

Prompt-Induced Recovery Behavior Decomposition for Black-Box Pre-training Data Detection

Xin Fan^{1,*}, Miyamoto Ryoto¹, Fan Mo^{2,5}, Chongxian Chen¹,
Tsuneo Matsumoto³, Fuyuko Kido², Hayato Yamana⁴

¹Department of CSCE, Waseda University, Tokyo, Japan

²Waseda Research Institute for Science and Engineering, Tokyo, Japan

³National Consumer Affairs Center of Japan, Kanagawa, Japan

⁴Faculty of Science and Engineering, Waseda University, Tokyo, Japan

⁵ZOZO Research, Chiba, Japan

*Correspondence: fan_xin@fuji.waseda.jp

Abstract

Large language models are pretrained on massive, largely undisclosed corpora that may contain copyrighted, privacy-sensitive, or otherwise unauthorized content, making pre-training data detection essential for data-rights auditing, privacy protection, and benchmark contamination analysis. However, most existing methods follow a gray-box setting, relying on internal signals such as token probabilities, logits, or losses, which are unavailable for commercial LLMs that expose only the final generated text. Recent black-box methods instead use word-recovery rate as the detection signal, but overlook that a word’s discriminative value can vary substantially across target models and target texts. To address this limitation, we propose **PIRBD-PDD**, a target LLM- and text-adaptive framework for black-box pre-training data detection via *Prompt-Induced Recovery Behavior Decomposition*. **PIRBD-PDD** decomposes the target LLM’s recovery behavior into memorization-oriented and context-oriented modes through prompt induction, contrasts their recovery outcomes to identify high-discriminability words, and computes the final detection score from their recovery rate under the original, non-induced prompt. Experiments across diverse LLMs and data domains show that **PIRBD-PDD** consistently outperforms state-of-the-art black-box baselines. Further analyses provide empirical evidence for the proposed decomposition and show that high-discriminability words provide more robust and informative signals for pre-training data detection.¹

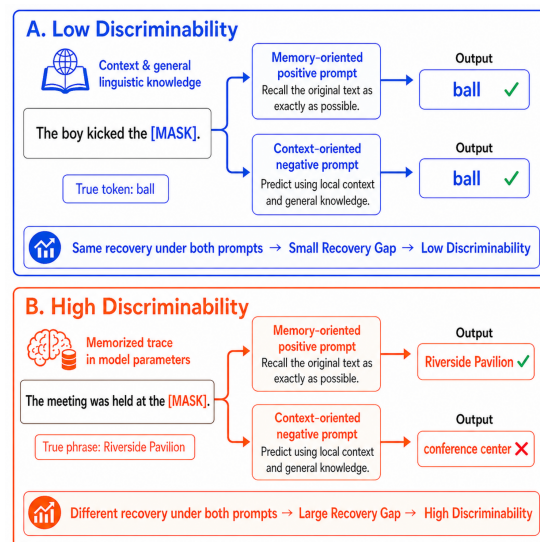


Figure 1: Conceptual illustration of prompt-induced recovery behavior decomposition. A larger recovery gap between memory-oriented and context-oriented recovery behavior indicates higher-discriminability and stronger evidence of pre-training exposure.

1 Introduction

Large language models (LLMs) are pretrained on massive corpora comprising trillions of tokens (Kaplan et al., 2020; Du et al., 2022; Touvron et al., 2023; Wei et al., 2021; Wang et al., 2025), drawn from Web pages, books, news articles, code repositories, social platforms, and other human-generated sources. However, these corpora are often only partially disclosed or entirely undisclosed (Yang et al., 2023; Touvron et al., 2023; Achiam et al., 2023; Bai et al., 2023), limiting the ability of data owners, researchers, and regulators to verify whether specific texts were used during train-

¹Code is available on GitHub: <https://github.com/FANXIN-Gmail/PIRBD-PDD>

ing. This opacity is particularly concerning because pre-training data may contain copyrighted works (Chang et al., 2023), privacy-sensitive information (Beigi and Liu, 2020; Carlini et al., 2021), proprietary documents (Maini et al., 2024), benchmark test instances (Cao et al., 2024; Dong et al., 2024; Magar and Schwartz, 2022), or other content whose use is unauthorized or difficult to verify (Longpre et al., 2024; Samuelson, 2023). Recent copyright and data-rights disputes involving AI developers further underscore the need for reliable technical evidence of data usage (Kim et al., 2023; Samuelson, 2023). Consequently, auditing whether specific content was included in LLM pre-training has become increasingly important for data rights protection, privacy preservation, and trustworthy model evaluation.

This motivates the task of *pre-training data detection*: given a target text and a target LLM, determine whether the text was included in the model’s pre-training corpus (Chen et al., 2024; Das et al., 2025; Maini et al., 2024; Meeus et al., 2025; Fan et al., 2025, 2027; Miyamoto et al., 2026). This task is challenging because LLMs are trained on massive corpora with only one or a few exposures to each individual text (Hoffmann et al., 2022; Muennighoff et al., 2023), leaving weak detection signals that can be easily confounded with recovery oriented by generalized linguistic knowledge or local contextual cues.

Most existing pre-training data detection methods, commonly following a gray-box assumption, extract discriminative detection signals from intermediate outputs of the target LLM, such as token probabilities, logits, losses, or conditional likelihood changes (Shi et al., 2023; Zhang et al., 2025, 2024; Xie et al., 2024). In commercial LLM APIs, however, these intermediate outputs are usually hidden and only final generated text is exposed, which corresponds to a black-box setting. Consequently, the detection signals required by gray-box methods are unavailable, severely limiting their applicability to real-world AI deployments.

Black-box detection is more challenging because final text outputs are less discriminative than intermediate outputs aligned with the LLM’s pre-training objective. A few recent black-box methods instead attempt to infer pre-training exposure from the target LLM’s recovery behavior on the target text. *PETAL* (He et al., 2025) assumes that words within a target text are equally discriminative for detection. This assumption is problematic

because many words can be recovered from syntax, local context, or general knowledge, whereas only a small subset provides exposure-specific evidence. For example, in “The board approved Project Orion on March 17, 2021,” words such as “approved” or “on” are largely context-predictable, while “Project Orion” and “March 17, 2021” are more likely to provide exposure-specific evidence.

InfoTracer (Qi et al., 2026) takes a step toward selective evidence aggregation by focusing on high-traceability words. Nevertheless, its selection strategy is governed by fixed rules, which cannot assess the model-dependent discriminability of words across different target text–target LLM pairs. For example, in a sentence such as “The 2019 Nobel Prize in Literature was awarded to Peter Handke,” a target LLM with strong factual knowledge may recover “Peter Handke” from the event description alone, regardless of whether it has seen the exact sentence. For another LLM lacking this background knowledge, recovering the same name is more likely to rely on exposure to the target text. Thus, the same word can be weakly or strongly discriminative depending on the target LLM. These limitations suggest that effective black-box pre-training data detection requires adaptively identifying target-LLM- and text-dependent high-discriminability words whose recovery is more attributable to memorization-oriented recall than to context-oriented prediction.

Motivated by these limitations, we propose *PIRBD-PDD*, a target LLM- and text-adaptive framework for black-box pre-training data detection. Our key hypothesis is that a target LLM’s reproduction behavior reflects two entangled sources: memorized traces of the target text and generalized linguistic knowledge derived from the surrounding context. Without disentangling these sources, a high recovery rate may not reliably indicate pre-training exposure, since some words can be recovered from contextual cues alone. Figure 1 illustrates this intuition. For a contextually predictable word such as “The boy kicked the [MASK]”, both memory-oriented and context-oriented prompts may recover “ball”, making the recovery weak evidence of exposure. In contrast, for a more specific word such as “The meeting was held at the [MASK]”, recovering “Riverside Pavilion” under a memory-oriented prompt but not under a context-oriented prompt suggests that the recovery is more likely oriented by memorized textual traces.

Based on this insight, we introduce *Prompt-Induced Recovery Behavior Decomposition* to identify semantic units, semantically meaningful words or phrases in the target text, that provide the most discriminative evidence of pre-training exposure. Specifically, **PIRBD-PDD** queries the target LLM with a memory-oriented positive prompt and a context-oriented negative prompt, and contrasts their recovery outcomes to separate memorization-oriented recovery from context-oriented prediction. Semantic units that are strongly recovered under the memory-oriented prompt but weakly recovered under the context-oriented prompt are selected as high-discriminability semantic units. The final detection score is then computed from the recovery rate of these positions under the original, non-induced prompt, preserving the LLM’s natural recovery behavior while focusing on more exposure-specific evidence.

Experiments are organized to evaluate **PIRBD-PDD** from three perspectives: overall detection performance on *WikiMIA* (Shi et al., 2023) and *News* (Qi et al., 2026) across open-source and commercial LLMs, the effectiveness of positive and negative prompts in decomposing recovery behavior, and the sensitivity of detection performance to the number of selected high-discriminability semantic units.

Our contributions are summarized as follows:

- **Methodologically**, we propose **PIRBD-PDD**, a target LLM- and text-adaptive framework for black-box pre-training data detection based on selected high-discriminability semantic units.
- **Conceptually**, we introduce *Prompt-Induced Recovery Behavior Decomposition*, which distinguishes and contrasts memorization-oriented recovery with context-oriented prediction to identify semantic units that provide more exposure-specific evidence for each target text–target LLM pair.
- **Empirically**, experiments on *WikiMIA* and *News* demonstrate the superiority of **PIRBD-PDD** over state-of-the-art black-box baselines across open-source and commercial LLMs, improving the average AUC from 0.629 to 0.666 on *WikiMIA* and from 0.718 to 0.824 on *News*.

2 Problem Statement

Pre-training data detection (Shi et al., 2023; Qi et al., 2026) aims to determine whether a suspect text was included in the pre-training corpus of a target LLM. Formally, let M denote a pre-trained LLM with an undisclosed pre-training corpus $\mathcal{D}_{\text{pre}}$, and let $\mathbf{x}$ denote a suspect text. The goal is to design a detection method $\mathcal{A}$ that predicts whether $\mathbf{x}$ is a seen text, i.e., $\mathbf{x} \in \mathcal{D}_{\text{pre}}$, or an unseen text, i.e., $\mathbf{x} \notin \mathcal{D}_{\text{pre}}$:

$$\mathcal{A}(\mathbf{x}, M) \rightarrow \{0, 1\},$$

where $\mathcal{A}(\mathbf{x}, M) = 1$ indicates that $\mathbf{x}$ is predicted to be included in the pre-training corpus, and $\mathcal{A}(\mathbf{x}, M) = 0$ otherwise. We study this problem under a strict black-box setting. The detection method $\mathcal{A}$ can only interact with the target LLM through suspect texts or prompts derived from the suspect text $\mathbf{x}$; for example, it may mask a word in $\mathbf{x}$ and ask the LLM to recover the masked word from the surrounding context. The detection method $\mathcal{A}$ has no access to the LLM’s pre-training corpus, architecture, parameters, logits, token probabilities, losses, hidden states, or gradients. This black-box setting reflects practical detection scenarios for opaque commercial LLMs.

3 Method

Motivated by the observation that semantic units contribute unequally to pre-training data detection, and that their discriminative value is specific to each target model–text pair, **PIRBD-PDD** aims to identify semantic units that provide the most informative evidence of pre-training exposure for the given target LLM and target text. As illustrated in Figure 2, **PIRBD-PDD** extracts semantic units from input text, generates plausible alternatives for each semantic unit, identifies high-discriminability units through *Prompt-Induced Recovery Behavior Decomposition*, and computes the final detection score from their recovery rates.

3.1 Semantic Unit Extraction.

Given a textual input $\mathbf{x}$ with the text length $\mathcal{T}$, we first extract a set of semantic units $\mathcal{S}_{\mathbf{x}} = \{s_i\}_{i=1}^U$ from the original text with the number of U as shown in part 1 of Figure 2. Each s_i denotes a single-token or multi-token semantic unit with an identifiable surface form and linguistic role, capturing localized meaning beyond purely syntactic structure. In practice, we first tokenize $\mathbf{x}$ with

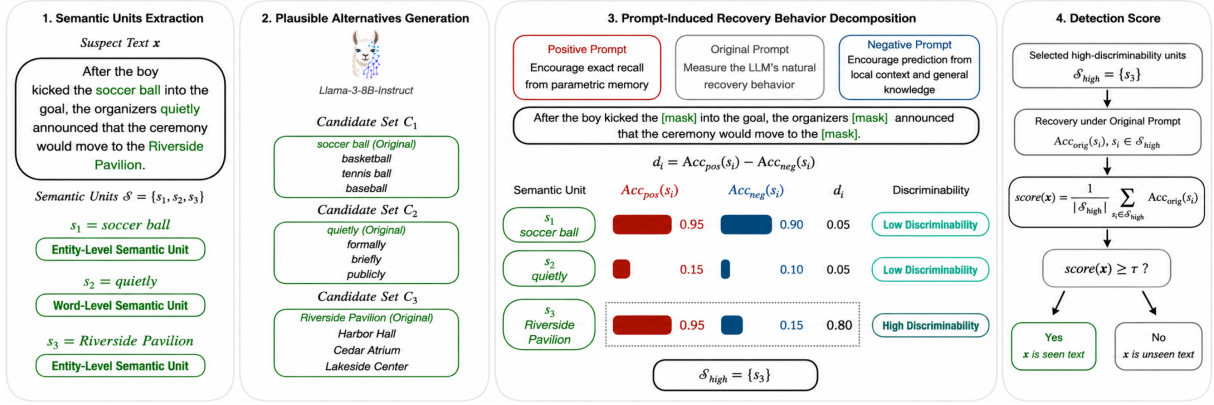


Figure 2: Overview. Given a suspect text, the **PIRBD-PDD** first extracts semantic units and generates plausible alternatives for each semantic unit. It then decomposes the target LLM’s recovery behavior through positive and negative prompt induction, selects high-discriminability semantic units whose recovery is more memory-oriented than context-oriented, and computes the final detection score from their recovery rate under the original prompt.

NLTK into a sequence of tokens. We then apply spaCy named entity recognition to aggregate relevant tokens into unified semantic entities, which serve as entity-level semantic units (e.g., “New York”, “Best Actor”, or “Luke Evans”) in the textual input x . After that, we utilize NLTK part-of-speech tags to extract numerical tokens, month names, and selected content words, including adjectives, adverbs, and verbs, which serve as word-level semantic units (e.g., “storm”, “threaten”, or “rapid”). Function words and purely syntactic tokens are discarded because their recovery is usually dominated by local grammar rather than memorization. Each semantic unit is represented by its surface expression, start index, span length, and linguistic tag, yielding a structured set of semantically meaningful positions grounded in the original text. The pseudo-code is attached in Appendix G

3.2 Plausible Alternatives Generation.

For each semantic unit $s_i \in \mathcal{S}$, we generate a set of plausible alternatives $\mathcal{A}_i = \{c_{i,n}\}_{n=1}^{\mathcal{N}}$, where $\mathcal{N}$ is the number of plausible alternatives. The generation strategy depends on two types of semantic units: entity-level semantic units and word-level semantic units. For entity-level semantic units, we use an in-context learning prompt that provides several examples of an entity and its conceptually similar alternatives, and then asks the auxiliary model to produce alternatives for the target semantic unit s_i . For other word-level semantic units, we keep the original text x as context and mark the selected semantic unit s_i with special delimiters. The auxiliary model is then prompted to produce

alternatives that are semantically and contextually compatible with the marked semantic unit s_i under the original text x . In this way, the generated alternatives preserve conceptual, semantic, and contextual plausibility while differing from the original expression of semantic units s_i . Finally, we define the candidate set for s_i by adding the original semantic unit to its plausible alternatives, yielding $C_i = \{s_i\} \cup \mathcal{A}_i$ as shown in part 2 of Figure 2. Candidate sets constrain the target LLM to a valid response space, thereby stabilizing recovery evaluation. We further assess their effect through an ablation study in Appendix E. The prompt templates are provided in Appendix A.

3.3 Prompt-Induced Recovery Behavior Decomposition.

The core idea of **PIRBD-PDD** rests on an intuitive hypothesis: a target LLM’s recovery behavior at a semantic unit is jointly shaped by two different patterns:

- Memorization-oriented recall of text-specific traces.
- Context-oriented prediction from local context and general linguistic knowledge.

We introduce *Prompt-Induced Recovery Behavior Decomposition*, which identifies high-discriminability semantic units by decomposing and contrasting memory-oriented and context-oriented recovery behaviors.

Specifically, for each semantic unit $s_i \in \mathcal{S}_x$, we construct a masked input $x_{s_i}^{\text{mask}}$ by replacing s_i in the original text x with a mask token. We

then construct the prompt using the masked input $\mathbf{x}_{s_i}^{\text{mask}}$ and the candidate set $\mathcal{C}_i$, and query the target LLM to select the candidate that best recovers the masked semantic unit. To decompose the LLM’s recovery behavior, we design two prompt conditions: a positive prompt that induces memorization-oriented recovery from parametric memory, and a negative prompt that induces context-oriented prediction based on local context and general linguistic knowledge. For each semantic unit, we query the target LLM multiple times under each prompt condition and compute the recovery rate as the proportion of responses that correctly select the original semantic unit s_i . We then compare the LLM’s recovery results under these two prompt-induced behaviors to estimate the discriminability of s_i . Semantic units with high recovery rates under both prompts are often easily inferable from context, whereas those with low recovery rates under both prompts provide limited evidence for reliable detection. In contrast, semantic units with a large positive-negative recovery-rate gap are more informative, because their recovery is sensitive to memory-oriented prompting but not sufficiently supported by context-oriented prediction. We therefore assign each semantic unit a discriminative score:

$$d_i = \text{Acc}_{\text{pos}}(s_i) - \text{Acc}_{\text{neg}}(s_i), \quad (1)$$

where $\text{Acc}_{\text{pos}}(s_i)$ and $\text{Acc}_{\text{neg}}(s_i)$ denote the recovery rates of s_i under the positive and negative prompts, respectively. We rank all semantic units in $\mathcal{S}_x$ by d_i and select the top- $\mathcal{K}$ semantic units as the high-discriminability subset $\mathcal{S}_{\text{high}} \subseteq \mathcal{S}_x$ as shown in part 3 of Figure 2, where $\mathcal{K}$ is a hyperparameter. The prompts are provided in Appendix A.

3.4 Detection Score.

After identifying the high-discriminability semantic-unit subset $\mathcal{S}_{\text{high}} \subseteq \mathcal{S}_x$, we compute the final detection score for the input text $\mathbf{x}$ based on the target LLM’s natural recovery behavior on these high-discriminability semantic units. For each $s_i \in \mathcal{S}_{\text{high}}$, we similarly query the target LLM multiple times with the masked input $\mathbf{x}_{s_i}^{\text{mask}}$ and the candidate set $\mathcal{C}_i$, but under the original, non-induced prompt. The $\text{Acc}_{\text{orig}}(s_i)$ is consistently defined as the proportion of responses that correctly select the original semantic unit s_i from $\mathcal{C}_i$. The detection score is then defined as the average recovery accuracy over the selected

high-discriminability semantic units, as shown in part 4 of Figure 2:

$$\text{score}(\mathbf{x}) = \frac{1}{|\mathcal{S}_{\text{high}}|} \sum_{s_i \in \mathcal{S}_{\text{high}}} \text{Acc}_{\text{orig}}(s_i). \quad (2)$$

A larger score indicates that the target LLM can consistently recover the original semantic units whose recovery is unlikely to be explained by local context alone. We make the final prediction by a pre-defined thresholding, aligning with previous work (Zhang et al., 2024):

$$\hat{y}(\mathbf{x}) = \begin{cases} 1, & \text{if } \text{score}(\mathbf{x}) \geq \tau, \\ 0, & \text{if } \text{score}(\mathbf{x}) < \tau. \end{cases} \quad (3)$$

where $\hat{y}(\mathbf{x}) = 1$ indicates that $\mathbf{x}$ is predicted as a seen text, and $\hat{y}(\mathbf{x}) = 0$ indicates an unseen text. The threshold τ can be calibrated on a small validation set or selected according to the desired false-positive rate in the target detection scenario.

4 Experiments

In this section, we conduct comprehensive experiments and analyses to evaluate the effectiveness and robustness of *PIRBD-PDD* for black-box pre-training data detection. Our evaluation covers both open-source and commercial LLMs across different model scales, text lengths, and data domains. We report the main results using AUC as the primary metric, with additional results under *TPR@5%FPR* and *Balanced Accuracy* provided in the Appendix D and Appendix C, respectively. We further evaluate robustness across data domains, prompt-induced behavioral decomposition (supplemented by separate seen/unseen recovery-rate analyses in Appendix B), and the impact of hyperparameter $\mathcal{K}$. Appendix F presents theoretical and empirical comparisons of computational cost.

4.1 Settings

Benchmarks and Models. We conduct our main experiments on two complementary benchmarks, *WikiMIA* (Shi et al., 2023) and *News* (Qi et al., 2026). *WikiMIA* is a standard benchmark built from temporally partitioned Wikipedia texts, with likely seen samples drawn from texts before the target model’s release or knowledge cut-off and unseen samples from later texts. Following prior work, we evaluate *LLaMA-7B* (Touvron et al., 2023), *LLaMA-13B*, and *LLaMA-30B*

on 32-, 64-, 128-, and 256-token splits to assess robustness across model scales and text lengths. *News*, introduced by *InfoTracer*, provides a realistic commercial black-box setting using temporally separated public news articles as likely seen and unseen texts. Specifically, the LLMs evaluated by *News* includes *GPT-3.5*², *GPT-4o*², *Claude-4*³, *Gemini-2.5*⁴, *GLM-4-Air*⁵, *DeepSeek-V4*⁶, *Qwen-3-7B*, and *Qwen-3-14B*.

Baselines. We compare *PIRBD-PDD* with two state-of-the-art black-box pre-training data detection methods, *PETAL* (He et al., 2025) and *InfoTracer* (Qi et al., 2026). *PETAL* approximates probability-based detection under black-box access. It queries the target model to obtain generated tokens, measures token-level semantic similarity between generated tokens and ground-truth tokens using a semantic encoder, and maps these similarity scores to pseudo token probabilities with the help of a surrogate model. The resulting approximate probabilities are then used to compute a probability-based detection score. *InfoTracer* asks the target model to recover the original expression of words from semantically confounding alternatives. It selects words by hand-crafted criteria rules and uses the recovery rate on these words as the detection score.

Evaluation Metrics. Following prior work, we mainly evaluate detection performance using *AUC*, *Balanced Accuracy*, and *TPR@5%FPR*. *AUC* measures the overall separability between seen and unseen texts across all decision thresholds. *Balanced Accuracy* computes the average of true positive rate and true negative rate, which is suitable for potentially imbalanced settings. *TPR@5%FPR* measures the true positive rate when the false positive rate is constrained to 5%, reflecting detection performance under strict false-alarm requirements.

Implementation Details. For open-source models, all experiments are implemented with PyTorch and Hugging Face Transformers, and are run on a single NVIDIA A100 80GB GPU using fp16 precision unless otherwise specified. Commercial LLMs are evaluated through their official

APIs. For *PETAL*, we use *all-MiniLM-L6-v2* as the sentence encoder and *GPT-2 XL* as the surrogate model, with greedy decoding. For *InfoTracer*, we use the precomputed reproduction results released in its official GitHub repository, together with the benchmark- and model-specific configurations from the original implementation, to closely follow the reported experimental setting. For *PIRBD-PDD*, both semantic alternative generation and semantic unit evaluation use a fixed decoding configuration with temperature 0.3 and top-0.9. The auxiliary model for semantic alternative generation is fixed as *Llama-3-8B-Instruct*. For the hyperparameter $\mathcal{K}$, we perform a small sweep over $\{10\%, 20\%, 30\%, 40\%, 50\%\}$ on a held-out validation set. We set $\mathcal{K} = 40\%$ for *WikiMIA*; for *News*, we use $\mathcal{K} = 30\%$ for *Qwen-7B*, *Qwen-14B* and *Claude*, $\mathcal{K} = 20\%$ for others.

4.2 Results

WikiMIA. Table 1 reports the AUC results on *WikiMIA* across LLaMA model scales and input lengths. Overall, *PIRBD-PDD* achieves the best performance in almost all settings. Across all compared methods, AUC tends to increase with larger models and longer inputs, consistent with prior findings in grey-box detection methods (Shi et al., 2023; Zhang et al., 2024). From a black-box perspective, larger models may leave stronger exposure traces in output behavior, while longer inputs provide richer evidence for distinguishing seen texts from unseen texts. Moreover, the performance margin of *PIRBD-PDD* over baselines generally widens as model size and text length increase. This trend suggests that larger models make the discriminative value among semantic units more pronounced, while longer inputs offer more semantic units with high discriminability. These results demonstrate the superiority and robustness of *PIRBD-PDD* across model scales and input lengths.

News. On *News*, in addition to the standard sample-level AUC results as reported on *WikiMIA*, we further compare set-level AUC results across multiple commercial LLMs and different set sizes $\mathcal{G}$ in Figure 3. Sample-level detection assigns a detection score to each text independently, whereas set-level detection treats a small set of suspected texts as the decision object and computes its final score by averaging the sample-level scores within the set. The set-level detection reflects

²<https://platform.openai.com/>

³<https://platform.claude.com/docs/en/release-notes/system-prompts>

⁴<https://ai.google.dev/gemini-api/docs>

⁵<https://open.bigmodel.cn/dev/api>

⁶<https://api-docs.deepseek.com/>

Len.	Model	Methods		
		<i>PIRBD-PDD</i>	<i>InfoTracer</i>	<i>PETAL</i>
32	LLaMA-7B	0.608	0.598	0.588
	LLaMA-13B	0.593	0.609	0.601
	LLaMA-30B	0.650*	0.611	0.617
64	LLaMA-7B	0.629*	0.605	0.548
	LLaMA-13B	0.618	0.596	0.566
	LLaMA-30B	0.655*	0.619	0.584
128	LLaMA-7B	0.654*	0.604	0.578
	LLaMA-13B	0.639*	0.610	0.599
	LLaMA-30B	0.679*	0.628	0.619
256	LLaMA-7B	0.726*	0.660	0.596
	LLaMA-13B	0.786*	0.698	0.612
	LLaMA-30B	0.759*	0.705	0.618

Table 1: AUC results on *WikiMIA*. The best result in each setting is shown in bold. * indicates that *PIRBD-PDD* significantly outperforms *Infotracer* under paired bootstrap testing with $p \leq 0.05$.

whether a collection of potentially unauthorized texts has been used for training rather than to judge each text individually. *PIRBD-PDD* consistently outperforms *InfoTracer* across all evaluated commercial LLMs and set sizes, with a widening advantage as the set size increases. Notably, with only five samples, *PIRBD-PDD* achieves around or above 0.90 AUC on most commercial models. This demonstrates its effectiveness in realistic commercial black-box settings, even with limited suspect samples.

4.3 Robustness across Data Domains

To assess cross-domain generalization beyond *WikiMIA* and *News*, we conduct a small-scale evaluation on *MIMIR* (Duan et al., 2024), a challenging benchmark where substantial n -gram overlap between seen and unseen texts leaves many detectors near chance. To limit *InfoTracer*’s additional API cost, we sample 100 seen and 100 unseen texts from each of seven subsets and evaluate *PIRBD-PDD* and *InfoTracer* using *Pythia-6.9B* as the target model. As shown in Table 2, *PIRBD-PDD* achieves higher AUC and Balanced Accuracy on six of seven subsets, raising the averages from 0.5746 to 0.6078 and from 0.5867 to 0.6184, respectively. Its average TPR@5%FPR is only slightly higher (0.0764 vs. 0.0745), with gains on four subsets. Overall, these preliminary results provide additional evidence that *PIRBD-PDD* generalizes across diverse data domains, particularly in terms of AUC and Balanced Accuracy,

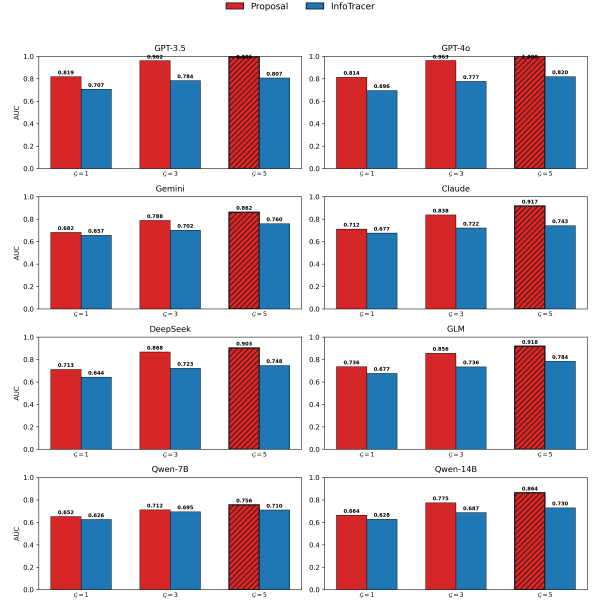


Figure 3: AUC results on *News* under different set sizes.

while maintaining low-FPR performance comparable to that of *InfoTracer*.

4.4 Effectiveness of Prompt-Induced Recovery Behavior Decomposition

To evaluate the effect of prompt induction on model recovery behavior, we compare recovery-rate distributions under positive, original, and negative prompts. As shown in Figure 4 and Figure 5, the positive prompt consistently shifts the recovery-rate distribution upward relative to the original prompt, whereas the negative prompt generally suppresses recovery accuracy. This pattern holds across *WikiMIA* and *News*, covering different text lengths, model scales, and commercial LLMs. The separation is especially clear on *News*, where commercial models exhibit a consistent ordering of positive, original, and negative recovery rates. Appendix B further shows that prompt-condition differences are more pronounced for seen than unseen texts. Based on these empirical observations and analyses, we attribute the observed differences to the positive prompt strengthening memorization-oriented recovery and the negative prompt encouraging context-oriented prediction, and prompt induction’s effectiveness to separating these recovery behaviors. This explanation is intended as an empirically grounded interpretation of the observed behavioral differences, rather than a rigorous demonstration of causality. Under this interpretation, prompt-induced behav-

Subset	InfoTracer			PIRBD-PDD		
	AUC	ACC	TPR@5%FPR	AUC	ACC	TPR@5%FPR
arxiv	0.5662	0.5800	0.0200	0.5420	0.5821	0.0783
dm_mathematics	0.6128	0.6200	0.1000	0.6645	0.6250	0.0844
github	0.5131	0.5569	0.0612	0.5421	0.5480	0.0716
hackernews	0.5368	0.5400	0.1200	0.5641	0.5862	0.0623
pile_cc	0.5678	0.5700	0.0200	0.6359	0.6579	0.0939
pubmed_central	0.6248	0.6400	0.0600	0.6504	0.6805	0.0814
wikipedia	0.6006	0.6000	0.1400	0.6554	0.6491	0.0626
Average	0.5746	0.5867	0.0745	0.6078	0.6184	0.0764

Table 2: AUC, ACC, and TPR@5%FPR results on *MIMIR*. The best result for each subset and evaluation metric is shown in bold.

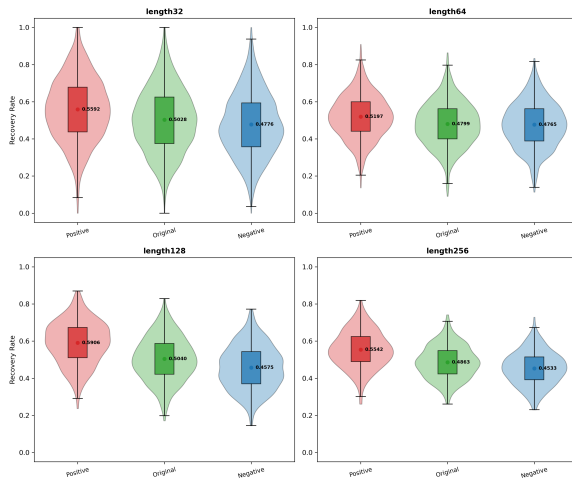


Figure 4: Recovery-Rate distributions under different prompts on WikiMIA Dataset.

ioral decomposition helps identify semantic units whose recovery is more indicative of memorization, supporting *PIRBD-PDD*’s core design.

4.5 Impact of Hyperparameter $\mathcal{K}$

The hyperparameter $\mathcal{K}$ determines the proportion of top-ranked high-discriminability semantic units used for the final detection score. To assess its impact, we vary $\mathcal{K}$ from 10% to 100% on the *News* dataset and report AUC under set size $\mathcal{G} = 1$ across target LLMs, as shown in Figure 6. Overall, performance is highest when a moderate subset is selected, with most models achieving the best or near-best AUC around 20%–30%. More broadly, performance remains relatively stable across the 10%–40% range for different target LLMs, demonstrating a reasonable degree of empirical robustness to the choice of $\mathcal{K}$. As $\mathcal{K}$ increases further, the AUC generally declines, suggesting that addi-

tional low-discriminability units introduce context-oriented or weakly informative signals that dilute the detection evidence. These results indicate that discriminative evidence is concentrated in a small subset of semantic units, and that a moderate $\mathcal{K}$ better balances evidence coverage and noise suppression. In practical applications, we recommend selecting $\mathcal{K}$ from a small validation set tailored to the target model and target dataset, with a search range of 10%–40%.

5 Related Work

5.1 Grey-Box Detection Methods

Grey-box detection methods infer pre-training exposure from internal output statistics, such as token probabilities, logits, or losses. For likelihood-based methods, *Min-K%* (Shi et al., 2023) aggregates the negative log-likelihood of low-likelihood tokens, *Min-K%++* (Zhang et al., 2025) normalizes this score using the mean and variance of the next-token distribution, and *DC-PDD* (Zhang et al., 2024) calibrates token probabilities with reference-corpus token frequencies to reduce frequency bias. Perturbation-based methods, such as *RECALL* (Xie et al., 2024), *Con-RECALL* (Wang et al., 2025), and *PAC* (Ye et al., 2024), exploit changes in likelihood or negative log-likelihood under prefix insertion, substitution, or other perturbations. Although these methods are highly effective, with some achieving AUC scores above 0.9 on the *WikiMIA* benchmark, they are difficult to apply to commercial LLMs because closed APIs typically do not expose complete probability distributions, logits, or losses.

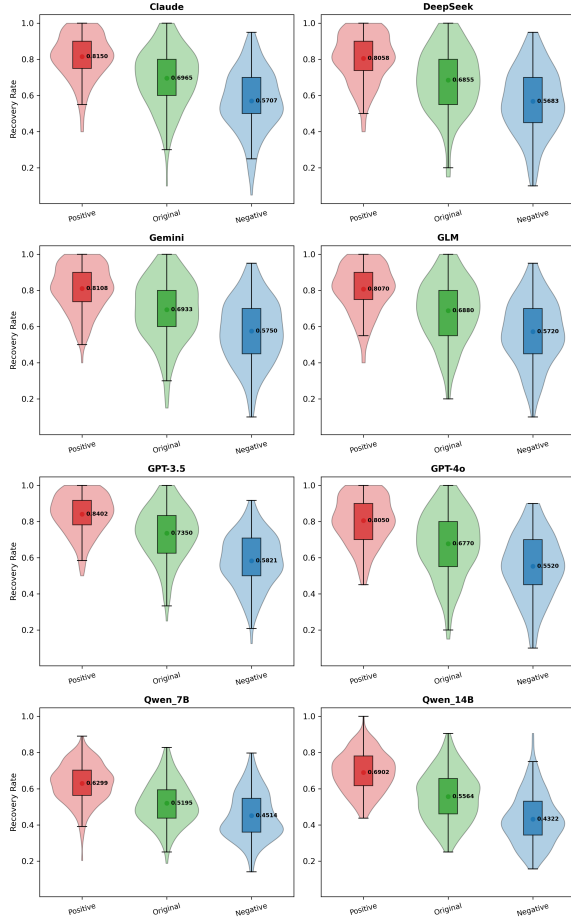


Figure 5: Recovery-Rate distributions under different prompts on *News* Dataset.

5.2 Black-Box Detection Methods

Black-box detection methods seek to infer pre-training exposure from observable generation behavior rather than internal likelihoods. *PETAL* (He et al., 2025) approximates token-level probabilities through semantic similarity and uses the resulting perplexity-like score for detection. However, it still aggregates evidence over tokens largely uniformly, overlooking that many tokens can be recovered from syntax, local context, common collocations, or general knowledge, while only a subset provides exposure-specific evidence. *InfoTracer* (Qi et al., 2026) measures the recovery rate of selected high-traceability semantic units as the detection signal. Nevertheless, its selection strategy relies on general rules, which cannot explicitly account for the fact that the traceability of a semantic unit may vary substantially across different target LLMs. In contrast, *PIRBD-PDD* decomposes the target model’s reproduction behavior through prompt induction. By

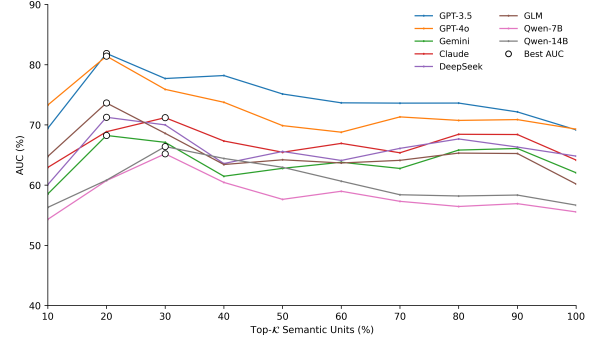


Figure 6: AUC Results with different K under set size $G = 1$ on *News* Dataset.

contrasting memory-oriented and context-oriented recovery results, it adaptively identifies high-discriminability semantic units for each target model and target text, thereby improving detection performance.

6 Conclusion

In this work, we proposed *PIRBD-PDD*, a black-box pre-training data detection framework that adaptively identifies high-discriminability semantic units through *prompt-induced recovery behavior decompositions*. By distinguishing memorization-oriented recovery from context-oriented prediction, *PIRBD-PDD* focuses detection on semantic units that provide more reliable evidence of pre-training exposure. Experiments show substantial and consistent gains across both open-source and commercial LLMs. *PIRBD-PDD* improves the average AUC over the SOTA baseline by 3.7 and 10.5 percentage points on *WikiMIA* and *News*, respectively. On the *News* dataset, its best performance reaches an AUC of 0.995 on *GPT-3.5* and 1.0 on *GPT-4o* with a small set size. Further analyses support the effectiveness of prompt-induced behavior decomposition and high-discriminability semantic-unit selection, demonstrating the value of target-adaptive semantic evidence for auditing pre-training data usage in opaque LLMs.

Limitations

Although *PIRBD-PDD* demonstrates strong effectiveness under black-box access, it still has several limitations. First, its performance depends on the quality of semantic unit extraction and alternative generation, as noisy units or insufficiently confounding alternatives may weaken the detection signal. Second, the prompt-induced behavioral

contrast relies on the target model’s instruction-following behavior, whose effectiveness may vary across model families, decoding strategies, and safety policies. Future work can improve semantic unit construction and develop more adaptive prompt optimization strategies.

Ethics Statement

This work aims to support responsible detection of pre-training data usage in opaque LLMs, benefiting data-rights protection, privacy auditing, copyright compliance, and benchmark contamination analysis. However, pre-training data detection also carries dual-use risks, as it may be misused to infer the inclusion of privacy-sensitive or proprietary texts. In addition, detection errors may lead to ethical and practical harms. False positives may result in unfounded accusations of unauthorized data use, causing reputational, legal, or commercial consequences for model developers. False negatives, conversely, may fail to identify actual unauthorized use, weakening data-rights protection and giving affected parties a false sense of assurance. Accordingly, our experiments use public or established benchmarks, and the method should only be applied in authorized auditing scenarios. Detection results should be treated as statistical evidence rather than definitive legal proof, and should be interpreted with appropriate thresholds, uncertainty awareness, and, where possible, corroborating evidence. Practical deployment should comply with relevant privacy regulations, legal procedures, and platform policies.

Acknowledgments

Funding: This work was supported by the Ministry of Education, Culture, Sports, Science and Technology (MEXT) through the Supporting Pioneering Research through AI for 1,000 Discovery Challenges (SPReAD) program [grant number 26273232], the Japan Science and Technology Agency (JST) SPRING program [grant number JPMJSP2128].

We gratefully acknowledge the assistance of GPT-5.6 Sol with grammar and spelling checks, layout and formatting checks, and light language editing to improve clarity while preserving the authors’ intended meaning. All AI-assisted revisions were reviewed and approved by the authors, who take full responsibility for the final manuscript.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.
- Ghazaleh Beigi and Huan Liu. 2020. A survey on privacy in social media: Identification, mitigation, and applications. *ACM Transactions on Data Science*, 1(1):1–38.
- Jialun Cao, Wuqi Zhang, and Shing-Chi Cheung. 2024. Concerned with data contamination? assessing countermeasures in code language model. *arXiv preprint arXiv:2403.16898*.
- Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pages 2633–2650.
- Kent K Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. 2023. Speak, memory: An archaeology of books known to chatgpt/gpt-4. *arXiv preprint arXiv:2305.00118*.
- Tong Chen, Akari Asai, Niloofar Miresghallah, Se-won Min, James Grimmermann, Yejin Choi, Han-naneh Hajishirzi, Luke Zettlemoyer, and Pang Wei Koh. 2024. Copybench: Measuring literal and non-literal reproduction of copyright-protected text in language model generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15134–15158.
- Debeshee Das, Jie Zhang, and Florian Tramèr. 2025. Blind baselines beat membership inference attacks for foundation models. In *2025 IEEE Security and Privacy Workshops (SPW)*, pages 118–125. IEEE.
- Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. 2024. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. *arXiv preprint arXiv:2402.15938*.
- Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

- Michael Duan, Anshuman Suri, Niloofar Miresghalah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hananeh Hajishirzi. 2024. Do membership inference attacks work on large language models? *arXiv preprint arXiv:2402.07841*.
- Xin Fan, Fan Mo, Miyamoto Ryoto, Eint Sandi Aung, Chongxian Chen, Tsuneo Matsumoto, Fuyuko Kido, and Hayato Yamana. 2025. Dad-pdd: Dual-aspect discrepancies-based pre-training data detection for llms. In *International Conference on Neural Information Processing*, pages 94–108. Springer.
- Xin Fan, Miyamoto Ryoto, Fan Mo, Chongxian Chen, Tsuneo Matsumoto, Fuyuko Kido, and Hayato Yamana. 2027. Rads-pdd: Reproducibility-aware dynamic similarity-based pre-training data detection against llms. *Neural Networks*, 205:109393.
- Yu He, Boheng Li, Liu Liu, Zhongjie Ba, Wei Dong, Yiming Li, Zhan Qin, Kui Ren, and Chun Chen. 2025. Towards label-only membership inference attack against pre-trained large language models. In *USENIX Security*.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. An empirical analysis of compute-optimal large language model training. *Advances in neural information processing systems*, 35:30016–30030.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.
- Siwon Kim, Sangdoo Yun, Hwaran Lee, Martin Gubri, Sungroh Yoon, and Seong Joon Oh. 2023. Propile: Probing privacy leakage in large language models. *Advances in Neural Information Processing Systems*, 36:20750–20762.
- Shayne Longpre, Robert Mahari, Anthony Chen, Naana Obeng-Marnu, Damien Sileo, William Brannon, Niklas Muennighoff, Nathan Khazam, Jad Kabbara, Kartik Perisetla, et al. 2024. A large-scale audit of dataset licensing and attribution in ai. *Nature Machine Intelligence*, 6(8):975–987.
- Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation. *arXiv preprint arXiv:2203.08242*.
- Pratyush Maini, Hengrui Jia, Nicolas Papernot, and Adam Dziedzić. 2024. Llm dataset inference: Did you train on my dataset? *Advances in Neural Information Processing Systems*, 37:124069–124092.
- Matthieu Meeus, Igor Shilov, Shubham Jain, Manuel Faysse, Marek Rei, and Yves-Alexandre de Montjoye. 2025. Sok: Membership inference attacks on llms are rushing nowhere (and how to fix it). In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 385–401. IEEE.
- Ryoto Miyamoto, Xin Fan, Fuyuko Kido, Tsuneo Matsumoto, and Hayato Yamana. 2026. Openlvm-mia: A controlled benchmark revealing the limits of membership inference attacks on large vision-language models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2680–2689.
- Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. 2023. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36:50358–50376.
- Tao Qi, Jinhua Yin, Dongqi Cai, Yueqi Xie, Huili Wang, Zhiyang Hu, Peiru Yang, Guoshun Nan, Zhili Zhou, Chuhan Wu, et al. 2026. Auditing unauthorized training data from ai generated content using information isotopes. *Nature Communications*.
- Pamela Samuelson. 2023. Generative ai meets copyright. *Science*, 381(6654):158–161.
- Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2023. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Cheng Wang, Yiwei Wang, Bryan Hooi, Yujun Cai, Nanyun Peng, and Kai-Wei Chang. 2025. Con-recall: Detecting pre-training data in llms via contrastive decoding. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1013–1026.
- Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.
- Roy Xie, Junlin Wang, Ruomin Huang, Minxing Zhang, Rong Ge, Jian Pei, Neil Gong, and Bhuwan Dhingra. 2024. Recall: Membership inference via relative conditional log-likelihoods. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8671–8689.
- Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Wentao Ye, Jiaqi Hu, Liyao Li, Haobo Wang, Gang Chen, and Junbo Zhao. 2024. Data contamination calibration for black-box llms. *arXiv preprint arXiv:2405.11930*.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. 2025. [Min-k%++: Improved baseline for pre-training data detection from large language models](#). In *The Thirteenth International Conference on Learning Representations*.

Weichao Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024. Pre-training data detection for large language models: A divergence-based calibration method. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5263–5274.

A Prompt Templates

A.1 Prompt Templates for Behavioral Decomposition

Figure 7 presents the three prompt templates used in our experiments. All prompts share the same multiple-choice cloze-question backbone, while the positive and negative prompts insert prompt-specific induction instructions into this backbone. The positive prompt encourages memory-oriented recovery of the original wording, whereas the negative prompt encourages context-oriented prediction based only on local context and general language understanding. The masked passage and candidate options are kept identical across the three prompts, so differences in recovery behavior mainly reflect the effect of prompt-induced behavioral contrast.

Original Prompt

Shared backbone

Choose the word or phrase that was used in the <masked token> token. You must choose one option.

Passage: <masked passage>

Options:

- <option A>
- <option B>
- <option C>
- <option D>

Answer:

Positive Prompt

Positive induction

You may have seen this passage during training.

Shared backbone

Choose the word or phrase that was used in the <masked token> token. You must choose one option.

Positive induction

Fill in the masked token by recalling the exact original wording if possible.

Shared backbone

Passage: <masked passage>

Options:

- <option A>
- <option B>
- <option C>
- <option D>

Answer:

Negative Prompt

Negative induction

Assume you have never seen this passage before.

Shared backbone

Choose the word or phrase that was used in the <masked token> token. You must choose one option.

Negative induction

Fill in the masked token using only local context and general language understanding.

Shared backbone

Passage: <masked passage>

Options:

- <option A>
- <option B>
- <option C>
- <option D>

Answer:

Figure 7: Prompt templates used for prompt-induced behavior decomposition. Gray blocks denote the shared multiple-choice backbone, while blue and orange blocks denote the prompt-specific induction instructions in the positive and negative prompts, respectively.

A.2 Prompt Templates for Alternatives Generation

Figure 8 shows the prompt templates used for generating alternatives for semantic units. The design of these alternative-generation prompts follows the implementation practice of *InfoTracer* (Qi et al., 2026). We use different prompts according to the type of semantic unit. For entity-level semantic units, the prompt asks the auxiliary model to generate conceptually similar expressions, such as entities of the same category or semantically comparable concepts. For word-level semantic units selected by NLTK part-of-speech tagging, the prompt asks the auxiliary model to generate three context-aware synonyms for the selected word, where the target word is marked with double asterisks in the passage. The generated alternatives are then combined with the original expression to form the candidate set for multiple-choice recovery.

Entity-Level Alternatives Generation Prompt

NER candidate-generation prompt

I would like to generate conceptually similar words for the given word. Below are a few examples of how this works:

Given: "America"
Similar Words: "England", "China", "France"

Given: "New York"
Similar Words: "Tokyo", "London", "Paris"

Given: "Hawaii"
Similar Words: "Fiji", "Maldives", "Bali"

Given: "Best Actor"
Similar Words: "Lead Performer", "Top Actor", "Outstanding Actor"

Given: "Luke Evans"
Similar Words: "Hugh Jackman", "Orlando Bloom", "Gerard Butler"

Given: "<target entity>"
Similar Words:

A.3 Auxiliary Model and Target Model

In our setting, the auxiliary model and the target model need not be identical because they serve distinct roles. The auxiliary model is provided with the correct semantic unit and relevant context, and is used only to generate several appropriate alternatives to the correct answer. In contrast, the target model receives the masked input and must identify the correct semantic unit from the candidate set. Because the auxiliary model is explicitly given the correct answer, whether it has previously memorized the target text is unlikely to introduce substantial confounding into the detection signal. We use *Llama-3-8B-Instruct* as the auxiliary model, following the alternative-generation setting of *InfoTracer* (Qi et al., 2026), to keep alternative generation comparable across methods and reduce the possibility that performance differences are driven by the use of a stronger auxiliary model. Future work will systematically evaluate how different auxiliary models affect detection performance.

Word-Level Alternatives Generation Prompt

NLTK word candidate-generation prompt

Please read the following passage, then generate three synonyms for the selected word. The selected word is enclosed between **. Below are a few examples of how this works:

Passage: Hurricane Ana was the second tropical cyclone in 2014 to **threaten** the U.S. state of Hawaii with a direct hit, after Iselle in August. The twenty-first named storm and fifteenth hurricane of the 2014 Pacific hurricane season, Ana formed from a disturbance that formed in the Central Pacific in mid-October. It rapidly consolidated, and a tropical depression developed by October 13. Aided by favorable

synonyms: "endanger", "menace", "jeopardize"

Passage: <passage with the selected word enclosed by **>

synonyms:

Figure 8: Prompt templates used for candidate generation. Blue blocks denote the prompt for named entities extracted by spaCy NER, while orange blocks denote the prompt for content words selected by NLTK part-of-speech tagging.

B Prompt-Induced Behavioral Decomposition

Figure 10 and Figure 9 show the prompt-induced recovery-rate distributions on seen texts. Across both datasets, the positive prompt consistently shifts recovery rates upward from the original prompt, whereas the negative prompt generally suppresses them. This effect is especially stable on *News*, where commercial LLMs exhibit a clear positive–original–negative ordering, and is weaker but still observable on *WikiMIA*. These results indicate that prompt induction can effectively modulate memorization-related recovery signals in exposed texts.

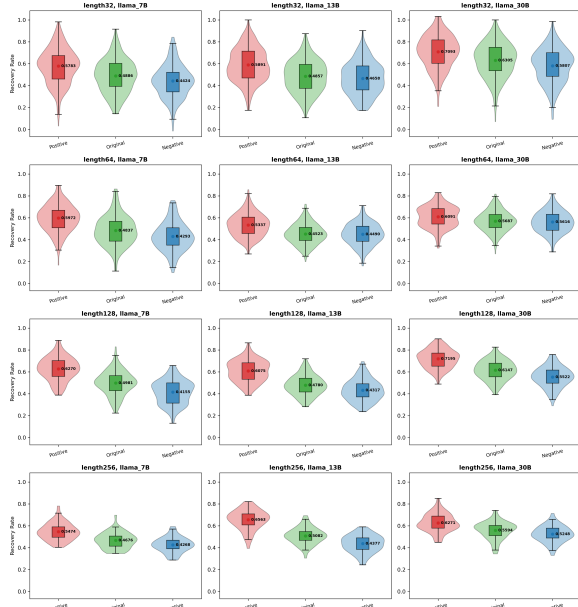


Figure 9: Recovery-Rate distributions over Seen texts under different prompts on *WikiMIA* Dataset.

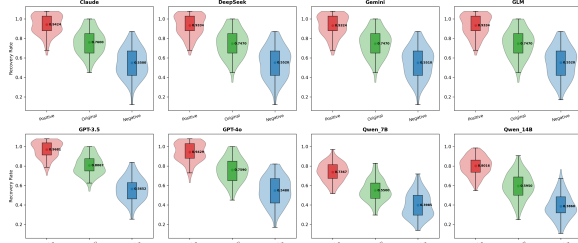


Figure 10: Recovery-Rate distributions over Seen texts under different prompts on *News* Dataset.

Figure 12 and Figure 11 show that prompt-induced differences are much weaker on unseen texts. On *News*, the positive and negative prompts only mildly increase or suppress recovery rates,

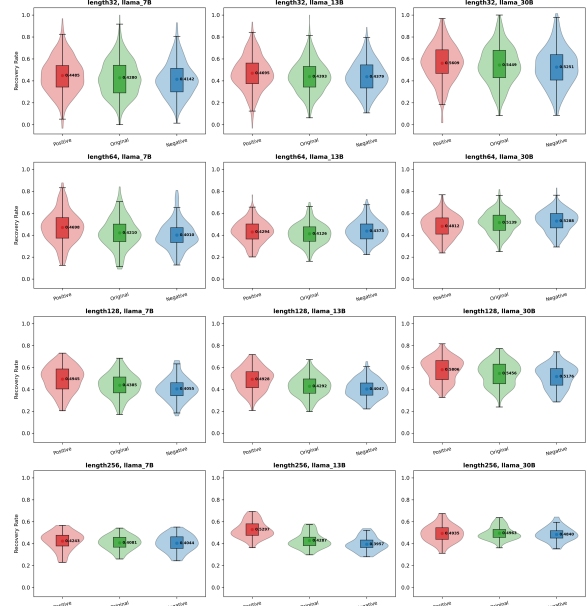


Figure 11: Recovery-Rate distributions over Unseen texts under different prompts on *WikiMIA* Dataset.

while on *WikiMIA* the three distributions often overlap and the negative prompt is not always below the original prompt. This weaker modulation helps explain the lower separability on *WikiMIA* and the relatively limited TPR of *PIRBD-PDD* in the very low-FPR region.

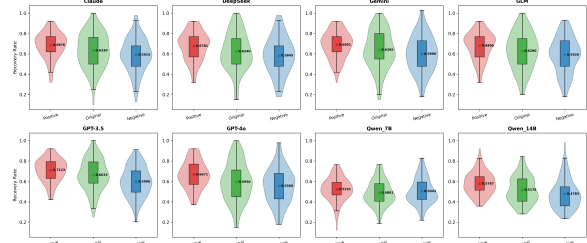


Figure 12: Recovery-Rate distributions over Unseen texts under different prompts on *News* Dataset.

C Balanced Accuracy

WikiMIA. Figure 13 reports the Balanced Accuracy results on *WikiMIA* across different LLaMA model scales and input lengths. Overall, **PIRBD-PDD** achieves the best or comparable performance in most settings, with its advantage becoming more evident on longer inputs. In particular, when the input length increases to 256, **PIRBD-PDD** consistently outperforms both *InfoTracer* and *PETAL* across LLaMA-7B, LLaMA-13B, and LLaMA-30B. This trend is consistent with the AUC results in the main text: longer inputs provide more semantic units, allowing **PIRBD-PDD** to identify more high-discriminability units and aggregate stronger exposure-specific evidence. These results further demonstrate the robustness of **PIRBD-PDD** across model scales and input lengths under a threshold-dependent metric.

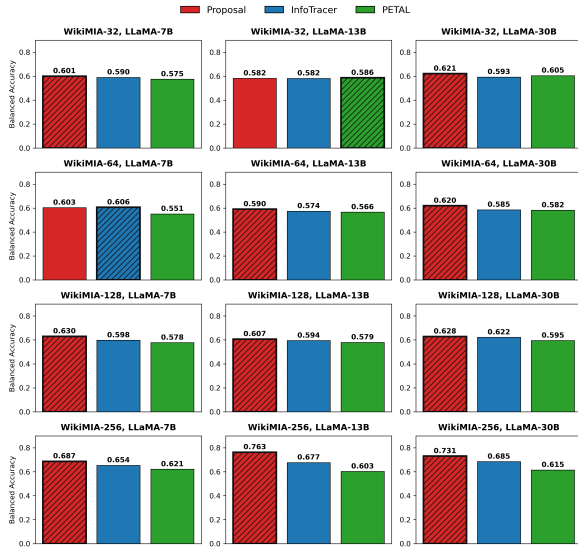


Figure 13: Balanced Accuracy on the *WikiMIA* dataset.

News. Figure 14 shows the Balanced Accuracy results on *News* across commercial LLMs and different set sizes $\mathcal{G}$. **PIRBD-PDD** consistently outperforms *InfoTracer* in most settings, and the performance gain generally increases as the set size becomes larger. When $\mathcal{G} = 5$, **PIRBD-PDD** achieves substantially higher Balanced Accuracy across all evaluated commercial models, reaching 0.964 on *GPT-3.5*, 1.0 on *GPT-4o*, and around or above 0.850 on several other models. The results confirm the superiority and robustness of **PIRBD-PDD** in realistic commercial black-box settings.

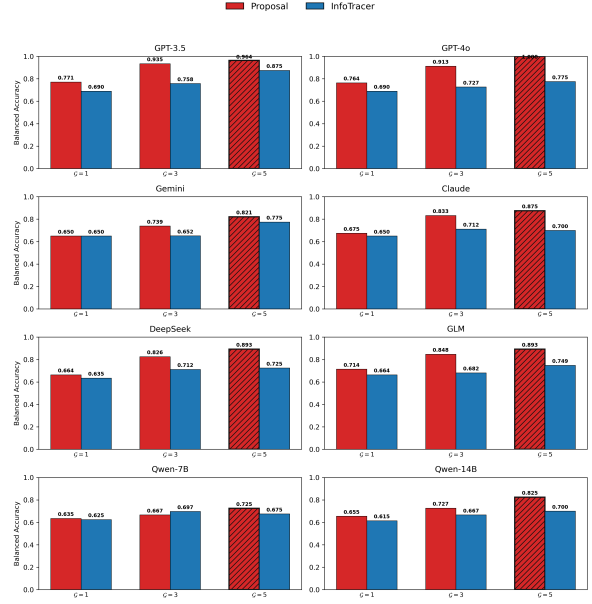


Figure 14: Balanced Accuracy on *News* under different set sizes.

D TPR@5%FPR

WikiMIA. Figure 15 presents the TPR@5%FPR results on *WikiMIA*. Compared with the AUC and Balanced Accuracy results, the performance under this stricter low-FPR metric is more mixed. **PIRBD-PDD** achieves competitive or superior results in several settings, especially for some longer-input cases such as *WikiMIA-256*, but it does not consistently dominate *InfoTracer* and *PETAL* across all model scales and input lengths. This suggests that although **PIRBD-PDD** improves the overall separability between seen and unseen texts, achieving high recall under an extremely low false-positive constraint remains challenging. A likely reason is that prompt-induced decomposition may not fully separate context-oriented recovery on unseen texts; consequently, some semantic units that can be recovered from local context are mistakenly selected as high-discriminability units, producing high detection scores for certain unseen texts and hurting performance in the low-FPR region.

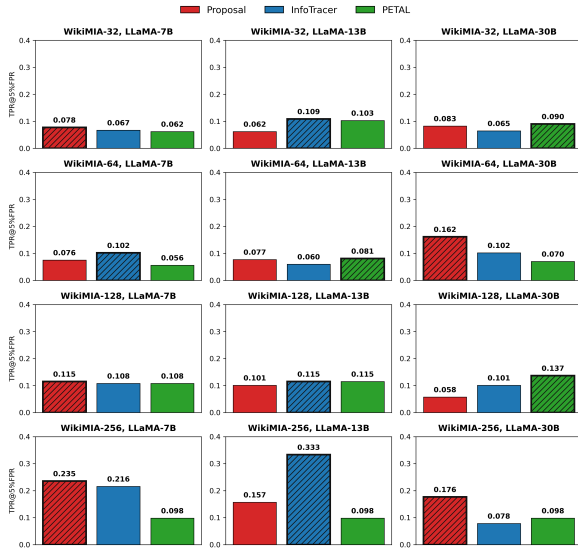


Figure 15: TPR@5%FPR on *WikiMIA* Dataset.

News. Figure 16 reports the TPR@5%FPR results on *News*. In general, **PIRBD-PDD** benefits from larger set sizes and achieves clear improvements over *InfoTracer* on several commercial LLMs when $G = 5$, including *GPT-3.5*, *GPT-4o*, *Gemini*, and *Claude*. However, the gains are less consistent than those observed for AUC and Balanced Accuracy, and *InfoTracer* remains stronger in some settings such as *Qwen* models. These results indicate that **PIRBD-PDD** provides

strong overall detection performance, while low-FPR detection remains a challenging direction for further improvement.

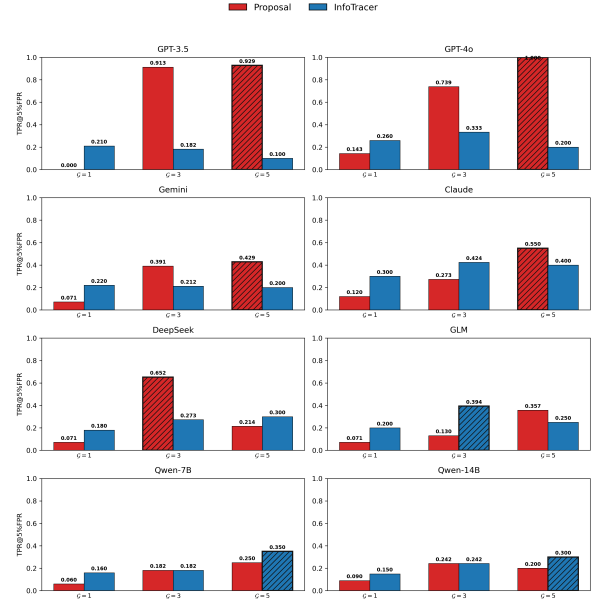


Figure 16: TPR@5%FPR on *News* under different set sizes.

E Effect of Candidate Sets

Candidate sets constrain the target model to a valid response space. Since instruction-following capabilities vary across LLMs, models with weaker instruction-following ability may not fully understand the task or generate the expected response; candidate sets can mitigate this issue and improve the reliability of recovery evaluation. To examine their effect, we conduct a small-scale ablation in which the target model is asked to recover each masked semantic unit without being provided with candidate options. As shown in Table 3, removing candidate sets reduces AUC across both target-model sizes and text lengths, with the relative decreases shown in parentheses. Moreover, the proportion of semantic units discarded for not following the required output format increases from 2.68% to 9.06% at length 32 and from 2.18% to 8.59% at length 128. Overall, these findings support using candidate sets as a practical mechanism for stabilizing recovery evaluation.

Len.	<i>LLaMA-7B</i>	<i>LLaMA-13B</i>
32	0.596 (↓ 1.174%)	0.579 (↓ 1.352%)
128	0.633 (↓ 2.119%)	0.633 (↓ 0.616%)

Table 3: AUC results on *WikiMIA* without candidate sets. Parentheses indicate the relative decrease from the standard candidate-set setting.

F Computational Cost

The computational cost of **PIRBD-PDD** depends mainly on five factors: the number of semantic units extracted from the target text, the per-query cost of the auxiliary model used to generate alternatives, the number of alternatives per unit, the per-query cost of the target model, and the number of target-model queries per unit. *InfoTracer* incurs similar alternative-generation and recovery costs but additionally queries a commercial LLM, such as *GPT-4o*, to assess the importance of each semantic element. *PETAL* instead combines target- and surrogate-model queries, semantic-encoder inference, and regression-model fitting.

The resulting per-query cost also depends on the deployment setting. With open-source target models, it mainly reflects local computation, such as GPU time; with commercial models, it additionally includes API charges and request-processing time. To make these cost structures directly comparable, let L denote the target-text length, U the number of semantic units extracted by **PIRBD-PDD**, and E the number of semantic elements evaluated by *InfoTracer*. Table 4 summarizes the dominant operations of each method; actual wall-clock time and monetary cost also depend on the execution environment.

Method	Auxiliary/ Surrogate Model	Target Model	Commercial LLM	Semantic Encoder	Additional Cost
<i>PIRBD-PDD</i>	U queries	$3U$ queries	–	–	–
<i>InfoTracer</i>	E queries	E queries	E queries	–	–
<i>PETAL</i>	$L/2$ queries	$L/2$ queries	–	$L/2$ inferences	Regression-model fitting

Table 4: Theoretical computational-cost comparison.

We further examine practical efficiency by measuring the locally executed end-to-end per-sample runtime of the three methods on *WikiMIA* across two text lengths and two target-model scales using a single NVIDIA A100 80GB GPU. The measurements include all locally executed method-specific preprocessing and inference. As shown in Table 5, **PIRBD-PDD** is faster than *PETAL* in all four settings, while requiring more local runtime than *InfoTracer*. Importantly, the reported *InfoTracer* runtime excludes the request-processing time and monetary cost of its commercial-LLM API queries and therefore does not represent its complete deployment cost. More broadly, prior work has primarily emphasized detection performance and threshold robustness, with less systematic attention to computational efficiency. Nevertheless, computational cost remains an important practical consideration for deployment.

Target model	Length	Methods		
		<i>PIRBD-PDD</i>	<i>InfoTracer</i>	<i>PETAL</i>
LLaMA-7B	32	6.29	2.17	24.55
LLaMA-7B	128	12.37	4.59	35.52
LLaMA-13B	32	7.54	3.14	28.13
LLaMA-13B	128	19.87	8.83	40.97

Table 5: Per-sample runtime on *WikiMIA* (seconds).

G Pseudo Code

Algorithm 1 Semantic Unit Extraction

Require: Input text x

Ensure: Semantic unit set $\mathcal{S} = \{s_i\}_{i=1}^M$

- 1: Tokenize x into a sequence of tokens $(x_1, \dots, x_T)$
 - 2: Initialize $\mathcal{S} \leftarrow \emptyset$
 - 3: Extract named-entity spans from x
 - 4: Add each named-entity span to $\mathcal{S}$ as a semantic unit
 - 5: Obtain POS tags for tokens in x
 - 6: **for** $t = 1$ to T **do**
 - 7: **if** x_t is not covered by a named entity and satisfies the semantic-unit criteria **then**
 - 8: Add x_t to $\mathcal{S}$ as a semantic unit
 - 9: **end if**
 - 10: **end for**
 - 11: Discard function words and purely syntactic tokens from $\mathcal{S}$
 - 12: **return** $\mathcal{S} = \{s_i\}_{i=1}^M$
-

Semantic Unit Extraction. We use NLTK for tokenization and part-of-speech tagging, which segment the input into tokens and assign token-level word classes, and spaCy for named entity recognition, which identifies labeled contiguous spans such as people, organizations, and locations. These tools support word-level and entity-level semantic-unit extraction, respectively. We adopt them primarily to match the preprocessing setup of *InfoTracer* (Qi et al., 2026) and thereby maintain a fair comparison. Because tokenization, part-of-speech tagging, and entity-boundary predictions determine which semantic units enter candidate generation and detection scoring, different preprocessing choices may affect detection performance. Systematically comparing extraction algorithms is beyond the primary objective of this work, and we leave such an analysis for future work.