RAISE: POST-TRAINING MEMBERSHIP INFERENCE MITIGATION VIA MODEL EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Membership inference attacks (MIAs) pose privacy risks to fine-tuning data. After fine-tuning, a privacy audit may reveal that particular fine-tuning records carry strong membership signals. Yet applying training-time defenses at this point would require rerunning fine-tuning. We present a novel post-training approach that formulates MIA mitigation as a model-editing problem. Given selected member records, our approach, RAISE (**R**eference-**A**ligned **I**nference-**S**ignal **E**diting), directly edits the fine-tuned checkpoint without rerunning fine-tuning. Our analysis shows that the membership signal is concentrated in a small number of token positions with unusually low losses relative to non-member references. RAISE localizes the corresponding spans and uses constrained checkpoint editing to align their losses with ranges derived from non-member references, while limiting changes to unrelated model behavior. Across six domains and ten MIAs, RAISE reduces mean attack AUC from 0.855 to 0.576 and mean TPR at 1% FPR from 25.1% to 2.8%. It also outperforms the evaluated post-training baselines, generalizes across five model configurations, and remains effective against adaptive attacks (AUC below 0.58), while retaining substantial fine-tuning utility.

1 INTRODUCTION

Fine-tuning adapts general-purpose language models to proprietary, personal, or otherwise sensitive data. This adaptation can induce systematic differences in model behavior between records that participated in fine-tuning and those that did not. Membership inference attacks exploit these differences to infer whether a candidate record was included in the fine-tuning set (Shokri et al., 2017; Yeom et al., 2018; Carlini et al., 2021). For models adapted on sensitive corpora, such inferences can reveal unwanted associations between particular records and the model's fine-tuning data.

Most defenses against this risk assume that protection is planned before or during training. Differentially private and token-aware defenses modify the training procedure itself (Abadi et al., 2016; Li et al., 2024; Zhang et al., 2025b; Tran et al., 2025); other approaches construct an additional protected model through distillation (Li et al., 2026) or perturb the outputs exposed at inference time (Jia et al., 2019; Fu et al., 2026). These approaches are poorly matched to a common post-training scenario: a model has already been fine-tuned, a privacy audit identifies particular member records with strong membership signals, and the operator wants to mitigate their exposure directly in the existing checkpoint. In this setting, rerunning the original training pipeline can be expensive or impractical, while a serving-time defense introduces a persistent deployment dependency.

This raises a natural question:

> *Can membership risk for selected fine-tuning records be reduced directly in an existing checkpoint, while preserving the useful behavior acquired during fine-tuning?*

A key challenge is that membership evidence is not a single model behavior that can be removed uniformly. Likelihood-based MIAs aggregate token-level prediction statistics over an entire record (Carlini et al., 2021; Shi et al., 2024; Zhang et al., 2025a; Xie et al., 2024), but different token predictions contribute very differently to the resulting membership signal. Moreover, low loss alone is not sufficient to identify privacy-relevant behavior: some tokens are naturally predictable regardless

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.

of membership. An effective post-training intervention therefore requires two decisions: *which local predictions carry the strongest membership-related signal*, and *how those predictions should be adjusted*.

Our analysis reveals a simple but important structure in this problem. Membership-related prediction discrepancies are highly localized when member losses are measured relative to same-domain non-member references. A small fraction of token positions captures a disproportionate share of the positive member–reference loss gap, and this concentration persists under disjoint reference sets. More importantly, score-space counterfactuals show that adjusting these high-gap positions toward held-out non-member behavior reduces record-level Loss-attack separability more effectively than modifying an equal number of randomly selected or lowest-loss positions. These results identify a clear intervention principle: *membership mitigation should focus on predictions that are unusually favorable to members relative to the behavior supported by non-member references*.

Motivated by this principle, we introduce RAISE (**R**eference-**A**ligned **I**nference-**S**ignal **E**diting), a post-training defense that treats membership mitigation for selected records as a localized model-editing problem. Given a fine-tuned checkpoint, a set of member records to protect, and a same-domain non-member reference pool, RAISE first identifies contiguous spans whose losses are unusually low relative to reference behavior. Rather than assigning an arbitrary suppression target, it constructs for each span a bounded loss interval supported by non-member reference quantiles. RAISE then selects a write location by considering both where member–reference separation is exposed and how sensitive reference behavior is to intervention. Finally, it optimizes a hidden-state target that moves the selected span into the reference-supported range and realizes this target through a constrained parameter update. The result is an edited checkpoint that directly modifies the model itself, without requiring access to the complete fine-tuning dataset or rerunning fine-tuning.

We evaluate RAISE on six MIMIR domains against ten representative membership inference attacks, covering both reference-free and reference-based attack families. Across all attack–domain pairs, RAISE reduces mean AUC-ROC from 0.855 to 0.576 and mean TPR at 1% FPR from 25.1% to 2.8%. It also achieves substantially lower attack performance than the evaluated post-training baselines, generalizes across five model configurations, and remains effective against adaptive attacks designed around its editing mechanism, for which the maximum defended mean AUC in our sweep remains below 0.58. At the same time, the edited models retain substantial domain-specific utility acquired through fine-tuning. Ablations further show that reference-guided localization improves over random and lowest-loss selection, that the choice of write location materially affects the privacy–utility trade-off, and that protection improves as a larger fraction of vulnerable members is edited.

Contributions. Our main contributions are:

- We show that member–reference prediction discrepancies are concentrated in a relatively small subset of token positions. Using disjoint references and score-space counterfactuals, we demonstrate that these high-gap positions provide more effective intervention targets than random or lowest-loss positions under matched token budgets.
- We formulate protection of selected known members as a post-training checkpoint-editing problem and introduce RAISE, which localizes anomalous spans, derives bounded targets from non-member reference behavior, selects a privacy-aware write location, and realizes the desired prediction changes through constrained parameter updates.
- We evaluate RAISE across six domains, ten MIAs, and five model configurations, compare it with representative post-training defenses, study adaptive attacks tailored to its mechanism, and quantify the resulting utility trade-offs and key design choices.

2 BACKGROUND AND THREAT MODEL

2.1 BACKGROUND AND RELATED WORK

Membership Inference in Language Models.

Membership inference determines whether a candidate record participated in a model's training process by exploiting systematic behavioral differences between members and non-members (Shokri

et al., 2017; Yeom et al., 2018). For language models, many effective attacks are built on likelihood-related signals: training records can receive higher token probabilities, and therefore lower losses, than comparable non-members (Carlini et al., 2021; Mattern et al., 2023). Existing attacks aggregate these signals in different ways, including sequence likelihoods, compression-based calibration, reference-model comparisons, and neighbourhood-based scores (Carlini et al., 2021; Mattern et al., 2023). More recent methods emphasize particular subsets of tokens, normalized token-level statistics, or changes in likelihood under alternative contexts (Shi et al., 2024; Zhang et al., 2025a; Xie et al., 2024; Wang et al., 2025). These attacks highlight an important distinction for mitigation: *low loss alone does not imply membership-specific behavior, since some text is intrinsically easy to predict.* Reference-based comparisons instead reveal when a member receives unusually favorable predictions relative to an appropriate non-member baseline (Mattern et al., 2023). RAISE builds on this observation, using same-domain non-members to identify localized member–reference discrepancies and to define the prediction behavior toward which those discrepancies should be corrected.

Defenses against Membership Inference Attacks. Existing MIA defenses can be distinguished by *when* and *where* they intervene. Training-time approaches incorporate privacy protection directly into optimization. Differentially private training provides formal record-level guarantees under a privacy budget (Abadi et al., 2016), while empirical defenses such as MIST and DuoLearn modify the training objective to reduce membership signal (Li et al., 2024; Tran et al., 2025). SOFT instead selectively obfuscates fine-tuning data before optimization (Zhang et al., 2025b). These methods can be effective when privacy protection is planned in advance, but require access to and modification of the original training pipeline. Post-training defenses operate after the target model has already been trained, but differ substantially in how protection is realized. GUARD constructs a separate defended model through distillation from a pretrained-model-anchored target distribution (Li et al., 2026). Serving-time defenses such as MemGuard (Jia et al., 2019), EPD (Fu et al., 2026), and Gaussian Logit Noise (Rahimian et al., 2020) alter the outputs exposed to an attacker during inference. Direct parameter-update methods, including NPO (Zhang et al., 2024) and Forget CE (Maini et al., 2024), can modify an existing checkpoint, but are designed primarily around forgetting or likelihood suppression of designated records.

RAISE targets *a different post-training objective*: reducing the empirical member–non-member distinguishability of selected known members directly in the existing checkpoint. Rather than globally suppressing their likelihoods, it identifies reference-relative prediction anomalies and moves them toward behavior supported by non-member references. This makes membership distinguishability itself the object of the checkpoint edit.

Model Editing. Model editing provides a natural mechanism for implementing targeted behavioral changes in a pretrained or fine-tuned model while limiting interference with unrelated predictions. Parameter-based editing methods such as ROME, MEMIT, and AlphaEdit directly modify selected model weights to realize targeted behavioral changes while limiting interference with preserved knowledge (Meng et al., 2022; 2023; Fang et al., 2025). Privacy-oriented model editing has applied targeted interventions to privacy-related neurons (Wu et al., 2023; 2024), sensitive-token representations (Ashuach et al., 2025), PII-leakage circuits (Hughes et al., 2026), and memorized private information (Ruzzetti et al., 2025). These works demonstrate that privacy behavior can be modified through localized parameter interventions, but their primary target is sensitive content or memorized information. RAISE instead uses model editing to address *membership distinguishability*. It converts reference-relative membership signals into explicit hidden-state targets and writes those targets into the checkpoint through constrained parameter updates. In this way, model editing serves not merely as an implementation tool, but as the mechanism that connects diagnosed membership behavior to a localized post-training correction.

2.2 Problem Setup and Threat Model

Problem Setup. We study post-training membership mitigation for a completed fine-tuning event. Let D_{FT} denote the dataset used to fine-tune a language model, and let M_0 denote the resulting fine-tuned checkpoint. For a candidate record x, we define its membership label as $y(x) = \mathbb{I}[x \in D_{\mathrm{FT}}]$. Membership is therefore defined with respect to this particular fine-tuning event, regardless of whether the same record may also have appeared during pretraining.

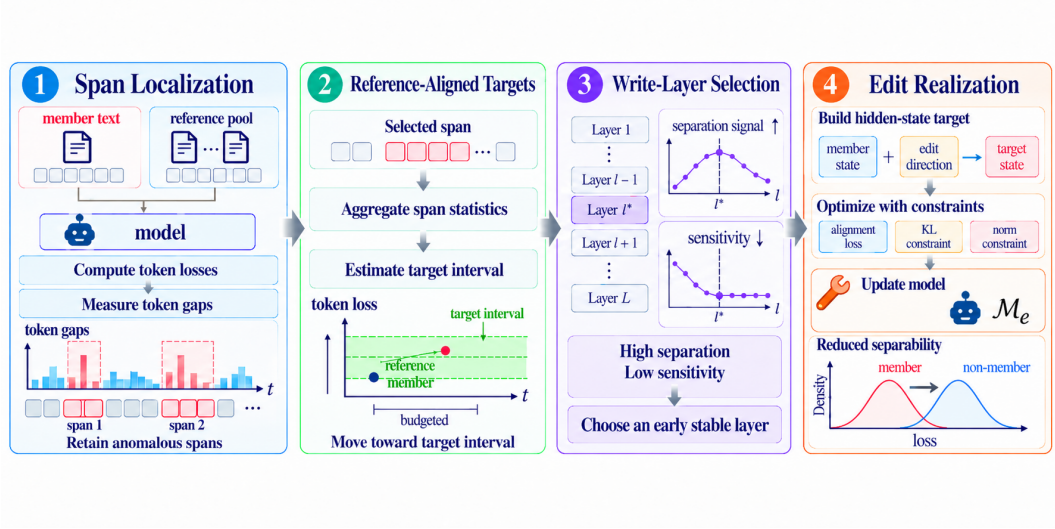


Figure 1: Overview of RAISE

After fine-tuning, the model owner identifies a set of known members $S \subseteq D_{\mathrm{FT}}$ for protection. The goal is to transform M_0 into an edited checkpoint M_e such that records in S exhibit lower empirical membership distinguishability, while preserving the useful behavior acquired through fine-tuning. The defender has white-box access to M_0 and can directly update its parameters. In addition, the defender has access to a same-domain non-member development pool $D_{\mathrm{dev}}^{\mathrm{def}}$, from which reference subsets $B \subseteq D_{\mathrm{dev}}^{\mathrm{def}}$ are sampled to characterize non-member prediction behavior and construct reference-aligned edit targets. Formally, the post-training mitigation problem can be viewed as

$$(M_0,\, S,\, D_{\mathrm{dev}}^{\mathrm{def}}) \;\longrightarrow\; M_e.$$

The procedure operates directly on the existing checkpoint and does not require access to the complete fine-tuning dataset or repetition of the original fine-tuning process. Its protection scope is the supplied set S: RAISE aims to reduce membership signal for selected known members rather than uniformly protecting every record in D_{FT}.

Threat Model. The adversary aims to infer whether a candidate record x participated in the target fine-tuning event (Duan et al., 2024; Fu et al., 2024; Zhang et al., 2025b). After mitigation, the adversary interacts with the deployed edited model M_e and computes a membership score using the information available to the corresponding attack. We consider the range of capabilities assumed by representative language-model MIAs. Target-only attacks may use observed-token probabilities or logits (Carlini et al., 2021; Shi et al., 2024), while other attacks may additionally require the full next-token distribution (Zhang et al., 2025a), an external pretrained or reference model (Carlini et al., 2021; Mattern et al., 2023; Fu et al., 2024), or auxiliary member and non-member records for calibration or conditioning (Xie et al., 2024; Wang et al., 2025; Zhang et al., 2025b). We grant each attack the auxiliary information required by its original definition, rather than assuming that all attackers share a single uniform set of resources.

Our primary evaluation therefore tests whether membership information remaining in M_e can still be exploited by attacks with different scoring rules and auxiliary information. We further consider adaptive attacks in which the adversary knows the general design of RAISE and constructs attack variants around its editing mechanism, while not observing run-specific defender information such as the sampled reference records or selected edit spans.

3 REFERENCE-ALIGNED INFERENCE-SIGNAL EDITING

Framework overview.

3.1 MOTIVATION

Our goal is to reduce membership-distinguishing behavior in selected member records through targeted checkpoint edits. A central challenge is that record-level MIA scores aggregate prediction behavior over an entire sequence, while the underlying membership signal need not be distributed uniformly across tokens. Moreover, low loss alone is not a reliable indicator of membership-specific behavior, since many tokens are intrinsically easy to predict. We therefore seek predictions that are not merely low-loss, but *unusually favorable to members relative to comparable non-member behavior*. This leads to two key questions: *where is the membership-related discrepancy concentrated, and toward what prediction behavior should it be adjusted?*

We study these questions on ArXiv using same-domain non-member references. To separate localization from evaluation, token positions are selected using one reference subset and assessed using a disjoint held-out subset. The results reveal a pronounced concentration of reference-relative membership signal: at a 10% token budget, positions with the largest member–reference loss gaps capture 23.37% of the held-out positive-gap mass, compared with 10.13% for random selection and 10.48% for lowest-loss selection. Thus, a relatively small subset of predictions accounts for a disproportionate share of the member's advantage over non-members, and these positions cannot be identified as effectively from absolute loss alone. We further test whether these localized discrepancies are meaningful intervention targets. In a score-space counterfactual, moving the selected high-gap positions toward held-out non-member reference losses reduces record-level Loss-attack separability more than modifying the same number of randomly selected or lowest-loss positions. Together, these observations motivate *selective reference alignment*: identify local predictions that are anomalously favorable to members and move them toward a range supported by non-member reference behavior. This principle directly motivates the span localization and reference-aligned target construction used by RAISE. Full experimental protocols and analyses are provided in Appendix A.

3.2 RAISE FRAMEWORK

Given a fine-tuned checkpoint M_0, a supplied set of known members S, and same-domain non-member references $B \subseteq D_{\text{dev}}^{\text{def}}$, RAISE produces an edited checkpoint M_e. As illustrated in Figure 1, the framework addresses four questions in sequence: *where* membership-related prediction discrepancies are concentrated, *toward what behavior* they should be adjusted, *where in the network* the intervention should be realized, and *how* the requested change can be written into the checkpoint while limiting interference with unrelated behavior.

Span localization. We first identify local predictions that are unusually favorable to a member relative to non-member reference behavior. For a member record x, let $\ell_t(x; M_0) = -\log p_{M_0}(x_t \mid x_{<t})$ denote the observed-token loss, and let $Q_t(q; x, B)$ denote the q-quantile of its corresponding reference-loss distribution. Reference estimates use token-conditioned statistics when sufficiently supported and otherwise back off to broader reference statistics.

For a contiguous span s, define its mean member loss and averaged reference quantile as

$$\bar{\ell}_s = \frac{1}{|s|} \sum_{t \in s} \ell_t(x; M_0), \qquad Q_s(q) = \frac{1}{|s|} \sum_{t \in s} Q_t(q; x, B).$$

We score the span by its standardized reference-relative deviation:

$$g_s = \frac{Q_s(0.5) - \bar{\ell}_s}{\max(Q_s(0.75) - Q_s(0.25), \epsilon)}, \tag{1}$$

where $\epsilon > 0$ stabilizes the denominator. A positive g_s indicates that the member span receives lower loss than the behavior supported by its non-member references, normalized by the local reference-loss spread. We prioritize spans with strong positive standardized gaps and substantial positive member–reference discrepancy, and greedily resolve overlaps to obtain a compact set of editing targets. Reference construction and span-selection details are provided in Appendix B.2.

Reference-aligned target construction. For each retained span, we define a reference-supported target interval

$$[\tau_s^-, \tau_s^+] = [Q_s(q^-), Q_s(q^+)], \tag{2}$$

where $q^- \leq q^+$ are fixed reference quantiles. Rather than forcing the edited prediction to match a single reference value, this interval specifies a bounded range of behavior supported by non-members. RAISE requests only positive loss movement toward this range: spans that already require no upward adjustment generate no edit request. The requested movement is further limited by per-span and per-record budgets, with excessive requests clipped or omitted. Full target-construction and budgeting rules are given in Appendix B.3.

Privacy-aware intervention-location selection. The reference-aligned targets specify what prediction behavior should change, but not where the change should be introduced in the network. We seek a representation layer at which member–reference separation is already exposed while reference behavior remains relatively insensitive to intervention. For each candidate layer l, we measure member–reference separation $A_{\text{sep}}(l)$ using sequence losses obtained by projecting its intermediate states through the final normalization and language-model head. We separately estimate reference sensitivity $U(l)$ from activation and reference-loss gradient norms. After identifying a candidate region from the separation profile, we select the earliest layer l^* whose sensitivity is within a fixed tolerance of the minimum in that region. This favors an intervention point that exposes the relevant membership discrepancy without unnecessarily choosing a more sensitive representation. Definitions and the complete selection rule are provided in Appendix B.

Reference-aligned edit realization. For each retained span, let h_s denote its anchor representation at the selected layer l^*. We optimize a local perturbation δ_s and form a target representation $z_s = h_s + \delta_s$. With the checkpoint parameters fixed, the target search moves the resulting mean span loss into the reference-supported interval $[\tau_s^-, \tau_s^+]$, while regularizing changes to the model's output distribution and constraining the perturbation magnitude.

Once a target representation is obtained, we realize it through a constrained parameter update. The update maps the corresponding request representation toward z_s while restricting parameter changes to a subspace estimated from auxiliary preservation representations, reducing interference with unrelated behavior. Full target-search objectives, norm constraints, and projected writeback details are provided in Appendix B.5 and Appendix B.6.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Dataset. Following prior work (Zhang et al., 2025b), we construct our evaluation using MIMIR (Duan et al., 2024). MIMIR derives candidate texts from the training and test partitions of the Pile and provides n-gram-filtered splits that control lexical overlap between member and non-member samples. Like prior work (Zhang et al., 2025b), we evaluate our approach on six subsets provided by MIMIR: *ArXiv*, *HackerNews*, *PubMed*, *Pile-CC*, *Wikipedia*, and *GitHub*, using the 13-gram (`ngram_13_0.8`) split as the default.

Models. We conduct experiments with **Llama-3-8B** (Grattafiori et al., 2024), **Llama-3.2-3B** (Meta AI, 2024), **GPT-J-6B** (Wang & Komatsuzaki, 2021), **Qwen3-8B** (Yang et al., 2025) and **Pythia-6.9B** (Biderman et al., 2023). We select these models because they are publicly available, widely used, and represent a range of different architectures. Our primary results are reported on **Llama-3-8B**, while results for the other models are provided in Appendix C.4.

Membership Inference Attacks. We evaluate RAISE against ten representative MIAs spanning multiple sources of membership signal. The reference-based attacks are Ratio (Carlini et al., 2021), SPV-MIA (Fu et al., 2024), and the learned Ensemble attack (Zhang et al., 2025b). The remaining attacks include Loss (Yeom et al., 2018), Zlib and Lowercase (Carlini et al., 2021), Min-K% Prob (Shi et al., 2024), Min-K%++ (Zhang et al., 2025a), ReCaLL (Xie et al., 2024), and Con-ReCall (Wang et al., 2025).

For attacks requiring an external reference model, we use `llama-7b` for Llama-3 and Llama-3.2, `gpt2-xl` for GPT-J and Pythia, and `Qwen3-0.6B-Base` for Qwen3-8B. Unless otherwise specified, we follow the implementations and hyperparameter settings used by SOFT (Zhang et al.,

Table 1: Membership inference performance before and after applying RAISE across six domains. Values report bootstrap means over 1,000 resamples; corresponding standard deviations are provided in Appendix D.

MIAs	ArXiv		HackerNews		PubMed		Pile-CC		Wikipedia		GitHub	
	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE
Loss (Yeom et al., 2018)	0.893	0.589	0.897	0.617	0.920	0.578	0.845	0.516	0.796	0.502	0.820	0.658
Zlib (Carlini et al., 2021)	0.886	0.582	0.912	0.599	0.922	0.613	0.877	0.511	0.812	0.537	0.843	0.723
Lowercase (Carlini et al., 2021)	0.851	0.570	0.878	0.616	0.886	0.597	0.849	0.519	0.807	0.510	0.800	0.648
Min-K% Prob (Shi et al., 2024)	0.638	0.533	0.639	0.523	0.647	0.560	0.664	0.524	0.627	0.540	0.565	0.529
Min-K%++ (Zhang et al., 2025a)	0.847	0.547	0.771	0.526	0.896	0.615	0.830	0.508	0.816	0.554	0.864	0.693
Ratio (Carlini et al., 2021)	0.989	0.630	0.987	0.630	0.991	0.552	0.997	0.507	0.973	0.613	0.889	0.508
ReCaLL (Xie et al., 2024)	0.929	0.604	0.926	0.638	0.953	0.595	0.864	0.521	0.804	0.511	0.826	0.678
Con-ReCall (Wang et al., 2025)	0.829	0.597	0.735	0.540	0.846	0.584	0.746	0.543	0.734	0.521	0.812	0.649
Ensemble (Zhang et al., 2025b)	0.882	0.556	0.902	0.592	0.938	0.569	0.912	0.677	0.816	0.674	0.768	0.644
SPV-MIA (Fu et al., 2024)	0.997	0.503	1.000	0.578	1.000	0.560	0.999	0.584	0.989	0.571	0.988	0.523
Average	**0.874**	**0.571**	**0.865**	**0.586**	**0.900**	**0.582**	**0.858**	**0.541**	**0.817**	**0.553**	**0.818**	**0.625**

(a) AUC-ROC. Values closer to 0.5 indicate weaker membership distinguishability.

MIAs	ArXiv		HackerNews		PubMed		Pile-CC		Wikipedia		GitHub	
	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE
Loss (Yeom et al., 2018)	0.163	0.001	0.023	0.000	0.333	0.000	0.023	0.000	0.041	0.000	0.220	0.025
Zlib (Carlini et al., 2021)	0.181	0.012	0.031	0.000	0.283	0.004	0.021	0.000	0.036	0.000	0.286	0.074
Lowercase (Carlini et al., 2021)	0.249	0.011	0.160	0.022	0.187	0.015	0.048	0.003	0.086	0.005	0.129	0.006
Min-K% Prob (Shi et al., 2024)	0.209	0.037	0.155	0.034	0.308	0.051	0.034	0.000	0.066	0.000	0.070	0.016
Min-K%++ (Zhang et al., 2025a)	0.204	0.021	0.023	0.000	0.219	0.000	0.015	0.000	0.056	0.003	0.187	0.009
Ratio (Carlini et al., 2021)	0.420	0.025	0.247	0.035	0.609	0.051	0.886	0.078	0.271	0.043	0.069	0.005
ReCaLL (Xie et al., 2024)	0.223	0.000	0.027	0.000	0.436	0.001	0.029	0.000	0.072	0.000	0.224	0.037
Con-ReCall (Wang et al., 2025)	0.174	0.055	0.119	0.020	0.303	0.013	0.016	0.003	0.090	0.000	0.236	0.069
Ensemble (Zhang et al., 2025b)	0.062	0.000	0.126	0.000	0.413	0.000	0.263	0.043	0.028	0.000	0.060	0.000
SPV-MIA (Fu et al., 2024)	0.924	0.087	1.000	0.118	1.000	0.094	0.954	0.145	0.822	0.140	0.921	0.295
Average	**0.281**	**0.025**	**0.191**	**0.023**	**0.409**	**0.023**	**0.229**	**0.027**	**0.157**	**0.019**	**0.240**	**0.054**

(b) TPR at 1% FPR. Lower values indicate stronger protection.

2025b). For every attack, the undefended and edited checkpoints are evaluated on the same candidate records with identical attack configurations and auxiliary models, enabling direct paired comparisons.

Defense Configurations. We compare RAISE with three representative defenses that can be applied after the original fine-tuning stage: Gaussian Logit Noise (Rahimian et al., 2020), NPO (Zhang et al., 2024), and Forget-CE (Maini et al., 2024). Gaussian Logit Noise perturbs the logits exposed at inference time. NPO updates the checkpoint using a preference-based forgetting objective, whereas Forget-CE performs an additional LoRA-based update that combines forget-set cross-entropy optimization with reference-model KL regularization. Together, these baselines cover serving-time perturbation and direct post-training parameter modification.

Membership-inference metrics. Following prior work (Zhang et al., 2025b; Carlini et al., 2022), we report AUC-ROC and TPR at 1% FPR. AUC-ROC measures how well an attack ranks members above non-members; values closer to 0.5 indicate weaker distinguishability under the prespecified attack-score orientation. TPR at 1% FPR captures high-confidence membership inference in a low-false-positive regime, with lower values indicating stronger protection. We report bootstrap means over 1,000 resamples, with standard deviations in Appendix D.

Model utility. We evaluate utility using perplexity and domain-specific LLM-as-a-Judge evaluation. Perplexity is computed with the EleutherAI Language Model Evaluation Harness (Gao et al., 2023), with lower values indicating better predictive performance. LLM-as-a-Judge measures retention of domain-specific capabilities acquired during fine-tuning using the same evaluation questions and fixed judge, prompt, scoring, and aggregation configurations.

4.2 DEFENSE EFFECTIVENESS AGAINST MEMBERSHIP INFERENCE ATTACKS

We first evaluate whether RAISE reduces membership distinguishability across diverse attack families and data domains. We use `Llama-3-8B` as the primary target model. For each domain, the model is fine-tuned with LoRA on 1,000 member records for three epochs, after which 100 members are designated for protection. These protected records are used as the member population for evaluation and are paired with 100 non-members randomly selected from the same domain. We compare the fine-tuned checkpoint immediately before editing (*Undefended*) with the corresponding edited checkpoint (RAISE), keeping the evaluation records and attack configurations fixed. Table 1(a) reports AUC-ROC and Table 1(b) reports TPR at 1% FPR. RAISE consistently suppresses

Table 2: Comparison of **RAISE** with post-training baseline defenses on Pile-CC. MIA results are reported as bootstrap means over 1,000 resamples; corresponding standard deviations are provided in Appendix D. Perplexity measures model utility, with lower values indicating better utility. The best and second-best defense results are shown in bold and italics, respectively.

Methods	Loss	Zlib	Lowercase	Min-K% Prob	Min-K%++	Ratio	ReCaLL	Con-ReCall	Ensemble	SPV-MIA	Perplexity
Undefended	0.845	0.877	0.849	0.664	0.830	0.997	0.864	0.746	0.912	0.999	11.64
Noise (Rahimian et al., 2020)	0.841	0.872	0.841	0.664	0.824	0.996	0.859	0.738	0.915	0.999	12.51
NPO (Zhang et al., 2024)	0.819	0.855	0.808	0.646	0.800	0.993	0.838	0.720	0.919	0.996	12.58
Forget-CE (Maini et al., 2024)	*0.742*	*0.774*	*0.719*	*0.604*	*0.707*	*0.974*	*0.761*	*0.643*	*0.787*	*0.972*	11.76
RAISE	**0.516**	**0.511**	**0.519**	**0.524**	**0.508**	**0.507**	**0.521**	**0.543**	**0.677**	**0.584**	12.59

(a) AUC-ROC results.

Methods	Loss	Zlib	Lowercase	Min-K% Prob	Min-K%++	Ratio	ReCaLL	Con-ReCall	Ensemble	SPV-MIA	Perplexity
Undefended	0.023	0.021	0.048	0.034	*0.015*	0.886	0.029	0.016	0.263	0.954	11.64
Noise (Rahimian et al., 2020)	0.023	0.018	0.045	0.032	*0.015*	0.838	0.030	0.016	0.242	0.954	12.51
NPO (Zhang et al., 2024)	0.022	0.019	0.030	0.032	*0.015*	0.771	0.028	0.016	0.380	0.863	12.58
Forget-CE (Maini et al., 2024)	*0.012*	*0.017*	*0.021*	*0.024*	*0.015*	*0.393*	*0.007*	**0.000**	**0.000**	*0.607*	11.76
RAISE	**0.000**	**0.000**	**0.003**	**0.000**	**0.000**	**0.078**	**0.000**	*0.003*	*0.043*	**0.145**	12.59

(b) TPR at 1% FPR.

membership inference across the full attack–domain evaluation. Averaged over the ten attacks and six domains, AUC decreases from 0.855 to 0.576, while TPR at 1% FPR decreases from 25.1% to 2.8%. Both metrics decrease in every evaluated domain, and the same trend holds across all ten attacks, showing that the improvement is not driven by a particular attack family or dataset. The defended domain-level average AUCs range from 0.541 to 0.625, while average TPR at 1% FPR remains between 1.9% and 5.4%. The reduction is also substantial for attacks that are particularly strong on the undefended checkpoints. For example, SPV-MIA achieves AUCs between 0.988 and 1.000 before mitigation across the six domains, but only 0.503–0.584 after applying RAISE. These results demonstrate that localized reference-aligned editing substantially reduces both overall member–non-member separability and high-confidence membership inference across heterogeneous attack signals and domains.

4.3 MODEL UTILITY VIA LLM-AS-A-JUDGE

We next evaluate whether the privacy gains of RAISE preserve the domain-specific capabilities acquired during fine-tuning. Following the LLM-as-a-Judge evaluation methodology used in prior work (Zhang et al., 2025b; Zheng et al., 2023), we adopt the same general evaluation framework in our setting, with several configuration choices informed by prior work. For a controlled comparison, the Base, Fine-tuned, and RAISE checkpoints are evaluated on the same domain-specific questions using the same judge configuration, prompts, scoring criteria, and aggregation procedure. As shown in Figure 2, fine-tuning increases the average judge score from 3.47 for the Base model to 7.39. After applying RAISE, the model retains an average score of 6.33 and remains above the Base model in every evaluated domain. Overall, RAISE preserves a substantial portion of the domain-specific utility acquired through fine-tuning while substantially reducing membership distinguishability.

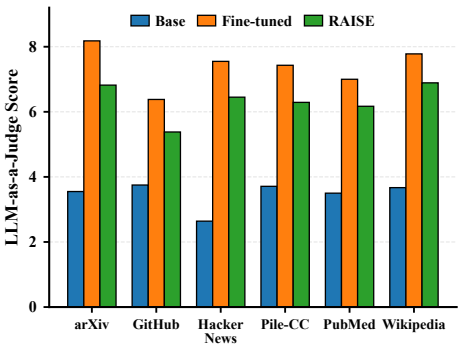


Figure 2: Utility evaluation using LLM-as-a-Judge.

4.4 COMPARISON WITH EXISTING DEFENSES

We compare RAISE with three representative post-training defenses on Pile-CC under the default Llama-3-8B setting: inference-time Gaussian Logit Noise, NPO, and Forget-CE. Table 2 reports performance across the ten evaluated MIAs together with model perplexity. Gaussian Logit Noise

and NPO provide only modest reductions in membership distinguishability, with average AUCs of 0.855 and 0.839, respectively, compared with 0.858 for the undefended model. Forget-CE provides a stronger baseline, reducing the average AUC to 0.768 and TPR at 1% FPR to 0.110, but remains vulnerable to Ratio (AUC 0.974). In comparison, RAISE reduces the average AUC to 0.541 and TPR to 0.027, with Ratio AUC further reduced to 0.507. RAISE therefore achieves substantially stronger membership mitigation across the evaluated attacks. Its perplexity (12.59) is comparable to NPO (12.58), although Forget-CE better preserves utility (11.76).

4.5 ADAPTIVE ATTACKS

We further evaluate whether an attacker with knowledge of the RAISE mechanism can recover residual membership signal after mitigation. We consider adaptive variants that exploit token-level loss patterns and reference-model discrepancies to reduce the influence of positions most affected by the defense. The attacker is assumed to know the general design of RAISE, but not run-specific defender information such as the selected spans or defender-side reference records. Full attack definitions and hyperparameter sweeps are provided in Appendix E. At the primary setting $q = 0.10$, the strongest evaluated adaptive variant reaches a defended bootstrap-mean AUC of 0.546 on Pile-CC with Llama-3-8B, compared with 0.516 for the standard Loss attack after RAISE and 0.845 before mitigation. Across the full hyperparameter sweep, the largest defended mean AUC is 0.577, observed for Reference-Delta Variant B at $q = 0.20$. Overall, the evaluated adaptive attacks recover only limited residual separation relative to the undefended checkpoint.

4.6 ABLATION STUDY

Effect of Span Localization. We compare RAISE with random and lowest-loss span selection on Pile-CC while keeping the downstream editing procedure fixed (Appendix C.1). As shown in Table 3, RAISE reduces the mean AUC across the ten attacks from 0.8580 to 0.5412, compared with 0.6051 for lowest-loss selection and 0.6752 for random selection. The Loss-attack AUC follows the same ordering: 0.5163 for RAISE, 0.6162 for lowest-loss selection, and 0.7143 for random selection. RAISE also achieves the lowest maximum AUC across the evaluated attacks, at 0.6776, compared with 0.7679 and 0.8697 for the lowest-loss and random variants, respectively. These results show that RAISE's reference-guided localization yields lower aggregate attack performance than the two alternative localization strategies at the evaluated operating point.

Table 3: Span localization on Pile-CC (AUC ↓). Mean and Max. denote the mean and maximum over ten attacks. Bold marks column minima.

Method	Loss	Mean	Max.
Unedited	0.8451	0.8580	0.9978
Random	0.7143	0.6752	0.8697
Lowest-loss	0.6162	0.6051	0.7679
RAISE	**0.5163**	**0.5412**	**0.6776**

Additional ablations. RAISE remains effective across four additional model configurations on Pile-CC, reducing ten-attack macro-average AUCs from 0.789–0.825 to 0.499–0.579. In a diagnostic layer sweep on GitHub, the automatically selected Layer 22 achieves the lowest macro-average AUC among the evaluated layers (0.5865, versus 0.8181 before editing). Finally, increasing protection coverage from 0 to 100% reduces macro-average AUC from 0.882 to 0.568. Full curves, utility measurements, and experimental details are provided in Appendix C.

5 CONCLUSION

We introduced RAISE, a post-training approach that formulates membership mitigation for selected fine-tuning records as a model-editing problem. Motivated by the non-uniform distribution of reference-relative prediction differences, RAISE uses selective reference alignment to reduce membership-distinguishing behavior through targeted checkpoint editing. Evaluations across domains, attacks, and model configurations show reduced membership distinguishability while retaining substantial fine-tuning utility, including under the evaluated adaptive strategies. These results support selective reference alignment as an empirical mitigation for supplied member records in existing checkpoints, without rerunning the original fine-tuning procedure.

6 AI USE STATEMENT

In this work, we used generative AI tools to improve the clarity, grammar, and readability of the manuscript. We also used an LLM as an automated judge to evaluate domain-specific model utility, following the protocol described in Section 4.3. The judge was used only to score model-generated answers for utility evaluation; membership-inference metrics were computed numerically from attack scores and did not rely on LLM judgments.

7 ETHICS STATEMENT

This work studies privacy risks arising from membership inference against fine-tuned language models and develops a post-training defense for selected known member records. Our experiments use benchmark texts from six MIMIR domains: ArXiv, HackerNews, PubMed, Pile-CC, Wikipedia, and GitHub. Membership labels refer specifically to participation in the target fine-tuning event and do not imply that non-member records were absent from model pretraining. Our evaluation includes adaptive attacks that could potentially be misused to infer sensitive associations between records and fine-tuning datasets. These analyses are intended to assess defense limitations and support authorized privacy auditing and mitigation. RAISE provides empirical protection for supplied member records under the evaluated settings, rather than a formal privacy guarantee or proof of complete data removal. Reduced attack performance should not be interpreted as evidence that protected records were never used for training or as guaranteed protection for all training records against arbitrary adversaries. Our goal is to improve the privacy of existing fine-tuned models while making the protection scope and privacy–utility trade-offs explicit.

8 REPRODUCIBILITY STATEMENT

We describe the RAISE framework and experimental setup in Sections 3 and 4, including the models, datasets, baselines, and evaluation protocols. Additional implementation details, hyperparameter settings, and evaluation procedures are provided in the appendices to support reproducibility.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Tomer Ashuach, Martin Tutek, and Yonatan Belinkov. REVS: Unlearning sensitive information in language models via rank editing in the vocabulary space. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 14774–14797, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.763. URL https://aclanthology.org/2025.findings-acl.763/.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usvsn Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2397–2430. PMLR, 2023. URL https://proceedings.mlr.press/v202/biderman23a.html.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pp. 2633–2650, 2021.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1897–1914, 2022. doi: 10.1109/SP46214.2022.9833649.

Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. Do membership inference attacks work on large language models? *arXiv preprint arXiv:2402.07841*, 2024.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. Alphaedit: Null-space constrained model editing for language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=HvSytvg3Jh.

Haowei Fu, Bo Ni, Han Xu, Kunpeng Liu, Dan Lin, and Tyler Derr. Ensemble privacy defense for knowledge-intensive LLMs against membership inference attacks. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 2786–2799, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-386-9. doi: 10.18653/v1/2026.findings-eacl.145. URL https://aclanthology.org/2026.findings-eacl.145/.

Wenjie Fu, Huandong Wang, Chen Gao, Guanghua Liu, Yong Li, and Tao Jiang. Membership inference attacks against fine-tuned large language models via self-prompt calibration. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=PAWQvrForJ.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023. URL https://zenodo.org/records/10256836.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer

Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mo-

han, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Anthony Hughes, Vasisht Duddu, N. Asokan, Nikolaos Aletras, and Ning Ma. PATCH: Mitigating PII leakage in language models with privacy-aware targeted circuit PatcHing. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 5139–5153, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-386-9. doi: 10.18653/v1/2026.findings-eacl.271. URL https://aclanthology.org/2026.findings-eacl.271/.

Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. Memguard: Defending against black-box membership inference attacks via adversarial examples. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, pp. 259–274, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367479. doi: 10.1145/3319535.3363201. URL https://doi.org/10.1145/3319535.3363201.

Jiacheng Li, Ninghui Li, and Bruno Ribeiro. MIST: Defending against membership inference attacks through Membership-Invariant subspace training. In *33rd USENIX Security Symposium (USENIX Security 24)*, pp. 2387–2404, Philadelphia, PA, August 2024. USENIX Association. ISBN 978-1-939133-44-1. URL https://www.usenix.org/conference/usenixsecurity24/presentation/li-jiacheng.

Qiufeng Li, Shu Hong, Tian Lan, and Weidong Cao. GUARD: Gold-unchanged anchored distillation for defending LLMs against membership inference attacks, 2026. URL https://openreview.net/forum?id=CjAE8ScfTK.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary Chase Lipton, and J Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=B41hNBoWLo.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. Membership inference attacks against language models via neighbourhood comparison. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 11330–11343, 2023.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=-h6WAS6eE4.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS.

Meta AI. Llama 3.2-3B model card. Hugging Face Model Card, September 2024. URL https://huggingface.co/meta-llama/Llama-3.2-3B.

Shadi Rahimian, Tribhuvanesh Orekondy, and Mario Fritz. Sampling attacks: Amplification of membership inference attacks by repeated queries. *arXiv preprint arXiv:2009.00395*, 2020.

Elena Sofia Ruzzetti, Giancarlo A. Xompero, Davide Venditti, and Fabio Massimo Zanzotto. Private memorization editing: Turning memorization into a defense to strengthen data privacy in large language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16572–16592, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.810. URL https://aclanthology.org/2025.acl-long.810/.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *International Conference on Learning Representations*, volume 2024, pp. 51826–51843, 2024.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks Against Machine Learning Models . In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, Los Alamitos, CA, USA, May 2017. IEEE Computer Society. doi: 10.1109/SP.2017.41. URL https://doi.ieeecomputersociety.org/10.1109/SP.2017.41.

Toan Tran, Ruixuan Liu, and Li Xiong. Tokens for learning, tokens for unlearning: Mitigating membership inference attacks in large language models via dual-purpose training. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 22872–22888, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1174. URL https://aclanthology.org/2025.findings-acl.1174/.

Ben Wang and Aran Komatsuzaki. GPT-J-6B: A 6 billion parameter autoregressive language model. EleutherAI, May 2021. URL https://github.com/kingoflolz/mesh-transformer-jax.

Cheng Wang, Yiwei Wang, Bryan Hooi, Yujun Cai, Nanyun Peng, and Kai-Wei Chang. Con-recall: Detecting pre-training data in llms via contrastive decoding. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 1013–1026, 2025.

Xinwei Wu, Junzhuo Li, Minghui Xu, Weilong Dong, Shuangzhi Wu, Chao Bian, and Deyi Xiong. DEPN: Detecting and editing privacy neurons in pretrained language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 2875–2886, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.174. URL https://aclanthology.org/2023.emnlp-main.174/.

Xinwei Wu, Weilong Dong, Shaoyang Xu, and Deyi Xiong. Mitigating privacy seesaw in large language models: Augmented privacy neuron editing via activation patching. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 5319–5332, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.315. URL https://aclanthology.org/2024.findings-acl.315/.

Roy Xie, Junlin Wang, Ruomin Huang, Minxing Zhang, Rong Ge, Jian Pei, Neil Zhenqiang Gong, and Bhuwan Dhingra. Recall: Membership inference via relative conditional log-likelihoods. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8671–8689, 2024.

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

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting . In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pp. 268–282, Los Alamitos, CA, USA, July 2018. IEEE Computer Society. doi: 10.1109/CSF.2018.00027. URL https://doi.ieeecomputersociety.org/10.1109/CSF.2018.00027.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Yang, and Hai Li. Min-k%++: Improved baseline for pre-training data detection from large language models. In *International Conference on Learning Representations*, volume 2025, pp. 64845–64862, 2025a.

Kaiyuan Zhang, Siyuan Cheng, Hanxi Guo, Yuetian Chen, Zian Su, Shengwei An, Yuntao Du, Charles Fleming, Ashish Kundu, Xiangyu Zhang, and Ninghui Li. Soft: Selective data obfuscation for protecting llm fine-tuning against membership inference attacks. In *34th USENIX Security Symposium (USENIX Security 25)*, Seattle, WA, 2025b. USENIX Association.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=MXLBXjQkmb.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=uccHPGDlao.

A Empirical Motivation for Selective Reference Alignment

This appendix provides the empirical analysis underlying the design principle introduced in Section 3.1. Our goal is to examine whether membership-related prediction differences are distributed uniformly across a record, or instead concentrate in localized regions that can serve as more effective intervention targets. Importantly, the analyses below operate directly in score space and are used only to motivate the targeting strategy of RAISE; they do not by themselves establish the effect of a realizable parameter update.

A.1 Reference-Relative Token Gaps

Record-level membership scores aggregate token-level prediction statistics and can therefore obscure where member–non-member differences occur within a sequence. Moreover, low token loss alone is not sufficient to identify an anomalous prediction pattern, since some tokens are intrinsically easy to predict. We therefore examine member-token losses relative to empirical non-member reference behavior.

For a tokenized member record $x = (x_1, \ldots, x_T)$, the observed-token loss is

$$\ell_t(x; M_0) = -\log p_{M_0}(x_t \mid x_{<t}). \tag{3}$$

For each member-token position, we construct an empirical distribution of corresponding reference-token losses from a same-domain non-member reference pool B. Token-conditioned statistics are used when sufficiently supported; otherwise, the implementation backs off to broader reference statistics. The reference construction, support thresholds, and backoff rules are described in Appendix B.2.

Let $r_t(x; B)$ denote the median of the resulting reference-loss distribution. We define the member–reference gap as

$$d_t(x) = r_t(x; B) - \ell_t(x; M_0). \tag{4}$$

A positive value of $d_t(x)$ indicates that the member token receives lower loss than its reference median. This distinguishes reference-relative anomalies from tokens that merely have low absolute loss because they are intrinsically predictable.

Our analysis uses 100 member records and a 100-record same-domain non-member reference pool, with sequences truncated to at most 512 tokens. All model measurements are obtained from the fine-tuned checkpoint M_0 before applying any editing. Unless otherwise stated, the results in this appendix are reported on ArXiv.

A.2 Localization under Disjoint References

We first examine whether large reference-relative gaps identify positions whose anomalous behavior persists beyond the particular reference records used for selection. To separate token selection from reference-based measurement, we deterministically partition the 100-record reference pool for each member into 20 selection records, B_{sel}, and 80 disjoint held-out records, B_{hold}.

We compute two independent reference-relative gaps:

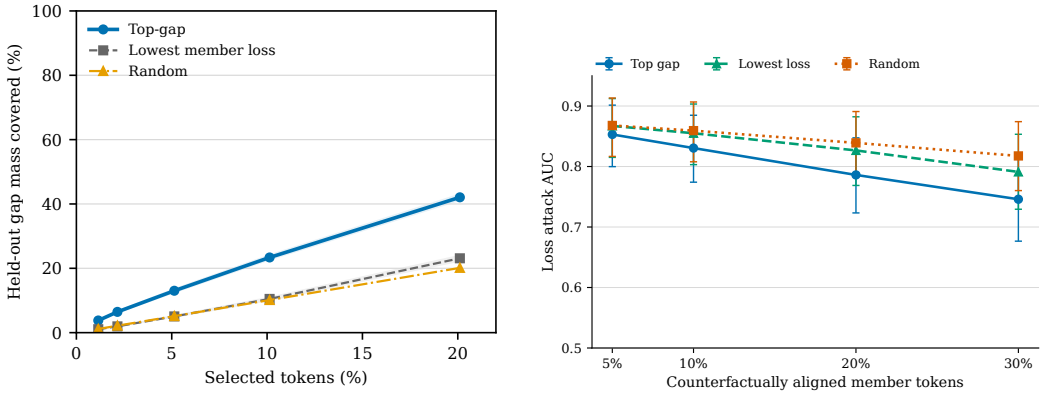
$$\begin{aligned} d_t^{\mathrm{sel}}(x) &= r_t(x; B_{\mathrm{sel}}) - \ell_t(x; M_0), \\ d_t^{\mathrm{hold}}(x) &= r_t(x; B_{\mathrm{hold}}) - \ell_t(x; M_0). \end{aligned} \tag{5}$$

Let $\mathcal{I}_k(x)$ contain the top $k\%$ of token positions ranked using only $d_t^{\mathrm{sel}}(x)$. For a record with nonzero held-out positive-gap mass, we measure the fraction of that mass captured by the selected positions:

$$C_k^{\mathrm{hold}}(x) = \frac{\sum_{t \in \mathcal{I}_k(x)} [d_t^{\mathrm{hold}}(x)]_+}{\sum_{t=1}^{T} [d_t^{\mathrm{hold}}(x)]_+}, \tag{6}$$

where $[z]_+ = \max(z, 0)$.

We compare reference-relative top-gap selection with two token-count-matched baselines: random selection and lowest-loss selection based on $\ell_t(x; M_0)$.



(a) Held-out positive-gap coverage.

(b) Counterfactual Loss-attack AUC.

Figure 3: Empirical motivation for selective reference alignment on ArXiv. Positions selected by large reference-relative gaps capture more held-out positive-gap mass and yield larger counterfactual reductions in record-level Loss-attack AUC than token-count-matched random or lowest-loss baselines. Token selection and evaluation use disjoint non-member reference subsets.

As shown in Figure 3(a), positions selected using large reference-relative gaps capture a disproportionate share of positive-gap mass under the disjoint held-out references. At a 10% token budget, top-gap selection captures 23.37% of the held-out positive-gap mass, compared with 10.13% for random selection and 10.48% for lowest-loss selection. At a 20% token budget, top-gap coverage increases to 42.07%, compared with 20.13% for random selection.

Because the positions are selected using only B_{sel} and evaluated using the disjoint B_{hold}, this concentration cannot be explained solely by the particular reference samples used to rank the tokens. Instead, the result indicates that reference-relative prediction differences are concentrated in a comparatively small subset of token positions.

A.3 Counterfactual Contribution to Record-Level Separability

The concentration analysis above shows where reference-relative differences occur, but does not establish whether these positions are useful targets for reducing record-level membership separability. We therefore construct a score-space counterfactual using the same disjoint-reference protocol.

For positions selected using B_{sel}, we remove the positive member–reference advantage by replacing the member loss with at least the held-out reference median:

$$\widetilde{\ell}_t(x) = \begin{cases} \max(\ell_t(x; M_0), r_t(x; B_{\text{hold}})), & t \in \mathcal{I}_k(x), \\ \ell_t(x; M_0), & \text{otherwise.} \end{cases} \tag{7}$$

Token positions are selected exclusively using B_{sel}, whereas their counterfactual loss values are determined using B_{hold}. We recompute the member's Loss-attack score as

$$\widetilde{s}_{\text{Loss}}(x) = -\frac{1}{T}\sum_{t=1}^{T} \widetilde{\ell}_t(x), \tag{8}$$

while leaving all non-member scores unchanged.

Figure 3(b) shows that top-gap alignment produces the largest reduction in record-level Loss-attack AUC across the evaluated token budgets. At a 10% token budget, the AUC decreases from 0.876 to 0.831 under top-gap alignment, compared with 0.855 for lowest-loss selection and 0.859 for random selection. Paired-bootstrap comparisons favor top-gap alignment over both baselines across all evaluated budgets, with the corresponding 95% confidence intervals excluding zero.

A.4 IMPLICATIONS FOR RAISE

Together, these results support two observations underlying the design of RAISE. First, membership-related prediction differences are highly non-uniform within a record: token positions with large reference-relative gaps capture substantially more held-out positive-gap mass than token-count-matched alternatives. Second, under a score-space counterfactual, aligning these positions toward held-out non-member reference behavior reduces record-level Loss separability more effectively than modifying randomly selected or simply lowest-loss positions.

These findings motivate *selective reference alignment*: under a limited intervention budget, the defense should prioritize local prediction behavior that is anomalously favorable to members relative to non-member references, and move that behavior toward a reference-supported range.

The counterfactual analysis is intentionally limited in scope. It modifies member losses directly while holding non-member scores fixed, and it matches the number of modified tokens rather than the total magnitude of the loss movement. It therefore does not establish that an equivalent reduction in membership separability can be achieved through a realizable parameter update. We use these results only as a targeting principle; the actual checkpoint-level mechanism of RAISE is described in Section 3.2, with additional implementation details provided in Appendix B.

B ADDITIONAL DETAILS OF RAISE

This appendix provides additional details of the RAISE pipeline described in Section 3.2. We first connect the token-level reference gaps studied in Appendix A to the span-level targets used by RAISE, and then describe reference construction, span selection, write-location selection, hidden-state target optimization, and projected parameter writeback. We conclude with the default hyper-parameters used in the main-model configuration.

B.1 FROM TOKEN GAPS TO SPAN-LEVEL TARGETS

The empirical analysis in Appendix A operates at the token level, whereas RAISE constructs editing requests over contiguous spans. We make this connection explicit below.

For a candidate span $s = [i, j]$, define the mean member loss and the averaged reference quantiles as

$$\begin{aligned} \bar{\ell}_s &= \frac{1}{|s|} \sum_{t \in s} \ell_t(x; M_0), \\ Q_s(q) &= \frac{1}{|s|} \sum_{t \in s} Q_t(q; x, B). \end{aligned} \tag{9}$$

Since $Q_t(0.5; x, B) = r_t(x; B)$, the numerator of the standardized span score used in the main method satisfies

$$Q_s(0.5) - \bar{\ell}_s = \frac{1}{|s|} \sum_{t \in s} d_t(x). \tag{10}$$

Thus, the span-level score aggregates the token-level member–reference gaps studied in Appendix A. Its denominator normalizes this quantity using a stabilized mean per-token reference interquartile spread.

Importantly, $Q_s(q)$ is an average of token-level reference quantiles rather than a quantile of the span-mean loss distribution. In particular,

$$Q_s(0.75) - Q_s(0.25) = \frac{1}{|s|} \sum_{t \in s} \left[Q_t(0.75; x, B) - Q_t(0.25; x, B) \right]. \tag{11}$$

The resulting quantity is therefore the mean token-level interquartile spread before applying the ϵ stabilization floor.

B.2 Reference Construction and Span Selection

Reference construction. For each protected member record, the main-model configuration samples at most 20 non-member references from a 100-record same-domain defense-development pool. Reference sampling uses seed $42 + record_id$, and sequences are truncated to at most 512 tokens.

The target model scores the sampled reference records to obtain observed-token losses. A token-conditioned reference estimate is used when at least three matching observations are available. When token-conditioned support is insufficient, the constructor uses same-position statistics when at least five observations are available. If neither condition is satisfied, it falls back to the global reference-loss distribution.

This backoff strategy allows the method to use more specific reference behavior when sufficiently supported while retaining a well-defined target for sparsely observed tokens.

Candidate spans. Candidate spans contain between 5 and 14 tokens. A one-token separation buffer is enforced between retained spans, and at most 20 spans are retained for each member record.

The selector prefers candidates whose lexical or numeric token fraction is at least 0.5. This is a preference rather than a hard constraint: if no candidate satisfies the criterion, the constructor falls back to the highest-ranked contiguous candidate.

Candidate spans are ranked using their reference-relative discrepancy and standardized gap. Overlap is discouraged using a penalty of 0.25. After the first selected span, an additional candidate must have a summed positive gap of at least 1.25 and a positive-gap ratio of at least 0.10. Overlapping candidates are resolved greedily.

B.3 Reference-Aligned Target Construction

For each retained span s, RAISE defines the reference-supported interval

$$[\tau_s^-, \tau_s^+] = [Q_s(q_-), Q_s(q_+)]. \tag{12}$$

The method additionally defines a nominal reference target

$$\tau_s^0 = Q_s(q_0), \qquad q_- \le q_0 \le q_+. \tag{13}$$

The interval specifies a bounded region of reference-supported behavior rather than requiring the edited span to match a single empirical target exactly. Only positive loss movements toward this range are requested.

Requests are subject to fixed per-span and per-record movement budgets. A request that exceeds the remaining budget is clipped or omitted. These constraints prevent a small number of spans from inducing arbitrarily large loss changes and provide explicit control over the total requested movement for each member.

For the main-model configuration, we use

$$q_- = q_0 = 0.60, \qquad q_+ = 0.85, \tag{14}$$

with upper-bound violation weight $\omega = 0.60$. The per-span loss-movement cap is 1.0, the record-budget coefficient is 2.25, and the configured strength range is $[1.05, 2.10]$.

The loss bounds supplied to the hidden-state optimizer are denoted $[\widehat{\tau}_s^-, \widehat{\tau}_s^+]$ when necessary to distinguish the final bounded optimization targets from the raw reference quantiles $[\tau_s^-, \tau_s^+]$.

B.4 Write-Location Selection

The reference-aligned span targets specify the desired change in output-loss space, but not where in the network that change should be written. RAISE therefore selects a write layer using two complementary diagnostics: (i) whether member–reference separation is visible at a layer, and (ii) how sensitive reference behavior is to an intervention at that layer.

Layerwise member–reference separation. For transformer block l, we project its output through the model's final normalization and language-model head and compute the resulting sequence-level observed-token loss $\bar{\ell}_l(x)$. This intermediate projection is used only as a relative layer-selection diagnostic and is not treated as a calibrated language model.

For independent uniform draws X_S from the protected-member set S and X_B from the reference pool B, we define

$$A_{\text{sep}}(l) = \Pr[\bar{\ell}_l(X_B) > \bar{\ell}_l(X_S)] + \frac{1}{2}\Pr[\bar{\ell}_l(X_B) = \bar{\ell}_l(X_S)]. \tag{15}$$

This quantity is the AUC of negative intermediate-layer loss with members treated as the positive class. Larger values indicate that members systematically receive lower projected loss than reference non-members. Values below 0.5 correspond to the opposite ranking direction. The diagnostic is computed using S and B and is not restricted to the subset of records for which a positive editing request is ultimately retained.

Reference sensitivity. Comparable member–reference separation does not imply comparable sensitivity to intervention. Let $\mathcal{L}_B$ denote the mean observed-token loss on the reference pool, and let $a_{l,t}$ denote the output activation of the candidate module at reference-token position t. We compute

$$g_{l,t} = \frac{\partial \mathcal{L}_B}{\partial a_{l,t}}, \tag{16}$$

and define

$$U(l) = \frac{1}{|\mathcal{T}_B|} \sum_{t \in \mathcal{T}_B} \|g_{l,t}\|_2 \|a_{l,t}\|_2, \tag{17}$$

where $\mathcal{T}_B$ contains the scored reference-token positions.

The gradient–activation norm product provides a first-order proxy for the sensitivity of reference loss to a relative activation change. Smaller $U(l)$ therefore favors layers at which a local perturbation is expected to have less immediate influence on reference behavior.

Selection rule. RAISE first identifies a candidate region in which the relevant member–reference separation is reliably exposed according to $A_{\text{sep}}(l)$. Within this region, it selects the earliest layer whose $U(l)$ is within a fixed tolerance of the minimum candidate sensitivity.

This rule favors a representation at which the membership-related separation is already visible without unnecessarily selecting a later or more sensitive layer. Both $A_{\text{sep}}(l)$ and $U(l)$ are selection proxies rather than formal guarantees of post-edit locality; the resulting utility is therefore evaluated directly after editing.

Candidate-region construction can depend on the observed layerwise loss profile. For example, in the Llama-3-8B GitHub configuration used in our diagnostics, the selector identifies an inverted profile with onset at layer 28, a pre-crossing candidate band $[20, 24]$, and selects layer 22 using a tolerance parameter of 0.35. This configuration is an instance of the general selection rule rather than a universal positive-separation threshold.

B.5 HIDDEN-STATE TARGET OPTIMIZATION

For each retained span, RAISE optimizes a hidden-state target at the selected layer while keeping the checkpoint parameters fixed. The anchor position is chosen immediately before the span whenever available.

For zero-based stored span indices, we use

$$\text{anchor_pos} = \max(0, \text{start} - 1). \tag{18}$$

A span beginning at position zero therefore uses the stored position-zero activation. This boundary case does not assign a beginning-of-sequence-token interpretation to that representation.

Starting from the anchor representation h_s, the method optimizes a perturbation δ_s and forms

$$z_s = h_s + \delta_s. \tag{19}$$

The optimization objective described in Section 3.2 encourages the resulting span loss to lie inside the bounded reference-aligned interval while penalizing excessive deviation in output distributions and hidden-state magnitude.

The main-model target search uses 25 optimization steps with learning rate 0.1. The KL regularization factor is 0.0625, and the hidden-state norm-clamp factor is 1.15. The KL term retains the direction

$$D_{\mathrm{KL}}(p_{\delta_s} \| p_0), \tag{20}$$

where p_0 and p_{δ_s} denote the unperturbed and intervened next-token distributions on the fixed generic prompts used by the target search.

B.6 Projected Parameter Writeback

After obtaining the hidden-state targets, RAISE realizes them in the checkpoint using the null-space-constrained projected update principle of AlphaEdit (Fang et al., 2025). We summarize the corresponding parameterization below.

For a linear map with input dimension d_{in} and output dimension d_{out}, we use columnwise keys:

$$\begin{aligned} W, A, \Delta W &\in \mathbb{R}^{d_{\mathrm{out}} \times d_{\mathrm{in}}}, \\ K &\in \mathbb{R}^{d_{\mathrm{in}} \times m}, \\ V, R_W &\in \mathbb{R}^{d_{\mathrm{out}} \times m}, \\ P &\in \mathbb{R}^{d_{\mathrm{in}} \times d_{\mathrm{in}}}, \end{aligned} \tag{21}$$

where m is the number of request keys supplied to a writeback operation.

The activation targets z_s specify the desired internal changes generated by the target-search stage, whereas V denotes desired outputs in the local coordinates of the edited linear map. These quantities therefore play different roles.

Let K_{pres} denote auxiliary preservation keys. AlphaEdit forms a projector

$$P = U_{\perp} U_{\perp}^{\top} \tag{22}$$

from an approximate null space of the second-moment matrix of the preservation representations. For an update $\Delta W = AP$, its change on fixed preservation keys is

$$(W + \Delta W)K_{\mathrm{pres}} - WK_{\mathrm{pres}} = APK_{\mathrm{pres}}. \tag{23}$$

If $PK_{\mathrm{pres}} = 0$ exactly, the corresponding local linear-map outputs are unchanged. With a thresholded approximate null space, preservation is approximate and depends on both the key statistics and the magnitude of the update.

For the request keys, the residual identity is

$$(W + \Delta W)K - V = \Delta WK - (V - WK). \tag{24}$$

Defining

$$R_W = V - WK, \tag{25}$$

the projected solve therefore seeks an update whose action ΔWK fits the desired request residual while remaining constrained to the projected subspace.

These identities describe the local parameterization and preservation mechanism of the writeback step. They do not imply exact realization of the hidden-state targets or exact preservation of the complete nonlinear model.

For Llama-family configurations, the edited linear map is the MLP down-projection. The corresponding modules are `fc_out` for GPT-J and `dense_4h_to_h` for Pythia. The layer at which the target-search loss is evaluated and the module whose weights are ultimately updated serve different roles and need not be identified with one another.

B.7 Default Hyperparameters

Table 4 summarizes the main configuration used for RAISE. These values are held fixed in the primary experiments unless an experiment explicitly varies the corresponding component.

Table 4: Default RAISE configuration used in the primary experiments.

Parameter	Default value
Reference samples per member	≤ 20
Reference-sampling seed	$42 + record_id$
Maximum sequence length	512
Token-conditioned support threshold	3
Same-position support threshold	5
Candidate span length	5–14
Maximum spans per member	20
Span-separation buffer	1 token
Preferred lexical/numeric fraction	0.5
Overlap penalty	0.25
Minimum summed positive gap	1.25
Minimum positive-gap ratio	0.10
q_-	0.60
q_0	0.60
q_+	0.85
Upper-bound violation weight ω	0.60
Per-span loss-movement cap	1.0
Record-budget coefficient	2.25
Configured strength range	$[1.05, 2.10]$
Target-search steps	25
Target-search learning rate	0.1
KL coefficient	0.0625
Norm-clamp factor	1.15

C ADDITIONAL ABLATION RESULTS

This appendix provides the complete experimental settings and additional results for the ablation studies summarized in the main text. We examine four complementary aspects of RAISE: (1) whether reference-guided span localization identifies effective intervention targets, (2) how membership distinguishability changes as the fraction of protected members increases, (3) how the write location affects the realized privacy–utility trade-off, and (4) whether the observed mitigation persists across different target-model configurations. Unless otherwise stated, we summarize membership distinguishability using the macro-average AUC across the ten evaluated MIAs, denoted by $\bar{A}$. AUC values closer to 0.5 indicate weaker membership distinguishability under the prespecified attack-score orientations.

C.1 EFFECT OF SPAN LOCALIZATION

Experimental setup. We first examine whether the reference-relative localization criterion identifies more effective intervention targets than simpler alternatives. We compare RAISE with two span-localization variants on Pile-CC. RAISE prioritizes spans with large reference-relative loss gaps, whereas the lowest-loss variant selects spans with the smallest mean token loss and the random variant selects spans randomly using seed 101. All variants use the same downstream editing procedure after span selection.

We evaluate the resulting checkpoints against ten MIAs and report the Loss-attack AUC, the mean AUC, and the maximum AUC across these attacks. The unedited checkpoint M_0 serves as the baseline. We additionally report NLL on Pile-CC and Wikipedia to characterize the effect of different localization strategies on predictive utility.

Results and interpretation. Table 5 shows that reference-guided localization yields lower attack performance than the two alternative localization strategies across all three reported summaries. The

Table 5: Span-localization comparison on Pile-CC. Mean and Max. denote the mean and maximum AUC across the ten evaluated attacks, respectively. Bold marks the minimum attack AUC in each summary column.

	Attack AUC			NLL	
Method	Loss	Mean	Max.	Pile-CC	Wikipedia
Unedited M_0	0.8451	0.8580	0.9978	2.4544	1.7306
Random	0.7143	0.6752	0.8697	2.4612	1.7790
Lowest-loss	0.6162	0.6051	0.7679	2.4789	1.7951
RAISE	**0.5163**	**0.5412**	**0.6776**	2.5329	1.7956

mean AUC decreases from 0.8580 for the unedited checkpoint to 0.5412 under RAISE, compared with 0.6051 for lowest-loss selection and 0.6752 for random selection. The Loss-attack AUC follows the same trend: 0.5163 for RAISE, 0.6162 for lowest-loss selection, and 0.7143 for random selection. Likewise, the maximum AUC across the ten attacks is lowest under RAISE at 0.6776, compared with 0.7679 and 0.8697 for the lowest-loss and random variants, respectively.

These results indicate that the benefit of RAISE is not explained simply by modifying positions with low absolute loss. Instead, reference-relative discrepancies identify more effective intervention targets under the same localization setting, consistent with the selective reference-alignment principle motivating the method.

C.2 Effect of Protection Coverage

Experimental setup. RAISE is designed to protect a supplied set of known members rather than uniformly modify every record in the original fine-tuning dataset. In practice, a model owner may identify vulnerable or sensitive member records incrementally. We therefore examine how aggregate membership distinguishability changes as the fraction of protected members increases. We construct a fixed evaluation population containing 200 members and a separate pool of 200 evaluation non-members from Pile-CC. The evaluation non-member pool excludes the 100 calibration non-members. We use nested prefixes of the member pool with $N \in \{10, 20, 50, 100, 150, 200\}$ protected members, corresponding to 5%–100% nominal protection coverage. Each condition is evaluated on the same 400 records, so the evaluation composition remains fixed as N changes.

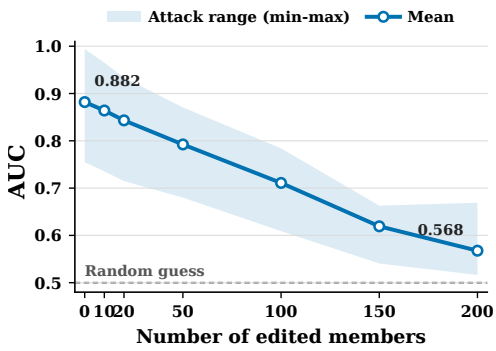


Figure 4: Effect of protection coverage on Pile-CC. The evaluation population is fixed at 200 members and 200 non-members while the number of protected members increases from 0 to 200.

Each condition starts from the same original fine-tuned checkpoint and uses the same target policy and edit layer. The $N = 0$ point is the original checkpoint's shared baseline evaluation. The checkpoints for successive values of N are constructed separately from the same original model rather than by sequentially accumulating edits. A companion matched-size evaluation uses N members and N non-members from the same pools. The fixed-population results reported here directly measure how aggregate membership distinguishability changes as protection coverage expands.

Results and interpretation. Figure 4 and Table 6 show a monotonic decrease in macro-average AUC as nominal protection coverage increases. The AUC decreases from 0.8819 without editing to 0.7923 at 25% coverage, 0.7109 at 50%, 0.6192 at 75%, and 0.5677 when all 200 members are designated for protection.

Table 6: Protection coverage on the fixed population (200 members and 200 non-members).

N	Coverage	Requests	$\bar{A}$
0	0%	0	0.8819
10	5%	169	0.8641
20	10%	335	0.8432
50	25%	828	0.7923
100	50%	1,668	0.7109
150	75%	2,430	0.6192
200	100%	3,216	0.5677

Table 7: The original checkpoint has $\bar{A} = 0.8181$. PPL changes are edited minus original; smaller increases indicate better preservation on the measured population. * marks the layer chosen by the proposed selector.

Layer	$\bar{A}$	$\Delta\text{PPL}_{\text{GitHub}}$	$\Delta\text{PPL}_{\text{Wiki}}$
5	0.7563	1.8839	0.4075
10	0.7130	1.8442	0.3903
15	0.6210	1.8046	0.3331
20	0.6105	0.4600	0.2256
22*	0.5865	0.7650	0.2759
25	0.6726	0.3317	0.1200
30	0.6725	0.5482	0.0811

Under partial coverage, the remaining unedited members continue to exhibit membership-distinguishing behavior and therefore sustain aggregate attack performance. As more members are incorporated into the protected set, this residual separation is progressively reduced. These results show that the privacy benefit of RAISE accumulates with protection coverage, supporting incremental deployment as additional sensitive member records are identified.

C.3 EFFECT OF LAYER SELECTION

Experimental setup. The reference-aligned targets specify what prediction behavior should change, but the realized intervention also depends on where that change is written into the network. An effective editing location should permit a meaningful reduction in membership-related behavior while limiting interference with unrelated model behavior. We therefore examine whether the proposed layer-selection strategy identifies an effective intervention location. We apply the same span requests to Layers 5, 10, 15, 20, 22, 25, and 30 of the GitHub fine-tuned Llama-3-8B model. The seven conditions reuse a byte-identical span manifest containing 1,411 requests for 86 of the 100 designated members. Their materialized editing hyperparameters are identical except for the layer index, and each condition is evaluated on the same 100 members and 100 non-members. Layer 22 is selected automatically by the proposed layer-selection procedure; downstream MIA performance is not used to choose the layer.

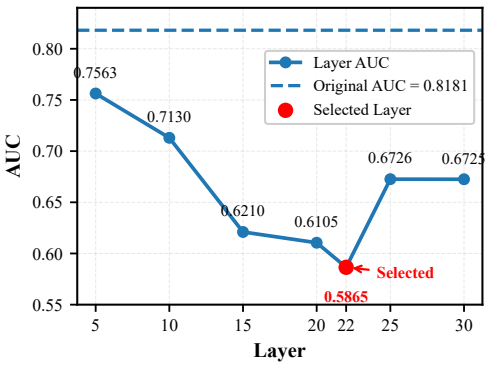


Figure 5: Effect of the write layer on macro-average MIA AUC for the GitHub fine-tuned Llama-3-8B model. Layer 22 is selected automatically by the proposed selector.

Results and interpretation. Figure 5 and Table 7 show that the selected Layer 22 achieves the lowest macro-average attack AUC among all seven evaluated locations, reducing AUC from 0.8181 for the original fine-tuned model to 0.5865. Applying the same requests at other layers yields higher AUC: Layers 15 and 20 produce values of 0.6210 and 0.6105, respectively, whereas Layers 25 and 30 produce 0.6726 and 0.6725.

Table 8: Cross-model results on Pile-CC, using 100 members and the 100-non-member population. Each model has 100 designated members. Reduction is $\bar{A}_{\text{before}} - \bar{A}_{\text{after}}$. All averages include the same ten attack rows within each model.

Model	$\bar{A}_{\text{before}}$	$\bar{A}_{\text{after}}$	Reduction
Llama-3.2-3B	0.7984	0.4989	0.2995
GPT-J-6B	0.8247	0.5790	0.2457
Pythia-6.9B	0.8004	0.5595	0.2409
Qwen3-8B	0.7891	0.5411	0.2480
Four-model mean	0.8031	0.5446	0.2585

Table 9: Per-attack point-estimate AUCs underlying the cross-model averages. Each pair reports the original fine-tuned checkpoint (Before) and RAISE (After).

	Llama-3.2-3B		GPT-J-6B		Pythia-6.9B		Qwen3-8B	
Attack	Before	After	Before	After	Before	After	Before	After
Loss	0.7475	0.4933	0.7962	0.6166	0.7745	0.5795	0.7448	0.5351
Zlib	0.7870	0.5012	0.8428	0.6420	0.8220	0.6148	0.7770	0.5398
Lowercase	0.7749	0.4764	0.8314	0.5702	0.8040	0.5406	0.7543	0.4346
Min-K% Prob	0.7752	0.5170	0.8270	0.5778	0.7816	0.5729	0.7532	0.5383
Min-K%++	0.7072	0.4855	0.7402	0.5062	0.7066	0.4939	0.7217	0.5287
ReCaLL	0.7595	0.4782	0.8152	0.6264	0.7919	0.5731	0.7581	0.5219
Con-ReCall	0.6265	0.4954	0.6528	0.4984	0.6382	0.5342	0.6208	0.4702
Ratio	0.9780	0.4388	0.9262	0.6354	0.9246	0.5840	0.9572	0.5636
Ensemble	0.8310	0.6971	0.8173	0.5857	0.7671	0.6217	0.8068	0.7406
SPV-MIA	0.9972	0.4063	0.9974	0.5312	0.9937	0.4805	0.9967	0.5378

These results indicate that the effectiveness of the realized intervention depends materially on the write location. In this diagnostic sweep, the automatically selected Layer 22 coincides with the strongest measured membership mitigation among the evaluated layers, rather than relying on a manually chosen early or late intervention point.

The write location also affects utility preservation. Layer 22 increases PPL by 0.7650 on GitHub and 0.2759 on Wikipedia, which is substantially smaller than the increases observed at Layers 5, 10, and 15. Later layers yield smaller PPL changes in some cases, but also provide weaker attack mitigation. Thus, the selected layer provides a favorable operating point in this diagnostic sweep: it achieves the strongest measured mitigation while avoiding the substantially larger perplexity increases produced by the earliest evaluated locations.

C.4 Effect of Target Model

Experimental setup. We next examine whether the observed mitigation is specific to the primary Llama-3-8B configuration or persists across different model architectures and parameter scales. We evaluate Llama-3.2-3B, GPT-J-6B, Pythia-6.9B, and Qwen3-8B on Pile-CC. The recorded fine-tuning protocol uses 1,000 member records, three epochs, maximum sequence length 512, per-device batch size 1, gradient accumulation 8, learning rate 2×10^{-4}, and weight decay 0.01. The LoRA configuration uses rank 8, scaling factor 16, and dropout 0.05. The four target-model configurations share the same 100 designated members, 100 calibration non-members, and 100-record non-member evaluation population.

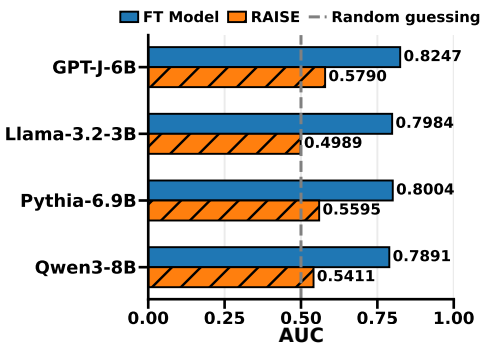


Figure 6: Cross-model evaluation of RAISE on Pile-CC. Bars report ten-attack macro-average AUC before and after RAISE for four additional target-model configurations.

Results and interpretation. Figure 6 and Table 8 show consistent reductions in the ten-attack macro-average AUC across all four additional target-

Table 10: Bootstrap statistics corresponding to Table 1 in the main text. Values report mean ± standard deviation over 1,000 bootstrap resamples for membership inference performance before and after applying RAISE across six domains.

MIAs	ArXiv		HackerNews		PubMed		Pile-CC		Wikipedia		GitHub	
	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE
Loss (Yeom et al., 2018)	0.893 ± 0.026	0.589 ± 0.036	0.897 ± 0.015	0.617 ± 0.031	0.920 ± 0.008	0.578 ± 0.043	0.845 ± 0.025	0.516 ± 0.033	0.796 ± 0.032	0.502 ± 0.045	0.820 ± 0.024	0.658 ± 0.031
Zlib (Carlini et al., 2021)	0.886 ± 0.028	0.582 ± 0.025	0.912 ± 0.017	0.599 ± 0.031	0.922 ± 0.012	0.613 ± 0.044	0.877 ± 0.017	0.511 ± 0.029	0.812 ± 0.031	0.537 ± 0.045	0.843 ± 0.023	0.723 ± 0.034
Lowercase (Carlini et al., 2021)	0.851 ± 0.025	0.570 ± 0.033	0.878 ± 0.025	0.616 ± 0.057	0.886 ± 0.012	0.597 ± 0.046	0.849 ± 0.024	0.519 ± 0.028	0.807 ± 0.025	0.510 ± 0.041	0.800 ± 0.038	0.648 ± 0.046
Min-K% Prob (Shi et al., 2024)	0.638 ± 0.042	0.533 ± 0.041	0.639 ± 0.030	0.523 ± 0.034	0.647 ± 0.037	0.560 ± 0.029	0.664 ± 0.036	0.524 ± 0.040	0.627 ± 0.035	0.540 ± 0.035	0.565 ± 0.021	0.529 ± 0.014
Min-K%++ (Zhang et al., 2025a)	0.847 ± 0.032	0.547 ± 0.048	0.771 ± 0.017	0.526 ± 0.037	0.896 ± 0.007	0.615 ± 0.031	0.830 ± 0.032	0.508 ± 0.033	0.816 ± 0.033	0.554 ± 0.040	0.864 ± 0.012	0.693 ± 0.031
Ratio (Carlini et al., 2021)	0.989 ± 0.007	0.630 ± 0.049	0.987 ± 0.008	0.630 ± 0.038	0.991 ± 0.006	0.552 ± 0.041	0.997 ± 0.002	0.507 ± 0.051	0.973 ± 0.011	0.613 ± 0.046	0.889 ± 0.018	0.508 ± 0.017
ReCaLL (Xie et al., 2024)	0.929 ± 0.022	0.604 ± 0.035	0.926 ± 0.013	0.638 ± 0.029	0.953 ± 0.007	0.595 ± 0.047	0.864 ± 0.025	0.521 ± 0.034	0.804 ± 0.033	0.511 ± 0.046	0.826 ± 0.026	0.678 ± 0.037
Con-ReCall (Wang et al., 2025)	0.829 ± 0.037	0.597 ± 0.026	0.735 ± 0.037	0.540 ± 0.036	0.846 ± 0.030	0.584 ± 0.031	0.746 ± 0.046	0.543 ± 0.041	0.734 ± 0.034	0.521 ± 0.044	0.812 ± 0.025	0.649 ± 0.046
Ensemble (Zhang et al., 2025b)	0.882 ± 0.016	0.556 ± 0.040	0.902 ± 0.019	0.592 ± 0.051	0.938 ± 0.016	0.569 ± 0.034	0.912 ± 0.016	0.677 ± 0.045	0.816 ± 0.028	0.674 ± 0.034	0.768 ± 0.040	0.644 ± 0.019
SPV-MIA (Fu et al., 2024)	0.997 ± 0.002	0.503 ± 0.041	1.000 ± 0.000	0.578 ± 0.041	1.000 ± 0.000	0.560 ± 0.046	0.999 ± 0.001	0.584 ± 0.043	0.989 ± 0.006	0.571 ± 0.043	0.988 ± 0.006	0.523 ± 0.044
Average	**0.874**	**0.571**	**0.865**	**0.586**	**0.900**	**0.582**	**0.858**	**0.541**	**0.817**	**0.553**	**0.818**	**0.625**

(a) AUC-ROC. Values closer to 0.5 indicate weaker membership distinguishability.

MIAs	ArXiv		HackerNews		PubMed		Pile-CC		Wikipedia		GitHub	
	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE	Undefended	RAISE
Loss (Yeom et al., 2018)	0.163 ± 0.153	0.001 ± 0.003	0.023 ± 0.028	0.000 ± 0.000	0.333 ± 0.122	0.000 ± 0.000	0.023 ± 0.025	0.000 ± 0.000	0.041 ± 0.043	0.000 ± 0.000	0.220 ± 0.156	0.025 ± 0.028
Zlib (Carlini et al., 2021)	0.181 ± 0.142	0.012 ± 0.014	0.031 ± 0.033	0.000 ± 0.000	0.283 ± 0.204	0.004 ± 0.006	0.021 ± 0.034	0.000 ± 0.000	0.036 ± 0.034	0.000 ± 0.000	0.286 ± 0.201	0.074 ± 0.055
Lowercase (Carlini et al., 2021)	0.249 ± 0.129	0.011 ± 0.021	0.160 ± 0.142	0.022 ± 0.024	0.187 ± 0.088	0.015 ± 0.013	0.048 ± 0.102	0.003 ± 0.009	0.086 ± 0.067	0.005 ± 0.014	0.129 ± 0.173	0.006 ± 0.017
Min-K% Prob (Shi et al., 2024)	0.209 ± 0.149	0.037 ± 0.036	0.155 ± 0.141	0.034 ± 0.033	0.308 ± 0.161	0.051 ± 0.070	0.034 ± 0.054	0.000 ± 0.000	0.066 ± 0.059	0.000 ± 0.000	0.070 ± 0.064	0.016 ± 0.017
Min-K%++ (Zhang et al., 2025a)	0.204 ± 0.164	0.021 ± 0.027	0.023 ± 0.026	0.000 ± 0.000	0.219 ± 0.122	0.000 ± 0.000	0.015 ± 0.019	0.000 ± 0.000	0.056 ± 0.050	0.003 ± 0.008	0.187 ± 0.154	0.009 ± 0.016
Ratio (Carlini et al., 2021)	0.420 ± 0.373	0.025 ± 0.034	0.247 ± 0.281	0.035 ± 0.056	0.609 ± 0.268	0.051 ± 0.055	0.886 ± 0.034	0.078 ± 0.023	0.271 ± 0.181	0.043 ± 0.033	0.069 ± 0.062	0.005 ± 0.007
ReCaLL (Xie et al., 2024)	0.223 ± 0.183	0.000 ± 0.000	0.027 ± 0.034	0.000 ± 0.000	0.436 ± 0.247	0.001 ± 0.003	0.029 ± 0.040	0.000 ± 0.000	0.072 ± 0.061	0.000 ± 0.000	0.224 ± 0.167	0.037 ± 0.036
Con-ReCall (Wang et al., 2025)	0.174 ± 0.117	0.055 ± 0.048	0.119 ± 0.066	0.020 ± 0.010	0.303 ± 0.167	0.013 ± 0.022	0.016 ± 0.020	0.003 ± 0.009	0.090 ± 0.055	0.000 ± 0.000	0.236 ± 0.162	0.069 ± 0.041
Ensemble (Zhang et al., 2025b)	0.062 ± 0.185	0.000 ± 0.000	0.126 ± 0.253	0.000 ± 0.000	0.413 ± 0.338	0.000 ± 0.000	0.263 ± 0.323	0.043 ± 0.068	0.028 ± 0.084	0.000 ± 0.000	0.060 ± 0.120	0.000 ± 0.000
SPV-MIA (Fu et al., 2024)	0.924 ± 0.041	0.087 ± 0.031	1.000 ± 0.000	0.118 ± 0.050	1.000 ± 0.000	0.094 ± 0.043	0.954 ± 0.028	0.145 ± 0.038	0.822 ± 0.068	0.140 ± 0.091	0.921 ± 0.031	0.295 ± 0.058
Average	**0.281**	**0.025**	**0.191**	**0.023**	**0.409**	**0.023**	**0.229**	**0.027**	**0.157**	**0.019**	**0.240**	**0.054**

(b) TPR at 1% FPR. Lower values indicate stronger protection.

model configurations. Before editing, the model-level averages range from 0.7891 to 0.8247; after applying RAISE, they range from 0.4989 to 0.5790. The reductions range from 0.2409 to 0.2995 AUC points. Averaged equally across the four models, the macro-average AUC decreases from 0.8031 to 0.5446, corresponding to an average reduction of 0.2585.

The largest numerical reduction is observed for Llama-3.2-3B, for which the macro-average AUC decreases from 0.7984 to 0.4989. GPT-J-6B, Pythia-6.9B, and Qwen3-8B similarly decrease from 0.8247, 0.8004, and 0.7891 to 0.5790, 0.5595, and 0.5411, respectively.

Table 9 provides the corresponding attack-level results. Lower reported AUCs are observed after editing for all ten evaluated attacks on each of the four target-model configurations. For example, Loss AUC decreases from 0.7475 to 0.4933 on Llama-3.2-3B and from 0.7448 to 0.5351 on Qwen3-8B. SPV-MIA similarly decreases from 0.9937–0.9974 before editing to 0.4063–0.5378 after editing.

Together, these results indicate that the observed mitigation is not specific to the primary Llama-3-8B configuration and persists across the additional architectures and model scales evaluated here.

D BOOTSTRAP VARIABILITY

To quantify sampling variability, we perform 1,000 bootstrap resamples of the member and non-member evaluation records for each attack. The main text reports bootstrap means for readability, while Tables 10 and 11 report the corresponding mean ± standard deviation. Bootstrap resampling is performed independently for each attack while keeping the evaluated checkpoints, attack configurations, and candidate records fixed.

E ADAPTIVE TRIMMING ATTACKS

We design two adaptive membership inference attacks to test whether an attacker can recover membership signal by filtering token scores according to RAISE's editing mechanism. Adaptive Trimmed-Loss uses the target model's token losses to identify positions to exclude. Reference-Delta Trim additionally uses a public pretrained model to rank positions by their target–reference loss differences. Together, these attacks examine whether absolute loss or relative loss provides an effective way to bypass the effect of the edit.

Table 11: Bootstrap statistics corresponding to Table 2 in the main text. MIA results are reported as mean ± standard deviation over 1,000 bootstrap resamples on Pile-CC. Perplexity measures model utility, with lower values indicating better utility. The best and second-best defense results are shown in bold and italics, respectively.

Methods	Loss	Zlib	Lowercase	Min-K% Prob	Min-K%++	Ratio	ReCaLL	Con-ReCall	Ensemble	SPV-MIA	Perplexity
Undefended	0.845 ± 0.025	0.877 ± 0.017	0.849 ± 0.024	0.664 ± 0.036	0.830 ± 0.032	0.997 ± 0.002	0.864 ± 0.025	0.746 ± 0.046	0.912 ± 0.016	0.999 ± 0.001	11.64
Noise (Rahimian et al., 2020)	0.841 ± 0.025	0.872 ± 0.018	0.841 ± 0.033	0.664 ± 0.036	0.824 ± 0.032	0.996 ± 0.002	0.859 ± 0.026	0.738 ± 0.050	0.915 ± 0.020	0.999 ± 0.001	12.51
NPO (Zhang et al., 2024)	0.819 ± 0.027	0.855 ± 0.020	0.808 ± 0.026	0.646 ± 0.039	0.800 ± 0.032	0.993 ± 0.003	0.838 ± 0.027	0.720 ± 0.048	0.919 ± 0.019	0.996 ± 0.001	12.58
Forget-CE (Maini et al., 2024)	*0.742 ± 0.031*	*0.774 ± 0.026*	*0.719 ± 0.036*	*0.604 ± 0.039*	*0.707 ± 0.035*	*0.974 ± 0.011*	*0.761 ± 0.030*	*0.643 ± 0.049*	*0.787 ± 0.040*	*0.972 ± 0.009*	11.76
RAISE	**0.516 ± 0.033**	**0.511 ± 0.029**	**0.519 ± 0.028**	**0.524 ± 0.040**	**0.508 ± 0.033**	**0.507 ± 0.051**	**0.521 ± 0.034**	**0.543 ± 0.041**	**0.677 ± 0.045**	**0.584 ± 0.043**	12.59

(a) AUC-ROC results. Values closer to 0.5 indicate weaker membership distinguishability under the prespecified attack-score orientation.

Methods	Loss	Zlib	Lowercase	Min-K% Prob	Min-K%++	Ratio	ReCaLL	Con-ReCall	Ensemble	SPV-MIA	Perplexity
Undefended	0.023 ± 0.025	0.021 ± 0.034	0.048 ± 0.102	0.034 ± 0.054	*0.015 ± 0.019*	0.886 ± 0.034	0.029 ± 0.040	0.016 ± 0.020	0.263 ± 0.323	0.954 ± 0.028	11.64
Noise (Rahimian et al., 2020)	0.023 ± 0.025	0.018 ± 0.026	0.045 ± 0.093	0.032 ± 0.048	*0.015 ± 0.019*	0.838 ± 0.046	0.030 ± 0.042	0.016 ± 0.020	0.242 ± 0.297	0.954 ± 0.028	12.51
NPO (Zhang et al., 2024)	0.022 ± 0.023	0.019 ± 0.028	0.030 ± 0.090	0.032 ± 0.048	*0.015 ± 0.019*	0.771 ± 0.051	0.028 ± 0.037	0.016 ± 0.020	0.380 ± 0.194	0.863 ± 0.052	12.58
Forget-CE (Maini et al., 2024)	*0.012 ± 0.014*	*0.017 ± 0.023*	*0.021 ± 0.062*	*0.024 ± 0.027*	*0.015 ± 0.019*	*0.393 ± 0.180*	*0.007 ± 0.022*	**0.000 ± 0.000**	**0.000 ± 0.000**	*0.607 ± 0.090*	11.76
RAISE	**0.000 ± 0.000**	**0.000 ± 0.000**	**0.003 ± 0.009**	**0.000 ± 0.000**	**0.000 ± 0.000**	**0.078 ± 0.023**	**0.000 ± 0.000**	*0.003 ± 0.009*	*0.043 ± 0.068*	**0.145 ± 0.038**	12.59

(b) TPR at 1% FPR. Lower values indicate stronger protection against high-confidence membership inference.

E.1 Design Rationale and Attacker Access

RAISE raises the losses of selected low-loss spans toward ranges supported by the defender's non-member reference data. This motivates an attacker who tries to discount positions whose losses reflect the intervention and extract membership evidence from the remaining positions. High-loss trimming tests the hypothesis that edited positions enter the upper tail of the target's loss distribution. Reference-delta trimming tests whether a public model provides a more informative ordering of positions.

The attacker has token-probability query access to the deployed target model and the candidate sequence. Reference-Delta Trim also receives access to a fixed public pretrained reference model. The attack computes a score separately for each queried checkpoint. Defender calibration records, selected spans, target intervals, layer choices, and hidden-state targets remain outside this interface. Membership labels enter only the evaluation of the completed scores.

Both attacks change the aggregation of token losses after scoring the complete sequence. Retained tokens therefore keep their original causal contexts: filtering a token's contribution to the score does not remove that token from the model input. This construction isolates the effect of adaptive score aggregation.

E.2 Adaptive Trimmed-Loss

For a candidate sequence x, let ℓ_t^M denote the negative log-probability assigned by target model M to its observed token at valid position t, conditioned on the original preceding tokens. Let T be the number of valid observed-token positions. For trimming ratio q, we retain

$$K_q = \max\{1, \lceil T(1-q) \rceil\}. \tag{26}$$

Writing the losses in ascending order as $\ell_{(1)}^M \le \cdots \le \ell_{(T)}^M$, the membership score is

$$s_{\mathrm{trim}}(x; M, q) = -\frac{1}{K_q} \sum_{i=1}^{K_q} \ell_{(i)}^M. \tag{27}$$

Larger scores indicate stronger membership evidence. The score removes exactly $T - K_q$ contributions from the highest-loss end of the sequence. At $q = 0$, it reduces to standard Loss, the negative mean loss over all valid positions.

We evaluate $q \in \{0, 0.02, 0.05, 0.10, 0.20, 0.30\}$. The primary adaptive setting is fixed at $q = 0.10$; the other nonzero ratios measure sensitivity to the amount of filtering. Every ratio applies the same scoring rule to every candidate.

Table 12: Adaptive Trimmed-Loss on Pile-CC: Entries are bootstrap mean ± sample standard deviation over 1,000 replicates. $q = 0$ is standard Loss and the bold ratio is the fixed primary setting.

	AUC		TPR at 1% FPR (%)	
q	Undefended	RAISE	Undefended	RAISE
0.00	0.845 ± 0.025	0.516 ± 0.033	2.300 ± 2.500	0.000 ± 0.000
0.02	0.834 ± 0.029	0.526 ± 0.043	6.655 ± 5.060	0.000 ± 0.000
0.05	0.822 ± 0.029	0.518 ± 0.043	6.451 ± 4.491	0.008 ± 0.126
0.10	0.811 ± 0.032	0.509 ± 0.044	5.793 ± 4.991	0.046 ± 0.332
0.20	0.784 ± 0.033	0.490 ± 0.041	4.906 ± 4.116	0.764 ± 0.973
0.30	0.758 ± 0.034	0.469 ± 0.040	4.679 ± 3.413	0.726 ± 0.954

Table 13: Reference-Delta Trim on Pile-CC. Entries are bootstrap mean ± sample standard deviation over 1,000 replicates. The Loss row is computed from these same target caches. A removes the largest signed differences; B removes the smallest. The bold ratio is the fixed primary setting.

		AUC		TPR at 1% FPR (%)	
Variant	q	Undefended	RAISE	Undefended	RAISE
Loss	0.00	0.845 ± 0.025	0.516 ± 0.033	2.300 ± 2.500	0.000 ± 0.000
A	0.05	0.788 ± 0.031	0.566 ± 0.042	5.127 ± 4.098	0.438 ± 0.840
B	0.05	0.839 ± 0.028	0.487 ± 0.043	7.028 ± 5.364	0.000 ± 0.000
A	**0.10**	0.711 ± 0.038	0.546 ± 0.043	3.372 ± 3.442	0.765 ± 0.978
B	**0.10**	0.857 ± 0.027	0.515 ± 0.043	8.297 ± 6.625	0.000 ± 0.000
A	0.20	0.487 ± 0.040	0.460 ± 0.040	1.539 ± 1.484	0.767 ± 0.971
B	0.20	0.897 ± 0.023	0.577 ± 0.043	13.283 ± 11.277	0.000 ± 0.000

E.3 Reference-Delta Trim

A public reference model supplies a second ordering that accounts for token difficulty. Let ℓ_t^R be the reference model's loss at the same observed-token position and define

$$\delta_t = \ell_t^M - \ell_t^R. \tag{28}$$

A positive difference identifies a token that the target predicts less confidently than the reference; a negative difference identifies a target-model advantage. We evaluate both ends of this ordering because an edit can reduce a target advantage without making the target loss exceed the reference loss.

Let π sort positions by increasing (δ_t, t), with the original position resolving ties. Variant A removes the largest differences and retains $I_A(q) = \{\pi_1, \ldots, \pi_{K_q}\}$. Variant B removes the smallest differences and retains $I_B(q) = \{\pi_{T-K_q+1}, \ldots, \pi_T\}$. For $v \in \{A, B\}$,

$$s_{\Delta,v}(x; M, R, q) = -\frac{1}{K_q} \sum_{t \in I_v(q)} \ell_t^M. \tag{29}$$

The reference determines which positions contribute; the final score remains an average of target-model losses. We evaluate both variants at $q \in \{0.05, 0.10, 0.20\}$ and report them separately, with $q = 0.10$ fixed as the primary ratio.

E.4 Evaluation Setup

Observed-token scoring. The trimmed-loss caches use the existing causal shift and the shifted attention mask, retaining only valid observed-token losses. The reference caches tokenize the candidate with the shared tokenizer, prepend the start-of-sequence token once, and score candidate tokens under the identity context. The start token supplies context and padding contributes no loss. The respective context caps are 512 and 2,048 tokens; all evaluated candidates fit within both caps. The attacks reuse these cached values for all trimming ratios.

Metrics and resampling. We report member-oriented AUC and TPR at 1% FPR. AUC gives half credit to tied member/non-member scores. TPR is the largest empirical true-positive rate attained by a threshold with FPR at most 0.01; thresholds preserve ties and use no interpolation. With 100 non-members, the false-positive budget is one record. Each table reports the mean and sample standard deviation across 1,000 stratified bootstrap replicates. Each replicate samples 100 members and 100 non-members with replacement and recomputes both metrics.

E.5 RESULTS

High-loss filtering leaves RAISE scores close to chance. Table 12 shows that the primary $q = 0.10$ attack yields a RAISE bootstrap-mean AUC of 0.509, compared with 0.516 for standard Loss. Its mean TPR is 0.046% at 1% FPR. Across the full sweep, the largest RAISE mean AUC is 0.526 at $q = 0.02$, while all RAISE mean TPRs remain below 0.8%. The undefended comparator retains greater separability throughout the sweep. Its AUC decreases as the trimming ratio increases, indicating that aggressive filtering can also discard useful attack information.

Reference-guided filtering yields modest residual separation. In Table 13, the matched standard-Loss baseline has mean AUC 0.845 before editing and 0.516 after RAISE. At the primary ratio, Variants A and B produce RAISE AUCs of 0.546 and 0.515, respectively. Their mean TPRs are 0.765% and 0.000%, compared with 3.372% and 8.297% on the corresponding undefended checkpoint. RAISE thus retains low measured attack performance under both directions of reference-guided filtering.

The largest observed RAISE mean AUC in this sensitivity sweep is 0.577 for Variant B at $q = 0.20$, compared with 0.897 for the same attack on the matched undefended checkpoint. Its RAISE mean TPR remains 0.000%. This maximum describes the sensitivity sweep; the primary comparison continues to use $q = 0.10$ and both fixed variants.

Implication for RAISE. The two attacks probe absolute-loss and reference-relative ways to filter token contributions. In the evaluated settings, RAISE's low measured membership-inference performance persists under the fixed 10% trimming rules, and reference-guided filtering recovers only a fraction of the matched baseline AUC difference. Across all tested trimming ratios, RAISE mean AUC remains below 0.58 and mean TPR at 1% FPR remains below 0.8%.