# INFERRING LARGE LANGUAGE MODEL PROVENANCE VIA TOKEN RANKING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Model provenance detection determines whether a model was derived from a claimed parent through fine-tuning or other adaptations. Many methods infer provenance relationships by measuring the similarity between two models from their generated responses. However, these approaches rely solely on sampled tokens, leaving the rich information associated with alternative candidate tokens unexplored. In this paper, we propose **RankMPT**, a novel method that leverages the rank information of candidate tokens to detect model provenance. Specifically, RankMPT measures the average rank correlation between the tail candidate tokens of two models across the prompt tokens, which can be approximated via repeated high-temperature sampling in black-box settings. We then perform a hypothesis test to decide whether the score is significantly higher than the correlation level expected in the absence of a provenance relationship. Our method captures discriminative signals from rarely selected tail tokens, making it less sensitive to confounding similarities among models arising from shared domains or tasks. Extensive experiments demonstrate the superiority of RankMPT, improving the average accuracy by up to 23% in the black-box setting of LeafBench.

## 1 INTRODUCTION

The rapid proliferation of large language models (LLMs) has led to an increasingly complex ecosystem in which models are frequently reused through fine-tuning, distillation, and other adaptation techniques (Sun et al., 2024; Mai et al., 2024). These derived models can be readily distributed and deployed through platforms such as Hugging Face and GitHub with limited oversight, raising concerns about unauthorized model reuse and the security of model supply chains (Yoon et al., 2025; Shang et al., 2026). This underscores the importance of *model provenance detection*, which determines whether a model has been derived from a claimed parent model and enables tracing the source of a suspected model (Yu & Wang, 2025; Shi et al., 2025; Xu et al., 2026).

To address this problem, existing methods typically devise a pairwise similarity score from model responses to distinguish provenance-related pairs from unrelated ones (Shang et al., 2026; Shi et al., 2025; Ren et al., 2025). For example, Nikolic et al. (2025) compute next-token textual matching while Wu et al. (2025) measure response-level semantic similarity. The assumption behind this is that a derived model tends to generate outputs similar to those of its parent models. However, those sampled tokens can be unreliable for provenance detection, since two independent medical diagnosis models may produce very similar outputs simply due to shared domain constraints (Zhang & Zheng, 2025). This motivates us to leverage rich information from candidate tokens to detect provenance.

In this paper, we tackle this issue by systematically exploring the token distribution structures of two models under the same inputs, revealing that the token rank exhibits a particularly distinctive signal for provenance(see Figure 1). Formally, we present **RankMPT**, a novel method that distinguishes provenance by utilizing the informative structures extracted from token distributions. Specifically, given logit access, RankMPT measures the rank correlation between two models over candidate tokens shared in their low-logit tails. Without logit access, this correlation can be approximated from token frequencies under high-temperature sampling, enabling black-box provenance detection. It then applies an asymptotic test to determine whether the observed similarity significantly exceeds the level expected under no provenance relationship. Our method captures discriminative structures



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

in the disfavor candidate tokens that are less likely to be selected by the model, enabling more robust to confounding similarities arising from data distribution and training objective.

We evaluated our method against representative white-box and black-box provenance baselines across two public benchmarks, *BenchA* with 110 LLMs and *LeafBench* with 65 LLMs, collectively spanning 15 mainstream model families and 6 types of model transformations. The results show that our method achieves the best or tied-best accuracy across diverse model families and transformation in black-box and white-box settings(Section 5.2). Our analysis further demonstrates that the rank structures in low-logit tail tokens remain better discriminative than those in highly preferred tokens (Section 5.3). Finally, additional evaluations in challenging pairs where two models shows highly behavioral similar but without derivation provenance, demonstrate its robustness in distinguishing provenance from confounding similarities (Section 6). Our code and data are available at `https://anonymous.4open.science/r/RankMPT-9B4E`.

Our key contributions are summarized as follows:

1. We reveal a distinctive rank-based pattern in token selection that distinguishes provenance-related model pairs from unrelated ones, providing a discriminative signal for provenance.
2. We propose **RankMPT**, a novel method that introduces a rank-based correlation measure for provenance detection. By exploiting rarely selected candidate tokens, RankMPT is more robust to confounding factors such as data distributions and training objectives.
3. We conduct extensive experiments to show that RankMPT can improve provenance detection across diverse model families and adaptation mechanisms. Further evaluation on behaviorally similar but unrelated models confirms its robustness for provenance.

## 2 Related Work

The problem of model provenance is related to designing schemes that verify model ownership. Both aim to identify relationships between models. We review these two lines of work below.

**Model provenance detection.** Existing methods mainly design similarity scores to distinguish model pairs with provenance relationships from unrelated pairs. Based on the required model access, they can be broadly divided into white-box and black-box methods. *White-box* methods rely on access to model parameters or internal representations, and construct scores from attention statistics (Yoon et al., 2025), parameter similarities or invariants (Zeng et al., 2024), hidden embeddings (Zhang et al., 2025), or selected dimensions of representations (Tong et al., 2026).These methods are often designed to capture broader lineage relationships, such as shared model families or backbones, with limited applicability to closed-source models. *Black-box* methods require only query access for broader applicability. They infer provenance from generated outputs, for example, using next-token agreement (Nikolic et al., 2025), semantic or functional features (Wu et al., 2025; Kuditipudi et al., 2025; Shi et al., 2025), or chain-of-thought features (Ren et al., 2025). However, such sampled tokens may not reliably distinguish provenance since independent models can produce similar responses to common prompts due to shared domain constraints. Our work instead exploits richer information from unselected tokens to derive a more discriminative measure of provenance.

**Model ownership verification.** This task aims to prove that a model belongs to a particular owner by establishing a distinctive identity, mainly through watermarking or fingerprinting. *Watermarking* embeds predefined signatures into a model, such as backdoor triggers or generation biases, and verifies ownership by recovering these signatures (Peng et al., 2023; Zhang et al., 2024; Yang et al., 2025; Kirchenbauer et al., 2023; Dathathri et al., 2024). It requires intervention during model development and may be weakened by subsequent fine-tuning (Nasery et al., 2025). *Fingerprinting* instead identifies a model without modifying training by constructing fingerprints, such as using curated prompts to elicit model-specific behavior (Hu et al., 2025; Pasquini et al., 2025; Chen et al., 2026), or by analyzing model characteristics on general-purpose tasks through hidden representations (Zhang et al., 2025; Zeng et al., 2024) or generated outputs (Wu et al., 2025; Ren et al., 2025). These methods focus on building an identity for a single model rather than capturing pairwise evidence for derivation, yet making them unsuitable for provenance detection. For example, two base models developed by the same organization may have very similar identities. Our work instead targets pairwise signals that can reflect a derivation relationship between two models.

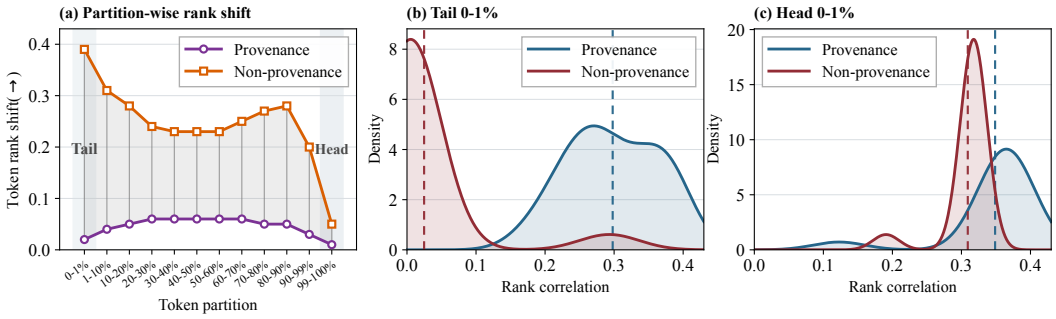


Figure 1: **Tail-token rank structure more clearly reveals model provenance**. We evaluate on 15 provenance pairs $\{C_i, C_j\}_{i=1}^{15}$ and 15 non-provenance pairs $\{C_i, C_j\}_{i=1}^{15}$, where all $C_i$ are fined-tuned on the same task(see details in Appendix A.1). **(a)** Average token-rank shift between two models (Finlayson et al., 2026), where lower values indicate stronger rank consistency, across token partitions ordered by parent-model logits from low-logit tokens (Tail) to high-logit tokens (Head). The largest shift gap between provenance and non-provenance pairs occurs in the tail. **(b)** Distribution of rank correlations over the 1% lowest-logit shared tokens and **(c)** the 1% highest-logit shared tokens. Provenance and non-provenance correlation distributions are clearly separated in the tail but substantially overlap in the head, highlighting the stronger discriminability of tail-token rankings.

## 3 Preliminaries

We first recall the prediction space of a large language model. Given a model $M$ with vocabulary $\mathcal{V}_M$ and a tokenized input sequence $x = (x_1, \ldots, x_T) \in \mathbb{R}^T$ with max length $T$, at each prediction position $t \in T$, the model $M$ maps the preceding context to a logit vector

$$\boldsymbol{\ell}_t^M = \left[\ell_t^M(v)\right]_{v \in \mathcal{V}_M} \in \mathbb{R}^{|\mathcal{V}_M|}, \tag{1}$$

where $\ell_t^M(v)$ denotes the logit assigned to candidate token $v \in \mathcal{V}_M$, with larger logits corresponding to higher next-token probabilities. During auto-regressive generation, a decoding rule selects a single token from this prediction distribution at each step, typically favoring high-probability candidates. Beyond the selected token, each prediction therefore contains a $|\mathcal{V}_M|$-dimensional preference structure over alternative token candidates.

**Problem statement.** Model provenance detection can be formulated as a binary classification problem. A pair of models is considered a *provenance pair* if they exhibit a direct parent–child relationship established through an adaptation process. Otherwise, the pair is regarded as *non-provenance pair*. Given a suspected child model $C$, a potential parent model $P$, and a set of query prompts $\mathcal{X} = \{x_i\}_{i=1}^n$, the goal is to construct similarity score $S(\cdot)$ and map it into a binary decision:

$$g(P, C; \mathcal{X}) = \begin{cases} \text{provenance}, & \text{if } S(P, C; \mathcal{X}) > S_0, \\ \text{non-provenance}, & \text{otherwise .} \end{cases} \tag{2}$$

where $S_0$ is typically a tuned threshold or a critical value under the non-provenance null hypothesis. The key challenge is to design a score function $S(\cdot)$ that captures inherited properties rather than generic similarity induced by shared architectures, training data, or tasks. Prior works primarily rely on model outputs for similarity estimation, capturing what the model prefers to generate. However, generated tokens provide only limited information and may be unreliable for provenance detection. For example, two independently trained medical LLMs may produce similar outputs under shared domain constraints (Zhang & Zheng, 2025). In this work, we consider the richer information in candidate tokens to construct a more discriminative score, which we describe in the next section.

## 4 Methodology

In this section, we introduce RankMPT, which leverages rank information among low-preference token candidates to construct a provenance score and determines provenance by testing whether the score is significantly higher than expected for non-provenance pairs. The overall procedure is summarized in Algorithm 1, and we describe each component below.

**Tail-Token Ranking Score.** We introduce a provenance score based on the ranking consistency of low-logit candidate tokens, which we refer to as *tail tokens*. Our key observation is that tail-token rankings distinguish provenance pairs from non-provenance pairs much more clearly than high-preference tokens (see Figure 1). This is consistent with previous findings that model adaptation primarily reshapes high-confidence token preferences (Meng et al., 2026; Wang et al., 2026; Mai et al., 2024), suggesting that tail tokens are more likely to preserve the structure inherited from the parent model. We provide a systematic empirical analysis in Appendix A.2.

In particular, with logit access, for each model $M \in \{P, C\}$, we compute the mean-logit vector $\bar{\boldsymbol{\ell}}^M = \frac{1}{n}\sum_{i=1}^{n} \bar{\boldsymbol{\ell}}_i^M$, where $\bar{\boldsymbol{\ell}}_i^M$ is either the average logit vector over all prediction positions of prompt $x_i$ or the logit vector at its last prediction position. We then select the tail $\gamma$ fraction of tokens by sorting $\bar{\boldsymbol{\ell}}^M$ in ascending order:

$$\mathcal{I}_M = \operatorname{argsort}^{\uparrow}_{v \in \mathcal{V}_M} \bar{\ell}^M(v)\,[1 : \lceil \gamma |\mathcal{V}_M| \rceil]\,. \tag{3}$$

We use the mean logit vector to extract rank correlation signals, as it provides a more stable and noise-robust alternative to per-sample argmin frequency. We take their intersection across the two models as the *shared tail-token set* $\mathcal{I} := \mathcal{I}_P \cap \mathcal{I}_C$. Intuitively, provenance-related models are expected to preserve similar token rankings within this set through model adaptation, whereas independently trained models exhibit no such consistently inherited alignment.

Restricting the candidate tokens of both models to the tail tokens $\mathcal{I}$, we compare their token rankings across query prompts. For each prompt $x_i$, we compute the Spearman rank correlation:

$$\rho_i = \text{Spearman}\left(\bar{\boldsymbol{\ell}}_i^P[\mathcal{I}], \bar{\boldsymbol{\ell}}_i^C[\mathcal{I}]\right). \tag{4}$$

This yields multiple observable agreement scores $\{\rho_i\}_{i=1}^n$, characterizing how consistently the two models agree across prompts. Models empirically exhibit higher scores with their true parents than with unrelated models, making ranking agreement a discriminative signal for provenance.

Notably, the ranking agreement can also be approximated under **black-box access**, since it depends on relative token order rather than exact logit values. Inspired by prior work (Sam et al., 2026), when logits are unavailable, we alternatively recover token rankings through repeated high-temperature sampling. Specifically, empirical token frequencies are used as surrogates for logits, as a higher temperature increases the observability of low-probability tokens while preserving their relative ordering. We then compute the rank correlation using the empirical frequency vectors. Due to sampling noise, finite samples may not recover the same tail tokens as under white-box access, potentially reducing discriminability. We evaluate the resulting loss in discriminability in Section 5.

**Provenance testing.** Given the prompt-wise alignment scores $\{\rho_i\}_{i=1}^n$, a straightforward approach is to classify a model pair as provenance when its average score exceeds a fixed threshold $\rho_0$. However, such thresholding ignores the variability of alignment across prompts and can therefore be sensitive to the queried samples. Instead, we test whether the observed alignment is significantly higher than that expected for non-provenance pairs.

Although unrelated models may exhibit weak correlations due to shared architectures, training data, or other common factors, we find that their correlations remain consistently low across model families and prompt domains (see Appendix B). This provides a stable upper bound on incidental agreement. We therefore use $\rho_0$ as a conservative upper bound under non-provenance and formulate

$$H_0 : \mathbb{E}[\rho_i] \le \rho_0, \qquad H_1 : \mathbb{E}[\rho_i] > \rho_0. \tag{5}$$

Since correlation coefficients are bounded and the scores of provenance pairs often concentrate near the upper boundary, their differences are severely compressed. We therefore apply a Fisher transformation $z_i = \operatorname{arctanh}(\rho_i)$, which maps the bounded correlations to an unbounded scale, providing a better-behaved scale for aggregation and studentization (Fieller et al., 1957). Accordingly, we map the reference correlation to the same scale as $z_0 = \operatorname{arctanh}(\rho_0)$ and aggregate the transformed prompt-wise scores using the studentized statistic

$$Z_{P,C} = \frac{\bar{z} - z_0}{s_z/\sqrt{N}}, \tag{6}$$

where $\bar{z}$ and $s_z$ denote the sample mean and standard deviation of $\{z_i\}_{i=1}^N$. Under the null boundary, $Z_{P,C}$ is asymptotically standard normal; we provide the proof in Appendix C. We therefore obtain

**Algorithm 1** Provenance Testing via Token Rank Correlation

**Require:** model pair $(P, C)$; prompts $\mathcal{X}$; access mode $b \in \{\mathrm{WB}, \mathrm{BX}\}$; token ratio $\gamma$; prefixes per prompt $K$ and samples per prefix $S$ ; background level $\rho_o$; significance level $\alpha$

▶ ***Step 1: Localize shared tail tokens***

**if** $b = \mathrm{WB}$ **then**
  Obtain the full logit tensors $\boldsymbol{\ell}^P, \boldsymbol{\ell}^C$ over $\mathcal{X}$
  Compute the mean-logit vectors $\bar{\boldsymbol{\ell}}^P, \bar{\boldsymbol{\ell}}^C$ over $\mathcal{X}$
  $\mathcal{I} \leftarrow \mathrm{Tail}_{-\gamma\%}(\bar{\boldsymbol{\ell}}^P) \cap \mathrm{Tail}_{-\gamma\%}(\bar{\boldsymbol{\ell}}^C)$
**else**
  $\hat{\mathcal{I}}^P, \hat{\mathcal{I}}^C \leftarrow$ tokens of $P, C$ from $K \times S$ generations per $x_n$ over $\mathcal{X}$
  $\hat{\boldsymbol{\ell}}^P, \hat{\boldsymbol{\ell}}^C \leftarrow$ empirical frequencies of generated tokens per prompt in $\mathcal{X}$
  $\boldsymbol{\ell}^P, \boldsymbol{\ell}^C \leftarrow \hat{\boldsymbol{\ell}}^P, \hat{\boldsymbol{\ell}}^C$ and $\mathcal{I} \leftarrow \hat{\mathcal{I}}^P \cap \hat{\mathcal{I}}^C$
**end if**

▶ ***Step 2: Obtain prompt-wise Spearman scores***

**for** each input $x_i \in \mathcal{X}$ **do**
  $\rho_i \leftarrow$ mean Spearman correlation across aligned positions of $\boldsymbol{\ell}_i^P[\mathcal{I}]$ and $\boldsymbol{\ell}_i^C[\mathcal{I}]$
  $z_i \leftarrow \mathrm{arctanh}(\rho_i)$
**end for**

▶ ***Step 3: Hypothesis test against null***

Compute the one-sided z-score $Z_{P,C}$ by aggregating $\{z_i\}_{i=1}^{|\mathcal{X}_{\mathrm{test}}|}$
$p_{P,C} \leftarrow 1 - \Phi(Z_{P,C})$
**return** $\widehat{y} = \mathbf{1}(p_{P,C} < \alpha)$

the one-sided $p$-value $p_{P,C} = 1 - \Phi(Z_{P,C})$, where $\Phi(\cdot)$ is the standard normal cumulative distribution function. We declare $(P, C)$ a provenance pair when $p_{P,C} < \alpha$. This testing formulation accounts for both the magnitude and variability of the prompt-wise alignment scores, providing a more reliable provenance decision than direct thresholding.

**Rationale of RankMPT** Our operating hypothesis is that models with provenance relationships exhibit stronger agreement in which tokens are likely or unlikely to be predicted than unrelated models. This motivates the use of rank correlation, as absolute token probabilities are less reliable for cross-model comparison. Our empirical analysis in Appendix A shows that such rank-based correlations provide clearer separation between provenance and non-provenance pairs than numerical statistics, particularly when restricted to tail tokens. RankMPT therefore builds on two properties: first, the relative ordering of token candidates forms a discriminative preference structure and can be approximated from black-box outputs without access to exact values. Second, it focuses on the ranking structure of tail tokens, which are less directly shaped by task-driven predictions, making the resulting scores more indicative of provenance and less dominated by data-induced similarities.

## 5 Experiments

In this section, we first describe the experimental setup in Section 5.1, then evaluate RankMPT across diverse model derivations and parent families in Section 5.2. Section 5.3 further studies the decision rule and sensitivity to token selection, sampling budget, and prompt count.

### 5.1 Experimental Setup

**Benchmarks and Prompt Sets.** We evaluate on two public real-world provenance benchmarks, *BenchA* (Nikolic et al., 2025) and *LeafBench* (Shao et al., 2025). BenchA contains 110 LLMs from 10 families with 100 positive parent–child pairs and 100 negatives, focusing on general fine-tuning derivations, while LeafBench contains 65 LLMs from 7 families with 59 positive pairs spanning instruction tuning (*Instruction*), general fine-tuning (*General FT*), parameter-efficient fine-tuning (*PEFT*), quantization (*Quantization*), model merging (*Merging*), and knowledge distillation (*Distillation*). For prompts, we integrate three widely used datasets: RandomPrompt (Nikolic et al., 2025), CommonSenseQA (Talmor et al., 2019), and Math500 (Lightman et al., 2024), with 500 sampled instances from each. Appendix D.1 summarizes their prior use in provenance methods.

Table 1: Accuracy of different methods on LeafBench (Shao et al., 2025) by derivation types. Parentheses show gaps from the strongest baseline. "Overall" is full-benchmark accuracy; best in bold.

| Types | # Pairs | White-box Methods | | | | Black-box Methods | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | REEF | Intrinsic | SeedPrints | RankMPT-WB | CoTRSF | LLMDNA | NT-FWER | RankMPT-BX |
| Instruction | 7 | 0.80±0.03 | 1.00±0.00 | 1.00±0.01 | 1.00±0.01 (-0.00) | 0.60±0.03 | 0.20±0.03 | 0.60±0.03 | 1.00±0.01 (+0.40) |
| General FT | 18 | 0.61±0.01 | 0.92±0.00 | 0.85±0.02 | 1.00±0.01 (+0.08) | 0.62±0.02 | 0.15±0.02 | 0.31±0.02 | 0.92±0.01 (+0.30) |
| PEFT | 9 | 1.00±0.01 | 0.63±0.00 | 1.00±0.01 | 1.00±0.01 (+0.00) | 1.00±0.01 | 0.50±0.03 | 0.63±0.02 | 0.88±0.02 (-0.12) |
| Quantization | 9 | 0.86±0.02 | 0.43±0.00 | 0.71±0.02 | 0.92±0.01 (+0.06) | 0.71±0.02 | 0.29±0.02 | 0.57±0.03 | 0.86±0.02 (+0.15) |
| Merging | 9 | 0.88±0.02 | 1.00±0.00 | 0.88±0.02 | 0.88±0.02 (-0.12) | 0.50±0.03 | 0.00±0.00 | 0.63±0.02 | 0.88±0.02 (+0.25) |
| Distillation | 7 | 0.95±0.02 | 0.93±0.01 | 1.00±0.01 | 1.00±0.01 (+0.00) | 0.50±0.03 | 0.00±0.00 | 0.50±0.03 | 0.83±0.02 (+0.33) |
| **Overall** | 59 | 0.94±0.01 | 0.82±0.00 | 0.94±0.01 | **0.98±0.01 (+0.04)** | 0.66±0.02 | 0.19±0.02 | 0.51±0.02 | **0.89±0.01 (+0.23)** |

Table 2: Accuracy of different methods on BenchA (Nikolic et al., 2025) by claimed parent model, with 10 positive and 10 negative pairs each. Parentheses show differences from the strongest baseline. "Overall" denotes full-benchmark accuracy, best in bold.

| Models | White-box Methods | | | | Black-box Methods | | | |
|---|---|---|---|---|---|---|---|---|
| | REEF | Intrinsic | SeedPrints | RankMPT-WB | CoTRSF | LLMDNA | NT-FWER | RankMPT-BX |
| Qwen2-1.5B | 0.84±0.02 | 0.74±0.03 | 0.88±0.02 | 0.96±0.02 (+0.08) | 0.52±0.03 | 0.57±0.02 | 0.70±0.03 | 0.84±0.02 (+0.14) |
| Qwen2.5-1.5B | 0.80±0.02 | 0.64±0.03 | 1.00±0.00 | 0.97±0.02 (-0.03) | 0.52±0.02 | 0.63±0.03 | 0.84±0.02 | 0.80±0.03 (-0.04) |
| TinyLlama-1.1B | 0.89±0.02 | 0.92±0.02 | 1.00±0.00 | 1.00±0.00 (+0.00) | 0.58±0.03 | 0.63±0.02 | 0.88±0.02 | 0.89±0.02 (+0.01) |
| DS-Coder-1.3B | 0.75±0.03 | 0.75±0.02 | 1.00±0.00 | 1.00±0.00 (+0.00) | 0.70±0.03 | 0.50±0.03 | 0.50±0.02 | 0.75±0.03 (+0.05) |
| Gemma-2-2B | 0.94±0.02 | 0.82±0.02 | 0.91±0.02 | 0.99±0.01 (+0.05) | 0.74±0.03 | 0.59±0.02 | 0.61±0.03 | 0.94±0.02 (+0.20) |
| Gemma-2B | 0.90±0.02 | 0.80±0.03 | 0.89±0.02 | 1.00±0.00 (+0.10) | 0.44±0.03 | 0.60±0.02 | 0.68±0.03 | 0.90±0.02 (+0.22) |
| Llama-3.2-1B | 0.83±0.02 | 0.71±0.03 | 0.92±0.02 | 0.96±0.02 (+0.04) | 0.68±0.03 | 0.71±0.02 | 0.92±0.02 | 0.83±0.03 (-0.09) |
| Llama-3.2-3B | 0.80±0.03 | 0.73±0.02 | 0.92±0.02 | 0.95±0.02 (+0.03) | 0.50±0.03 | 0.64±0.03 | 1.00±0.00 | 0.80±0.02 (-0.20) |
| Phi-3-mini-4k | 0.90±0.02 | 0.92±0.02 | 0.77±0.03 | 0.97±0.02 (+0.05) | 0.31±0.03 | 0.58±0.03 | 0.64±0.02 | 0.90±0.02 (+0.26) |
| Phi-2 | 0.76±0.03 | 0.83±0.02 | 0.88±0.02 | 0.96±0.02 (+0.08) | 0.56±0.03 | 0.65±0.02 | 0.78±0.03 | 0.76±0.03 (-0.02) |
| **Overall** | 0.84±0.01 | 0.78±0.02 | 0.92±0.01 | **0.98±0.01 (+0.06)** | 0.56±0.02 | 0.61±0.02 | 0.76±0.02 | **0.84±0.02 (+0.08)** |

**Baselines.** We compare our method against representative white-box and black-box provenance methods. White-box baselines include ***Intrinsic*** (Yoon et al., 2025) and ***REEF*** (Zhang et al., 2025), which analyze attention-parameter correlations and internal-representation similarity via centered kernel alignment, respectively, both using fixed-threshold classification; and ***SeedPrints*** (Tong et al., 2026) that computes Kendall's $\tau$ correlations on embeddings and uses a one-sided $z$-test for provenance. Black-box baselines include ***NT-FWER*** (Nikolic et al., 2025), which compares next-token outputs against a set of reference model and controls the Family-Wise Error Rate (FWER) across multiple $z$-test; ***LLMDNA*** (Wu et al., 2025), which encodes generated outputs and classifies them with a trained support vector machine; and ***CoTSRF*** (Ren et al., 2025), which uses Kullback–Leibler divergence between chain-of-thought feature distributions with parent-wise thresholding.

**Implementation Details.** For clarity, we denote the white-box and black-box variants of our method as ***RankMPT-WB*** and ***RankMPT-BX***, respectively. For correlation computation, we set $m = 0.1\%$ and max_input_length = 512 for RankMPT-WB; use $K = 8$ and $S = 96$ with temperature = 10.0, max_new_tokens =1 and top-$k$ = 0 for RankMPT-BX. Other settings are default. For decision, we set $\rho_o = 0.1$ and $\alpha = 0.05$. These settings are used unless otherwise specified. We report Accuracy for classification performance, and the Area Under the ROC Curve (AUC) and Kolmogorov–Smirnov statistic (KS) for score-level discriminability. All metrics are averaged over three runs, with standard deviations reported accordingly. Baseline implementations are detailed in Appendix D.2.

### 5.2 Main Results

**RankMPT is effective across various derivation techniques.** We first test whether the tail-ranking signal survives derivation techniques that can alter a model to varying degrees, e.g., fine-tuning and PEFT. Table 1 reports the detection rate on LeafBench provenance pairs, grouped by six derivation types. RankMPT-WB and RankMPT-BX achieve overall detection rates above 0.8 for every derivation type. Each variant is best or tied-best on five of them, with the clearest advantage on merging and distillation. These results indicate that the tail-ranking structure of child models retains substan-

Table 3: Score separability between negative and positive pairs for different methods on BenchA. The best values are in bold.

| Methods | Metrics | |
|---|---|---|
| | AUC↑ | KS↑ |
| **White-box** | | |
| REEF | 0.850 ± 0.023 | 0.650 ± 0.031 |
| Intrinsic | 0.930 ± 0.015 | 0.860 ± 0.017 |
| SeedPrints | **0.980 ± 0.025** | **0.920 ± 0.044** |
| RankMPT-WB | 0.970 ± 0.003 | **0.930** ± 0.028 |
| **Black-box** | | |
| CoTRSF | 0.550 ± 0.042 | 0.140 ± 0.013 |
| LLMDNA | 0.680 ± 0.073 | 0.360 ± 0.057 |
| NT-FWER | 0.790 ± 0.016 | 0.470 ± 0.053 |
| RankMPT-BX | **0.900 ± 0.014** | **0.740 ± 0.008** |

Table 4: Accuracy of our correlation scoring under different decision strategies. "-" indicates results omitted due to positive-only pairs.

| Variants | Benchmarks | |
|---|---|---|
| | BenchA | LeafBench |
| **White-box** | | |
| +Fixed threshold $\rho_o$ | 0.846 ± 0.020 | 0.921 ± 0.022 |
| +Optimal threshold | 0.962 ± 0.022 | - |
| +Multiple test w. FWER | 0.890 ± 0.011 | 0.854 ± 0.013 |
| +Studentized test (Ours) | **0.978 ± 0.017** | **0.978 ± 0.007** |
| **Black-box** | | |
| +Fixed threshold $\rho_o$ | 0.819 ± 0.017 | 0.831 ± 0.049 |
| +Optimal threshold | 0.839 ± 0.006 | - |
| +Multiple test w. FWER | 0.774 ± 0.031 | 0.695 ± 0.058 |
| +Studentized test (Ours) | **0.842** ± 0.005 | **0.893 ± 0.008** |

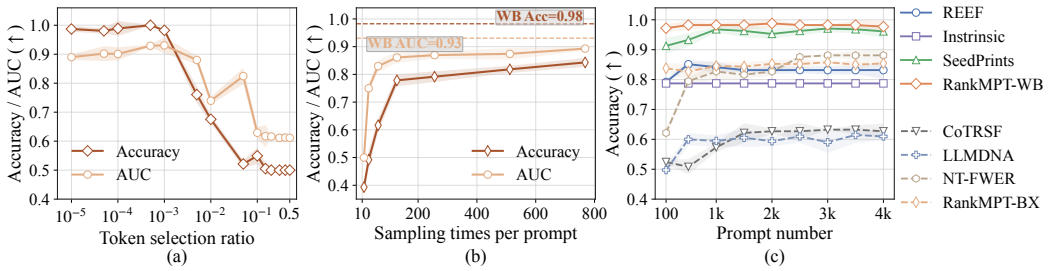


Figure 2: Hyper-parameters ablation on BenchA: (a) token selection ratio$\gamma$ for RankMPT-WB, (b) sampling budget $K \times S$ (fix $K = 8$, vary $S$) for RankMPT-BX, and (c) Number of input prompts; note that Math500 has only 500 prompts, with extras sampled evenly from others.

tial traces of their parent models, even after diverse parameter-altering transformations, enabling the resulting scores to clearly distinguish provenance-related models from unrelated ones.

**RankMPT generalizes across different model families.** We further examine the effect of parent architecture and family artifacts on performance. Table 2 reports accuracy on BenchA grouped by the ten claimed-parent families, each contributing 10 positive and 10 negative pairs. RankMPT-WB reaches the highest or tied-highest accuracy on seven of ten families and RankMPT-BX on six of ten, without family-specific configuration tuning. Since our detector compares relative token-bias alignment rather than architecture-dependent similarity (e.g., hidden geometry), it is potentially less affected by general patterns shared across the same family. For consistency, we report the corresponding per-family breakdown on LeafBench in Appendix E.1, which shows the same trend.

**Tail-token ranking consistency yields stronger discriminability for provenance.** We isolate the discriminability of the similarity score from the subsequent hypothesis test by analyzing the AUC and KS statistics between positive and negative pairs on BenchA. As shown in Table 3, RankMPT achieves the highest KS among white-box methods and outperforms all black-box baselines on both metrics with 0.90 AUC and 0.74 KS. We note that stronger separability does not necessarily imply higher accuracy for testing-based detectors, as their final decision also depends on null-distribution calibration. The separability here suggests that the extracted provenance signal is fundamentally detectable, independent of how the decision rule is tuned.

### 5.3 Ablation Study

**Ablation on decision rules.** We evaluate its contribution by applying alternative decision rules to the same correlation scores. Table 4 compares a ***fixed threshold*** at $\rho_0$, a per-run ***optimal threshold*** selected by grid search over pairwise scores, and a ***multiple-testing*** strategy using human-curated

Table 5: Accuracy of different methods across sibling model pairs, grouped by shared parent model.

| Method | Sibling Pairs by Shared Parent | | | | | Avg. Acc. (*Best* in bold) |
|---|---|---|---|---|---|---|
| | Phi-2 | Gemma2-2B | Coder-1.3B | Qwen2.5-1.5B | Llama3.2-1B | |
| **White-box** | | | | | | |
| REEF | $0.73 \pm 0.46$ | $0.73 \pm 0.46$ | $0.40 \pm 0.20$ | $0.56 \pm 0.19$ | $0.53 \pm 0.23$ | $0.59 \pm 0.31$ |
| Intrinsic | $0.50 \pm 0.00$ | $0.20 \pm 0.00$ | $0.40 \pm 0.00$ | $0.20 \pm 0.00$ | $0.33 \pm 0.00$ | $0.33 \pm 0.00$ |
| Seedprints | $0.53 \pm 0.23$ | $0.40 \pm 0.00$ | $0.53 \pm 0.23$ | $0.44 \pm 0.19$ | $0.62 \pm 0.20$ | $0.51 \pm 0.17$ |
| RankMPT-WB | $0.65 \pm 0.00$ | $0.75 \pm 0.08$ | $0.69 \pm 0.30$ | $0.63 \pm 0.14$ | $0.68 \pm 0.06$ | $\mathbf{0.68 \pm 0.12}$ |
| **Black-box** | | | | | | |
| CoTRSF | $0.73 \pm 0.12$ | $0.87 \pm 0.12$ | $0.73 \pm 0.12$ | $0.87 \pm 0.12$ | $0.77 \pm 0.11$ | $0.79 \pm 0.11$ |
| LLMDNA | $0.87 \pm 0.23$ | $0.82 \pm 0.23$ | $1.00 \pm 0.00$ | $0.73 \pm 0.12$ | $0.33 \pm 0.23$ | $0.76 \pm 0.16$ |
| NT-FWER | $0.67 \pm 0.00$ | $0.74 \pm 0.08$ | $0.82 \pm 0.12$ | $0.86 \pm 0.07$ | $0.73 \pm 0.02$ | $0.76 \pm 0.01$ |
| RankMPT-BX | $0.86 \pm 0.00$ | $0.74 \pm 0.12$ | $0.80 \pm 0.00$ | $0.86 \pm 0.15$ | $0.75 \pm 0.12$ | $\mathbf{0.80 \pm 0.08}$ |

Table 6: Accuracy of different methods on architecture-similar pairs across architecture types.

| Method | Architecture-Similar Model Pairs | | | | | Avg. Acc. (*Best* in bold) |
|---|---|---|---|---|---|---|
| | Qwen2/2 | Llama3/3 | Llama2/3 | Phi2/3 Mini | Gemma2/1 | |
| **White-box** | | | | | | |
| REEF | $0.75 \pm 0.13$ | $0.79 \pm 0.00$ | $0.63 \pm 0.08$ | $0.61 \pm 0.14$ | $0.75 \pm 0.10$ | $0.70 \pm 0.09$ |
| Intrinsic | $0.60 \pm 0.00$ | $0.80 \pm 0.00$ | $0.70 \pm 0.00$ | $0.70 \pm 0.00$ | $0.71 \pm 0.00$ | $0.70 \pm 0.00$ |
| Seedprints | $0.86 \pm 0.20$ | $0.75 \pm 0.07$ | $0.75 \pm 0.05$ | $0.50 \pm 0.00$ | $0.86 \pm 0.25$ | $0.74 \pm 0.11$ |
| RankMPT-WB | $0.73 \pm 0.35$ | $0.92 \pm 0.00$ | $0.57 \pm 0.00$ | $0.80 \pm 0.00$ | $0.94 \pm 0.10$ | $\mathbf{0.79 \pm 0.09}$ |
| **Black-box** | | | | | | |
| CoTRSF | $0.74 \pm 0.07$ | $0.57 \pm 0.12$ | $0.80 \pm 0.08$ | $0.84 \pm 0.19$ | $0.75 \pm 0.25$ | $\mathbf{0.74 \pm 0.14}$ |
| LLMDNA | $0.50 \pm 0.25$ | $0.67 \pm 0.00$ | $0.71 \pm 0.17$ | $0.75 \pm 0.00$ | $0.75 \pm 0.13$ | $0.68 \pm 0.11$ |
| NT-FWER | $0.60 \pm 0.00$ | $0.80 \pm 0.00$ | $0.65 \pm 0.00$ | $0.75 \pm 0.08$ | $0.83 \pm 0.12$ | $0.70 \pm 0.04$ |
| RankMPT-BX | $0.67 \pm 0.03$ | $0.62 \pm 0.08$ | $0.84 \pm 0.08$ | $0.83 \pm 0.00$ | $0.73 \pm 0.06$ | $\mathbf{0.74 \pm 0.05}$ |

unrelated reference models following NT-FWER (Nikolic et al., 2025). Specifically, our studentized testing against the background correlation $\rho$ achieves the highest accuracy in both the white- and black-box settings, reaching 0.978 and 0.842, respectively, which exceeds the optimal thresholding and the reference-based multiple testing. Together, these results show that our testing principle accounts for both the magnitude and variability of the correlation scores, enabling better provenance decisions than directly thresholding.

**Ablation on hyper-parameters.** We examine the sensitivity of RankMPT to three controllable resources: the token-selection ratio $\gamma$, the sampling budget $K \times S$, and the number of input prompts $N$. Figure 2(a) shows that RankMPT-WB performs best when only a small fraction of tail tokens is selected, with both accuracy and AUC remaining high for $\gamma \in [0.1\%, 1\%]$ and degrading as more tokens are included. This suggests that the most discriminative provenance signal is concentrated in a narrow tail region. Figure 2(b) shows that increasing the sampling budget steadily improves RankMPT-BX, with most of the gain obtained at moderate budgets and diminishing returns thereafter, gradually approaching the white-box performance. Figure 2(c) shows that RankMPT-WB remains consistently strong even with a small number of prompts, while RankMPT-BX improves modestly as more prompts are provided. Overall, these results show that RankMPT benefits from a concentrated tail signal, while remaining effective with relatively small sampling and prompt budgets. Additional sensitivity analysis on the background correlation is provided in Appendix E.2.

## 6 Discussion

**Performance on different model relatedness.** In addition to the provenance relationship, model relatedness can arise from other factors, such as overlapping training corpus and shared architectures. In this part, we study the effectiveness of our method in detecting the provenance relationship on related model pairs. We therefore evaluate RankMPT on related but non-provenance pairs constructed

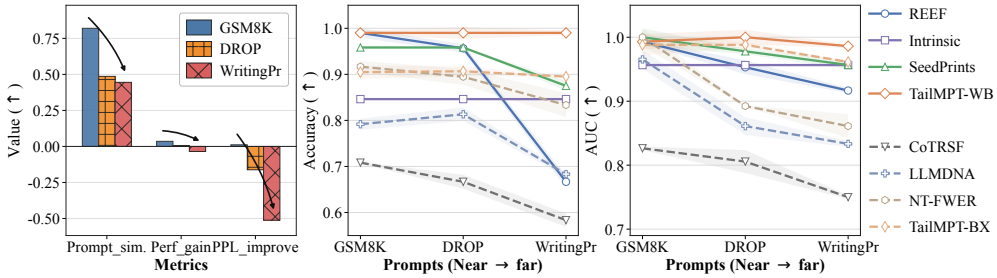


Figure 3: (*Left*) Prompt domain discrepancy from the fine-tuned data GSM8K training set, measured by semantic similarity of prompt; performance gain and perplexity improvement of child models after fine-tuning. (*Middle*) Accuracy and (*Right*) AUC of different methods across three prompt domains. Solid lines denote white-box methods, while dashed lines denote black-box methods.

from BenchA (Nikolic et al., 2025): 20 *architecture-similar* pairs from independently pretrained checkpoints with related architectures, and 20 *sibling* pairs sharing the same parent but no direct derivation relation (Appendix F.1). Tables 5 and 6 show that RankMPT achieves rejection accuracies of up to 0.79 and 0.80 on the two settings, respectively, remaining best or tied-best against the corresponding baselines. Notably, white-box methods degrade more on sibling pairs, plausibly because shared ancestry preserves internal structures that resemble direct inheritance. However, we do not view this as a general flaw, since broader lineage-tracing tasks may reasonably treat siblings as positive cases. Overall, RankMPT can distinguish direct provenance from confounding similarities induced by shared architecture or ancestry to some extent.

**Performance under different prompts.** Since LLM behaviors vary across tasks and domains, behavior-based provenance methods may be sensitive to prompt. Here, we investigate the effect of prompts from different domains on provenance detection. Specifically, we collect 15 unrelated and 15 provenance pairs whose child models are fine-tuned on GSM8K, and use 300 prompts from three domains increasingly distant from the fine-tuning task. As shown in Figure 3, RankMPT maintains relatively stable accuracy and AUC as the prompt domain shifts, while black-box baselines show larger AUC degradation, and white-box baselines retain better separability but still lose accuracy. This suggests that prompt shift can affect existing methods both in their provenance signals and in their decisions. Note that we do not claim such sensitivity a methodological failure, as an excellent performance on particular prompt sets can still be practically useful. However, the stability of RankMPT may support its broader deployment without being confined to a narrow query regime. Additional results across prompt sets on BenchA are provided in Appendix F.2.

## 7 Conclusion

This work seeks to robustly detect LLM provenance by exploiting persistent patterns in the candidate-token space. We propose RankMPT, which leverages the rank consistency between two models over their low-logit tail candidates for provenance testing. Such correlation can be approximated via high-temperature sampling, enabling black-box provenance detection. RankMPT focuses on rarely selected tokens and is therefore less affected by confounding similarities from shared tasks or domains. Extensive experiments demonstrate the effectiveness and robustness of our method. We hope our work provides a new perspective on model provenance and motivates further exploration of robust signals in latent behavioral traces for provenance detection.

**Limitations and Future Directions.** Our method nevertheless has several limitations that motivate future work. First, the black-box variant requires repeated sampling to approximate token distributions, increasing inference cost. Second, our method also relies on a background correlation $\rho_0$; while a common conservative bound works across the evaluated prompts and model families, its generality to broader model populations remains unclear. Future work may improve black-box efficiency and study adversarial or more complex derivations.

## AI USE STATEMENT

Generative AI tools were used to assist with language editing and improving the clarity and presentation of the paper. They were not used to generate experimental data or determine the reported numerical results. All AI-assisted content was reviewed, revised, and verified by the authors, who take full responsibility for the final manuscript.

## ETHICS STATEMENT

Model provenance detection can support model auditing, license compliance, and the identification of unauthorized model reuse. However, provenance decisions may carry legal or ownership implications, and false positives could lead to incorrect attribution. We therefore view our method as a source of statistical evidence rather than a standalone basis for legal or enforcement decisions. Our experiments use publicly available models and datasets and do not involve human subjects or private user data.

## REPRODUCIBILITY STATEMENT

We provide the implementation and evaluation code necessary to reproduce our results. Section 3 specifies the proposed provenance signal and statistical testing procedure, with the complete pipeline summarized in Algorithm 1. Section 5.1 reports the benchmarks, prompt sets, evaluation protocol, and main hyperparameters. The appendices provide additional experimental results, baseline implementation details, the construction of evaluation pairs, and the theoretical analysis of our statistical test. We also release the code and associated evaluation data to facilitate reproducibility.

## REFERENCES

Sixu Chen, Xiang Chen, Hongyao Yu, Jiaxin Hong, Hao Fang, Shuoyang Sun, Bin Chen, and Shu-Tao Xia. Prompt2fingerprint: Plug-and-play llm fingerprinting via text-to-weight generation. *arXiv preprint arXiv:2605.18474*, 2026.

Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, Vandana Bachani, Alex Kaskasoli, Robert Stanforth, Tatiana Matejovicova, et al. Scalable watermarking for identifying large language model outputs. *Nature*, 634(8035):818–823, 2024.

Edgar C Fieller, Herman O Hartley, and Egon S Pearson. Tests for rank correlation coefficients. i. *Biometrika*, 44(3/4):470–481, 1957.

Matthew Finlayson, Andreas Grivas, Xiang Ren, and Swabha Swayamdipta. Token rankings are unforgeable language model signatures. *arXiv preprint arXiv:2606.04459*, 2026.

Yuepeng Hu, Zhengyuan Jiang, Mengyuan Li, Osama Ahmed, Zhicong Huang, Cheng Hong, and Neil Gong. Fingerprinting LLMs via prompt injection. *arXiv preprint arXiv:2509.25448*, 2025.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International Conference on Machine Learning (ICML)*, pp. 17061–17084. PMLR, 2023.

Rohith Kuditipudi, Jing Huang, Sally Zhu, Diyi Yang, Christopher Potts, and Percy Liang. Blackbox model provenance via palimpsestic membership inference. *arXiv preprint arXiv:2510.19796*, 2025.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Zheda Mai, Arpita Chowdhury, Ping Zhang, Cheng-Hao Tu, Hong-You Chen, Vardaan Pahuja, Tanya Berger-Wolf, Song Gao, Charles Stewart, Yu Su, and Wei-Lun Chao. Fine-tuning is fine, if calibrated. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 136084–136119. Curran Associates, Inc., 2024. doi: 10.52202/079017-4323.

Haoming Meng, Kexin Huang, Shaohang Wei, Chiyu Ma, Shuo Yang, Guoyin Wang, Bolin Ding, Jingren Zhou, et al. Sparse but critical: A token-level analysis of distributional shifts in rlvr fine-tuning of llms. In *International Conference on Learning Representations*, volume 2026, pp. 124171–124210, 2026.

Anshul Nasery, Edoardo Contente, Alkin Kaz, Pramod Viswanath, and Sewoong Oh. Are robust LLM fingerprints adversarially robust? *arXiv preprint arXiv:2509.26598*, 2025.

Ivica Nikolic, Teodora Baluta, and Prateek Saxena. Model provenance testing for large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NurIPS)*, 2025.

Dario Pasquini, Evgenios M Kornaropoulos, and Giuseppe Ateniese. LLMmap: Fingerprinting for large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 299–318, 2025.

Wenjun Peng, Jingwei Yi, Fangzhao Wu, Shangxi Wu, Bin Bin Zhu, Lingjuan Lyu, Binxing Jiao, Tong Xu, Guangzhong Sun, and Xing Xie. Are you copying my model? protecting the copyright of large language models for eaas via backdoor watermark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7653–7668, 2023.

Zhenzhen Ren, GuoBiao Li, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. CoTSRF: Utilize chain of thought as stealthy and robust fingerprint of large language models. *arXiv preprint arXiv:2505.16785*, 2025.

Dylan Sam, Marc Finzi, and Zico Kolter. Predicting the performance of black-box language models with follow-up queries. *Advances in Neural Information Processing Systems*, 38:158819–158854, 2026.

Zhuoyi Shang, Jiasen Li, Pengzhen Chen, Yanwei Liu, Xiaoyan Gu, and Weiping Wang. Attesting model lineage by consisted knowledge evolution with fine-tuning trajectory. *arXiv preprint arXiv:2601.11683*, 2026.

Shuo Shao, Yiming Li, Yu He, Hongwei Yao, Wenyuan Yang, Dacheng Tao, and Zhan Qin. Sok: Large language model copyright auditing via fingerprinting. *arXiv preprint arXiv:2508.19843*, 2025.

Qin Shi, Amber Yijia Zheng, Qifan Song, and Raymond A. Yeh. Knowledge distillation detection for open-weights models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

Shangquan Sun, Wenqi Ren, Jingzhi Li, Rui Wang, and Xiaochun Cao. Logit standardization in knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15731–15740, 2024.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. 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)*, pp. 4149–4158, 2019.

Yao Tong, Haonan Wang, Siquan Li, Kenji Kawaguchi, and Tianyang Hu. Seedprints: Fingerprints can even tell which seed your large language model was trained from. In *International Conference on Learning Representations*, volume 2026, pp. 44530–44558, 2026.

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

Tianle Wang, Zhongyuan Wu, Shenghao Jin, Hao Xu, Wei Chen, and Ning Miao. Not all steps are informative: On the linearity of llms' rlvr training. *arXiv e-prints*, pp. arXiv–2601, 2026.

Zhaomin Wu, Haodong Zhao, Ziyang Wang, Jizhou Guo, Qian Wang, and Bingsheng He. LLM DNA: Tracing model evolution via functional representations. *arXiv preprint arXiv:2509.24496*, 2025.

Nan Xu, Chunting Zhou, Asli Celikyilmaz, and Xuezhe Ma. Look-back decoding for open-ended text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 1039–1050, 2023.

Zhenhua Xu, Yiran Zhao, Mengting Zhong, Dezhang Kong, Changting Lin, Tong Qiao, and Meng Han. Dnf: Dual-layer nested fingerprinting for large language model intellectual property protection. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 13972–13976, 2026. doi: 10.1109/ICASSP55912.2026.11461762.

Yunfei Yang, Xiaojun Chen, Yuexin Xuan, Zhendong Zhao, Xin Zhao, and He Li. DeepTracer: Tracing stolen model via deep coupled watermarks. *arXiv preprint arXiv:2511.08985*, 2025.

Do-hyeon Yoon, Minsoo Chun, Thomas Allen, Hans Müller, Min Wang, and Rajesh Sharma. Intrinsic fingerprint of llms: Continue training is not all you need to steal a model! *arXiv preprint arXiv:2507.03014*, 2025.

Runpeng Yu and Xinchao Wang. Neural phylogeny: Fine-tuning relationship detection among neural networks. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 39776–39799, 2025.

Boyi Zeng, Lizheng Wang, Yuncong Hu, Yi Xu, Chenghu Zhou, Xinbing Wang, Yu Yu, and Zhouhan Lin. Huref: Human-readable fingerprint for large language models. *Advances in Neural Information Processing Systems*, 37:126332–126362, 2024.

Hanxiu Zhang and Yue Zheng. SELF: A robust singular value and eigenvalue approach for LLM fingerprinting. *arXiv preprint arXiv:2512.03620*, 2025.

Jie Zhang, Dongrui Liu, Chen Qian, Linfeng Zhang, Yong Liu, Yu Qiao, and Jing Shao. REEF: Representation encoding fingerprints for large language models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations (ICLR)*, volume 2025, pp. 48092–48117, 2025.

Ruisi Zhang, Shehzeen Samarah Hussain, Paarth Neekhara, and Farinaz Koushanfar. REMARK-LLM: A robust and efficient watermarking framework for generative large language models. In *33rd USENIX Security Symposium (USENIX Security 24)*, pp. 1813–1830, 2024.

# A A MOTIVATING PILOT STUDY

To examine *whether certain structures in the prediction distribution can provide useful signals* for provenance detection, we conduct a pilot study designed to isolate genuine provenance from a shared-task confound by controlling for unrelated pairs exposed to the same tasks. We describe the evaluated model pairs and prompt set below, and explore what structures in token predictions can reveal model provenance.

## A.1 SETUP

**Model pairs.** We collect LLMs from Hugging Face and manually verify their provenance relationships using the associated code and documentation, including README files and papers. This yields 15 parent–child pairs $\{P_i, C_i\}_{i=1}^{15}$, where each child model is fine-tuned from its corresponding parent on the GSM8K training set. We further construct 15 non-provenance pairs by pairing child models derived from different parents. Table 7 summarizes all model pairs, which we collectively refer to as *Src-bench*.

**Prompts.** To evaluate sensitivity to domain shift, we consider three prompt sets summarized in Table 8: the GSM8K test set (*GSM8K*), the DROP validation set (*DROP*), and the WritingPrompts test set (*WritingPr*). These prompt sets are progressively less similar to the GSM8K training domain, as measured by cosine similarity. Table 9 further reports the corresponding parent–child gaps in accuracy and perplexity across the three domains.

**Implementation Details.** To obtain the logit vectors for each prompt, we perform a forward pass using the Transformers package with a maximum input length of 512 tokens, while keeping all other settings at their default values. Baseline implementations follow Appendix D.2.

## A.2 TOWARD A ROBUST PROVENANCE SIGNAL

**Rank-based information provides more effective signal for provenance detection.** Figure 4 compares seven metrics capturing different relationships between model predictions, including *ranking consistency* with Spearman correlation, *numerical and distributional differences* with KL divergence, L1, L2, and average logit difference, and *directional or linear relationships* with cosine similarity and Pearson correlation. Spearman consistently achieves the highest AUC across most selection ratios, while the other metrics are either substantially weaker or less stable. Notably, its performance peaks when only a small fraction of tokens is selected and gradually decreases as more tokens are included. These results suggest that model provenance is more strongly reflected in the relative ranking of token predictions than in their absolute values or overall similarity.

**Ranking in low-logit tail tokens keep discriminative across different model scales.** Figure 5 visualizes the 0.1% shared tail-logit ranking for positive (parent–child) and same-family unrelated pairs across 3B-70B parameter scales, where same-family negative pairs serve as a stricter negative control since they share more common factors (e.g., tokenizer, pretraining data) than cross-family pairs. We can observe that, under different scale, the positive pair exhibits a shared, monotonically decreasing tail-logit curve whereas the unrelated shows lower discernible alignment. This supports

Table 7: Detailed model IDs per pair from HuggingFace for domain shift analysis. Positive pairs are parent–child models where the child is fine-tuned on GSM8K training set; negative pairs are random combinations of well-known foundational LLMs with no provenance relation.

| | Positive Pairs | | Negative Pairs | |
|---|---|---|---|---|
| # | Parent Model ID | Fine-tuned Child Model ID | Potential Parent Model ID | Unrelated Child Model ID |
| 1 | Qwen/Qwen2.5-3B-Instruct | BounharAbdelaziz/Qwen2.5-3B-GRPO-Math-GSM8K | BounharAbdelaziz/Qwen2.5-3B-GRPO-Math-GSM8K | HenryShan/Llama-3.2-3B-GSM8K-CoT |
| 2 | meta-llama/Llama-3.2-3B-Instruct | HenryShan/Llama-3.2-3B-GSM8K-CoT | HenryShan/Llama-3.2-3B-GSM8K-CoT | jahyungu/Qwen2.5-1.5B-Instruct_gsm8k |
| 3 | Qwen/Qwen2.5-1.5B-Instruct | jahyungu/Qwen2.5-1.5B-Instruct_gsm8k | jahyungu/Qwen2.5-1.5B-Instruct_gsm8k | jtatman/gpt2-open-instruct-v1-gsm8k |
| 4 | Qwen/Qwen2.5-3B-Instruct | Jason-hu/Qwen2.5-3B-GSM8K-GRPO-H200 | Jason-hu/Qwen2.5-3B-GSM8K-GRPO-H200 | jtatman/gpt2-open-instruct-v1-gsm8k |
| 5 | Qwen/Qwen2.5-3B-Instruct | Jason-hu/Qwen2.5-3B-GSM8K-SFT | Jason-hu/Qwen2.5-3B-GSM8K-SFT | jtatman/gpt2-open-instruct-v1-gsm8k |
| 6 | vicgalle/gpt2-open-instruct-v1 | jtatman/gpt2-open-instruct-v1-gsm8k | jtatman/gpt2-open-instruct-v1-gsm8k | HenryShan/Llama-3.2-3B-GSM8K-CoT |
| 7 | Qwen/Qwen2.5-1.5B-Instruct | nilarnabdebnath/Qwen2.5-1.5B-Instruct_gsm8k | nilarnabdebnath/Qwen2.5-1.5B-Instruct_gsm8k | jtatman/gpt2-open-instruct-v1-gsm8k |
| 8 | HuggingFaceTB/SmolLM2-135M-Instruct | prithivMLmods/SmolLM2_135M_Grpo_Gsm8k | prithivMLmods/SmolLM2_135M_Grpo_Gsm8k | realtreetune/rho-1b-sft-GSM8K |
| 9 | unsloth/phi-4-bnb-4bit | mrm8488/phi-4-14B-grpo-gsm8k-2e | mrm8488/phi-4-14B-grpo-gsm8k-2e | panzs19/LEMMA-LLAMA-3-70B |
| 10 | microsoft/rho-math-1b-v0.1 | realtreetune/rho-1b-sft-GSM8K | realtreetune/rho-1b-sft-GSM8K | tungpl/gsm8k-llama3-grpo |
| 11 | unsloth/Meta-Llama-3.1-8B-Instruct | sarapatel/llama3.1-8b-grpo-gsm8k-run1 | sarapatel/llama31-8b-grpo-gsm8k-run1 | MoNE-Pruning/Qwen2-57B-A14B-Instruct-MoNE-48-gsm8k-100 |
| 12 | unsloth/Llama-3.2-3B-Instruct | tungpl/gsm8k-llama3-grpo | tungpl/gsm8k-llama3-grpo | sarapatel/llama31-8b-grpo-gsm8k-run1 |
| 13 | MegaScience/Qwen3-4B-MegaScience | pymlex/qwen3-4b-gsm8k | pymlex/qwen3-4b-gsm8k | jahyungu/Qwen2.5-1.5B-Instruct_gsm8k |
| 14 | Qwen/Qwen2-57B-A14B-Instruct | MoNE-Pruning/Qwen2-57B-A14B-Instruct-MoNE-48-gsm8k-100 | MoNE-Pruning/Qwen2-57B-A14B-Instruct-MoNE-48-gsm8k-100 | jahyungu/Qwen2.5-1.5B-Instruct_gsm8k |
| 15 | meta-llama/Meta-Llama-3-70B | panzs19/LEMMA-LLAMA-3-70B | panzs19/LEMMA-LLAMA-3-70B | sarapatel/llama3.1-8b-grpo-gsm8k-run1 |

Table 8: Dataset sources and their cosine similarity to the GSM8K training set. Similarity is computed using `Qwen/Qwen3-Embedding-8B` last-layer [CLS] embeddings.

| Dataset | HuggingFace ID : Split | Description | Cos. Sim. to GSM8K |
|---|---|---|---|
| GSM8K | `openai/gsm8k:test` | Multi-step math word problems (arithmetic reasoning) | 0.8200 |
| DROP | `EleutherAI/drop:validation` | Paragraph-based discrete reasoning (counting, sorting, comparison) | 0.4864 |
| WritingPrompts | `euclaise/writingprompts:test` | Creative story generation from prompts | 0.4449 |

Table 9: Model performance and perplexity changes under domain shift. Performance is measured as pass@1 accuracy for GSM8K and DROP, and as MAUVE score (ranging from 0 to 1)) for WritingPrompts, which evaluates consistency with human text (Xu et al., 2023). Large $\Delta_{perf}$ and $\Delta_{PPL}$ indicate the normalized improvement over the parent in performance and perplexity respectively.

| Dataset | $P$ (parent) | $C$ (derived child) | $\Delta_{perf}$ (C-P)/(1-P) | $\text{PPL}_P$ | $\text{PPL}_C$ | $\Delta_{PPL}$ (P-C)/(P-1) |
|---|---|---|---|---|---|---|
| GSM8K | 0.4071 | 0.5300 | 0.2079 | 2.0198 | 1.9103 | 0.0644 |
| DROP | 0.1461 | 0.1460 | −0.0001 | 4.8065 | 5.0514 | −0.0578 |
| WritingPrompts | 0.4692 | 0.3636 | −0.1768 | 1.8859 | 2.0012 | −0.1318 |

prior observations (Wang et al., 2026) that adaptation objectives primarily sharpen preferences over task-relevant tokens and pull them toward the top of the distribution, leaving the tail relatively unaffected. Consequently, tail-token ordering retains structure inherited from the parent model, making it a likely feasible signal for lineage identification.

**Tail-token ranking provides a more robust provenance signal than prior methods.** We compare the discriminability of 0.1% tail-token ranking with representative prior scoring functions across in-domain and far-domain prompts. Figure 6 compares the Spearman correlation on low-logit tokens against two representative alternatives: generation-based semantic similarity, used by LLMDNA Wu et al. (2025), and representation-space alignment, used by REEF Zhang et al. (2025). We observe that the discriminability of both baselines degrades sharply under domain shift, though to varying degrees. In contrast, the token-ranking consistency score maintains near-perfect separation between positive and negative pairs across all three domains. These results confirm that the ranking-based correlation is robust to domain shift.

## B Rationale for Background Correlation

Unrelated models can exhibit nonzero correlation even in the absence of a provenance relationship. This background correlation can arise from at least two sources: shared tokenizer or vocabulary statistics, since low-frequency tokens tend to receive low rankings across models regardless of lineage, and overlap in pretraining data, e.g., through commonly used corpora. Consequently, identity-token rankings between unrelated models are not necessarily independent. Setting $\rho_0$ inappropriately may therefore attribute such incidental alignment to shared provenance, potentially increasing the false classification rate. This motivates an empirical examination of the appropriate choice of $\rho_0$. Hence, we first characterize the distribution of correlation scores across related and unrelated model pairs, and then evaluate the sensitivity of the detector to the specific choice of $\rho_0$.

**Distribution of $\rho$.** We measure the mean correlation $\bar{\rho}_{(P,C)} = \frac{1}{\sum_n |\mathcal{T}_n|} \sum_{n,t} \rho_{n,t}$ for both positive and negative pairs of Src-bench listed in Table 7 over input prompts, across different prompt domains and model families. Figure 7 shows the resulting distribution of $\bar{\rho}_{(P,C)}$ per pair class across the three prompt domains. Positive pairs concentrate near $\bar{\rho} \approx 0.97$ in all three domains, while negative pairs form a well-separated cluster whose mean is centered close to zero and far below the positive-pair level, confirming that unrelated models exhibit at most a small background correlation. Figure 8 further breaks this down per individual negative pair across domains: the estimated baseline is relatively stable across different unrelated pairs, supporting the use of a single, loosely conservative upper bound $\rho_0 = 0.1$ as a shared null baseline, rather than a pair- or domain-specific calibration. **Overall, the observed background correlation is small and varies only modestly across the evaluated unrelated pairs and prompt domains.**

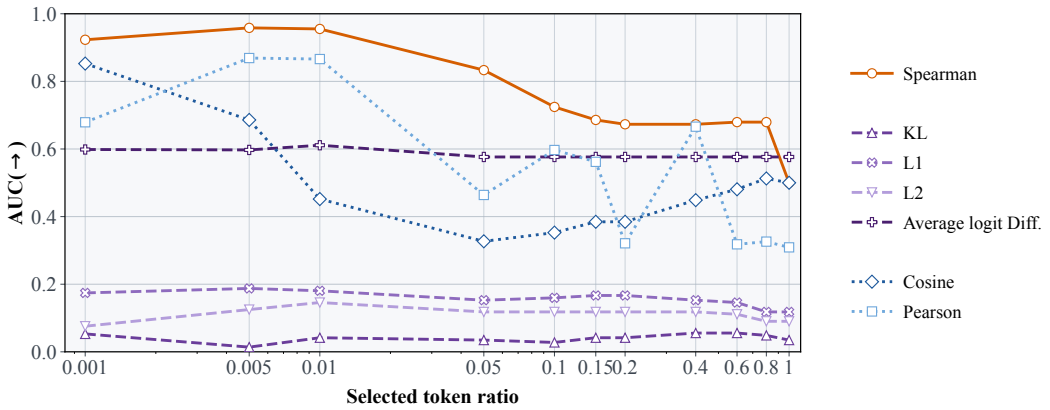


Figure 4: Comparison of provenance scoring functions across different selected token ratios. Tokens are ordered by the parent-model logits, and each score is computed using the lowest-logit fraction specified on the x-axis. AUC evaluates separation between provenance and non-provenance pairs.

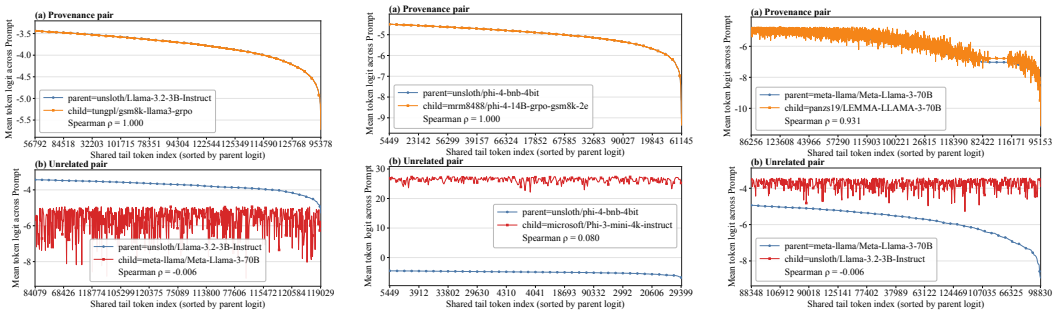


Figure 5: Visualization of 0.1% shared tail-logit token ranking for positive and same-family unrelated pairs at 3B, 14B, and 70B scales. Token IDs are sorted by parent logit; Spearman $\rho$ between the two LLM rankings is reported for each pair.

**Sensitivity to $\rho_0$.** To examine whether the detector depends on the precise choice of $\rho_0$, we further apply our white-box testing pipeline to Src-Bench while varying $\rho_0$ from $-0.5$ to $0.8$. We report accuracy, precision, and recall averaged over three runs on the GSM8K, DROP, and WritingPr prompt sets. As shown in Figure 9, the detector maintains consistently strong performance over a broad range of $\rho_0$ (roughly 0–0.5), with accuracy remaining around 0.95–1.00. Performance degrades noticeably only for overly large values of $\rho_0$, primarily due to reduced recall. **The detector is there-**

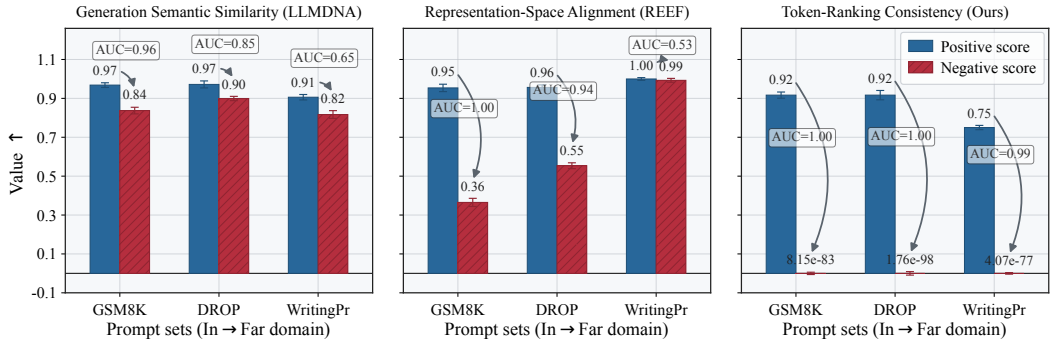


Figure 6: Discriminability of different methods across three query prompt domains. Bars show score variance across runs; AUC averaged over positive and negative pairs is annotated above.

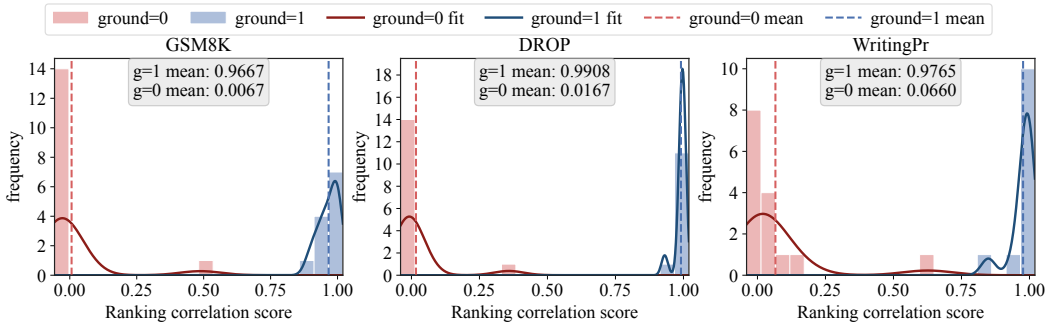


Figure 7: Distribution of pair-wise correlation scores for positive (g=1) and negative (g=0) model pairs in Src-Bench across three prompt domains. Text box annotates the mean score of each group.

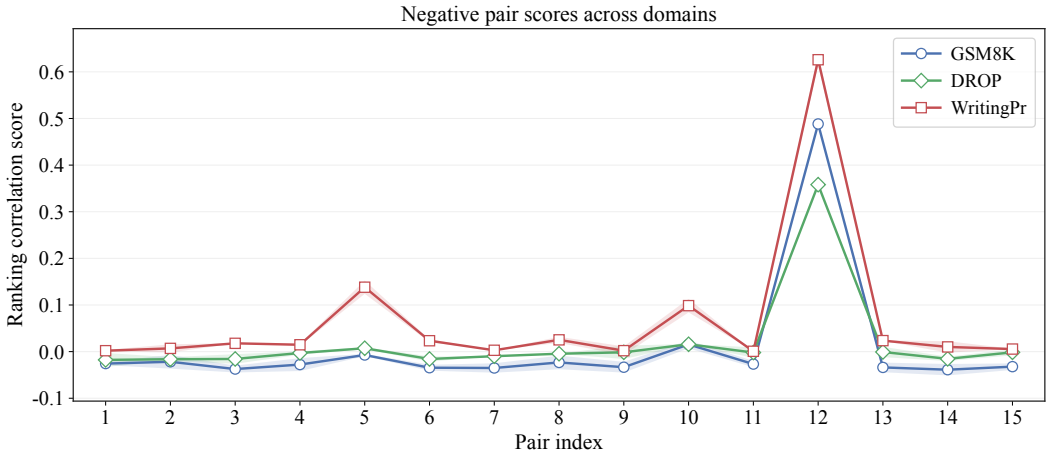


Figure 8: Mean correlation scores of negative model pairs in Src-Bench across three prompt domains. The background correlations remain relatively stable across pairs and domains.

**fore relatively insensitive to the exact choice of $\rho_0$ within a broad, practically relevant range.** Together with the stable background correlations observed above, this supports the use of a single conservative global $\rho_0$ without pair- or domain-specific tuning.

## C Asymptotic Normality and Validity of the Provenance Test

This section provides a detailed proof establishing the asymptotic normality and validity of the pair-level statistic $Z_{P,C}$ defined in Section 4. For clarity, we first recall the hypothesis test defined in the main text:

$$H_0 : \mathbb{E}[\rho_i] \leq \rho_0 \qquad \text{vs.} \qquad H_1 : \mathbb{E}[\rho_i] > \rho_0, \tag{7}$$

where $\rho_i$ denotes the prompt-level ranking correlation and $\rho_0$ is the conservative upper bound under non-provenance. Let $z_0 = \operatorname{arctanh}(\rho_0)$ and $z_i = \operatorname{arctanh}(\rho_i)$ , we fconduct the corresponding one-sided test on the Fisher scale:

$$H_0 : \mu_z \leq z_0 \qquad \text{vs.} \qquad H_1 : \mu_z > z_0, \tag{8}$$

where $\mu_z = \mathbb{E}[z_n]$ denotes the population mean of the prompt-level Fisher-transformed score.

Under this formulation, we analyze the asymptotic behavior of $Z_{P,C}$ under the null hypothesis. We first establish the asymptotic distribution of $Z_{P,C}$ at the boundary of the null hypothesis, i.e., $\mu_z = z_0$ (equivalently, $\mathbb{E}[\rho_i] = \rho_0$), and then show that this boundary is asymptotically the least favorable case over the composite null.

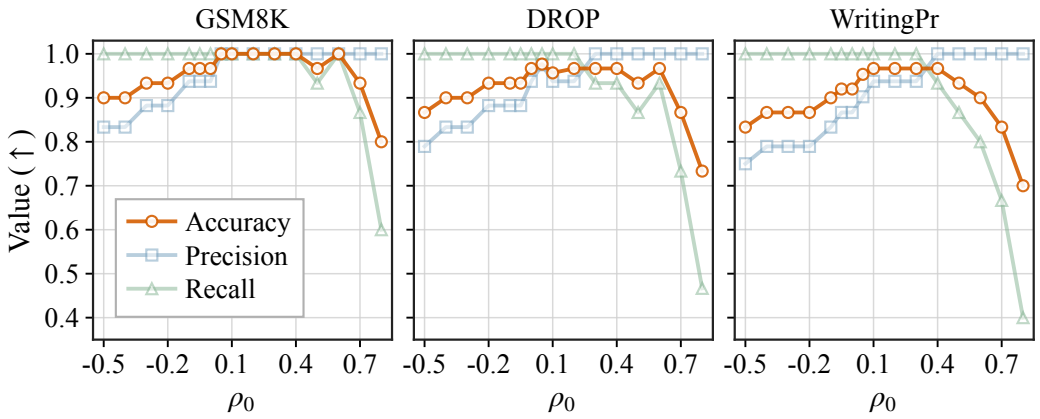


Figure 9: Accuracy, precision, and recall of our white-box detector across $\rho_0 \in [-0.5, 0.8]$ on SrcBench. A performance plateau over conservative $\rho_0$ values indicates limited sensitivity to the choice.

**Proof of the asymptotic normality at the null boundary.** Recall that the identity-token set $\mathcal{I}$ is identified from aggregate behavior over the prompt set $\mathcal{X}$, while the test statistic is computed from prompt-level ranking scores on the same set. Although this induces dependence between token selection and the resulting scores, we assume the dependence to be *weak*, as $\mathcal{I}$ is determined collectively by many prompts and any individual prompt has only limited influence on the selection. We therefore condition on the selected identity-token set $\mathcal{I}$ and treat it as fixed throughout the analysis.

Let $U = |\mathcal{X}|$. With the identity-token set $\mathcal{I}$ selected from the separate localization prompts $\mathcal{X}_{\text{loc}}$, we assume that the test prompts in $\mathcal{X}_{\text{test}}$ are i.i.d., inducing i.i.d. prompt-level observations $\{z_n\}_{n=1}^{U}$ with $\mu_z = \mathbb{E}[z_n]$, $\sigma_z^2 = \text{Var}(z_n)$, and $0 < \sigma_z^2 < \infty$.

By the classical central limit theorem (Van der Vaart, 2000), we have $\sqrt{U}(\bar{z} - \mu_z)/\sigma_z \xrightarrow{d} \mathcal{N}(0,1)$, where $\bar{z} = \frac{1}{U}\sum_{n=1}^{U} z_n$. Since $\sigma_z$ is typically unknown among finite samples, we estimate it by $\widehat{\sigma}_z^2 = \frac{1}{N-1}\sum_{n=1}^{N}(z_n - \bar{z})^2$. By consistency of the sample variance, $\widehat{\sigma}_z \xrightarrow{p} \sigma_z$. Therefore, by Slutsky's theorem (Van der Vaart, 2000),

$$\frac{\sqrt{U}(\bar{z} - \mu_z)}{\widehat{\sigma}_z} \xrightarrow{d} \mathcal{N}(0,1). \tag{9}$$

At the boundary of the null hypothesis, $\mu_z = z_0 = \text{arctanh}(\rho_0)$, the test statistic becomes

$$Z_{P,C} = \frac{\sqrt{N}\,[\bar{z} - z_0]}{\widehat{\sigma}_z} = \frac{\sqrt{N}\,[\bar{z} - \mu_z]}{\widehat{\sigma}_z}. \tag{10}$$

Hence, from Eq. (9), we can obtain

$$Z_{P,C} \xrightarrow{d} \mathcal{N}(0,1). \tag{11}$$

Thus, the standard normal distribution is the asymptotic distribution of $Z_{P,C}$ at the null boundary.

**Proof of the validity over the composite null.** It remains to show that the boundary $\mu_z = z_0$ is asymptotically the least favorable case of the composite null. For any fixed point in the interior of the null, $\mu_z < z_0$, let $\delta = z_0 - \mu_z > 0$. Then,

$$\begin{aligned} Z_{P,C} &= \frac{\sqrt{N}(\bar{z} - z_0)}{\widehat{\sigma}_z} \\ &= \frac{\sqrt{N}(\bar{z} - \mu_z)}{\widehat{\sigma}_z} - \frac{\sqrt{N}\delta}{\widehat{\sigma}_z}. \end{aligned} \tag{12}$$

The first term is $O_p(1)$ by Eq. (9), whereas $\widehat{\sigma}_z \xrightarrow{p} \sigma_z > 0$ implies $\sqrt{N}\delta/\widehat{\sigma}_z \xrightarrow{p} +\infty$. Therefore, for every fixed $\mu_z < z_0$,

$$Z_{P,C} \xrightarrow{p} -\infty. \tag{13}$$

Consider the one-sided rejection rule $Z_{P,C} > \Phi^{-1}(1-\alpha)$. At the null boundary $\mu_z = z_0$, Eq. (11) gives $\Pr_{\mu_z=z_0}\left(Z_{P,C} > \Phi^{-1}(1-\alpha)\right) \longrightarrow \alpha$. For every fixed point in the interior of the null, $\mu_z < z_0$, Eq. (13) gives $\Pr_{\mu_z}\left(Z_{P,C} > \Phi^{-1}(1-\alpha)\right) \longrightarrow 0$.

Consequently, for every fixed $\mu_z \leq z_0$,

$$\limsup_{N\to\infty} \Pr_{\mu_z}\left(Z_{P,C} > \Phi^{-1}(1-\alpha)\right) \leq \alpha. \tag{14}$$

Therefore, the null boundary $\mu_z = z_0$ is asymptotically the least favorable case, and the proposed provenance test is asymptotically valid at level $\alpha$ over the composite null $H_0$.

□

# D Additional Details for Experiments

## D.1 Rational for used prompt set

Table 10 summarizes the prompt datasets used in our experiments, together with their domains and usage in prior provenance baselines. RandomPrompt has been adopted by NT-FWER, while Math500 and CommonSenseQA are used by representative white- and black-box methods such as REEF, CoTSRF, and LLMDNA. We therefore select these prompt sets to align our evaluation with existing provenance studies and to ensure that the comparison is conducted under prompt distributions already established in prior work, rather than on inputs tailored to our method.

Table 10: Overview of the prompt datasets used in our experiments in Section 5.

| Dataset | #Prompt | Domain | Usage in Baselines |
|---|---|---|---|
| RandomPrompt (Nikolic et al., 2025) | ∼100k | Simple factual continuation | NT-FWER |
| Math500 (Lightman et al., 2024) | 500 | Mathematical problems | REEF, CoTSRF, LLMDNA, REEF |
| CommonSenseQA train (Talmor et al., 2019) | ∼9.74k | Common-sense reasoning | LLMDNA, REEF |

## D.2 Baseline Implementation Details

We compare against representative white-box and black-box provenance methods, covering fingerprints derived from model parameters, hidden representations, logits, generated outputs, and chain-of-thought responses. For each method, we follow the original fingerprint construction and statistical procedure whenever applicable, while adapting only the input format or decision rule required by our pairwise evaluation protocol.

### D.2.1 White-box baselines

**Intrinsic** (Yoon et al., 2025) derives fingerprints from layer-wise statistics of the attention projections. We follow the original construction by computing the standard deviation of the query (Q), key (K), value (V), and output (O) projection matrices weights at each layer, z-normalizing each resulting sequence, and measuring Pearson correlation between corresponding projection statistics. When the two models have different depths, we linearly interpolate the shorter sequence to the depth of the other model. The four projection-wise correlations are averaged to obtain the pairwise fingerprint score. We use the threshold of 0.8 specified by the original method for binary provenance decisions.

**REEF** (Zhang et al., 2025) characterizes a model through the similarity of its intermediate representations. Given the same prompts, we extract the last non-padding hidden state at relative depths 40%, 50%, 60%, 70%, and 80%, and compute linear CKA between the corresponding representations of two models. Following the original rule, a higher CKA indicates greater similarity. To fit our pairwise evaluation, we use the maximum CKA across the five depths as the pairwise score and classify a pair as related when the score exceeds 0.9.

**SeedPrint** (Tong et al., 2026) captures shared ordering patterns in model representations. Following the original method, we extract the last-layer embedding at the final non-padding position and select the 1,000 dimensions with the lowest mean activation as identity dimensions. We normalize the selected embeddings with a row-wise softmax and compute Kendall–Tau correlations across prompts

for each identity dimension. The resulting correlations are compared against the prescribed null distribution using a one-sided Mann–Whitney U test, and a pair is classified as related (prediction=1) when $p < 0.05$; otherwise, it is classified as unrelated (prediction=0).

#### D.2.2 BLACK-BOX BASELINES

**NT-FWER** (Nikolic et al., 2025) detects provenance from agreement in next-token predictions while accounting for multiple candidate reference models. For a suspect–reference pair, we treat token-level prediction agreement as a Bernoulli outcome and use the resulting mean match rate as the pairwise evidence. Following the original testing framework, the suspect is evaluated against a set of control reference models using one-sided tests of the difference in match rates, with Fisher's exact test used for small counts and a pooled two-proportion test otherwise. We apply Holm–Bonferroni correction across the controls and use $\alpha = 0.05$. A suspect is classified as originating from the reference only when its match rate is higher than that of every control and all corresponding corrected tests reject the null hypothesis. For token generation, we set `max_new_length` = 1 and other parameters use the default values following the original paper.

**LLMDNA** (Wu et al., 2025) represents a model by a low-dimensional functional signature derived from its generated responses. Following the original construction, we encode responses using CLS-pooled Qwen3-Embedding-8B representations, concatenate the representations in a fixed prompt order, and apply a fixed Gaussian projection to obtain a 128-dimensional DNA representation. We use seed 42 for the projection. For pairwise classification, we use the element-wise absolute difference between the two DNA representations as the pair feature, standardize the features, and train an RBF SVM with $C = 1$ and $\gamma =$ `scale`. We perform five-fold cross-validation over model pairs, where each fold uses an 80/20 train-test split. The final benchmark result is obtained by aggregating the results across all five folds. During training, we sample 100 prompts from each of `randomPrompt`, `math500`, and `CommonSenseQA`, while evaluation uses the larger prompt sets defined by each benchmark. For text generation, we set `max_length` = 1024, `temperature` = 0.7, `top_p`= 0.9, following the original paper.

**CoTSRF** (Ren et al., 2025) exploits distributional characteristics of generated chain-of-thought responses. Following the original paper, we train a unified response encoder (equivalently, the CoT extractor) for each claimed parent model. For each parent, two of its responses serve as anchor and positive, and a response from another parent serves as negative, with a triplet margin of 5. The architecture builds on `allenai/longformer-base-4096`, using mean pooling followed by a 128-dimensional projection. We optimize each model for 3,000 epochs with Adam under the triplet margin loss. To fit our experiments, we sample 50 prompts from each of GSM8k, DROP, WritingPrompt, RandomPrompt, CommonSenseQA, and Math500. We generate four responses at `temperature` = 1.5 per prompt; these are then used to construct multiple triplets per parent model, as described above. In inference, for a pair (P,C), the CoT extractor corresponding to P first represents the responses from the claimed parent and the suspect child in the learned embedding space, and the Symmetric KL divergence is used to measure the alignment between these two feature distributions. The mean divergence over the input prompts are taken as the pair-wise score. Since the original method uses source-specific decision boundaries, we determine a separate threshold for each source as the midpoint between the mean scores of its positive and negative pairs.

## E ADDITIONAL RESULTS FOR EXPERIMENTS

### E.1 PERFORMANCE ON DIFFERENT FAMILY

**Our method maintains strong provenance accuracy across diverse LLM families.** We provide complementary results of family-wise performance on Leafbench. As shown in Table 11 RankMPT-WB achieves an average accuracy of 0.98 and reaches the highest or tied-highest accuracy on five families, while RankMPT-BX averages 0.89 and performs best on six families. These results indicate that our method generalizes reliably across heterogeneous model families rather than relying on family-specific characteristics.

Table 11: Performance comparison across different models. All std. values are $\leq$0.03 and omitted. Values in parentheses indicate the improvement of RankMPT over the strongest baseline.

| Models | #Pair | White-box Methods | | | | Black-box Methods | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | REEF | Intrinsic | SeedPrint | RankMPT-WB | CoTRSF | LLMDNA | NT-FWER | RankMPT-BX |
| Qwen2.5-7B | 11 | 1.00 | 0.82 | 0.80 | 1.00 (+0.00) | 0.90 | 0.20 | 0.70 | 1.00 (+0.10) |
| Qwen2.5-14B | 6 | 1.00 | 0.83 | 1.00 | 1.00 (+0.00) | 1.00 | 0.00 | 1.00 | 0.83 (−0.17) |
| Llama-3.1-8B | 11 | 0.82 | 0.91 | 1.00 | 1.00 (+0.00) | 0.36 | 0.36 | 0.45 | 0.82 (+0.36) |
| Mistral-7B-v0.3 | 6 | 1.00 | 0.83 | 1.00 | 1.00 (+0.00) | 1.00 | 0.00 | 0.43 | 1.00 (+0.00) |
| Gemma-2-2B | 6 | 1.00 | 0.71 | 1.00 | 1.00 (+0.00) | 0.57 | 0.43 | 0.29 | 0.86 (+0.29) |
| TinyLlama-1.1B | 6 | 0.83 | 0.83 | 0.83 | 0.83 (+0.00) | 0.17 | 0.00 | 0.17 | 0.83 (+0.67) |
| Llama-2-7B | 6 | 1.00 | 0.71 | 1.00 | 1.00 (+0.00) | 0.83 | 0.17 | 0.43 | 1.00 (+0.17) |
| **Overall** | **52** | 0.94 | 0.82 | 0.94 | **0.98** (+0.04) | 0.68 | 0.19 | 0.51 | **0.89** (+0.21) |

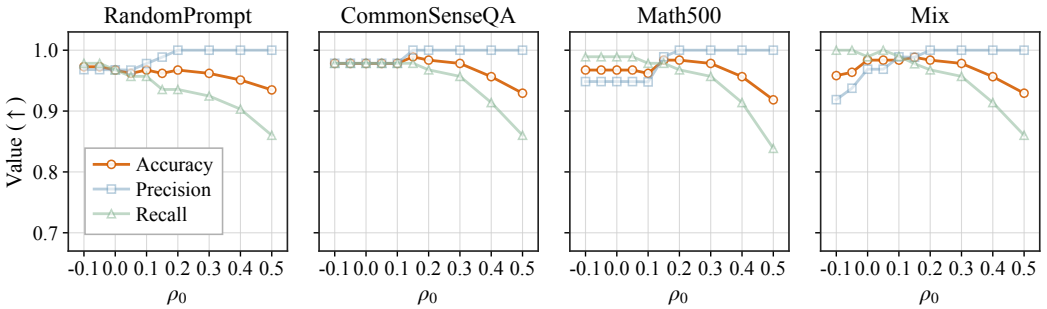


Figure 10: Performance of the RankMPT detector with varying $\rho_0$ on BenchA across four prompt sets. We sample 1,000, 1,000, and 500 prompts from RandomPrompt, CommonSenseQA, and Math500, respectively; Mix combines 500 prompts from each of these three sets, as in Section 5.

### E.2 Effect of Background Correlation

**RankMPT performs stably over a broad moderate range of $\rho_0$.** We additionally provide a sensitivity analysis of RankMPT with respect to $\rho$ on BenchA. For completeness, we also report the performance when using each prompt category individually, measured by accuracy, precision, and recall. As shown in Figure 10, for $\rho_0 \in [-0.1, 0.3]$, accuracy varies modestly across the four prompt sets (approximately 0.96–0.99), but starts to decline more noticeably at $\rho_0 \geq 0.4$, reaching roughly 0.92–0.93 at $\rho_0 = 0.5$ for most settings. Meanwhile, precision increases toward 1.0, whereas recall decreases increasingly as $\rho_0$ grows. This behavior is consistent with Figure 11: negative pairs are concentrated near zero, while positive pairs have a broader low-correlation tail. Increasing $\rho_0$ therefore removes false positives but eventually also rejects more borderline positive pairs, explaining the higher precision and lower recall and accuracy at larger $\rho_0$. These findings are consistent with those observed on Src-bench, as shown in Appendix B.

## F Additional Results for Discussion

### F.1 Performance over Hard Negatives across Prompt Sets

Table 12 and Table 13 list the sibling negative pairs and architecture-similar negative pairs used in used in Section 6. We then examine the performance of two challenging negative-pair types(Arc-Sim and Sibling) under different prompts as follow.

**RankMPT is robust to hard negatives across different prompt sets**. Table 14 reports the accuracy under RandomPrompts, CommonSenseQA, and Math500. we can observed that RankMPT achieves the best average performance in both white-box and black-box settings, while remaining consistently strong across prompt sets. This indicates that RankMPT can reliably distinguish hard negatives under diverse prompt distributions.

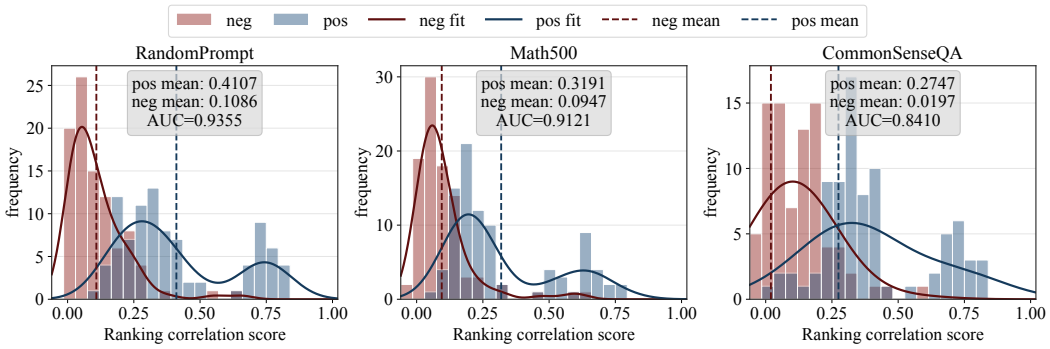


Figure 11: Distribution of pair-wise rank correlation across 100 positive ("pos") and 100 negative ("neg") pairs from *BenchA* under three prompt sets. The mean score of each group and their AUC are annotated in the text box.

Table 12: List of 20 sibling pairs (label=0) with their corresponding shared parent.

| **Claimed parent model** | **Negative child model** | **Shared parent** |
|---|---|---|
| pkarypis/phi2-lima | nguyenthanhdo/phi-half | microsoft/phi-2 |
| pkarypis/phi2-lima | ashmitkx/phi-2-instruct | microsoft/phi-2 |
| pkarypis/phi2-lima | stanfordAIMI/GREEN-Phi2 | microsoft/phi-2 |
| pkarypis/phi2-lima | MaziyarPanahi/phi-2-logical-sft | microsoft/phi-2 |
| hishab/titulm-gemma-2-2b-v1.0 | google/gemma-2-2b-it | google/gemma-2-2b |
| hishab/titulm-gemma-2-2b-v1.0 | harishnair04/Gemma-medtr-2b-sft-v2 | google/gemma-2-2b |
| hishab/titulm-gemma-2-2b-v1.0 | 1024m/gemma-2-2b-it-Base | google/gemma-2-2b |
| hishab/titulm-gemma-2-2b-v1.0 | rinna/gemma-2-baku-2b | google/gemma-2-2b |
| stojchet/nl-all2 | TechxGenus/CursorCore-DS-1.3B-LC | deepseek-ai/deepseek-coder-1.3b-base |
| stojchet/nl-all2 | CodeGPTPlus/deepseek-coder-1.3b-typescript | deepseek-ai/deepseek-coder-1.3b-base |
| stojchet/nl-all2 | JetBrains/deepseek-coder-1.3B-kexer | deepseek-ai/deepseek-coder-1.3b-base |
| stojchet/nl-all2 | stojchet/kto10k1 | deepseek-ai/deepseek-coder-1.3b-base |
| Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct | jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT | Qwen/Qwen2.5-1.5B-Instruct |
| Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct | opencompass/CompassJudger-1-1.5B-Instruct | Qwen/Qwen2.5-1.5B-Instruct |
| Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct | Syed-Hasan-8503/Qwen2.5-1.5B-Instruct-WO-Adam-mini | Qwen/Qwen2.5-1.5B-Instruct |
| win10/Qwen2.5-2B-Instruct | jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT | Qwen/Qwen2.5-1.5B-Instruct |
| phamhai/Llama-3.2-1B-CyberFrog | Isotonic/OrcaAgent-llama3.2-1b | meta-llama/Llama-3.2-1B-Instruct |
| phamhai/Llama-3.2-1B-CyberFrog | Hjgugugjhuhjggg/mergekit-passthrough-owrmdht | meta-llama/Llama-3.2-1B-Instruct |
| phamhai/Llama-3.2-1B-CyberFrog | wy2001/creativestorywriter | meta-llama/Llama-3.2-1B-Instruct |
| phamhai/Llama-3.2-1B-CyberFrog | torchtorchkimtorch/Llama-3.2-Korean-GGACHI-1B-Instruct-v1 | meta-llama/Llama-3.2-1B-Instruct |

### F.2 Performance across prompt sets BenchA

**Performance degradation under prompt shift.** Figure 3 reports classification accuracy and AUC for existing methods in distinguishing positive from negative pairs across prompt domains. We observe two distinct effects. At the *signal level*, separability degrades as the prompt domain shifts away from the fine-tuning domain. Black-box methods relying on output-level statistics drop to near-chance AUC ($< 0.5$) under far-domain prompts, while white-box methods retain AUC above 0.9, indicating that discriminative information persists despite domain shift. At the *decision level*, however, accuracy degrades for both method types, even when white-box AUC remains high. This reveals that fixed thresholds or test statistics calibrated on one domain do not generalize to another. In contrast, our method, which uses token ranking in the shared low-preference region between two LLMs, maintains both high AUC and high accuracy across all three domains. This shows that our score captures model-specific structure under domain shift, and that our hypothesis-testing framework adaptively recalibrates its null distribution from the data, rather than relying on a fixed domain-specific reference.

In this part, we provide additional results for different methods across four prompt sets. With reference to the dataset sizes used in existing work, we choose 1,000 samples for RandomPrompt, 1,000 samples for CommonSenseQA, and 500 samples for Math500, so that the overall data scale is roughly comparable. The Mixed Prompts setting is consistent with that used in Section 5, combining 500 samples from each of the first three prompt sets.

Table 13: List of 20 architecture-similar pairs (label=0) with their corresponding architecture styles.

| Claimed parent model | Negative child model | Arch. style (parent / child) |
|---|---|---|
| google/gemma-2-2b | ale-bay/zephyr-2b-gemma-dpo | Gemma 2 / Gemma 1 |
| google/gemma-2-2b | google/gemma-2b-aps-it | Gemma 2 / Gemma 1 |
| google/gemma-2b | google/gemma-2-2b | Gemma 1 / Gemma 2 |
| google/gemma-2b | hishab/titulm-gemma-2-2b-v1.0 | Gemma 1 / Gemma 2 |
| meta-llama/Llama-3.2-1B-Instruct | prithivMLmods/Llama-Doctor-3.2-3B-Instruct | Llama 3 / Llama 3 |
| meta-llama/Llama-3.2-1B-Instruct | TinyLlama/TinyLlama-1.1B-Chat-v1.0 | Llama 3 / Llama 2 |
| meta-llama/Llama-3.2-3B-Instruct | jeremierostan/llama32-1b-restorative | Llama 3 / Llama 3 |
| meta-llama/Llama-3.2-3B-Instruct | meta-llama/Llama-3.2-1B-Instruct | Llama 3 / Llama 3 |
| microsoft/phi-2 | trucnhi160703/ft-phi3-4k | Phi-2 / Phi-3 Mini |
| microsoft/phi-2 | chaoweihuang/FactAlign-Phi-3-Mini | Phi-2 / Phi-3 Mini |
| microsoft/Phi-3-mini-4k-instruct | ashmitkx/phi-2-instruct | Phi-3 Mini / Phi-2 |
| microsoft/Phi-3-mini-4k-instruct | stanfordAIMI/GREEN-Phi2 | Phi-3 Mini / Phi-2 |
| Qwen/Qwen2-1.5B | winstcha/Qwen2.5-1.5B-FT-v2 | Qwen 2 / Qwen 2 |
| Qwen/Qwen2-1.5B | jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT | Qwen 2 / Qwen 2 |
| Qwen/Qwen2.5-1.5B-Instruct | BEE-spoke-data/Qwen2-1.5B-stepbasin-books | Qwen 2 / Qwen 2 |
| Qwen/Qwen2.5-1.5B-Instruct | Quardo/TR-Qwen2-1.5B-002-241130 | Qwen 2 / Qwen 2 |
| TinyLlama/TinyLlama-1.1B-Chat-v1.0 | AcademieDuNumerique/Llama-3.2-3B-SQL-Instruct | Llama 2 / Llama 3 |
| TinyLlama/TinyLlama-1.1B-Chat-v1.0 | DoeyLLM/OneLLM-Doey-V1-Llama-3.2-1B-it | Llama 2 / Llama 3 |
| TinyLlama/TinyLlama-1.1B-Chat-v1.0 | iqwiki-kor/Llama3.2-3B-MP-RM | Llama 2 / Llama 3 |
| TinyLlama/TinyLlama-1.1B-Chat-v1.0 | jkazdan/llama3b-real-and-synthetic-sftsd0 | Llama 2 / Llama 3 |

Table 14: Accuracy of different methods on hard negative pairs across different prompt sets. All stds $< 0.06$ and omitted for brevity. The best value in each setting is in bold.

| Methods | RandomPrompts | | CommonSenseQA | | Math500 | | Avg. | |
|---|---|---|---|---|---|---|---|---|
| | Arc-Sim | Sibling | Arc-Sim | Sibling | Arc-Sim | Sibling | Arc-Sim | Sibling |
| **White-box** | | | | | | | | |
| REEF | 0.67 | 0.21 | **0.86** | 0.67 | 0.77 | 0.47 | 0.76 | 0.45 |
| Intrinsic | 0.74 | 0.33 | 0.74 | 0.33 | 0.74 | 0.33 | 0.74 | 0.33 |
| Seedprints | 0.82 | 0.64 | 0.80 | 0.55 | 0.82 | **0.66** | 0.82 | 0.62 |
| RankMPT-WB | **0.86** | **0.71** | **0.86** | **0.71** | **0.84** | 0.63 | **0.85** | **0.69** |
| **White-box** | | | | | | | | |
| CoTRSF | 0.53 | 0.77 | 0.72 | **0.87** | **0.78** | 0.77 | 0.68 | 0.80 |
| LLMDNA | 0.43 | 0.69 | 0.66 | 0.73 | 0.70 | 0.73 | 0.60 | 0.72 |
| NT-FWER | 0.68 | 0.75 | 0.64 | 0.67 | 0.63 | 0.76 | 0.65 | 0.73 |
| RankMPT-BX | **0.73** | **0.79** | **0.75** | 0.80 | 0.71 | **0.82** | **0.73** | **0.81** |

### F.2.1 Score separability

**Rank correlation remains stable and discriminative across prompt sets.** As a complementary analysis, we report the AUC and KS separability between positive and negative pairs on *BenchA* for each method across four prompt sets in Table 15. Among white-box methods, Our method exhibits the most stable separability across prompt domains, whereas REEF shows larger domain-dependent variation. Among black-box methods, Our method consistently outperforms all baselines and is substantially more stable across domains, with AUC variation of at most 0.06, compared with up to 0.50 for the baselines. We further visualize the distributions of ranking correlation scores for positive and negative pairs in Figure 11. Together, these results suggest that ranking-based provenance signals derived from low-logit tokens are more robust to prompt-domain shifts than full-output- or representation-based alternatives.

### F.2.2 Classification performance

**RankMPT consistently achieves the highest classification accuracy across different prompt sets.** Figure 12 compares white-box and black-box provenance methods on BenchA across four prompt sets. RankMPT-WB achieves an average accuracy of 0.98, outperforming the strongest white-box baseline by 6 points, while RankMPT-BX achieves 0.84, exceeding the best black-box baseline by 15 points. As expected, white-box methods consistently outperform their black-box counterparts, since access to full-vocabulary logits preserves fine-grained token preference infor-

Table 15: Score separability of LLM provenance detectors on *BenchA* pairs across different inputs.

| Methods | Metrics | Input Prompt Sets | | | |
|---|---|---|---|---|---|
| | | RandomPrompt | CommonSenseQA | Math500 | Mixed Prompts |
| REEF | AUC↑ | 0.89±0.02 | 0.89±0.00 | 0.76±0.06 | 0.85±0.02 |
| | KS↑ | 0.85±0.01 | 0.73±0.03 | 0.62±0.01 | 0.65±0.03 |
| Intrinsic | AUC↑ | 0.97±0.01 | 0.97±0.01 | 0.97±0.07 | 0.97±0.00 |
| | KS↑ | 0.92±0.01 | 0.92±0.03 | 0.92±0.03 | 0.92±0.04 |
| SeedPrints | AUC↑ | 0.99±0.01 | 0.98±0.01 | 0.93±0.04 | 0.93±0.01 |
| | KS↑ | 0.95±0.02 | 0.96±0.03 | 0.86±0.01 | 0.86±0.01 |
| RankMPT-WB | AUC↑ | 0.98±0.01 | 0.98±0.01 | 0.97±0.01 | 0.98±0.02 |
| | KS↑ | 0.95±0.00 | 0.96±0.00 | 0.95±0.00 | 0.96±0.02 |
| CoTRSF | AUC↑ | 0.56±0.00 | 0.65±0.08 | 0.62±0.00 | 0.55±0.04 |
| | KS↑ | 0.21±0.05 | 0.24±0.05 | 0.24±0.01 | 0.14±0.01 |
| LLMDNA | AUC↑ | 0.62±0.04 | 0.71±0.03 | 0.61±0.05 | 0.68±0.07 |
| | KS↑ | 0.33±0.05 | 0.34±0.04 | 0.23±0.02 | 0.36±0.05 |
| NT-FWER | AUC↑ | 0.72±0.09 | 0.29±0.01 | 0.35±0.09 | 0.79±0.01 |
| | KS↑ | 0.53±0.03 | 0.44±0.02 | 0.22±0.04 | 0.47±0.05 |
| RankMPT-BX | AUC↑ | 0.89±0.02 | 0.90±0.02 | 0.95±0.01 | 0.90±0.01 |
| | KS↑ | 0.69±0.03 | 0.76±0.01 | 0.83±0.01 | 0.74±0.00 |

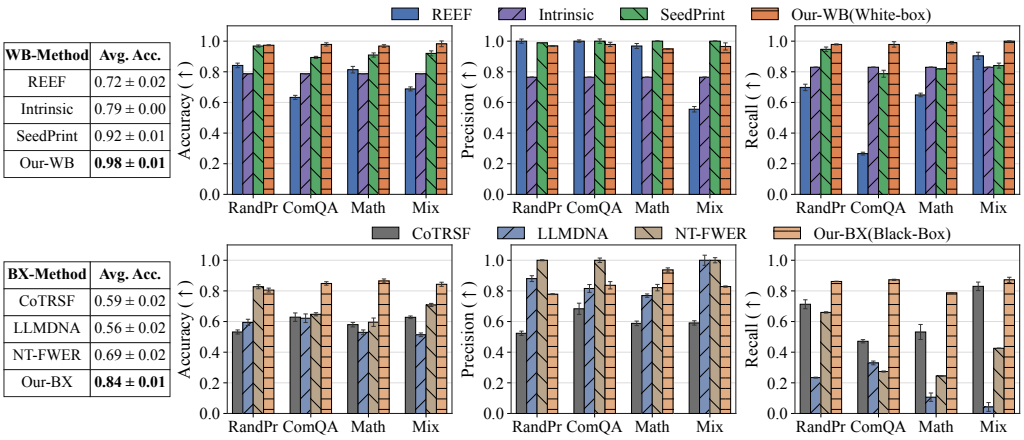


Figure 12: Performance of different methods on BenchA across four prompt sets.

mation, whereas black-box methods must estimate such information from finite output samples. Notably, both variants of our method remain stable across prompt sets, while competing methods exhibit larger performance variation, indicating greater robustness to prompt shifts.