COUNTERFACTUAL AND VERBATIM MEMORIZATION DISAGREE: FROM IN-CONTEXT LEARNING THEORY TO LLM DATA SELECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) can memorize their training data, but whether such memorization is desirable remains an open question. Some forms of memorization are associated with verbatim training-data leakage, while others capture the contribution of individual training samples to model generalization. This creates an apparent dilemma: suppressing memorization could reduce verbatim training-data leakage, but may also hurt model utility. This paper shows that the dilemma is largely metric-dependent. We study two notions of LLM memorization: counterfactual memorization (CF-Mem), which characterizes how model behavior changes when a sample is included in or excluded from training, and verbatim memorization (VB-Mem), which measures how easily training content can be reproduced verbatim from an LLM. Through an in-context learning (ICL)-theoretic analysis, we prove that these two notions are driven by distinct mechanisms and can be inversely related: samples with higher CF-Mem tend to exhibit lower VB-Mem, while CF-Mem is positively associated with LLM generalization. This disagreement suggests that not all forms of memorization are harmful. Motivated by these findings, we propose a CF-Mem-guided LLM training data selection method. Our method efficiently estimates CF-Mem using a small proxy model and allocates more token budget to samples with higher CF-Mem scores. This yields a smaller training set that preserves samples important for generalization while reducing those associated with high VB-Mem. Across reasoning benchmarks, our method substantially reduces verbatim extraction risk and training tokens while preserving most of the full-data fine-tuning performance.

1 INTRODUCTION

Large language models (LLMs) can memorize their training data, but whether such memorization is desirable remains an open question. While many theoretical and empirical studies have shown that memorization can contribute to model generalization on rare or long-tail samples (Feldman, 2020; Brown et al., 2021; Morris et al., 2026; Ye et al., 2026; Zhou et al., 2026), memorized training content can also be reproduced verbatim through LLM generation (Carlini et al., 2021; Schwarzschild et al., 2024; Nasr et al., 2025). Such verbatim training-data leakage can expose sensitive or copyrighted content and has already raised substantial legal concerns (Brittain, 2024; Leipold & Wolfe, 2024; Brittain, 2026). Taken together, these observations reveal an apparent dilemma: suppressing memorization may reduce verbatim training-data leakage, but may also remove training information that is important for model utility and generalization. Resolving this dilemma therefore requires a more precise understanding of what different notions of LLM memorization actually measure.

Existing notions of LLM memorization can be roughly divided into two families. The first is *counterfactual memorization (CF-Mem)*, which aims to disentangle memorization from generalization by examining how LLM behavior changes when a sample is included in or excluded from training (Zhang et al., 2023). The second is *verbatim memorization (VB-Mem)*, which measures how easily a training sample can be reproduced verbatim from an LLM, typically by prompting the model with natural or specially constructed inputs that elicit the target content (Carlini et al., 2021; Schwarzschild et al., 2024; Hayes et al., 2025; Sander et al., 2025). Although both are commonly

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.

referred to as "LLM memorization", their relationship to each other remains unclear. More importantly, how they relate to LLM generalization is also not yet well understood.

To better understand these relationships, to our knowledge, this paper presents the first theoretical analysis that jointly characterizes CF-Mem and VB-Mem for LLMs under a unified framework based on in-context learning (ICL) theory (Von Oswald et al., 2023; Zhang et al., 2024; Fu et al., 2025). We study linear transformers trained on noisy ICL linear-classification tasks and derive closed-form solutions for both CF-Mem and VB-Mem. Our results show that the two memorization notions capture fundamentally different properties of the training-data distribution: rare or long-tail samples that are dissimilar to most training samples tend to have higher CF-Mem but lower VB-Mem, whereas samples surrounded by many similar training samples exhibit the opposite behavior. We further derive a high-probability generalization upper bound and show that the bound becomes tighter as the average CF-Mem of the training data increases. Together, these results suggest that not all forms of LLM memorization are necessarily harmful: CF-Mem is associated with model generalization, while VB-Mem is more closely related to verbatim training-data leakage.

In light of these theoretical findings, we further propose a CF-Mem-guided training data selection strategy for LLMs. The key principle is to retain samples with higher CF-Mem scores more frequently while removing more samples with lower scores. Our theory suggests that this favors samples that are more important for generalization while reducing those with higher verbatim memorization risk. To make the method practical, we estimate CF-Mem using a lightweight proxy model rather than repeatedly training the target LLM, motivated by our finding that CF-Mem largely reflects properties of the training-data distribution rather than the specific model. Experiments on real-world LLMs show that our method substantially reduces verbatim extraction risk while preserving model utility compared with random subsets under the same token budget.

2 Related Work

In-context learning (ICL) theory. ICL theory studies how transformer-based LLMs make predictions from task-specific sequential inputs (*i.e.*, "prompts") without updating model parameters. It has been successfully applied to analyze LLM training (Lin et al., 2024; Wu et al., 2024; Zhang et al., 2025), generalization (Chen et al., 2024; Shi et al., 2024; Frei & Vardi, 2025), and robustness (Fu et al., 2025; Li et al., 2025; Kumano et al., 2026). One line of work constructs transformers layer by layer to emulate learning different function classes (Von Oswald et al., 2023; Ahn et al., 2023; Mahankali et al., 2024; Wang et al., 2024). More recent studies use simplified linear-attention models as tractable proxies for analyzing ICL prediction and generalization (Zhang et al., 2024; Frei & Vardi, 2025; Lu et al., 2025; Fu & Wang, 2026). Our analysis follows this second line of work and is the first to apply ICL theory to LLM memorization.

Memorization in LLMs. Memorization has long been studied in deep learning, particularly for its role in generalization on rare or long-tail samples (Zhang et al., 2017; Arpit et al., 2017; Feldman, 2020; Feldman & Zhang, 2020; Brown et al., 2021; Zhou et al., 2026). Recent work shows that memorizing training data, especially factual knowledge, can benefit LLM generalization (Morris et al., 2026; Ye et al., 2026), while also enabling verbatim reproduction of training content and raising privacy and copyright concerns (Carlini et al., 2021; Nasr et al., 2025). Existing audits broadly study two notions: counterfactual memorization (CF-Mem), which measures how including or excluding a sample affects LLM performance (Zhang et al., 2023), and verbatim memorization (VB-Mem), which measures how easily training content can be reproduced through adversarial prompting (Carlini et al., 2021; Schwarzschild et al., 2024; Nasr et al., 2025) or probabilistic extraction (Hayes et al., 2025; Sander et al., 2025). Our work theoretically shows that CF-Mem and VB-Mem characterize different properties of training data and relate differently to generalization and verbatim leakage.

Data selection for LLMs. Existing LLM data selection methods primarily aim to improve training efficiency or model performance, using signals include data relevance (Xie et al., 2023b), mixture importance (Xie et al., 2023a), diversity (Tirumala et al., 2023), data quality (Liu et al., 2024; Just et al., 2026), and sample influence (Xia et al., 2024; Wang et al., 2025; Ding et al., 2026). Although data cleaning techniques such as deduplication can indirectly reduce memorization and repeated generation (Lee et al., 2022), existing data selection methods generally do not treat verbatim training-data leakage as an explicit optimization objective. Our work instead uses CF-Mem to select training data with the explicit goal of preserving model utility while reducing verbatim leakage.

3 PRELIMINARIES

Large language models (LLMs). Let $\mathcal{V} := \{1, \ldots, |\mathcal{V}|\}$ be a token set. An LLM is a parameterized distribution p_θ that maps sequential token inputs to sequential token outputs. For an input prompt $x = (x_1, \ldots, x_{|x|}) \in \mathcal{V}^{|x|}$, the likelihood of generating an output response $y = (y_1, \ldots, y_{|y|}) \in \mathcal{V}^{|y|}$ is $p_\theta(y \mid x) = \prod_{i=1}^{|y|} p_\theta\big(y_i \mid x \oplus y_{1:(i-1)}\big)$, where "$\oplus$" concatenates token sequences.

Memorization in LLMs. Given a training set $\mathcal{D} = \{z = (x, y)\}$ and a stochastic training algorithm $\mathcal{A}$ that produces an LLM $p_\theta \sim \mathcal{A}(\mathcal{D})$, memorization studies how information from individual training samples is internalized and reproduced by the trained model. Existing notions of memorization can be divided into *counterfactual memorization (CF-Mem)* and *verbatim memorization (VB-Mem)*.

CF-Mem aims to distinguish the memorization of a specific training sample from generalization. Its key idea is to compare model performance on the sample when it is included in or excluded from training. Formally, CF-Mem is defined as follows.

Definition 1 (Counterfactual Memorization (Zhang et al., 2023)). *Let $\mathcal{A}$ be a stochastic training algorithm that maps a dataset $T \subseteq \mathcal{D}$ to a trained model, and let $M(p_\theta, z)$ measure the performance of model p_θ on an sample z. The counterfactual memorization of a training sample $z \in \mathcal{D}$ is*

$$\mathrm{Mem}_{\mathrm{CF}}(z, \mathcal{D}, \mathcal{A}) := \mathop{\mathbb{E}}_{T \subset \mathcal{D},\, z \in T}[M(\mathcal{A}(T), z)] - \mathop{\mathbb{E}}_{T' \subset \mathcal{D},\, z \notin T'}[M(\mathcal{A}(T'), z)].$$

The expectations are taken over both the sampling of the training subsets and the randomness of $\mathcal{A}$.

In practice, counterfactual memorization can be estimated by training models on random subsets of $\mathcal{D}$ and comparing their average performance on z between the subsets that include and exclude it.

In contrast, VB-Mem characterizes whether training data can be reproduced exactly by the trained LLM. A common way to quantify VB-Mem is to provide the model with a prefix of a training sequence and evaluate whether it can reproduce the remaining suffix. We adopt *probabilistic discoverable extraction* (Hayes et al., 2025) to efficiently audit VB-Mem by measuring how likely a training suffix is to be exactly reproduced from its prefix over repeated generations.

Definition 2 ((n, p)-Discoverable Extraction (Hayes et al., 2025)). *Given a training sample $z := x \oplus y$ that is split into an a-token prefix $z_{1:a}$ and a k-token suffix $z_{a+1:a+k}$, let $\hat{z}^{(k,i)} \sim p_\theta(\cdot|z_{1:a})$ denote the i-th independent autoregressive generation of k tokens from the LLM p_θ, starting from the same prefix $z_{1:a}$. The training example z is (n, p)-discoverably extractable if*

$$\mathbb{P}(\exists i \in [n],\ \hat{z}^{(k,i)} = z_{a+1:a+k}) \geq p.$$

The probability is taken over the randomness of the n independent autoregressive generations.

Intuitively, (n, p)-discoverable extraction measures the probability that the target suffix is reproduced at least once across multiple generations. Unlike CF-Mem, which requires comparing models trained with and without a sample, VB-Mem can be evaluated directly from a trained LLM.

4 THEORETICAL ANALYSIS FOR LLM MEMORIZATION

This section studies LLM memorization through the in-context learning (ICL) theory. We will first introduce the theoretical framework of learning ICL linear classification, then theoretically characterize the relationship between counterfactual memorization and extraction-based memorization, and finally connect both memorization metrics to LLM generalization.

4.1 LEARNING ICL LINEAR CLASSIFICATION

ICL theory and linear classification. In the ICL theory, a task-specific *prompt* input is a sequence $(x_{\tau,1}, y_{\tau,1}, \cdots, x_{\tau,N}, y_{\tau,N}, x_{\tau,q})$, where τ is the task index, $\{(x_{\tau,i}, y_{\tau,i})\}_{i=1}^N$ are the N in-context training samples, and $x_{\tau,q}$ is the query sample. For an ICL model, its goal is to make a prediction for the query sample $x_{\tau,q}$ based on the first N in-context training samples in the ICL prompt input.

We focus on the task family of ICL linear classification. For task τ, let $w_\tau \in \mathbb{R}^d$ be its corresponding task weight drawn from $w_\tau \sim U(\mathbb{S}^{d-1})$, where $U(\mathbb{S}^{d-1})$ denotes the uniform distribution on the

unit sphere $\mathbb{S}^{d-1}$. For each ICL training point $x_{\tau,i} \in \mathbb{R}^d$ $(1 \le i \le N)$ and the query point $x_{\tau,q} \in \mathbb{R}^d$, we assume they are drawn via $x_{\tau,i} := y_{\tau,i} w_\tau + \xi_{\tau,i}$ and $x_{\tau,q} := y_{\tau,q} w_\tau + \xi_{\tau,q}$, where $y_{\tau,i}, y_{\tau,q} \sim U(\{\pm 1\})$ are classification labels and can only be -1 or 1, and $\xi_{\tau,i}, \xi_{\tau,q} \sim \mathcal{N}(0, \Lambda_\xi)$ are noise with covariance $\Lambda_\xi \in \mathbb{R}^{d\times d}$. Then, the ICL input E_τ specified by the task τ is given by

$$E_\tau := \begin{pmatrix} X_\tau & x_{\tau,q} \\ Y_\tau & 0 \end{pmatrix} = \begin{pmatrix} x_{\tau,1} & \cdots & x_{\tau,N} & x_{\tau,q} \\ y_{\tau,1} & \cdots & y_{\tau,N} & 0 \end{pmatrix} \in \mathbb{R}^{(d+1)\times(N+1)}. \tag{1}$$

Linear transformers and generalization risk. Linear transformers have been widely adopted in the theoretical analysis for ICL. We follow Zhang et al. (2024) to study the single-layer *linear self-attention (LSA)* model $f_{\mathrm{LSA},\theta}$, a specific type of linear transformer, defined as follows,

$$f_{\mathrm{LSA},\theta}(E_\tau) := E_\tau + W^V E_\tau \cdot \frac{E_\tau^\top W^{KQ} E_\tau}{N},$$

where $\theta := (W^V, W^{KQ})$ is the model parameter, $W^V \in \mathbb{R}^{(d+1)\times(d+1)}$ is the value projection matrix, and $W^{KQ} \in \mathbb{R}^{(d+1)\times(d+1)}$ is a matrix fused from the key and query projection matrices. For the query point $x_{\tau,q}$, its predicted classification label is calculated as $\hat{y}_{q,\theta}(E_\tau) := \mathrm{Sign}(h_\theta(E_\tau))$, where $h_\theta(E_\tau)$ is the prediction logit and is given by the right-bottom entry of the output matrix of the LSA model, *i.e.*, $h_\theta(E_\tau) := f_{\mathrm{LSA},\theta}(E_\tau)_{(d+1)\times(N+1)}$. We partition W^V and W^{KQ} as $W^\square = \begin{pmatrix} W_{11}^\square & w_{12}^\square \\ (w_{21}^\square)^\top & w_{22}^\square \end{pmatrix}$, $\square \in \{V,\ KQ\}$, where $W_{11}^\square \in \mathbb{R}^{d\times d}$, $w_{22}^\square \in \mathbb{R}$, and $w_{12}^\square, w_{21}^\square \in \mathbb{R}^{d\times 1}$. Then, the LSA logit function h_θ can be written as follows,

$$h_\theta(E_\tau) := f_{\mathrm{LSA},\theta}(E_\tau)_{(d+1)\times(N+1)} = \begin{pmatrix} (w_{21}^V)^\top & w_{22}^V \end{pmatrix} \cdot \frac{E_\tau E_\tau^\top}{N} \cdot \begin{pmatrix} W_{11}^{KQ} \\ (w_{21}^{KQ})^\top \end{pmatrix} \cdot x_{\tau,q}.$$

Without loss of generality, we follow Zhang et al. (2024) and Frei & Vardi (2025) to fix w_{21}^V and w_{21}^{KQ} to $0_{d\times 1}$. The LSA logit function h_θ thus simplifies to

$$h_\theta(E_\tau) := w_{22}^V \cdot \frac{1}{N} Y_\tau X_\tau^\top \cdot W_{11}^{KQ} \cdot x_{\tau,q}. \tag{2}$$

Since the simplified logit function depends on the LSA parameters only through $w_{22}^V W_{11}^{KQ}$, we henceforth use $\theta := w_{22}^V W_{11}^{KQ}$ to denote this effective parameter. The generalization risk $\mathcal{R}(\theta)$ for the LSA classifier $\hat{y}_{q,\theta}$ is then measured through the binary classification error as below,

$$\mathcal{R}(\theta) := \mathbb{P}_{\tilde{w}_\tau}(\hat{y}_{q,\theta}(\tilde{E}_\tau) \ne y_{\tau,q}) = \mathop{\mathbb{E}}_{\tilde{w}_\tau \sim U(\mathbb{S}^{d-1})} \mathop{\mathbb{E}}_{\tilde{E}_\tau | \tilde{w}_\tau} \mathbb{1}[y_{\tau,q} \cdot h_\theta(\tilde{E}_\tau) \le 0], \tag{3}$$

where $\mathbb{1}[\cdot]$ is the indicator function, the logit function $h_\theta(\tilde{E}_\tau)$ is given by Eq. (2), and $\tilde{E}_\tau$ is generated according to Eq. (1) given the task weight $\tilde{w}_\tau$.

Training algorithm $\mathcal{A}$. Let $\mathcal{S} := \{w_i\}_{i=1}^B$ be a training set of B ICL tasks. To train an LSA $f_{\mathrm{LSA},\theta}$ on the set $\mathcal{S}$, the stochastic training algorithm $\mathcal{A}$ draws one ICL input E_τ according to Eq. (1) for every $w_\tau \in \mathcal{S}$ and optimizes θ via the following regularized max-margin problem,

$$\min_\theta \mathcal{L}(\theta, \mathcal{S}) := \min_\theta \Big\{ \frac{1}{B} \sum_{\tau=1}^B -y_{\tau,q} \cdot h_\theta(E_\tau) + \frac{\lambda}{2} \|w_{22}^V W_{11}^{KQ}\|_F^2 \Big\}, \tag{4}$$

where $\lambda > 0$ is the regularization coefficient. The training goal of Eq. (4) is to encourage positive prediction margins $y_{\tau,q} h_\theta(E_\tau)$, which also occur in the risk $\mathcal{R}(\theta)$ in Eq. (3). It also admit a closed-form solution $\theta_* \sim \mathcal{A}(\mathcal{S})$, as shown in the following Lemma 4.1 (see Appendix A.2 for the proof).

Lemma 4.1 (Closed-form LSA solution)**.** *Let $\xi'_{\tau,i} := y_{\tau,i}\xi_{\tau,i}$ and $\xi'_{\tau,q} := y_{\tau,q}\xi_{\tau,q}$. Algorithm $\mathcal{A}$ learns the effective parameter θ_* through Eq. (4) as follows,*

$$\theta_* = (w_{22}^V W_{11}^{KQ})_* = \frac{1}{\lambda B} \sum_{\tau=1}^B \Big(w_\tau + \frac{1}{N} \sum_{i=1}^N \xi'_{\tau,i} \Big) (w_\tau + \xi'_{\tau,q})^\top. \tag{5}$$

Thus $\theta_* \sim \mathcal{A}(\mathcal{S})$, with randomness induced by the Gaussian noise $\xi'_{\tau,i}, \xi'_{\tau,q} \sim \mathcal{N}(0, \Lambda_\xi)$.

4.2 ICL MEMORIZATION

We now formalize counterfactual memorization and verbatim memorization within the ICL framework, and also analyze the relationship between them.

ICL CF-Mem. We apply Definition 1 to the task-level samples $w_\tau \in \mathcal{S}$. Following the generalization risk in Eq. (3), we use the expected signed prediction margin $M(\theta, w_\tau) := \mathbb{E}_{E_\tau|w_\tau}[y_{\tau,q} h_\theta(E_\tau)]$ as the performance measure, where E_τ is an independently sampled ICL prompt for task w_τ. A larger $M(\theta, w_\tau)$ indicates better prediction performance. Under this measure, CF-Mem admits a closed-form solution, as shown in the following Theorem 4.2 (see Appendix A.3 for the proof).

Theorem 4.2 (Closed-form solution of ICL CF-Mem)**.** *Suppose the performance measure function is $M(\theta, w_\tau) := \mathbb{E}_{E_\tau|w_\tau}[y_{\tau,q} h_\theta(E_\tau)]$, $\mathcal{A}(\varnothing) = 0_{d\times d}$, and $\kappa_B := \frac{2^B - B - 1}{B 2^{B-1}}$. For any task-level training sample $w_\tau \in \mathcal{S}$, its CF-Mem score admits the following closed-form formalization:*

$$\mathrm{Mem}_{\mathrm{CF}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) = \frac{1}{\lambda 2^{B-1}} + \frac{\kappa_B}{\lambda}\Big(1 - \frac{1}{B-1}\sum_{\tau \neq \tau_0} (w_\tau^\top w_{\tau_0})^2\Big).$$

ICL VB-Mem. Discoverable extraction in Definition 2 cannot be directly applied to our ICL setting because it relies on autoregressive generation, while our model performs binary classification. We therefore follow its likelihood intuition: a training sample is more strongly memorized when assigned a higher likelihood. For binary ICL classification, this likelihood increases with the signed margin $y_{\tau,q} h_\theta(E_\tau)$. We thus use the signed margin on the training prompt as a tractable measure of VB-Mem, where a larger value indicates stronger memorization of the training sample.

Definition 3 (ICL verbatim memorization)**.** *For any task-level training sample $w_{\tau_0} \in \mathcal{S}$, let $E^{\mathrm{tr}}_{\tau_0}$ denote the ICL prompt sampled for w_{τ_0} during training following Eq. (1), and let θ_* be the resulting trained parameter. The ICL VB-Mem score for w_{τ_0} is defined as*

$$\mathrm{Mem}_{\mathrm{VB}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) := \mathop{\mathbb{E}}_{\theta_* \sim \mathcal{A}(\mathcal{S})}[y^{\mathrm{tr}}_{\tau_0,q} h_{\theta_*}(E^{\mathrm{tr}}_{\tau_0})].$$

Under Definition 3, ICL VB-Mem also admits a closed-form solution, given in the following Theorem 4.3 (see Appendix A.4 for the proof).

Theorem 4.3 (Closed-form solution of ICL VB-Mem)**.** *Suppose $c_\xi := (1 + \frac{\mathrm{tr}(\Lambda_\xi)}{N})(1 + \mathrm{tr}(\Lambda_\xi))$. For any task-level training sample $w_\tau \in \mathcal{S}$, its ICL verbatim memorization admits the following closed-form formalization:*

$$\mathrm{Mem}_{\mathrm{VB}}(w_{\tau_0}, S, \mathcal{A}) = \frac{1}{\lambda B}\Big[c_\xi + \sum_{\tau \neq \tau_0} (w_\tau^\top w_{\tau_0})^2\Big].$$

Relationship between CF-Mem and VB-Mem. Theorems 4.2 and 4.3 show that CF-Mem and VB-Mem depend on the same task-similarity term $\sum_{\tau \neq \tau_0} (w_\tau^\top w_{\tau_0})^2$, but in opposite directions. This yields a negative relationship, as shown in the following Corollary 4.4.

Corollary 4.4 (Relationship between CF-Mem and VB-Mem)**.** *Under the assumptions of Theorems 4.2 and 4.3, for any $w_{\tau_0} \in \mathcal{S}$,*

$$\mathrm{Mem}_{\mathrm{CF}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) = C - \frac{B\kappa_B}{B-1}\,\mathrm{Mem}_{\mathrm{VB}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}),$$

where $C := \frac{1}{\lambda 2^{B-1}} + \frac{\kappa_B (B - 1 + c_\xi)}{\lambda (B-1)}$ is independent of w_{τ_0}.

Thus, task samples surrounded by many similar training tasks have higher VB-Mem but lower CF-Mem, while rare or dissimilar task samples exhibit the opposite behavior. This shows that the two memorization notions capture fundamentally different properties of the training data.

4.3 CONNECTING MEMORIZATION WITH GENERALIZATION

In the previous section, we showed that CF-Mem and VB-Mem are negatively related and capture different properties of the training data. We now study how these two notions relate to model generalization. We first derive a generalization upper bound for the ICL training algorithm in Section 4.1, as stated in the following Theorem 4.5 (see Appendix A.5 for the proof).

Theorem 4.5 (High-probability upper bound on ICL generalization risk)**.** *Let $\theta_* \sim \mathcal{A}(\mathcal{S})$ be the trained model parameter. For any $\delta \in (0,1)$ and $0 < \eta < \frac{\sqrt{N}}{\|\Lambda_\xi\|_2 L_\delta}$, with probability at least $1-\delta$ over the randomness of $\mathcal{A}$,*

$$\mathcal{R}(\theta_*) \le \exp\Big(\eta\epsilon_\delta + \frac{\eta^2 L_\delta^2 V_\xi}{2(1-\eta L_\delta \|\Lambda_\xi\|_2/\sqrt{N})}\Big) \mathop{\mathbb{E}}_{\tilde{w}_\tau}\Big[\exp\Big(-\frac{\eta}{\lambda B}\sum_{\tau'=1}^{B}(\tilde{w}_\tau^\top w_{\tau'})^2\Big)\Big],$$

where $w_{\tau'} \in \mathcal{S}$, $\tilde{w}_\tau \sim U(\mathbb{S}^{d-1})$, $V_\xi := \|\Lambda_\xi\|_2^2/N + \|\Lambda_\xi\|_2/N + \|\Lambda_\xi\|_2$, and

$$L_\delta := \frac{1}{\lambda}\Big(1 + \frac{\|\Lambda_\xi\|_2^{1/2}}{\sqrt{N}}\Big(\sqrt{d} + \sqrt{2\log\frac{4B}{\delta}}\Big)\Big)\Big(1 + \|\Lambda_\xi\|_2^{1/2}\Big(\sqrt{d} + \sqrt{2\log\frac{4B}{\delta}}\Big)\Big),$$

$$\epsilon_\delta := \frac{2}{\lambda}\Big[\sqrt{\frac{2(d\log 9 + \log\frac{4}{\delta})}{B}\Big(\Big(1+\frac{1}{N}\Big)\|\Lambda_\xi\|_2 + \frac{\|\Lambda_\xi\|_2^2}{N}\Big)} + \frac{\|\Lambda_\xi\|_2}{B\sqrt{N}}\Big(d\log 9 + \log\frac{4}{\delta}\Big)\Big].$$

Theorem 4.5 shows that generalization depends on how well the training tasks cover a fresh task direction through $\tilde{w}_\tau^\top M_\mathcal{S} \tilde{w}_\tau$, where $M_\mathcal{S} := B^{-1}\sum_{\tau=1}^{B} w_\tau w_\tau^\top$. Meanwhile, Theorems 4.2 and 4.3 show that the CF-Mem and VB-Mem of each training sample w_τ depend on its similarity to the other training tasks. Averaging these sample-wise scores over $\mathcal{S}$ connects them to the overall pairwise similarity $\|M_\mathcal{S}\|_F^2 = B^{-2}\sum_{\tau,\tau'}(w_\tau^\top w_{\tau'})^2$. Specifically, a larger average CF-Mem corresponds to a smaller $\|M_\mathcal{S}\|_F^2$, while a larger average VB-Mem corresponds to a larger $\|M_\mathcal{S}\|_F^2$.

Since $\mathrm{tr}(M_\mathcal{S}) = 1$, a smaller $\|M_\mathcal{S}\|_F^2$ means that the training-task directions are less concentrated and more evenly distributed. Thus, training sets containing more high-CF-Mem and low-VB-Mem samples provide broader coverage of unseen task directions, while low-CF-Mem and high-VB-Mem samples are associated with more redundant directions. This provides a generalization perspective on the opposite behaviors of CF-Mem and VB-Mem.

5 LLM Memorization and Training Data Selection

So far, we have developed an ICL-theoretic framework for analyzing different notions of memorization in LLMs. We now leverage these theoretical results to design an LLM training data selection method to reduce *verbatim memorization* while preserving *model generalization*.

5.1 Implications of ICL Memorization Theory

Our theoretical analysis leads to two main implications. ***First, CF-Mem and VB-Mem capture different properties of training data.*** Theorems 4.2 and 4.3 show that rare or dissimilar samples tend to have higher CF-Mem but lower VB-Mem, while samples similar to many training samples show the opposite pattern. ***Second, different notions of memorization have different relationships with model generalization.*** Our analysis in Theorem 4.5, together with Theorems 4.2 and 4.3, shows that higher average CF-Mem and lower average VB-Mem are associated with better model generalization. This suggests that memorization useful for generalization can differ from memorization associated with verbatim data leakage.

These findings motivate a CF-Mem-guided training data selection strategy. The main principle is to prioritize samples with higher CF-Mem scores when reducing the training set, while removing lower-CF-Mem samples more aggressively. In this way, the selected data preserve more information associated with model generalization while reducing samples that are more likely to exhibit high verbatim memorization and training-data leakage.

5.2 CF-mem-guided Data Selection

We now describe our CF-Mem-guided data selection. Given a training set $\mathcal{D} = \{z_i\}_{i=1}^n$, let $\|z_i\|_{\mathrm{tok}}$ denote the number of tokens in sample z_i. Given a token budget ratio $R \in (0,1]$, our goal is to select a subset $\mathcal{D}' \subseteq \mathcal{D}$ such that $\sum_{z_i\in\mathcal{D}'}\|z_i\|_{\mathrm{tok}} \le R\sum_{z_i\in\mathcal{D}}\|z_i\|_{\mathrm{tok}}$, while prioritizing samples with higher CF-Mem scores. The CF-Mem selection consists of three steps: (1) sort samples by their CF-Mem scores and partition them into bins; (2) assign each bin a selection probability using

Algorithm 1 CF-Mem-Guided Training Data Selection

Input: Training set $\mathcal{D} = \{z_i\}_{i=1}^n$, CF-Mem scores $\{\widehat{m}_i^{\mathrm{CF}}\}_{i=1}^n$, token budget ratio R, number of bins K.
Output: Selected training set $\mathcal{D}'$.

Sort and relabel samples such that $\widehat{m}_1^{\mathrm{CF}} \le \widehat{m}_2^{\mathrm{CF}} \le \cdots \le \widehat{m}_n^{\mathrm{CF}}$
for $k = 1, \ldots, K$ **do**
  $\mathcal{B}_k \leftarrow \{z_i\}_{i=\lfloor (k-1)n/K \rfloor + 1}^{\lfloor kn/K \rfloor}$
  $C_k \leftarrow \sum_{z_i \in \mathcal{B}_k} \|z_i\|_{\mathrm{tok}}$
end for
$\phi(x) \leftarrow \sin^2(\frac{\pi x}{2})$; $r \leftarrow \frac{\sum_{k=1}^K \phi(k/K) C_k}{\sum_{k=1}^K C_k}$; $c \leftarrow \frac{1-R}{1-r}$; $p_{\min} \leftarrow \frac{R-r}{1-r}$
for $k = 1, \ldots, K$ **do**
  $p_k \leftarrow p_{\min} + c \cdot \phi(k/K)$
  Randomly shuffle samples in $\mathcal{B}_k$
  Select the longest prefix $\mathcal{S}_k \subseteq \mathcal{B}_k$ satisfying $\sum_{z_i \in \mathcal{S}_k} \|z_i\|_{\mathrm{tok}} \le p_k C_k$
end for
return $\mathcal{D}' := \cup_{k=1}^K \mathcal{S}_k$

a non-decreasing function; and (3) sample examples from each bin according to the resulting token budget. The overall procedure is summarized in Algorithm 1.

Partitioning training set by sorted CF-Mem scores. Let $\widehat{m}_i^{\mathrm{CF}}$ denote the estimated CF-Mem score of sample z_i. We first sort the training samples by their CF-Mem scores and relabel them such that $\widehat{m}_1^{\mathrm{CF}} \le \widehat{m}_2^{\mathrm{CF}} \le \cdots \le \widehat{m}_n^{\mathrm{CF}}$. Given a positive integer $K \le n$, we partition the sorted samples into K consecutive bins $\mathcal{B}_1, \ldots, \mathcal{B}_K$, where $\mathcal{B}_k := \{(z_i, \widehat{m}_i^{\mathrm{CF}})\}_{i=\lfloor (k-1)n/K \rfloor + 1}^{\lfloor kn/K \rfloor}$. Thus, $\mathcal{B}_1$ contains samples with the lowest CF-Mem scores and $\mathcal{B}_K$ contains those with the highest, while the numbers of samples across bins differ by at most one.

Bin-wise selection and token allocation. We next assign a selection rate and a corresponding token selection budget to each bin. Let $\phi : [0,1] \to [0,1]$ be a non-decreasing function with $\phi(0) = 0$ and $\phi(1) = 1$. We define a selection function $\varphi : [0,1] \to [0,1]$ as

$$\varphi(x) := p_{\min} + (\min\{p_{\min} + c, 1\} - p_{\min}) \cdot \phi(x), \qquad x \in [0,1], \tag{6}$$

where $p_{\min} \in (0,1]$ and $c \ge 0$ are hyperparameters of φ. For each bin $\mathcal{B}_k$, we assign a selection rate $p_k = \varphi(k/K)$ based on this selection function. Let $C_k := \sum_{z_i \in \mathcal{B}_k} \|z_i\|_{\mathrm{tok}}$ denote the total number of tokens in the bin $\mathcal{B}_k$. The corresponding token budget is therefore $p_k C_k$, meaning that the selected samples from $\mathcal{B}_k$ should contain at most $p_k C_k$ tokens in total.

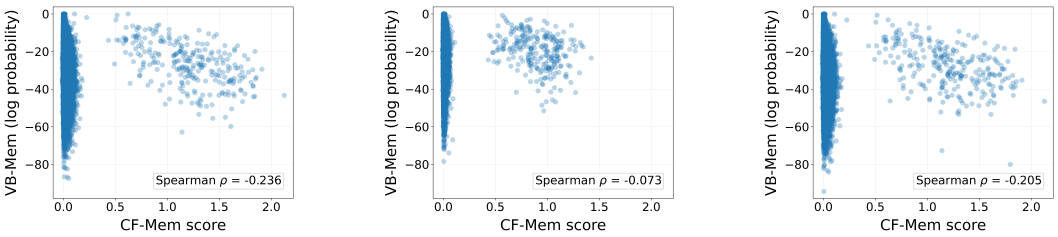
The motivation behind the selection-rate function φ is to assign larger selection rates to bins with higher CF-Mem scores, thereby preserving more high-CF-Mem samples in the selected training set. Since ϕ is non-decreasing, the selection rate increases with the CF-Mem rank of each bin. The lower endpoint $p_{\min} > 0$ prevents low-CF-Mem regions from being completely removed. Meanwhile, $p_{\max} := \min\{p_{\min} + c, 1\}$ is the selection rate assigned to the highest-CF-Mem bin and controls how strongly high-CF-Mem samples are preserved.

We next discuss how to choose the non-decreasing function ϕ and the hyperparameters $p_{\min}$ and c in the selection function φ defined in Eq. (6). For ϕ, we simply adopt a cosine schedule as below,

$$\phi(x) := \frac{1}{2}\Big(1 - \cos(\pi x)\Big) = \sin^2(\frac{\pi x}{2}), \qquad x \in [0,1].$$

This choice is inspired by the successful application of cosine annealing for learning-rate scheduling. The resulting function ϕ smoothly increases from 0 to 1, progressively allocates larger selection rates and token budgets to bins with higher CF-Mem scores. Meanwhile, for the hyperparameters $p_{\min}$ and c, given a target token budget ratio R, we ideally require two conditions: the total allocated token budget exactly matches $R \sum_{k=1}^K C_k$, and $p_{\max} = 1$, so that the highest-CF-Mem bin is fully preserved and the selection rates span the largest possible range under the given budget. These two conditions lead to the following system of equations, which admits a closed-form solution:

$$\begin{cases} p_{\max} := p_{\min} + c = 1, \\ \displaystyle\sum_{k=1}^{K} \Big(p_{\min} + c \cdot \phi(\frac{k}{K})\Big) C_k = R \sum_{k=1}^{K} C_k \end{cases} \quad \Rightarrow \quad \begin{cases} c = \dfrac{1-R}{1-r}, \\ p_{\min} = \dfrac{R-r}{1-r}, \end{cases}$$



(a) Qwen2.5-7B, Mixture-of-Thoughts (b) Qwen2.5-7B, OpenThoughts-114K (c) Llama-3.1-8B, Mixture-of-Thoughts

Figure 1: Empirical relationship between CF-Mem and VB-Mem. For each setting, we first train the target model on the full 50K training set and then compute the VB-Mem score of each training sample. CF-Mem scores are independently estimated using the Qwen2.5-0.5B proxy model. Each point represents one training sample, and the reported Spearman correlation measures the rank correlation between CF-Mem and VB-Mem.

where $r := \frac{\sum_{k=1}^{K} \phi(k/K) C_k}{\sum_{k=1}^{K} C_k}$. With this closed-form solution, the selection procedure no longer requires hyperparameter tuning and only requires specifying the target token budget ratio R.

Sampling under the token budget. Finally, we draw training samples from each bin according to the token budget determined in the previous step. For each bin $\mathcal{B}_k$, we randomly select samples without replacement until the total number of selected tokens reaches the allocated budget $p_k C_k$. Since we select complete training samples, the exact token count may not perfectly match the budget. The selected samples from all bins are then combined to form the final selected training set.

Efficient CF-Mem estimation. We estimate CF-Mem following the random-subset procedure of Zhang et al. (2023). Across L runs, we randomly sample a fraction ρ of the training set as $\mathcal{D}_l$, train a model θ_l on each subset, and evaluate all training samples using M. For each sample z_i, we estimate $\widehat{m}_i^{\mathrm{CF}} = \mathrm{Avg}_{l:z_i \in \mathcal{D}_l} M(\theta_l, z_i) - \mathrm{Avg}_{l:z_i \notin \mathcal{D}_l} M(\theta_l, z_i)$. To reduce computation, we estimate CF-Mem using a small proxy model instead of the target LLM. Theorem 4.2 suggests that relative CF-Mem is largely determined by a sample's relationship with the training distribution. Since our method only uses CF-Mem rankings, a small proxy model provides an efficient approximation without repeatedly training the target LLM.

6 Experiments

In this section, we empirically evaluate our CF-Mem-guided data selection method for preserving model utility while reducing verbatim training-data leakage.

6.1 Experimental Setup

Data selection. Given a training set, we estimate CF-Mem using Qwen2.5-0.5B (Qwen et al., 2024) as a lightweight proxy model following the random-subset procedure in Section 5. Each subset contains 10% of the training samples, with sequences truncated to 16K tokens. We then apply Algorithm 1 using the estimated CF-Mem scores. We compare against two token-matched baselines: *Random*, which samples uniformly, and *Longest*, which ranks samples by sequence length and otherwise follows the same selection procedure. See Appendix B.1 for details.

Model training. We train on two reasoning datasets, OpenThoughts-114K and Mixture-of-Thoughts, using the first 50K samples from each. Full-parameter fine-tuning is performed on the Qwen2.5-7B and Llama-3.1-8B base models with LLaMA-Factory. Training uses AdamW with an initial learning rate of 1×10^{-4}, cosine learning-rate scheduling, batch size 128, and a maximum sequence length of 32K tokens, with over-length samples dropped. See Appendix B.2 for details.

Evaluations. For model utility, we evaluate on five mathematical benchamrk: GSM8K, MATH-500, AMC23, AIME25, and AIME26. For the sampling hyperparmaeters, we set temperature as 0.6, top-p as 0.95, and the max token length as 32K.

Table 1: Performance of different training data selection methods on Qwen2.5-7B. All selection methods use a 40% token budget.

Training Set	Data Selection	GSM8K Avg@8	MATH-500 Avg@8	AMC23 Avg@8	AIME25 Pass@8	AIME26 Pass@8	Average
Original Model		79.79	59.82	32.81	20.00	13.33	41.15
OpenThoughts-114K (50K)	Random (40%)	90.19	75.42	54.37	33.33	36.67	58.00
	Longest (40%)	88.79	73.00	48.75	36.67	33.33	56.11
	CF-Mem (40%) (Ours)	89.62	74.50	51.88	40.00	36.67	58.53
Mixture-of-Thoughts (50K)	Random (40%)	89.01	73.20	47.19	36.67	20.00	53.21
	Longest (40%)	88.45	72.30	46.25	36.67	20.00	52.73
	CF-Mem (40%) (Ours)	88.77	72.60	50.62	26.67	26.67	53.07

Table 2: VB-Mem auditing results on Qwen2.5-7B. Each value is the generation-probability threshold (%) at the corresponding top VB-Mem percentile. Lower values indicate lower verbatim leakage risk. All selection methods use a 40% token budget.

Training Set	Data Selection	Top 0.1%	Top 0.2%	Top 0.5%	Top 1%	Top 2%
OpenThoughts-114K (50K)	Random (40%)	98.35	92.40	46.77	6.79	1.21
	Longest (40%)	98.24	89.86	46.08	8.79	1.14
	CF-Mem (40%) (Ours)	**94.24**	**76.56**	**25.83**	**4.60**	**0.99**
Mixture-of-Thoughts (50K)	Random (40%)	87.28	31.84	5.64	0.80	0.08
	Longest (40%)	96.61	78.61	13.06	2.30	0.17
	CF-Mem (40%) (Ours)	**60.68**	**23.00**	**2.90**	**0.48**	**0.06**

For VB-Mem auditing, we follow the probabilistic extraction view in Definition 2. For each training sample, we provide the first 50 tokens as the prefix and measure the probability of exactly reproducing the following 50 tokens. Rather than estimating this probability through repeated sampling, we compute it directly from the autoregressive token probabilities. A larger probability indicates stronger VB-Mem. See Appendix B.3 for details.

6.2 Experimental Results

CF-Mem and VB-Mem are negatively correlated in real LLMs. To directly examine the theoretical relationship between CF-Mem and VB-Mem, we train each target model on the full 50K training set and compute VB-Mem for every training sample. We then compare these scores with CF-Mem estimated independently using the Qwen2.5-0.5B proxy model. As shown in Figure 1, CF-Mem and VB-Mem exhibit negative Spearman correlations across different target models and training datasets. This empirical result supports our theoretical finding that the two memorization notions capture different and opposing properties of training samples.

CF-Mem selection preserves model utility and outperforms the length-based baseline. Under the same 40% token budget, CF-Mem achieves comparable overall performance to Random on both training datasets. On OpenThoughts-114K, CF-Mem slightly improves the average score from 58.00 to 58.53, while on Mixture-of-Thoughts it remains close to Random (53.07 vs. 53.21). At the same time, CF-Mem outperforms Longest on both datasets, achieving 58.53 vs. 56.11 on OpenThoughts-114K and 53.07 vs. 52.73 on Mixture-of-Thoughts. These results suggest that CF-Mem-based selection can reduce the training data while preserving model utility, and that its benefit cannot be explained simply by favoring longer reasoning trajectories.

CF-Mem selection reduces verbatim leakage. As shown in Table 2, CF-Mem consistently yields lower generation-probability thresholds than Random across all evaluated VB-Mem percentiles. The reduction is particularly clear among the most memorized samples: at the top 0.1% and 0.5%, the thresholds decrease from 87.28 to 60.68 and from 5.64 to 2.90, respectively. This shows that CF-Mem-guided selection effectively suppresses the high-risk tail of verbatim memorization.

Length-based selection can increase verbatim leakage. Longest produces the highest VB-Mem thresholds across all percentiles, despite being a heuristic for selecting information-rich reasoning trajectories. In contrast, CF-Mem achieves lower leakage while maintaining comparable model utility, suggesting that CF-Mem provides a better selection signal when both utility and verbatim leakage are considered. Similar trends are observed on Llama-3.1-8B (see Appendix 4).

AI USE STATEMENT

In this work, we used generative AI tools for implement methods and manuscript drafting and polishing. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, pp. 233–242. PMLR, 2017.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 02 2013. ISBN 9780199535255.

Blake Brittain. Openai says new york times 'hacked' chatgpt to build copyright lawsuit, February 2024. URL https://www.reuters.com/technology/cybersecurity/openai-says-new-york-times-hacked-chatgpt-build-copyright-lawsuit-2024-02-27/.

Blake Brittain. New york times-led group asks court to sanction openai in us copyright dispute, July 2026. URL https://www.reuters.com/legal/litigation/new-york-times-led-group-asks-court-sanction-openai-us-copyright-dispute-2026-07-09/.

Gavin Brown, Mark Bun, Vitaly Feldman, Adam Smith, and Kunal Talwar. When is memorization of irrelevant training data necessary for high-accuracy learning? In *Proceedings of the 53rd annual ACM SIGACT symposium on theory of computing*, pp. 123–132, 2021.

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.

Xingwu Chen, Lei Zhao, and Difan Zou. How transformers utilize multi-head attention in in-context learning? A case study on sparse linear regression. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Meng Ding, Zeqing Zhang, Di Wang, and Lijie Hu. In-run data shapley for adam optimizer. *arXiv preprint arXiv:2602.00329*, 2026.

Vitaly Feldman. Does learning require memorization? A short tale about a long tail. In *Proceedings of the 52nd annual ACM SIGACT symposium on theory of computing*, pp. 954–959, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 2881–2891. Curran Associates, Inc., 2020.

Spencer Frei and Gal Vardi. Trained transformer classifiers generalize and exhibit benign overfitting in-context. In *The Thirteenth International Conference on Learning Representations*, 2025.

Shaopeng Fu and Di Wang. Understanding and improving continuous LLM adversarial training via in-context learning theory. In *The Fourteenth International Conference on Learning Representations*, 2026.

Shaopeng Fu, Liang Ding, Jingfeng Zhang, and Di Wang. Short-length adversarial training helps LLMs defend long-length jailbreak attacks: Theoretical and empirical evidence. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jamie Hayes, Marika Swanberg, Harsh Chaudhari, Itay Yona, Ilia Shumailov, Milad Nasr, Christopher A Choquette-Choo, Katherine Lee, and A Feder Cooper. Measuring memorization in language models via probabilistic extraction. In *Proceedings of the 2025 conference of the nations of the Americas chapter of the association for computational linguistics: human language technologies (volume 1: long papers)*, pp. 9266–9291, 2025.

Hoang Anh Just, Myeongseob Ko, and Ruoxi Jia. The signal is in the steps: Local scoring for reasoning data selection. In *Forty-third International Conference on Machine Learning*, 2026.

Soichiro Kumano, Hiroshi Kera, and Toshihiko Yamasaki. Adversarially pretrained transformers may be universally robust in-context learners. In *The Fourteenth International Conference on Learning Representations*, 2026.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8424–8445. Association for Computational Linguistics, May 2022. doi: 10.18653/v1/2022.acl-long.577. URL https://aclanthology.org/2022.acl-long.577/.

Lauren G. Leipold and Owen R. Wolfe. Rules for use of ai-generated evidence in flux, September 2024. URL https://www.reuters.com/legal/legalindustry/rules-use-ai-generated-evidence-flux-2024-09-23/.

Tianle Li, Chenyang Zhang, Xingwu Chen, Yuan Cao, and Difan Zou. On the robustness of transformers against context hijacking for linear classification. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Licong Lin, Yu Bai, and Song Mei. Transformers as decision makers: Provable in-context reinforcement learning via supervised pretraining. In *The Twelfth International Conference on Learning Representations*, 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 *The Twelfth International Conference on Learning Representations*, 2024.

Yue M. Lu, Mary Letey, Jacob A. Zavatone-Veth, Anindita Maiti, and Cengiz Pehlevan. Asymptotic theory of in-context learning by linear attention. *Proceedings of the National Academy of Sciences*, 122(28):e2502599122, 2025. doi: 10.1073/pnas.2502599122.

Arvind V. Mahankali, Tatsunori Hashimoto, and Tengyu Ma. One step of gradient descent is provably the optimal in-context learner with one layer of linear self-attention. In *The Twelfth International Conference on Learning Representations*, 2024.

John Xavier Morris, Chawin Sitawarin, Chuan Guo, Narine Kokhlikyan, G. Edward Suh, Alexander M Rush, Kamalika Chaudhuri, and Saeed Mahloujifar. How much can language models memorize? In *Forty-third International Conference on Machine Learning*, 2026.

Milad Nasr, Javier Rando, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Florian Tramèr, and Katherine Lee. Scalable extraction of training data from aligned, production language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Tom Sander, Bargav Jayaraman, Mark Ibrahim, Kamalika Chaudhuri, and Chuan Guo. Rethinking the role of verbatim memorization in LLM privacy. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Avi Schwarzschild, Zhili Feng, Pratyush Maini, Zachary Lipton, and J Zico Kolter. Rethinking llm memorization through the lens of adversarial compression. *Advances in Neural Information Processing Systems*, 37:56244–56267, 2024.

Zhenmei Shi, Junyi Wei, Zhuoyan Xu, and Yingyu Liang. Why larger language models do in-context learning differently? In *Forty-first International Conference on Machine Learning*, 2024.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari S. Morcos. D4: Improving LLM pretraining via document de-duplication and diversification. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pp. 35151–35174. PMLR, 2023.

Jiachen T. Wang, Prateek Mittal, Dawn Song, and Ruoxi Jia. Data shapley in one training run. In *The Thirteenth International Conference on Learning Representations*, 2025.

Zhijie Wang, Bo Jiang, and Shuai Li. In-context learning on function classes unveiled for transformers. In *Forty-first International Conference on Machine Learning*, 2024.

Jingfeng Wu, Difan Zou, Zixiang Chen, Vladimir Braverman, Quanquan Gu, and Peter Bartlett. How many pretraining tasks are needed for in-context learning of linear regression? In *The Twelfth International Conference on Learning Representations*, 2024.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. In *Forty-first International Conference on Machine Learning*, 2024.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, volume 36, pp. 69798–69818, 2023a.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Jiayuan Ye, Vitaly Feldman, and Kunal Talwar. Cram less to fit more: Training data pruning improves memorization of facts. In *Forty-third International Conference on Machine Learning*, 2026.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual memorization in neural language models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Ruiqi Zhang, Spencer Frei, and Peter L Bartlett. Trained transformers learn linear models in-context. *Journal of Machine Learning Research*, 25(49):1–55, 2024.

Yedi Zhang, Aaditya K Singh, Peter E. Latham, and Andrew M Saxe. Training dynamics of in-context learning in linear attention. In *Forty-second International Conference on Machine Learning*, 2025.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 3: system demonstrations)*, pp. 400–410, 2024.

Mo Zhou, Haoyang Ma, and Rong Ge. Memorizing long-tail data can help generalization through composition. In *The Fourteenth International Conference on Learning Representations*, 2026.

A PROOFS

A.1 TECHNICAL LEMMAS

Lemma A.1 (c.f. Theorem 5.6 in Boucheron et al. (2013); Gaussian concentration inequality)**.** *Let $X = (X_1, \dots, X_n)$ be a vector of n independent standard normal random variables. Let $f : \mathbb{R}^n \to \mathbb{R}$ denote an L-Lipschitz function. Then, for all $t > 0$,*

$$\mathbb{P}\Big(f(X) - \mathbb{E}f(X) \geq t\Big) \leq e^{-t^2/(2L^2)}.$$

Lemma A.2. *Let $v \sim \mathcal{N}(0, I_d)$ be a d-dimensional standard Gaussian random vector. For any symmetric matrix $A \in \mathbb{R}^{d\times d}$, $b \in \mathbb{R}^d$, and $c \in \mathbb{R}$, if $I_d - 2A \succ 0$, then*

$$\mathop{\mathbb{E}}_{v}\Big[e^{v^\top Av + b^\top v + c}\Big] = \frac{\exp\Big(c + \frac{1}{2}b^\top (I_d - 2A)^{-1}b\Big)}{\sqrt{\det(I_d - 2A)}}.$$

Proof. Since A is symmetric, let $A = Q\Lambda Q^\top$, where $\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_d)$ and Q is orthogonal. Define $z := Q^\top v$ and $\tilde{b} := Q^\top b$. Since $v \sim \mathcal{N}(0, I_d)$ and Q is orthogonal, we also have $z \sim \mathcal{N}(0, I_d)$. Moreover,

$$v^\top Av + b^\top v = z^\top \Lambda z + \tilde{b}^\top z = \sum_{i=1}^{d}\Big(\lambda_i z_i^2 + \tilde{b}_i z_i\Big).$$

Therefore, using the independence of the coordinates of z,

$$\mathop{\mathbb{E}}_{v}\Big[e^{v^\top Av + b^\top v + c}\Big] = e^c \prod_{i=1}^{d}\left[\frac{1}{\sqrt{2\pi}}\int_{\mathbb{R}} \exp\Big(\lambda_i z_i^2 + \tilde{b}_i z_i - \frac{z_i^2}{2}\Big)\mathrm{d}z_i\right].$$

Since $I_d - 2A \succ 0$, we have $1 - 2\lambda_i > 0$ for every i. For each coordinate,

$$\lambda_i z_i^2 + \tilde{b}_i z_i - \frac{z_i^2}{2} = -\frac{1-2\lambda_i}{2}z_i^2 + \tilde{b}_i z_i = -\frac{1-2\lambda_i}{2}\Big(z_i - \frac{\tilde{b}_i}{1-2\lambda_i}\Big)^2 + \frac{\tilde{b}_i^2}{2(1-2\lambda_i)}.$$

Hence,

$$\begin{aligned}
&\frac{1}{\sqrt{2\pi}}\int_{\mathbb{R}} \exp\Big(\lambda_i z_i^2 + \tilde{b}_i z_i - \frac{z_i^2}{2}\Big)\mathrm{d}z_i \\
&= \exp\Big(\frac{\tilde{b}_i^2}{2(1-2\lambda_i)}\Big)\int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{1-2\lambda_i}{2}\Big(z_i - \frac{\tilde{b}_i}{1-2\lambda_i}\Big)^2\Big)\mathrm{d}z_i \\
&= \frac{1}{\sqrt{1-2\lambda_i}} \exp\Big(\frac{\tilde{b}_i^2}{2(1-2\lambda_i)}\Big).
\end{aligned}$$

Substituting this back yields

$$\begin{aligned}
\mathop{\mathbb{E}}_{v}\Big[e^{v^\top Av + b^\top v + c}\Big] &= e^c \prod_{i=1}^{d} \frac{1}{\sqrt{1-2\lambda_i}} \exp\Big(\frac{\tilde{b}_i^2}{2(1-2\lambda_i)}\Big) \\
&= \frac{1}{\sqrt{\det(I_d - 2\Lambda)}} \exp\Big(c + \frac{1}{2}\tilde{b}^\top (I_d - 2\Lambda)^{-1}\tilde{b}\Big) \\
&= \frac{1}{\sqrt{\det(Q)\cdot\det(I_d - 2\Lambda)\cdot\det(Q^\top)}} \exp\Big(c + \frac{1}{2}b^\top Q(I_d - 2\Lambda)^{-1}Q^\top b\Big) \\
&= \frac{\exp\Big(c + \frac{1}{2}b^\top (I_d - 2A)^{-1}b\Big)}{\sqrt{\det(I_d - 2A)}}.
\end{aligned}$$

The proof is completed. □

Lemma A.3 (Bilinear Gaussian exponential moment)**.** *Let $u, v \sim \mathcal{N}(0, I_d)$ be independent standard Gaussian vectors. For any $A \in \mathbb{R}^{d\times d}$, $b, c \in \mathbb{R}^d$, and $s \in \mathbb{R}$ satisfying $|s|\|A\|_2 < 1$, define*

$$Z := u^\top Av + b^\top u + c^\top v.$$

Then,

$$\mathop{\mathbb{E}}_{u,v}[e^{sZ}] = \frac{\exp\left(\frac{s^2}{2}\|c\|_2^2 + \frac{1}{2}(s^2Ac + sb)^\top (I_d - s^2AA^\top)^{-1}(s^2Ac + sb)\right)}{\sqrt{\det(I_d - s^2AA^\top)}}. \tag{7}$$

Moreover,

$$\log \mathop{\mathbb{E}}_{u,v}[e^{sZ}] \le \frac{s^2\big(\|A\|_F^2 + \|b\|_2^2 + \|c\|_2^2\big)}{2(1 - |s|\|A\|_2)}. \tag{8}$$

Proof. Conditioning on u and integrating over v gives

$$\mathop{\mathbb{E}}_{v}[e^{sZ} \mid u] = \exp\left(sb^\top u + \frac{s^2}{2}\|A^\top u + c\|_2^2\right) = \exp\left(\frac{s^2}{2}u^\top AA^\top u + (s^2Ac + sb)^\top u + \frac{s^2}{2}\|c\|_2^2\right).$$

Applying Lemma A.2 to u yields Eq. (7).

For the upper bound, since $|s|\|A\|_2 < 1$,

$$\|(I_d - s^2AA^\top)^{-1}\|_2 \le \frac{1}{1 - s^2\|A\|_2^2}.$$

Hence,

$$\frac{s^2}{2}\|c\|_2^2 + \frac{1}{2}(s^2Ac + sb)^\top (I_d - s^2AA^\top)^{-1}(s^2Ac + sb) \le \frac{s^2(\|b\|_2^2 + \|c\|_2^2)}{2(1 - |s|\|A\|_2)}.$$

Moreover,

$$\frac{1}{\sqrt{\det(I_d - s^2AA^\top)}} \le \exp\left(\frac{s^2\|A\|_F^2}{2(1 - |s|\|A\|_2)}\right).$$

Combining the two bounds proves Eq. (8). □

A.2 Proof of Lemma 4.1

Proof of Lemma 4.1. Let $\xi'_{\tau,i} := y_{\tau,i}\xi_{\tau,i}$ and $\xi'_{\tau,q} := y_{\tau,q}\xi_{\tau,q}$. Since $y_{\tau,i}, y_{\tau,q} \sim U(\{\pm 1\})$ and $\xi_{\tau,i}, \xi_{\tau,q} \sim \mathcal{N}(0, \Lambda_\xi)$, we thus have that $\xi'_{\tau,i}, \xi'_{q,i} \sim \mathcal{N}(0, \Lambda_\xi)$. With these notations and by applying Eq. (2), each margin loss $(-y_{\tau,q} \cdot h_\theta(E_\tau))$ from the objective function defined in Eq. (4) can then be re-written as follows,

$$\begin{aligned}
(-y_{\tau,q} \cdot h_\theta(E_\tau)) &= -y_{\tau,q} \cdot \left(w_{22}^V \cdot \frac{1}{N} Y_\tau X_\tau^\top \cdot W_{11}^{KQ} \cdot x_{\tau,q}\right) \\
&= -y_{\tau,q} \cdot \frac{1}{N}\sum_{i=1}^N y_{\tau,i} \cdot (y_{\tau,i}w_\tau + \xi_{\tau,i})^\top \cdot (w_{22}^V W_{11}^{KQ}) \cdot (y_{\tau,q}w_\tau + \xi_{\tau,q}). \\
&= -\frac{1}{N}\sum_{i=1}^N (w_\tau + y_{\tau,i}\xi_{\tau,i})^\top \cdot (w_{22}^V W_{11}^{KQ}) \cdot (w_\tau + y_{\tau,q}\xi_{\tau,q}). \\
&= -\frac{1}{N}\sum_{i=1}^N (w_\tau + \xi'_{\tau,i})^\top \cdot (w_{22}^V W_{11}^{KQ}) \cdot (w_\tau + \xi'_{\tau,q}).
\end{aligned}$$

Therefore, the gradient of each margin loss is given by

$$\partial_{w_{22}^V W_{11}^{KQ}}(-y_{\tau,q} \cdot h_\theta(E_\tau)) = -\frac{1}{N}\sum_{i=1}^N (w_\tau + \xi'_{\tau,i}) \cdot (w_\tau + \xi'_{\tau,q})^\top.$$

Combining these gradients with Eq. (4) gives

$$\partial_{w_{22}^V W_{11}^{KQ}} \mathcal{L}(\theta, \mathcal{S}) = \frac{1}{B} \sum_{\tau=1}^{B} \partial_{w_{22}^V W_{11}^{KQ}} (-y_{\tau,q} \cdot h_\theta(E_\tau)) + \frac{\lambda}{2} \cdot \partial_{w_{22}^V W_{11}^{KQ}} (\|w_{22}^V W_{11}^{KQ}\|_F^2)$$
$$= -\frac{1}{BN} \sum_{\tau=1}^{B} \sum_{i=1}^{N} (w_\tau + \xi'_{\tau,i})(w_\tau + \xi'_{\tau,q})^\top + \lambda \cdot w_{22}^V W_{11}^{KQ}.$$

Setting this gradient to zero yields

$$(w_{22}^V W_{11}^{KQ})_* = \frac{1}{\lambda BN} \sum_{\tau=1}^{B} \sum_{i=1}^{N} (w_\tau + \xi'_{\tau,i})(w_\tau + \xi'_{\tau,q})^\top.$$

The proof is completed. □

A.3 Proof of Theorem 4.2

Proof of Theorem 4.2. When the performance measure is $M(\theta, w_\tau) := \mathbb{E}_{E_\tau | w_\tau}[y_{\tau,q} h_\theta(E_\tau)]$, according to Definition 1, the CF-Mem for the task sample $w_{\tau_0} \in \mathcal{S}$ is written as follows,

$$\begin{aligned}
&\mathrm{Mem}_{\mathrm{CF}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) \\
&:= \mathop{\mathbb{E}}_{T \subset \mathcal{D},\, z \in T} \mathop{\mathbb{E}}_{\theta \sim \mathcal{A}(T)} \mathop{\mathbb{E}}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})] - \mathop{\mathbb{E}}_{T' \subset \mathcal{D},\, z \notin T'} \mathop{\mathbb{E}}_{\theta \sim \mathcal{A}(T')} \mathop{\mathbb{E}}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})] \\
&= \mathop{\mathbb{E}}_{T \subset \mathcal{D} \setminus \{w_{\tau_0}\}} \Big[\mathop{\mathbb{E}}_{\theta \sim \mathcal{A}(T \cup \{w_{\tau_0}\})} \mathop{\mathbb{E}}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})] - \mathop{\mathbb{E}}_{\theta \sim \mathcal{A}(T)} \mathop{\mathbb{E}}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})] \Big]. \qquad (9)
\end{aligned}$$

Before calculating the closed-form solution of Eq. (9), we first calculate the term $\mathbb{E}_{\theta \sim \mathcal{A}(T)} \mathbb{E}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})]$ for any nonempty training subset $T \subset \mathcal{D}$. For $T = \varnothing$, this expectation is zero by $\mathcal{A}(\varnothing) = 0_{d \times d}$. Specifically, for the inner expectation calculation, we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})] \\
&= \mathop{\mathbb{E}}_{y_{\tau_0,i}, y_{\tau_0,q}, \xi_{\tau_0,i}, \xi_{\tau_0,q}} \Big[y_{\tau_0,q} \cdot \frac{1}{N} Y_{\tau_0} X_{\tau_0}^\top \cdot w_{22}^V W_{11}^{KQ} \cdot x_{\tau_0,q} \Big] \\
&= \frac{1}{N} \sum_{i=1}^{N} \mathop{\mathbb{E}}_{\xi'_{\tau_0,i}, \xi'_{\tau_0,q}} \Big[(w_{\tau_0} + \xi'_{\tau_0,i})^\top \cdot (w_{22}^V W_{11}^{KQ}) \cdot (w_{\tau_0} + \xi'_{\tau_0,q}) \Big] \\
&= w_{\tau_0}^\top \cdot (w_{22}^V W_{11}^{KQ}) \cdot w_{\tau_0},
\end{aligned}$$

where the second equality follows because the noises $\xi'_{\tau,i}, \xi'_{\tau,q} \sim \mathcal{N}(0, \Lambda_\xi)$ are independent, zero-mean, and independent of the model parameter. As a result, by further applying Lemma 4.1, the overall expectation term $\mathbb{E}_{\theta \sim \mathcal{A}(T)} \mathbb{E}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})]$ can be further written as

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\theta \sim \mathcal{A}(T)} \mathop{\mathbb{E}}_{E_{\tau_0} | w_{\tau_0}} [y_{\tau_0,q} h_\theta(E_{\tau_0})] \\
&= w_{\tau_0}^\top \cdot \mathop{\mathbb{E}}_{(w_{22}^V W_{11}^{KQ})_* \sim \mathcal{A}(T)} [(w_{22}^V W_{11}^{KQ})_*] \cdot w_{\tau_0}, \\
&= \frac{1}{\lambda \cdot |T| \cdot N} \sum_{w_\tau \in T} \sum_{i=1}^{N} w_{\tau_0}^\top \cdot \mathop{\mathbb{E}}_{\xi'_{\tau,i}, \xi'_{\tau,q} \sim \mathcal{N}(0, \Lambda_\xi)} \Big[(w_\tau + \xi'_{\tau,i})(w_\tau + \xi'_{\tau,q})^\top \Big] \cdot w_{\tau_0} \\
&= \frac{1}{\lambda \cdot |T| \cdot N} \sum_{w_\tau \in T} \sum_{i=1}^{N} w_{\tau_0}^\top \cdot w_\tau w_\tau^\top \cdot w_{\tau_0} \\
&= \frac{1}{\lambda \cdot |T|} \sum_{w_\tau \in T} (w_\tau^\top w_{\tau_0})^2. \qquad (10)
\end{aligned}$$

We now insert Eq. (10) into Eq. (9). When $T = \varnothing$, we define the excluded-subset performance to be zero. Expressions involving division by $|T|$ are only used when $T \neq \varnothing$.

$$
\begin{aligned}
&\mathrm{Mem}_{\mathrm{CF}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) \\
&= \mathop{\mathbb{E}}_{T \subset \mathcal{D}\setminus\{w_{\tau_0}\}} \Big[\frac{1}{\lambda \cdot (|T|+1)} \sum_{w_\tau \in T \cup \{w_{\tau_0}\}} (w_\tau^\top w_{\tau_0})^2 - \frac{1}{\lambda \cdot |T|} \sum_{w_\tau \in T} (w_\tau^\top w_{\tau_0})^2 \Big] \\
&= \frac{1}{\lambda} \cdot \mathop{\mathbb{E}}_{T \subset \mathcal{D}\setminus\{w_{\tau_0}\}} \Big[\frac{1}{|T|+1} \cdot (w_{\tau_0}^\top w_{\tau_0})^2 - \frac{1}{|T| \cdot (|T|+1)} \sum_{w_\tau \in T} (w_\tau^\top w_{\tau_0})^2 \Big] \\
&= \frac{1}{\lambda \cdot 2^{B-1}} \cdot \Big[\sum_{i=0}^{B-1} \frac{\binom{B-1}{i}}{(i+1)} \cdot (w_{\tau_0}^\top w_{\tau_0})^2 - \sum_{w_\tau \in \mathcal{D}\setminus\{w_{\tau_0}\}} \sum_{i=0}^{B-2} \frac{\binom{B-2}{i}}{(i+1)\cdot(i+2)} (w_\tau^\top w_{\tau_0})^2 \Big] \\
&= \frac{1}{\lambda \cdot 2^{B-1}} \cdot \Big[\sum_{i=0}^{B-1} \binom{B}{i+1} \cdot \frac{1}{B} \cdot (w_{\tau_0}^\top w_{\tau_0})^2 - \sum_{w_\tau \in \mathcal{D}\setminus\{w_{\tau_0}\}} \sum_{i=0}^{B-2} \binom{B}{i+2} \cdot \frac{1}{B\cdot(B-1)} \cdot (w_\tau^\top w_{\tau_0})^2 \Big] \\
&\overset{*}{=} \frac{1}{\lambda \cdot B \cdot 2^{B-1}} \cdot \Big[\sum_{i=1}^{B} \binom{B}{i} \cdot 1 - \sum_{w_\tau \in \mathcal{D}\setminus\{w_{\tau_0}\}} \sum_{i=2}^{B} \binom{B}{i} \cdot \frac{1}{(B-1)} \cdot (w_\tau^\top w_{\tau_0})^2 \Big] \\
&= \frac{1}{\lambda \cdot B \cdot 2^{B-1}} \cdot \Big[2^B - 1 - \sum_{\tau \neq \tau_0} \frac{2^B - B - 1}{(B-1)} \cdot (w_\tau^\top w_{\tau_0})^2 \Big] \\
&= \frac{1}{\lambda \cdot 2^{B-1}} + \frac{2^B - B - 1}{\lambda \cdot B \cdot 2^{B-1}} \cdot \Big(1 - \frac{1}{(B-1)} \sum_{\tau \neq \tau_0} (w_\tau^\top w_{\tau_0})^2 \Big),
\end{aligned}
$$

where $(*)$ is due to the fact that w_τ is drawn uniformly from a ball-sphere $w_\tau \sim U(\mathbb{S}^{d-1})$ and thus $\|w_\tau\|_2 = 1$.

The proof is completed. □

A.4 Proof of Theorem 4.3

Proof of Theorem 4.3. By applying Eq. (2), the VB-Mem score in Definition 3 can be re-written as

$$
\mathrm{Mem}_{\mathrm{VB}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) = \frac{1}{N} \sum_{i=1}^{N} \mathop{\mathbb{E}}_{\theta_* \sim \mathcal{A}(\mathcal{S})} \left[(w_{\tau_0} + \xi'_{\tau_0,i})^\top \theta_* (w_{\tau_0} + \xi'_{\tau_0,q}) \right], \tag{11}
$$

where $\xi'_{\tau,i}, \xi'_{\tau,q} \sim \mathcal{N}(0, \Lambda_\xi)$ are the signed noises in the training prompt E_τ^{tr}. The expectation is over the randomness of the training algorithm, including the sampled training prompts.

By Lemma 4.1,

$$
\theta_* = \frac{1}{\lambda B N} \sum_{\tau=1}^{B} \sum_{j=1}^{N} (w_\tau + \xi'_{\tau,j})(w_\tau + \xi'_{\tau,q})^\top.
$$

Substituting this into Eq. (11) gives

$$
\begin{aligned}
&\mathrm{Mem}_{\mathrm{VB}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) \\
&= \frac{1}{\lambda B N^2} \sum_{\tau=1}^{B} \sum_{i=1}^{N} \sum_{j=1}^{N} \mathbb{E} \left[(w_{\tau_0} + \xi'_{\tau_0,i})^\top (w_\tau + \xi'_{\tau,j})(w_\tau + \xi'_{\tau,q})^\top (w_{\tau_0} + \xi'_{\tau_0,q}) \right].
\end{aligned}
$$

Since the in-context sample noises are independent of the query noises, the expectation factorizes. Using their zero means and independence,

$$
\begin{aligned}
\mathbb{E} \left[(w_{\tau_0} + \xi'_{\tau_0,i})^\top (w_\tau + \xi'_{\tau,j}) \right] &= w_{\tau_0}^\top w_\tau + \mathbb{1}[\tau = \tau_0,\ i = j] \, \mathrm{tr}(\Lambda_\xi), \\
\mathbb{E} \left[(w_\tau + \xi'_{\tau,q})^\top (w_{\tau_0} + \xi'_{\tau_0,q}) \right] &= w_\tau^\top w_{\tau_0} + \mathbb{1}[\tau = \tau_0] \, \mathrm{tr}(\Lambda_\xi).
\end{aligned}
$$

Thus, for $\tau \neq \tau_0$, each term contributes $(w_\tau^\top w_{\tau_0})^2$. For $\tau = \tau_0$, using $\|w_{\tau_0}\| = 1$, summing over i and j gives

$$\left(N^2 + N\,\mathrm{tr}(\Lambda_\xi)\right)\left(1 + \mathrm{tr}(\Lambda_\xi)\right).$$

Therefore,

$$\begin{aligned}\mathrm{Mem}_{\mathrm{VB}}(w_{\tau_0}, \mathcal{S}, \mathcal{A}) &= \frac{1}{\lambda B}\Big[\sum_{\tau \neq \tau_0}(w_\tau^\top w_{\tau_0})^2 + \Big(1 + \frac{\mathrm{tr}(\Lambda_\xi)}{N}\Big)(1 + \mathrm{tr}(\Lambda_\xi))\Big] \\ &= \frac{1}{\lambda B}\Big[c_\xi + \sum_{\tau \neq \tau_0}(w_\tau^\top w_{\tau_0})^2\Big].\end{aligned}$$

This completes the proof. □

A.5 Proof of Theorem 4.5

Proof of Theorem 4.5. For the ICL generalization risk $\mathcal{R}(\theta)$ defined in Eq. (3), we can bound it as

$$\begin{aligned}\mathcal{R}(\theta) &= \mathop{\mathbb{E}}_{\tilde{w}_\tau}\mathop{\mathbb{E}}_{\tilde{E}_\tau|\tilde{w}_\tau}\mathbb{1}[y_{\tau,q}h_\theta(\tilde{E}_\tau) \leq 0] \\ &\overset{(*)}{\leq} \mathop{\mathbb{E}}_{\tilde{w}_\tau}\mathop{\mathbb{E}}_{\tilde{E}_\tau|\tilde{w}_\tau}\exp\Big(-\eta \cdot y_{\tau,q}h_\theta(\tilde{E}_\tau)\Big) \\ &\overset{(**)}{=} \mathop{\mathbb{E}}_{\tilde{w}_\tau,\tilde{\xi}'_{\tau,i},\tilde{\xi}'_{\tau,q}}\exp\Big(-\eta \cdot \Big(\tilde{w}_\tau + \frac{1}{N}\sum_{i=1}^N \tilde{\xi}'_{\tau,i}\Big)^\top \cdot (w_{22}^V W_{11}^{KQ}) \cdot (\tilde{w}_\tau + \tilde{\xi}'_{\tau,q})\Big),\end{aligned} \tag{12}$$

where $\eta > 0$ is an arbitrary positive real, random task weight follows $w_\tau \sim U(\mathbb{S}^{d-1})$, data noises follow $\tilde{\xi}'_{\tau,i}, \tilde{\xi}'_{\tau,q} \sim \mathcal{N}(0, \Lambda_\xi)$, $(*)$ is due to the fact that $\mathbb{1}[z \leq 0] \leq e^{-\eta z}$, and $(**)$ is by Eq. (2). In the remaining of the proof, we will first upper bound the generalization risk for arbitrary model parameter θ step by step, and then analyze it on the optimal model parameter $\theta_* \sim \mathcal{A}(\mathcal{S})$ obtained from the ICL training algorithm $\mathcal{A}$ and the ICL trainset $\mathcal{S}$.

Step 1: Decomposing evaluation-noise term. For the Gaussian random terms $\frac{1}{N}\sum_{i=1}^N \tilde{\xi}'_{\tau,i}$ and $\tilde{\xi}'_{\tau,q}$ in Eq. (12), we can re-written them as

$$\frac{1}{N}\sum_{i=1}^N \tilde{\xi}'_{\tau,i} = \frac{1}{\sqrt{N}}\Lambda_\xi^{\frac{1}{2}}u, \qquad \tilde{\xi}'_{\tau,q} = \Lambda_\xi^{\frac{1}{2}}v,$$

where $u, v \sim \mathcal{N}(0, I_d)$. Inserting them back into Eq. (12), we can therefore simplify the ICL generalization bound as follows,

$$\mathcal{R}(\theta) \leq \mathop{\mathbb{E}}_{\tilde{w}_\tau}\Big[e^{-\eta\tilde{w}_\tau^\top(w_{22}^V W_{11}^{KQ})\tilde{w}_\tau} \cdot \mathop{\mathbb{E}}_{u,v}[e^{-\eta V_\theta(\tilde{w}_\tau,u,v)}]\Big], \tag{13}$$

where

$$\begin{aligned}V_\theta(\tilde{w}_\tau, u, v) := \frac{1}{\sqrt{N}}u^\top\Lambda_\xi^{\frac{1}{2}} \cdot (w_{22}^V W_{11}^{KQ}) \cdot \Lambda_\xi^{\frac{1}{2}}v + \frac{1}{\sqrt{N}}u^\top\Lambda_\xi^{\frac{1}{2}} \cdot (w_{22}^V W_{11}^{KQ}) \cdot \tilde{w}_\tau \\ + \tilde{w}_\tau^\top \cdot (w_{22}^V W_{11}^{KQ}) \cdot \Lambda_\xi^{\frac{1}{2}}v\end{aligned}$$

is the evaluation-noise term and $u, v \sim \mathcal{N}(0, I_d)$ are standard Gaussian noises.

Step 2: Bounding the evaluation-noise term. We next bound the evaluation-noise term $\mathbb{E}_{u,v}[e^{-\eta V_\theta(\tilde{w}_\tau,u,v)}]$ in Eq. (13). Let $\theta := w_{22}^V W_{11}^{KQ}$. Since $u, v \sim \mathcal{N}(0, I_d)$ are independent,

$$V_\theta(\tilde{w}_\tau, u, v) = u^\top Av + b^\top u + c^\top v,$$

where

$$A := \frac{1}{\sqrt{N}}\Lambda_\xi^{1/2}\theta\Lambda_\xi^{1/2}, \qquad b := \frac{1}{\sqrt{N}}\Lambda_\xi^{1/2}\theta\tilde{w}_\tau, \qquad c := \Lambda_\xi^{1/2}\theta^\top\tilde{w}_\tau.$$

By Lemma A.3 with $s = -\eta$, whenever $\eta\|A\|_2 < 1$,

$$\mathop{\mathbb{E}}_{u,v}\Big[e^{-\eta V_\theta(\tilde{w}_\tau,u,v)}\Big] \le \exp\left(\frac{\eta^2\big(\|A\|_F^2 + \|b\|_2^2 + \|c\|_2^2\big)}{2(1-\eta\|A\|_2)}\right). \tag{14}$$

Using $\|\tilde{w}_\tau\|_2 = 1$, we have

$$\begin{aligned}
\|A\|_F^2 &\le \frac{\|\Lambda_\xi\|_2^2}{N}\|\theta\|_F^2,\\
\|b\|_2^2 &\le \frac{\|\Lambda_\xi\|_2}{N}\|\theta\|_F^2,\\
\|c\|_2^2 &\le \|\Lambda_\xi\|_2\|\theta\|_F^2,\\
\|A\|_2 &\le \frac{\|\Lambda_\xi\|_2}{\sqrt{N}}\|\theta\|_F.
\end{aligned}$$

Therefore,

$$\mathop{\mathbb{E}}_{u,v}\Big[e^{-\eta V_\theta(\tilde{w}_\tau,u,v)}\Big] \le \exp\left(\frac{\eta^2\big(\|\Lambda_\xi\|_2^2/N + \|\Lambda_\xi\|_2/N + \|\Lambda_\xi\|_2\big)\|\theta\|_F^2}{2\big(1-\eta\|\Lambda_\xi\|_2\|\theta\|_F/\sqrt{N}\big)}\right), \tag{15}$$

where a sufficient condition is $\eta < \frac{\sqrt{N}}{\|\Lambda_\xi\|_2\|\theta\|_F}$.

Step 3: High-probability control of the parameter norm. By combining Eqs. (13) and (15), we know that if we want to bound the generalization risk $\mathcal{R}(\theta_*)$ for trained model parameter $\theta_* \sim \mathcal{A}(\mathcal{S})$, we just need to bound the parameter norm $\|\theta_*\|_F$ with high probability. To this end, we first re-write the closed-form trained model parameter in Lemma 4.1 as follows,

$$(w_{11}^v W_{22}^{KQ})_* = \frac{1}{\lambda B}\sum_{\tau=1}^{B}\Big(w_\tau + \frac{1}{\sqrt{N}}\Lambda_\xi^{\frac{1}{2}}u_\tau\Big)(w_\tau + \Lambda_\xi^{\frac{1}{2}}v_\tau)^\top,$$

where $u_\tau, v_\tau \sim \mathcal{N}(0, I_d)$ are standard Gaussian noises and each $w_\tau \in \mathcal{S}$ is from the ICL training set $\mathcal{S}$. As a result,

$$\begin{aligned}
\|(w_{11}^v W_{22}^{KQ})_*\|_F &\le \frac{1}{\lambda B}\sum_{\tau=1}^{B}\|w_\tau + \frac{1}{\sqrt{N}}\Lambda_\xi^{\frac{1}{2}}u_\tau\|_2 \cdot \|w_\tau + \Lambda_\xi^{\frac{1}{2}}v_\tau\|_2\\
&\le \frac{1}{\lambda B}\sum_{\tau=1}^{B}(\|w_\tau\|_2 + \frac{1}{\sqrt{N}}\|\Lambda_\xi^{\frac{1}{2}}\|_2 \cdot \|u_\tau\|_2) \cdot (\|w_\tau\|_2 + \|\Lambda_\xi^{\frac{1}{2}}\|_2 \cdot \|v_\tau\|_2)\\
&= \frac{1}{\lambda B}\sum_{\tau=1}^{B}(1 + \frac{1}{\sqrt{N}}\|\Lambda_\xi\|_2^{\frac{1}{2}} \cdot \|u_\tau\|_2) \cdot (1 + \|\Lambda_\xi\|_2^{\frac{1}{2}} \cdot \|v_\tau\|_2).
\end{aligned} \tag{16}$$

We next control $\|u_\tau\|_2$ and $\|v_\tau\|_2$. For a single standard Gaussian variable norm $\|v_\tau\|_2$, since $\|\cdot\|_2$ is a 1-Lipschitz function, by applying Lemma A.1, we have for any $t > 0$,

$$\mathbb{P}(\|v_\tau\|_2 - \mathbb{E}[\|v_\tau\|_2] \ge t) \le e^{-t^2/2}.$$

For the expectation $\mathbb{E}[\|v_\tau\|_2]$, by applying Jensen's Inequality,

$$\mathbb{E}[\|v_\tau\|_2] = \mathbb{E}\Big[\sqrt{\|v_\tau\|_2^2}\Big] \le \sqrt{\mathbb{E}[\|v_\tau\|_2^2]} = \sqrt{d}.$$

As a result, by setting $t_\delta := \sqrt{2\log\frac{4B}{\delta}}$ for any $\delta \in (0,1)$, we have

$$\mathbb{P}\big(\|u_\tau\|_2 - \sqrt{d} \ge t_\delta\big) \le \mathbb{P}\big(\|u_\tau\|_2 - \mathbb{E}\|u_\tau\|_2 \ge t_\delta\big) \le e^{-t_\delta^2/2} = \frac{\delta}{4B}.$$

Applying the same argument to all $\{u_\tau, v_\tau\}_{\tau=1}^B$ and taking a union bound, with probability at least $1 - \frac{\delta}{2}$ we have that

$$\|u_\tau\|_2,\ \|v_\tau\|_2 \le \sqrt{d} + t_\delta, \qquad \forall \tau \in [B].$$

Combining this with Eq. (16), we obtain

$$\|\theta_*\|_F \le \frac{1}{\lambda B}\sum_{\tau=1}^{B}\Big(1+\frac{\|\Lambda_\xi\|_2^{1/2}}{\sqrt{N}}(\sqrt{d}+t_\delta)\Big)\Big(1+\|\Lambda_\xi\|_2^{1/2}(\sqrt{d}+t_\delta)\Big)$$
$$= \frac{1}{\lambda}\Big(1+\frac{\|\Lambda_\xi\|_2^{1/2}}{\sqrt{N}}(\sqrt{d}+t_\delta)\Big)\Big(1+\|\Lambda_\xi\|_2^{1/2}(\sqrt{d}+t_\delta)\Big) =: L_\delta, \tag{17}$$

where $t_\delta := \sqrt{2\log\frac{4B}{\delta}}$.

As a result, with probability at least $1-\frac{\delta}{2}$, for any $\eta < \frac{\sqrt{N}}{\|\Lambda_\xi\|_2 L_\delta}$, Eq. (15) gives

$$\mathop{\mathbb{E}}_{u,v}\Big[e^{-\eta V_{\theta_*}(w_\tau,u,v)}\Big] \le \exp\Big(\frac{\eta^2 L_\delta^2\big(\|\Lambda_\xi\|_2^2/N + \|\Lambda_\xi\|_2/N + \|\Lambda_\xi\|_2\big)}{1-\eta L_\delta\|\Lambda_\xi\|_2/\sqrt{N}}\Big). \tag{18}$$

Step 4: Bounding the clean-margin deviation. We next control the clean-margin term $\tilde{w}_\tau^\top\theta_*\tilde{w}_\tau$ in Eq. (13), where $\tilde{w}_\tau \sim U(\mathbb{S}^{d-1})$ is independent of the training set $\mathcal{S}$. By Lemma 4.1,

$$\mathop{\mathbb{E}}_{\theta_*\sim\mathcal{A}(\mathcal{S})}[\theta_*] = \frac{1}{\lambda B}\sum_{\tau'=1}^{B} w_{\tau'}w_{\tau'}^\top, \qquad \mathop{\mathbb{E}}_{\theta_*\sim\mathcal{A}(\mathcal{S})}[\tilde{w}_\tau^\top\theta_*\tilde{w}_\tau] = \frac{1}{\lambda B}\sum_{\tau'=1}^{B}(\tilde{w}_\tau^\top w_{\tau'})^2. \tag{19}$$

Let $\bar{\xi}'_{\tau'} := N^{-1}\sum_{i=1}^{N}\xi'_{\tau',i}$. Then,

$$\tilde{w}_\tau^\top\theta_*\tilde{w}_\tau - \frac{1}{\lambda B}\sum_{\tau'=1}^{B}(\tilde{w}_\tau^\top w_{\tau'})^2 = \frac{1}{\lambda B}\sum_{\tau'=1}^{B} Z_{\tau'}, \tag{20}$$

where

$$Z_{\tau'} := a_{\tau'}(X_{\tau'}+Y_{\tau'}) + X_{\tau'}Y_{\tau'}, \qquad a_{\tau'} := \tilde{w}_\tau^\top w_{\tau'},$$

with $X_{\tau'} := \tilde{w}_\tau^\top\bar{\xi}'_{\tau'}$ and $Y_{\tau'} := \tilde{w}_\tau^\top\xi'_{\tau',q}$.

For a fixed $\tilde{w}_\tau$, let $\sigma_\tau^2 := \tilde{w}_\tau^\top\Lambda_\xi\tilde{w}_\tau$. Then $X_{\tau'} = \frac{\sigma_\tau}{\sqrt{N}}u_{\tau'}$ and $Y_{\tau'} = \sigma_\tau v_{\tau'}$ for independent $u_{\tau'}, v_{\tau'} \sim \mathcal{N}(0,1)$, and hence

$$Z_{\tau'} = \frac{\sigma_\tau^2}{\sqrt{N}}u_{\tau'}v_{\tau'} + \frac{a_{\tau'}\sigma_\tau}{\sqrt{N}}u_{\tau'} + a_{\tau'}\sigma_\tau v_{\tau'}.$$

Applying Lemma A.3 with $A = \sigma_\tau^2/\sqrt{N}$, $b = a_{\tau'}\sigma_\tau/\sqrt{N}$, and $c = a_{\tau'}\sigma_\tau$, we obtain

$$\log\mathbb{E}[e^{sZ_{\tau'}}] \le \frac{s^2}{2(1-|s|\|\Lambda_\xi\|_2/\sqrt{N})}\left[\left(1+\frac{1}{N}\right)\|\Lambda_\xi\|_2 + \frac{\|\Lambda_\xi\|_2^2}{N}\right], \tag{21}$$

for $|s| < \sqrt{N}/\|\Lambda_\xi\|_2$, where we use $|a_{\tau'}| \le 1$ and $\sigma_\tau^2 \le \|\Lambda_\xi\|_2$.

Define

$$D_\tau := \tilde{w}_\tau^\top\theta_*\tilde{w}_\tau - \frac{1}{\lambda B}\sum_{\tau'=1}^{B}(\tilde{w}_\tau^\top w_{\tau'})^2 = \frac{1}{\lambda B}\sum_{\tau'=1}^{B} Z_{\tau'}.$$

Since $\{Z_{\tau'}\}_{\tau'=1}^{B}$ are independent, for any $|s| < \frac{\lambda B\sqrt{N}}{\|\Lambda_\xi\|_2}$,

$$\begin{aligned}
\log \mathop{\mathbb{E}}_{\theta_* \sim \mathcal{A}(\mathcal{S})}[e^{sD_\tau}] &= \log \mathbb{E}\left[\exp\left(\frac{s}{\lambda B}\sum_{\tau'=1}^{B} Z_{\tau'}\right)\right] \\
&= \log \prod_{\tau'=1}^{B} \mathbb{E}\left[\exp\left(\frac{s}{\lambda B} Z_{\tau'}\right)\right] \\
&= \sum_{\tau'=1}^{B} \log \mathbb{E}\left[\exp\left(\frac{s}{\lambda B} Z_{\tau'}\right)\right] \\
&\le \sum_{\tau'=1}^{B} \frac{\frac{s^2}{\lambda^2 B^2}\left[(1+\frac{1}{N})\|\Lambda_\xi\|_2 + \frac{\|\Lambda_\xi\|_2^2}{N}\right]}{2\left(1 - \frac{|s|\|\Lambda_\xi\|_2}{\lambda B\sqrt{N}}\right)} \\
&= \frac{s^2\left[(1+\frac{1}{N})\|\Lambda_\xi\|_2 + \frac{\|\Lambda_\xi\|_2^2}{N}\right]}{2\lambda^2 B\left(1 - \frac{|s|\|\Lambda_\xi\|_2}{\lambda B\sqrt{N}}\right)}.
\end{aligned} \tag{22}$$

Let

$$v := \frac{1}{\lambda^2 B}\left[\left(1+\frac{1}{N}\right)\|\Lambda_\xi\|_2 + \frac{\|\Lambda_\xi\|_2^2}{N}\right], \qquad c := \frac{\|\Lambda_\xi\|_2}{\lambda B\sqrt{N}}.$$

Then Eq. (22) becomes

$$\log \mathbb{E}[e^{sD_\tau}] \le \frac{vs^2}{2(1-c|s|)}.$$

For any $t > 0$, applying the Chernoff bound with $s = \sqrt{2t/v}/(1 + c\sqrt{2t/v})$ gives

$$\Pr_{\mathcal{A}}\left(D_\tau > \sqrt{2vt} + ct\right) \le e^{-t}.$$

Applying the same argument to $-D_\tau$ yields

$$\Pr_{\mathcal{A}}\left(|D_\tau| > \sqrt{2vt} + ct\right) \le 2e^{-t}. \tag{23}$$

Equivalently,

$$\Pr_{\mathcal{A}}\left(|D_\tau| > \frac{1}{\lambda}\left[\sqrt{\frac{2t}{B}\left(\left(1+\frac{1}{N}\right)\|\Lambda_\xi\|_2 + \frac{\|\Lambda_\xi\|_2^2}{N}\right)} + \frac{\|\Lambda_\xi\|_2 t}{B\sqrt{N}}\right]\right) \le 2e^{-t}. \tag{24}$$

We next make the bound uniform over $\tilde{w}_\tau \in \mathbb{S}^{d-1}$. Define

$$\Delta := \frac{1}{2}\left(\theta_* - \mathbb{E}[\theta_*] + \theta_*^\top - \mathbb{E}[\theta_*]^\top\right).$$

Since $\tilde{w}_\tau^\top(\theta_* - \mathbb{E}[\theta_*])\tilde{w}_\tau = \tilde{w}_\tau^\top \Delta \tilde{w}_\tau$, let $\mathcal{N}_{1/4}$ be a $1/4$-net of $\mathbb{S}^{d-1}$ with $|\mathcal{N}_{1/4}| \le 9^d$. Setting $t = d\log 9 + \log\frac{4}{\delta}$ in Eq. (23) and taking a union bound gives, with probability at least $1 - \delta/2$,

$$\max_{w \in \mathcal{N}_{1/4}} |w^\top \Delta w| \le \frac{1}{\lambda}\left[\sqrt{\frac{2(d\log 9 + \log\frac{4}{\delta})}{B}\left(\left(1+\frac{1}{N}\right)\|\Lambda_\xi\|_2 + \frac{\|\Lambda_\xi\|_2^2}{N}\right)} + \frac{\|\Lambda_\xi\|_2}{B\sqrt{N}}\left(d\log 9 + \log\frac{4}{\delta}\right)\right].$$

Using the standard $1/4$-net bound, $\sup_{\|w\|_2=1}|w^\top \Delta w| \le 2\max_{w\in\mathcal{N}_{1/4}}|w^\top \Delta w|$. Therefore, with probability at least $1 - \delta/2$,

$$\tilde{w}_\tau^\top \theta_* \tilde{w}_\tau \ge \frac{1}{\lambda B}\sum_{\tau'=1}^{B}(\tilde{w}_\tau^\top w_{\tau'})^2 - \epsilon_\delta, \qquad \forall \tilde{w}_\tau \in \mathbb{S}^{d-1}, \tag{25}$$

where

$$\epsilon_\delta := \frac{2}{\lambda}\left[\sqrt{\frac{2(d\log 9+\log\frac{4}{\delta})}{B}\left(\left(1+\frac{1}{N}\right)\|\Lambda_\xi\|_2+\frac{\|\Lambda_\xi\|_2^2}{N}\right)}+\frac{\|\Lambda_\xi\|_2}{B\sqrt{N}}\left(d\log 9+\log\frac{4}{\delta}\right)\right]. \tag{26}$$

Combining Eq. (25) with the evaluation-noise bound from Step 3 (*i.e.*, Eq. (18)) and taking a union bound over the two training events, with probability at least $1-\delta$,

$$\mathcal{R}(\theta_*) \le \exp\left(\eta\epsilon_\delta+\frac{\eta^2L_\delta^2\left(\|\Lambda_\xi\|_2^2/N+\|\Lambda_\xi\|_2/N+\|\Lambda_\xi\|_2\right)}{2\left(1-\eta L_\delta\|\Lambda_\xi\|_2/\sqrt{N}\right)}\right)\cdot\mathop{\mathbb{E}}_{\tilde{w}_\tau}\left[\exp\left(-\frac{\eta}{\lambda B}\sum_{\tau'=1}^{B}(\tilde{w}_\tau^\top w_{\tau'})^2\right)\right].$$

The proof is completed. □

B Additional Experimental Details

B.1 Data Selection Details

CF-Mem estimation. We estimate CF-Mem using Qwen2.5-0.5B base LLM as the proxy model. Following the random-subset procedure in Section 5, we train the proxy model on randomly sampled subsets for 50 runs, with each subset containing 10% of the training samples. Sequences are truncated to at most 16K tokens, while over-length samples are not removed. We fine-tune the proxy model with LoRA for one epoch using a batch size of 64. We apply LoRA to all linear layers, with rank 32, alpha 128, and dropout 0.1. We use AdamW with a learning rate of 1×10^{-4}, cosine learning-rate scheduling, and zero weight decay.

The proxy model is publicly available at:

- Qwen2.5-0.5B: https://huggingface.co/Qwen/Qwen2.5-0.5B.

Data selection baselines. We compare CF-Mem-guided selection with two baselines under the same token budget:

- *Random* randomly samples complete training samples until reaching the target token budget.
- *Longest* ranks samples by token length and applies the same binning and cosine selection procedure as CF-Mem-guided selection (see Section 5). This baseline tests whether the gains mainly come from favoring longer reasoning trajectories.

B.2 Model Training Details

Training datasets. We use OpenThoughts-114K and Mixture-of-Thoughts as the training datasets. For each dataset, we use only the first 50K samples for training and data selection.

The datasets are publicly available at:

- OpenThoughts-114K:https://huggingface.co/datasets/open-thoughts/OpenThoughts-114k.
- Mixture-of-Thoughts: https://huggingface.co/datasets/open-r1/Mixture-of-Thoughts.

Models and training. We perform full-parameter fine-tuning on the base versions of Qwen2.5-7B and Llama-3.1-8B using LLaMA-Factory (Zheng et al., 2024). We train models using AdamW with a learning rate of 1×10^{-4}, cosine learning-rate scheduling, batch size 128, and a maximum sequence length of 32K tokens. Samples exceeding the maximum sequence length are dropped.

The base models are publicly available at:

- Qwen2.5-7B: https://huggingface.co/Qwen/Qwen2.5-7B.
- Llama-3.1-8B: https://huggingface.co/meta-llama/Llama-3.1-8B.

B.3 Evaluation Details

Utility evaluation. We evaluate model utility on five mathematical reasoning benchmarks: GSM8K, MATH-500, AMC23, AIME25, and AIME26. Inference is performed with SGLang using a maximum generation length of 32K tokens, temperature 0.6, and top-p 0.95. For each problem, we generate eight responses and report Avg@8 on GSM8K, MATH-500, and AMC23, and Pass@8 on AIME25 and AIME26.

The evaluation datasets are publicly available at:

- GSM8K: https://huggingface.co/datasets/openai/gsm8k.
- MATH-500: https://huggingface.co/datasets/HuggingFaceH4/MATH-500.
- AMC23: https://huggingface.co/datasets/math-ai/amc23.
- AIME25: https://huggingface.co/datasets/MathArena/aime_2025.
- AIME26: https://huggingface.co/datasets/MathArena/aime_2026.

VB-Mem auditing. Our VB-Mem evaluation follows the probabilistic extraction interpretation in Definition 2. For each training sample, we use its first 50 tokens as the prefix and the subsequent 50 tokens as the target suffix. Let the target suffix be $(y_1, \ldots, y_{50})$ and the prefix be x. We compute its one-generation exact-reproduction probability as

$$p_{\mathrm{VB}} := \prod_{j=1}^{50} p_\theta(y_j \mid x, y_{<j}),$$

or equivalently its log-probability

$$\log p_{\mathrm{VB}} = \sum_{j=1}^{50} \log p_\theta(y_j \mid x, y_{<j}).$$

We use $\log p_{\mathrm{VB}}$ as the VB-Mem score, where a larger value indicates that the target suffix is more likely to be reproduced exactly.

Unlike Monte Carlo estimation through repeated generations, this score directly computes the exact one-generation extraction probability under the decoding distribution. For n independent generations, the probability that the target suffix is reproduced at least once is therefore

$$1 - (1 - p_{\mathrm{VB}})^n,$$

which directly connects our score to (n, p)-discoverable extraction. We compute token probabilities using temperature 1.0 and top-k 40.

C Additional Experimental Results

We further report results on Llama-3.1-8B to examine whether the observations in the main experiments transfer across model families. As shown in Table 3, on Mixture-of-Thoughts, CF-Mem selection achieves comparable utility to Random under the same 40% token budget, with average scores of 23.29 and 23.77, respectively. The available results are consistent with the main Qwen2.5-7B experiments, suggesting that CF-Mem-guided selection can preserve model utility beyond a single target model.

Table 4 further reports VB-Mem auditing results. CF-Mem consistently reduces the generation-probability thresholds across the high-VB-Mem tail compared with Random and Longest. For example, at the top 0.1%, the threshold decreases from 70.12 for Random and 74.01 for Longest to 41.29 for CF-Mem. This provides additional evidence that CF-Mem-guided selection can reduce verbatim leakage while maintaining comparable model utility.

Table 3: Performance of different training data selection methods on Llama-3.1-8B. All selection methods use a 40% token budget.

Training Set	Data Selection	GSM8K Avg@8	MATH-500 Avg@8	AMC23 Avg@8	AIME25 Pass@8	AIME26 Pass@8	Average
Original Model		13.03	6.17	1.56	0.00	0.00	4.15
Mixture-of-Thoughts (50K)	Random (40%)	68.62	32.00	11.56	3.33	3.33	23.77
	Longest (40%)	64.03	30.50	10.31	0.00	6.67	22.30
	CF-Mem (40%) (Ours)	66.22	30.77	12.81	3.33	3.33	23.29

Table 4: VB-Mem auditing results on Llama-3.1-8B. Each value is the generation-probability threshold (%) at the corresponding top VB-Mem percentile. Lower values indicate lower verbatim leakage risk. All selection methods use a 40% token budget.

Training Set	Data Selection	Top 0.1%	Top 0.2%	Top 0.5%	Top 1%	Top 2%
Mixture-of-Thoughts (50K)	Random (40%)	70.12	18.95	1.41	0.14	**0.01**
	Longest (40%)	74.01	40.77	5.21	0.22	0.02
	CF-Mem (40%) (Ours)	**41.29**	**14.82**	**0.89**	**0.10**	**0.01**