Window-based Membership Inference Attacks Against Fine-tuned Large Language Models

Yuetian Chen
Purdue University

Yuntao Du
Purdue University

Kaiyuan Zhang
Purdue University

Ashish Kundu
Cisco Research

Charles Fleming
Cisco Systems

Bruno Ribeiro
Purdue University

Ninghui Li
Purdue University

Abstract

Most membership inference attacks (MIAs) against Large Language Models (LLMs) rely on global signals, like average loss, to identify training data. This approach, however, dilutes the subtle, localized signals of memorization, reducing attack effectiveness. We challenge this global-averaging paradigm, positing that membership signals are more pronounced within localized contexts. We introduce WBC (Window-Based Comparison), which exploits this insight through a *sliding window approach* with *sign-based aggregation*. Our method slides windows of varying sizes across text sequences, with each window casting a binary vote on membership based on loss comparisons between target and reference models. By ensembling votes across geometrically spaced window sizes, we capture memorization patterns from token-level artifacts to phrase-level structures. Extensive experiments across eleven datasets demonstrate that WBC substantially outperforms established baselines, achieving higher AUC scores and 2–3× improvements in detection rates at low false positive thresholds. Our findings reveal that aggregating localized evidence is fundamentally more effective than global averaging, exposing critical privacy vulnerabilities in fine-tuned LLMs.

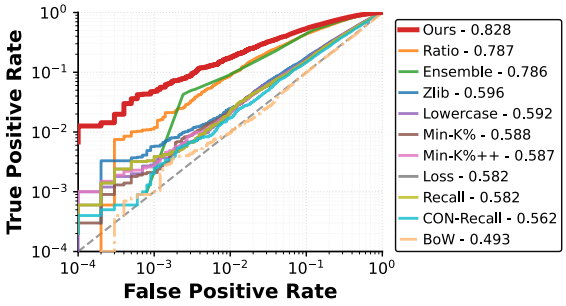

Figure 1: **Performance comparison of membership inference attacks on Pythia-2.8B.** ROC curves showing true positive rate vs. false positive rate (both on log scale) for various MIA methods evaluated on Web Samples V2 split. The diagonal dashed line represents random guessing performance. Numbers in legends indicate AUC scores. Our proposed WBC Attack significantly outperforms existing baselines across all false positive rate regimes, demonstrating superior membership inference capability.

1 Introduction

Large Language Models (LLMs) have achieved transformative success across numerous applications [14, 65, 76]. However, the training of these models often involves extensive datasets that can contain private or sensitive information. This practice introduces significant privacy risks, including the memorization and potential leakage of training data [8, 33, 36, 44]. Membership Inference Attacks (MIAs)—which determine whether specific samples were included in training data—are the primary method for quantifying these risks [75], with successful attacks directly demonstrating information leakage [24, 53, 64, 86].

These privacy vulnerabilities are particularly pronounced during the fine-tuning stage, where models are adapted to specialized, often proprietary, datasets [10, 53, 54]. Most established MIAs in this setting are reference-based, operating by comparing a global statistic—typically the average per-token loss—between a fine-tuned target model and a pre-trained reference model [1, 24, 80, 84].

In a reference-based MIA, for each instance, one computes token-level losses (negative log predicted probability) from the reference model minus those from the target model. To understand how to best utilize such token loss difference sequences for membership inference, we analyzed the distributions of loss differences for 10 million tokens. Our analysis reveals that membership signals manifest as sparse, extremal events rather than uniform distributional shifts. Furthermore, these membership signals are intermixed with sparser and more extreme events caused by domain-specific tokens, which have high loss reduction and occur with similar frequencies in non-members as well as members. These events create

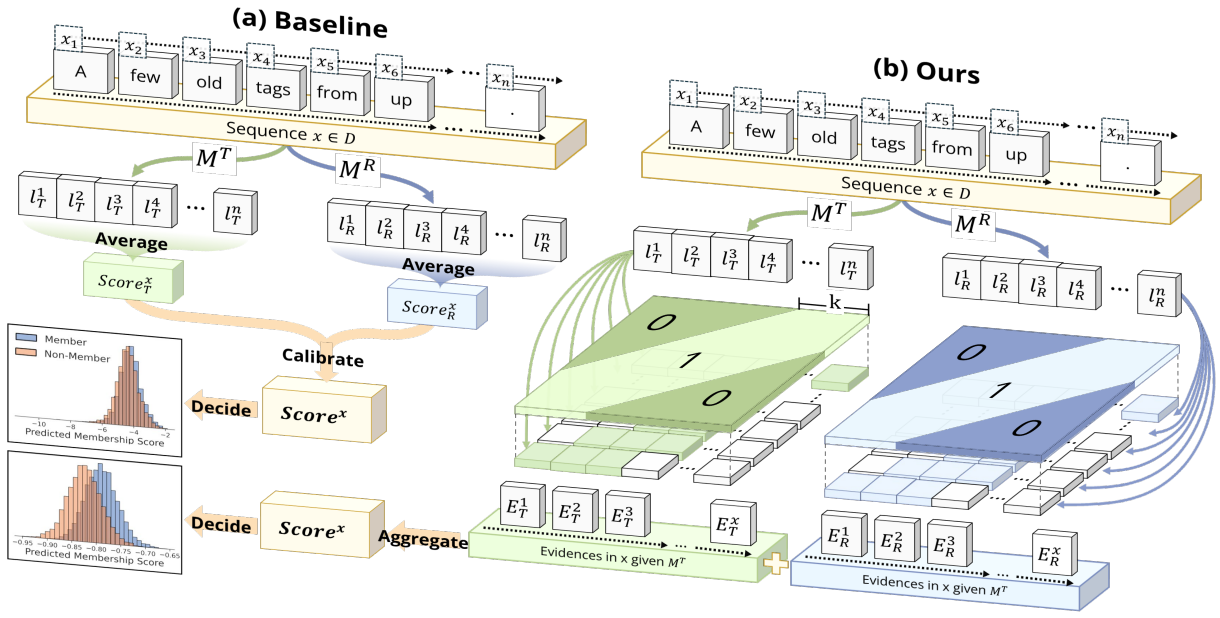

Figure 2: **Overview of the Window-Based Comparison (WBC) Attack.** Unlike baseline methods **(a)** that rely on comparing a single, noisy global average of per-token losses, our approach **(b)** introduces a local aggregation step. We slide a window across the loss sequences from the target ($\mathcal{M}^{\mathrm{T}}$) and reference ($\mathcal{M}^{\mathrm{R}}$) models, making a binary comparison for each window. The final membership score is the sum of this local evidence, a process that filters noise and provides a more sensitive measure, leading to better separation between member and non-member distributions.

long-tailed noise with potentially infinite variance; global averaging becomes unreliable—a single outlier can dominate the entire statistic, obscuring genuine membership signals concentrated in localized regions. This raises a critical question: *Can robust detection methods overcome the fundamental challenge of extracting sparse membership signals from long-tailed noise dominated by domain adaptation effects?*

Building on the insights gained from the empirical analysis, we construct a mathematical model for the token loss reduction sequences using the point process theory from extreme value statistics [22, 41]. Theoretical analysis using the model shows that the most reliable membership signals are concentrated in localized token sequences. While individual token losses are too volatile to be trusted, aggregating them over short windows (e.g., 3-10 consecutive tokens) can filter noise without destroying the underlying signal.

Taking advantage of these theoretical insights, we introduce the Window-Based Comparison (WBC) attack, illustrated in Figure 2. Our method employs a *sliding window approach* that performs hundreds of local comparisons, combined with *sign-based aggregation* that counts the fraction of windows favoring membership. Our contributions are as follows:

1. We are the first to use empirical analysis of distributions of token-level loss signals to understand how to construct more effective MIAs. Our analysis results in several intriguing findings. For example, counter-intuitively, the strongest membership signals occur on tokens where the fine-tuned model has a higher loss than the reference model. See Section 4.1.1 for discussion of these findings. We conjecture that a similar analysis could result in useful insights for MIA in pre-trained LLMs and possibly other paradigms, including vision-language [66] and diffusion language models [60].

2. We formalize these observations with a mixture of point process models, which explains why global averaging is suboptimal and provides theoretical grounding for localized detection. We thus propose the WBC attack that replaces global averaging with sliding window analysis. Our method captures localized memorization patterns while maintaining robustness to long-tailed noise. A geometric ensemble strategy aggregates evidence across multiple window sizes, eliminating parameter tuning.

3. Through extensive experiments on eleven diverse datasets using various models, we show that WBC significantly outperforms thirteen baseline attacks. Averaged across all datasets, WBC achieves an AUC of 0.839 compared to the strongest baseline's 0.754, and improves the True Positive Rate at 1% False Positive Rate by 2.8× (from 5.2% to 14.6%). Figure 1 shows this superiority on a representative example, the Web Samples V2 dataset.

Our findings prove that the aggregation of local signals is a more potent attack vector than previously established global methods. This work not only introduces a more effective MIA

but also underscores the need for defenses that can account for these localized memorization patterns.

2 Related Works

MIAs against LLMs initially showed limited success on pre-training data [13, 20, 49, 73], attributed to massive datasets, few epochs, and fuzzy boundaries [8, 39, 89]. Early LLM MIAs adapted loss thresholding [72, 75, 85], reference model calibration [7, 52, 80], and likelihood ratio tests [7, 19, 84]. Fine-tuned LLMs demonstrated significantly higher vulnerability [48, 53], motivating specialized attacks: neighborhood comparison [30, 47], loss trajectory exploitation [43, 45], token probability analysis [74, 87], and self-prompt calibration (SPV-MIA) [24]. Advanced methods include instruction-based detection (MIA-Tuner) [25], user-level inference [17, 38], context-aware attacks [9, 81], semantic-based approaches (SMIA) [55], and alignment-specific attacks [23], and instruction-tuned models [32]. Label-only attacks [11, 29], extraction methods [8, 57], and aggregation strategies [64] further expanded the threat landscape.

3 Preliminaries

This section provides the necessary background on autoregressive language models, establishing the formulation for per-token loss that underpins our attack. It then formally defines the threat model, detailing the adversary's objective, knowledge, and capabilities.

3.1 Autoregressive Language Model

Next-Token Prediction. The dominant paradigm for Large Language Models (LLMs) is autoregressive modeling [6, 68]. For a sequence of discrete tokens $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, these models operate on the unidirectional dependency assumption, where the probability of observing a token x_i depends only on its prefix $(x_1, \ldots, x_{i-1})$. This permits factorization of the joint probability as:

$$p(\mathbf{x}) = \prod_{i=1}^{n} p(x_i \mid x_1, \ldots, x_{i-1}) \tag{1}$$

A model $\mathcal{M}$ with parameters θ is trained to maximize this likelihood over a large corpus, typically by minimizing the cross-entropy loss. The core metric derived from this process, and the fundamental signal for our attack, is the per-token loss (negative log-likelihood) for a given sequence:

$$\ell_i^{\mathcal{M}} = -\log p_{\mathcal{M}}(x_i \mid x_1, \ldots, x_{i-1}) \tag{2}$$

This value, $\ell_i^{\mathcal{M}}$, quantifies the model's "surprise" at seeing token x_i given the preceding context.

Fine-Tuning and Memorization. While pre-trained on vast, general-domain corpora, LLMs are often specialized for downstream tasks via fine-tuning on smaller, targeted datasets. This process adapts the model's parameters to the new data distribution [15, 67]. A well-documented side effect of fine-tuning is that it amplifies the model's memorization of the training samples [52, 86]. Consequently, for a data sample $\mathbf{x}$ that is part of the fine-tuning set (a "member"), the target model $\mathcal{M}^{\mathrm{T}}$ will exhibit significantly lower average loss compared to its loss on unseen data. Reference-based attacks leverage this phenomenon by comparing the target model's loss against that of a reference model $\mathcal{M}^{\mathrm{R}}$ (typically the pre-trained base model), which has not been exposed to the fine-tuning data. The discrepancy in loss reduction between members and non-members provides a potent signal for membership inference.

3.2 Threat Model and Attacker Capabilities

Adversary's Objective. We investigate Membership Inference Attacks (MIAs) in the context of fine-tuned LLMs. The adversary's goal is to determine if a specific data record $\mathbf{x}$ was part of the private fine-tuning dataset, D_{train}. Given a target model $\mathcal{M}^{\mathrm{T}}$ fine-tuned on D_{train}, the adversary constructs an attack function $A(\mathbf{x}, \mathcal{M}^{\mathrm{T}}, \mathcal{M}^{\mathrm{R}})$ that outputs a real-valued score or a binary prediction $\hat{m}(\mathbf{x}) \in \{0, 1\}$, where $\hat{m}(\mathbf{x}) = 1$ signifies a prediction that $\mathbf{x} \in D_{\text{train}}$.

Access Model and Knowledge. We assume a *score-based black-box* access model, a realistic scenario for attacks against deployed LLMs where internal model details are inaccessible [74, 87, 90]. In this setting, the adversary's capabilities are limited to querying the target model $\mathcal{M}^{\mathrm{T}}$ with a text sequence $\mathbf{x}$ and receiving only the corresponding sequence of per-token loss values $\{\ell_i^{\mathrm{T}}\}_{i=1}^{n}$. This distinguishes our setting from stricter black-box scenarios where only sequence-level averaged losses are returned [24, 34, 50, 52, 75, 80]. This assumption reflects two standard deployment scenarios: (1) *Open-weight adaptation*, where practitioners fine-tune public models (e.g., via HuggingFace [82]) on proprietary data, granting attackers access to the model weights for local inference; and (2) *API-based inference*, where standard serving backends like vLLM [40] explicitly support parameters such as `prompt_logprobs`, returning the exact signal required by WBC. Furthermore, consistent with reference-based attack literature [24, 80], the adversary is assumed to have identical score-based black-box access to a reference model, $\mathcal{M}^{\mathrm{R}}$. The most principled choice for this reference is the pre-trained base model from which $\mathcal{M}^{\mathrm{T}}$ was fine-tuned, as this best isolates the memorization signal induced by the fine-tuning process. As shown in Section 5.5, WBC remains robust and continues to outperform baselines even when $\mathcal{M}^{\mathrm{R}}$ is a misaligned model. The adversary has no access to the model's internal components, such as its parameters, gradients, or hidden-state activations, nor to the membership status of any sample.

4 Window-Based Comparison Attack

A fundamental limitation of prior membership inference attacks is their reliance on global statistics that aggregate losses over entire texts. This document-level averaging is dominated by extremal events, rare tokens with outlier losses that overwhelm genuine membership signals. Our empirical analysis reveals that these outliers can be 10–100 times larger than typical fluctuations, making global statistics unreliable for detecting the sparse, localized memorization patterns that distinguish members from non-members. This motivates our window-based approach: by evaluating hundreds of local comparisons instead of a single global average, we can isolate membership signals from contaminating noise.

WBC attack exploits this clustering by systematically evaluating contiguous text windows. Rather than computing a single global statistic, we perform hundreds of local comparisons and aggregate their outcomes. This approach transforms the noisy, high-dimensional problem of token-level analysis into a robust voting mechanism across multiple granularities.

4.1 Theoretical Foundation: Extremal Events and Window-Based Detection

In this section, we provide a plausible explanation for why our sign-based localized window-based detection outperforms global averaging for membership inference. Our analysis of over 10 million token-level comparisons reveals that membership signals are governed by a regime of extremal events. These extremes arise from two sources: (1) tokens where the fine-tuned model achieves dramatic perplexity reduction compared to the reference model, appearing in both members and non-members due to domain adaptation, and (2) membership-specific tokens where fine-tuning provides additional confidence due to memorization. The former creates extreme values in the loss difference distribution that dominate global averages but carry no membership information, masking the discriminative signals from the latter. We argue that this behavior aligns with a mixture of point processes, where non-informative extremal events from domain adaptation obscure the sparse true membership signals. Our sliding window approach isolates these localized patterns, and our sign-based binary aggregation strategy provides robustness to the long-tailed distributions created by both types of extremes.

4.1.1 Empirical Structure of Membership Signals

To understand the nature of membership signals in fine-tuned LLMs, we conducted an empirical analysis of token-level loss differences. We fine-tuned Pythia-2.8B [5] on the Khan Academy subset of Cosmopedia [3], following standard practices with a learning rate of 5×10^{-5} and training for 3 epochs on 10,000 samples. We then evaluated both the fine-tuned target model $\mathcal{M}^{\mathrm{T}}$ and the original pre-trained reference model

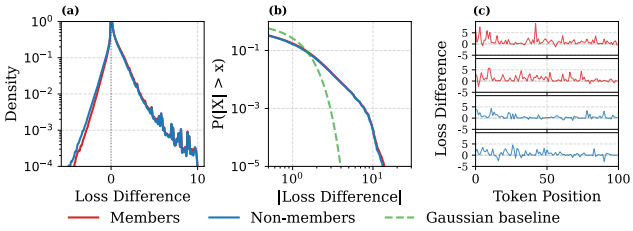

Figure 3: **Empirical distribution of token-level loss differences.** (a) Log-scale density plots reveal long-tailed distributions with a subtle rightward shift for members. The difference is most pronounced in the left tail, not the mean. (b) Complementary CDF confirms the long-tailed behavior of the token loss. (c) Time series shows sparse, scattered extremes.

$\mathcal{M}^{\mathrm{R}}$ on balanced sets of 10,000 member samples (from the training set) and 10,000 non-member samples (held-out data from the same distribution).

For each text sequence $\mathbf{x} = (x_1, \ldots, x_n)$, we computed the per-token loss (negative log-likelihood) for both models, yielding loss sequences $\{\ell_j^{\mathrm{T}}\}_{j=1}^n$ and $\{\ell_j^{\mathrm{R}}\}_{j=1}^n$. The token-level loss difference $\Delta_j = \ell_j^{\mathrm{R}} - \ell_j^{\mathrm{T}}$ quantifies how much more confident the fine-tuned model is compared to the reference model at each position. Positive values indicate positions where fine-tuning improved prediction.

Figure 3(a) presents the empirical distribution of these loss differences on a logarithmic scale to emphasize tail behavior. The distributions reveal several critical insights.

First, the curves for members and non-members look quite similar, but there is a small but non-trivial mean difference between members (0.393) and non-members (0.331). This suggests that using global averaging for membership inference would work to some limited extent. In order to achieve higher attack effectiveness, we need to identify and extract finer-grained membership signals from the token-loss sequences. Furthermore, the overwhelming majority of tokens show near-zero differences, suggesting that any fine-grained membership signals would appear as relatively rare events.

Second, the right tail regions for both members and non-members extend to very high positive values and are almost overlapping. This suggests that *tokens with dramatic loss reduction are not good membership signals*. While this observation may appear counterintuitive, it makes perfect sense. For tokens to result in high loss reduction, they must appear many times during fine-tuning, meaning that they are domain-specific features that appear frequently across the dataset, in non-members as well as members. We call these *domain-specific tokens*. High loss reduction on these tokens are the result of domain adaptation that occurs because of fine-tuning.

This fact creates significant challenges for distinguishing members from non-members. For example, the approach of using the maximum loss reduction across all tokens to deter-

mine membership would not be effective.

Third, the most noticeable difference between the two distributions is that the one for members shows a small but consistent rightward shift in the left tail region. As this region consists of tokens on which the fine-tuned model actually performs worse than the reference model, it counter-intuitively suggests that *the strongest membership signals appear in tokens where the target model has higher loss*.

We believe that the reasons for this phenomenon are as follows. First, for a token to be a good signal of membership of an instance, it should appear in this instance and few other instances. As a result, we expect this token's loss to be just slightly lower compared to the case that the instance is a non-member. Second, as the fine-tuned model needs to increase predicted probabilities for domain-specific features, it would have to reduce predicted probabilities for some other tokens. Since members are used in fine-tuning, such reductions, when they occur, would be less than those for non-members.

Compared to the right-tail region, the left-tail region extends less horizontally, but has higher density. This suggests that such membership signals have less magnitude than the noise signals from the right region, but are less rare.

Lastly, both member and non-member distributions exhibit long tails with excess kurtosis exceeding 18, far surpassing the value of 0 expected for Gaussian distributions. They also show positive skewness (2.82 for members, 2.63 for non-members), confirming the asymmetric nature of these extremal events. This means that the extremal events from both left-tail and right-tail regions will occur frequently enough to be exploited for membership inference. These observations motivate modeling membership signals as a mixture of extremal event processes rather than a uniform or Gaussian process.

Figure 3(b) shows the log-log scale of the token loss complementary cumulative distribution (i.e., 1-CDF). It reveals a long-tailed behavior, a hallmark of distributions whose averages are dominated by rare extreme events. For members, approximately 1.77% of tokens exceed three standard deviations from the mean, compared to only 0.3% expected under a Gaussian distribution.

Figure 3(c) illustrates the sequential structure of these signals by plotting loss differences across token positions for representative samples. Instead of clustered patterns suggestive of memorization of contiguous passages, we observe sparse, scattered spikes distributed seemingly at random throughout the sequences. Statistical analysis confirms this observation: computing the clustering coefficient (ratio of observed to expected spacing between extreme values) [12, 71] yields values of 1.049 for members and 1.053 for non-members, where 1.0 indicates perfectly random placement under a Poisson point process [16]. This absence of spatial clustering suggests that membership signals manifest as isolated, extremal events rather than coherent, memorized passages.

4.1.2 Modeling Membership Signals as Extremal Events

The empirical observations we made above from Figure 3 collectively suggest that membership inference should be framed not as detecting distribution shifts through averages but as identifying and aggregating evidence from sparse extremal events—a perspective grounded in point process theory from extreme value statistics [22, 41]. From these observations, we model the per-token loss difference point process as a superposition of three components:

$$\Delta_j(\mathbf{x}) = \ell_j^{\mathrm{R}} - \ell_j^{\mathrm{T}} = \mathbb{I}[\mathbf{x} \in D_{\text{train}}] \cdot \delta_j(\mathbf{x}) + \xi_j + \varepsilon_j, \quad (3)$$

Here ε_j represents baseline noise with a small mean μ_ε and variance σ^2, capturing typical prediction fluctuations. The term ξ_j captures *domain-specific tokens* (e.g., frequent technical terms) that exhibit high loss reduction due to domain adaptation independent of membership. Since these features appear in both members and non-members, ξ_j acts as high-magnitude noise that dominates the right tail of the distribution. We model these events due to domain-specific tokens as a point process where events occur with low probability but large magnitude. Specifically, $\xi_j = Z_j \cdot |Y_j|$ where $Z_j \sim \text{Bernoulli}(\rho_\xi)$ with $\rho_\xi \ll 1$ marks rare token occurrence, and Y_j follows a long-tailed distribution representing the magnitude of these events. From our observations, approximately 1.8% of tokens exhibit extreme values exceeding 3σ, suggesting $\rho_\xi \approx 0.02$. The term $\delta_j(\mathbf{x}) \geq 0$ represents the membership signal at position j. Unlike domain features, these signals manifest as smaller, more frequent loss reductions resulting from the memorization of specific instances, causing $\Delta_j(\mathbf{x})$ to increase when $\mathbf{x}$ is a member as opposed to a non-member.

Similarly, membership signals ($\delta_j(\mathbf{x})$'s) follow a sparse pattern: $\delta_j(\mathbf{x}) = B_j \cdot \gamma_j$ where $B_j \sim \text{Bernoulli}(\rho_\delta)$ indicates whether position j contains memorized content, and $\gamma_j > 0$ represents the signal strength when present.

The Masking Effect of Extremal Events in Global Averages. Under our model, global averaging reveals why traditional membership inference approaches struggle with fine-tuned LLMs. The global average loss difference across all tokens becomes:

$$\bar{\Delta} = \frac{1}{n}\sum_{j=1}^{n}\Delta_j = \underbrace{\mathbb{I}[\mathbf{x} \in D_{\text{train}}] \cdot \rho_\delta \bar{\gamma}}_{\text{membership signal}} + \underbrace{\frac{1}{n}\sum_{j=1}^{n}\xi_j}_{\text{rare token noise}} + \underbrace{\bar{\varepsilon}}_{\text{baseline noise}}. \quad (4)$$

The first term represents the true membership signal with expected value $\rho_\delta \bar{\gamma}$. However, the second term, arising from rare token events, has variance proportional to $\rho_\xi \cdot \mathbb{E}[Y_j^2]$. For long-tailed distributions of Y_j, this variance can be extremely large. Hence, a single extreme ξ_j value can dominate the entire average, as these outliers can be 10-100 times larger than typical fluctuations.

This simple model explains why global averaging methods can have weak MIA signals: the signal-to-noise ratio deteriorates not due to the weakness of membership signals, but because rare token events create overwhelming noise that cannot be averaged away. With only $\rho_\xi n \approx 0.02n$ extreme events in a sequence, the law of large numbers converges slowly (or not at all for some long-tailed distributions), making global statistics unreliable signals.

4.1.3 Window-Based Detection and Robust Aggregation

Instead of attempting to detect a global mean shift corrupted by long-tailed noise, we reformulate membership inference as detecting localized extremal events. This formulation aligns with standard problems in local sequence analysis, where the design space ranges from change-point detection algorithms to various aggregation functions. However, unlike change-point detection, which typically seeks contiguous regime shifts, our empirical analysis at Section 4.1.1 indicates that membership signals do not appear as such sharp transitions; rather, they manifest as continuous, low-degree signals that are consistently buried under the dominated signal of domain adaptation. We therefore adopt a sliding window approach as a particularly effective instance of this paradigm to capture these scattered events. For a window of size w starting at position i, we compute the windowed sum:

$$S_i(w) = \sum_{j=i}^{i+w-1} \Delta_j, \tag{5}$$

where the summation spans w consecutive tokens from position i to position $i+w-1$. By sliding this window across the entire sequence, we obtain $n-w+1$ different windows.

The distribution of these windowed sums follows a contaminated mixture:

$$S_i(w) \sim (1-p_{\text{extreme}}) \cdot \mathcal{F}_{\text{normal}} + p_{\text{extreme}} \cdot \mathcal{F}_{\text{long}} \tag{6}$$

where p_{extreme} is the probability that a window contains either a membership signal or a rare token event, and $\mathcal{F}_{\text{long}}$ represents a long-tailed distribution. Consider two natural approaches to aggregate these windowed differences into a membership score:

$$T_{\text{mean}} = \frac{1}{n-w+1} \sum_{i=1}^{n-w+1} S_i(w), \qquad \text{(mean-based)} \tag{7}$$

$$T_{\text{sign}} = \frac{1}{n-w+1} \sum_{i=1}^{n-w+1} \mathbb{I}[S_i(w) > 0]. \qquad \text{(sign-based)} \tag{8}$$

The mean-based approach uses the magnitude of each window's difference for MIA, while the sign-based approach uses only the direction. Classical robust statistics theory shows which method offers superior statistical power and under which conditions. The Pitman asymptotic relative efficiency (ARE) quantifies the relative sample sizes needed by two tests to achieve equal power [42, 63]. For location testing under contaminated distributions, the ARE is given by

$$\text{ARE}(\text{sign}, \text{mean}) = 4f^2(0) \cdot \text{Var}[S_i(w)], \tag{9}$$

where $f(0)$ is the density of the centered distribution at zero and $\text{Var}[S_i(w)]$ is the variance of $S_i(w)$. When ARE exceeds unity, the sign test requires fewer samples than the mean test for equal power [77].

For the long-tailed contaminated mixture in Equation (6), $\text{Var}[S_i(w)]$ is dominated by the long-tailed extremal events $\mathcal{F}_{\text{long}}$. Meanwhile, the density $f(0)$ remains bounded because most windows cluster near zero. Consequently, the ARE grows with contamination degree, strongly favoring the sign test. To illustrate this phenomenon, let's examine an extreme scenario: an unrealistic long-tailed distribution, such as Cauchy contamination. In this case, the ARE can be infinite, implying that the mean test has zero asymptotic efficiency relative to the sign test [35]. This result underscores the robustness of the sign test when variance is dominated by extremes. This theoretical result directly motivates our use of the sign test:

$$T_{\text{sign}} = \frac{1}{n-w+1} \sum_{i=1}^{n-w+1} \mathbb{I}[S_i(w) > 0] \tag{10}$$

$$= \frac{1}{n-w+1} \sum_{i=1}^{n-w+1} \mathbb{I}\left[\sum_{j=i}^{i+w-1} \ell_j^{\text{R}} > \sum_{j=i}^{i+w-1} \ell_j^{\text{T}}\right]. \tag{11}$$

This statistic counts the fraction of windows where the reference model loss exceeds the target model loss, regardless of difference magnitude. By adapting this standard non-parametric test to the domain of membership inference, we inherit several key known robustness guarantees:

Breakdown point. The breakdown point of an estimator is the largest fraction of data that can be arbitrarily corrupted without the estimator becoming uninformative [18, 35]. The sign test achieves the maximum possible breakdown point of 0.5, meaning it remains reliable even when up to half the windows contain arbitrarily large contaminating values from rare tokens. In our context, "window contamination" refers to windows where rare token events ξ_j create extreme values that dominate the window sum. Even if 50% of windows contain such extreme contaminations, the sign test still correctly identifies the majority vote, whereas the mean would be completely dominated by these outliers.

Scale invariance. The sign test is invariant to monotone transformations of the data [42]. Whether the loss differences are measured in nats, bits, or any monotone transformation thereof, the sign test yields identical results. Unlike the mean, which is sensitive to non-linear scaling, this invariance ensures consistent detection regardless of calibration differences.

Bounded output. Unlike the mean, which can take arbitrary values depending on outlier magnitudes, the sign statistic

is naturally bounded in [0,1], representing the fraction of windows favoring membership. This bounded range facilitates consistent threshold selection across different datasets and models without requiring dataset-specific normalization.

The fundamental distinction is that mean-based aggregation attempts to measure "how much lower is the loss on average," while sign-based comparison asks "how often is the loss lower", a more robust question under long-tailed noise. For empirically observed contamination levels ($p_{\text{extreme}} \approx 0.05 - 0.10$), the ARE typically exceeds 2 to 5, meaning sign-based comparison requires 2 to 5 times fewer samples than mean aggregation for equivalent detection power. This explains our empirical results in Section 5.3.3, where sign-based aggregation consistently outperforms mean, median, and min aggregation across all datasets, with particularly pronounced advantages in high-precision regimes.

4.1.4 Window Size Trade-off and Ensemble Strategy

The use of windows in detecting membership signals introduces a trade-off, governed by the window size w. On one hand, smaller windows (e.g., $w = 1$ or 2) increase the number of (correlated) tests that can be performed, as there are $n - w + 1$ possible windows in a sequence of length n. However, this comes at the cost of a poor signal-to-noise ratio for each individual window sum $S_i(w) = \sum_{j=i}^{i+w-1} \Delta_j$. Specifically, the expected signal scales with $\rho_\delta w \bar{\gamma}$, while the standard deviation due to baseline noise scales with $\sqrt{w}\sigma$. For very small w, the probability that a window containing a signal yields $S_i(w) > 0$ is barely above 0.5, making each binary test uninformative despite the large number of tests available.

Conversely, excessively large window sizes introduce three key challenges. In probability theory [58], our task is related to the well-known *scan statistic tests*, and the effect of w is well-documented. First, the *effective sample size* of any statistics diminishes sharply with window size w. Although the total number of windows is $n - w + 1$, which decreases only linearly with w, adjacent windows share $w - 1$ tokens, inducing strong dependencies. Consequently, the effective number of independent tests is closer to n/w, the maximum number of non-overlapping windows, which decays rapidly as w increases.

Second, large windows elevate the *risk of contamination by rare token events*. When a window contains an extreme rare token event ξ_j, its magnitude can dominate the sum $S_i(w)$, overshadowing the accumulated membership signal. The probability of such contamination is proportional to $1 - (1 - \rho_\xi)^w \approx w\rho_\xi$ for small ρ_ξ, where ρ_ξ is the rare token probability.

Third, *signal dilution* becomes pronounced. In a window of size w, only $\rho_\delta w$ tokens are expected to be memorized, while the remaining $(1 - \rho_\delta)w$ tokens are non-memorized. For sparse signals ($\rho_\delta \ll 1$), this averaging dilutes the membership signal, reducing the likelihood that windows containing signals yield positive sums. Empirically, an intermediate window (3–10 tokens) balances these competing factors. However, determining the optimal window size analytically requires parameters that cannot be estimated reliably: the signal sparsity ρ_δ, rare token frequency ρ_ξ, signal strength $\bar{\gamma}$, and rare token distribution $\mathbb{E}[Y^2]$. These parameters vary across datasets and even within documents—e.g., technical sections exhibit different patterns than narrative prose. We provide a detailed analysis of this optimization problem in Appendix A.

Thus, instead of pursuing an elusive optimal window size, we adopt an ensemble approach that provides robustness through diversification. We employ a geometric progression that densely samples small windows while maintaining coverage of larger scales:

$$w_k = \text{round}\left(w_{\min} \cdot \left(\frac{w_{\max}}{w_{\min}} \right)^{\frac{k-1}{|W|-1}} \right) \tag{12}$$

where $k \in \{1, \ldots, |W|\}$, and $w_{\min}$, $w_{\max}$ bound the range of scales. The geometric spacing ensures that the ratio between consecutive window sizes remains approximately constant: $w_{k+1}/w_k \approx (w_{\max}/w_{\min})^{1/(|W|-1)}$. This provides equal relative resolution across scales, a principle from scale-space theory that optimizes multi-resolution analysis. Small windows, where memorization signals are most likely, receive dense sampling, while larger windows are sampled more sparsely to provide coverage without redundancy. The final ensemble score uses uniform weights, yielding

$$S_{\text{WBC}} = \frac{1}{|W|} \sum_{k=1}^{|W|} T_{\text{sign}}(w_k). \tag{13}$$

Uniform weighting follows the principle of maximum entropy—minimizing worst-case regret under parameter uncertainty. This equal weighting also provides variance reduction through averaging partially correlated measurements, with ensemble variance decreasing as $\tau^2[1 + (|W| - 1)\rho]/|W|$ when correlations ρ are modest.

This equal weighting is theoretically justified under parameter uncertainty. By the principle of maximum entropy, uniform weights minimize worst-case regret when the true optimal window size is unknown. Additionally, uniform averaging provides variance reduction: if individual window scores have variance τ^2 and correlation ρ, the ensemble variance becomes $\tau^2[1 + (|W| - 1)\rho]/|W|$, which decreases with ensemble size when correlations are modest.

The ensemble strategy solves two fundamental problems. First, it provides robustness against unknown parameters (ρ_δ, ρ_ξ, $\bar{\gamma}$, $\mathbb{E}[Y^2]$) that vary across datasets and even within documents. Second, it captures heterogeneous memorization patterns that no single window size can detect optimally—small windows (2-4 tokens) capture token-level artifacts, medium windows (5-10 tokens) detect phrases, and larger windows identify paragraph-level patterns. When texts mix technical

Algorithm 1: WBC Attack

Inputs : Target model $\mathcal{M}^{\mathrm{T}}$; Reference model $\mathcal{M}^{\mathrm{R}}$; Input $\mathbf{x} = \{x_j\}_{j=1}^{n}$; Window size schemes W

1 **forall** $k \in \{T, R\}$ **do**
2 **for** $j = 1$ **to** n **do**
3 $\ell_j^k \leftarrow -\log p_{\mathcal{M}^k}(x_j \mid x_1, \ldots, x_{j-1})$
4 $S_{\mathrm{WBC}} \leftarrow 0$
5 **forall** $w \in W$ **do**
6 $sum^{\mathrm{T}} \leftarrow 0$; $sum^{\mathrm{R}} \leftarrow 0$ **for** $j = 1$ **to** w **do**
7 $sum^{\mathrm{T}} \leftarrow sum^{\mathrm{T}} + \ell_j^{\mathrm{T}}$ $sum^{\mathrm{R}} \leftarrow sum^{\mathrm{R}} + \ell_j^{\mathrm{R}}$
 ▷ Initialize sums for first window
8 count $\leftarrow 0$ **if** $sum^R > sum^T$ **then**
9 count $\leftarrow$ count $+ 1$
10 **for** $i = 2$ **to** $n - w + 1$ **do**
11 $sum^{\mathrm{T}} \leftarrow sum^{\mathrm{T}} - \ell_{i-1}^{\mathrm{T}} + \ell_{i+w-1}^{\mathrm{T}}$
12 $sum^{\mathrm{R}} \leftarrow sum^{\mathrm{R}} - \ell_{i-1}^{\mathrm{R}} + \ell_{i+w-1}^{\mathrm{R}}$
13 **if** $sum^R > sum^T$ **then**
14 count $\leftarrow$ count $+ 1$
 ▷ Slide & compare
15 $T_{\mathrm{sign}}(w) \leftarrow$ count$/(n - w + 1)$
16 $S_{\mathrm{WBC}} \leftarrow S_{\mathrm{WBC}} + T_{\mathrm{sign}}(w)$
17 $S_{\mathrm{WBC}} \leftarrow S_{\mathrm{WBC}}/|W|$
18 **return** S_{WBC}

terms with quoted passages, the ensemble naturally combines these complementary detection capabilities.

This ensemble consistently outperforms the empirically best single window size, as validated in our ablation studies (Sections 5.3.2). The strategy thus achieves both robustness and superior performance—transforming theoretical insights into a practical attack requiring no parameter tuning.

4.2 Algorithmic Specification

The complete procedure for our attack is detailed in Algorithm 1. The procedure unfolds as follows. First, the per-token negative log-likelihoods (i.e., the negative log-probability of each token conditioned on its prefix) are computed for the input $\mathbf{x}$ using both the target ($\mathcal{M}^{\mathrm{T}}$) and reference ($\mathcal{M}^{\mathrm{R}}$) models, yielding the loss sequences $\{\ell_j^{\mathrm{T}}\}_{j=1}^{n}$ and $\{\ell_j^{\mathrm{R}}\}_{j=1}^{n}$ (Lines 1-3). The algorithm then iterates through each window size $w \in W$ (Line 5). For a given w, it initializes the sums for the first window (Lines 6-7) and performs the first sign-based comparison (Lines 8-9). It then slides the window from the second position to the end, incrementally updating the sums and the comparison count at each step (Lines 10-14). Once all windows for size w are processed, the normalized count yields the window-specific score $T_{\mathrm{sign}}(w)$ (Line 15), which is added to the total ensemble score (Line 16). Finally, this total is averaged over the number of window sizes $|W|$ to produce the final score S_{WBC} (Lines 17-18). In practice, we further accelerate this computation by leveraging optimized convolution implementations. Details and performance evaluation are provided in Appendix C.

5 Experiments

This section details the empirical evaluation of our proposed WBC attack. We first describe the experimental setup, including the datasets, models, and baseline attacks. We then present our main results, demonstrating the superior performance of our method, followed by ablation studies and analyses that provide deeper insights into why our approach is effective.

5.1 Experimental Setup

Datasets. We evaluate on eleven datasets spanning synthetic and real-world domains. The first category consists of six subsets from Cosmopedia [3], a large-scale synthetic dataset generated by Mixtral-8x7B-Instruct-v0.1 [37]. We utilize the Khan Academy, Stanford, Stories, Web Samples v2, AutoMathText, and WikiHow subsets as these represent the scale and quality of data commonly used in modern fine-tuning practices, where practitioners increasingly rely on high-quality synthetic data for domain adaptation. The second category comprises real-world document benchmarks, including WikiText-103 [51] and XSum [56] for direct comparison with prior work [24], supplemented by Amazon Reviews [59], CC-News [28], and Reddit [70] to extend evaluation across diverse domains at similar scales. All evaluations use balanced 10,000-sample splits for members/non-members with a minimum 512-token length.

Models and Fine-Tuning. We primarily analyze Pythia-2.8B [5], with additional experiments on Pythia scaling suite (including the 160M, 410M, 1B, 1.4B, and 6.9B parameter models), GPT-2 [68], GPT-J-6B [78], Llama-3.2-3B [21], and a state-space model, Mamba-1.4B [27]. $\mathcal{M}^{\mathrm{T}}$ are fine-tuned from pre-trained bases, which serve as $\mathcal{M}^{\mathrm{R}}$.

Attack Baselines. We evaluate our proposed method against a comprehensive suite of thirteen established MIA baselines, covering both reference-free and reference-based approaches. The reference-free attacks include the straightforward Loss score (average negative log-likelihood) [85], methods based on input perturbations like ZLIB (which measures text compressibility) and Lowercase (which measures loss change after case modification) [8], tail-end distribution methods including Min-K% [74] and Min-K%++ [87], and distribution-based DC-PDD [90]. The reference-based attacks include the foundational Ratio and Difference methods that compare the target model's average loss to a reference model's [80]; context-aware query methods like ReCall [83] and CON-ReCall [79]

Table 1: MIA performance (AUC, TPR@10%FPR, TPR@1%FPR, TPR@0.1%FPR) across different datasets. Each cell shows mean with std. dev. as a subscript. Best-performing results are highlighted. We observe that WBC consistently outperforms all baseline methods across all reported metrics and datasets.

MIAs	Khan Academy				Stanford				Stories			
	AUC	T@10%	T@1%	T@0.1%	AUC	T@10%	T@1%	T@0.1%	AUC	T@10%	T@1%	T@0.1%
Loss [85]	$0.568_{\pm.003}$	$0.151_{\pm.003}$	$0.017_{\pm.002}$	$0.001_{\pm.001}$	$0.590_{\pm.006}$	$0.170_{\pm.008}$	$0.023_{\pm.003}$	$0.003_{\pm.001}$	$0.587_{\pm.004}$	$0.146_{\pm.008}$	$0.020_{\pm.001}$	$0.003_{\pm.001}$
ZLIB [8]	$0.583_{\pm.004}$	$0.161_{\pm.003}$	$0.019_{\pm.003}$	$0.003_{\pm.001}$	$0.606_{\pm.006}$	$0.178_{\pm.009}$	$0.024_{\pm.003}$	$0.002_{\pm.000}$	$0.596_{\pm.004}$	$0.157_{\pm.005}$	$0.019_{\pm.002}$	$0.003_{\pm.001}$
Lowercase [8]	$0.586_{\pm.004}$	$0.154_{\pm.006}$	$0.013_{\pm.002}$	$0.002_{\pm.001}$	$0.596_{\pm.003}$	$0.173_{\pm.004}$	$0.024_{\pm.002}$	$0.003_{\pm.001}$	$0.593_{\pm.004}$	$0.172_{\pm.004}$	$0.018_{\pm.002}$	$0.002_{\pm.001}$
Min-K% [74]	$0.595_{\pm.002}$	$0.182_{\pm.004}$	$0.024_{\pm.002}$	$0.002_{\pm.000}$	$0.593_{\pm.005}$	$0.168_{\pm.006}$	$0.024_{\pm.002}$	$0.003_{\pm.001}$	$0.588_{\pm.003}$	$0.145_{\pm.004}$	$0.019_{\pm.003}$	$0.003_{\pm.001}$
Min-K%++ [87]	$0.596_{\pm.002}$	$0.180_{\pm.005}$	$0.022_{\pm.003}$	$0.002_{\pm.001}$	$0.592_{\pm.004}$	$0.161_{\pm.006}$	$0.023_{\pm.002}$	$0.004_{\pm.001}$	$0.586_{\pm.003}$	$0.150_{\pm.006}$	$0.018_{\pm.002}$	$0.002_{\pm.001}$
BoWs [13]	$0.499_{\pm.004}$	$0.101_{\pm.003}$	$0.011_{\pm.001}$	$0.001_{\pm.001}$	$0.498_{\pm.003}$	$0.109_{\pm.003}$	$0.012_{\pm.001}$	$0.001_{\pm.001}$	$0.502_{\pm.005}$	$0.100_{\pm.002}$	$0.011_{\pm.001}$	$0.002_{\pm.001}$
ReCall [83]	$0.568_{\pm.003}$	$0.151_{\pm.003}$	$0.017_{\pm.002}$	$0.001_{\pm.001}$	$0.590_{\pm.006}$	$0.170_{\pm.008}$	$0.023_{\pm.003}$	$0.003_{\pm.001}$	$0.587_{\pm.004}$	$0.146_{\pm.008}$	$0.020_{\pm.001}$	$0.003_{\pm.001}$
CON-Recall [79]	$0.563_{\pm.003}$	$0.142_{\pm.004}$	$0.014_{\pm.002}$	$0.002_{\pm.001}$	$0.570_{\pm.006}$	$0.159_{\pm.007}$	$0.018_{\pm.002}$	$0.003_{\pm.000}$	$0.558_{\pm.004}$	$0.143_{\pm.006}$	$0.020_{\pm.002}$	$0.002_{\pm.001}$
DC-PDD [90]	$0.567_{\pm.003}$	$0.136_{\pm.003}$	$0.015_{\pm.002}$	$0.002_{\pm.001}$	$0.570_{\pm.003}$	$0.140_{\pm.003}$	$0.014_{\pm.002}$	$0.002_{\pm.001}$	$0.574_{\pm.002}$	$0.146_{\pm.002}$	$0.019_{\pm.002}$	$0.002_{\pm.001}$
SPV-MIA [24]	$0.695_{\pm.003}$	$0.240_{\pm.006}$	$0.049_{\pm.004}$	$0.005_{\pm.003}$	$0.760_{\pm.003}$	$0.260_{\pm.010}$	$0.077_{\pm.005}$	$0.012_{\pm.003}$	$0.763_{\pm.004}$	$0.360_{\pm.007}$	$0.070_{\pm.006}$	$0.018_{\pm.005}$
Ratio [80]	$0.703_{\pm.003}$	$0.264_{\pm.004}$	$0.037_{\pm.004}$	$0.003_{\pm.001}$	$0.781_{\pm.004}$	$0.401_{\pm.010}$	$0.080_{\pm.008}$	$0.013_{\pm.001}$	$0.769_{\pm.004}$	$0.389_{\pm.011}$	$0.091_{\pm.008}$	$0.022_{\pm.006}$
Difference [80]	$0.692_{\pm.002}$	$0.259_{\pm.005}$	$0.045_{\pm.002}$	$0.003_{\pm.001}$	$0.742_{\pm.005}$	$0.360_{\pm.011}$	$0.080_{\pm.008}$	$0.021_{\pm.004}$	$0.719_{\pm.005}$	$0.348_{\pm.007}$	$0.079_{\pm.006}$	$0.014_{\pm.003}$
Ensemble [88]	$0.687_{\pm.003}$	$0.217_{\pm.004}$	$0.061_{\pm.002}$	$0.007_{\pm.001}$	$0.738_{\pm.004}$	$0.142_{\pm.003}$	$0.075_{\pm.006}$	$0.011_{\pm.004}$	$0.758_{\pm.004}$	$0.338_{\pm.004}$	$0.048_{\pm.003}$	$0.014_{\pm.003}$
WBC (Ours)	$\mathbf{0.837}_{\pm.003}$	$\mathbf{0.538}_{\pm.007}$	$\mathbf{0.146}_{\pm.009}$	$\mathbf{0.026}_{\pm.009}$	$\mathbf{0.854}_{\pm.003}$	$\mathbf{0.583}_{\pm.008}$	$\mathbf{0.194}_{\pm.012}$	$\mathbf{0.034}_{\pm.017}$	$\mathbf{0.808}_{\pm.005}$	$\mathbf{0.494}_{\pm.007}$	$\mathbf{0.160}_{\pm.013}$	$\mathbf{0.034}_{\pm.003}$

MIAs	Web Samples v2				Auto Math Text				WikiHow			
	AUC	T@10%	T@1%	T@0.1%	AUC	T@10%	T@1%	T@0.1%	AUC	T@10%	T@1%	T@0.1%
Loss [85]	$0.579_{\pm.004}$	$0.153_{\pm.006}$	$0.022_{\pm.003}$	$0.004_{\pm.001}$	$0.553_{\pm.005}$	$0.133_{\pm.005}$	$0.014_{\pm.001}$	$0.002_{\pm.001}$	$0.592_{\pm.004}$	$0.162_{\pm.005}$	$0.022_{\pm.002}$	$0.004_{\pm.001}$
ZLIB [8]	$0.593_{\pm.004}$	$0.175_{\pm.005}$	$0.023_{\pm.002}$	$0.005_{\pm.001}$	$0.562_{\pm.005}$	$0.141_{\pm.005}$	$0.015_{\pm.002}$	$0.002_{\pm.001}$	$0.604_{\pm.004}$	$0.171_{\pm.004}$	$0.022_{\pm.002}$	$0.002_{\pm.000}$
Lowercase [8]	$0.592_{\pm.004}$	$0.166_{\pm.005}$	$0.021_{\pm.003}$	$0.003_{\pm.001}$	$0.568_{\pm.003}$	$0.141_{\pm.008}$	$0.015_{\pm.002}$	$0.001_{\pm.001}$	$0.614_{\pm.004}$	$0.182_{\pm.007}$	$0.024_{\pm.002}$	$0.001_{\pm.001}$
Min-K% [74]	$0.587_{\pm.005}$	$0.158_{\pm.007}$	$0.023_{\pm.002}$	$0.002_{\pm.000}$	$0.581_{\pm.006}$	$0.150_{\pm.006}$	$0.021_{\pm.001}$	$0.002_{\pm.000}$	$0.600_{\pm.004}$	$0.178_{\pm.005}$	$0.025_{\pm.002}$	$0.004_{\pm.001}$
Min-K%++ [87]	$0.586_{\pm.004}$	$0.153_{\pm.005}$	$0.020_{\pm.002}$	$0.003_{\pm.001}$	$0.583_{\pm.005}$	$0.150_{\pm.005}$	$0.022_{\pm.002}$	$0.003_{\pm.001}$	$0.598_{\pm.004}$	$0.174_{\pm.005}$	$0.027_{\pm.003}$	$0.004_{\pm.001}$
BoWs [13]	$0.493_{\pm.004}$	$0.097_{\pm.004}$	$0.010_{\pm.001}$	$0.002_{\pm.001}$	$0.500_{\pm.004}$	$0.105_{\pm.005}$	$0.009_{\pm.001}$	$0.001_{\pm.000}$	$0.498_{\pm.004}$	$0.096_{\pm.003}$	$0.008_{\pm.001}$	$0.001_{\pm.000}$
ReCall [83]	$0.579_{\pm.004}$	$0.153_{\pm.006}$	$0.022_{\pm.003}$	$0.004_{\pm.001}$	$0.553_{\pm.005}$	$0.133_{\pm.005}$	$0.014_{\pm.001}$	$0.002_{\pm.001}$	$0.592_{\pm.004}$	$0.162_{\pm.005}$	$0.022_{\pm.002}$	$0.004_{\pm.001}$
CON-Recall [79]	$0.560_{\pm.004}$	$0.144_{\pm.003}$	$0.017_{\pm.001}$	$0.002_{\pm.001}$	$0.547_{\pm.005}$	$0.135_{\pm.005}$	$0.015_{\pm.001}$	$0.002_{\pm.001}$	$0.571_{\pm.005}$	$0.161_{\pm.005}$	$0.019_{\pm.002}$	$0.002_{\pm.001}$
DC-PDD [90]	$0.569_{\pm.003}$	$0.143_{\pm.003}$	$0.019_{\pm.002}$	$0.003_{\pm.001}$	$0.560_{\pm.006}$	$0.130_{\pm.006}$	$0.015_{\pm.002}$	$0.002_{\pm.001}$	$0.571_{\pm.004}$	$0.143_{\pm.003}$	$0.016_{\pm.002}$	$0.002_{\pm.001}$
SPV-MIA [24]	$0.787_{\pm.003}$	$0.456_{\pm.006}$	$0.088_{\pm.004}$	$0.008_{\pm.006}$	$0.759_{\pm.003}$	$0.369_{\pm.006}$	$0.063_{\pm.006}$	$0.007_{\pm.004}$	$0.719_{\pm.003}$	$0.300_{\pm.007}$	$0.053_{\pm.005}$	$0.013_{\pm.004}$
Ratio [80]	$0.788_{\pm.003}$	$0.435_{\pm.009}$	$0.090_{\pm.007}$	$0.012_{\pm.003}$	$0.768_{\pm.002}$	$0.378_{\pm.005}$	$0.075_{\pm.002}$	$0.011_{\pm.006}$	$0.714_{\pm.003}$	$0.261_{\pm.006}$	$0.042_{\pm.003}$	$0.008_{\pm.002}$
Difference [80]	$0.739_{\pm.003}$	$0.365_{\pm.008}$	$0.094_{\pm.008}$	$0.019_{\pm.007}$	$0.700_{\pm.004}$	$0.300_{\pm.009}$	$0.051_{\pm.006}$	$0.009_{\pm.003}$	$0.709_{\pm.003}$	$0.278_{\pm.005}$	$0.047_{\pm.003}$	$0.009_{\pm.002}$
Ensemble [88]	$0.786_{\pm.003}$	$0.477_{\pm.004}$	$0.086_{\pm.002}$	$0.005_{\pm.013}$	$0.749_{\pm.002}$	$0.359_{\pm.004}$	$0.051_{\pm.002}$	$0.003_{\pm.002}$	$0.724_{\pm.003}$	$0.338_{\pm.004}$	$0.065_{\pm.005}$	$0.018_{\pm.001}$
WBC (Ours)	$\mathbf{0.843}_{\pm.003}$	$\mathbf{0.573}_{\pm.008}$	$\mathbf{0.198}_{\pm.012}$	$\mathbf{0.045}_{\pm.004}$	$\mathbf{0.814}_{\pm.003}$	$\mathbf{0.504}_{\pm.007}$	$\mathbf{0.153}_{\pm.009}$	$\mathbf{0.040}_{\pm.008}$	$\mathbf{0.802}_{\pm.003}$	$\mathbf{0.451}_{\pm.010}$	$\mathbf{0.096}_{\pm.007}$	$\mathbf{0.019}_{\pm.006}$

that test a model's ability to complete a prefix; self-prompt verification with SPV-MIA [24]; and classifier-based attacks including a data-oriented Bag-of-Words baseline [13] and an Ensemble attack that uses multiple loss-based statistics as features [88]. Our proposed method, WBC, is evaluated using its ensemble configuration with $w_{\min} = 2$ and $w_{\max} = 40$.

Evaluation Metrics. We evaluate attack performance using AUC-ROC, TPR at low FPR thresholds (10%, 1%, 0.1%), and model utility via perplexity shown in Appendix B.1. All metrics are reported with mean and standard deviation over 100 bootstrap runs following [4].

5.2 Main Results

The main attack performance results, presented in Table 1, demonstrate that WBC decisively outperforms all baselines across the six Cosmopedia datasets shown; results for the five real-world datasets are provided in Appendix B.2 due to space constraints, with consistently strong performance across all eleven datasets. WBC achieves an average AUC of 0.826 compared to the strongest baseline, Ratio's 0.754, with consistent improvements across diverse domains; for instance, on the Stanford dataset, WBC reaches an AUC of 0.845, significantly surpassing the 0.781 achieved by Ratio. The superiority of our method is most pronounced in the critical low-FPR regime. On the Web Samples v2 dataset, WBC achieves a True Positive Rate at 1% False Positive Rate (TPR@1%FPR) of 19.8%, more than doubling the 9.4% from the strongest baseline on that metric, Difference. At the extreme 0.1% FPR level on Khan Academy, WBC identifies 2.6% of members—a 3.7-fold increase over the next best method, Ensemble (0.7%). This performance contrasts with reference-free baselines, whose near-random performance (AUC ≈ 0.6) validates that a reference model is essential for effectively attacking modern fine-tuned LLMs.

5.3 Ablation Studies

We conduct systematic ablation studies to validate each component of WBC and quantify their individual contributions.

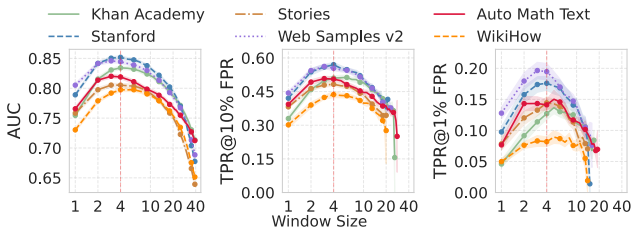

Figure 4: **Window size trade-off.** Performance of single-window WBC attacks as a function of window size $w \in [1, 39]$ on six datasets. Metrics include AUC, TPR@10% FPR, and TPR@1% FPR. Shaded regions show the standard deviation over 100 bootstrap samples.

Table 2: **Ensemble configuration comparison on Khan Academy dataset.** Performance of different window size combinations. The full ensemble achieves optimal balance between coverage and computational cost.

Configuration	AUC	TPR@10%	TPR@1%
Single Best (w=4)	$0.8341_{\pm.0024}$	$0.5103_{\pm.0105}$	$0.1262_{\pm.0094}$
Small Range	$0.8340_{\pm.0028}$	$0.5195_{\pm.0091}$	$0.1296_{\pm.0090}$
Large Range	$0.7836_{\pm.0030}$	$0.4293_{\pm.0079}$	$0.0011_{\pm.0110}$
Full Ensemble	$0.8369_{\pm.0027}$	$0.5384_{\pm.0076}$	$0.1464_{\pm.0088}$
Extended	$0.8268_{\pm.0030}$	$0.5209_{\pm.0084}$	$0.1378_{\pm.0085}$
Linear Spacing	$0.8241_{\pm.0034}$	$0.5147_{\pm.0096}$	$0.1389_{\pm.0087}$
Random Selection	$0.8065_{\pm.0026}$	$0.4720_{\pm.0090}$	$0.1325_{\pm.0078}$

5.3.1 Optimal Window Size Determination

Figure 4 empirically confirms our theoretical window size trade-off. Performance generally peaks at small windows ($w \in [3, 4]$) before degrading, reflecting the balance between accumulating sufficient signal and avoiding dilution. The exact optimum varies by dataset, e.g., Khan Academy peaks at $w = 4$ (AUC 0.834) while Stories optimizes at $w = 3$ (AUC 0.806), driven by unobservable parameters like signal sparsity ρ_δ and rare token frequency ρ_ξ.

Performance declines with larger windows; on Khan Academy, increasing w from 4 to 32 drops AUC to 0.702, with TPR@1%FPR collapsing to near zero beyond $w = 20$. This validates our model: larger windows increase contamination probability and reduce the effective sample size (e.g., from 509 tests at $w = 4$ to 257 at $w = 256$), overwhelming the sparse membership signals.

5.3.2 Ensemble Composition Analysis

We evaluate different ensemble strategies for aggregating evidence across scales. Table 2 compares seven configurations on Khan Academy, with consistent trends observed across all datasets shown in Appendix B.3. Our geometric ensemble follows the progression defined in Equation 12 with $w_{\min} = 2$, $w_{\max} = 40$, and $|W| = 10$. This achieves the highest AUC, outperforming all alternatives.

Systematic baselines includes: *Linear spacing* with $W = \{w_{\min} + i\Delta : i \in [0, |W| - 1]\}$ where $\Delta = (w_{\max} - w_{\min})/(|W| - 1)$ reaching AUC 0.8241, a 1.5% drop. *Extended coverage* using $W = \{w : w = w_{\min} + ki, k \in \mathbb{N}, w < n/2\}$ with larger stride $i = 16$ reaches windows near $n/2$ but yields only AUC 0.8268. Range-restricted ensembles isolate scales: *Small range* $W = [w_{\min}, w_{\min} + 4]$ preserves 99.7% of full performance, whereas *Large range* $W = w \in [18, 50] : |w - w'| \geq 7$ falls to 0.7836 AUC with near-zero TPR@1%FPR, confirming that large windows add noise rather than signal. Notably, most reasonable configurations achieve AUC above 0.80, demonstrating the robustness of window-based aggregation—even random selection of 10 windows yields AUC 0.8065. However, the geometric ensemble's systematic design provides crucial advantages in high-precision regimes: while AUC improves modestly (0.3%), TPR@1%FPR increases by 16% over the single best window. This disproportionate gain in low-FPR detection, where security applications operate, justifies the minimal computational overhead (6% additional cost) and establishes geometric spacing as the optimal configuration for practical deployment. The consistent superiority across datasets confirms that geometric progression effectively balances the exploration-exploitation trade-off without requiring dataset-specific tuning.

5.3.3 Aggregation Method Comparison

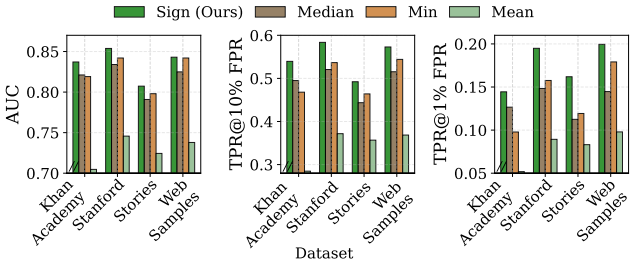

Figure 5: **Aggregation method comparison across datasets.** (a) AUC performance across four aggregation strategies. (b,c) TPR at low FPR thresholds shows amplified advantages in high-precision regimes.

Figure 5 validates our theoretical prediction that sign-based aggregation provides superior robustness to long-tailed loss distributions. Sign aggregation achieves the highest average AUC (0.835) compared to mean (0.728), median (0.818), and min (0.825), with a 2.2× advantage in TPR@1%FPR.

The relative performance of magnitude-based methods (mean, median, min) varies by dataset, reflecting different noise characteristics. However, all underperform sign aggregation, which achieves provable robustness through its bounded [0,1] output range and invariance to outlier magnitudes. Min

aggregation's strong performance suggests membership manifests as consistently lower losses across multiple windows rather than isolated memorized passages.

5.3.4 Text Length Scaling

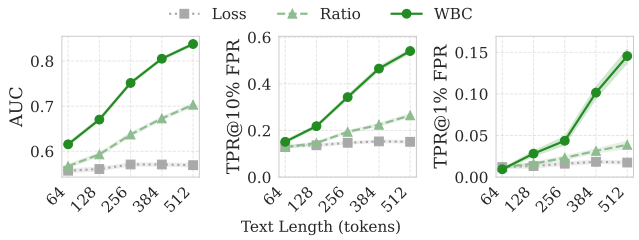

Figure 6: **Attack performance scaling with text length.** Loss, Ratio, and WBC on texts truncated to different length. (a) WBC 's margin over baselines grows with L, reaching 20.8% over Ratio at 512 tokens. (b,c) TPR scales super-linearly for WBC.

Window-based analysis requires sufficient text to extract multiple measurements. Figure 6 examines performance scaling by truncating 512-token samples to lengths $L \in \{64, 128, 256, 384, 512\}$. WBC's advantage over baselines increases monotonically with length. At $L = 64$ (permitting only 62 windows of size 3), WBC achieves AUC 0.606 versus Ratio's 0.544—an 11.4% improvement. This gap widens to 15.4% at $L = 256$ and 20.8% at $L = 512$. The scaling is super-linear for precision metrics: TPR@1%FPR grows from 0.019 to 0.135 for WBC (7.1× increase) versus 0.009 to 0.037 for Ratio (4.1× increase). The Loss baseline shows negligible length dependence, maintaining AUC ≈ 0.55 across all lengths, confirming that reference comparison is essential regardless of text length. The relative gain of WBC over Ratio—measured as $(AUC_{\text{WBC}} - AUC_{\text{Ratio}})/AUC_{\text{Ratio}}$—increases from 11.4% to 20.8% as length grows, demonstrating that our voting mechanism's statistical power scales with the number of windows. Performance plateaus beyond $L = 384$: extending to 512 tokens improves AUC by only 0.9% (0.833 vs 0.825).

5.3.5 Component Importance Quantification

Table 3 quantifies each component's contribution. The reference model comparison proves essential; without it, AUC drops to 0.569 (near random), confirming that membership signals emerge from differential behavior between fine-tuned and base models rather than absolute confidence. While design choices have modest individual impacts—ensemble aggregation (+0.3%), geometric spacing (+2.6%), sliding windows (+3.9%)—the sign-based scoring proves critical with a 15.8% performance drop when removed. More importantly, these components dramatically affect high-precision detection: sliding windows alone account for 51.2% of TPR@1%FPR (0.0715 vs 0.1464), demonstrating that maximizing measurement count is crucial for confident membership identification.

Table 3: **Component ablation study on Khan Academy dataset.** Removal of WBC components reveals reference model comparison and sign-based scoring as critical, contributing 32.0% and 15.9% of total performance, respectively.

Configuration	AUC	Δ	TPR@10%	TPR@1%
Full WBC	**0.8369**$_{\pm.0027}$	—	**0.5384**$_{\pm.0076}$	**0.1464**$_{\pm.0088}$
Design choices				
w/o ensemble	0.8341$_{\pm.0026}$	−0.3%	0.5093$_{\pm.0092}$	0.1278$_{\pm.0091}$
w/o geometric	0.8152$_{\pm.0029}$	−2.6%	0.5023$_{\pm.0088}$	0.1373$_{\pm.0080}$
w/o sliding	0.8042$_{\pm.0030}$	−3.9%	0.4195$_{\pm.0062}$	0.0715$_{\pm.0099}$
Core components				
w/o sign-based	0.7045$_{\pm.0038}$	−15.8%	0.2835$_{\pm.0077}$	0.0508$_{\pm.0046}$
w/o reference	0.5693$_{\pm.0040}$	−31.9%	0.1510$_{\pm.0053}$	0.0175$_{\pm.0022}$

5.4 Generalization Across Model Scales and Architectures

We test its generalizability across two key dimensions: model scale and architectural family. In these experiments, we compare WBC against the representative baseline, Ratio, to provide a consistent reference point and evaluate the relative performance gap in new contexts.

5.4.1 Performance Across Model Scales

The vulnerability to our window-based attack intensifies dramatically with model scale. Figure 7 shows attack performance on the Pythia suite from 160M to 6.9B parameters, revealing that vulnerability to window-based attacks increases dramatically with model scale. While both WBC and Ratio perform near randomly at 160M parameters (AUC ≈ 0.51), their performance diverges sharply as model size grows.

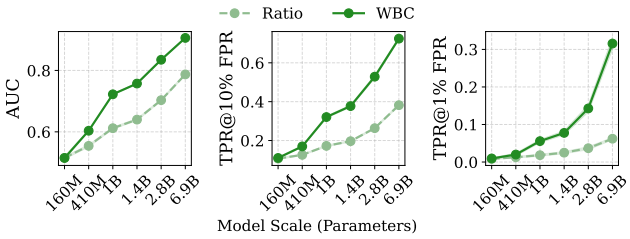

Figure 7: **Performance comparison of WBC and Ratio methods across model scales.** Metrics include AUC, TPR@10% FPR, and TPR@1% FPR. Shaded regions denote standard deviations.

At 2.8B parameters, WBC achieves AUC 0.835 versus

Table 4: **Model performance comparison on Khan Academy dataset.** WBC consistently outperforms the Ratio baseline across all architectures and scales.

Model	Method	Attack Performance			
		AUC	TPR@10%	TPR@1%	TPR@0.1%
GPT-2 [68]	Ratio	$0.505_{\pm.004}$	$0.101_{\pm.003}$	$0.009_{\pm.001}$	$0.001_{\pm.001}$
	WBC	$\mathbf{0.505}_{\pm.003}$	$\mathbf{0.104}_{\pm.003}$	$\mathbf{0.013}_{\pm.002}$	$\mathbf{0.001}_{\pm.001}$
Mamba-1.4B [27]	Ratio	$0.592_{\pm.003}$	$0.154_{\pm.005}$	$0.018_{\pm.002}$	$0.001_{\pm.000}$
	WBC	$\mathbf{0.673}_{\pm.002}$	$\mathbf{0.274}_{\pm.004}$	$\mathbf{0.048}_{\pm.003}$	$\mathbf{0.004}_{\pm.001}$
Pythia-2.8B [5]	Ratio	$0.703_{\pm.003}$	$0.264_{\pm.004}$	$0.037_{\pm.004}$	$0.003_{\pm.001}$
	WBC	$\mathbf{0.833}_{\pm.002}$	$\mathbf{0.529}_{\pm.009}$	$\mathbf{0.135}_{\pm.008}$	$\mathbf{0.026}_{\pm.009}$
Llama-3.2-3B [21]	Ratio	$0.861_{\pm.003}$	$0.551_{\pm.010}$	$0.135_{\pm.014}$	$0.019_{\pm.006}$
	WBC	$\mathbf{0.942}_{\pm.002}$	$\mathbf{0.831}_{\pm.006}$	$\mathbf{0.467}_{\pm.015}$	$\mathbf{0.193}_{\pm.028}$
GPT-J-6B [78]	Ratio	$0.868_{\pm.002}$	$0.564_{\pm.009}$	$0.126_{\pm.007}$	$0.013_{\pm.005}$
	WBC	$\mathbf{0.961}_{\pm.001}$	$\mathbf{0.887}_{\pm.003}$	$\mathbf{0.566}_{\pm.016}$	$\mathbf{0.209}_{\pm.043}$

Ratio's 0.703. The advantage is most pronounced in high-confidence detection: TPR@1%FPR reaches 14.3% for WBC versus 3.7% for Ratio, nearly a 4× improvement. This widening gap reflects the fact that larger models possess greater memorization capacity, thereby exposing increased vulnerability to MIA. While global averaging dilutes these sharp local patterns, causing Ratio's effectiveness to plateau, WBC's window-based detection captures them directly, with performance scaling in tandem with model capacity.

5.4.2 Performance Across Model Architectures

To ensure our findings are not specific to the Pythia architecture, Table 4 demonstrates WBC's generalizability across diverse architectures on the Khan Academy dataset. On large-scale transformers, WBC substantially outperforms Ratio: GPT-J-6B achieves AUC 0.961 versus 0.868, with TPR@1%FPR of 56.6% versus 12.6% (4.5× improvement); Llama-3.2-3B shows similar gains with TPR@1%FPR of 46.7% versus 13.5% (3.4× improvement). The advantage extends beyond transformers—on the state-space model Mamba-1.4B, WBC maintains superiority (AUC 0.673 vs. 0.592), confirming that local signal aggregation is architecturally agnostic. Only GPT-2 shows near-random performance for both methods, consistent with minimal memorization in smaller models on complex datasets. These results establish that window-based detection exploits fundamental memorization patterns that transcend specific architectural choices.

5.5 Robustness to Misaligned References

We simulate a scenario where the adversary does not have access to the exact base model used for fine-tuning. We fix the target model as Pythia-2.8B and evaluate performance using distinct reference models with size mismatch (Pythia-160m, Pythia-1.4B, and Pythia-6.9B) and architecture mismatch (GPT-J-6B).

Table 5: **Impact of Reference Model Mismatch.** We attack a Pythia-2.8B target using reference models with different sizes and different architectures.

Ref. Model	Method	Khan Academy (Target: Pythia-2.8B)			
		AUC	TPR@10%	TPR@1%	TPR@0.1%
Pythia-160m	Ratio	$0.656_{\pm.003}$	$0.205_{\pm.005}$	$0.029_{\pm.003}$	$0.001_{\pm.001}$
	WBC	$\mathbf{0.692}_{\pm.002}$	$\mathbf{0.294}_{\pm.005}$	$\mathbf{0.062}_{\pm.005}$	$\mathbf{0.011}_{\pm.004}$
Pythia-1.4B	Ratio	$0.692_{\pm.003}$	$0.245_{\pm.007}$	$0.034_{\pm.003}$	$0.003_{\pm.001}$
	WBC	$\mathbf{0.774}_{\pm.002}$	$\mathbf{0.407}_{\pm.008}$	$\mathbf{0.077}_{\pm.004}$	$\mathbf{0.016}_{\pm.006}$
Pythia-6.9B	Ratio	$0.695_{\pm.002}$	$0.255_{\pm.003}$	$0.037_{\pm.002}$	$0.002_{\pm.001}$
	WBC	$\mathbf{0.769}_{\pm.002}$	$\mathbf{0.369}_{\pm.008}$	$\mathbf{0.071}_{\pm.006}$	$\mathbf{0.014}_{\pm.004}$
GPT-J-6B	Ratio	$\mathbf{0.685}_{\pm.003}$	$0.256_{\pm.003}$	$0.038_{\pm.003}$	$0.002_{\pm.001}$
	WBC	$0.673_{\pm.003}$	$\mathbf{0.269}_{\pm.006}$	$\mathbf{0.044}_{\pm.003}$	$\mathbf{0.003}_{\pm.002}$

Table 6: **Attack performance under differential privacy on Khan Academy dataset.** Evaluation at privacy budgets $\varepsilon \in \{1, 4, 8, \infty\}$ with fixed $\delta = 10^{-5}$.

ε	Method	Attack Performance				PPL
		AUC	TPR@10%	TPR@1%	TPR@0.1%	
∞	Ratio	$0.703_{\pm.003}$	$0.264_{\pm.004}$	$0.037_{\pm.004}$	$0.003_{\pm.001}$	3.49
	WBC	$\mathbf{0.837}_{\pm.003}$	$\mathbf{0.538}_{\pm.008}$	$\mathbf{0.146}_{\pm.008}$	$\mathbf{0.026}_{\pm.009}$	
8	Ratio	$0.642_{\pm.004}$	$0.198_{\pm.006}$	$0.024_{\pm.003}$	$0.002_{\pm.001}$	3.85
	WBC	$\mathbf{0.751}_{\pm.003}$	$\mathbf{0.358}_{\pm.008}$	$\mathbf{0.078}_{\pm.006}$	$\mathbf{0.013}_{\pm.003}$	
4	Ratio	$0.591_{\pm.005}$	$0.147_{\pm.005}$	$0.015_{\pm.002}$	$0.001_{\pm.000}$	4.68
	WBC	$\mathbf{0.674}_{\pm.004}$	$\mathbf{0.254}_{\pm.007}$	$\mathbf{0.042}_{\pm.005}$	$\mathbf{0.006}_{\pm.002}$	
1	Ratio	$0.524_{\pm.006}$	$0.108_{\pm.004}$	$0.008_{\pm.002}$	$0.000_{\pm.000}$	4.77
	WBC	$\mathbf{0.561}_{\pm.005}$	$\mathbf{0.135}_{\pm.005}$	$\mathbf{0.019}_{\pm.003}$	$\mathbf{0.001}_{\pm.001}$	

The results are summarized in Table 5. When using Pythia-1.4B (a model half the size of the target) as a reference, WBC achieves an AUC of 0.774 and a TPR of 7.7% at 1% FPR, significantly outperforming the Ratio baseline (AUC 0.692, TPR 3.4%). With a different architecture (GPT-J), WBC maintains a performance advantage in the critical low false-positive regime, though it trails slightly in overall AUC (0.673 vs 0.685). This AUC deficit likely stems from tokenization and architectural disparities, introducing noise into local comparisons that global averaging smooths out.

5.6 Defense Evaluation

Given that WBC is highly effective, it is important to assess whether privacy-preserving training techniques can defend against it. We evaluate WBC against three defense mechanisms spanning different protection strategies: differential privacy, parameter-efficient training, and data obfuscation.

5.6.1 Differential Privacy

Table 6 reveals that while DP-SGD reduces absolute attack success, WBC maintains substantial relative advantages across all privacy budgets. At moderate privacy ($\varepsilon = 8$), WBC

Table 7: **Attack performance under LoRA on Khan Academy dataset.** Evaluation with ranks $r \in \{8, 16, 32, 64\}$ and scaling factor $\alpha = 2r$.

r	Method	Attack Performance				Trn. %	PPL
		AUC	TPR@10%	TPR@1%	TPR@0.1%		
∞	Ratio	$0.703_{\pm.003}$	$0.264_{\pm.004}$	$0.037_{\pm.004}$	$0.003_{\pm.001}$	100%	3.49
	WBC	$\mathbf{0.837}_{\pm.003}$	$\mathbf{0.538}_{\pm.008}$	$\mathbf{0.146}_{\pm.008}$	$\mathbf{0.026}_{\pm.009}$		
64	Ratio	$0.615_{\pm.003}$	$0.186_{\pm.005}$	$0.022_{\pm.002}$	$0.002_{\pm.001}$	0.75%	3.77
	WBC	$\mathbf{0.748}_{\pm.002}$	$\mathbf{0.378}_{\pm.007}$	$\mathbf{0.074}_{\pm.005}$	$\mathbf{0.009}_{\pm.002}$		
32	Ratio	$0.592_{\pm.003}$	$0.164_{\pm.004}$	$0.018_{\pm.002}$	$0.001_{\pm.000}$	0.38%	3.82
	WBC	$\mathbf{0.698}_{\pm.002}$	$\mathbf{0.312}_{\pm.006}$	$\mathbf{0.048}_{\pm.004}$	$\mathbf{0.005}_{\pm.001}$		
16	Ratio	$0.564_{\pm.003}$	$0.142_{\pm.004}$	$0.014_{\pm.001}$	$0.001_{\pm.000}$	0.19%	3.88
	WBC	$\mathbf{0.634}_{\pm.002}$	$\mathbf{0.234}_{\pm.005}$	$\mathbf{0.029}_{\pm.003}$	$\mathbf{0.003}_{\pm.001}$		
8	Ratio	$0.541_{\pm.004}$	$0.124_{\pm.003}$	$0.011_{\pm.002}$	$0.000_{\pm.000}$	0.09%	3.93
	WBC	$\mathbf{0.578}_{\pm.003}$	$\mathbf{0.171}_{\pm.004}$	$\mathbf{0.018}_{\pm.002}$	$\mathbf{0.002}_{\pm.001}$		

achieves AUC 0.751 versus Ratio's 0.642, with a more pronounced 3.25× advantage in TPR@1%FPR (0.078 vs 0.024). This gap persists at strong privacy ($\varepsilon = 1$): while both methods approach random guessing, WBC still achieves 2.4× higher TPR@1%FPR. While global noise reduces overall signal strength, local coherence within text windows remains partially intact. Notably, stronger privacy guarantees ($\varepsilon = 1$) provide limited additional protection—AUC decreases from $\varepsilon = 8$ (0.751 to 0.561) while perplexity increases by 37%. This modest utility cost makes DP-SGD more practical than previously thought, though window-based attacks remain partially effective even under strong privacy guarantees.

5.6.2 Low-Rank Adaptation

LoRA [31], while designed for parameter efficiency, provides unintended privacy benefits by constraining memorization capacity [2, 46, 88]. The defense mechanism stems from capacity constraints: low-rank factorizations $W = W_0 + BA$ with $r \ll \min(d,k)$ force the model to compress information into a rank-r subspace, favoring generalizable patterns over sample-specific memorization. We note that this protection relies on accepting a slight utility decrease; recent work indicates LoRA is not a robust defense under strict utility matching [69]. While both attacks degrade under LoRA, WBC maintains a consistent advantage—2.67× higher TPR@1%FPR at rank 32 (0.048 vs 0.018) compared to 3.95× for full fine-tuning. This persistence suggests that even constrained memorization retains localized patterns that window-based detection exploits. LoRA thus offers meaningful protection with minimal utility cost, though sophisticated attacks targeting local signals remain partially effective.

5.6.3 Selective data Obfuscation in LLM Fine-Tuning

SOFT [88] defends against membership inference by selectively paraphrasing influential training samples identified through loss-based thresholds. Table 8 shows SOFT's effectiveness: it reduces both WBC and Ratio from strong detection (AUC 0.837 and 0.703, respectively) to near-random performance (AUC ≈ 0.494). At TPR@1%FPR, WBC's advantage over Ratio shrinks from 3.9× to marginal (0.012 vs 0.010), a 92% reduction in detection capability.

Table 8: **Attack performance against SOFT defense on Khan Academy dataset.** SOFT configured with selection ratio $\alpha = 0.3$ and paraphrase ratio $\beta = 0.5$.

Defense	Method	Attack Performance				PPL
		AUC	TPR@10%	TPR@1%	TPR@0.1%	
Baseline	Ratio	$0.703_{\pm.003}$	$0.264_{\pm.004}$	$0.037_{\pm.004}$	$0.003_{\pm.001}$	3.49
	WBC	$\mathbf{0.837}_{\pm.003}$	$\mathbf{0.538}_{\pm.008}$	$\mathbf{0.146}_{\pm.008}$	$\mathbf{0.026}_{\pm.009}$	
SOFT	Ratio	$0.494_{\pm.003}$	$0.100_{\pm.004}$	$0.010_{\pm.001}$	$0.001_{\pm.000}$	3.48
	WBC	$\mathbf{0.494}_{\pm.004}$	$\mathbf{0.100}_{\pm.003}$	$\mathbf{0.012}_{\pm.002}$	$\mathbf{0.001}_{\pm.000}$	

6 Conclusion

We demonstrated that membership inference attacks against fine-tuned LLMs fail as global averaging is fundamentally unsuitable for detecting sparse, extremal memorization events. Instead, membership signals manifest as rare, localized patterns that are masked by long-tailed noise from domain adaptation. We thus presented WBC, a membership inference attack that replaces global averaging with localized window-based analysis. The method slides windows of varying sizes across token sequences, computes binary comparisons between reference and target model losses, and aggregates these local votes using sign statistics, achieving 2-3× higher detection rates than existing methods across eleven datasets. While our empirical results demonstrate the effectiveness of the sign test, the shift from global to localized analysis opens a rich design space. We note that WBC represents one effective instance of this paradigm; given the sparse, spike-like structure of membership signals we identified, future work could explore alternative aggregation functions or statistical modeling techniques to further enhance detection.

Acknowledgments. This work was funded in part by the National Science Foundation (NSF) awards, CNS-2212160, CNS-2504819, CNS-2247794, and CNS-2207204, Amazon Research Award, and CISCO Research Award. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

7 Ethical Considerations

We conducted a comprehensive stakeholder-based ethics analysis following the Menlo Report principles. Our research develops a membership inference attack that achieves 2–3×

higher detection rates than existing methods, raising important privacy considerations that we carefully evaluated throughout the research process.

Stakeholders and Impacts. We identified three primary stakeholder groups: ML practitioners and organizations deploying fine-tuned LLMs who may face increased privacy risks from our more effective attack; individuals whose data appears in training datasets and may be exposed through successful membership inference; and the broader research community working to understand and mitigate privacy risks in machine learning systems. For practitioners, our attack demonstrates that current assumptions about membership inference difficulty significantly underestimate actual risks, particularly for localized memorization patterns. This knowledge enables more informed decisions about data handling and privacy-preserving training techniques. For individuals, while our attack could theoretically enable adversaries to determine if their data was used in training, we note that our experiments used only publicly available datasets and synthetic data, avoiding any direct privacy violations. The research community benefits from our theoretical insights into why localized analysis outperforms global averaging, enabling the development of more targeted defenses.

We implemented several measures to minimize potential harm. First, all experiments used publicly available models and datasets, eliminating risks to proprietary systems or private data. We specifically chose datasets like Cosmopedia (synthetic data), WikiText-103, and other public corpora where membership disclosure poses minimal privacy risk. Second, we evaluated multiple defense mechanisms, including differential privacy, SOFT, and LoRA, providing concrete guidance for practitioners seeking to protect their systems. Our results show that while these defenses reduce attack effectiveness, they do not eliminate the vulnerability, highlighting the need for continued research into privacy-preserving training methods. Third, we responsibly disclose the limitations of our attack, including its requirement for score-based access to both target and reference models, making it impractical for many real-world scenarios without legitimate access.

Justification. We proceeded with publication after determining that the benefits outweigh the potential harms. Our attack serves as a diagnostic tool for quantifying existing privacy risks—it requires only score-based black-box access that legitimate users already possess, introducing no new attack surfaces. By revealing that localized memorization is more detectable than previously believed and simultaneously evaluating defenses, we enable practitioners to make informed privacy decisions rather than relying on false confidence in inadequate measures. Publication through peer review at a defensive security venue ensures our findings reach those developing countermeasures rather than malicious actors.

8 Open Science

In accordance with USENIX Security's commitment to reproducible research, we make all artifacts necessary for evaluating and reproducing our contributions publicly available. Our open science approach balances reproducibility with responsible disclosure of potentially sensitive attack techniques.

The complete implementation of our Window-Based Comparison attack, including all baseline methods, training and evaluation scripts, is available at https://github.com/Stry233/WBC and archived at https://doi.org/10.5281/zenodo.17968678. The archive includes the core WBC attack implementation with configurable window sizes and aggregation methods, implementations of all thirteen baseline attacks used in our evaluation, scripts for fine-tuning models on the datasets used in our experiments, and evaluation metrics including AUC and TPR at various FPR thresholds. All code is properly documented with clear instructions for setup and execution. The full version of this paper can be found at https://arxiv.org/abs/2601.02751v1.

Our experiments utilize publicly available datasets accessible through standard channels: Cosmopedia [3] subsets via HuggingFace [82], WikiText-103 [51], XSum [56], Amazon Reviews [59], CC-News [28], and Reddit [70] datasets through their respective public repositories. We provide scripts to automatically download and preprocess these datasets into the format required for our experiments. For models, we use the publicly available Pythia suite (160M to 6.9B parameters) [5], GPT-2 [6], GPT-J-6B [78], Llama-3.2-3B [21], and Mamba-1.4B [27], all accessible through HuggingFace [82]. Our repository includes model configuration files and fine-tuning scripts to reproduce the exact target models used in our evaluation. Reproducing our results requires access to GPUs capable of running models up to 6.9B parameters. We provide guidance for scaling experiments to smaller models for researchers with limited computational resources. The complete experimental suite required approximately 500 GPU-hours on NVIDIA A100 GPUs, though individual experiments can be run with fewer resources.

References

[1] Guy Amit, Abigail Goldsteen, and Ariel Farkash. Sok: Reducing the vulnerability of fine-tuned language models to membership inference attacks. *arXiv preprint arXiv:2403.08481*, 2024.

[2] Guy Amit, Abigail Goldsteen, and Ariel Farkash. Sok: Reducing the vulnerability of fine-tuned language models to membership inference attacks, 2024.

[3] Loubna Ben Allal, Anton Lozhkov, Guilherme Penedo, Thomas Wolf, and Leandro von Werra. Cosmopedia, 2024.

[4] Patrice Bertail, Stéphan Clémençcon, and Nicolas Vayatis. On bootstrapping the roc curve. *Advances in Neural Information Processing Systems*, 21, 2008.

[5] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023.

[6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

[7] 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)*, pages 1897–1914, 2022.

[8] 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)*, pages 2633–2650, 2021.

[9] Hongyan Chang, Ali Shahin Shamsabadi, Kleomenis Katevas, Hamed Haddadi, and Reza Shokri. Context-aware membership inference attacks against pre-trained large language models. *arXiv preprint arXiv:2409.13745*, 2024.

[10] Xiaoyi Chen, Siyuan Tang, Rui Zhu, Shijun Yan, Lei Jin, Zihao Wang, Liya Su, Zhikun Zhang, XiaoFeng Wang, and Haixu Tang. The janus interface: How fine-tuning in large language models amplifies the privacy risks. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 1285–1299, 2024.

[11] Christopher A. Choquette-Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. Label-only membership inference attacks, 2021.

[12] Philip J Clark and Francis C Evans. Distance to nearest neighbor as a measure of spatial relationships in populations. *Ecology*, 35(4):445–453, 1954.

[13] Debeshee Das, Jie Zhang, and Florian Tramèr. Blind baselines beat membership inference attacks for foundation models, 2025.

[14] DeepSeek-AI. Deepseek-v3 technical report, 2025.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[16] Peter J Diggle. *Statistical analysis of spatial and spatio-temporal point patterns*. CRC press, 2013.

[17] Nirav Diwan, Tanmoy Chakravorty, and Zubair Shafiq. Fingerprinting fine-tuned language models in the wild, 2021.

[18] David L Donoho and Peter J Huber. The notion of breakdown point. *A festschrift for Erich L. Lehmann*, 157184, 1983.

[19] Yuntao Du, Jiacheng Li, Yuetian Chen, Kaiyuan Zhang, Zhizhen Yuan, Hanshen Xiao, Bruno Ribeiro, and Ninghui Li. Cascading and Proxy Membership Inference Attacks. In *33th Annual Network and Distributed System Security Symposium (NDSS)*, 2026.

[20] 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?, 2024.

[21] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024.

[22] Paul Embrechts, Thomas Mikosch, and Claudia Klüppelberg. Modelling extremal events: for insurance and finance, 1997.

[23] 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, 2025.

[24] Wenjie Fu, Huandong Wang, Chen Gao, Guanghua Liu, Yong Li, and Tao Jiang. Practical membership inference attacks against fine-tuned large language models via self-prompt calibration, 2024.

[25] Wenjie Fu, Huandong Wang, Chen Gao, Guanghua Liu, Yong Li, and Tao Jiang. Mia-tuner: adapting large language models as pre-training text detector. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 27295–27303, 2025.

[26] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[27] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[28] Felix Hamborg, Norman Meuschke, Corinna Breitinger, and Bela Gipp. news-please: A generic news crawler and extractor. In *Proceedings of the 15th International Symposium of Information Science*, pages 218–223, March 2017.

[29] Yu He, Boheng Li, Liu Liu, Zhongjie Ba, Wei Dong, Yiming Li, Zhan Qin, Kui Ren, and Chun Chen. Towards label-only membership inference attack against pre-trained large language models, 2025.

[30] Sorami Hisamoto, Matt Post, and Kevin Duh. Membership inference attacks on sequence-to-sequence models: Is my data in your machine translation system? *Transactions of the Association for Computational Linguistics*, 8:49–63, December 2020.

[31] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[32] Yuke Hu, Zheng Li, Zhihao Liu, Yang Zhang, Zhan Qin, Kui Ren, and Chun Chen. Membership inference attacks against vision-language models, 2025.

[33] Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. Are large pre-trained language models leaking your personal information?, 2022.

[34] Zhiheng Huang, Yannan Liu, Daojing He, and Yu Li. Dfmia: A distribution-free membership inference attack on fine-tuned large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(1):343–351, Apr. 2025.

[35] Peter J Huber. Robust statistics. In *International encyclopedia of statistical science*, pages 1248–1251. Springer, 2011.

[36] Huseyin A. Inan, Osman Ramadan, Lukas Wutschitz, Daniel Jones, Victor Rühle, James Withers, and Robert Sim. Training data leakage analysis in language models, 2021.

[37] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[38] Nikhil Kandpal, Krishna Pillutla, Alina Oprea, Peter Kairouz, Christopher A. Choquette-Choo, and Zheng Xu. User inference attacks on large language models, 2024.

[39] Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. Copyright violations and large language models. *arXiv preprint arXiv:2310.13771*, 2023.

[40] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

[41] M Ross Leadbetter, Georg Lindgren, and Holger Rootzén. Extremes and related properties of random sequences and processes. *Springer Series in Statistics*, 1983.

[42] Erich Leo Lehmann and Joseph P Romano. *Testing statistical hypotheses*. Springer, 2005.

[43] Hao Li, Zheng Li, Siyuan Wu, Chengrui Hu, Yutong Ye, Min Zhang, Dengguo Feng, and Yang Zhang. Seqmia: Sequential-metric based membership inference attack, 2024.

[44] Haoran Li, Yulin Chen, Jinglong Luo, Jiecong Wang, Hao Peng, Yan Kang, Xiaojin Zhang, Qi Hu, Chunkit Chan, Zenglin Xu, Bryan Hooi, and Yangqiu Song. Privacy in large language models: Attacks, defenses and future directions, 2024.

[45] Yiyong Liu, Zhengyu Zhao, Michael Backes, and Yang Zhang. Membership inference attacks by exploiting loss trajectory, 2022.

[46] Zihao Luo, Xilie Xu, Feng Liu, Yun Sing Koh, Di Wang, and Jingfeng Zhang. Privacy-preserving low-rank adaptation against membership inference attacks for latent diffusion models, 2024.

[47] Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. Membership inference attacks against language models via neighbourhood comparison, 2023.

[48] Matthieu Meeus, Shubham Jain, Marek Rei, and Yves-Alexandre de Montjoye. Did the neurons read your book? document-level membership inference for large

language models. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 2369–2385, 2024.

[49] Matthieu Meeus, Igor Shilov, Shubham Jain, Manuel Faysse, Marek Rei, and Yves-Alexandre de Montjoye. Sok: Membership inference attacks on llms are rushing nowhere (and how to fix it), 2025.

[50] Wenlong Meng, Zhenyuan Guo, Lenan Wu, Chen Gong, Wenyan Liu, Weixian Li, Chengkun Wei, and Wenzhi Chen. Rr: Unveiling llm training privacy through recollection and ranking. *arXiv preprint arXiv:2502.12658*, 2025.

[51] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

[52] Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. Quantifying privacy risks of masked language models using membership inference attacks. *arXiv preprint arXiv:2203.03929*, 2022.

[53] Fatemehsadat Mireshghallah, Archit Uniyal, Tianhao Wang, David Evans, and Taylor Berg-Kirkpatrick. An empirical analysis of memorization in fine-tuned autoregressive language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1816–1826, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[54] Fatemehsadat Mireshghallah, Archit Uniyal, Tianhao Wang, David Evans, and Taylor Berg-Kirkpatrick. Memorization in nlp fine-tuning methods. *arXiv preprint arXiv:2205.12506*, 2022.

[55] Hamid Mozaffari and Virendra J. Marathe. Semantic membership inference attack against large language models, 2024.

[56] Shashi Narayan, Shay B Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*, 2018.

[57] Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A Feder Cooper, Daphne Ippolito, Christopher A Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*, 2023.

[58] Joseph I. Naus. Approximations for distributions of scan statistics. *Journal of the American Statistical Association*, 77(377):177–181, 1982.

[59] Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 188–197, 2019.

[60] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025.

[61] Alan V Oppenheim. *Discrete-time signal processing*. Pearson Education India, 1999.

[62] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019.

[63] Edwin JG Pitman. Notes on non-parametric statistical inference. Technical report, North Carolina State University. Dept. of Statistics, 1949.

[64] Haritz Puerto, Martin Gubri, Sangdoo Yun, and Seong Joon Oh. Scaling up membership inference: When and how attacks succeed on large language models, 2025.

[65] Qwen. Qwen2.5 technical report, 2025.

[66] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[67] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[68] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[69] Delong Ran, Xinlei He, Tianshuo Cong, Anyu Wang, Qi Li, and Xiaoyun Wang. Lora-leak: Membership inference attacks against lora fine-tuned language models, 2025.

[70] Nils Reimers and Iryna Gurevych. sentence-transformers/reddit-title-body: Reddit title-body dataset. https://huggingface.co/datasets/sentence-transformers/reddit-title-body, 2021. Accessed: 2025-07-21.

[71] Brian D Ripley. The second-order analysis of stationary point processes. *Journal of applied probability*, 13(2):255–266, 1976.

[72] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pages 5558–5567. PMLR, 2019.

[73] Ali Satvaty, Suzan Verberne, and Fatih Turkmen. Undesirable memorization in large language models: A survey, 2025.

[74] Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models, 2024.

[75] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models, 2017.

[76] Meta Llama Team. The llama 3 herd of models, 2024.

[77] Aad W Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

[78] Ben Wang and Aran Komatsuzaki. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

[79] Cheng Wang, Yiwei Wang, Bryan Hooi, Yujun Cai, Nanyun Peng, and Kai-Wei Chang. Con-recall: Detecting pre-training data in llms via contrastive decoding, 2025.

[80] Lauren Watson, Chuan Guo, Graham Cormode, and Alex Sablayrolles. On the importance of difficulty calibration in membership inference attacks. *arXiv preprint arXiv:2111.08440*, 2021.

[81] Rui Wen, Zheng Li, Michael Backes, and Yang Zhang. Membership inference attacks against in-context learning, 2024.

[82] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[83] Roy Xie, Junlin Wang, Ruomin Huang, Minxing Zhang, Rong Ge, Jian Pei, Neil Zhenqiang Gong, and Bhuwan Dhingra. Recall: Membership inference via relative conditional log-likelihoods. *arXiv preprint arXiv:2406.15968*, 2024.

[84] Jiayuan Ye, Aadyaa Maddi, Sasi Kumar Murakonda, Vincent Bindschaedler, and Reza Shokri. Enhanced membership inference attacks against machine learning models. In *Proceedings of the 2022 ACM SIGSAC conference on computer and communications security*, pages 3093–3106, 2022.

[85] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting, 2018.

[86] Shenglai Zeng, Yaxin Li, Jie Ren, Yiding Liu, Han Xu, Pengfei He, Yue Xing, Shuaiqiang Wang, Jiliang Tang, and Dawei Yin. Exploring memorization in fine-tuned language models, 2024.

[87] Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. Min-k

[88] Kaiyuan Zhang, Siyuan Cheng, Hanxi Guo, Yuetian Chen, Zian Su, Shengwei An, Yuntao Du, Charles Fleming, Ashish Kundu, Xiangyu Zhang, and Ninghui Li. Soft: Selective data obfuscation for protecting llm fine-tuning against membership inference attacks, 2025.

[89] Ruisi Zhang, Seira Hidano, and Farinaz Koushanfar. Text revealer: Private text reconstruction via model inversion attacks against transformers. *arXiv preprint arXiv:2209.10505*, 2022.

[90] Weichao Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. Pretraining data detection for large language models: A divergence-based calibration method. *arXiv preprint arXiv:2409.14781*, 2024.

A Optimal Window Size Analysis

To formalize the challenge of selecting an optimal window size w, we analyze the sign test's detection capability. For a window of size w, let $p_w^{(1)}$ denote the probability that $S_i(w) > 0$ for members and $p_w^{(0)}$ for non-members. Under our model, $p_w^{(0)} = 0.5$ due to symmetric noise distribution, while:

$$p_w^{(1)} = \Phi\left(\frac{\rho_\delta w\bar{\gamma}}{\sqrt{w\sigma^2 + \rho_\xi w\mathbb{E}[Y^2]}}\right), \tag{14}$$

where Φ is the standard normal CDF. The separation $p_w^{(1)} - 0.5$ determines the sign test's discriminative power.

The sign statistic $T_{\text{sign}}(w)$ counts the fraction of windows with positive sums. For non-members, $\mathbb{E}[T_{\text{sign}}(w)] = 0.5$, while for members, $\mathbb{E}[T_{\text{sign}}(w)] = p_w^{(1)}$. The detectability depends on both this separation and the variance of $T_{\text{sign}}(w)$.

Due to window overlap, adjacent windows exhibit strong positive correlation:

$$\text{Var}[T_{\text{sign}}(w)] = \frac{1}{(n-w+1)^2} \sum_{i,j} \text{Cov}[\mathbb{I}[S_i(w) > 0], \mathbb{I}[S_j(w) > 0]]. \tag{15}$$

The covariance structure depends on window overlap: windows i and j share $\max(0, w - |i-j|)$ tokens. For $|i-j| \geq w$, the windows are disjoint and approximately independent. Since each group of w consecutive windows contains at most one independent observation, the effective sample size is approximately n/w. For a proportion estimator with effective sample size $n_{\text{eff}} \approx n/w$:

$$\text{Var}[T_{\text{sign}}(w)] \approx \frac{p_w^{(1)}(1-p_w^{(1)})}{n/w} = \frac{w \cdot p_w^{(1)}(1-p_w^{(1)})}{n} \tag{16}$$

The power of the sign test thus becomes:

$$\text{Power}(w) \propto \frac{(p_w^{(1)} - 0.5)^2}{\text{Var}[T_{\text{sign}}(w)]} \approx \frac{(p_w^{(1)} - 0.5)^2 \cdot n}{w \cdot p_w^{(1)}(1-p_w^{(1)})}. \tag{17}$$

This reveals the fundamental trade-off: the numerator $(p_w^{(1)} - 0.5)^2$ increases with w as more tokens improve signal accumulation, but the denominator contains w, reflecting the loss of independent tests. Even if doubling w increases $(p_w^{(1)} - 0.5)$ by 40%, the power may still decrease by 30% due to the linear penalty from reduced effective sample size.

The optimal window size w^* that maximizes this power depends critically on the parameters ρ_δ, ρ_ξ, $\bar{\gamma}$, and $\mathbb{E}[Y^2]$. These parameters are unknown and vary across datasets and even within documents. Attempting to estimate them creates a circular dependency: identifying signal-containing windows requires knowing the parameters, but parameter estimation requires identifying these windows. This makes finding a single optimal window size infeasible in practice, motivating our ensemble approach.

B Supplementary Results

B.1 Utility of Fine-tuned LLMs

To verify that fine-tuning maintains model utility beyond memorization, we evaluate perplexity on both member (train) and non-member (test) sets. Table 9 presents perplexity scores for Pythia-2.8B before and after fine-tuning across all datasets. Lower perplexity indicates better language modeling capability. Fine-tuning consistently improves perplexity on both member and non-member data, demonstrating genuine learning rather than mere memorization.

Table 9: Perplexity scores for Pythia-2.8B before (pretrained) and after fine-tuning.

Dataset	Pretrained		Fine-tuned	
	Member	Non-member	Member	Non-member
WikiText-103	10.383	10.359	8.823	8.840
XSum	9.805	9.711	9.171	9.232
Amazon Reviews	18.656	18.359	14.626	14.907
CC-News	9.922	9.867	9.538	9.612
Reddit	13.094	13.016	11.874	12.074
Khan Academy	4.772	4.750	3.490	3.601
Stanford	6.824	6.810	4.693	4.873
Stories	9.636	9.632	5.981	6.345
Web Samples v2	8.397	8.418	5.710	6.021
AutoMathText	6.030	6.026	4.328	4.516
wikiHow	6.799	6.762	4.514	4.723

B.2 Extended Main Results

Table 10 presents comprehensive results on five real-world document benchmarks: WikiText-103, XSum, Amazon Reviews, CC News, and Reddit. These datasets complement the Cosmopedia results in the main text and demonstrate the generalizability of WBC across diverse text domains. The strong performance across both synthetic and real-world datasets validates that localized signal aggregation captures fundamental memorization patterns that transcend dataset characteristics, establishing WBC as a robust and generalizable approach for membership inference against fine-tuned LLMs.

B.3 Extended Ensemble Composition Analysis

Table 11 validates the robustness of geometric window spacing across three diverse datasets. The *Full Ensemble* yields top-tier results in 7 of 9 metric combinations. Notably, it improves TPR@1%FPR by 19.7% over the single best window on WikiHow (0.0967 vs 0.0808), with significant gains also observed on Stanford (+10.9%) and Stories (+8.9%).

Restricting analysis to *Large Range* windows causes AUC degradation exceeding 5% in all cases, confirming that memorization manifests primarily as localized, small-scale patterns. The consistent performance hierarchy — *Full Ensemble* $\approx$ *Small Range* $\gg$ *Large Range* — demonstrates that geometric spacing captures fundamental memorization properties independent of domain. This universality establishes the Full Ensemble as the optimal deployment strategy, offering robust detection without dataset-specific hyperparameter tuning.

Table 10: MIA performance (AUC, TPR@10%FPR, TPR@1%FPR, TPR@0.1%FPR) across different datasets.

MIAs	WikiText-103				XSum				Amazon Reviews			
	AUC	T@10%	T@1%	T@0.1%	AUC	T@10%	T@1%	T@0.1%	AUC	T@10%	T@1%	T@0.1%
Loss [85]	$0.532_{\pm.006}$	$0.122_{\pm.005}$	$0.013_{\pm.002}$	$0.001_{\pm.001}$	$0.531_{\pm.003}$	$0.120_{\pm.005}$	$0.012_{\pm.002}$	$0.001_{\pm.001}$	$0.523_{\pm.004}$	$0.110_{\pm.004}$	$0.010_{\pm.001}$	$0.001_{\pm.000}$
ZLIB [8]	$0.531_{\pm.005}$	$0.129_{\pm.005}$	$0.012_{\pm.002}$	$0.001_{\pm.001}$	$0.532_{\pm.004}$	$0.123_{\pm.006}$	$0.010_{\pm.001}$	$0.001_{\pm.001}$	$0.523_{\pm.003}$	$0.113_{\pm.004}$	$0.009_{\pm.001}$	$0.001_{\pm.001}$
Lowercase [8]	$0.533_{\pm.003}$	$0.122_{\pm.006}$	$0.015_{\pm.003}$	$0.002_{\pm.001}$	$0.524_{\pm.002}$	$0.113_{\pm.004}$	$0.014_{\pm.002}$	$0.002_{\pm.000}$	$0.529_{\pm.004}$	$0.119_{\pm.004}$	$0.014_{\pm.002}$	$0.002_{\pm.001}$
Min-K% [74]	$0.537_{\pm.005}$	$0.116_{\pm.005}$	$0.015_{\pm.002}$	$0.001_{\pm.001}$	$0.533_{\pm.003}$	$0.131_{\pm.006}$	$0.014_{\pm.001}$	$0.001_{\pm.001}$	$0.521_{\pm.003}$	$0.110_{\pm.004}$	$0.012_{\pm.002}$	$0.001_{\pm.000}$
Min-K%++ [87]	$0.538_{\pm.005}$	$0.114_{\pm.006}$	$0.016_{\pm.003}$	$0.001_{\pm.001}$	$0.534_{\pm.004}$	$0.128_{\pm.004}$	$0.014_{\pm.002}$	$0.001_{\pm.000}$	$0.523_{\pm.002}$	$0.114_{\pm.003}$	$0.011_{\pm.002}$	$0.001_{\pm.000}$
BoWs [13]	$0.502_{\pm.005}$	$0.099_{\pm.006}$	$0.010_{\pm.001}$	$0.001_{\pm.001}$	$0.505_{\pm.005}$	$0.098_{\pm.004}$	$0.008_{\pm.001}$	$0.001_{\pm.001}$	$0.492_{\pm.003}$	$0.090_{\pm.003}$	$0.008_{\pm.001}$	$0.001_{\pm.000}$
ReCall [83]	$0.532_{\pm.006}$	$0.122_{\pm.005}$	$0.013_{\pm.002}$	$0.001_{\pm.001}$	$0.531_{\pm.003}$	$0.120_{\pm.005}$	$0.012_{\pm.002}$	$0.001_{\pm.001}$	$0.523_{\pm.004}$	$0.110_{\pm.004}$	$0.010_{\pm.001}$	$0.001_{\pm.000}$
CON-Recall [79]	$0.527_{\pm.003}$	$0.121_{\pm.004}$	$0.011_{\pm.002}$	$0.001_{\pm.001}$	$0.521_{\pm.003}$	$0.113_{\pm.004}$	$0.012_{\pm.002}$	$0.002_{\pm.001}$	$0.514_{\pm.003}$	$0.111_{\pm.004}$	$0.009_{\pm.002}$	$0.001_{\pm.001}$
DC-PDD [90]	$0.531_{\pm.006}$	$0.116_{\pm.004}$	$0.014_{\pm.001}$	$0.001_{\pm.001}$	$0.527_{\pm.004}$	$0.124_{\pm.004}$	$0.016_{\pm.002}$	$0.001_{\pm.001}$	$0.517_{\pm.004}$	$0.105_{\pm.005}$	$0.013_{\pm.001}$	$0.001_{\pm.001}$
SPV-MIA [24]	$0.592_{\pm.004}$	$0.176_{\pm.006}$	$0.028_{\pm.003}$	$0.002_{\pm.001}$	$0.789_{\pm.004}$	$0.345_{\pm.012}$	$0.020_{\pm.002}$	$0.003_{\pm.001}$	$0.812_{\pm.003}$	$0.472_{\pm.010}$	$0.085_{\pm.005}$	$0.003_{\pm.001}$
Ratio [80]	$0.580_{\pm.004}$	$0.152_{\pm.004}$	$0.013_{\pm.002}$	$0.001_{\pm.001}$	$0.783_{\pm.003}$	$0.336_{\pm.016}$	$0.015_{\pm.002}$	$0.002_{\pm.001}$	$0.799_{\pm.002}$	$0.351_{\pm.008}$	$0.014_{\pm.001}$	$0.001_{\pm.000}$
Difference [80]	$0.591_{\pm.002}$	$0.169_{\pm.004}$	$0.014_{\pm.003}$	$0.001_{\pm.001}$	$0.796_{\pm.003}$	$0.335_{\pm.015}$	$0.018_{\pm.002}$	$0.005_{\pm.001}$	$0.804_{\pm.002}$	$0.372_{\pm.014}$	$0.015_{\pm.004}$	$0.001_{\pm.000}$
Ensemble [88]	$0.599_{\pm.003}$	$0.187_{\pm.005}$	$\mathbf{0.035_{\pm.002}}$	$0.003_{\pm.001}$	$0.799_{\pm.003}$	$0.497_{\pm.002}$	$0.008_{\pm.003}$	$0.009_{\pm.003}$	$0.819_{\pm.002}$	$0.497_{\pm.014}$	$0.067_{\pm.001}$	$0.000_{\pm.000}$
WBC (Ours)	$\mathbf{0.784_{\pm.003}}$	$\mathbf{0.420_{\pm.007}}$	$0.028_{\pm.005}$	$\mathbf{0.004_{\pm.001}}$	$\mathbf{0.903_{\pm.002}}$	$\mathbf{0.729_{\pm.007}}$	$\mathbf{0.137_{\pm.025}}$	$\mathbf{0.019_{\pm.002}}$	$\mathbf{0.901_{\pm.002}}$	$\mathbf{0.715_{\pm.005}}$	$\mathbf{0.163_{\pm.018}}$	$\mathbf{0.017_{\pm.005}}$

MIAs	CC News				Reddit			
	AUC	T@10%	T@1%	T@0.1%	AUC	T@10%	T@1%	T@0.1%
Loss [85]	$0.517_{\pm.004}$	$0.109_{\pm.005}$	$0.011_{\pm.002}$	$0.001_{\pm.001}$	$0.510_{\pm.004}$	$0.109_{\pm.006}$	$0.014_{\pm.002}$	$0.001_{\pm.000}$
ZLIB [8]	$0.518_{\pm.004}$	$0.114_{\pm.003}$	$0.010_{\pm.001}$	$0.002_{\pm.001}$	$0.514_{\pm.004}$	$0.110_{\pm.005}$	$0.013_{\pm.001}$	$0.001_{\pm.000}$
Lowercase [8]	$0.522_{\pm.004}$	$0.113_{\pm.006}$	$0.012_{\pm.002}$	$0.002_{\pm.001}$	$0.514_{\pm.004}$	$0.106_{\pm.006}$	$0.015_{\pm.002}$	$0.002_{\pm.001}$
Min-K% [74]	$0.519_{\pm.005}$	$0.114_{\pm.005}$	$0.011_{\pm.002}$	$0.001_{\pm.001}$	$0.518_{\pm.005}$	$0.108_{\pm.005}$	$0.012_{\pm.002}$	$0.001_{\pm.001}$
Min-K%++ [87]	$0.519_{\pm.004}$	$0.113_{\pm.004}$	$0.010_{\pm.002}$	$0.001_{\pm.000}$	$0.519_{\pm.004}$	$0.108_{\pm.005}$	$0.011_{\pm.002}$	$0.001_{\pm.000}$
BoWs [13]	$0.496_{\pm.004}$	$0.098_{\pm.004}$	$0.009_{\pm.001}$	$0.001_{\pm.001}$	$0.496_{\pm.003}$	$0.096_{\pm.003}$	$0.009_{\pm.001}$	$0.000_{\pm.000}$
ReCall [83]	$0.517_{\pm.004}$	$0.109_{\pm.005}$	$0.011_{\pm.002}$	$0.001_{\pm.001}$	$0.510_{\pm.004}$	$0.109_{\pm.006}$	$0.014_{\pm.002}$	$0.001_{\pm.000}$
CON-Recall [79]	$0.513_{\pm.004}$	$0.110_{\pm.004}$	$0.011_{\pm.002}$	$0.001_{\pm.001}$	$0.510_{\pm.004}$	$0.105_{\pm.003}$	$0.013_{\pm.002}$	$0.001_{\pm.001}$
DC-PDD [90]	$0.513_{\pm.003}$	$0.107_{\pm.005}$	$0.011_{\pm.002}$	$0.001_{\pm.001}$	$0.520_{\pm.006}$	$0.106_{\pm.005}$	$0.011_{\pm.002}$	$0.001_{\pm.000}$
SPV-MIA [24]	$0.804_{\pm.003}$	$0.323_{\pm.008}$	$0.056_{\pm.006}$	$0.010_{\pm.003}$	$0.758_{\pm.003}$	$0.368_{\pm.009}$	$0.062_{\pm.004}$	$0.002_{\pm.002}$
Ratio [80]	$0.793_{\pm.002}$	$0.300_{\pm.007}$	$0.013_{\pm.001}$	$0.002_{\pm.001}$	$0.700_{\pm.004}$	$0.148_{\pm.009}$	$0.013_{\pm.001}$	$0.002_{\pm.001}$
Difference [80]	$0.820_{\pm.002}$	$0.344_{\pm.012}$	$0.013_{\pm.002}$	$0.001_{\pm.000}$	$0.708_{\pm.003}$	$0.160_{\pm.005}$	$0.013_{\pm.001}$	$0.002_{\pm.001}$
Ensemble [88]	$0.813_{\pm.002}$	$0.334_{\pm.004}$	$0.077_{\pm.030}$	$0.015_{\pm.003}$	$0.740_{\pm.004}$	$0.288_{\pm.005}$	$0.053_{\pm.002}$	$0.000_{\pm.000}$
WBC (Ours)	$\mathbf{0.906_{\pm.001}}$	$\mathbf{0.731_{\pm.013}}$	$\mathbf{0.083_{\pm.002}}$	$\mathbf{0.015_{\pm.001}}$	$\mathbf{0.836_{\pm.003}}$	$\mathbf{0.463_{\pm.019}}$	$\mathbf{0.067_{\pm.003}}$	$\mathbf{0.002_{\pm.001}}$

C Accelerated Implementation

In our practical implementation, we accelerate the window-based computation by reformulating it as a 1D convolution operation. The incremental window update strategy in Algorithm 1 is mathematically equivalent to convolving the loss sequences with a uniform kernel $\mathbf{k} = [1, 1, \ldots, 1]$ of length w. For each window position i, the sum $S_i(w) = \sum_{j=i}^{i+w-1} \ell_j$ can be expressed as:

$$S_i(w) = (\ell * \mathbf{k})[i] \tag{18}$$

where $*$ denotes the convolution operation.

This reformulation allows us to leverage highly optimized convolution implementations from established signal processing [61] and deep learning libraries [26]. Modern frameworks provide vectorized and parallelized convolution operations that exploit SIMD instructions, cache-efficient memory access patterns, and GPU acceleration when available. For long sequences, Fast Fourier Transform (FFT) based convolution can further reduce complexity from $O(n \cdot w)$ to $O(n \log n)$ per window size. In our experiments, we use PyTorch's `F.conv1d` operation [62], which selects the most efficient implementation based on input dimensions and available hardware.

Table 11: **Ensemble configuration comparison across datasets.** Heat coloring indicates relative performance rank.

Dataset	Configuration	AUC	TPR@10%	TPR@1%
Stanford	Single Best	$0.8514_{\pm.0026}$	$0.5673_{\pm.0115}$	$0.1749_{\pm.0109}$
	Small Range	$0.8531_{\pm.0026}$	$0.5789_{\pm.0093}$	$0.1822_{\pm.0122}$
	Large Range	$0.7975_{\pm.0031}$	$0.4340_{\pm.0106}$	$0.0000_{\pm.0000}$
	Full Ensemble	$0.8539_{\pm.0024}$	$0.5832_{\pm.0082}$	$0.1940_{\pm.0117}$
	Extended	$0.8457_{\pm.0027}$	$0.5644_{\pm.0098}$	$0.1700_{\pm.0103}$
	Linear Spacing	$0.8431_{\pm.0024}$	$0.5655_{\pm.0073}$	$0.1869_{\pm.0117}$
	Random	$0.8429_{\pm.0025}$	$0.5498_{\pm.0085}$	$0.1723_{\pm.0118}$
Stories	Single Best	$0.8049_{\pm.0029}$	$0.4807_{\pm.0099}$	$0.1469_{\pm.0111}$
	Small Range	$0.8077_{\pm.0030}$	$0.5013_{\pm.0086}$	$0.1440_{\pm.0081}$
	Large Range	$0.7536_{\pm.0033}$	$0.3709_{\pm.0087}$	$0.0000_{\pm.0000}$
	Full Ensemble	$0.8074_{\pm.0030}$	$0.4941_{\pm.0070}$	$0.1600_{\pm.0133}$
	Extended	$0.7974_{\pm.0029}$	$0.4781_{\pm.0086}$	$0.1374_{\pm.0116}$
	Linear Spacing	$0.7969_{\pm.0032}$	$0.4757_{\pm.0095}$	$0.1443_{\pm.0093}$
	Random	$0.7930_{\pm.0033}$	$0.4770_{\pm.0076}$	$0.1450_{\pm.0106}$
WikiHow	Single Best	$0.7968_{\pm.0032}$	$0.4339_{\pm.0138}$	$0.0808_{\pm.0076}$
	Small Range	$0.7915_{\pm.0037}$	$0.4272_{\pm.0089}$	$0.0859_{\pm.0075}$
	Large Range	$0.7584_{\pm.0035}$	$0.3296_{\pm.0249}$	$0.0000_{\pm.0000}$
	Full Ensemble	$0.8008_{\pm.0031}$	$0.4513_{\pm.0098}$	$0.0967_{\pm.0074}$
	Extended	$0.7898_{\pm.0029}$	$0.4248_{\pm.0084}$	$0.0782_{\pm.0057}$
	Linear Spacing	$0.7932_{\pm.0034}$	$0.4400_{\pm.0110}$	$0.0942_{\pm.0057}$
	Random	$0.7951_{\pm.0026}$	$0.4405_{\pm.0083}$	$0.0976_{\pm.0072}$