_A = \frac{p_D + \kappa r_t p_F}{1 + \kappa r_t}. \tag{7}$$

The added contextual mass is $\rho_t = \kappa r_t$; it generally differs from a global auxiliary-data ratio. Auxiliary fine-tuning approximates this construction only when alignment is justified; matching input marginals alone does not ensure matching prefix laws.

The amplified pair supplies a removal direction but need not identify its strength (Appendix C.5); Section 4 provides calibrated recovery.

3 Reliability under Neural Approximation

The conditional mixtures in Eq. equation 7 suggest two operations: reverse the auxiliary shift and update its contextual weight. To define the quantities analyzed here, let d, b be factual and auxiliary probability vectors, s a scalar state, and $\ell > 0$ an auxiliary-to-factual token likelihood ratio. Define the recovery operator for $s < 1$ and the update map for any s by

$$\mathcal{R}_\kappa(d, b, s) = \frac{(1 + \kappa)d - (1 + \kappa s)b}{\kappa(1 - s)}, \; F(s, \ell) = \frac{(1 + \kappa s)\ell - 1}{\kappa}. \tag{8}$$

The ideal operations use $d = p_D$, $b = p_A$, and source posterior $s = r_t$; their neural counterparts use q_D, q_A and decoder state $\widehat{r}_t$. We first quantify their sensitivity to approximation and estimation errors. Section 4 then proves why these operations identify the retain conditional under controlled amplification.

3.1 Error Propagation to the Retraining Target

Let $p_A^\dagger$ be the specified population law of the actual auxiliary construction. At a fixed $h = h_t$, suppress the token index on $r_t, \widehat{r}_t, z_t$ and define

$$\begin{aligned} \epsilon_D &= \|q_D - p_D\|_1, \quad \epsilon_A = \left\|q_A - p_A^\dagger\right\|_1, \quad \epsilon_{\text{align}} = \left\|p_A^\dagger - p_A\right\|_1, \\ \epsilon_r &= |\widehat{r} - r|, \qquad \epsilon_R = \|q_R - p_R\|_1, \qquad d_{DA} = \|p_D - p_A\|_1. \end{aligned} \tag{9}$$

Without a specified auxiliary population target, the analysis uses the combined discrepancy $\|q_A - p_A\|_1$; separating its causes requires additional information.

Theorem 3.1 (Recovery under approximation)**.** *Assume the local conditional mixtures in Eq. equation 7 with $\kappa > 0$ and $r, \widehat{r} \in [0, 1 - \eta]$ for $\eta \in (0, 1)$. Then $p_R = \mathcal{R}_\kappa(p_D, p_A, r)$. For $z = \mathcal{R}_\kappa(q_D, q_A, \widehat{r})$,*

$$\|z - p_R\|_1 \le \underbrace{\frac{1 + \kappa}{\kappa\eta}\epsilon_D + \frac{1 + \kappa(1 - \eta)}{\kappa\eta}(\epsilon_A + \epsilon_{\text{align}}) + \frac{(1 + \kappa)d_{DA}}{\kappa\eta^2}\epsilon_r}_{B(h)}. \tag{10}$$

For a repair map $\mathcal{S}$ returning a probability vector,

$$\|\mathcal{S}(z) - q_R\|_1 \le B(h) + \|\mathcal{S}(z) - z\|_1 + \epsilon_R. \tag{11}$$

Linearity separates model error from ratio error, controlled by $\partial_r \mathcal{R}_\kappa$ (Appendix C.7). Recovery becomes sensitive as $r \to 1$ or $\kappa \to 0$. The margin must constrain both the source posterior and the decoder state. Equation equation 11 includes repair distortion and the ideal-to-retrained discrepancy.

3.2 Likelihood Training and Predictive Error

For a specified target law, excess NLL equals average forward KL and bounds average predictive error; transfer to evaluation contexts requires coverage (Proposition C.2, Appendix C.6). Empirical loss alone does not control unseen contexts or characterize the combined NLL/reverse-KL population optimum, which is addressed separately by Theorem 5.1.

3.3 Recursive Estimation and Stabilization

Let $\ell_i = p_A(y_i \mid h_i)/p_D(y_i \mid h_i) > 0$ and $\widehat{\ell}_i = q_A(y_i \mid h_i)/q_D(y_i \mid h_i) > 0$ be ideal and neural token likelihood ratios, and let $\widetilde{r}_t$ be the unprojected decoder state initialized at the BOS-prefill output $a(x)$. On common positive support, initialization and local log-ratio errors accumulate additively in log-state space, retaining cross-step dependence. Exponentiation can introduce state bias even from centered log errors. Proposition C.3 in Appendix C.8 gives the exact identity, state-error bound, and moment statements; the appendix also gives a concentration bound under a conditional error model.

A concrete finite-sample regime. Appendix C.9 establishes ratio error $O_p(N_D^{-1/2}+N_A^{-1/2})$ for smoothed empirical predictors from N_D, N_A independent ideal factual/amplified sequences in a finite sequence space. It assumes known initial source posterior and $\kappa > 0$, fixed sample-independent test prefixes with factual probability at least $m > 0$, and terminal projection onto $[0, b]$ containing the source posteriors, with $b < 1$; projection is not fed back into the recursion. Constants worsen for rare prefixes. The construction and moment bounds are given there; these guarantees do not establish convergence of LLM fine-tuning.

The strict-margin recursion. The reference algorithm projects the raw decoder-state update F in Eq. equation 8 onto $[0, 1-\eta]$ at each step. For a finite likelihood-ratio bound L and local error bounds $\delta_t \geq 0$, if the source posteriors stay in this interval, $\ell_t \leq L$, and $|\widehat{\ell}_t - \ell_t| \leq \delta_t$, then

$$|\widehat{r}_{t+1} - r_{t+1}| \leq L|\widehat{r}_t - r_t| + \frac{1+\kappa(1-\eta)}{\kappa}\delta_t. \tag{12}$$

At fixed horizons, vanishing initialization and local likelihood errors imply decoder-state consistency for feasible source posteriors. Stepwise projection breaks the product identity; Appendix C.10 unrolls this bound and accounts for source posteriors exceeding the cap.

4 CURE: Counterfactual Recovery by Ratio Estimation

CURE combines controlled amplification with BOS-based ratio estimation. Let $\Gamma_t = p_A(x, y_{<t})/p_D(x, y_{<t})$ be the joint prompt–response prefix density ratio and $\Lambda_t = p_A(y_{<t} \mid x)/p_D(y_{<t} \mid x)$ its response-conditional counterpart. Joint probabilities condition on the shared BOS marker.

Theorem 4.1 (Calibrated CURE Recovery)**.** *Assume shared-source mixture and controlled amplification for the joint prompt–response laws in Eq. equation 25, with known $\kappa > 0$ and BOS source posterior $\alpha_0 = \Pr(S = F \mid \mathrm{BOS})$. On positive full-data prefix support:* (i) Recover the retain prediction. *Whenever $r_t < 1$,*

$$p_R = \mathcal{R}_\kappa(p_D, p_A, r_t) = \frac{(1+\kappa)p_D - (1+\kappa r_t)p_A}{\kappa(1-r_t)}. \tag{13}$$

(ii) Identify the source posterior from BOS. *The ideal recursion starts at α_0 and applies F to each observed prompt token and selected response token using its ideal likelihood ratio. After prompt processing, $r_1 = \alpha(x)$ is a derived source posterior. For response prefixes,*

$$1 + \kappa r_t = (1+\kappa\alpha_0)\,\Gamma_t = (1+\kappa\alpha(x))\Lambda_t, r_{t+1} = \frac{(1+\kappa r_t)\ell_t - 1}{\kappa},\ \ell_t = \frac{p_A(y_t \mid h_t)}{p_D(y_t \mid h_t)}. \tag{14}$$

The update holds for every selected token with positive factual probability, regardless of the token-selection policy. Together, these equations identify the source posterior and recover p_R at each supported prefix with $r_t < 1$ from the two BOS-conditioned observation laws and their BOS mixture parameter.

Proof sketch. Eliminating p_F gives Eq. equation 13. Marginalizing the joint amplification law gives the BOS prefix identity; conditioning on x gives its response counterpart. The chain rule gives the same token update through prompt and response (Appendix C.3).

The unlearning operation. The recovery rule subtracts the amplified shift with the identified multiplier:

$$p_R = p_D - \underbrace{\frac{1+\kappa r_t}{\kappa(1-r_t)}}_{\gamma_\kappa(r_t)} \underbrace{(p_A - p_D)}_{\text{amplification shift}} . \tag{15}$$

Section 5 implements this correction and the state update with neural predictors; Theorem 3.1 bounds approximation error relative to the retrained predictor q_R.

BOS initialization and source-posterior interpretation. CURE obtains $a(x)$ by prefilling the prompt from the training-derived BOS state a_0 (Eq. equation 19). With ideal response likelihood ratios, the unprojected decoder state satisfies

$$\widetilde{r}_t - r_t = (a(x) - \alpha(x))\Lambda_t. \tag{16}$$

The decoder state equals the source posterior when $a_0 = \alpha_0$, both neural models realize the joint observation laws, and clipping leaves the ideal recursion unchanged. Appendix B details the correspondence and approximation errors; Section 5 gives the implemented prefill.

5 Implementing CURE

CURE implements the recovery and update operations of Theorem 4.1 using frozen q_D, a trained auxiliary q_A approximating p_A, and decoder state $\widehat{r}_t$. Figure 2 shows the workflow; Appendix E.1 gives the strict-margin reference decoder (Algorithm 1) and distinguishes it from the reported implementation. The intervention requires this decoder.

The CURE decoding rule. Choose a margin $\eta \in (0,1)$ and restrict $\widehat{r}_t$ to $[0, 1-\eta]$. At each context, use normalized, untempered probabilities to compute

$$z_t = \mathcal{R}_\kappa(q_D, q_A, \widehat{r}_t) = q_D - \gamma_\kappa(\widehat{r}_t)(q_A - q_D). \tag{17}$$

Since z_t sums to one but can have negative entries, we shift and renormalize:

$$\zeta_t = \max\{0, -\min_v z_t(v)\}, \qquad p_{\mathrm{CURE}}(v \mid h_t) = \frac{z_t(v) + \zeta_t}{1 + |\mathcal{V}|\zeta_t}. \tag{18}$$

Repair preserves token ordering but changes sampling probabilities; its distortion (Eq. equation 52) enters Theorem 3.1's bound relative to q_R. Under imperfect alignment, $\widehat{r}_t = 0$ gives $q_D + (q_D - q_A)/\kappa$, so returning q_D requires an explicit fallback.

BOS initialization and decoding. CURE determines its BOS decoder state from the auxiliary fine-tuning configuration as

$$a_0 := \frac{\beta}{\kappa\lambda}, \qquad \lambda > 0, \quad 0 \le \beta \le \kappa\lambda, \tag{19}$$

Here β, λ weight negative log-likelihood (NLL) and reverse KL in Eq. equation 20. Initialize at $\mathrm{clip}(a_0, 0, 1-\eta)$ and process the teacher-forced prompt with the same clipped state update used during generation, yielding $a(x)$ without response scoring (Appendix B). Each response step corrects and repairs probabilities, selects a token, then applies F with that token's auxiliary-to-factual likelihood ratio and clips the next state to $[0, 1-\eta]$; beams keep separate states and scores. Use $\mathrm{clip}(s, l, u) = \min\{u, \max\{l, s\}\}$ and ratio temperature $\tau = 1$; apply generation temperature only after recovery and repair.

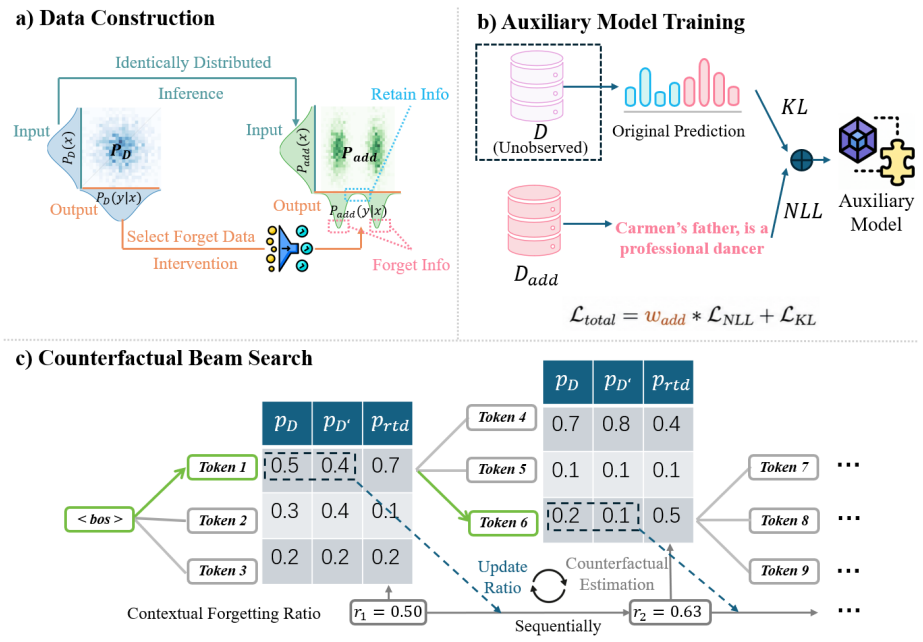


Figure 2: **Overview of CURE.** (a) Construct forget-related data with desired input alignment. (b) Train the auxiliary with likelihood and KL regularization. (c) Correct token probabilities and update each beam's decoder state. Displayed forgetting ratios denote decoder states; scores and states are illustrative; Algorithm 1 specifies the reference decoding rule. Appendix E.1 explains historical labels and alignment limitations.

Construct the auxiliary observation. Adapt q_A to forget-related examples supplied with the request or generated and filtered (Appendix E.1). The target is Eq. equation 6; exact population realization requires the calibration conditions in Theorem 5.1.

For empirical auxiliary context–token distribution $\mathcal{D}_{\text{aux}}$ over (h, v), regularization context law μ_{aux}, weights $\beta \geq 0$ and $\lambda > 0$, and Kullback–Leibler divergence D_{KL}, train with

$$\mathcal{L}_{\text{aux}} = \beta \mathbb{E}_{(h,v)\sim\mathcal{D}_{\text{aux}}}[-\log q_A(v \mid h)] + \lambda \mathbb{E}_{h\sim\mu_{\text{aux}}} D_{\text{KL}}(q_A(\cdot \mid h) \| q_D(\cdot \mid h)). \tag{20}$$

Write the population auxiliary law as $\nu_{\text{aux}}(h)\pi_{\text{aux}}(v \mid h)$, with local weights $\beta_h = \beta\nu_{\text{aux}}(h)$ and $\lambda_h = \lambda\mu_{\text{aux}}(h)$. These account for different NLL and regularization context laws.

Theorem 5.1 (Controlled amplification via calibrated reverse-KL training)**.** *Fix a context with $q_D = p_D > 0$ and $\beta_h, \lambda_h > 0$. Let p_A be the controlled conditional in Eq. equation 7 and $K_h = D_{\text{KL}}(p_A \| p_D)$. Over the token probability simplex,*

$$J_h(q) = \beta_h \mathbb{E}_{\pi_{\text{aux}}(\cdot|h)}[-\log q(v)] + \lambda_h D_{\text{KL}}(q \| p_D) \tag{21}$$

has unique minimizer p_A if and only if

$$\pi_{\text{aux}}(v \mid h) = \pi_h^\star(v) := p_A(v) \left[1 + \frac{\lambda_h}{\beta_h}\left(\log \frac{p_A(v)}{p_D(v)} - K_h\right)\right]. \tag{22}$$

This vector is a probability law exactly when

$$\beta_h/\lambda_h \geq K_h - \min_v \log[p_A(v)/p_D(v)]. \tag{23}$$

With coverage and simultaneous realization of these optima at every supported prefix of a common finite horizon, the auxiliary sequence law equals Eq. equation 6.

Table 1: Main TOFU results (percentage scale). Retain is trained without the forget set. Fgt/Rtd: forget/retain ROUGE-L; M.U.: Model Utility; F.Q.: the KS-test p-value multiplied by 100.

	TOFU-1%				TOFU-5%				TOFU-10%			
Method	Fgt↓	Rtd↑	M.U.↑	F.Q.↑	Fgt↓	Rtd↑	M.U.↑	F.Q.↑	Fgt↓	Rtd↑	M.U.↑	F.Q.↑
Original	99.8	99.5	62.7	1.43	98.7	99.5	62.7	5×10^{-17}	99.1	99.5	62.8	3×10^{-26}
Retain	42.7	98.9	63.2	100	39.2	98.9	63.2	100	39.6	99.2	64.6	100
GA	77.7	99.0	63.1	0.67	0.31	0.49	0.00	10^{-117}	0.30	0.49	0.00	10^{-237}
DPO	3.90	30.3	43.8	57.8	4.01	3.94	1.88	5.21	3.12	3.41	0.00	3×10^{-13}
NPO	45.7	70.9	59.7	40.4	52.2	51.6	17.1	54.5	19.1	14.8	0.00	5×10^{-4}
SimNPO	76.9	97.8	69.5	0.22	69.3	97.4	70.8	2.6×10^{-12}	66.7	96.9	68.8	1.3×10^{-11}
RMU	20.2	99.3	68.9	1.63	16.2	99.5	69.8	2.53	8.1	99.6	69.8	11.1
ULD	58.9	46.2	57.4	10^{-4}	61.7	60.5	57.4	10^{-19}	55.0	60.0	60.6	2×10^{-36}
FLAT-TV	51.1	83.8	69.1	2.9	0.7	0.6	0.0	3.8×10^{-62}	0.6	0.6	0.0	4.4×10^{-126}
ASU-KL	43.5	67.3	66.4	40.5	36.8	51.6	63.7	0.9	33.9	71.83	66.81	0.058
CURE	46.7	99.5	80.4	99.0	38.1	93.4	79.4	86.6	38.3	98.4	81.4	99.4

All methods are evaluated under the common protocol of Appendix E.3, starting from the same Llama-3.1-8B-Instruct checkpoint, so the overview is protocol-matched across methods. FLAT-TV and ASU-KL use seed-0 runs at their fixed final checkpoints. Across-seed uncertainty is not available.

Table 2: MUSE results. Know/Verb measure knowledge/verbatim memorization; Util measures retained utility (all three: ROUGE-L F1 $\times100$). Priv is signed relative ROC-AUC deviation from the retrain reference (Eq. equation 69); closer to zero is preferable.

	MUSE-News				MUSE-Books			
Method	Know↓	Verb↓	Priv.→ 0	Util.↑	Know↓	Verb↓	Priv.→ 0	Util.↑
Original	63.9	56.4	-99.81	54.8	45.88	99.70	-57.15	69.40
Retrain	31.9	20.2	-0.02	55.7	30.14	13.89	-0.04	69.05
GA	0.00	0.00	$+41.22$	0.00	0.00	0.00	-19.99	0.00
NPO	36.75	15.10	-65.59	33.00	13.04	18.09	-51.59	39.70
SimNPO-GD	47.09	13.90	$+11.90$	40.31	0.00	0.00	-21.60	48.21
ULD	61.66	56.62	-99.85	51.28	39.30	94.46	-53.87	59.98
FLAT-TV	0.00	0.00	$+3.04$	0.00	0.56	0.72	-47.90	0.58
ASU-KL	49.15	4.57	$+108.28$	50.83	14.23	5.04	-42.28	59.73
CURE	10.4	11.4	-8.61	52.1	12.01	9.64	-26.5	69.53

All evaluations use the common protocol of Appendix E.4 (100 examples per generation metric; 100 privacy texts per group). GA, NPO, ASU-KL, and ULD use seed-0 runs; SimNPO-GD uses the released author checkpoints; Original and Retrain are the released reference models; CURE uses its standard configuration. ULD uses a fixed logit coefficient -0.3.

The proof is in Appendix C.2, with approximation errors analyzed in Section 3. The theorem concerns the combined training objective; Proposition C.2 concerns error for a specified likelihood target. Equation equation 19 alone neither ensures data calibration nor identifies a_0 with the BOS source posterior α_0.

6 Experiments

Tables 1 and 2 summarize TOFU and MUSE results. The baseline protocols are in Appendices E.3 and E.4.

6.1 Behavior Relative to the Retain-Only Target

TOFU evaluates removal of synthetic author biographies (Maini et al., 2024). Table 1 covers all three forget splits. CURE's forget ROUGE-L is 46.7/38.1/38.3 against the retain-only reference's 42.7/39.2/39.6; retain ROUGE-L is 99.5/93.4/98.4 against 98.9/98.9/99.2. On 10%, both generation scores are close to the reference, whereas the 5% split exposes a 5.5-point retain-score deficit despite high MU. These aggregate scores do not establish equality of the next-token distributions. FQ is a Kolmogorov–Smirnov (KS) test p-value: a high value indicates failure to reject the null, not predictive equivalence or a fraction erased. On 10%, GA and DPO collapse retained utility, while RMU preserves retain overlap with lower FQ. Under the recorded baseline protocol, FLAT-TV also loses utility on 5% and 10%, and ASU-KL has substantial retention costs (Appendix E.3).

MUSE tests memorization, privacy deviation, and retained utility (Shi et al., 2024). Table 2 reports News and Books evaluations for all methods. CURE's retained utility is 52.1 versus Retrain's 55.7 on News and 69.53 versus 69.05 on Books. Both memorization scores fall below their respective retrain values; stronger suppression therefore does not by itself imply closer recovery of the reference behavior. Signed privacy deviations of -8.61 on News and -26.5 on Books also indicate residual disagreement with the retrain reference. Relative to ULD, CURE improves utility while reducing both memorization scores on both domains. FLAT-TV shows that low memorization can accompany almost complete utility loss; News ASU-KL retains utility but has a large privacy deviation. On Books, SimNPO-GD suppresses memorization at a retention cost. These comparisons describe the recorded decoding configurations; testing the recovery target further requires common-prefix distribution measurements and matched decoders (Appendix E.4). Qwen3-8B and toxicity results appear in Appendix E.5.

6.2 Contextual Correction and Auxiliary Sensitivity

In two TOFU examples, adaptive control attains retain/forget ROUGE-L of 100.0/23.8, versus 50.0/90.5 for a linear schedule and 12.5/76.2 for a sinusoidal schedule. These cases illustrate context sensitivity, not a test-set ablation (Figure 4; Appendix E.11). Varying auxiliary optimization weight β from 0.1 to 0.9 yields retain ROUGE-L 98.33–98.40 and forget ROUGE-L 35.78–39.23 (Table 9); amplification κ is held fixed at its specified value throughout the sweep.

6.3 Imperfect Approximation and Cost

The input-dependent decoder state $a(x)$ is computed by BOS prefill as described in Section 5; identifying it with the theoretical source posterior requires the conditions in Appendix B. Auxiliary-convergence results appear in Table 7. On TOFU-10%, CURE uses 7 minutes for preparation and 50 minutes/54 GB for inference, versus GA's 62 minutes and 35 minutes/15 GB. Its shorter preparation trades off against inference cost (Appendices E.7 and E.9). The executed controlled-sequence experiment in Appendix D tests recovery against known source posteriors and retain laws.

7 Discussion and Conclusion

This work frames retain-only prediction as an identification problem. Bidirectional Mixture Identification connects retain recovery to source-posterior identification; under the stated mixture and calibration assumptions, controlled amplification and BOS-based prefix ratios yield a constructive recovery rule. We characterize when auxiliary training realizes the required observation and bound realization, alignment, state-estimation, and repair errors. CURE implements this rule through a deployed decoder with frozen factual parameters and state updates through prompts and responses. A controlled-sequence experiment tests recovery with known source laws, and TOFU and MUSE evaluations characterize forgetting–utility trade-offs.

Future work includes improving auxiliary calibration and state estimation for rare prefixes and weak retain support. Common-prefix distribution measurements, matched decoding, and broader privacy evaluations offer directions for assessing distributional recovery and refining CURE's practical behavior.

AI USE STATEMENT

Large language model tools were used solely for language polishing of the manuscript, including grammar correction, phrasing improvements, and formatting consistency checks. All technical content, mathematical derivations, experimental design, implementation, data analysis, and scientific conclusions were developed and verified entirely by the authors. No AI tool was used to generate research ideas, methods, results, figures, or tables, and the authors take full responsibility for the accuracy and integrity of the entire paper.

ETHICS STATEMENT

The intended use is to reduce unwanted predictive influence while retaining useful behavior. CURE leaves the factual checkpoint unchanged, and its protection depends on use of the specified decoder. A forget-aligned auxiliary may concentrate sensitive or copyrighted information and requires access controls consistent with those applied to the original data.

REPRODUCIBILITY STATEMENT

The main text states the model assumptions, auxiliary objective, and decoding rule. Complete proofs and statistical scope appear in Appendices B and C. Appendix D gives the controlled experiment's distributions, estimator, seed, trial count, and uncertainty calculation; its script and numerical outputs accompany the source. The approximate-calibration checks are provided in `experiments/theory_checks`. Appendices E.1, E.3, and E.4 record the LLM configuration and benchmark protocols. Appendix F collects supplementary robustness and implementation measurements and identifies their remaining protocol gaps.

REFERENCES

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Semi-supervised novelty detection. *Journal of Machine Learning Research*, 11(99):2973–3009, 2010. URL http://jmlr.org/papers/v11/blanchard10a.html.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 141–159, 2021. doi: 10.1109/SP40001.2021.00019.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning in llms. *arXiv preprint arXiv:2310.02238*, 2023.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for llm unlearning, 2025. URL https://arxiv.org/abs/2410.07163.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 3356–3369, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.301. URL https://aclanthology.org/2020.findings-emnlp.301/.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models. *arXiv preprint arXiv:2210.01504*, 2022.

Jiabao Ji, Yujian Liu, Yang Zhang, Gaowen Liu, Ramana Rao Kompella, Sijia Liu, and Shiyu Chang. Reversing the forget-retain objectives: An efficient llm unlearning framework from logit difference. *arXiv preprint arXiv:2406.08607*, 2024.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. The wmdp benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218*, 2024.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. Dexperts: Decoding-time controlled text generation with experts and anti-experts. *arXiv preprint arXiv:2105.03023*, 2021.

Alisa Liu, Xiaochuang Han, Yizhong Wang, Yulia Tsvetkov, Yejin Choi, and Noah A Smith. Tuning language models by proxy. *arXiv preprint arXiv:2401.08565*, 2024.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*, 2024.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. In-context unlearning: Language models as few shot unlearners. *arXiv preprint arXiv:2310.07579*, 2023.

Clayton Scott. A Rate of Convergence for Mixture Proportion Estimation, with Application to Learning from Noisy Labels. In Guy Lebanon and S. V. N. Vishwanathan (eds.), *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pp. 838–846, San Diego, California, USA, 09–12 May 2015. PMLR. URL https://proceedings.mlr.press/v38/scott15.html.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. Muse: Machine unlearning six-way evaluation for language models, 2024. URL https://arxiv.org/abs/2407.06460.

Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

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

Yilun Zhu, Aaron Fjeldsted, Darren Holland, George Landon, Azaree Lintereur, and Clayton Scott. Mixture proportion estimation beyond irreducibility. In *International Conference on Machine Learning*, pp. 42962–42982. PMLR, 2023.

A Relation to Existing Unlearning and Decoding Methods

Mixture-proportion estimation. Inversion given a known component is elementary mixture algebra. Classical mixture-proportion estimation (MPE) asks whether a mixture and one component identify the generating weight: generally they do not; irreducibility equates that weight with the maximal feasible proportion (Blanchard et al., 2010). Convergence rates require additional distributional conditions (Scott, 2015), while posterior side information can relax irreducibility (Zhu et al., 2023). Our result relates the contextual weight to the retain prediction conditional on the ideal forget-source law. Knowing this distribution is stronger than possessing finite forget examples. The auxiliary supplies an operational observation; it does not eliminate amplification or BOS source-posterior calibration requirements.

Parameter-update baselines and evaluation. NPO uses a negative-preference objective to balance forgetting and utility (Zhang et al., 2024); SimNPO removes its reference-model dependence (Fan et al., 2025). RMU instead acts on model representations (Li et al., 2024). These objectives change the target checkpoint, whereas CURE adapts an auxiliary and intervenes in prediction. This architectural difference does not imply superiority: Table 1 compares complete methods, including different generation rules. The decoding ablations in Appendix F.3 isolate some within-CURE effects, but do not replace a cross-method comparison at matched search budgets.

Inference-time unlearning. Inference-time unlearning includes in-context interventions and auxiliary-model decoding (Pawelczyk et al., 2023; Ji et al., 2024). DExperts adds an expert-minus-anti-expert logit contrast to base-model logits (Liu et al., 2021); proxy tuning transfers a small model's adaptation through a prediction contrast (Liu et al., 2024); unlearning from logit difference (ULD) subtracts assistant-model logits (Ji et al., 2024). At a fixed context h, a two-model logit contrast with constant coefficient $\gamma_{\mathrm{logit}} \geq 0$ has the normalized form

$$p_{\mathrm{logit}}(v \mid h) = \frac{q_D(v \mid h) q_A(v \mid h)^{-\gamma_{\mathrm{logit}}}}{\sum_{u \in \mathcal{V}} q_D(u \mid h) q_A(u \mid h)^{-\gamma_{\mathrm{logit}}}}. \tag{24}$$

Here u is a vocabulary summation index and q_A is strictly positive throughout $\mathcal{V}$. The fixed coefficient γ_{logit} is distinct from CURE's context-dependent probability-space multiplier $\gamma_\kappa(r_t)$ in Eq. equation 15.

CURE inverts a probability-space mixture with coefficients determined by a decoder state. Its raw output can be signed under mismatch, making probability repair part of the error analysis. Density-ratio estimation supplies the statistical perspective (Sugiyama et al., 2012); CURE uses prefix likelihoods to estimate contextual source posteriors. Throughout, "counterfactual" refers to dataset removal. Because CURE leaves the base checkpoint unchanged, we assess predictions under its decoder without claiming parameter erasure.

B BOS Initialization and Additional Calibration Details

B.1 Identifying the Source Posterior by BOS Prefill

The prompt is itself an observed prefix of the token stream. Write $x = (x_1, \ldots, x_m)$ for the complete serialized prompt after a shared BOS marker, including any template and system tokens. All probabilities below condition on this marker, which is not itself scored. Let $\alpha_0 = \Pr(S = F \mid \mathrm{BOS})$ be the source posterior at BOS, before any prompt token is read. To identify $\alpha(x)$ from prompt likelihoods, extend the conditional model to joint prompt–response laws:

$$\begin{aligned} p_D(x, y) &= (1 - \alpha_0) p_R(x, y) + \alpha_0 p_F(x, y), \\ p_A(x, y) &= \frac{p_D(x, y) + \kappa \alpha_0 p_F(x, y)}{1 + \kappa \alpha_0}. \end{aligned} \tag{25}$$

This assumption specifies amplification on prompt prefixes as well as responses; the conditional laws alone do not specify the prompt marginals.

Proposition B.1 (Source-posterior identification by BOS prefill)**.** *Under Eq. equation 25, let x satisfy $p_D(x) > 0$. Then*

$$\alpha(x) = \frac{\alpha_0 p_F(x)}{p_D(x)} = \frac{(1+\kappa\alpha_0)p_A(x)/p_D(x) - 1}{\kappa}. \tag{26}$$

Equivalently, initialize the ideal source-posterior recursion at $s_0 = \alpha_0$ and read the observed prompt tokens using

$$s_j = \frac{(1+\kappa s_{j-1})\, p_A(x_j \mid x_{<j})/p_D(x_j \mid x_{<j}) - 1}{\kappa}, \quad j = 1, \ldots, m. \tag{27}$$

Then $s_j = \Pr(S = F \mid x_{\le j})$, so the source posterior at the start of the response is $r_1 = s_m = \alpha(x)$. Conditioning the joint laws on x recovers Eqs. equation 4 and equation 6.

Proof sketch. Marginalizing Eq. equation 25 to each prompt prefix gives $(1+\kappa\alpha_0)p_A(x_{\le j})/p_D(x_{\le j}) = 1 + \kappa s_j$. The chain rule gives the token update (Appendix C.4). Thus the ideal recursion obtains the source posterior $\alpha(x)$ by prefill rather than taking it as a separate input; the practical decoder substitutes its fixed BOS hyperparameter and neural likelihoods as described below.

The BOS source posterior and decoder initialization. Under a uniform empirical mixture over training sequences, $\alpha_0 = |\mathcal{D}_f|/|\mathcal{D}|$. If sequence i has known sampling mass $\omega_i \ge 0$ with positive total, the corresponding empirical source posterior at BOS is $\sum_{i:\, d_i \in \mathcal{D}_f} \omega_i / \sum_{i:\, d_i \in \mathcal{D}} \omega_i$. These identities require the source model to use that sampling unit and weighting. In the implementation, the decoder state starts at the fixed hyperparameter a_0, shared across prompts in a configuration. The exact source-posterior interpretation requires $a_0 = \alpha_0$; a common BOS marker by itself does not establish this equality. At a nonempty prompt, $\alpha(x)$ generally differs from α_0.

Neural initialization and its error. Algorithm 1 substitutes q_D, q_A, a_0 into the BOS prefill and uses the resulting decoder state $a(x)$ to initialize generation. It clips the decoder state during both prefill and generation; in the calibrated ideal case, exact equality to the source posterior holds when that posterior stays within the clipping interval, with projection effects treated in Appendix C.10. Prompt scoring requires the same tokenization and serialization in both models, and auxiliary alignment on prompt prefixes. Response-only auxiliary training does not establish this alignment merely because both models accept a BOS token.

The resulting initialization error propagates according to Eq. equation 16. Appendix C.4 decomposes it into BOS-initialization and prompt-likelihood errors.

C Proofs and Statistical Scope

Throughout, logarithms are natural, the token vocabulary $\mathcal{V}$ is finite, and vector norms are L_1 unless otherwise specified. The vector $\mathbf{1}$ has one entry equal to one for each token in $\mathcal{V}$. The source-posterior identification argument uses the joint prompt–response laws conditioned on BOS. For conditional probability calculations at a fixed input x, we write $\alpha = \alpha(x)$ and suppress the context in probability vectors. Prefix identities require positive full-data probability; log-likelihood identities require positive probabilities under both models along the path. Conditional kernels on zero-mass component prefixes may be chosen arbitrarily because their mixture weights are zero.

C.1 Proof of Theorem 2.1 and Source-Weight Ambiguity

At a fixed context, write $r = r_t$. In the source mixture of Eq. equation 1, p_D and p_F are supplied. Rearranging gives

$$(1 - r)p_R = p_D - rp_F.$$

For $0 \le r < 1$, division by $1 - r$ proves Eq. equation 2 and uniqueness of p_R. Conversely,

$$p_D - p_R = r(p_F - p_R).$$

If $p_F \neq p_R$, taking L_1 norms and using $r \ge 0$ gives Eq. equation 3, including $r = 0$. These norm ratios identify the weight under the stated mixture identity; a norm ratio alone does not certify that arbitrary supplied distributions satisfy that identity. At $r = 1$, the retain component is absent and cannot be identified from this mixture. If $p_F = p_R$, then $p_D = p_F$ and the weight is not identified.

Knowing the forget law does not generally identify its weight. The same supplied pair $p_D = (1/2, 1/2)$ and $p_F = (1, 0)$ admits both $(r, p_R) = (1/4, (1/3, 2/3))$ and $(1/2, (0, 1))$. More generally, on the finite vocabulary define

$$r_{\max} = \min_{v:\, p_F(v) > 0} \frac{p_D(v)}{p_F(v)}.$$

The set in this minimum is nonempty and $0 \le r_{\max} \le 1$. For $0 \le r < 1$, the vector $(p_D - rp_F)/(1 - r)$ sums to one and is nonnegative exactly when $r \le r_{\max}$. Thus every $r \in [0, r_{\max}] \cap [0, 1)$ gives a feasible retain distribution. If $r_{\max} = 0$, only $r = 0$ is feasible. If $r_{\max} = 1$, then $p_D = p_F$ and all feasible weights give the same retain law $p_R = p_F$.

For $r_{\max} < 1$, an additional irreducibility assumption can select the maximal feasible weight (Blanchard et al., 2010; Scott, 2015; Zhu et al., 2023). In this finite vocabulary, irreducibility of p_R with respect to p_F means

$$\min_{v:\, p_F(v) > 0} \frac{p_R(v)}{p_F(v)} = 0.$$

Indeed, the source mixture gives

$$r_{\max} = r + (1 - r) \min_{v:\, p_F(v) > 0} \frac{p_R(v)}{p_F(v)},$$

so this restriction implies $r = r_{\max}$. It is an additional structural assumption, not a consequence of supplying p_F or observing a finite forget set.

Direct posterior recursion when the source law is known. Under Eq. equation 4, known $\alpha(x)$ and the sequence laws p_D, p_F give Eq. equation 5 directly. On any path with $p_D(y_{\le t} \mid x) > 0$, factoring the prefix probabilities yields

$$r_{t+1} = r_t \frac{p_F(y_t \mid h_t)}{p_D(y_t \mid h_t)}, \qquad r_1 = \alpha(x). \tag{28}$$

Here $y_{\le t} = (y_1, \ldots, y_t)$. If a forget-source prefix has zero probability, its source posterior is zero and its conditional kernel may be chosen arbitrarily. Thus a known source posterior and a known forget law permit direct tracking without an auxiliary observation; this comparison does not make $\alpha(x)$ an input to CURE, which computes its decoder state by BOS prefill.

C.2 Proof and Scope of Theorem 5.1

Fix h and abbreviate $\pi = \pi_{\text{aux}}(\cdot \mid h)$, $a = p_A$, $d = p_D$, $b = \beta_h > 0$, and $l = \lambda_h > 0$. The finite-vocabulary assumption and $d > 0$ imply $a > 0$. The population objective at this context is

$$J(q) = -b \sum_v \pi(v) \log q(v) + l \sum_v q(v) \log \frac{q(v)}{d(v)}.$$

It has a minimizer on the probability simplex, using the usual extended-value convention for cross-entropy. A minimizing coordinate cannot be zero: if $\pi(v) > 0$, the loss is infinite; if $\pi(v) = 0$, moving positive mass into that coordinate has a negative infinite one-sided entropy derivative. On the positive simplex, the Hessian is diagonal with entries $b\pi(v)/q(v)^2 + l/q(v) > 0$. Thus the minimizer is unique, and an interior probability vector is the minimizer exactly when its gradient is constant across coordinates.

At $q = a$, stationarity with normalization multiplier ξ is

$$-\frac{b\pi(v)}{a(v)} + l\left(\log \frac{a(v)}{d(v)} + 1\right) + \xi = 0. \tag{29}$$

Multiplying by $a(v)$ and summing over v gives $\xi = b - l(1 + K_h)$. Substitution yields

$$\pi(v) = a(v)\left[1 + \frac{l}{b}\left(\log \frac{a(v)}{d(v)} - K_h\right)\right],$$

which proves both directions of Eq. equation 22. Its sum is $1 + (l/b)(K_h - K_h) = 1$. Since $a(v) > 0$, it is nonnegative exactly when $b/l \geq K_h - \min_v \log(a(v)/d(v))$. This proves feasibility. In particular, for any positive target a, a sufficiently large finite b/l makes the calibrated data law feasible. If $a = d$, the right-hand side of the feasibility bound is zero and $\pi_h^\star = d$ for all positive weights.

Applying this argument at every supported context identifies every conditional of the amplified sequence law. The autoregressive chain rule then identifies the sequence law itself, assuming a common finite horizon (with EOS padding) or almost-sure termination. This step requires training coverage at every relevant prefix and simultaneous realizability of these conditionals; a result restricted to observed contexts does not determine unseen continuations.

Why the calibrated data are forget-related. For $\rho = \kappa r_t > 0$, set $s(v) = p_F(v)/d(v)$ and $z(v) = a(v)/d(v) = (1 + \rho s(v))/(1 + \rho)$. Equation equation 22 gives $\pi_h^\star(v)/d(v) = g(z(v))$, where

$$g(z) = z\left[1 + \frac{l}{b}(\log z - K_h)\right], \qquad g'(z) = 1 + \frac{l}{b}(\log z - K_h + 1) > 0$$

on the interval between the smallest and largest $z(v)$. The last inequality follows because feasibility makes the bracket in $g(z)$ nonnegative at the smallest z, and it increases thereafter. Since z increases with s, the calibrated data-to-factual density ratio increases with forget enrichment. This is a relative enrichment property; the calibrated data may include tokens with zero forget-source mass and need not be samples from p_F alone.

Excess reverse-KL training loss controls realization error. Under calibration, for any positive probability vector q, subtracting the objective and canceling its stationary linear term gives the exact identity

$$J_h(q) - J_h(p_A) = \beta_h \sum_v \pi_h^\star(v)\left[\frac{q(v)}{p_A(v)} - 1 - \log \frac{q(v)}{p_A(v)}\right] + \lambda_h D_{\text{KL}}(q \| p_A). \tag{30}$$

The bracket is nonnegative because $u - 1 - \log u \geq 0$ for $u > 0$. Pinsker's inequality therefore yields

$$\|q - p_A\|_1 \leq \sqrt{\frac{2[J_h(q) - J_h(p_A)]}{\lambda_h}}. \tag{31}$$

This is a population, fixed-context guarantee under the theorem's calibration and exact-reference assumptions. It does not turn small empirical loss into a guarantee at unobserved prefixes or ensure that a restricted neural family attains p_A.

Proposition C.1 (Approximate calibration with a deployed reference)**.** *Fix a context with $\beta_h, \lambda_h > 0$, a probability vector $\pi = \pi_{\mathrm{aux}}(\cdot \mid h)$, and positive p_A, q_D. Let $p_A^\dagger$ minimize*

$$J_h^{q_D}(q) = -\beta_h \sum_v \pi(v) \log q(v) + \lambda_h D_{\mathrm{KL}}(q \| q_D).$$

Define the target stationarity residual, up to a constant, by

$$g_h(v) = -\frac{\beta_h \pi(v)}{p_A(v)} + \lambda_h \left(\log \frac{p_A(v)}{q_D(v)} + 1 \right). \tag{32}$$

Then

$$\left\| p_A^\dagger - p_A \right\|_1 \leq \min \left\{ 2, \frac{\mathrm{osc}(g_h)}{2\lambda_h} \right\}. \tag{33}$$

For any positive q_A with excess loss $\Delta_h = J_h^{q_D}(q_A) - J_h^{q_D}(p_A^\dagger)$ over this minimizer,

$$\|q_A - p_A\|_1 \leq \sqrt{2\Delta_h / \lambda_h} + \frac{\mathrm{osc}(g_h)}{2\lambda_h}. \tag{34}$$

Here $\mathrm{osc}(g) = \max_v g(v) - \min_v g(v)$. When the calibrated law $\pi_h^\star$ of Theorem 5.1 is feasible, there is a scalar c_h such that

$$g_h(v) = c_h + \beta_h \frac{\pi_h^\star(v) - \pi(v)}{p_A(v)} + \lambda_h \log \frac{p_D(v)}{q_D(v)}. \tag{35}$$

Thus the residual separates auxiliary-law and reference-model mismatch.

Proof. The preceding existence and strict-convexity argument applies with q_D as reference, giving a unique positive minimizer. Write $a = p_A$, $q^* = p_A^\dagger$, $b = \beta_h$, and $l = \lambda_h$. For any positive probability vector q, direct subtraction of the gradients gives

$$\begin{aligned}\langle \nabla J(q) - \nabla J(a), q - a \rangle = b \sum_v \frac{\pi(v)[q(v) - a(v)]^2}{q(v)a(v)} \\ + l\{D_{\mathrm{KL}}(q\|a) + D_{\mathrm{KL}}(a\|q)\} \geq l \|q - a\|_1^2,\end{aligned}$$

where Pinsker's inequality is applied to each KL term. At $q = q^*$, the gradient is constant across coordinates, and $\sum_v [q^*(v) - a(v)] = 0$. Hence

$$l \|q^* - a\|_1^2 \leq -\langle g_h, q^* - a \rangle \leq \tfrac{1}{2} \mathrm{osc}(g_h) \|q^* - a\|_1.$$

The last inequality follows by subtracting the midpoint of the largest and smallest coordinates of g_h. Division when the norm is nonzero proves Eq. equation 33; the zero case is immediate. Distances between probability vectors are at most two.

The objective's Bregman remainder at its stationary minimizer is

$$J(q_A) - J(q^*) = b \sum_v \pi(v) \left[\frac{q_A(v)}{q^*(v)} - 1 - \log \frac{q_A(v)}{q^*(v)} \right] + l D_{\mathrm{KL}}(q_A \| q^*).$$

Thus $\|q_A - q^*\|_1 \leq \sqrt{2\Delta_h / l}$. The triangle inequality yields Eq. equation 34. Finally, subtract ideal-reference stationarity at the calibrated law to obtain Eq. equation 35. □

The result is conditional on population quantities and does not certify empirical data construction. Small absolute data error may give a large residual where $p_A(v)$ is small, while reference error enters through a log ratio. This identifies a coverage requirement that more accurate optimization alone cannot resolve. The accompanying script `scripts/verify_auxiliary_calibration.py` checks calibration, stationarity, and these inequalities on finite probability vectors; it is a numerical check of the formulas, not an LLM experiment or a substitute for the proof.

Uncalibrated forget NLL is not generally equivalent. Consider $p_D = (1/3, 1/3, 1/3)$, $p_F = (1/2, 1/3, 1/6)$, $r_t = 1/2$, and $p_R = (1/6, 1/3, 1/2)$. These satisfy Eq. equation 1. With $\kappa = 2$, the amplified target is $a = (5/12, 1/3, 1/4)$. If the auxiliary data conditional is simply $\pi = p_F$, stationarity between coordinates one and two requires

$$\frac{l}{b} = \frac{1/5}{\log(5/4)} \approx 0.8963,$$

whereas coordinates three and two require

$$\frac{l}{b} = \frac{1/3}{\log(4/3)} \approx 1.1587.$$

No positive loss weights satisfy both, so this controlled target cannot be obtained by uncalibrated forget NLL plus reverse-KL, even with unrestricted optimization.

For comparison, replacing the regularizer by the opposite direction gives

$$\arg\min_q \{b\mathbb{E}_{p_F}[-\log q] + lD_{\mathrm{KL}}(p_D\|q)\} = \frac{lp_D + bp_F}{l+b}.$$

This arithmetic mixture follows by collecting cross-entropy terms and equals the local controlled target when $b/l = \kappa r_t$. It is a different objective from Eq. equation 20; a global loss-weight ratio also need not equal this context-dependent quantity.

C.3 Proof of Theorem 4.1

Assume the joint prompt–response laws of Eq. equation 25, with $\kappa > 0$ and BOS source posterior $\alpha_0 = \Pr(S = F \mid \mathrm{BOS})$. For any supported prefix z after BOS, marginalization and Bayes' rule give

$$(1 + \kappa\alpha_0)\frac{p_A(z)}{p_D(z)} = 1 + \kappa\frac{\alpha_0 p_F(z)}{p_D(z)} = 1 + \kappa \Pr(S = F \mid z). \tag{36}$$

The empty BOS prefix has probability one under both laws. Factoring successive prefix probabilities therefore shows that the ideal source-posterior recursion, initialized at α_0, processes each observed prompt token using the same update as each selected response token. After the complete prompt it yields $\alpha(x) = \Pr(S = F \mid x)$; this quantity is derived by prefill rather than separately supplied to the decoder. Taking $z = (x, y_{<t})$ gives $1 + \kappa r_t = (1 + \kappa\alpha_0)\Gamma_t$.

Conditioning the joint laws on x yields Eqs. equation 4 and equation 6, as shown in Appendix C.4. Under amplification, the unnormalized source masses at a response prefix are $(1 - \alpha)p_R(y_{<t} \mid x)$ and $(1 + \kappa)\alpha p_F(y_{<t} \mid x)$. Normalizing and conditioning on the next token yields Eq. equation 7. Eliminating the forget component gives

$$(1 + \kappa r_t)p_A - p_D = \kappa r_t p_F, \tag{37}$$

$$(1 + \kappa)p_D - (1 + \kappa r_t)p_A = \kappa(p_D - r_t p_F) = \kappa(1 - r_t)p_R. \tag{38}$$

For $0 \le r_t < 1$, division by $\kappa(1 - r_t)$ proves the recovery identity in Eq. equation 13, including $r_t = 0$. Rearranging gives Eq. equation 15. Marginalizing the conditional amplification law gives

$$\Lambda_t = \frac{p_A(y_{<t} \mid x)}{p_D(y_{<t} \mid x)} = \frac{1 + \kappa\alpha(x) p_F(y_{<t} \mid x)/p_D(y_{<t} \mid x)}{1 + \kappa\alpha(x)} = \frac{1 + \kappa r_t}{1 + \kappa\alpha(x)}. \tag{39}$$

Consequently, $1 + \kappa r_t = (1 + \kappa\alpha_0)\Gamma_t = (1 + \kappa\alpha(x))\Lambda_t$. At the empty response prefix, $\Lambda_1 = 1$ and $r_1 = \alpha(x)$ is the result of ideal prompt prefill. For any selected token with $p_D(y_t \mid h_t) > 0$, the chain rule gives $\Gamma_{t+1} = \Gamma_t \ell_t$ and $\Lambda_{t+1} = \Lambda_t \ell_t$. Substitution proves Eq. equation 14. Combining the identified source posterior with Eq. equation 13 proves Theorem 4.1 at every supported context with $r_t < 1$, regardless of how the path's tokens were selected.

The practical decoder instead computes $a_0 = \beta/(\kappa\lambda)$ from its training configuration and uses neural prompt likelihoods to obtain $a(x)$. Its equality to the source posterior requires calibrated BOS initialization, ideal joint observation laws, and inactive clipping. With ideal response likelihood ratios, the unprojected decoder state initialized at this computed $a(x)$ satisfies $1 + \kappa\widetilde{r}_t = (1 + \kappa a(x))\Lambda_t$. Subtracting the source-posterior identity proves Eq. equation 16, which propagates prefill error without requiring $\alpha(x)$ as an algorithm input.

C.4 Proof and Calibration of Proposition B.1

In this subsection, source laws describe the complete serialized token stream after BOS, so the input x is an observed prefix rather than an externally conditioned variable. For any prefix z with $p_D(z) > 0$, marginalizing Eq. equation 25 and applying Bayes' rule yields

$$\frac{p_A(z)}{p_D(z)} = \frac{1 + \kappa\alpha_0 p_F(z)/p_D(z)}{1 + \kappa\alpha_0} = \frac{1 + \kappa \Pr(S = F \mid z)}{1 + \kappa\alpha_0}. \tag{40}$$

At the empty prefix, both prefix probabilities equal one and the source posterior is α_0. Taking $z = x_{\le j}$ and factoring successive prefix probabilities proves Eq. equation 27 by induction, with $s_m = \alpha(x)$. Dividing the joint factual law by $p_D(x)$ gives the conditional mixture with weight $\alpha(x)$. Likewise,

$$p_A(y \mid x) = \frac{p_D(x, y) + \kappa\alpha_0 p_F(x, y)}{p_D(x) + \kappa\alpha_0 p_F(x)} = \frac{p_D(y \mid x) + \kappa\alpha(x) p_F(y \mid x)}{1 + \kappa\alpha(x)}.$$

This recovers the conditional amplification used in Theorem 4.1. Equivalently, that theorem's prefix identity can be written over the entire prompt–response prefix:

$$1 + \kappa r_t = (1 + \kappa\alpha_0)\frac{p_A(x, y_{<t})}{p_D(x, y_{<t})}.$$

For an empty prompt, $\alpha(x) = \alpha_0$. For a nonempty prompt, source-specific prompt likelihoods generally make the two different. Merely attaching the same deterministic BOS token to each sequence identifies neither the BOS source posterior nor the joint amplification law.

BOS-initialization and prompt-likelihood errors. Let $\widetilde{a}(x)$ denote the unprojected decoder state after neural prefill initialized at a_0, and define the accumulated prompt log-ratio error by

$$E_x = \log\frac{1 + \kappa a_0}{1 + \kappa\alpha_0} + \sum_{j=1}^{m}\left[\log\frac{q_A(x_j \mid x_{<j})}{q_D(x_j \mid x_{<j})} - \log\frac{p_A(x_j \mid x_{<j})}{p_D(x_j \mid x_{<j})}\right]. \tag{41}$$

On common positive support, telescoping the prompt likelihood product gives

$$1 + \kappa\widetilde{a}(x) = (1 + \kappa\alpha(x))e^{E_x}, \qquad |\widetilde{a}(x) - \alpha(x)| \le \frac{1 + \kappa\alpha(x)}{\kappa}(e^{|E_x|} - 1). \tag{42}$$

This is the prompt counterpart of Proposition C.3. Terminal projection cannot increase the error when $\alpha(x) \in [0, 1-\eta]$. Algorithm 1 instead projects after each prompt token, so its error is governed by the stepwise bound in Appendix C.10, applied to prompt tokens with initial discrepancy $|\operatorname{clip}(a_0, 0, 1-\eta) - \alpha_0|$. The product identity is exact for the raw prefill; it need not hold after intermediate clipping. Any resulting prefill error becomes the initialization error in Eq. equation 16.

With ideal prompt likelihoods, the initialization contribution simplifies to $\widetilde{a}(x) - \alpha(x) = (a_0 - \alpha_0)p_A(x)/p_D(x)$. Thus BOS prefill reduces the required initialization from an input-dependent function to one global scalar; it does not automatically correct a misspecified scalar.

BOS source-posterior calibration and coverage. If the ideal source model draws sequence d_i with probability proportional to ω_i, its empirical source posterior at BOS is the total forget sampling mass divided by total sampling mass. Uniform sequence sampling reduces this to the sequence-count fraction. Counting authors, documents, windows, or tokens gives the same BOS source posterior only when the sampling design makes those weights equivalent. A neural factual predictor trained on such data still need not equal the ideal mixture law; its discrepancy is a realization error, not a consequence of BOS formatting. Similarly, Theorem 5.1 applies to prompt scoring only if its data-calibration and coverage conditions also hold on prompt prefixes. Training solely on response positions does not supply that guarantee.

C.5 Reverse Identification for the Amplified Observation Model

Assume Eq. equation 7 with known $\kappa > 0$. When $0 < r_t < 1$ and $p_D \neq p_R$, eliminating the forget component gives

$$p_A - p_D = \frac{\kappa(1-r_t)}{1+\kappa r_t}(p_D - p_R). \tag{43}$$

Taking L_1 norms identifies the positive scalar $c_t = \kappa(1-r_t)/(1+\kappa r_t)$, whose derivative is $-\kappa(1+\kappa)/(1+\kappa r_t)^2 < 0$. The four ideal distributions are therefore collinear, and the coefficient has a unique inverse:

$$c_t = \frac{\|p_A - p_D\|_1}{\|p_D - p_R\|_1}, \qquad r_t = \frac{\kappa - c_t}{\kappa(1+c_t)}, \qquad p_F = p_R + \frac{p_D - p_R}{r_t}. \tag{44}$$

These equations recover r_t, p_F uniquely when p_R is supplied and the mixture model holds. Using an approximate retain predictor instead yields a proxy diagnostic whose accuracy depends on that predictor. The residual in Eq. equation 43 must also be checked: a norm ratio alone does not establish collinearity. Without a retain reference or another source of weight information, the local pair need not be sufficient.

The amplified pair alone need not identify the endpoint. For $\kappa = 1$, the same pair $p_D = (1/2, 1/2)$ and $p_A = (3/5, 2/5)$ is generated both by $(r, p_R, p_F) = (1/4, (1/3, 2/3), (1, 0))$ and by $(1/2, (1/5, 4/5), (4/5, 1/5))$. Both satisfy Eq. equation 7. Thus the amplified pair need not identify the retain endpoint; supplying the weight or a nondegenerate retain reference removes this ambiguity under the respective assumptions above.

C.6 Likelihood Training and Predictive Error

Let μ be a context distribution, p a target conditional, and $P(h, v) = \mu(h)p(v \mid h)$ their joint law. The expected negative log-likelihood (NLL) of a predictor q is $L_P(q) = \mathbb{E}_P[-\log q(v \mid h)]$; all logarithms are natural.

Proposition C.2 (Excess likelihood and predictive error)**.** *For any predictor q with finite loss, writing $\Delta_P(q) = L_P(q) - L_P(p)$,*

$$\Delta_P(q) = \mathbb{E}_\mu D_{\mathrm{KL}}(p\|q), \qquad \mathbb{E}_\mu \|q - p\|_1 \leq \sqrt{2\Delta_P(q)}. \tag{45}$$

For empirical P, the unrestricted optimum is the empirical conditional frequency on observed contexts. Equation equation 45 still holds for a parameter-sharing neural predictor; it does not assert that its excess loss is small.

If an evaluation context law ν satisfies $\nu(h) \le W\mu(h)$ for a finite coverage bound W, then $\mathbb{E}_\nu \|q - p\|_1 \le \sqrt{2W\Delta_P(q)}$. Without coverage or a generalization argument, empirical loss does not control unseen generation contexts.

Proof. Subtracting conditional entropy from cross-entropy gives

$$L_P(q) - L_P(p) = \sum_h \mu(h) \sum_v p(v \mid h) \log \frac{p(v \mid h)}{q(v \mid h)} = \mathbb{E}_\mu D_{\mathrm{KL}}(p \| q). \tag{46}$$

Pinsker's inequality gives $\|q - p\|_1 \le \sqrt{2D_{\mathrm{KL}}(p\|q)}$ pointwise, and Jensen's inequality establishes Eq. equation 45. For empirical P, let $c(h, v)$ count occurrences of the context–token pair (h, v) and let $c(h) = \sum_{v \in \mathcal{V}} c(h, v)$ be its context count. Then $p(v \mid h) = c(h, v)/c(h)$ for $c(h) > 0$, and minimizing over independent conditional probability vectors yields these frequencies. Unobserved contexts remain unconstrained, and a parameter-sharing model need not attain this unrestricted optimum.

For an evaluation context law ν satisfying $\nu(h) \le W\mu(h)$ with finite W, define the context-density ratio $w(h) = \nu(h)/\mu(h)$ on the support of μ. Then

$$\mathbb{E}_\nu \|q - p\|_1 \le \sqrt{2\mathbb{E}_\nu D_{\mathrm{KL}}(p\|q)} \le \sqrt{2W\mathbb{E}_\mu D_{\mathrm{KL}}(p\|q)}. \tag{47}$$

For a predictor q in a model family $\mathcal{Q}$, the excess loss decomposes into

$$\Delta_P(q) = \big[L_P(q) - \inf_{q' \in \mathcal{Q}} L_P(q')\big] + \big[\inf_{q' \in \mathcal{Q}} L_P(q') - L_P(p)\big]. \tag{48}$$

The first term is the optimization gap; the second is the approximation gap, which includes restrictions from parameter sharing. Transferring a statement from empirical to population loss requires a separate generalization argument.

C.7 Proof of Theorem 3.1

Insert $\mathcal{R}_\kappa(p_D, p_A, \widehat{r})$ between the estimated and oracle recoveries to separate predictive error from ratio error. Linearity in the predictive distributions gives

$$\begin{aligned} &\|\mathcal{R}_\kappa(q_D, q_A, \widehat{r}) - \mathcal{R}_\kappa(p_D, p_A, \widehat{r})\|_1 \\ &\quad \le \frac{1+\kappa}{\kappa(1-\widehat{r})}\epsilon_D + \frac{1+\kappa\widehat{r}}{\kappa(1-\widehat{r})}\|q_A - p_A\|_1 \\ &\quad \le \frac{1+\kappa}{\kappa\eta}\epsilon_D + \frac{1+\kappa(1-\eta)}{\kappa\eta}(\epsilon_A + \epsilon_{\mathrm{align}}). \end{aligned} \tag{49}$$

For the remaining term, direct subtraction or differentiation gives

$$\mathcal{R}_\kappa(p_D, p_A, \widehat{r}) - \mathcal{R}_\kappa(p_D, p_A, r) = \frac{(1+\kappa)(\widehat{r} - r)}{\kappa(1-\widehat{r})(1-r)}(p_D - p_A). \tag{50}$$

The factors $1 - \widehat{r}$ and $1 - r$ are both at least η. Taking norms and combining the two terms proves Eq. equation 10. Two applications of the triangle inequality give Eq. equation 11.

For the shift repair, write $\zeta = \max\{0, -\min_{v \in \mathcal{V}} z(v)\}$ and let $\mathcal{S}$ denote the repair map in Eq. equation 18. The repair is valid because $\sum_v z(v) = 1$, $z(v) + \zeta \ge 0$, and $1 + |\mathcal{V}|\zeta \ge 1$. Its difference from the raw vector is

$$\mathcal{S}(z) - z = \frac{\zeta(\mathbf{1} - |\mathcal{V}|z)}{1 + |\mathcal{V}|\zeta}, \tag{51}$$

Taking norms gives the explicit repair cost

$$\|p_{\mathrm{CURE}} - z_t\|_1 = \frac{\zeta_t}{1 + |\mathcal{V}|\zeta_t} \|\mathbf{1} - |\mathcal{V}| z_t\|_1 . \tag{52}$$

This makes the vocabulary-size dependence explicit. For comparison, let $\Delta = \{p \in \mathbb{R}^{|\mathcal{V}|} : p(v) \geq 0,\ \sum_v p(v) = 1\}$ be the probability simplex, and let Π_Δ be its Euclidean projection. Here $\|z\|_2 = (\sum_v z(v)^2)^{1/2}$. The projection satisfies $\|\Pi_\Delta(z) - p_R\|_2 \leq \|z - p_R\|_2$; this L_2 property does not establish an L_1 bound. Theorem 3.1 accounts for repair distortion directly and assumes no non-expansiveness of the repair map.

When shift repair preserves recovery. Let $V = |\mathcal{V}|$ be the vocabulary size and $p_{\mathrm{unif}} = \mathbf{1}/V$ the uniform token distribution. Then

$$\mathcal{S}(z) - p_R = \frac{z - p_R + V\zeta(p_{\mathrm{unif}} - p_R)}{1 + V\zeta}, \qquad \|\mathcal{S}(z) - p_R\|_1 \leq \frac{B(h) + 2V\zeta}{1 + V\zeta}. \tag{53}$$

The inequality follows from $\|z - p_R\|_1 \leq B(h)$ and $\|p_{\mathrm{unif}} - p_R\|_1 \leq 2$. Since $z - p_R$ sums to zero, each negative coordinate has magnitude at most $\|z - p_R\|_1 /2$; hence $\zeta \leq B(h)/2$ and the simpler bound $\|\mathcal{S}(z) - p_R\|_1 \leq (V + 1)B(h)$ follows. At fixed vocabulary size, vanishing raw error therefore implies vanishing repaired error. The potentially large vocabulary factor motivates measuring $V\zeta$ directly for a sharper diagnostic. Error relative to the retrained predictor additionally includes ϵ_R.

C.8 Recursive Estimation and Accumulated Error

Let $\ell_i = p_A(y_i \mid h_i)/p_D(y_i \mid h_i) > 0$ and $\widehat{\ell}_i = q_A(y_i \mid h_i)/q_D(y_i \mid h_i) > 0$ be the ideal and neural token likelihood ratios. Here $\widetilde{r}_t$ is the unprojected decoder state, starting from the BOS-prefill output $a = a(x)$; r_t denotes the source posterior. Define the accumulated log-state error E_t, local log-ratio error e_i, and initial error E_1:

$$E_t = \log \frac{1 + \kappa\widetilde{r}_t}{1 + \kappa r_t}, \quad e_i = \log \widehat{\ell}_i - \log \ell_i, \quad E_1 = \log \frac{1 + \kappa a}{1 + \kappa\alpha(x)}. \tag{54}$$

Proposition C.3 (Exact accumulated-error identity)**.** *Let $r_1 = \alpha(x)$ denote the source posterior after the prompt and $r_{i+1} = F(r_i, \ell_i)$ its oracle recursion, and let $\widetilde{r}_1 = a$ and $\widetilde{r}_{i+1} = F(\widetilde{r}_i, \widehat{\ell}_i)$ be the response decoder-state recursion. On common positive support,*

$$E_t = E_1 + \sum_{i<t} e_i, \qquad |\widetilde{r}_t - r_t| \leq \frac{1 + \kappa r_t}{\kappa}\big(e^{|E_t|} - 1\big). \tag{55}$$

For deterministic E_1 and finite second moments, $\mathbb{E}E_t = E_1 + \sum_{i<t} \mathbb{E}e_i$ and $\mathrm{Var}(E_t) = \sum_{i,j<t} \mathrm{Cov}(e_i, e_j)$.

Proof. Unrolling the positive likelihood recursion yields

$$\frac{1 + \kappa\widetilde{r}_t}{1 + \kappa r_t} = \frac{1 + \kappa a}{1 + \kappa\alpha(x)} \prod_{i<t} \frac{\widehat{\ell}_i}{\ell_i}. \tag{56}$$

Taking logarithms proves the additive identity. Exponentiating gives $\widetilde{r}_t - r_t = (1 + \kappa r_t)(e^{E_t} - 1)/\kappa$, and $|e^u - 1| \leq e^{|u|} - 1$ for real u gives the error bound. Linearity of expectation and the covariance formula establish the moment statements whenever second moments exist.

For a filtration $(\mathcal{F}_i)_{i\geq 0}$ recording information through step i, suppose $e_i = b_i + \xi_i$, where b_i is predictable drift and ξ_i is conditionally mean-zero noise with deterministic sub-Gaussian scale $\sigma_i \geq 0$. Then for every

failure probability $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\left| E_t - E_1 - \sum_{i<t} b_i \right| \le \sqrt{2\log(2/\delta) \sum_{i<t} \sigma_i^2}. \tag{57}$$

Precisely, b_i is measurable with respect to $\mathcal{F}_{i-1}$, ξ_i is measurable with respect to $\mathcal{F}_i$, and the conditional sub-Gaussian assumption is

$$\mathbb{E}[\exp(s\xi_i) \mid \mathcal{F}_{i-1}] \le \exp(s^2\sigma_i^2/2), \quad s \in \mathbb{R}. \tag{58}$$

Iterating conditional expectations bounds the moment generating function of $\sum_{i<t} \xi_i$ by $\exp(s^2 \sum_{i<t} \sigma_i^2/2)$. Markov's inequality, optimization over the real parameter s, and a union bound for the two tails prove Eq. equation 57. For a fixed positive integer horizon T, a simultaneous bound for all $t \le T$ follows by replacing $\log(2/\delta)$ with $\log(2T/\delta)$. Random initialization requires including its error and dependence, rather than treating E_1 as a constant.

C.9 Finite-Sample Consistency, Bias, and Variance

We instantiate the likelihood estimator under a controlled sampling model. Fix x and a finite sequence space, such as sequences of a fixed positive integer length H over $\mathcal{V}$, padded with an end-of-sequence (EOS) token. Draw N_D independent sequences from $p_D(\cdot \mid x)$ and N_A independent sequences from $p_A(\cdot \mid x)$, with the two collections independent. Fix a test path independently of these samples. For a known, full-support probability law U on the sequence space, define the smoothed empirical laws

$$\widehat{P}_j = \frac{N_j \overline{P}_j + U}{N_j + 1}, \qquad j \in \{D, A\}, \tag{59}$$

where $j \in \{D, A\}$ labels the full-data or auxiliary sample, N_j is its size, and $\overline{P}_j$ is its empirical frequency measure. The next-token predictors are the conditionals of these laws. Write $\widehat{P}_j(y_{<t})$ for their prefix marginal probabilities at the fixed input x. Full-support smoothing makes every ratio well-defined, and the autoregressive product telescopes exactly to $\widehat{P}_A(y_{<t})/\widehat{P}_D(y_{<t})$. Probabilities, expectations, and variances in the following proposition are over the two random samples, with the test path fixed.

Proposition C.4 (Finite-sample bounds for the raw recursive estimator)**.** *Let N_D, N_A be positive integer sample sizes, T a positive integer test-prefix horizon, and $\delta \in (0,1)$ a failure probability. Assume a known source posterior $\alpha(x) \in [0,1]$ and $\kappa > 0$, ideal amplification, and test prefixes $t \le T$ with $\pi_{D,t} := p_D(y_{<t} \mid x) \ge m > 0$ and $r_t \in [0,b]$, where m is a prefix-probability lower bound and $b \in [0,1)$ is a ratio cap. Initialize the raw recursion at $\alpha(x)$ using the conditionals of Eq. equation 59, and write $\overline{r}_t = \mathrm{clip}(\widetilde{r}_t, 0, b)$ for its terminal projection at each queried prefix (this projection is not fed back into the recursion). Define the scale C, the prefix-deviation bounds ω_j, and the prefix mean-squared-error bounds υ_j by*

$$C = \frac{1 + \kappa\alpha(x)}{\kappa}, \quad \omega_j = \sqrt{\frac{\log(4T/\delta)}{2N_j}} + \frac{1}{N_j+1}, \quad \upsilon_j = \frac{1}{4N_j} + \frac{1}{(N_j+1)^2}. \tag{60}$$

If $\omega_D \le m/2$, then with probability at least $1-\delta$, simultaneously for $t \le T$,

$$|\overline{r}_t - r_t| \le C\left(\frac{2\omega_A}{m} + \frac{2\omega_D}{m^2}\right). \tag{61}$$

If $1/(N_D+1) \le m/4$, then for every such fixed prefix the ratio mean squared error is bounded by M, defined as

$$\mathbb{E}[(\overline{r}_t - r_t)^2] \le M := C^2\left(\frac{8\upsilon_A}{m^2} + \frac{8\upsilon_D}{m^4}\right) + 2b^2\exp(-N_D m^2/8). \tag{62}$$

In particular $|\mathbb{E}\overline{r}_t - r_t| \le \sqrt{M}$ *and* $\mathrm{Var}(\overline{r}_t) \le M$. *For fixed* T, m, *both sample sizes tending to infinity give consistency, vanishing bias, and vanishing variance.*

Proof. For $j \in \{D, A\}$, write $\pi_{j,t} = p_j(y_{<t} \mid x)$ and $\widehat{\pi}_{j,t} = \widehat{P}_j(y_{<t})$ for the true and smoothed empirical prefix probabilities. Telescoping gives $\widetilde{r}_t = ((1+\kappa\alpha)\widehat{\pi}_{A,t}/\widehat{\pi}_{D,t} - 1)/\kappa$. Each unsmoothed prefix frequency is an average of independent Bernoulli indicators, and smoothing changes it by at most $1/(N_j+1)$. Hoeffding's inequality and a union bound over at most $2T$ frequencies therefore give $|\widehat{\pi}_{j,t} - \pi_{j,t}| \le \omega_j$ for both $j \in \{D, A\}$ and all $t \le T$ with the stated probability. On this event $\widehat{\pi}_{D,t} \ge m/2$ and, since $\pi_{A,t} \le 1$,

$$\left|\frac{\widehat{\pi}_{A,t}}{\widehat{\pi}_{D,t}} - \frac{\pi_{A,t}}{\pi_{D,t}}\right| \le \frac{2}{m}|\widehat{\pi}_{A,t} - \pi_{A,t}| + \frac{2}{m^2}|\widehat{\pi}_{D,t} - \pi_{D,t}|. \tag{63}$$

Multiplication by C and non-expansiveness of interval projection relative to feasible r_t prove Eq. equation 61.

For the moment bound, let $G = \{|\widehat{\pi}_{D,t} - \pi_{D,t}| \le m/2\}$ be the event that the empirical denominator stays close to its true value, and let G^c denote its complement. Squaring Eq. equation 63 and using $(u+v)^2 \le 2u^2 + 2v^2$ for real u, v bounds the squared error on G by

$$C^2\left(\frac{8|\widehat{\pi}_{A,t} - \pi_{A,t}|^2}{m^2} + \frac{8|\widehat{\pi}_{D,t} - \pi_{D,t}|^2}{m^4}\right).$$

The variance of a smoothed Bernoulli frequency is at most $1/(4N_j)$ and its absolute bias is at most $1/(N_j+1)$, so its mean squared error is at most v_j. The smoothing condition and Hoeffding give $\Pr(G^c) \le 2\exp(-N_D m^2/8)$. On G^c, projection bounds $|\overline{r}_t - r_t| \le b$. Combining the two events proves Eq. equation 62. Finally, squared bias and variance are each at most the mean squared error around the fixed target. □

Scope. The known source posterior in this conditional finite-sample model is an analysis assumption, not an input to Algorithm 1; that algorithm obtains its decoder state from BOS prefill. Independence is required across sequences, with arbitrary token dependence within each sequence. The test path must be independent of the training samples. Adaptively selected test prefixes require uniform control over the candidate prefix set; for a fixed finite set, replacing T by its cardinality supplies this control. As the horizon grows, the prefix-probability lower bound m can decrease rapidly, so the fixed-horizon guarantee does not extend to arbitrarily long generations. The result also requires ideal alignment and calibration: proxy mismatch, persistent initialization error, or a fixed biased temperature can prevent consistency. It is a guarantee for the stated sequence estimator, rather than for neural optimization.

C.10 Stepwise Projection and Temperature

Let $b = 1 - \eta$ be the projection cap and $d_t^{\mathrm{proj}} = |\widehat{r}_t - r_t|$ the absolute discrepancy between the stepwise-projected decoder state and source posterior. Use the raw update $F(r, \ell) = ((1+\kappa r)\ell - 1)/\kappa$, the oracle likelihood-ratio bound $\ell_t \le L$, and local error bounds $|\widehat{\ell}_t - \ell_t| \le \delta_t$. If $r_{t+1} \in [0, b]$, interval projection yields

$$\begin{aligned} d_{t+1}^{\mathrm{proj}} &\le |F(\widehat{r}_t, \widehat{\ell}_t) - F(r_t, \ell_t)| \\ &\le \ell_t d_t^{\mathrm{proj}} + \frac{1+\kappa\widehat{r}_t}{\kappa}|\widehat{\ell}_t - \ell_t| \le L d_t^{\mathrm{proj}} + \frac{1+\kappa b}{\kappa}\delta_t. \end{aligned} \tag{64}$$

Iteration gives

$$d_t^{\mathrm{proj}} \le L^{t-1} d_1^{\mathrm{proj}} + \frac{1+\kappa b}{\kappa}\sum_{i<t} L^{t-1-i}\delta_i. \tag{65}$$

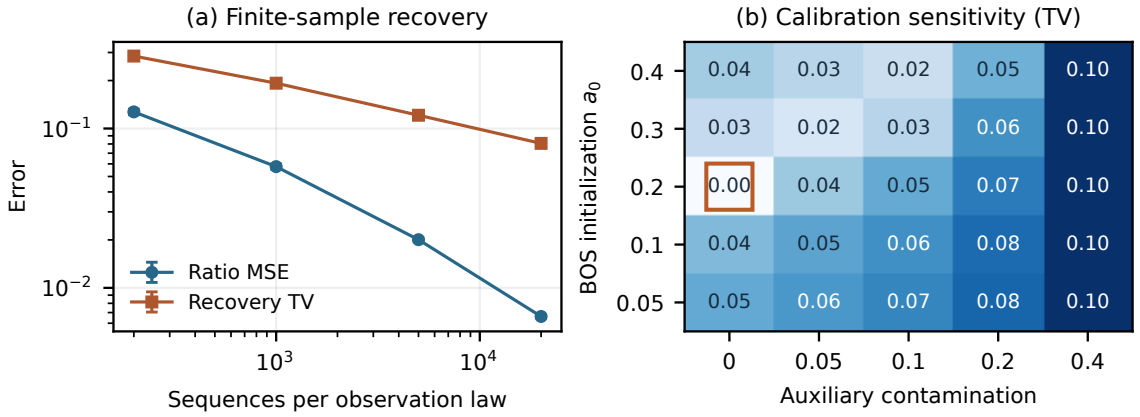


Figure 3: **Executed controlled experiment.** Finite-sample recovery and calibration sensitivity under known synthetic laws.

If the oracle exceeds the cap, the one-step inequality instead has the additional term $\mathrm{dist}(r_{t+1}, [0, b])$, where $\mathrm{dist}(r, [0, b]) = \inf_{s \in [0,b]} |r - s|$ is distance to the feasible interval. A cap stabilizes inversion but can introduce irreducible bias. For fixed T, finite L, feasible oracle ratios, and $d_1^{\mathrm{proj}}, \delta_i \to 0$ in probability, Eq. equation 65 proves consistency of the stepwise projected estimator. The finite sequence model above provides one regime in which the necessary conditional likelihood convergence holds.

If the ideal and learned full-data token probabilities satisfy $p_D(y_t \mid h_t), q_D(y_t \mid h_t) \geq m_0 > 0$, where m_0 is a common denominator-probability margin, direct subtraction yields

$$|\widehat{\ell}_t - \ell_t| \leq \frac{|q_A(y_t \mid h_t) - p_A(y_t \mid h_t)|}{m_0} + \frac{|q_D(y_t \mid h_t) - p_D(y_t \mid h_t)|}{m_0^2}. \tag{66}$$

This bound connects predictive error to local ratio error under a denominator-probability margin. For ratio temperature $\tau > 0$, define $q_j^{(\tau)}(v) = q_j(v)^{1/\tau} / Z_j(\tau)$ and $Z_j(\tau) = \sum_{v \in \mathcal{V}} q_j(v)^{1/\tau}$ for $j \in \{D, A\}$, with the context suppressed. Temperature transforms the log ratio as

$$\log \frac{q_A^{(\tau)}(v)}{q_D^{(\tau)}(v)} = \frac{1}{\tau} \log \frac{q_A(v)}{q_D(v)} + \log Z_D(\tau) - \log Z_A(\tau). \tag{67}$$

The normalization terms show why even perfect models need not recover the original density ratio at $\tau \neq 1$. The local error δ_t must include this discrepancy.

D Controlled Finite-Sequence Experiment

This experiment uses explicitly specified synthetic source laws to test identification and sampling error. It does not use an LLM, external data, or any TOFU/MUSE measurement. The complete implementation is `scripts/controlled_recovery.py`; `experiments/controlled/trials.csv` contains trial-level metrics and `results.json` the parameters and summaries.

Population laws and evaluation. The vocabulary has size three and every sequence has length $H = 6$. Retain and forget laws are products of the token probabilities $(0.6, 0.3, 0.1)$ and $(0.1, 0.3, 0.6)$, respectively.

The source posterior at BOS is $\alpha_0 = 0.2$ and $\kappa = 1$. We enumerate all $3^6 = 729$ sequences to compute $p_D = (1 - \alpha_0)p_R + \alpha_0 p_F$ and $p_A = (p_D + \kappa\alpha_0 p_F)/(1 + \kappa\alpha_0)$. Evaluation covers every prefix at depths $t = 0, \ldots, 5$, totaling 364 contexts. For a per-context quantity $g(z)$, its reported average is $H^{-1} \sum_{t=0}^{H-1} \sum_{z \in \mathcal{V}^t} p_D(z)g(z)$. This weights each depth equally and uses the factual prefix law within that depth. The factual-to-retain TV average is exactly 0.1.

Sampling and estimator. For each $N \in \{200, 1000, 5000, 20000\}$, we draw N independent sequences from each observation law, independently across the two laws and across 100 trials. Multinomial sequence counts implement the draws exactly. A NumPy generator with seed 20260923 controls the experiment. We add one total unit of uniform mass over all 729 sequences and normalize by $N + 1$, as in Eq. equation 59. Prefix marginals and token conditionals come from the same smoothed sequence law. We compute the raw ratio from these marginals, then project each queried estimate onto $[0, 1 - 10^{-6}]$ without feeding the projection back into the recursion. All oracle states in this experiment lie within that interval. Recovery uses the estimated token conditionals, the projected weight, and the shift repair in Eq. equation 18. This is the terminal-projected statistical estimator, distinguished from Algorithm 1's stepwise clipping.

Metrics and uncertainty. Ratio MSE averages squared error against the known source posterior. Recovery TV averages $\frac{1}{2} \|p_{\mathrm{CURE}} - p_R\|_1$ against the known retain conditional. We report means across trials; the intervals are normal approximations, mean ± 1.96 times the sample standard deviation divided by $\sqrt{100}$. They quantify Monte Carlo uncertainty of these means, not variation across LLM training runs. The exact-law control has maximum state error 4.6×10^{-16} and maximum raw recovery L_1 error 3.2×10^{-13} or less, consistent with floating-point arithmetic.

Table 3: Controlled sampling results, mean $\pm$ approximate 95% confidence-interval half-width over 100 trials.

Sequences per law	Ratio MSE	Repaired recovery TV
200	0.12717 ± 0.00513	0.28488 ± 0.00402
1,000	0.05765 ± 0.00226	0.19248 ± 0.00240
5,000	0.02007 ± 0.00054	0.12113 ± 0.00113
20,000	0.00662 ± 0.00017	0.08067 ± 0.00082

Misspecification study. Using exact factual probabilities, we set the auxiliary law to $(1-\varepsilon)p_A + \varepsilon p_R$, with $\varepsilon \in \{0, 0.05, 0.1, 0.2, 0.4\}$, and vary $a_0 \in \{0.05, 0.1, 0.2, 0.3, 0.4\}$. Both effects persist without sampling noise. At $\varepsilon = 0.4$, the auxiliary forget-source weight equals 0.2, so the auxiliary and factual laws coincide. The raw recovery then equals the factual conditional regardless of the estimated weight, giving average TV 0.1. Other combinations can partially offset their errors; low average recovery error alone therefore does not establish equality between the decoder state and source posterior.

Scope. This small full-support problem permits exhaustive ground-truth evaluation. It provides no evidence that neural training realizes the ideal laws, and the sampling results do not establish rates beyond the assumptions in Appendix C.9. The synthetic BOS source posterior 0.2 and projection margin 10^{-6} are specific to this study; they are separate from the LLM configuration.

E Additional Experiments and Implementation Details

This appendix first records the decoder and evaluation settings, then specifies the common TOFU and MUSE evaluation protocols, additional comparisons, model-scale diagnostics, auxiliary-adaptation studies, and generation examples. The tables report point estimates; repeated-run uncertainty is not available.

E.1 Auxiliary Training and Decoder Specification

The CURE configuration uses an 8-layer auxiliary model, low-rank adaptation (LoRA) with rank 16, AdamW learning rate 10^{-4}, 10 epochs, ratio temperature 1.0, and auxiliary loss weights $\beta = 0.5$, $\lambda = 1$. Consistent with Algorithm 1 and Section 5, all ratio computations use normalized, untempered probabilities ($\tau = 1$): tempering changes the density ratio (Appendix C.10). The source amplification is fixed to $\kappa = 1$; together, these settings determine the BOS decoder state by Eq. equation 19. The TOFU target is Llama-3.1-8B-Instruct, with Qwen3-8B evaluated separately. Figure 2(a,b) illustrates auxiliary construction and training; the documented objective uses the reverse Kullback–Leibler (KL) direction in Eq. equation 20.

The figure labels P_D and P_{add} denote the factual and added-data joint laws; their displayed x and $y \mid x$ factors are input marginals and output conditionals. The added-data law is distinct from the ideal amplified predictor p_A. The labels D and D_{add} denote the original and auxiliary datasets, with D_{add} corresponding to $\mathcal{D}_{\text{aux}}$. The loss labels $\mathcal{L}_{NLL}$ and $\mathcal{L}_{KL}$ denote auxiliary-data negative log likelihood (NLL) and reverse-KL regularization, respectively; $\mathcal{L}_{total}$ corresponds to Eq. equation 20 with $\lambda = 1$. Its weight w_{add} denotes the optimization weight β, distinct from source amplification κ.

In panel (c) of Figure 2, p_D, $p_{D'}$, and p_{rtd} label factual, auxiliary, and recovered token probabilities, respectively. For neural decoding, the first two correspond to q_D and q_A; the recovered column is a counterfactual output, not an observed retain-only oracle q_R. The labels r_1 and r_2 denote illustrative decoder states, written s_t for the decoder below, and $\langle \text{bos} \rangle$ is the beginning-of-sequence token. The displayed scores and state transition illustrate the beam-search structure, rather than a numerical execution of Algorithm 1; each continuation carries its own state and score.

Prompt prefill and initialization. The implementation starts ratio tracking at the shared BOS marker with $a_0 = \beta/(\kappa\lambda) = 0.5/(1 \cdot 1) = 0.5$ and processes the observed prompt before response decoding. This value follows from the auxiliary training configuration; it is not an independently selected initialization. The complete model input must use the same tokenization and serialization for q_D and q_A, including the chat template, system text, few-shot examples, and response-start marker when present. BOS is the common starting condition and is not scored as an observed token. Prompt tokens update the decoder state but are not resampled, repaired, or included in response beam scores. The resulting decoder state $a(x)$ initializes the first response token; its source-posterior interpretation requires the joint-mixture and calibration conditions of Proposition B.1. Appendix B.1 distinguishes the training-derived decoder state a_0 from the source posterior α_0 at BOS. The value 0.5 is not inferred from the forget-set fraction. The reported decoder uses beam width 4 and clips its decoder state to $[0, 1]$ during prompt prefill and response generation; each response beam carries its own state and score.

Upper endpoint and theoretical margin. The reported upper cap is 1, with numerical ε protection. Algorithm 1 uses $1 - \eta$ for $\eta > 0$, ensuring a finite inverse and the bound in Theorem 3.1. At $\widehat{r} = 1$, Eq. equation 17 divides by zero; subsequent repair cannot define that endpoint. Appendix F.2 records an ε-smoothed repair differing from Eq. equation 18, but its smoothing value and the raw inverse's endpoint rule remain unspecified. State clipping and denominator regularization are distinct, so the reported decoder cannot yet be equated with the strict-margin reference algorithm.

Strict-margin reference decoder. The algorithm assumes positive probabilities, as produced by finite, unmasked softmax logits; EOS denotes the end-of-sequence token and $\mathrm{expm1}(s) = e^s - 1$.

Algorithm 1 CURE decoding for one hypothesis

Require: Frozen q_D, auxiliary q_A with training weights $\beta \geq 0$, $\lambda > 0$; serialized prompt $x = (x_1, \ldots, x_m)$; $\kappa > 0$ with $\beta \leq \kappa\lambda$; $\eta \in (0, 1)$; untempered probabilities

Compute $a_0 = \beta/(\kappa\lambda)$; initialize $s_0 = \mathrm{clip}(a_0, 0, 1 - \eta)$ at BOS
for each observed prompt token x_j, $j = 1, \ldots, m$ **do**
    Set $d_j = \log q_A(x_j \mid x_{<j}) - \log q_D(x_j \mid x_{<j})$ (both conditioned on BOS)
    Set $s_j = \mathrm{clip}(\mathrm{expm1}(\log(1 + \kappa s_{j-1}) + d_j)/\kappa, 0, 1 - \eta)$
end for
Initialize $\widehat{r}_1 = a(x) = s_m$ and response log score $u_1 = 0$
for each prefix h_t until EOS or the length limit **do**
    Evaluate $q_D(\cdot \mid h_t)$ and $q_A(\cdot \mid h_t)$
    Recover $z_t = \mathcal{R}_\kappa(q_D, q_A, \widehat{r}_t)$; repair by Eq. equation 18
    Select the next token $v = y_t$ using the repaired distribution
    Set $d = \log q_A(v \mid h_t) - \log q_D(v \mid h_t)$
    Set $\widehat{r}_{t+1} = \mathrm{clip}(\mathrm{expm1}(\log(1 + \kappa\widehat{r}_t) + d)/\kappa, 0, 1 - \eta)$
    Store the continuation with score $u_{t+1} = u_t + \log p_{\mathrm{CURE}}(v \mid h_t)$ and its own state
end for
return The generated sequence and its score

Each observed or selected token updates the *next* decoder state. Avoid overflow by comparing log-states with clipping endpoints before exponentiation. All response beams inherit the same prefill state and then maintain separate states and caches, prune by score, and retain completed sequences. The intervention requires this decoder.

Accepted auxiliary-data law. The auxiliary predictor is adapted to forget-related examples, supplied with the forget request or generated and filtered by an acceptance rule. For a proposal distribution $Q(x, y)$ and acceptance probability $\pi_{\mathrm{acc}}(x, y) \in [0, 1]$, the accepted law is $Q(x, y)\pi_{\mathrm{acc}}(x, y)/\mathbb{E}_{(x,y)\sim Q}[\pi_{\mathrm{acc}}(x, y)]$, provided the denominator is positive. This formula alone does not establish controlled amplification. Under reverse-KL training, exact population equivalence requires the induced context–token law to satisfy Eq. equation 22, with its actual context marginal included in β_h. Setting the accepted conditional equal to p_A is generally insufficient because regularization shifts the optimum toward p_D. Theorem 5.1 characterizes the required calibration; it does not certify that a particular finite accepted dataset satisfies it. Departures from this condition contribute to auxiliary-alignment error.

The reported environment is Ubuntu 22.04.3, Python 3.11.11, PyTorch 2.2.2, Transformers 4.56.2, Accelerate 1.10.1, DeepSpeed 0.17.6, Hydra 1.3.2, and Datasets 2.16.0 on one NVIDIA A800 80GB GPU, with two Hygon C86 7390 CPUs (64 cores in total) and 2.0 TiB RAM. Deterministic seeding is used, but variability across independent training seeds is not reported.

E.2 Benchmark Metrics and Baseline Settings

TOFU splits and reference models. TOFU contains question–answer pairs about fictitious authors (Maini et al., 2024); the 1%, 5%, and 10% splits specify the forget proportion, with the remainder retained. Original/Target is the model before unlearning; Retain-only/Retrain is trained without the forget set. R-L abbreviates ROUGE-L, based on the longest common subsequence between a generated answer and its reference;

Fgt and Rtd denote the forget and retain splits. The two reference models provide behavioral comparisons. The retain-only predictor q_R differs from the retain-source conditional p_R in the identification theorem.

Truth Ratio and Forget Quality. For question x, let the finite answer collections $\mathcal{A}_{\mathrm{corr}}$ contain correct-answer paraphrases and $\mathcal{A}_{\mathrm{pert}}$ contain plausible incorrect answers. Vertical bars around an answer collection denote its cardinality. For evaluated model parameters θ, write $\mathrm{NLL}_\theta(y \mid x) = -|y|^{-1} \log P_\theta(y \mid x)$, where $|y|$ is the number of scored answer tokens and $P_\theta(y \mid x)$ is their autoregressive likelihood. This mean NLL differs from the token likelihood ratio ℓ_t. The evaluator used in all our TOFU experiments (Appendix E.3) implements

$$R_{\mathrm{truth}}(x) = \frac{\exp\Big[-|\mathcal{A}_{\mathrm{pert}}|^{-1} \sum_{\widetilde{y} \in \mathcal{A}_{\mathrm{pert}}} \mathrm{NLL}_\theta(\widetilde{y} \mid x)\Big]}{\exp\Big[-|\mathcal{A}_{\mathrm{corr}}|^{-1} \sum_{y \in \mathcal{A}_{\mathrm{corr}}} \mathrm{NLL}_\theta(y \mid x)\Big] + 10^{-10}}. \tag{68}$$

Thus answers in each branch are combined by the geometric mean of their length-normalized likelihoods, rather than an arithmetic mean of probabilities. The forget and retain branches use correct-answer paraphrases; Real Authors and World Facts use their original correct answers. This formula matches the evaluation code used for all reported TOFU results. A small ratio indicates preference for the correct answer; a value near one indicates comparable likelihoods. FQ is the two-sample Kolmogorov–Smirnov test p-value comparing forget-set Truth Ratio distributions for the unlearned and retain-only models, multiplied by 100. High FQ means failure to reject the null, without establishing predictive equivalence or erasure. Low FQ can reflect residual memorization or excessive suppression.

Model Utility. TOFU Model Utility (MU) harmonically aggregates scores on the retain set, Real Authors, and World Facts, assessing in-domain, neighboring, and general knowledge. It emphasizes low component scores and measures broader utility than retain ROUGE-L alone; higher is better. In our evaluations, each set contributes mean probability, ROUGE-L recall, and Truth Ratio utility $\max(0, 1 - R_{\mathrm{truth}}(x))$. The probability component uses the mean length-normalized original-answer likelihood on retain questions; on Real Authors and World Facts it normalizes that value by its sum with the perturbed-answer likelihoods, plus 10^{-10}. MU is 100 times the harmonic mean of the nine component means, and is zero if any component is zero. These definitions apply to all reported results.

MUSE-News metrics. MUSE-News contains BBC articles collected after August 2023, split into disjoint forget, retain, and holdout subsets (Shi et al., 2024). KnowMem (knowledge memorization) averages ROUGE-L F1 against reference answers to forget-set questions; VerbMem (verbatim memorization) uses true continuations of forget-set article prefixes. F1 is the harmonic mean of precision and recall. Utility applies KnowMem to the retain set. Lower forget scores and higher Utility indicate less measured memorization and better retention, respectively. The retain-only reference contextualizes suppression.

Toxicity protocol. RealToxicityPrompts uses Llama-2-7B and Detoxify scores (Gehman et al., 2020; Touvron et al., 2023), with 25 continuations per prompt. Maximum toxicity averages per-prompt maxima; mean toxicity averages continuation scores. Toxicity probability is the fraction of prompts with at least one continuation classified as toxic. Scores are multiplied by 100; the binary threshold and generation hyperparameters are unspecified. CoT denotes chain-of-thought prompting; Quark is the named toxicity-control baseline.

Baseline and auxiliary settings. Gradient ascent (GA), direct preference optimization (DPO), and negative preference optimization (NPO) use AdamW at 10^{-5} for 10 epochs. Unlearning from logit difference (ULD) trains its auxiliary at 10^{-4} for 20 epochs, with signed logit coefficient $w_{\mathrm{ULD}} = -1.7$ on TOFU-1%, $w_{\mathrm{ULD}} = -1.5$ on TOFU-5%/10%, and $w_{\mathrm{ULD}} = -0.3$ on MUSE-News. Separate hyperparameters for the simplified NPO baseline (SimNPO) and representation misdirection for unlearning (RMU) are unspecified. CURE extracts eight auxiliary layers while retaining the target's full depth; Appendix E.1 gives its

adaptation settings. Qwen3-8B uses full fine-tuning and the same TOFU protocol. Evaluation uses the official benchmark scripts. FLAT-TV denotes the total-variation variant of the FLAT loss-adjustment objective, and ASU-KL denotes attention smoothing unlearning with retain-side KL regularization. The GD suffix in SimNPO-GD denotes retain-side gradient descent. BF16 and FP32 denote bfloat16 and 32-bit floating-point arithmetic, respectively.

Tables 1 and 2 consolidate the benchmark results. The following protocols record baseline training and evaluation settings. Appendix F separately records decoding ablations and robustness probes.

E.3 TOFU Baseline Protocol

All methods in Table 1 start from the same Llama-3.1-8B-Instruct full-data checkpoint. All six training runs use seed 0 and their fixed final checkpoints: 6, 31, and 62 optimizer updates for the 1%, 5%, and 10% splits. These are adaptations of the author objectives to this backbone, not replications of numerical results on a different model.[1]

Evaluation. Generation uses the checkpoint's chat template at most 60 new tokens; likelihood evaluation uses at most 512 tokens with an answer mask. Evaluation uses BF16 model inference, FP32 likelihood arithmetic, and eager attention. Fgt and Rtd are mean ROUGE-L recall with stemming. The forget sets contain 40/200/400 questions; the fixed retain, Real Authors, and World Facts sets contain 400/100/117. MU is the harmonic mean of nine components: probability, ROUGE-L recall, and Truth Ratio utility on each of these three utility sets. All supplied answer perturbations enter Eq. equation 68. All columns use a 0–100 scale, with FQ equal to 100 times the raw KS p-value. The split-specific retain-reference logs were fixed by the saved evaluation configuration and checked for question–answer identity, rather than selected by their resulting p-values.

Training and scope. Training uses full-parameter AdamW, learning rate 10^{-5} with linear decay, weight decay 0.01, and nominal batch 32; epoch-tail updates retain their actual sample counts. The fixed step counts must not be described as five complete passes over each dataset. No variation across training seeds is estimated. All reported results use the evaluation protocol above, so the overview is protocol-matched across methods. FQ is evidence from a distributional test on the fixed reference logs, not a fraction of erased information or a proof of recovery. Model, data, evaluator, per-question output, and result hashes are recorded with the accompanying artifacts.

E.4 MUSE Baseline Protocol

Table 2 reports all methods under the single evaluation protocol specified in this appendix, for both News and Books. The evaluation covers the released Original and Retrain reference models, the author-released SimNPO-GD checkpoints, and our GA, NPO, ULD, ASU-KL, and FLAT-TV runs, together with CURE. News uses the official Llama-2-7B-derived target and Books the official ICLM-7B-derived target. All full-model runs initialize from the corresponding domain target; ULD adapts a separately extracted auxiliary. Every reported cell, including privacy scores, is measured under the protocol below.

Metrics and evaluation. Know, Verb, and Util abbreviate knowledge memorization, verbatim memorization, and retained utility; they are ROUGE-L F1 multiplied by 100. Priv is a signed relative deviation in area under the receiver operating characteristic curve (ROC AUC): closer to zero is preferable, and values are not clipped to $[0, 100]$. The arrows denote smaller ($\downarrow$), larger ($\uparrow$), or closer to zero ($\rightarrow 0$) as the preferred direction.

[1] Author code: FLAT, commit `bf2ef075`; ASU, commit `2bdc3628`.

We use the official MUSE data and metric definitions (Shi et al., 2024), with 100 examples per generation metric and 100 privacy texts in each of forget, retain, and holdout: 600 scored records per fully evaluated condition. Know and Util use the official 10-shot prompts at most 32 new tokens; Verb uses the published continuation prompts and at most 128 new tokens. ROUGE-L F1 is averaged per example without stemming. Evaluation uses FP32, eager attention, batch 1, and no chat template. Our explicitly fixed CUDA backend and dependency versions are runtime adaptations; they are not a claim of bitwise reproduction of the original authors' environment.

For each privacy text, let n_{tok} be its number of scored target tokens. Min40 averages the lowest $\lfloor 0.4 n_{\mathrm{tok}} \rfloor$ causal token log probabilities, where $\lfloor \cdot \rfloor$ rounds down to an integer. The main ROC uses forget labels 0 and holdout labels 1, without reversing its orientation. With measured AUC A_U for the evaluated model, we report

$$\mathrm{Priv} = 100 \left(\frac{A_U}{A_R} - 1 \right), \qquad A_R = \begin{cases} 0.4772, & \text{News}, \\ 0.5393, & \text{Books}. \end{cases} \tag{69}$$

These denominators are the fixed published retrain constants, not refitted to our new Retrain evaluations. Hence the latter have small nonzero Priv. Positive and negative deviations both indicate disagreement with that reference; a more negative value is not necessarily better privacy.

Privacy evaluation retains every shifted target. Records with 2049 token IDs including the beginning-of-sequence (BOS) token use 2048 forward positions and score all 2048 targets, omitting only the unused terminal output position; shorter records retain the ordinary full forward. No sliding window, target truncation, or rotary position embedding (RoPE) scaling is used. Same-shape causal checks and official-code reference checks pass at absolute tolerance 10^{-6}, including the first actual boundary record in each privacy group. Cross-shape numerical comparisons remain recorded diagnostics, not claims of bitwise agreement with a native 2049-position forward.

Training. The Books GA and NPO runs and both ASU-KL runs use full-parameter BF16 training with FP32 optimizer states, raw 2048-token blocks, nominal batch 32, constant AdamW learning rate 10^{-5}, and the fixed final checkpoint after 10 epochs (130 News or 180 Books updates). They do not train on evaluation question–answer (QA) or continuation records or select checkpoints on test scores. GA and NPO are forget-only, with weight decay 0; NPO uses the paper's sequence log-likelihood ratio with scale $\beta_{\mathrm{NPO}} = 0.1$, distinct from CURE's auxiliary-loss weight β, rather than the raw-logit variant found in the reference repository (Zhang et al., 2024). ASU-KL uses weight decay 0.01, attention temperature 2.4, forget weights 0.3/0.001 for News/Books, and retain-KL weight 1. SimNPO-GD uses the fixed author checkpoints containing retain-gradient regularization (Fan et al., 2025); their training seeds are not disclosed and their complete training histories are not reconstructed here. Evaluation uses seed 0. GA, NPO, ASU-KL, FLAT-TV, and ULD are our single-seed (seed 0) runs; Original and Retrain are released reference models, not newly trained seed-0 runs. No across-seed uncertainty is claimed.

FLAT-TV text adaptation. Each raw 2,048-token forget block shares its first 201 tokens with a refusal response sampled once from the authors' 49 copyright-refusal templates using seed 0. The loss is half the difference between the tanh-transformed mean token probabilities of the forget continuation and refusal, averaged across examples. There is no retain objective or reference model. Full-parameter BF16 training uses FP32 token probabilities and optimizer states, AdamW with constant learning rate 10^{-5}, zero weight decay, and gradient clipping at 1.0. Microbatches of four accumulate to batches of at most 32, weighting partial batches by their actual size. We evaluate the fixed final checkpoint after 10 epochs (130 News or 180 Books updates), without selection on test scores. This is a raw-text adaptation of FLAT-TV, not a reproduction of the authors' MUSE numbers.

Raw-text ULD adaptation on Books. We extract the first eight layers of the Books target, retaining its embeddings, final normalization, and output head, and adapt this auxiliary independently with rank-16 LoRA (scale 32, dropout 0.05). AdamW uses learning rate 10^{-4}, weight decay 0, and nominal forget batch 32 for 20 epochs (360 updates). Training uses raw forget and retain1 articles, split into windows of at most 2047 body tokens with a prepended BOS token, without augmentation or cross-article concatenation. The loss combines cross-entropy on forget windows with five times KL from the uniform vocabulary distribution to the auxiliary prediction on paired retain1 windows. Each branch averages over its valid causal positions. Retain windows are paired deterministically by cycling through their canonical order across forget epochs. Decoding applies softmax to the factual logits minus 0.3 times the auxiliary logits, without a top-logit filter. This coefficient is fixed, not a validated Books optimum. Both generation and privacy scoring use the same combined distribution. Final-checkpoint evaluation passes the dual-model reference, causal, and boundary checks. This raw-text adaptation differs from the News ULD configuration described in Appendix E.2.

Scope. All runs describe their specified final configurations without hyperparameter optimization. All methods are evaluated under the same protocol, so cross-method comparisons within each domain are matched. These measurements do not address longer-generation behavior, MUSE scalability, or sequential unlearning.

E.5 Additional Backbones and Behavioral Evaluation

On Qwen3-8B, CURE obtains MU of 74.0 and FQ of 68.1, with retain ROUGE-L of 79.7 versus the retain-only reference's 87.7. The second backbone therefore also exhibits a utility trade-off. On RealToxicityPrompts (Gehman et al., 2020), CURE has the lowest maximum, probability, and mean toxicity among the listed methods. This behavioral evaluation has no retain-only oracle; Appendix E.2 specifies its 25-generation protocol.

Table 4: Qwen3-8B results on TOFU-10%.

Method	Forget R-L	Retain R-L	MU	FQ
Original	90.9	86.4	70.1	2×10^{-15}
Retain-only	37.5	87.7	65.2	100
GA	0.0	0.0	0.0	10^{-237}
NPO	43.6	72.4	72.3	2×10^{-5}
DPO	65.3	80.2	58.0	10^{-7}
ULD	36.8	42.2	36.0	0.5
CURE	32.8	79.7	74.0	68.1

Table 5: RealToxicityPrompts results with Llama-2-7B. All three scores are multiplied by 100; lower values indicate less toxicity under the reported generation protocol.

Method	Maximum toxicity	Toxicity probability	Mean toxicity
Original	21.1	2.05	3.06
CoT	21.3	2.07	3.02
Quark	20.7	2.70	3.40
GA	20.9	1.83	2.89
ULD	24.7	1.01	3.79
CURE	17.3	0.85	2.74

E.6 Approximation Diagnostics

Using empirical conditional frequencies as references on 1,000 contexts sampled from generation trajectories, the reported average L_1 discrepancies are approximately 0.052 for the trained factual predictor and 0.048 for the auxiliary predictor. Table 6 summarizes a separate model-scale study. PPL denotes perplexity, the exponential of mean token NLL. These are comparisons to empirical frequency proxies: without their count construction, coverage, and relation to the ideal laws, they do not separately identify the realization or alignment errors in Theorem 3.1. The controlled study in Appendix D supplies known population references for its synthetic setting only. The scale and auxiliary-convergence results remain separate from the main benchmark comparisons.

Table 6: Model-scale diagnostics on TOFU-10%. These measurements form a separate study from Table 1.

Source model label	Forget R-L	Retain R-L	MU	FQ	PPL	Median L_1
Llama-3-1B	37.9	66.5	66.0	46.8	4.8	0.416
Llama-3-3B	42.9	79.9	72.8	75.8	4.4	0.271
Llama-3-8B	38.3	98.4	81.4	99.4	3.6	0.048

Table 7: MUSE-News auxiliary-convergence study, reported separately from the main comparison.

Training	KnowMem	VerbMem	Utility	PPL
1 epoch	14.2	9.27	57.7	5.15
3 epochs	14.2	9.51	58.3	4.86
5 epochs	14.4	9.84	59.8	4.65

E.7 Preparation and Inference Cost

Table 8: TOFU-10% preparation and inference costs. Time is measured in minutes and peak memory in GB.

	Preparation		Inference	
Method	Time	Peak memory	Time	Peak memory
DPO	88 min	81 GB	35 min	15 GB
NPO	54 min	81 GB	35 min	15 GB
GA	62 min	56 GB	35 min	15 GB
ULD	36 min	15 GB	72 min	21 GB
CURE	7 min	54 GB	50 min	54 GB

CURE uses vectorized candidate scoring and parallel dual-model forward passes. These timings apply to the measured workload; comparisons across workloads must account for batch size, beam width, output length, cache policy, precision, and timing boundaries.

E.8 Sensitivity to Auxiliary Training Weight

Table 9 varies the auxiliary training weight w_{add} on TOFU-10%. Retain ROUGE-L ranges from 98.33 to 98.40, forget ROUGE-L from 35.78 to 39.23, and MU from 81.36 to 82.28, showing limited sensitivity

within this range. The default is $w_{\mathrm{add}} = 0.5$. This weight is the optimization coefficient β in Eq. equation 20; the source amplification κ is held fixed. Under Eq. equation 19, varying β at fixed λ, κ also changes a_0. The reported sweep does not record whether its initialization followed this relation, so its results do not isolate training-weight effects or establish robustness of the coupled design.

Table 9: Auxiliary-weight sensitivity on TOFU-10% (percentage scale). The $w_{\mathrm{add}} = 0.5$ row matches the main configuration of Table 1, up to one-decimal rounding in the main table.

w_{add}	Retain R-L	Forget R-L	MU
0.1	98.40	38.66	81.78
0.3	98.40	39.23	82.03
0.5	98.38	38.32	81.36
0.7	98.33	38.34	82.16
0.9	98.33	35.78	82.28

E.9 Incremental Adaptation to Expanding Forget Sets

The continual-adaptation study expands the forget set from TOFU-1% to TOFU-5% and then TOFU-10%. Starting from the trained 1% auxiliary, each expansion uses one epoch of adaptation with replay from a buffer of retained samples. The updates take 58 and 108 seconds, respectively (Table 10); these costs exclude initial training.

Table 10: Continual auxiliary adaptation as TOFU forget sets expand. CL denotes a one-epoch update with replay. Performance metrics use percentage scaling; the initial training cost was not reported.

Stage	Update cost	Retain R-L	Forget R-L	MU	FQ
1%, fully trained	—	99.5	46.7	80.4	99.0
5%, CL, 1 epoch	58 s	80.3	40.2	79.0	74.5
10%, CL, 1 epoch	108 s	85.2	38.9	79.0	78.1

The shorter updates incur a retention and forget-quality trade-off. At 5%, retain ROUGE-L/FQ is 80.3/74.5, compared with 93.4/86.6 in the full-training study; at 10%, the values are 85.2/78.1 versus 98.4/99.4. The replay-buffer size and composition are unspecified, and these comparisons do not isolate replay from the smaller adaptation budget. The results establish feasibility for this sequence of requests, without establishing equivalence to full retraining or performance on arbitrary deletion sequences.

E.10 Generation Examples

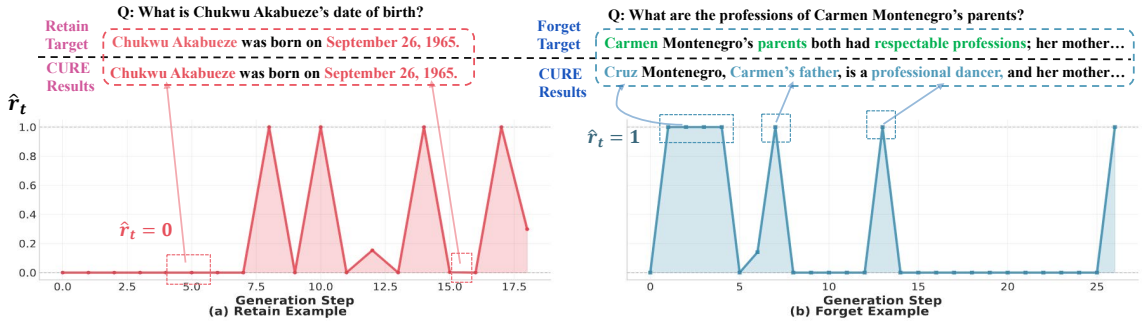


Figure 4: **Context-dependent decoding in two TOFU examples.** The retain example preserves the target date, while the forget example changes the response. The curves show the decoder state s_t (labeled $\widehat{r}_t$ in the plot), which follows the recursion of Algorithm 1 with cap $1 - \eta$ (we use $\eta = 0.1$); the displayed endpoint 1 is the rounded value of this cap. The examples illustrate adaptive behavior without establishing erasure or distributional equivalence.

E.11 Generation Under Alternative Control Schedules

Tables 11 and 12 give the complete responses under constant, linear, sinusoidal, and adaptive controls. As in Appendix E.1, s_t denotes the decoder state. The prescribed alternatives are

$$s_t^{\text{const}} = c, \qquad s_t^{\text{sin}} = \tfrac{1}{2}\sin(0.1t) + \tfrac{1}{2}, \qquad s_t^{\text{lin}} = \max(0, 1 - t/n), \tag{70}$$

where t is the generation step, n the prescribed generation length, and c a fixed control value; the superscripts label constant, sinusoidal, and linear schedules, and the sine argument is in radians. This c is unrelated to the shift coefficient c_t in Eq. equation 44. The result table labels the constant settings as $c = 1.0$ for the retain example and $c = 0$ for the forget example. Endpoint treatment and update-direction ambiguity are discussed in Appendix E.1.

Responses use the reported display preprocessing: newlines become spaces, repeated spaces collapse, and ☒ marks each non-displayable character. Ellipses, repetitions, and superscript characters are part of the output, not omissions. ROUGE-L is reported as recall against the reference answer.

Table 11: Retain-example responses under alternative controls. ROUGE-L recall is reported in percent against the reference answer.

Strategy	Reported response	R-L
Question: What is Chukwu Akabueze's date of birth?		
Reference	Chukwu Akabueze was born on September 26, 1965.	100.0
Constant ($c = 1.0$)	Ch... Chukwu Ak... Chukw... Chu... Chukwu Akabue	12.5
Linear	Ch^{3}kwu Akabueze^{3} born^{3} September 26,!988.	50.0
Sinusoidal	Ch… Chukwu Akab… Chukw… Chukw… Chukw… Chukw… Chukw… Chuk	12.5
Adaptive	Chukwu Akabueze was born on September 26, 1965.	100.0

Table 12: Forget-example responses under the same control families. Low overlap indicates reduced reproduction of this answer; it does not itself establish removal of the underlying knowledge.

Strategy	Reported response	R-L
Question: What are the professions of Carmen Montenegro's parents?		
Reference	Carmen Montenegro's parents both had respectable professions; her mother worked as a Waiter/Waitress, while her father was an Optometrist.	100.0
Constant ($c = 0$)	Carmen Montenegro's parents both had respectable professions; her mother worked as a Waiter/Waitress, while her father was an Optometrist.	100.0
Linear	Carm... Carmen Montenegro☒☒'s parents had diverse professions; her mother worked as a Waiter/Waitress, while her father was an Optometrist.	90.5
Sinusoidal	Carmen Montenegro☒☒☒ parents had distinct careers; her mother worked☒☒☒ a Waiter/Waitress, while her father was☒ ☒ ☒ a Optometrist	76.2
Adaptive	Cruz Montenegro, Carmen's father, is a professional dancer, and her mother, Eva Montenegro, is a historian.	23.8

Adaptive control reproduces the retain reference exactly, while the prescribed schedules produce repetitions or altered text. For the forget example, constant control reproduces the reference; adaptive control instead supplies different biographical assertions, with recall of 23.8. These qualitative comparisons illustrate context-sensitive behavior and the limits of output overlap as a forgetting criterion. They neither constitute a test-set ablation nor validate the identification theorem.

F Supplementary Robustness and Implementation Analyses

This appendix collects reported LLM measurements on auxiliary-data perturbation, numerical repair, decoding, repeated extraction, prompt variation, and accelerated inference. We retain distinct evaluation settings where their correspondence to the main benchmark is unspecified. Model-scale and convergence diagnostics, auxiliary-weight sensitivity, and baseline comparisons appear in Appendix E.6, Appendix E.8, and Table 1, respectively.

F.1 Sensitivity to Auxiliary-Data Perturbation

Auxiliary alignment is a property of the induced population law, whereas a perturbation experiment tests the behavior of a particular trained system. Table 13 varies the fraction of perturbed examples in a 400-example auxiliary dataset on TOFU-10%. From 1% to 50% perturbation, forget ROUGE-L increases from 40.39 to 44.11 and MU decreases from 72.49 to 69.59. Retain ROUGE-L varies nonmonotonically between 85.82 and 91.56. These changes show sensitivity to auxiliary-data quality rather than invariance to it.

Table 13: Auxiliary-data perturbation on TOFU-10%. Counts refer to perturbed examples out of 400. The record does not include a matched 0% row.

Perturbation	Examples	Forget R-L	Retain R-L	MU
1%	4/400	40.39	90.36	72.49
5%	20/400	41.62	85.82	70.62
10%	40/400	42.68	89.42	70.05
20%	80/400	43.26	86.43	70.51
50%	200/400	44.11	91.56	69.59

The perturbation operator, selection rule, and seeds are unspecified. Without a matched 0% row, the main benchmark cannot serve as an unperturbed control. This sweep does not isolate ϵ_{align} in Theorem 3.1 or establish a general robustness radius. Similarly, the empirical-frequency diagnostics in Appendix E.6 assess their chosen proxies, rather than certify the reverse-KL calibration in Theorem 5.1.

F.2 Raw Negative Entries and the Reported Repair Map

The raw probability-space inverse can have negative coordinates under model or decoder-state error. On Llama-3.1-8B with TOFU-10% and beam width 4, the reported negative-entry rates are 2.05% on forget prompts and 1.57% on retain prompts, measured over active-beam $\times$ full-vocabulary slots. Table 14 conditions this diagnostic on the recursive decoder state $\widehat{r}_t$. This state is distinct from the auxiliary NLL weight β.

Table 14: Raw negative-entry diagnostics. All entries are percentages. "Bin share" preserves the reported rounded occupancy column (called "mass" in the experiment record); it is not the probability mass assigned to negative tokens.

	Forget prompts		Retain prompts	
State interval	Negative entries	Bin share	Negative entries	Bin share
$[0, 0.1)$	1.23	86.5	0.02	98.4
$[0.1, 0.5)$	5.57	5.9	99.89	0.04
$[0.5, 0.9)$	5.89	1.8	99.98	0.05
$[0.9, 1]$	9.43	5.8	99.48	1.5

Most retain-side slots occur below 0.1, where the negative-entry rate is 0.02%. However, the conditional rates in the larger-state retain bins are nearly 100%. The small aggregate rate therefore does not imply that correction is benign at every context. Entry counts also do not quantify the magnitude of negative scores or the displacement caused by repair; those require additional measurements.

Shift, smooth, and renormalize. The supplementary implementation description specifies the following map for the raw score vector z, before taking logarithms for beam scoring:

$$\mathcal{S}_{\text{rep}}(z)(v) = \frac{z(v) - m(z) + \varepsilon_{\text{rep}}}{\sum_{u \in \mathcal{V}} [z(u) - m(z) + \varepsilon_{\text{rep}}]}, \qquad m(z) = \min_{u \in \mathcal{V}} z(u), \quad \varepsilon_{\text{rep}} > 0. \tag{71}$$

For a finite real score vector, each numerator is at least ε_{rep} and the denominator is positive. The repaired vector is strictly positive and sums to one, including when all raw scores coincide, in which case the result is uniform. The transformation preserves within-context token ordering; it can nevertheless change sampling probabilities and sequence beam scores.

This map differs from the reference repair in Eq. equation 18. The reference shifts only when a negative coordinate exists; Eq. equation 71 subtracts the minimum even from an already nonnegative vector and adds smoothing. Its repair displacement must therefore be assessed using $\|\mathcal{S}_{\text{rep}}(z) - z\|_1$ in Eq. equation 11, rather than assuming the two maps coincide. The reported negative-entry frequency alone does not bound that displacement.

Repair is separate from endpoint handling. Equation equation 71 assumes that the raw scores are finite. An ε added at this stage cannot resolve an earlier division by zero in Eq. equation 17 at $\widehat{r} = 1$. The record specifies the reported cap of 1 and the repair formula, but not the numerical value of ε_{rep} or the exact

protection used in the raw inverse. These remain necessary implementation details. Algorithm 1's strict-margin reference and Theorem 3.1 continue to require a positive η; the empirical cap of 1 does not establish that condition.

F.3 Temperature and State Bounds

Temperature after correction. Table 15 varies the corrected-distribution temperature T with beam width 4. Here $T = 1$ means no scaling. This parameter is distinct from the ratio temperature $\tau = 1$ in Appendix E.1: the density-ratio recursion uses untempered model probabilities. The record does not further specify the post-correction scaling implementation. Increasing T from 1.0 to 1.4 reduces forget ROUGE-L from 38.3 to 32.3, alongside a retain decrease from 98.4 to 95.3; decreasing T to 0.2 raises forget overlap to 66.7. Lower forget overlap alone is not a monotone measure of retain-only recovery, since the main retain-only reference has forget ROUGE-L of 39.6.

Table 15: Corrected-distribution temperature on TOFU-10%, with beam width 4.

T	Forget R-L	Retain R-L
1.4	32.3	95.3
1.2	34.5	95.5
1.0	38.3	98.4
0.8	48.7	97.6
0.6	55.7	98.1
0.4	61.3	98.8
0.2	66.7	99.2

Bounds on the decoder state. Table 16 varies the empirical projection interval. Raising its lower bound from 0 to 0.1 changes forget/retain ROUGE-L from 38.3/98.4 to 66.0/78.7. This is consistent with the usefulness of allowing low decoder states on retain prompts, but the sweep does not isolate a causal mechanism. Extending the interval below zero or above one tests a decoder-state update rule outside the range of a source posterior; those rows are not covered by the ideal recovery interpretation. No initialization or prompt-prefill ablation is supplied by this sweep.

Table 16: Recursive-state bounds on TOFU-10%, with beam width 4. The default cap of 1 requires separate numerical endpoint protection.

State interval	Forget R-L	Retain R-L
$[0, 1]$ (default)	38.3	98.4
$[-0.1, 1]$	46.4	97.5
$[0.1, 1]$	66.0	78.7
$[0, 0.9]$	42.9	97.2
$[0, 1.1]$	42.2	97.2

F.4 Repeated Extraction and Prompt Variation

Deterministic overlap and repeated extraction probe different behaviors. The supplementary extraction study uses the same 400 forget prompts and matched stochastic-sampling settings across methods. ExactLeak@k is the percentage of prompts for which an exact target answer is recovered within k sampled completions. Table 17 reports both one and five attempts. Exact-match normalization, sampling hyperparameters, and run-level outputs are not provided in the available record.

Table 17: Repeated exact-answer extraction on 400 forget prompts. Lower values indicate less observed leakage under this sampling protocol.

Method	ExactLeak@1 (%)	ExactLeak@5 (%)
Full model	92.75	95.75
Retain90	0.00	0.00
SimNPO	8.00	9.50
CURE	2.50	4.25

CURE has lower observed repeated-extraction rates than the full model and SimNPO, but its ExactLeak@5 remains 4.25%, versus 0% for the retain-only reference in this study. Increasing the sample budget exposes additional residual leakage. Thus the measurements support reduced extraction under the tested protocol, not unrecoverability, parameter erasure, or a formal privacy guarantee. Zero observed leakage for a reference on 400 prompts is likewise a finite-sample result.

Lightweight prompt wrappers. Four wrappers produce 1,600 prompt variants, evaluated with deterministic beam width 4. Table 18 retains the record's identifiers W1–W4 because their exact texts are unspecified. CURE's wrapper-average forget ROUGE-L is 38.60, compared with 38.30 on the original prompts; individual wrapper scores range from 36.68 to 40.11. A separate semantic-paraphrase probe reports CURE forget ROUGE-L of 33.01, without specifying the paraphrase generator or evaluation count.

Table 18: Forget ROUGE-L under lightweight prompt wrappers. The average covers W1–W4. This is a separate robustness evaluation, with its own reported reference scores.

Method	Original	W1	W2	W3	W4	Average
Full model	99.31	99.08	98.42	98.64	98.49	98.66
Retain90	16.69	16.61	16.76	16.69	16.60	16.67
SimNPO	70.06	70.14	68.56	69.06	69.11	69.22
CURE	38.30	40.11	37.52	36.68	40.10	38.60

The Retain90 original-prompt score here is 16.69, compared with 39.6 for the main table's retain-only reference. The record does not specify the checkpoint or evaluation differences needed to reconcile these values, so we do not merge the two evaluations or substitute one reference for the other. The wrapper and paraphrase results assess limited surface-form variation. They do not cover adaptive jailbreak search, multi-turn elicitation, or unrestricted extraction of long documents.

F.5 Accelerated Inference Prototype

Table 19 reports an implementation study on TOFU-10%, using a single NVIDIA A800 and timing 1,000 samples. The vectorized, parallel implementation corresponds to the 50-minute inference configuration in Appendix E.7. The additional prototype combines a top-32 candidate shortlist, key–value caching, and speculative draft–verify execution. It drafts four tokens using the factual model, invokes ratio-based auxiliary verification, and falls back to shortlist correction when verification fails.

Table 19: Reported inference time and peak memory for three CURE implementations on the same single-A800 workload. The prototype is separate from the main benchmark configuration.

Implementation	Time / 1,000 samples	Peak memory
Original implementation	70 min	59 GB
Vectorized scoring + parallel forwards	50 min	54 GB
+ shortlist, cache, draft–verify	22 min	22.4 GB

The prototype's reported time is approximately $3.18\times$ faster than the original implementation. These timings do not establish preservation of the full-vocabulary corrected distribution: the record supplies neither a lossless verification argument nor matched quality measurements for the prototype. Batch size, output-length distribution, precision, and detailed timing boundaries are also needed for replication. Consequently, the 22-minute result is an implementation opportunity rather than a replacement for the main quality–cost comparison. The 7–8B backbone experiments establish the measured scale only; the model-scale diagnostics do not by themselves predict performance or cost at larger scales.

F.6 Interpretation and Remaining Scope

These measurements address predictive unlearning in the dataset-removal sense, not structural causal identification. The inverse is closed form conditional on the source posterior; BOS prefill computes the decoder state without making the prompt-level source posterior an input. Its exact source-posterior interpretation still requires calibrated BOS initialization and aligned joint observation laws. Reverse-KL anchoring likewise does not certify ideal mixture alignment. Theorems 5.1 and 3.1 state the relevant conditions and separate errors. Beyond completing the missing protocol details, useful tests include initialization and prefill ablations, repair-displacement traces, repeated training seeds, and adaptive extraction. The post-trained 8B results do not establish exact retraining equivalence or performance at larger scales.