AUDITING PREFERENCE-BASED POST-TRAINING OF LLMS VIA STRONG MEMBERSHIP INFERENCE ATTACKS

Lorenzo Rossi, Kaif Shaikh, Franziska Boenisch, Adam Dziedzic*
CISPA Helmholtz Center for Information Security

ABSTRACT

Preference-based post-training is critical for aligning large language models (LLMs) with human intent; however, it raises privacy concerns as the instruction and feedback data used in this stage may contain sensitive information, such as personal identifiers or user-specific preferences. While membership inference attacks (MIAs) have been widely studied for pre-training and supervised fine-tuning, their effectiveness in the context of preference-based post-training remains less explored. In this work, we systematically evaluate privacy vulnerabilities in modern post-training pipelines and present a systematic analysis of *strong* MIAs for preference-based post-training. We introduce LiRA-J, a preference-aware variant of LiRA for membership inference on preference data. Through comprehensive experiments across a range of datasets and model families, we reveal privacy risks and compare the most prevalent post-training approaches, uncovering vulnerability patterns. Our analysis further examines key factors that affect privacy risk in preference-based post-training, including regularization strategies. Our findings highlight privacy vulnerabilities in preference-based post-training and underscore the need to audit aligned models with preference-aware membership inference protocols.

1 INTRODUCTION

Preference-based post-training is a key component in achieving state-of-the-art performance in large language models (LLMs) across a wide range of tasks (Team et al., 2025; Grattafiori et al., 2024; Team, 2025; Shao et al., 2024). Modern LLMs typically follow a multi-stage training pipeline that consists of a first large-scale pre-training on broad corpora to acquire general linguistic and factual competence, subsequent supervised fine-tuning (SFT) to adapt models to instruction-following behavior, and finally preference-based post-training to align model outputs with human judgments and deployment objectives (Liu et al., 2024; Zhou et al., 2023). This final stage often relies on pairwise preferences or other forms of human feedback within specialized optimization objectives, and it can substantially enhance reasoning, helpfulness, and overall response quality.

However, the reliance on high-quality instruction and feedback data raises significant privacy concerns: preference signals (*e.g.,* thumbs up/down or pairwise rankings) are often logged alongside the full prompts and responses being judged, which can include personally identifying details such as emails or account IDs, as well as sensitive information such as medical symptoms or financial information that may later be unintentionally leaked by the model. A standard way to evaluate such privacy risks is via membership inference attacks (MIAs) (Shokri et al., 2017), which aim to determine whether a given example was included in the model's training data. Prior work has demonstrated MIAs against LLMs in both pre-training and SFT regimes (Mattern et al., 2023; Shi et al., 2024; Hayes et al., 2025). Existing work on MIAs on preference-based post-training for LLM focuses primarily on theoretical insights and a limited set of methods (Feng et al., 2025). However, it does not systematically compare privacy risk across different preference-based post-training objectives, nor does it evaluate reference-based shadow-model MIAs. This leaves open how privacy risk varies across alignment methods under strong auditing protocols, a gap we address in this work.

*For correspondence, please contact Franziska Boenisch (boenisch@cispa.de) and Adam Dziedzic (dziedzic@cispa.de).

At the same time, effective privacy analysis increasingly requires evaluating *strong* MIAs. In particular, reference-free attacks that rely only on statistics from the target model can have limited effectiveness in practice, and often underestimate the MIA risk (Duan et al., 2024; Hayes et al., 2025). This has motivated *reference-based* MIAs, which calibrate membership evidence by comparing the target model's behavior to one or more independently trained reference models on the same domain, such as LiRA (Carlini et al., 2022) and RMIA (Zarifzadeh et al., 2024).

Crucially, most existing MIA methodology, including reference-based formulations, is developed and validated in pre-training or SFT settings, where the unit of training is a single pair *(prompt, response)* and objectives reduce to next-token prediction or classification. In contrast, preference-based post-training operates on tuples (x, y_w, y_l), where x is the input prompt and the training signal is a pairwise preference: the model is optimized to prefer the *winner* response y_w over the *loser* response y_l.

This structural mismatch suggests that MIA designs and evaluation protocols should be tailored to preference-based learning, rather than transferred directly from pre-training or SFT. We show empirically that this holds, and that naively porting strong MIAs from these settings can substantially misestimate privacy risk in preference-based post-training.

In this work, we systematically quantify the privacy risks of existing preference-based post-training for LLMs. Therefore, we propose a principled reference-based MIA that generalizes beyond simple loss metrics to capture the specificity of preference optimization. We conduct a comprehensive analysis across three diverse datasets (UltraFeedback (Cui et al., 2024), HH-RLHF (Bai et al., 2022), Chatbot Arena (Chiang et al., 2024)) and two model families (Qwen3 and Gemma 3). Our evaluation covers commonly used post-training methods, including SFT (Ouyang et al., 2022), DPO (Rafailov et al., 2023), IPO (Azar et al., 2024), ORPO (Hong et al., 2024), KTO (Ethayarajh et al., 2024). Finally, we isolate which factors most strongly drive privacy risk across settings, providing practical guidance for selecting post-training objectives and for designing reliable privacy evaluations.

Our contributions are as follows:

- **Limitations of Standard Assessments.** We demonstrate that standard MIAs developed for pre-training or SFT systematically underestimate privacy risks when applied to preference optimization. We show that privacy evaluations must be adapted to the tuple structure, rather than treating samples as independent input-output pairs.
- **A Tuple-Aware Evaluation Framework.** We propose a novel, preference-aware framework that explicitly leverages joint information from prompt-winner-loser tuples (x, y_w, y_l). By extending strong, reference-based MIAs to this setting (LiRA-J), our approach captures membership signals missed by methods that rely on winner-only statistics or that collapse tuples into scalar loss values.
- **Systematic Risk Quantification.** We conduct a comprehensive analysis across diverse datasets, models, and preference-based post-training methods. Using our stronger attack, we identify key vulnerability factors and reveal that leakage is driven by systematic asymmetries in which "winner" responses retain substantially more membership signal than "loser" responses.

2 Related Work

Post-Training Methods. Modern large language models (LLMs) are commonly refined after pre-training using post-training methods designed to improve instruction following, alignment, and preference adherence. A widely used baseline is *Supervised Fine-Tuning (SFT)* (Ouyang et al., 2022), which optimizes the model via likelihood maximization on curated instruction-response pairs. More recent approaches rely on preference-based objectives that leverage pairwise or groupwise comparisons instead of absolute targets. Prominent examples include *Direct Preference Optimization (DPO)* (Rafailov et al., 2023), *Identity Preference Optimization (IPO)* (Azar et al., 2024), and *Odds Ratio Preference Optimization (ORPO)* (Hong et al., 2024), which reframe preference learning as closed-form policy optimization without explicit reward models.

In parallel, scalable alternatives have been proposed to better handle noisy or heterogeneous feedback, which includes *Kahneman–Tversky Optimization (KTO)* (Ethayarajh et al., 2024). While these methods share a common goal of aligning model outputs with human preferences, they differ

substantially in their optimization objectives, statistical assumptions, and induced training dynamics. Formal definitions of these post-training methods are provided in Section A.

Membership Inference Attacks. MIAs study whether an adversary can determine if a data point x was included in the training set of a model h, thereby serving as a practical measure of empirical privacy leakage (Shokri et al., 2017; Carlini et al., 2021). In general, MIAs operate by exploiting systematic differences in a model's behavior on training versus non-training samples, as captured by statistics such as loss, likelihood, or confidence scores. For LLMs, several reference-free MIAs have been proposed due to their low computational overhead. Methods such as Min-K% (Shi et al., 2024) and Min-K%++ (Zhang et al., 2025), which rely on token probability thresholds to detect membership based on extreme token-level likelihoods. While efficient, these approaches often exhibit limited detection power and can fail to outperform random guessing in realistic settings (Duan et al., 2024). In the preference-based setting, Feng et al. (2025) propose PREMIA, which scores membership by comparing the aligned model against a fixed anchor model (the base or SFT checkpoint) via conditional likelihood ratios over the winner and loser responses. Unlike our approach, PREMIA relies on a single anchor and does not calibrate scores using independently trained reference models.

Reference-based MIAs constitute a strong class of empirical privacy audits for LLMs, as they aim to detect training-induced distributional shifts by comparing statistics from models trained with and without a candidate sample. LiRA (Carlini et al., 2022) formulates membership inference as a hypothesis test between H_{in} (the candidate x is a training member) and H_{out} (the candidate x is not a training member). Using shadow models trained with and without x, it estimates the distributions of a confidence statistic $\phi(x)$ (e.g., loss or a logit-based score) under H_{in} and H_{out}, and scores x by the likelihood ratio between the fitted densities:

$$\Lambda(x) = \frac{\mathcal{N}(\phi(x); \mu_{\text{in}}, \sigma^2_{\text{in}})}{\mathcal{N}(\phi(x); \mu_{\text{out}}, \sigma^2_{\text{out}})},$$

where $\mu_{\text{in}}, \sigma^2_{\text{in}}$ and $\mu_{\text{out}}, \sigma^2_{\text{out}}$ denote the mean and variance of $\phi(x)$ under membership and non-membership, respectively.

RMIA (Zarifzadeh et al., 2024) refines this approach by explicitly accounting for variation in intrinsic example difficulty through population normalization. Let θ denote the audited model parameters, and let $p(x)$ denote a population baseline distribution. RMIA is motivated by a Bayesian view of memorization in terms of Bayes factors, using the population-normalized likelihood ratio $p(x \mid \theta)/p(x)$ as evidence of membership. It compares the candidate x against population samples $z \sim p(z)$ and scores membership by the proportion of comparisons in which x has sufficiently stronger Bayes-factor evidence than z:

$$s_{\text{RMIA}}(x) = \frac{1}{|Z|} \sum_{z \in Z} \mathbf{1}\left[\frac{p(x \mid \theta)/p(x)}{p(z \mid \theta)/p(z)} \geq \gamma \right],$$

where Z is a set of reference points drawn from the population, $\mathbf{1}[\cdot]$ is the indicator function, and $\gamma \geq 1$ is a multiplicative margin controlling how strong the relative evidence must be for each comparison to contribute (larger γ yields a more conservative test).

InfoRMIA (Tao & Shokri, 2026) further removes RMIA's thresholded counting and replaces it with a continuous, information-theoretic population-normalized score:

$$s_{\text{InfoRMIA}}(x) = \log \frac{p(x \mid \theta)}{p(x)} + D_{\text{KL}}\big(p(z) \,\|\, p(z \mid \theta)\big),$$

where $D_{\text{KL}}(\cdot\|\cdot)$ denotes the Kullback–Leibler divergence, $p(z)$ is the population distribution over reference samples, and $p(z \mid \theta)$ denotes the model-induced distribution; for a fixed audited model θ, the KL term is constant with respect to x. In this work, we propose a preference-aware reference-based MIA that explicitly leverages the tuple structure of preference data (*e.g.,* prompt, winner response, loser response). Instead of treating each sample as a single input–output pair, our attack uses tuple-level signals to quantify the privacy risks of preference-trained LLMs.

3 A STRUCTURED REFERENCE-BASED APPROACH TO PREFERENCE-BASED MEMBERSHIP INFERENCE

We focus on the preference-based setting, where training data consists of tuples (x, y_l, y_w), with prompt x, two candidate responses y_l and y_w, and y_w preferred over y_l. Most existing preference-optimization post-training methods fit this setting.

Privacy game and strong membership inference. We follow the standard membership inference formulation (Shokri et al., 2017; Carlini et al., 2021), but instantiate it at the level of preference tuples by designing tuple-level observation statistics and decision rules. A training algorithm maps a dataset of preference tuples to a trained model. The adversary is given access to the trained model and the query tuple $z = (x, y_l, y_w)$, and must decide whether z was included in the training dataset. Performance is measured by advantage over random guessing; we also emphasize TPR at low FPR (*e.g.,* TPR @ 1% FPR) to capture the regime of high-confidence membership identification with few false positives (Carlini et al., 2022).

We consider strong MIAs that rely on multiple reference models to estimate how an observation statistic behaves when the query tuple is included versus excluded, as in LiRA (Carlini et al., 2022) and RMIA (Zarifzadeh et al., 2024). Concretely, the adversary trains K reference models that match the target model's architecture and training procedure, each on a subsample of an auxiliary dataset drawn from the same underlying distribution. Following Carlini et al. (2022), we form the reference training sets by independently including each candidate tuple with probability $1/2$ for each reference model. Consequently, each tuple appears in approximately $K/2$ reference models and is excluded from the remaining $\approx K/2$. For a given query tuple z, we partition the reference models into *IN* references and *OUT* references depending on whether their subsampled training set contains z. We use the leave-one-out evaluation procedure over reference models described by Carlini et al. (2022).

Observation statistics on prompt–response pairs. Strong MIAs require an observation statistic computed from a model on a query sample. In our setting, each tuple (x, y_l, y_w) yields two prompt–response pairs, (x, y_w) and (x, y_l). We denote by

$$V(f_\theta; x, y) \in \mathbb{R}^d \tag{1}$$

a score vector extracted from f_θ on a prompt–response pair (x, y), where d is a fixed dimensionality. Following Carlini et al. (2022), we instantiate V using the loss signals (or a scaled version of it) derived from f_θ, either as token-level NLL (Negative Log-Likelihood) values or as a rescaled version of the same signal, for example via a logit or hinge transformation. In practice, d can be made fixed by truncating or padding a token-level NLL vector to length d. We write

$$V_w := V(f_\theta; x, y_w), \qquad V_l := V(f_\theta; x, y_l). \tag{2}$$

Since the tuple (x, y_l, y_w) contains two prompt–response pairs, a key modeling choice is how to merge information from V_w and V_l into a membership decision for the tuple.

Winner-only (LiRA-W / RMIA-W / InfoRMIA-W). A direct baseline is to ignore the preference structure and query a single prompt–response pair, typically the winner pair (x, y_w). In this instantiation, we apply original scoring metrics.

Objective-scalar (LiRA-DPO / LiRA-IPO, and RMIA/InfoRMIA analogues). A second baseline reduces the tuple (x, y_l, y_w) to a single scalar derived from the preference objective and applies the standard decision rule. To instantiate this reduction, we evaluate a per-tuple preference loss under the model being attacked and use it as the observation statistic. Let f_θ denote a language model that induces a conditional distribution $p_\theta(y \mid x)$, and let f_0 denote the fixed *anchor* model used in preference optimization (typically the pre-alignment SFT checkpoint), inducing $p_0(y \mid x)$. For any tuple $z = (x, y_l, y_w)$, define the anchor-normalized log-probability ratio

$$r_\theta(x, y) := \log p_\theta(y \mid x) - \log p_0(y \mid x), \tag{3}$$

and the tuple logit

$$u_\theta(z) := r_\theta(x, y_w) - r_\theta(x, y_l). \tag{4}$$

We then obtain a scalar statistic by evaluating the corresponding per-tuple loss. For simplicity, we set $\beta = 1$ throughout:

$$s_\theta^{\mathrm{DPO}}(z) := -\log \sigma\big(u_\theta(z)\big), \tag{5}$$

$$s_\theta^{\mathrm{IPO}}(z) := \Big(u_\theta(z) - \tfrac{1}{2}\Big)^2. \tag{6}$$

A potential concern with objective-derived statistics is the explicit dependence on the anchor model p_0: both DPO and IPO are defined in terms of anchor-normalized log ratios, so it may appear that membership inference must incorporate p_0 when scoring tuples. In our reference-based calibration setting, however, we can show that the anchor term contributes no membership signal under our calibration. The p_0 terms enter $u_\theta(z)$ only through a tuple-dependent constant that is identical for all reference models, and therefore cancels in LiRA/RMIA/InfoRMIA scoring. As a result, we can ignore the anchor and use $\tilde{u}_\theta(z) := \log p_\theta(y_w \mid x) - \log p_\theta(y_l \mid x)$ without changing the membership scores; see Lemma B.1 in Section B.

LiRA-J: joint statistics for preference tuples. A preference tuple provides two complementary signals: the model's behavior on the preferred response and on the dispreferred response. In preference-based post-training, these two responses are optimized jointly, so membership can affect the model's scores on y_w and y_l in different (and potentially complementary) ways. Winner-only attacks observe only the preferred response and may miss signal that appears primarily in the rejected response. Collapsing the tuple to a single scalar can discard useful membership information. We therefore propose ***LiRA-J***, which retains both components by applying LiRA to a joint feature constructed from V_w and V_l(token-level loss feature vectors such as NLL, Stable, or Hinge, as in Carlini et al. (2022)).

Concretely, we form a joint statistic by concatenation,

$$S := \mathrm{concat}(V_w, V_l) \in \mathbb{R}^{2d}. \tag{7}$$

For a query tuple z, we evaluate S on each reference model ($K = 4$ in our experiments) and split the resulting joint scores into *IN* and *OUT* sets depending on whether the reference model's training set contains z, using the same IN/OUT construction and evaluation protocol as Carlini et al. (2022). We then fit Gaussian models to the IN and OUT sets with means μ_{in} and μ_{out}, computed as empirical averages of the joint statistics over the corresponding reference-model sets. To reduce sample complexity (and thus the number of reference models required), we assume a *shared* covariance Σ for the IN and OUT distributions and score membership using the log-likelihood ratio

$$\Lambda(S) := \log \frac{\mathcal{N}(S; \mu_{\mathrm{in}}, \Sigma)}{\mathcal{N}(S; \mu_{\mathrm{out}}, \Sigma)}. \tag{8}$$

Since the dimensionality $2d$ can be large relative to the reference-model budget, we represent each token-level feature as a fixed-length vector of dimension d by padding and use an isotropic shared covariance $\Sigma = \sigma^2 I$ with a single variance parameter. We estimate σ^2 by pooling IN and OUT residuals, computing the empirical variance separately for each dimension, and then averaging these variances across dimensions.

4 Experimental Setup

Datasets, Models, and Post-training Methods. For our experiments, we consider three datasets: UltraFeedback Binarized (Cui et al., 2024), HH-RLHF (Bai et al., 2022), and Chatbot Arena (Chiang et al., 2024). We evaluate across two model families, Qwen3 and Gemma 3, using the following checkpoints: Qwen3-0.6B, Qwen3-1.7B, Qwen3-4B, Qwen3-8B (Team, 2025) and Gemma 3 1B and Gemma 3 4B (Team et al., 2025). We focus on commonly used post-training methods: SFT (Ouyang et al., 2022), DPO (Rafailov et al., 2023), IPO (Azar et al., 2024), ORPO (Hong et al., 2024), and KTO (Ethayarajh et al., 2024). Due to computational cost, we omit KTO on Qwen3-4B/8B and KTO/DPO/IPO on Gemma 3 4B; all other (model, method) pairs are evaluated.

Reference Models, MIAs, and Evaluation Protocol. We evaluate privacy risk using strong reference-model MIAs following the evaluation protocol of Carlini et al. (2022). For each (dataset, model, method) configuration, we use 20,000 samples for evaluation and train $K = 4$ reference models on 20,000 samples drawn from the same data distribution. As baselines, we consider standard adaptations of LiRA (Carlini et al., 2022), RMIA (Zarifzadeh et al., 2024), and InfoRMIA (Tao & Shokri, 2026) that operate on the *prompt–winner* pair; we call them LiRA-W, RMIA-W, and InfoRMIA-W respectively. Additionally, we evaluate objective-scalar variants (LiRA-DPO, RMIA-DPO, and InfoRMIA-DPO) that reduce each preference tuple to a single scalar derived from the preference objective. Our primary attack, LiRA-J, is preference-aware and jointly models the winner and loser responses within a single likelihood-ratio test, retaining both components of the preference

tuple. We report attack performance using the true positive rate (TPR) at $\mathrm{FPR} = 1\%$, as emphasized by Carlini et al. (2022). We run a grid search over learning rates, batch sizes, KL penalties (when applicable), and epoch counts, and select hyperparameters to minimize differences in utility across post-training methods, where utility is measured by preference accuracy on a held-out validation set.

5 Research Questions and Experiments

We organize our empirical study around a set of research questions that characterize privacy leakage in preference-based post-training. We first evaluate whether standard MIAs, originally developed for pretraining and SFT, underestimate leakage when applied to prompt–winner–loser tuples. We then compare how membership risk varies across post-training objectives, and analyze how data properties and controllable training choices affect leakage. Each question is evaluated consistently across multiple datasets and model checkpoints to enable controlled comparisons.

5.1 RQ1: Do Current Reference-Based MIAs Underestimate the Privacy Risk?

Motivation. Most strong, reference-based MIAs for LLMs are developed and validated in pretraining or SFT regimes, where the training unit is a single prompt–response pair and the learning signal is token-level likelihood. Preference-based post-training instead operates on prompt–winner–loser tuples and updates the model through *comparisons* between two trajectories under objectives such as DPO, IPO, ORPO, and KTO, which can shift overfitting dynamics and concentrate membership signal in the *relative* behavior across winner and loser responses. Standard adaptations therefore risk a structural mismatch: winner-only attacks (LiRA-W, RMIA-W, InfoRMIA-W) discard the loser response and effectively treat preference data as SFT, while objective-scalar variants (LiRA-DPO, RMIA-DPO, InfoRMIA-DPO) compress the tuple into a single objective-derived scalar, potentially discarding token-level and cross-response information that strong, reference-based calibration can exploit. This motivates preference-aware auditing that preserves tuple structure and jointly scores winner and loser responses.

Summary of Findings. Across preference-based post-training methods, tuple-aware scoring is consistently stronger than winner-only or objective-scalar variants. In Table 1, LiRA-J achieves the best average rank (2.53) and ranks first for DPO (2.17), IPO (1.23), and KTO (1.93), indicating that winner-only scoring and objective-scalar reductions can underestimate membership risk under these objectives. In contrast, in the SFT regime, winner-only attacks remain competitive (RMIA-W ranks best at 2.41), consistent with their assumed training unit. The feature comparison in Table 2 (reported in Section D) further suggests that objective-agnostic, calibrated statistics (Stable and Hinge) are the most reliable across methods, while objective-derived scalars (DPO and IPO features) are the least reliable as attack statistics. Full per-model and per-dataset breakdowns are provided in Section E.

Detailed Results. Table 1 reports the average rank (mean $\pm$ std; lower is stronger MIA) of each strong, reference-based MIA, where ranks are computed by ordering attacks within each (model, dataset, post-training method) setting according to TPR at FPR=1% (higher indicates more leakage) and then averaging these ranks across settings for each post-training method (the "All" column averages across all methods). The table shows a clear shift between SFT and the other preference-optimization objectives. For SFT, RMIA-W is the strongest attack with average rank 2.41 and InfoRMIA-W is close at 2.69, while LiRA-J is not top-ranked (3.53). For preference-based objectives such as DPO/IPO/KTO, LiRA-J becomes the strongest overall (2.53) and is the top-ranked attack for multiple objectives: for DPO, LiRA-J achieves 2.17 compared to the next-best RMIA-W at 2.37; for IPO, LiRA-J achieves 1.23 whereas the next-best attack is substantially weaker (RMIA-DPO at 4.11); and for KTO, LiRA-J attains 1.93 compared to the next-best LiRA-DPO at 3.43. In contrast, objective-scalar variants are consistently weak in aggregate, with RMIA-DPO at 5.26 and InfoRMIA-DPO at 5.85 overall, supporting the hypothesis that compressing preference tuples to a single objective value often discards membership signal. ORPO shows a smaller gap between tuple-aware and winner-only attacks: RMIA-W achieves the best average rank (2.55), with InfoRMIA-W close behind (3.15), while LiRA-J ranks lower (3.58). This suggests that the supervised component in ORPO acts as an anchor that partially aligns it with SFT-style behavior, reducing the relative advantage of joint winner–loser scoring.

The feature ablation in Table 2 (Section D) reports the same average-rank aggregation, but using different scoring features to construct the attack statistic. It shows that Stable features achieve the best overall average rank (3.18) and are top-ranked for SFT, DPO, IPO, and KTO (e.g., 2.23 on KTO), with Hinge close behind (3.40). By contrast, objective-derived scalar features perform worst (DPO at 4.85 and IPO at 4.75), indicating that reusing the training objective as an attack statistic is not a reliable substitute for tuple-aware scoring. Full results across models and datasets, including method comparison tables, are in Section E.

Table 1: **LiRA-J is on average the strongest.** Average attack rank (mean $\pm$ std), where rank 1 denotes the attack with the highest TPR at FPR=1% within each (model, dataset, post-training method) setting, and ranks are averaged across settings. Best in bold. Second best underlined.

Attack	SFT	DPO	IPO	KTO	ORPO	All
RMIA-W	**2.41 $\pm$ 0.27**	<u>2.37 $\pm$ 0.27</u>	5.00 $\pm$ 0.31	5.30 $\pm$ 0.36	**2.55 $\pm$ 0.29**	<u>3.40 $\pm$ 0.16</u>
RMIA-DPO	5.62 $\pm$ 0.22	5.92 $\pm$ 0.17	<u>4.11 $\pm$ 0.21</u>	5.48 $\pm$ 0.28	5.29 $\pm$ 0.23	5.26 $\pm$ 0.11
InfoRMIA-W	<u>2.69 $\pm$ 0.26</u>	2.97 $\pm$ 0.22	5.95 $\pm$ 0.27	5.57 $\pm$ 0.27	<u>3.15 $\pm$ 0.27</u>	3.96 $\pm$ 0.16
InfoRMIA-DPO	6.68 $\pm$ 0.17	6.25 $\pm$ 0.22	4.89 $\pm$ 0.30	4.74 $\pm$ 0.36	6.33 $\pm$ 0.22	5.85 $\pm$ 0.13
LiRA-W	3.58 $\pm$ 0.26	5.62 $\pm$ 0.41	4.37 $\pm$ 0.39	3.68 $\pm$ 0.43	5.24 $\pm$ 0.40	4.49 $\pm$ 0.18
LiRA-DPO	5.03 $\pm$ 0.29	5.53 $\pm$ 0.33	6.39 $\pm$ 0.29	<u>3.43 $\pm$ 0.49</u>	4.58 $\pm$ 0.37	5.11 $\pm$ 0.17
LiRA-J	3.53 $\pm$ 0.39	**2.17 $\pm$ 0.39**	**1.23 $\pm$ 0.21**	**1.93 $\pm$ 0.44**	3.58 $\pm$ 0.48	**2.53 $\pm$ 0.18**

5.2 RQ2: Which Post-Training Methods Are More Vulnerable to MIA?

Motivation. Modern LLM deployment workflows employ a variety of post-training techniques that significantly differ in their training objectives. Given these differences, it is plausible that privacy risk varies substantially depending on the chosen alignment method, even when utility is held constant. Understanding how MIA vulnerability depends on the post-training method is critical for stakeholders who must balance alignment quality, safety, and data privacy when selecting a fine-tuning strategy.

Summary of Findings. Across models and datasets, SFT exhibits the highest membership leakage among the studied post-training methods when we allow the attacker to select the strongest signal from our suite of strong, reference-based MIAs. Concretely, for each (model, dataset) setting and for each post-training method, we compute the attack score as the *largest* TPR at FPR=1% attained by any strong, reference-based MIA in the suite, and then rank post-training methods within that (model, dataset) setting according to this value. Figure 1 reports the resulting average ranks aggregated over all settings (higher rank indicates higher TPR at FPR=1%, hence higher leakage). SFT consistently ranks as the most vulnerable method on average, while preference-based objectives show lower leakage overall, with IPO and KTO typically least vulnerable and ORPO intermediate. A plausible mechanism is that SFT directly maximizes likelihood on individual training trajectories, which can create example-specific likelihood spikes that strong, reference-based MIAs exploit; in contrast, preference objectives couple winner and loser responses and impose a relative, margin-like constraint that can diffuse updates across the tuple and behave like implicit regularization, reducing per-example overfitting signals.

Detailed Results. Figure 1 summarizes post-training method vulnerability by plotting the average rank (mean $\pm$ std) of each post-training method across all (model, dataset) configurations. Ranks are computed *within* each (model, dataset) configuration, where each method is assigned the largest TPR at FPR=1% achieved by any strong, reference-based MIA in our attack suite. The figure shows a clear separation between SFT and preference-based post-training: SFT attains the highest average rank (approximately 3.7), indicating that it most often yields the largest TPR at FPR=1% among the compared methods. In contrast, IPO and KTO attain the lowest average ranks (approximately 2.2–2.3), with DPO moderately higher (approximately 2.5) and ORPO higher still (approximately 2.9). Put differently, relative to IPO/KTO, SFT is worse by roughly 1.4–1.5 rank points on average, and this gap remains visible despite variability across settings (error bars).

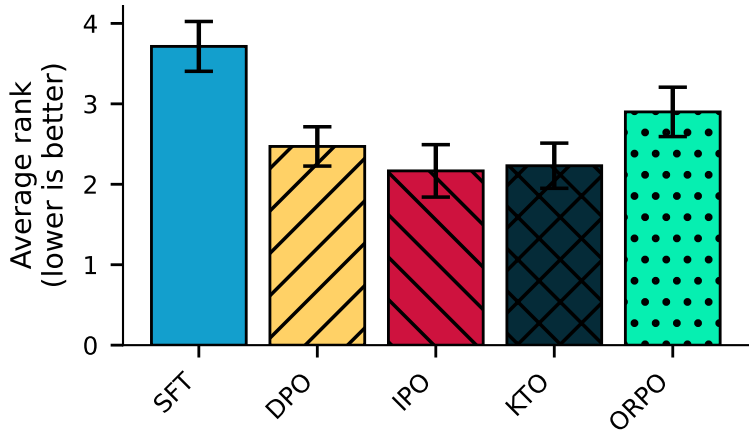


Figure 1: **Post-training method vulnerability under the strongest reference-based attack.** We compare post-training methods by their worst-case membership leakage. For each (model, dataset, method) configuration, we compute the maximum TPR at FPR$= 1\%$ attained by any strong, reference-based MIA in our suite, then rank methods within that configuration by this value (rank 1 indicates lowest leakage; higher rank indicates higher leakage). Bars report the mean rank across all configurations with $\pm$ one standard deviation. **Main takeaways:** (i) **SFT is consistently most vulnerable**, exhibiting the highest average rank, meaning it most often yields the largest TPR@1%FPR among the compared methods. (ii) **Preference-based objectives leak less overall**: IPO and KTO are typically least vulnerable, DPO is intermediate, and ORPO lies closer to SFT. (iii) Error bars indicate variability across settings, but the separation between SFT and preference-based methods remains visible across models and datasets.

5.3 RQ3: Is the Privacy Risk of Winner and Loser Responses the Same?

Motivation. Preference-based methods train on tuples that contain both a winner and a loser response for the same prompt, and the learning signal is explicitly asymmetric: the update increases preference for the winner while decreasing preference for the loser. This asymmetry can translate into different memorization and calibration behavior for the two responses. In particular, if post-training concentrates fitting capacity on making winner trajectories more likely (or more sharply separated from losers), then membership signal may be stronger for winner responses than for loser responses, even though both appear in training. Measuring whether winner and loser responses exhibit comparable membership leakage is therefore necessary to understand which parts of preference datasets are most exposed and to avoid drawing conclusions from winner-only evaluation.

Summary of Findings. Across all post-training methods, winner responses exhibit higher membership leakage than loser responses in the majority of (model, dataset) settings under our max-over-attacks definition, measured by TPR at FPR=1%. Figure 2 shows that winner responses are safer than loser responses less than half the time for every method (all bars are below 0.5), with the strongest skew for SFT and ORPO and the weakest skew for DPO. Section C provides a complementary view via Figure 4, which compares loser-response TPR at FPR=1% against winner-response TPR at FPR=1% for each (model, dataset) setting; most points fall below the diagonal $y = x$, indicating that loser responses typically yield lower leakage than winner responses for the same setting. A plausible mechanism is that preference optimization explicitly increases probability mass on winner responses, producing sharper, example-specific likelihood shifts that strong, reference-based MIAs exploit, while loser responses are pushed down more uniformly and can therefore exhibit weaker membership contrast.

Detailed Results. Figure 2 summarizes the winner–loser asymmetry as a win-rate. For each (model, dataset) setting and each post-training method, we compute winner-response leakage and loser-response leakage as the maximum TPR at FPR=1% across our suite of strong, reference-based MIAs evaluated on winner responses and loser responses, respectively. We then compute the fraction of settings for which winner-response leakage is lower than loser-response leakage; the dashed line at 0.5 corresponds to no systematic difference between winner and loser responses. The figure shows

win-rates below 0.5 for every method, meaning that winner responses are typically less safe than loser responses. Concretely, SFT has a win-rate of approximately 0.14, indicating that winner responses are safer than loser responses in only about 14% of settings; IPO and KTO are higher but still far from parity (approximately 0.28 and 0.31); ORPO is also low (approximately 0.20); and DPO is closest to parity at roughly 0.41 yet still indicates that winner responses are usually riskier. Section C provides additional context via Figure 4, which plots loser-response TPR at FPR=1% against winner-response TPR at FPR=1% for each evaluated (model, dataset) setting, with the diagonal $y = x$ denoting equal leakage. While DPO, IPO, and KTO largely concentrate near the diagonal at relatively small values, SFT and ORPO stand out as exceptions with large winner–loser gaps. For SFT, several settings have winner leakage in the 30–85 range while loser leakage stays near 2–10, for example a point at roughly $(80, 3)$, corresponding to a gap of about 77 percentage points. ORPO also contains conspicuous gaps, with points such as approximately $(65, 10)$ (gap ≈ 55 points) and $(60, 30)$ (gap ≈ 30 points). In contrast, representative DPO points lie close to parity, for example around $(22, 25)$ (gap ≈ 3 points), and KTO points cluster near the origin with small differences (for example around $(5, 4)$). Together, Figure 2 and Figure 4 show that membership leakage is systematically asymmetric across winner and loser responses, and that the magnitude of this asymmetry is more pronounced for SFT and ORPO.

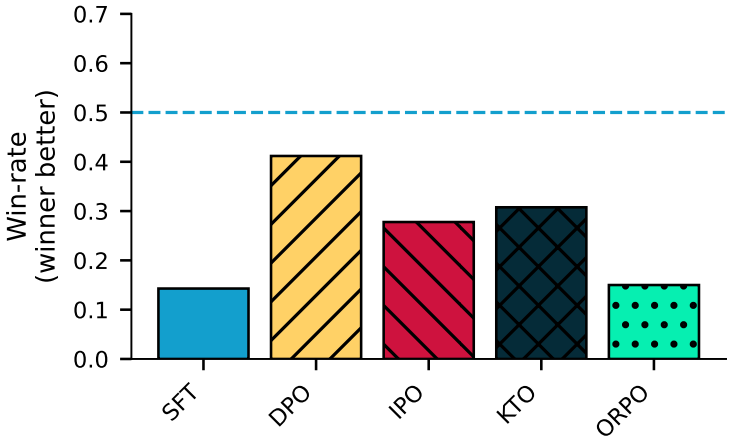
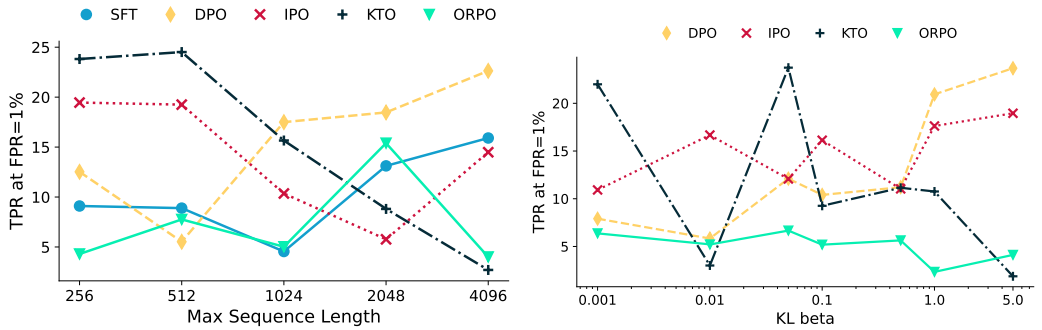


Figure 2: **Winner responses are typically more vulnerable than loser responses.** We quantify winner–loser asymmetry using a win-rate computed per post-training method. For each (model, dataset) setting, we evaluate strong, reference-based MIAs on winner responses and on loser responses separately, take the maximum TPR at FPR$= 1\%$ within each suite as the leakage score, and count a *win* when winner leakage is lower than loser leakage. Bars report the resulting win-rate aggregated across settings; the dashed line at 0.5 corresponds to no systematic difference. **Main takeaways:** (i) All methods lie below 0.5, indicating that winner responses are more often the higher-leakage component of the tuple. (ii) The asymmetry is most pronounced for SFT and ORPO (lowest win-rates), while DPO is closest to parity. (iii) This pattern supports the view that preference-style updates concentrate membership evidence on the preferred trajectory, so winner-only auditing can understate tuple-level privacy risk.

5.4 RQ4: How Critical Are Sequence Length and Regularization for Privacy?

Motivation. Modern alignment pipelines often involve long outputs (e.g., multi-step reasoning or extended responses) and frequently rely on explicit regularization to constrain post-training updates. Both factors plausibly affect privacy leakage: increasing the maximum sequence length can expand the surface over which membership signal appears, while stronger KL regularization is often motivated as a way to limit distribution shift and reduce information leakage. This question examines whether these intuitions hold empirically under our evaluation, and whether the effects are consistent across post-training methods.

Summary of Findings. Both sequence length and KL regularization have substantial, method-dependent effects on leakage, and neither exhibits a universal monotonic trend across post-training methods. Figure 3a shows that increasing the maximum sequence length can markedly increase



(a) **Leakage can increase or decrease with length depending on the objective.** LiRA-J TPR at FPR=1% versus maximum sequence length on Gemma 3 1B for each post-training method. Each point is the average across the three datasets.

(b) **Changing β does not uniformly reduce leakage and can increase it for some objectives.** LiRA-J TPR at FPR=1% versus β on Gemma 3 1B for preference-based objectives. Each point is the average across the three datasets.

leakage for some methods (notably DPO and SFT at long lengths), while decreasing leakage for others (notably KTO). Figure 3b shows that varying β can change leakage by large margins, but the direction and stability of this effect depend strongly on the objective: DPO increases at larger β, ORPO remains comparatively low across the sweep, and KTO exhibits pronounced non-monotonicity.

Detailed Results. Table 1 reports the average rank (mean $\pm$ std; lower is stronger MIA) of each strong, reference-based MIA across post-training methods. The table shows a clear shift between SFT and preference-based objectives. For SFT, winner-only attacks perform best: RMIA-W has the strongest average rank (2.41), followed by InfoRMIA-W (2.69), while LiRA-J is weaker (3.53). In contrast, for preference-based objectives such as DPO, IPO, and KTO, LiRA-J becomes the strongest overall (2.53) and ranks first for all three objectives, with average ranks of 2.17 for DPO, 1.23 for IPO, and 1.93 for KTO. By contrast, objective-scalar variants are consistently weak overall, with RMIA-DPO and InfoRMIA-DPO ranking 5.26 and 5.85, respectively, supporting the hypothesis that compressing preference tuples into a single objective value often discards membership signal. ORPO shows a smaller gap between tuple-aware and winner-only attacks: RMIA-W performs best (2.55), InfoRMIA-W is close behind (3.15), and LiRA-J ranks lower (3.58). This suggests that ORPO behaves more like SFT, reducing the relative advantage of joint winner-loser scoring. The feature ablation in Table 2 (Section D) uses the same average-rank aggregation with different scoring features. Stable achieves the best overall average rank (3.18) and ranks first for SFT, DPO, IPO, and KTO (e.g., 2.23 on KTO), with Hinge close behind (3.40). In contrast, objective-derived scalar features perform worst (DPO at 4.85 and IPO at 4.75), indicating that reusing the training objective as the attack statistic is a poor substitute for tuple-aware scoring. Full results across models and datasets, including method comparison tables, are in Section E.

6 Conclusion

In this work, we audit privacy leakage from preference-based post-training via membership inference on prompt–winner–loser tuples using strong, reference-based attacks under a consistent protocol across datasets, objectives, and model families. We show that evaluations based on a single objective scalar or winner-only statistics can underestimate risk, because preference objectives couple the winner and loser and thus create additional membership signal. To address this, we propose LiRA-J, which applies LiRA-style calibration to a joint statistic built from token-level loss features of both responses; empirically, LiRA-J is the strongest attack on average for several preference objectives, especially DPO/IPO/KTO, while winner-only attacks remain competitive for SFT. We further show that winner responses usually carry stronger membership evidence than loser responses, with the asymmetry especially pronounced for SFT and ORPO. Finally, practical training knobs can materially shift privacy risk: maximum response length and KL-related regularization can either increase or decrease leakage depending on the objective, sometimes non-monotonically, underscoring the need to evaluate privacy jointly with these choices rather than treating them as implementation details.

REFERENCES

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pp. 4447–4455. PMLR, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

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.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. ULTRAFEEDBACK: Boosting language models with scaled AI feedback. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=BOorDpKHiJ.

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? In *Conference on Language Modeling (COLM)*, 2024.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. In *International Conference on Machine Learning*. PMLR, 2024.

Qizhang Feng, Siva Rajesh Kasa, SANTHOSH KUMAR KASA, Hyokun Yun, Choon Hui Teo, and Sravan Babu Bodapati. Exposing privacy gaps: Membership inference attack on preference data for llm alignment. In Yingzhen Li, Stephan Mandt, Shipra Agrawal, and Emtiyaz Khan (eds.), *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 5221–5229. PMLR, 03–05 May 2025. URL https://proceedings.mlr.press/v258/feng25a.html.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Jamie Hayes, Ilia Shumailov, Christopher A Choquette-Choo, Matthew Jagielski, Georgios Kaissis, Milad Nasr, Meenatchi Sundaram Muthu Selva Annamalai, Niloofar Mireshghallah, Igor Shilov, Matthieu Meeus, et al. Exploring the limits of strong membership inference attacks on large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 11170–11189, 2024.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *ICLR*, 2024.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schoelkopf, 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.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=zWqr3MQuNs.

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. IEEE, 2017.

Jiashu Tao and Reza Shokri. (token-level) \textbf{InfoRMIA}: Stronger membership inference and privacy assessment for LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=4KVeb0Vv13.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Sajjad Zarifzadeh, Philippe Liu, and Reza Shokri. Low-cost high-power membership inference attacks. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 58244–58282, 2024.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. Min-k%++: Improved baseline for pre-training data detection from large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ZGkfoufDaU.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023.

A Extended Related Work: Post-training Methods

Supervised Fine-Tuning (SFT). SFT (Ouyang et al., 2022) adapts a pretrained language model to follow instructions by minimizing the token-level negative log-likelihood on supervised input–output pairs. Given a dataset $\mathcal{D}$ of prompts x and reference responses y, the objective is

$$\mathcal{L}_{\mathrm{SFT}} = -\mathbb{E}_{(x,y)\sim\mathcal{D}} \sum_{t=1}^{|y|} \log \pi_\theta(y_t \mid x, y_{<t}).$$

Direct Preference Optimization (DPO). DPO (Rafailov et al., 2023) learns from preference triplets (x, y_w, y_l) without an explicit reward model by optimizing a logistic loss on the (reference-relative) log-likelihood gap between the preferred response y_w and the loser response y_l:

$$\mathcal{L}_{\mathrm{DPO}} = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \log \sigma\Big(\beta\big[\log \tfrac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)} - \log \tfrac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)}\big]\Big),$$

where π_{ref} is a fixed reference policy, $\beta > 0$ is a temperature, and $\sigma(\cdot)$ is the logistic sigmoid.

Identity Preference Optimization (IPO). IPO (Azar et al., 2024) replaces DPO's logistic loss with a squared loss on the same reference-relative log-likelihood gap, targeting a fixed scale set by the regularization parameter (often denoted τ in the IPO paper). Using the same gap as in DPO, IPO minimizes

$$\mathcal{L}_{\mathrm{IPO}} = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Big(\big[\log \tfrac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)} - \log \tfrac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)}\big] - \tfrac{1}{\tau}\Big)^2,$$

which is designed to improve stability when preferences are close to deterministic.

Odds Ratio Preference Optimization (ORPO). ORPO (Hong et al., 2024) combines supervised imitation of the preferred response with an odds-ratio term that contrasts y_w and y_l for the same prompt. Following ORPO, let $P_\theta(y \mid x)$ denote a sequence-level likelihood proxy (e.g., exponentiated average log-likelihood), define $\mathrm{odds}_\theta(y \mid x) = \frac{P_\theta(y|x)}{1-P_\theta(y|x)}$, and optimize

$$\mathcal{L}_{\mathrm{ORPO}} = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Big[-\log \pi_\theta(y_w \mid x) - \lambda \log \sigma\Big(\log \tfrac{\mathrm{odds}_\theta(y_w|x)}{\mathrm{odds}_\theta(y_l|x)}\Big)\Big],$$

where $\lambda > 0$ controls the strength of the preference term.

Kahneman–Tversky Optimization (KTO). KTO (Ethayarajh et al., 2024) learns from unary feedback by treating the policy-reference log-ratio as an implicit reward and applying a prospect-theoretic value function. For labeled examples (x, y, s) with $s \in \{+1, -1\}$, implicit reward $r_\theta(x, y) = \log \frac{\pi_\theta(y|x)}{\pi_{\mathrm{ref}}(y|x)}$, and a prompt-dependent baseline $r_0(x)$, KTO minimizes

$$\mathcal{L}_{\mathrm{KTO}} = -\mathbb{E}_{(x,y,s)\sim\mathcal{D}}\, v\big(s \cdot (r_\theta(x, y) - r_0(x))\big),$$

where $v(\cdot)$ is the (asymmetric) value function.

B Invariance of Reference-Based MIAs to Anchor-Policy Shifts

This appendix justifies a simplification used in the objective-derived baseline in Section 3. DPO and IPO are written in terms of an *anchor-normalized* margin that depends on a fixed model f_0. At first glance, this suggests that membership inference must explicitly account for the anchor. The key observation is that, under multi-reference calibration, the anchor contributes only a constant that depends on the query tuple z but not on the model parameters. Since IN and OUT reference models are compared on the *same* tuple z, this constant is shared across both reference sets and therefore cancels from the resulting membership scores. We formalize this cancellation for LiRA, RMIA, and InfoRMIA.

Lemma B.1 (Anchor cancellation for DPO/IPO margins under LiRA, RMIA, and InfoRMIA)**.** *Let $z = (x, y_l, y_w)$ be a preference tuple. Let f_θ be a language model inducing $p_\theta(\cdot \mid x)$, and let f_0 be a fixed anchor model inducing $p_0(\cdot \mid x)$. Define the anchored margin*

$$u_\theta(z) := \Big(\log p_\theta(y_w \mid x) - \log p_0(y_w \mid x)\Big) - \Big(\log p_\theta(y_l \mid x) - \log p_0(y_l \mid x)\Big),$$

and the unanchored margin

$$\tilde{u}_\theta(z) := \log p_\theta(y_w \mid x) - \log p_\theta(y_l \mid x).$$

Then

$$u_\theta(z) = \tilde{u}_\theta(z) - c(z), \qquad c(z) := \log p_0(y_w \mid x) - \log p_0(y_l \mid x),$$

where $c(z)$ depends only on z and not on θ.

LiRA. *Fix a tuple z and suppose LiRA is calibrated on z by fitting Gaussian densities to the statistic $u_\theta(z)$ over IN and OUT reference models. Replacing $u_\theta(z)$ by $\tilde{u}_\theta(z)$ yields the same log-likelihood ratio score.*

RMIA. *Suppose RMIA is instantiated from a positive per-tuple likelihood function $L_\theta(z)$ and uses normalized ratios of the form*

$$\mathrm{LR}_\theta(z, \bar{z}) = \frac{L_\theta(z)/L(z)}{L_\theta(\bar{z})/L(\bar{z})}, \qquad L(z) = \mathbb{E}_{\theta'}\big[L_{\theta'}(z)\big],$$

where the expectation is taken over reference models θ'. If $L_\theta(z)$ is replaced by $L'_\theta(z) = a(z)\, L_\theta(z)$ for any $a(z) > 0$ independent of θ, then $\mathrm{LR}_\theta(z, \bar{z})$ is unchanged for all $\bar{z}$, hence RMIA scores are unchanged. In particular, taking $L_\theta(z) = \exp(u_\theta(z))$ implies $L_\theta(z) = \exp(-c(z)) \exp(\tilde{u}_\theta(z))$, so the anchor term contributes only through $a(z) = \exp(-c(z))$ and cancels.

InfoRMIA. *InfoRMIA is computed from the same likelihood-ratio quantities $\mathrm{LR}_\theta(z, \bar{z})$ (followed by logs and expectations). Therefore, invariance of $\mathrm{LR}_\theta(z, \bar{z})$ to $L_\theta(z) \mapsto a(z)L_\theta(z)$ implies that replacing u_θ by $\tilde{u}_\theta$ does not change InfoRMIA scores.*

Proof. **LiRA.** Fix a tuple z and consider the collection of scalar statistics obtained from reference models. Let

$$\{u^{(k)}\}_{k \in \mathcal{I}_{\mathrm{in}}(z)} \quad \text{and} \quad \{u^{(k)}\}_{k \in \mathcal{I}_{\mathrm{out}}(z)}$$

denote the values of $u_\theta(z)$ for IN and OUT reference models, and define $\tilde{u}^{(k)} := u^{(k)} + c(z)$ so that $\tilde{u}^{(k)} = \tilde{u}_\theta(z)$. Since $c(z)$ is constant across reference models for the fixed tuple z, adding $c(z)$ shifts every IN sample and every OUT sample by the same amount. Consequently, the fitted Gaussian means shift by the same amount:

$$\tilde{\mu}_{\mathrm{in}} = \mu_{\mathrm{in}} + c(z), \qquad \tilde{\mu}_{\mathrm{out}} = \mu_{\mathrm{out}} + c(z),$$

while the (shared) covariance Σ is unchanged. LiRA scores a test statistic s via the log-likelihood ratio

$$\Lambda(s) = \log \frac{\mathcal{N}(s; \mu_{\mathrm{in}}, \Sigma)}{\mathcal{N}(s; \mu_{\mathrm{out}}, \Sigma)}.$$

Evaluating the score at $\tilde{s} := s + c(z)$ under the shifted parameters gives

$$\Lambda(\tilde{s}) = \log \frac{\mathcal{N}(s + c(z); \mu_{\mathrm{in}} + c(z), \Sigma)}{\mathcal{N}(s + c(z); \mu_{\mathrm{out}} + c(z), \Sigma)} = \log \frac{\mathcal{N}(s; \mu_{\mathrm{in}}, \Sigma)}{\mathcal{N}(s; \mu_{\mathrm{out}}, \Sigma)} = \Lambda(s),$$

where the equality follows because a Gaussian density depends on s and μ only through the difference $s - \mu$. Thus the LiRA score is identical whether computed from $u_\theta(z)$ or $\tilde{u}_\theta(z)$.

RMIA. Let $L'_\theta(z) = a(z)L_\theta(z)$ where $a(z) > 0$ depends only on z. The population normalization used by RMIA becomes

$$L'(z) = \mathbb{E}_{\theta'}\big[L'_{\theta'}(z)\big] = \mathbb{E}_{\theta'}\big[a(z)L_{\theta'}(z)\big] = a(z)\, \mathbb{E}_{\theta'}\big[L_{\theta'}(z)\big] = a(z)L(z).$$

Therefore the normalized likelihood is unchanged:

$$\frac{L'_\theta(z)}{L'(z)} = \frac{a(z)L_\theta(z)}{a(z)L(z)} = \frac{L_\theta(z)}{L(z)}.$$

Applying the same argument to $\bar{z}$ shows that both the numerator and denominator in

$$\mathrm{LR}_\theta(z, \bar{z}) = \frac{L_\theta(z)/L(z)}{L_\theta(\bar{z})/L(\bar{z})}$$

are unchanged, hence $\mathrm{LR}_\theta(z, \bar{z})$ is unchanged for all $\bar{z}$. RMIA aggregates these ratios over choices of $\bar{z}$ using a fixed rule, so the final RMIA score is unchanged. Finally, for the anchored margin,

$$\exp(u_\theta(z)) = \exp(\tilde{u}_\theta(z) - c(z)) = \exp(-c(z)) \exp(\tilde{u}_\theta(z)),$$

which is exactly the multiplicative factor $a(z) = \exp(-c(z))$ covered by the argument above.

InfoRMIA. InfoRMIA applies logs and expectations to the same likelihood-ratio quantities $\mathrm{LR}_\theta(z, \bar{z})$. Since $\mathrm{LR}_\theta(z, \bar{z})$ is unchanged when $L_\theta(z)$ is multiplied by $a(z)$, any function of these ratios constructed via logs and averaging is also unchanged. Therefore InfoRMIA scores are identical whether computed from $u_\theta(z)$ or $\tilde{u}_\theta(z)$. □

C Comparison between winner and loser responses.

Figure 4 compares TPR at FPR=1% obtained from the loser response against the winner response, across all analyzed model variants, datasets and post-training methods. Most points lie below the line indicating lower leakage from loser response. Methods like SFT and ORPO exhibit large gaps with high leakage on the winner response and near random performance on the loser response. In contrast, DPO, IPO, and KTO cluster closer to the line and at lower absolute leakage levels, suggesting more balanced leakage across responses. These results confirm that privacy leakage in preference based post-training is not symmetric across responses. Membership signal concentrates primarily on the winner responses, consistent with objectives that explicitly increase likelihood on preferred trajectories, while loser responses are often suppressed more uniformly.

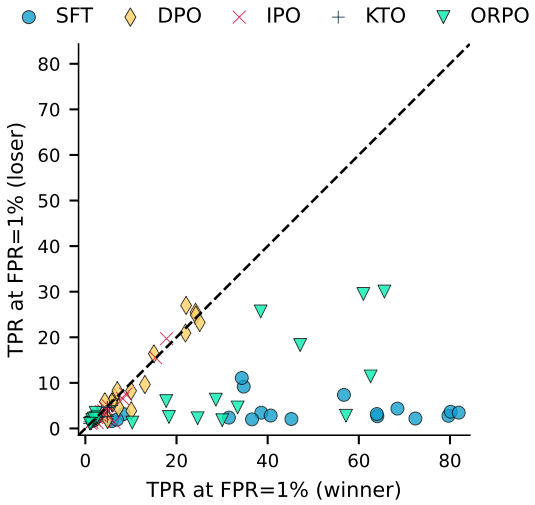


Figure 4: **Loser responses indicate systematically lower leakage from with particularly large variations for SFT and ORPO.** Comparison of membership leakage between winner and loser responses. TPR at FPR = 1% for the loser response versus the winner response across all evaluated (model, dataset, post-training method) configurations. Each point corresponds to one configuration; the diagonal indicates equal leakage.

D Comparison of different scoring metrics.

Table 2 reports average MIA rank across different scoring features used to instantiate strong reference based MIAs. Lower rank corresponds to stronger leakage detection. Across post-training methods, Stable and Hinge features consistently outperform raw loss and logit based features. Stable (logit scale) feature achieves the best overall average rank, performing strongly for SFT, DPO, IPO and KTO, while the Hinge score is the second best. In contrast, objective-derived scalar features (*e.g.,* RMIA-DPO, LiRA-DPO) perform poorly overall. This indicates that directly reusing the training objective as an attack statistic does not help with a strong signal for membership inference as it does not capture relevant token-level information and is not the most reliable feature for membership

signal. Overall, Stable and Hinge features consistently yield stronger and more robust signals across post-training methods.

Table 2: Average MIA rank (mean ± std) by feature.

Attack	SFT	DPO	IPO	KTO	ORPO	All
Stable	**3.32 ± 0.28**	**2.81 ± 0.30**	**2.98 ± 0.27**	**2.23 ± 0.23**	4.22 ± 0.35	**3.18 ± 0.14**
Hinge	3.70 ± 0.33	2.82 ± 0.35	3.38 ± 0.34	2.58 ± 0.29	**4.00 ± 0.44**	3.40 ± 0.17
Loss	5.32 ± 0.34	3.71 ± 0.33	3.38 ± 0.21	2.92 ± 0.25	4.73 ± 0.34	4.16 ± 0.15
Logit	4.08 ± 0.36	3.88 ± 0.36	3.04 ± 0.36	3.38 ± 0.32	4.77 ± 0.38	3.88 ± 0.17
DPO	3.84 ± 0.36	5.06 ± 0.28	6.17 ± 0.24	4.58 ± 0.33	4.55 ± 0.39	4.85 ± 0.16
IPO	3.79 ± 0.33	5.12 ± 0.28	5.53 ± 0.23	5.00 ± 0.32	4.45 ± 0.45	4.75 ± 0.15

E Full Comparison Between Different MIAs

We run the same of experiments on different set of models and datasets. Our results include the following models: **Qwen 3 0.6B** (Table 3, Table 9, Table 15), **Qwen 3 1.7B** (Table 4, Table 10, Table 16), **Qwen 3 4B** (Table 5, Table 11, Table 17), **Qwen 3 8B** (Table 6, Table 12, Table 18), **Gemma 3 1B** (Table 7, Table 13, Table 19) and **Gemma 3 4B** (Table 8, Table 14, Table 20). Our results show the impact of implemented MIAs varying by different model families as well as model sizes. LiRA-J consistently outperforms other methods across different post-training objectives on different models. With the increase in model size the gap between the membership scores from LiRA-J and other methods increases substantially. RMIA-W does consistently well with methods like SFT and ORPO but has poor performance with other methods like DPO, IPO and KTO.

Table 3: TPR at FPR=1% for different MIAs on HH-RLHF with Qwen 3 0.6B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	**29.6**	4.0	1.1	0.9	18.0
RMIA-DPO	5.5	1.3	1.4	0.6	**12.8**
InfoRMIA-W	6.2	1.3	1.0	1.0	**26.7**
InfoRMIA-DPO	5.5	1.4	1.5	0.6	**13.0**
LiRA-W	**10.0**	0.6	1.0	1.1	7.7
LiRA-DPO	7.1	1.0	0.7	1.9	**8.3**
LiRA-J	9.4	**13.7**	0.0	0.2	0.4

Table 4: TPR at FPR=1% for different MIAs on HH-RLHF with Qwen 3 1.7B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	**8.4**	2.4	1.1	0.9	3.8
RMIA-DPO	2.1	1.2	1.3	0.6	**2.2**
InfoRMIA-W	6.3	2.3	1.1	0.9	**9.8**
InfoRMIA-DPO	**4.6**	1.3	1.3	0.7	2.9
LiRA-W	**2.2**	0.6	0.7	0.2	0.9
LiRA-DPO	1.2	1.2	0.7	**1.5**	1.0
LiRA-J	**24.0**	11.3	15.1	17.7	23.0

F Ablation on the Sequence length

Table 21 reports TPR at FPR=1% for the Gemma 3 1B Model on all datasets, evaluated at different maximum sequence lengths. We report results for all post-training objectives using LiRA-J method with the Stable feature. Across datasets, the effect of sequence length is highly method-dependent: for example, on UltraFeedback Binarized, KTO changes from 26.43 (256) to 0.04 (2048), while DPO

Table 5: TPR at FPR=1% for different MIAs on HH-RLHF with Qwen 3 4B.

MIA	SFT	DPO	IPO	ORPO
RMIA-W	**27.8**	2.5	1.2	2.2
RMIA-DPO	**5.6**	1.4	1.6	1.2
InfoRMIA-W	**28.0**	2.5	1.1	1.9
InfoRMIA-DPO	**5.7**	1.2	1.5	1.2
LiRA-W	**10.9**	1.1	3.0	2.2
LiRA-DPO	**6.6**	0.4	1.0	1.4
LiRA-J	10.7	0.0	19.3	**20.2**

Table 6: TPR at FPR=1% for different MIAs on HH-RLHF with Qwen 3 8B.

MIA	SFT	DPO	IPO	ORPO
RMIA-W	**27.2**	5.1	1.5	1.5
RMIA-DPO	**5.4**	1.5	1.5	0.8
InfoRMIA-W	**27.0**	5.2	1.5	1.5
InfoRMIA-DPO	**5.4**	1.3	1.5	2.1
LiRA-W	**12.4**	0.3	1.2	0.6
LiRA-DPO	7.2	1.5	1.1	**7.9**
LiRA-J	**30.1**	20.8	4.6	26.1

ranges from 2.37 (512) to 20.96 (1024). On HH-RLHF Helpful, DPO moves from 0.05 (256) / 0.04 (1024) up to 30.43 (4096), whereas KTO attains 34.11 (512) but reaches 0.65 (4096). On Chatbot Arena, SFT increases from 4.24 (256) to 32.71 (4096), while ORPO reaches 34.09 (2048) but records 3.63 (4096); IPO shows poor results at 1024–2048 (1.17–1.28) compared to 24.55 (256).

G Ablation on the KL regularization

Table 22 Table 23 and Table 24 report LiRA-J TPR at FPR=1% under a KL sweep over $\beta \in \{0.001, 0.01, 0.05, 0.1, 0.5, 1.0, 5.0\}$ for Gemma 3 1B across all datasets used. DPO shows high sensitivity to β across datasets. On UltraFeedback Binarized and HH-RLHF Helpful, DPO ranges from poor results at smaller β values to substantially higher TPR at moderate values. IPO exhibits even higher sensitivity towards changes in KL values. KTO spans from very low to high TPR values depending on the regularization strength, indicating that strong performance is limited to narrow KL values. In contrast to other methods, ORPO remains quite stable across the KL sweep.

Table 7: TPR at FPR=1% for different MIAs on HH-RLHF with Gemma 3 1B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	7.9	3.3	1.2	1.1	**12.1**
RMIA-DPO	3.8	1.1	1.6	1.0	**7.7**
InfoRMIA-W	6.6	2.9	1.1	1.0	**20.8**
InfoRMIA-DPO	5.0	1.4	2.0	1.2	**7.7**
LiRA-W	**3.2**	0.7	0.1	0.5	0.5
LiRA-DPO	3.1	0.2	0.9	1.9	**4.0**
LiRA-J	3.8	0.0	3.4	0.1	**7.8**

Table 8: TPR at FPR=1% for different MIAs on HH-RLHF with Gemma 3 4B.

MIA	SFT	ORPO
RMIA-W	**39.7**	14.9
RMIA-DPO	6.6	**9.4**
InfoRMIA-W	**39.7**	14.4
InfoRMIA-DPO	6.5	**10.2**
LiRA-W	**25.1**	5.0
LiRA-DPO	**3.5**	2.9
LiRA-J	**34.3**	13.5

Table 9: TPR at FPR=1% for different MIAs on Ultrafeedback with Qwen 3 0.6B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	29.8	18.1	1.0	1.0	**55.6**
RMIA-DPO	**5.6**	2.0	1.0	1.0	4.2
InfoRMIA-W	**54.6**	9.7	1.7	0.9	48.6
InfoRMIA-DPO	**5.6**	1.4	1.1	1.3	4.1
LiRA-W	**11.5**	3.4	1.9	3.4	10.2
LiRA-DPO	9.5	3.9	1.5	1.4	**17.6**
LiRA-J	3.1	**27.2**	3.2	6.4	2.8

Table 10: TPR at FPR=1% for different MIAs on Ultrafeedback with Qwen 3 1.7B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	4.5	15.0	1.3	0.9	**32.7**
RMIA-DPO	1.5	**1.9**	1.1	1.0	1.3
InfoRMIA-W	**36.3**	10.0	1.4	0.9	23.0
InfoRMIA-DPO	1.6	**2.5**	1.2	1.3	1.2
LiRA-W	1.3	1.6	1.8	0.2	**2.1**
LiRA-DPO	1.6	**3.8**	1.6	1.2	3.1
LiRA-J	6.0	**18.8**	5.0	0.5	7.6

Table 11: TPR at FPR=1% for different MIAs on Ultrafeedback with Qwen 3 4B.

MIA	SFT	DPO	IPO	ORPO
RMIA-W	**76.1**	30.7	17.8	8.8
RMIA-DPO	**42.6**	1.2	2.2	1.1
InfoRMIA-W	**72.2**	22.5	14.0	6.5
InfoRMIA-DPO	**42.8**	5.1	1.9	1.2
LiRA-W	**53.1**	5.1	2.1	0.0
LiRA-DPO	**45.8**	7.2	3.2	11.0
LiRA-J	2.1	1.2	19.2	**32.4**

Table 12: TPR at FPR=1% for different MIAs on Ultrafeedback with Qwen 3 8B.

MIA	SFT	DPO	IPO	ORPO
RMIA-W	**77.6**	24.2	15.4	1.3
RMIA-DPO	**46.3**	2.5	2.0	0.9
InfoRMIA-W	**72.5**	20.2	13.3	1.3
InfoRMIA-DPO	**46.7**	3.4	2.0	1.2
LiRA-W	**52.4**	6.1	2.5	0.2
LiRA-DPO	**50.6**	6.3	2.4	2.7
LiRA-J	6.9	2.1	**15.0**	1.6

Table 13: TPR at FPR=1% for different MIAs on Ultrafeedback with Gemma 3 1B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	**44.6**	13.5	1.2	0.9	35.5
RMIA-DPO	3.3	1.5	1.3	1.0	**3.7**
InfoRMIA-W	**52.9**	3.7	1.2	1.0	43.4
InfoRMIA-DPO	3.4	1.8	1.4	1.3	**3.5**
LiRA-W	**5.8**	1.8	0.0	0.3	4.3
LiRA-DPO	7.7	2.7	0.3	2.4	**11.7**
LiRA-J	0.5	9.3	3.4	**13.4**	3.4

Table 14: TPR at FPR=1% for different MIAs on Ultrafeedback with Gemma 3 4B.

MIA	SFT	ORPO
RMIA-W	**81.6**	6.7
RMIA-DPO	**36.0**	2.0
InfoRMIA-W	**73.2**	32.8
InfoRMIA-DPO	**36.1**	1.7
LiRA-W	**56.1**	1.8
LiRA-DPO	**43.2**	9.1
LiRA-J	**10.0**	6.8

Table 15: TPR at FPR=1% for different MIAs on Chatbot Arena with Qwen 3 0.6B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	1.0	1.0	**1.0**	1.0	1.0
RMIA-DPO	0.6	0.8	**1.7**	0.9	0.7
InfoRMIA-W	1.0	1.1	0.9	**1.2**	0.9
InfoRMIA-DPO	0.8	0.7	**1.6**	0.9	0.8
LiRA-W	2.6	2.1	0.1	1.4	**2.7**
LiRA-DPO	1.2	**1.6**	0.7	0.3	1.3
LiRA-J	0.8	1.8	**24.6**	20.1	1.5

Table 16: TPR at FPR=1% for different MIAs on Chatbot Arena with Qwen 3 1.7B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	1.0	1.0	**1.2**	1.0	1.0
RMIA-DPO	0.7	1.1	**1.4**	0.8	0.7
InfoRMIA-W	1.1	**2.7**	1.1	1.4	1.1
InfoRMIA-DPO	0.8	1.1	**1.4**	0.9	0.8
LiRA-W	1.9	2.3	0.3	**2.9**	2.3
LiRA-DPO	1.5	1.6	**1.8**	1.6	1.6
LiRA-J	2.8	14.2	3.7	**33.2**	2.3

Table 17: TPR at FPR=1% for different MIAs on Chatbot Arena with Qwen 3 4B.

MIA	SFT	DPO	IPO	ORPO
RMIA-W	1.0	1.0	**1.2**	1.0
RMIA-DPO	0.8	1.6	**2.0**	1.1
InfoRMIA-W	5.7	**7.0**	1.2	2.3
InfoRMIA-DPO	0.8	1.5	**1.9**	1.1
LiRA-W	1.8	2.3	1.6	**2.6**
LiRA-DPO	1.1	1.5	0.4	**1.6**
LiRA-J	3.1	**12.7**	8.7	8.5

Table 18: TPR at FPR=1% for different MIAs on Chatbot Arena with Qwen 3 8B.

MIA	SFT	DPO	IPO	ORPO
RMIA-W	1.0	1.0	**1.4**	1.1
RMIA-DPO	1.0	1.4	**2.0**	1.3
InfoRMIA-W	**6.8**	5.8	1.2	2.5
InfoRMIA-DPO	1.0	1.4	**1.7**	1.5
LiRA-W	1.6	**2.4**	1.9	2.3
LiRA-DPO	1.6	**2.2**	0.8	2.1
LiRA-J	3.6	**5.3**	3.1	4.8

Table 19: TPR at FPR=1% for different MIAs on Chatbot Arena with Gemma 3 1B.

MIA	SFT	DPO	IPO	KTO	ORPO
RMIA-W	1.0	**12.6**	1.1	1.1	1.0
RMIA-DPO	0.7	1.6	**2.2**	0.8	0.8
InfoRMIA-W	1.4	**6.1**	1.1	1.1	0.9
InfoRMIA-DPO	0.8	1.6	**2.2**	1.0	0.8
LiRA-W	1.3	**1.6**	0.0	1.5	1.2
LiRA-DPO	0.9	**3.6**	1.3	0.2	0.8
LiRA-J	4.2	1.4	1.2	**15.0**	3.9

Table 20: TPR at FPR=1% for different MIAs on Chatbot Arena with Gemma 3 4B.

MIA	SFT	ORPO
RMIA-W	**1.3**	1.0
RMIA-DPO	**1.0**	0.6
InfoRMIA-W	**1.4**	1.0
InfoRMIA-DPO	1.0	**1.1**
LiRA-W	**1.6**	0.6
LiRA-DPO	1.6	**7.6**
LiRA-J	1.7	**24.1**

Table 21: TPR at FPR=1% for different sequence lengths on all datasets. Attack Method: LiRA-J.

Method	**UltraFeedback Binarized**					**HH-RLHF Helpful**					**Chatbot Arena**				
	256	512	1024	2048	4096	256	512	1024	2048	4096	256	512	1024	2048	4096
SFT	**16.10**	12.79	2.65	6.88	11.18	6.96	**8.26**	6.01	6.63	3.84	4.24	5.62	4.98	25.84	**32.71**
DPO	17.31	2.37	**20.96**	17.15	4.10	0.05	6.78	0.04	16.44	**30.43**	20.21	7.42	31.49	21.82	**33.37**
IPO	**18.28**	16.76	11.90	16.01	17.73	15.54	**18.62**	17.99	0.02	18.44	**24.55**	22.38	1.17	1.28	7.29
KTO	**26.43**	14.18	18.93	0.04	7.39	28.95	**34.11**	16.98	18.70	0.65	16.07	**25.25**	11.06	7.69	0.06
ORPO	6.71	**9.79**	3.18	7.89	5.47	3.01	**9.10**	8.44	4.19	2.90	3.18	4.39	3.45	**34.09**	3.63

Table 22: TPR at FPR=1% for different KL (β) values on UltraFeedback Binarized with Gemma 3 1B. MIA: LiRA-J.

Method	0.001	0.01	0.05	0.1	0.5	1.0	5.0
DPO	23.53	9.27	6.93	10.86	0.03	**23.89**	19.61
IPO	7.56	4.22	19.75	**24.42**	15.15	14.09	14.71
KTO	31.88	2.74	**34.10**	2.14	2.04	1.86	1.81
ORPO	**9.76**	5.30	7.52	7.70	4.27	1.90	3.94

Table 23: TPR at FPR=1% for different KL (β) values on HH-RLHF Helpful with Gemma 3 1B. MIA: LiRA-J.

Method	0.001	0.01	0.05	0.1	0.5	1.0	5.0
DPO	0.04	0.30	0.03	8.18	**33.17**	31.56	22.10
IPO	19.97	17.64	0.10	0.43	16.11	**33.72**	9.49
KTO	0.00	0.13	11.50	25.16	**27.80**	17.50	0.36
ORPO	6.06	7.15	8.57	4.37	**8.70**	1.32	5.02

Table 24: TPR at FPR=1% for different KL (β) values on Chatbot Arena with Gemma 3 1B. MIA: LiRA-J.

Method	0.001	0.01	0.05	0.1	0.5	1.0	5.0
DPO	0.16	7.93	**29.35**	12.16	0.43	7.36	29.31
IPO	5.24	28.17	16.38	23.54	1.84	5.06	**32.67**
KTO	**34.09**	6.14	25.61	0.47	3.61	12.93	3.45
ORPO	3.29	3.18	3.90	3.50	**3.93**	3.79	3.39