STEMMA: INDUCED DECISION REGIONS REVEAL LLM PROVENANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM provenance testing asks whether a suspect LLM belongs to the same lineage as a source. Existing black-box methods largely rely on evidence tied to particular elicitation interfaces or response-level characteristics, but these signals may shift under prompting, adaptation, or deployment without changes in model lineage, weakening provenance reliability. To address this limitation, we introduce induced decision regions by mapping open-ended outputs into a shared finite decision space, abstracting away surface-form variation and reframing provenance testing around induced decision region inheritance. Empirical analysis shows stronger inheritance in related models and reveals that the provenance signal is unevenly distributed across probes, with discriminative power concentrated in a small subset whose informativeness transfers across model pairs. Building on this formulation, we propose Stemma, a black-box LLM fingerprinting method that estimates induced decision region inheritance using probes selected for stability, robustness, and specificity. Across 770 source–suspect pairs from 56 public checkpoints spanning diverse model transformations, Stemma achieves 0.967 AUC and 87.1% TPR at 1% FPR. It further achieves 0.995 AUC and 92.4% TPR at 1% FPR on 1,260 pairs covering 91 deployment instances, demonstrating robustness to deployment variations. Across both benchmarks, Stemma substantially outperforms four representative baselines.

1 INTRODUCTION

Large language models (LLMs) have become widely deployed infrastructure for text generation, code assistance, and reasoning. Across their development and deployment lifecycles, models may undergo fine-tuning, adapter tuning, merging, compression, or distillation, while prompt templates and decoding settings further shape their observable behaviour (Dettmers et al., 2023; Yu et al., 2024; Shi et al., 2024). This complexity obscures model provenance, raising intellectual-property and licensing concerns when protected models are reused without authorisation (Stalnaker et al., 2026), as well as accountability concerns when unsafe behaviours, biases, or vulnerabilities propagate to downstream models (Hammoud et al., 2024; Zhang et al., 2024). These concerns motivate model provenance testing, which assesses whether a suspect model belongs to the same lineage as a source.

Black-box fingerprinting provides a practical approach to provenance testing. It characterises a source through behavioural signatures preserved across provenance-related models but distinguishable from independently developed models (Jiang et al., 2026). Requiring only query access to the suspect model, it remains applicable when weights, activations, training data, and development history are unavailable. The central challenge is to identify signatures that are persistent under adaptation and specific to the source. In classification models, the fixed label space naturally partitions inputs into decision regions. Existing methods exploit this structure using boundary-adjacent, adversarial, or error-region inputs and test whether source and suspect models assign them to the same regions (Cao et al., 2021; Godinot et al., 2025; Guan et al., 2022). Derived models often inherit these fine-grained structures, whereas independently trained models are less likely to reproduce them.

This paradigm does not transfer directly to LLMs. Open-ended generation produces an unstructured output space in which the same meaning may take many surface forms, making decision regions difficult to define and compare from black-box outputs. Existing black-box LLM fingerprinting methods instead rely largely on specific prompting, output similarity, elicited behaviours, or pertur-

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.

bation responses (Nikolic et al., 2025; Gubri et al., 2024; Shao et al., 2026). Because such evidence is tied to particular input formats or output characteristics, it may shift under prompting, deployment, or model adaptation, weakening the reliability of provenance testing.

We address this limitation by introducing *induced decision regions* for LLMs. By inducing a finite decision space over open-ended model behaviour, we shift provenance comparison from prompt-dependent, surface-level response signals to abstract decisions, reducing sensitivity to prompt and response variation and reframing provenance testing around induced decision region inheritance.

Building on this formulation, we propose Stemma[1] as a black-box LLM fingerprinting method for provenance testing. Stemma further identifies reliable evidence of inheritance through probes selected for stability across equivalent input representations, robustness to boundary shifts, and specificity against unrelated models. Across model families, adaptation types, and deployment settings, Stemma achieves stronger and more consistent provenance separation than existing methods.

Contributions. Our contributions are summarised as follows.

- We introduce induced decision regions for LLMs, providing a more abstract and comparable basis for black-box provenance testing. We empirically show that related models preserve these regions more strongly than unrelated models and that discriminative information is concentrated in a small subset of probes.
- We propose and open-source Stemma, a black-box LLM fingerprinting method that identifies reliable provenance evidence of induced decision region inheritance through complementary principles of stability, robustness, and specificity.
- We conduct extensive evaluations across 56 public checkpoints and 91 deployment instances, spanning diverse model families, adaptation types, and deployment settings. Stemma substantially outperforms four representative black-box baselines across multiple metrics, demonstrating consistently stronger provenance separation.

2 Background and Problem Setting

2.1 Related Work

Methods for model ownership and provenance testing fall into watermarking and fingerprinting (Ye et al., 2026). Watermarking embeds identifiable signals into model parameters or behaviours, requiring prior control of the protected model and restricting verification to models marked in advance (Uchida et al., 2017; Adi et al., 2018; Li et al., 2023; Gloaguen et al., 2026). Fingerprinting is non-invasive, instead exploiting distinctive characteristics that arise naturally during model development. Depending on access to the suspect model, fingerprinting methods can be classified as white-box or black-box: white-box methods use parameters, activations, or internal representations and therefore require internal access (Zhang et al., 2025; Yu & Wang, 2024), whereas black-box methods rely only on behaviours elicited through controlled queries, offering broader applicability but making reliable provenance testing more challenging.

Existing black-box LLM fingerprinting methods can be broadly grouped by how they obtain and characterise observable behaviour. The first group directly characterises responses to selected queries. Model Provenance Testing measures next-token agreement between the source and suspect relative to independently trained control models, while DuFFin compares response similarity over sampled questions (Nikolic et al., 2025; Yan et al., 2026). LLMmap learns query-response representations for model identity recognition, whereas ErrorTrace derives family-level signatures from aggregated cross-lineage failure patterns (Pasquini et al., 2025; Zang et al., 2025). Recent API-auditing methods further exploit output-rank distributions or response stability near estimated knowledge boundaries (Zhu et al., 2026; Fang et al., 2026).

The second group constructs adversarially optimised prompts to elicit target behaviours. TRAP targets model identification, while ProFLingo and RAFP test downstream persistence (Gubri et al., 2024; Jin et al., 2024; Tsai et al., 2025). LLMPrint induces inherited target-token preferences

[1]The name Stemma is inspired by stemmatic analysis, which infers textual lineage from inherited variants. Appendix M develops this analogy.

(Hu et al., 2026), while concurrent Targeted Counterfactual Fingerprinting (TCF) constructs GCG-optimised counterfactual target transitions in a finite answer space and tests their transfer (Wu et al., 2026). A third line, represented by ZeroPrint, characterises response variation under semantic perturbations through zeroth-order local Jacobians (Shao et al., 2026). Across these approaches, fingerprint evidence is tied to specific elicitation interfaces or characterised by response patterns, induced behaviours, or perturbation effects. Such signals may shift with prompting, deployment, or weight adaptation without changes in lineage, limiting stability and comparability of provenance evidence.

Prior work on deep classifiers shows that decision regions provide a structured and directly comparable basis for black-box provenance testing. IPGuard constructs probes near the source's decision boundaries and tests whether the suspect model preserves the region assignments (Cao et al., 2021). Model Lineage Closeness Analysis measures preservation of decision geometry through differences in decision boundary distances and prediction agreement (Tang et al., 2025). ADV-TRA extends boundary-sensitive probes to adversarial trajectories that traverse decision boundaries, improving tolerance to boundary shifts (Xu et al., 2024). Open-ended LLM generation, however, does not naturally expose a finite decision space over which analogous regions can be defined and compared. Stemma therefore induces finite decision regions from open-ended outputs and tests provenance through their inheritance, providing a more abstract and robust basis for provenance testing.

2.2 Problem Setting

We study pairwise black-box model provenance testing for LLMs. Given a source model S and a suspect model instance T, the objective is to determine whether T is provenance-related to S. A model instance comprises an underlying model and its deployment configuration. The pair is related if the underlying models of S and T are identical or connected through derivation or shared ancestry, possibly under different deployment configurations. Otherwise, the pair is unrelated, even if the models share similarities in architecture, training, or performance.

Access assumptions. We assume access to the source model's next-token logits for fingerprint construction, whereas the suspect model is available only through a black-box interface that accepts textual inputs and returns generated responses without exposing its parameters, logits, activations, training data, or development history. This asymmetric setting captures a practical provenance-auditing scenario in which the evaluator controls the source model and seeks to identify related model variants or deployed instances through exposed interfaces.

Adversarial scope. We consider a non-adaptive setting in which the provider may know the general fingerprinting method but not the test-specific probes. The suspect model may undergo ordinary model development and deployment transformations, but neither the model nor its interface is adapted to evade the method by detecting, filtering, or altering responses to fingerprint queries.

3 Induced Decision Regions for LLM Provenance

3.1 Formalisation

Formally, let $\mathcal{Q}$ denote the query space, $\mathcal{O}$ the observable model-output space, and $\mathcal{D} = \{1, \ldots, C\}$ a finite decision space. For an LLM instance M, let $o_M(q) \in \mathcal{O}$ denote its output on query $q \in \mathcal{Q}$. Given a score extraction rule $\tilde{g} : \mathcal{Q} \times \mathcal{O} \to \mathbb{R}^C$ and a decision extraction rule $g : \mathcal{Q} \times \mathcal{O} \to \mathcal{D}$, we define the model-specific soft and discrete decision maps as

$$\tilde{h}_M(q) = \tilde{g}\big(q, o_M(q)\big) \in \mathbb{R}^C, \qquad h_M(q) = g\big(q, o_M(q)\big) \in \mathcal{D}. \tag{1}$$

The soft map $\tilde{h}_M$ represents the model's relative preferences over the available decisions, while h_M denotes the selected decision. Depending on the available access, $h_M(q)$ is obtained either by taking $\arg\max_{d \in \mathcal{D}} \tilde{h}_{M,d}(q)$, with ties resolved deterministically, or by extracting it from generated text. In both cases, valid decisions lie in the same finite space $\mathcal{D}$ (practical instantiations may additionally yield invalid outcomes beyond this abstract formulation).

The discrete decision map h_M induces a partition of the query space by grouping queries assigned to the same decision. For each decision $d \in \mathcal{D}$, we define the induced decision region of M as

$$\mathcal{R}_d^M = \{q \in \mathcal{Q} \mid h_M(q) = d\}. \tag{2}$$

Conceptually, we refer to the arrangement of these regions and their separating boundaries as the induced decision geometry of M.

For provenance testing, we use *induced decision region inheritance* to denote the extent to which induced decision geometry is preserved between a source model S and a suspect model T. Since this geometry cannot be compared exhaustively over $\mathcal{Q}$, we assess its preservation using a finite probe set $\mathcal{P} \subset \mathcal{Q}$. Depending on the available output information, we quantify preservation using soft similarity or hard alignment:

$$\widehat{\mathrm{Sim}}_{\mathcal{P}}(S,T) = \frac{1}{|\mathcal{P}|}\sum_{q\in\mathcal{P}} \mathrm{pres}_{\mathrm{soft}}(S,T;q), \qquad \mathrm{pres}_{\mathrm{soft}}(S,T;q) = \mathrm{sim}\Big(\tilde{h}_S(q), \tilde{h}_T(q)\Big),$$
$$\widehat{\mathcal{A}}_{\mathcal{P}}(S,T) = \frac{1}{|\mathcal{P}|}\sum_{q\in\mathcal{P}} \mathrm{pres}_{\mathrm{hard}}(S,T;q), \quad \mathrm{pres}_{\mathrm{hard}}(S,T;q) = \mathbf{1}\{h_T(q) = h_S(q)\}. \tag{3}$$

Here, $\mathrm{sim}(\cdot,\cdot)$ denotes a similarity measure between soft decision vectors, while $\mathrm{pres}_{\mathrm{soft}}$ and $\mathrm{pres}_{\mathrm{hard}}$ denote the corresponding per-question preservation scores for soft similarity and hard alignment, respectively. Higher values of $\widehat{\mathrm{Sim}}_{\mathcal{P}}$ and $\widehat{\mathcal{A}}_{\mathcal{P}}$ indicate stronger preservation of the source model's induced decision geometry and hence stronger evidence of a provenance relationship.

3.2 Characterisation of Inheritance

We next examine whether and how induced decision region inheritance is observable in LLM behaviour. In general, any evaluation interface that maps open-ended outputs to a finite decision space can induce such regions, including constrained response formats, semantic output categories, and discrete preference judgments. In this paper, we instantiate this idea using multiple-choice questions, where the answer options define the decision space $\mathcal{D}$, as formalised in Appendix A.

Specifically, we analyse all MMLU test questions (Hendrycks et al., 2021) using Qwen2.5-7B as the source together with three transformation-matched pairs covering instruction-tuned, quantised, and distilled descendants. Each pair contains one model related to the source and one unrelated comparison model from the Qwen3-1.7B lineage. We normalise next-token logits over the four answer options into a probability distribution and measure soft similarity as $1 - \mathrm{TV}$, where TV denotes total variation distance. Hard alignment records whether the corresponding discrete decisions agree. Model identities and analysis details are provided in Appendix B.2.

Induced decision region inheritance. Figure 1(a) shows that related models exhibit consistently higher soft similarity and hard alignment to the source than unrelated models, providing direct empirical evidence of induced decision region inheritance.

Informativeness distribution. For $v \in \{\mathrm{soft}, \mathrm{hard}\}$, we define the informativeness of a probe set P as the average preservation gap between related and unrelated models across its probes:

$$\mathcal{I}_v(P) = \frac{1}{|P|}\sum_{q\in P} d_v(q), \qquad d_v(q) = \frac{1}{|R|}\sum_{T\in R} \mathrm{pres}_v(S,T;q) - \frac{1}{|U|}\sum_{T\in U} \mathrm{pres}_v(S,T;q). \tag{4}$$

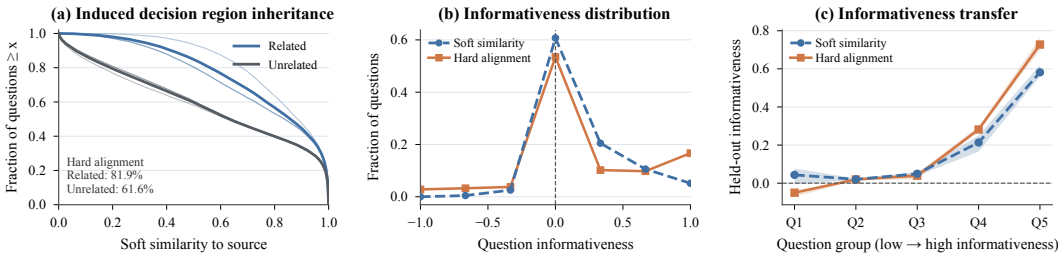

Figure 1: (a) CCDFs of soft similarity for related and unrelated models, with hard alignment for reference. (b) Question informativeness distributions for soft similarity and hard alignment. (c) Informativeness transfer across model pairs. Questions are ranked by soft-similarity informativeness on one pair and divided into five approximately equal groups Q_1–Q_5. Curves show held-out informativeness, with shading indicating the range across three folds.

Here, R and U denote the related and unrelated model sets, and $d_v(q) = \mathcal{I}_v(\{q\})$ is the informativeness of an individual probe. Figure 1(b) shows that informativeness is highly uneven across questions. More than half have near-zero or negative informativeness, providing little discriminative evidence or even evidence in the opposite direction, while a smaller subset exhibits substantially positive informativeness.

Informativeness transfer. To serve as probes for fingerprint construction, questions identified as informative should remain discriminative on held-out model pairs. Figure 1(c) evaluates this transfer using three folds. In each fold, one transformation-matched pair ranks questions by soft-similarity informativeness and partitions them into five approximately equal groups Q_1–Q_5, from lowest to highest informativeness. The remaining two pairs then measure the mean held-out informativeness of each group under soft similarity and hard alignment. The overall increase from Q_1 to Q_5 shows that question informativeness transfers across model pairs, with higher-ranked groups remaining more discriminative on held-out pairs.

Implications for fingerprint construction. Taken together, these analyses suggest that induced decision region inheritance can provide a practical basis for fingerprint construction using a limited number of probes. However, informativeness is an analytical measure rather than a practical construction objective, as it requires provenance-labelled model pairs and does not fully characterise fingerprint performance (Appendix B.1). Accordingly, rather than using informativeness as a direct construction objective, we propose Stemma, which constructs compact fingerprints using complementary properties of source decisions.

4 Stemma Design

Stemma is a black-box LLM fingerprinting method that tests provenance by estimating induced decision region inheritance from a limited set of queries. It instantiates induced decision regions using multiple-choice questions, whose candidate options define a finite decision space, and operates in three stages. Decision interface calibration selects and fixes a prompt template for each model under the relevant access mode to enable reliable decision extraction. Fingerprint construction selects probes by evaluating the source model's induced decisions in terms of stability, robustness, and specificity. Fingerprint verification queries the suspect model with these probes and measures how strongly it preserves the source decisions recorded in the fingerprint.

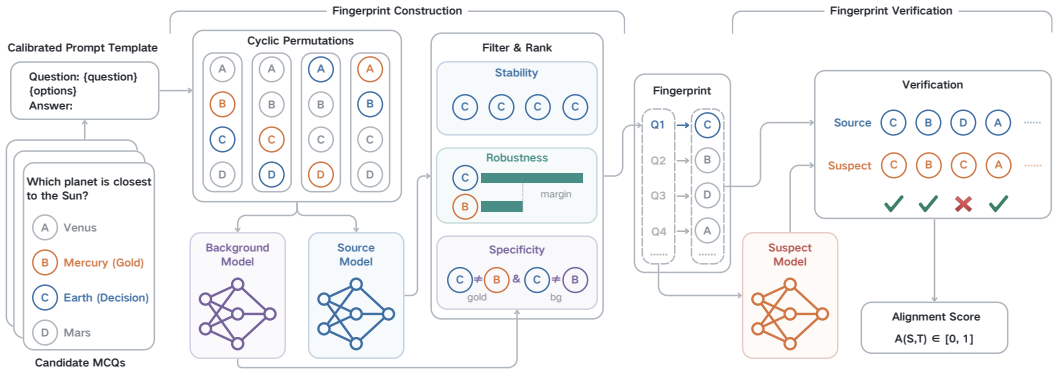

Figure 2: Overview of the Stemma workflow.

4.1 Decision Interface Calibration

Some multiple-choice prompt formats may produce unparseable responses or severe label bias, preventing reliable extraction of induced decisions. Stemma therefore calibrates each model under the relevant access mode on held-out questions by evaluating a predefined set of templates and selecting the one with the highest valid-choice extraction rate among those without severe label bias. Because all templates preserve the same questions and candidate options, calibration changes only the interface rendering while extracted decisions remain comparable in the same canonical decision space.

Importantly, calibration is performed only once before fingerprint construction or verification, after which the selected template is fixed for subsequent queries. The candidate templates and calibration configuration are provided in Appendix C.

4.2 Fingerprint Construction

Stemma constructs a source fingerprint by selecting probes whose induced decisions satisfy three complementary criteria for reliable inheritance testing, namely stability, robustness, and specificity.

Stability. Stability requires a probe's induced decision to remain largely consistent across semantically equivalent input representations, identifying probes insensitive to option order and choice-label bias (Zheng et al., 2024a; Pezeshkpour & Hruschka, 2024). Accordingly, Stemma evaluates each candidate under cyclic permutations of its answer options, such that every semantic option appears once under each choice label. After mapping the decisions back to the canonical options, stability is measured by their consistency rate across permutations, with higher rates indicating greater stability.

Robustness. Robustness requires a probe's induced decision to be separated from competing decisions, making its region assignment resistant to small boundary shifts. A probe may be consistent across option permutations yet remain close to a competing boundary, making its region assignment vulnerable to downstream adaptation or deployment changes. Stemma measures robustness by computing, for each option permutation, the log-probability margin between the source decision and the highest-scoring alternative, and then averaging these margins across permutations. Larger average margins indicate greater robustness.

Specificity. Specificity requires a probe's source decision to be uncommon among unrelated models, so that its preservation provides discriminative evidence of induced decision region inheritance. Because gold-answer decisions are often shared across related and unrelated models, Stemma first filters out questions whose source decisions match the gold answers. However, non-gold decisions may still be common because of question ambiguity, annotation errors, or attractive distractors. Stemma therefore measures how often an unrelated background model reproduces the source decision, with lower reproduction rates indicating greater specificity.

Probe Selection and Fingerprint Formation. For each candidate q, Stemma obtains source-model next-token logits over choice labels under cyclic option permutations, maps each permutation-level decision back to the canonical options, and defines $h_S(q)$ as the most frequently selected canonical decision. The three selection principles are operationalised by the permutation consistency rate $s(q)$ for stability, the average log-probability margin $r(q)$ for robustness, and the background decision alignment rate $b(q)$, together with the non-gold requirement $h_S(q) \neq d^*(q)$, for specificity. Here, $d^*(q)$ denotes the gold answer.

Stemma first applies the eligibility filters to the candidate pool, yielding

$$\mathcal{Q}_S^{\text{elig}} = \{q \in \mathcal{Q}_{\text{cand}} \mid s(q) > \tau_s,\ r(q) \geq \tau_r,\ b(q) < \tau_b,\ h_S(q) \neq d^*(q)\}\,, \tag{5}$$

where τ_s and τ_r are the minimum stability and robustness thresholds, respectively, and τ_b is the maximum background-alignment threshold. It then ranks the eligible candidates by robustness and specificity and forms the fingerprint from the top K probes:

$$\mathcal{F}_S = \left\{ (q, h_S(q)) \;\middle|\; q \in \text{TopK}_{q' \in \mathcal{Q}_S^{\text{elig}}} \left[w_r \widetilde{r}(q') - w_b \widetilde{b}(q') \right] \right\}, \tag{6}$$

where $\widetilde{r}$ and $\widetilde{b}$ are transformed versions of the robustness margin and background alignment rate, respectively, and w_r and w_b control their contributions. Exact score transformations, weights, and thresholds are provided in Appendix C. As a post-hoc diagnostic, the probe sets selected by Stemma also exhibit substantially higher informativeness, with the top-ranked 1% achieving more than twice the soft informativeness of equally sized random subsets (Appendix D).

4.3 Fingerprint Verification

Given a source fingerprint $\mathcal{F}_S$ and a suspect model T, Stemma queries each selected probe through T's calibrated prompt template under the same cyclic option permutations used in fingerprint construction. A rule-based extractor maps each generated response to either a displayed choice label or

an invalid outcome $\perp$. Valid extracted options are mapped back to the canonical decision space and compared with the stored source decision, while invalid outcomes are treated as non-aligned. These permutation-level comparisons are aggregated into the fingerprint alignment score:

$$A(S,T) = \frac{1}{KC} \sum_{(q,h_S(q))\in\mathcal{F}_S} \sum_{\pi\in\Pi(q)} \mathbf{1}\left[\hat{d}_T(q,\pi) = h_S(q)\right]. \tag{7}$$

Here, $\Pi(q)$ denotes the C cyclic permutations of probe q, and $\hat{d}_T(q,\pi) \in \mathcal{D} \cup \{\perp\}$ denotes the suspect outcome after mapping each valid extracted option back to the canonical decision space. Thus, $A(S,T) \in [0,1]$ is the fraction of the KC probe–permutation observations that preserve the source region assignment, with invalid outcomes contributing zero. Higher values indicate stronger induced decision region inheritance and hence stronger evidence of a provenance relationship.

5 Experiments

5.1 Experimental Setup

Benchmark and evaluation protocol. We construct a benchmark of 56 public checkpoints organised into 7 provenance groups spanning Qwen, Llama, Mistral, Falcon, and OLMo. Each group contains a pretrained source, an instruction-tuned source, and 6 additional variants spanning fine-tuning, parameter-efficient adaptation, model merging, quantisation, and distillation. Three of the groups are Qwen-based, providing hard negatives with similar architectures and training recipes but distinct provenance. Each source is evaluated against every other checkpoint, with within-group pairs labelled positive and cross-group pairs labelled negative. This yields 770 ordered pairs between sources and suspects, of which 98 are positive and 672 negative. For black-box querying, each checkpoint is queried using its recommended raw or chat interface.

Experimental setup and baselines. Unless otherwise stated, Stemma selects 40 probes per source from 3,000 randomly sampled MMLU questions and evaluates each under four cyclic permutations, yielding 160 fingerprint queries per verification. Phi-3.5-mini-instruct is used as the default background model, with no known provenance relationship with any benchmark model. We compare against four black-box LLM fingerprinting baselines: LLMmap, a reduced-budget LLMPrint, Model Provenance Testing (MPT), and ZeroPrint (Pasquini et al., 2025; Hu et al., 2026; Nikolic et al., 2025; Shao et al., 2026). Experiments are conducted on NVIDIA L40S GPUs with 48 GB of memory.

Evaluation metrics. We evaluate all methods using continuous provenance scores and report AUC to assess overall ranking performance. Since false-positive provenance claims pose substantial legal and reputational risks, we additionally report standardised pAUC over the FPR range $[0, 0.05]$ and TPR at 1% FPR for stringent low-FPR evaluation. We also report the discriminability index d' to quantify the separation between positive and negative score distributions. Unless otherwise stated, we report the mean and sample standard deviation over five runs using seeds 42–46. Appendix C details the Stemma configuration, while Appendices E–G detail benchmark construction and evaluation protocol, metric definitions and implementation, and baseline configurations, respectively.

5.2 Effectiveness on the Main Benchmark

Table 1: Performance on the main provenance benchmark. Best/second-best are bold/underlined.

Method	Pretrained sources				Instruct sources				All sources			
	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑
LLMmap	0.471 ±0.009	0.488 ±0.002	0.000 ±0.000	−0.292 ±0.019	<u>0.840</u> ±0.014	0.697 ±0.013	0.343 ±0.081	<u>1.381</u> ±0.067	<u>0.654</u> ±0.007	0.560 ±0.010	0.098 ±0.037	0.593 ±0.031
LLMPrint	0.478 ±0.041	0.497 ±0.008	0.008 ±0.011	−0.060 ±0.106	0.614 ±0.028	0.561 ±0.026	0.029 ±0.064	0.297 ±0.118	0.541 ±0.025	0.537 ±0.009	0.039 ±0.056	0.114 ±0.070
MPT	0.459 ±0.004	<u>0.498</u> ±0.000	<u>0.020</u> ±0.000	−0.279 ±0.003	0.722 ±0.001	0.694 ±0.002	0.343 ±0.009	1.046 ±0.002	0.555 ±0.001	<u>0.613</u> ±0.000	<u>0.192</u> ±0.005	0.500 ±0.001
ZeroPrint	<u>0.563</u> ±0.039	0.496 ±0.002	0.016 ±0.009	<u>0.067</u> ±0.080	0.775 ±0.007	<u>0.760</u> ±0.008	<u>0.498</u> ±0.018	1.196 ±0.034	0.623 ±0.016	0.611 ±0.005	0.182 ±0.020	<u>0.608</u> ±0.032
Stemma	**0.961** ±0.007	**0.948** ±0.014	**0.882** ±0.044	**2.861** ±0.210	**0.974** ±0.004	**0.945** ±0.010	**0.878** ±0.025	**2.964** ±0.049	**0.967** ±0.005	**0.944** ±0.002	**0.871** ±0.020	**2.864** ±0.130

Table 1 shows that Stemma consistently outperforms all baselines for both pretrained and instruction-tuned sources, as well as in aggregate, on discrimination and low-FPR metrics.

The gap is pronounced for pretrained sources, where source fingerprints use raw completion prompts while many deployed suspects are queried through chat templates. Under this interface difference, baselines approach random performance, suggesting that their provenance signals are sensitive to changes in the elicitation interface. Baselines improve for instruction-tuned sources, where black-box queries use chat templates, but do not close the gap to Stemma despite this input alignment. This suggests that Stemma's combination of induced decision regions and targeted probe selection reduces sensitivity to input-format and output surface-form variation while achieving stronger provenance separation. Further analyses are reported in Appendix H.

To control for interface heterogeneity, we evaluate an all-raw setting where fingerprint construction and verification use raw prompts. Under this uniform-format setting, Stemma trails MPT in AUC (0.967 vs. 0.989), while remaining stronger at low FPR (0.951 pAUC and 0.894 TPR@1% FPR vs. 0.946 and 0.849). Diagnostic analysis shows that this AUC gap is largely driven by two OLMo domain adapters that produce invalid responses without expected chat formatting. Excluding these suspects removes the gap (0.990 vs. 0.989 AUC). Thus, under conditions favourable to the baselines, Stemma retains stronger low-FPR discrimination, while its AUC deficit is largely explained by degraded task compliance under non-native raw prompting. Full results are reported in Appendix I.

5.3 Robustness under Deployment Variations

Table 2: Performance under deployment variants. Best/second-best are bold/underlined.

Method	Pretrained sources				Instruct sources				All sources			
	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑
LLMmap	0.482 ±0.004	0.487 ±0.000	<u>0.000</u> ±0.000	−0.209 ±0.016	<u>0.914</u> ±0.011	0.760 ±0.009	0.448 ±0.060	<u>1.881</u> ±0.092	<u>0.691</u> ±0.005	0.602 ±0.006	0.135 ±0.019	<u>0.804</u> ±0.022
LLMPrint	0.486 ±0.015	<u>0.489</u> ±0.001	<u>0.000</u> ±0.000	−0.010 ±0.066	0.610 ±0.015	0.513 ±0.008	0.000 ±0.000	0.405 ±0.033	0.539 ±0.011	0.508 ±0.007	0.000 ±0.000	0.165 ±0.046
MPT	0.481 ±0.003	0.487 ±0.000	<u>0.000</u> ±0.000	−0.247 ±0.003	0.778 ±0.001	0.728 ±0.000	0.452 ±0.000	1.220 ±0.003	0.571 ±0.001	0.621 ±0.000	0.226 ±0.000	0.562 ±0.001
ZeroPrint	<u>0.592</u> ±0.022	0.489 ±0.004	<u>0.000</u> ±0.000	<u>0.120</u> ±0.063	0.868 ±0.010	<u>0.827</u> ±0.008	<u>0.631</u> ±0.022	1.710 ±0.043	0.657 ±0.010	<u>0.654</u> ±0.008	<u>0.292</u> ±0.047	0.766 ±0.025
Stemma	**0.994** ±0.005	**0.962** ±0.021	**0.883** ±0.060	**3.479** ±0.243	**0.997** ±0.002	**0.983** ±0.009	**0.948** ±0.034	**4.186** ±0.355	**0.995** ±0.003	**0.973** ±0.013	**0.924** ±0.028	**3.599** ±0.209

To evaluate inference-time deployment robustness, we construct 11 variants for each of the 7 instruction-tuned checkpoints, spanning system and role-play prompts, reasoning-oriented prompting, retrieval augmentation, and decoding settings. Together with 14 source checkpoints, this yields 91 deployment instances and 1,260 source–suspect pairs, with 168 positive and 1,092 negative pairs.

Table 2 shows that Stemma maintains near-perfect AUC and substantially stronger low-FPR performance across these variations. Existing methods perform particularly poorly for pretrained sources and improve for instruction-tuned sources, but still fall consistently below Stemma. These results demonstrate Stemma's robustness under diverse black-box deployment configurations, consistent with reduced sensitivity to deployment-induced interface and response variation. Full benchmark details and supporting results are provided in Appendix J.

5.4 Ablation and Efficiency

The probe selection panel of Figure 3 isolates successive selection stages under the same query budget. The Random probes setting samples uniformly from the full candidate pool while retaining the same cyclic-permutation evaluation and aggregation. Its performance confirms that induced decision region inheritance is observable without targeted selection, although random probes provide weaker discrimination than the progressively selected probes. Restricting candidates to the non-gold pool improves specificity, while subsequent filtering removes probes that fail the stability, robustness, or specificity requirements. Finally, ranking the retained candidates and selecting the top probes yields the strongest performance. Together, these incremental gains support the proposed selection pipeline, while the larger gains in pAUC indicate that targeted selection is particularly valuable for distinguishing difficult-to-separate model pairs.

The pool size and probe budget panel examines how performance changes with the number of candidate questions and selected probes. A pool of 1,000 questions yields noticeably weaker performance. Results generally improve with 3,000–4,000 questions compared with 2,000, while increasing to

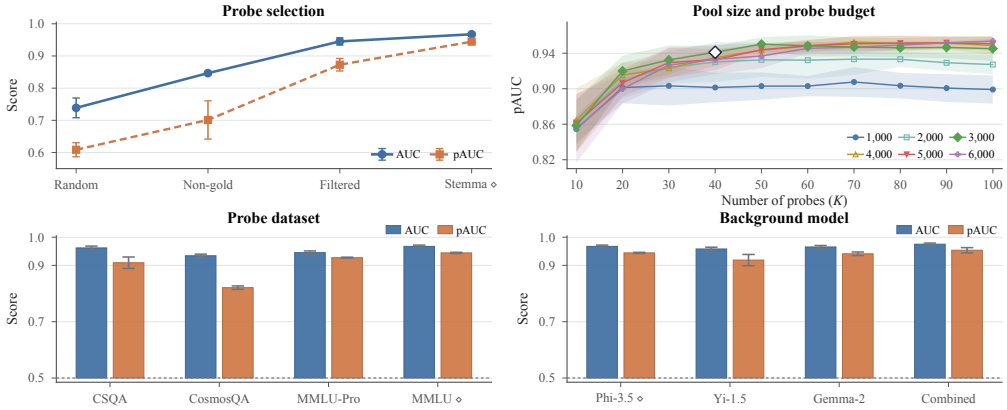

Figure 3: Ablation study of probe selection, pool size and probe budget, probe dataset, and background model. ⋄ marks the default. Error bars and shading indicate ± 1 SD over five runs.

5,000–6,000 provides no consistent further benefit, indicating that larger pools are not necessarily better. For the probe budget, performance improves rapidly up to $K = 40$ and changes only modestly thereafter, indicating limited sensitivity to exact budgets. The corresponding AUC results follow the same overall pattern, but with less pronounced differences. Thus, Stemma performs consistently across a broad range of pool sizes and probe budgets.

Differences across probe datasets are more pronounced in pAUC than in AUC, with MMLU-Pro and MMLU showing stronger aggregate low-FPR performance than the commonsense-oriented datasets, suggesting that dataset-dependent probe characteristics matter more under stringent low-FPR evaluation. The background-model panel shows that performance is relatively insensitive to background choice. Although combining all three backgrounds yields modest improvements, a single unrelated background model appears practically adequate. Detailed results are provided in Appendix K.

Computational cost. Stemma requires median runtimes of 269 seconds (4.5 minutes) for one-time fingerprint construction, including calibration, and 38 seconds per verification, within baseline ranges of 17–29,380 seconds and 3–55 seconds, respectively. These costs remain comparatively modest, supporting the practical efficiency of Stemma (Appendix L).

6 Discussion and Conclusion

Limitations and Future Work. Our current instantiation relies on multiple-choice questions and therefore assumes that the evaluated models can reliably process such questions and produce stable option-level decisions. Although practical provenance testing is likely to focus on capable models, this requirement may limit applicability to weaker models. However, multiple-choice questions are only one mechanism for inducing decision regions, and future work could explore more general interfaces for obtaining structured and observable decisions.

We also do not consider adaptive attackers with knowledge of the probe distribution. Because our candidate probes are drawn primarily from a single public question set, an attacker could deliberately alter the suspect model's behaviour on the corresponding question distribution, thereby weakening detection. This risk could be mitigated by constructing candidate pools from a diverse mixture of datasets or privately constructed question sets. Future work should further characterise the robustness limits of provenance testing under adaptive evasion.

Conclusion. We introduced induced decision regions for LLMs by mapping open-ended outputs into a finite decision space, enabling provenance to be assessed through abstract model decisions rather than particular prompt and response forms. Building on this formulation, we proposed Stemma, a black-box LLM fingerprinting method that selects probes for stability across equivalent input representations, robustness to boundary shifts, and specificity against unrelated models, allowing induced decision region inheritance to be assessed with a limited query budget. Across extensive evaluations, Stemma consistently demonstrated strong provenance separation, showing that induced decision region inheritance provides a robust signal for LLM provenance testing.

AI USAGE STATEMENT

We used generative AI tools for proofreading and language polishing, assistance with software code development and debugging, and literature discovery. Generative AI was not used to develop the core research ideas or conceptual framework, design the research methodology or experiments, or interpret the experimental results. All AI-assisted work was reviewed and verified by the authors.

ETHICS STATEMENT

Stemma scores provide statistical evidence of a potential provenance relationship rather than definitive proof of model ownership or unauthorised reuse. The method could be misused to support false provenance claims, and false positives may cause legal or reputational harm. Stemma should therefore not be used as the sole basis for attribution or enforcement decisions, but interpreted alongside model documentation, licensing and development records, and other technical evidence. Our experiments use only publicly available model checkpoints and benchmark datasets and involve no human subjects or private user data.

REPRODUCIBILITY STATEMENT

The complete Stemma procedure is described in Section 4, with decision interface calibration, fingerprint construction, fingerprint verification, and hyperparameter settings specified in Appendix C. The models used in the main experiments, provenance-group construction, and pair-labelling protocol are documented in Appendix E. Appendix F provides the exact evaluation-metric definitions and implementation details, while Appendix G describes the baseline implementations and configurations. The all-raw and deployment-robustness benchmark settings are specified in Appendices I and J, respectively. Full ablation results and runtime measurements are reported in Appendices K and L. The open-source implementation and configurations for Stemma are available at `https://anonymous.4open.science/r/submission-2B21/README.md`.

REFERENCES

Yossi Adi, Carsten Baum, Moustapha Cisse, Benny Pinkas, and Joseph Keshet. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. In *27th USENIX Security Symposium (USENIX Security 18)*, pp. 1615–1631. USENIX Association, 2018. doi: 10.5555/3277203.3277324.

Tara L. Andrews. Analysis of variation significance in artificial traditions using Stemmaweb. *Digital Scholarship in the Humanities*, 31(3):523–539, September 2016. ISSN 2055-7671. doi: 10.1093/llc/fqu072.

Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. IPGuard: Protecting Intellectual Property of Deep Neural Networks via Fingerprinting the Classification Boundary. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, ASIA CCS '21, pp. 14–25, New York, NY, USA, June 2021. Association for Computing Machinery. ISBN 978-1-4503-8287-8. doi: 10.1145/3433210.3437526.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient Finetuning of Quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 10088–10115. Curran Associates, Inc., 2023.

Yijia Fang, Yiqing Feng, Bingyu Li, and Mingxun Zhou. KBF: Knowledge Boundary as Fingerprint for Language Model and Black-Box API Auditing. arXiv preprint arXiv:2605.29524, May 2026.

Thibaud Gloaguen, Robin Staab, Nikola Jovanović, and Martin Vechev. LLM Fingerprinting via Semantically Conditioned Watermarks. In *The Fourteenth International Conference on Learning Representations*, 2026.

Augustin Godinot, Erwan Le Merrer, Camilla Penzo, François Taïani, and Gilles Trédan. Queries, representation & detection: The next 100 model fingerprinting schemes. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, volume 39 of *AAAI'25/IAAI'25/EAAI'25*, pp. 16817–16825. AAAI Press, February 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i16.33848.

Jiyang Guan, Jian Liang, and Ran He. Are You Stealing My Model? Sample Correlation for Fingerprinting Deep Neural Networks. In *Advances in Neural Information Processing Systems*, volume 35, pp. 36571–36584. Curran Associates, Inc., 2022.

Martin Gubri, Dennis Ulmer, Hwaran Lee, Sangdoo Yun, and Seong Joon Oh. TRAP: Targeted Random Adversarial Prompt Honeypot for Black-Box Identification. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 11496–11517, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.683.

Hasan Abed Al Kader Hammoud, Umberto Michieli, Fabio Pizzati, Philip Torr, Adel Bibi, Bernard Ghanem, and Mete Ozay. Model Merging and Safety Alignment: One Bad Model Spoils the Bunch. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 13033–13046, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.762.

Tuomas Heikkilä and Teemu Roos. Thematic Section on Studia Stemmatologica. *Digital Scholarship in the Humanities*, 31(3):520–522, September 2016. ISSN 2055-7671. doi: 10.1093/llc/fqw038.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*, 2021.

Yuepeng Hu, Zhengyuan Jiang, Mengyuan Li, Osama Ahmed, Zhicong Huang, Cheng Hong, and Neil Zhenqiang Gong. Fingerprinting LLMs via Prompt Injection. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11795–11810, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.541.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Cosmos QA: Machine reading comprehension with contextual commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2391–2401, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1243.

Hugging Face. Hugging Face Hub, 2026. URL `https://huggingface.co/models`.

Yongqi Jiang, Yansong Gao, Chunyi Zhou, Hongsheng Hu, Si Chen, Anmin Fu, and Willy Susilo. Intellectual property protection for deep learning model and dataset intelligence. *Engineering Applications of Artificial Intelligence*, 163:113024, January 2026. ISSN 0952-1976. doi: 10.1016/j.engappai.2025.113024.

Heng Jin, Chaoyu Zhang, Shanghao Shi, Wenjing Lou, and Y. Thomas Hou. ProFLingo: A Fingerprinting-based Intellectual Property Protection Scheme for Large Language Models. In *2024 IEEE Conference on Communications and Network Security (CNS)*, pp. 1–9, September 2024. doi: 10.1109/CNS62487.2024.10735575.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large Language Models are Zero-Shot Reasoners. *Advances in Neural Information Processing Systems*, 35:22199–22213, December 2022.

Linyang Li, Botian Jiang, Pengyu Wang, Ke Ren, Hang Yan, and Xipeng Qiu. Watermarking LLMs with Weight Quantization. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 3368–3378, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.220.

Donna K. McClish. Analyzing a portion of the ROC curve. *Medical Decision Making*, 9(3):190–195, 1989. doi: 10.1177/0272989X8900900307.

Ivica Nikolic, Teodora Baluta, and Prateek Saxena. Model Provenance Testing for Large Language Models. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025. doi: 10.52202/085713-1146.

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. ISBN 978-1-939133-52-6.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162.

Pouya Pezeshkpour and Estevam Hruschka. Large Language Models Sensitivity to The Order of Options in Multiple-Choice Questions. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2006–2017, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.130.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410.

Shuo Shao, Yiming Li, Hongwei Yao, Yifei Chen, Yuchen Yang, and Zhan Qin. Reading between the lines: Towards reliable black-box LLM fingerprinting via zeroth-order gradient estimation. In *Proceedings of the ACM Web Conference 2026*, pp. 2637–2648, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 979-8-4007-2307-0. doi: 10.1145/3774904.3792196.

Chufan Shi, Haoran Yang, Deng Cai, Zhisong Zhang, Yifan Wang, Yujiu Yang, and Wai Lam. A Thorough Examination of Decoding Methods in the Era of LLMs. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8601–8629, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.489.

Trevor Stalnaker, Nathan Wintersgill, Oscar Chaparro, Laura A. Heymann, Massimiliano Di Penta, Daniel M German, and Denys Poshyvanyk. An Empirical Analysis of Machine Learning Model and Dataset Documentation, Supply Chain, and Licensing Challenges on Hugging Face. *ACM Transactions on Software Engineering and Methodology*, 35(8):1–49, July 2026. doi: 10.1145/3776739.

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, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421.

Chen Tang, Lan Zhang, Qi Zhao, Xirong Zhuang, and Xiang-Yang Li. Model Lineage Closeness Analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(19):20796–20804, April 2025. doi: 10.1609/aaai.v39i19.34292.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv preprint arXiv:2307.09288, July 2023.

Yun-Yun Tsai, Jia Hao Liang, Chuan Guo, Junfeng Yang, and Laurens van der Maaten. RAFP: Identifying LLM lineages via rare-region fingerprints. arXiv preprint arXiv:2505.12682, May 2025.

Yusuke Uchida, Yuki Nagai, Shigeyuki Sakazawa, and Shin'ichi Satoh. Embedding Watermarks into Deep Neural Networks. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, ICMR '17, pp. 269–277, New York, NY, USA, June 2017. Association for Computing Machinery. ISBN 978-1-4503-4701-3. doi: 10.1145/3078971.3078974.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. Plan-and-Solve Prompting: Improving Zero-Shot Chain-of-Thought Reasoning by Large Language Models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2609–2634, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.147.

Noah Wang, Z.y. Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, Man Zhang, Zhaoxiang Zhang, Wanli Ouyang, Ke Xu, Wenhao Huang, Jie Fu, and Junran Peng. RoleLLM: Benchmarking, Eliciting, and Enhancing Role-Playing Abilities of Large Language Models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14743–14777, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.878.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, volume 37, pp. 95266–95290, 2024b. doi: 10.52202/079017-3018.

Yutong Wu, Xiaofan Bai, Shixin Li, Pingyi Hu, Ziqi Zhou, Zilong Wang, Xiaojing Ma, Songfeng Lu, Yuhong Li, Jin Xuan, Yi Wang, Dongmei Zhang, and Bin Benjamin Zhu. Targeted counterfactual fingerprinting for black-box LLM ownership verification. arXiv preprint arXiv:2608.08195, August 2026.

Tianlong Xu, Chen Wang, Gaoyang Liu, Yang Yang, Kai Peng, and Wei Liu. United We Stand, Divided We Fall: Fingerprinting Deep Neural Networks via Adversarial Trajectories. In *Advances*

in Neural Information Processing Systems, volume 37, pp. 69299–69328. Curran Associates, Inc., 2024. doi: 10.52202/079017-2213.

Yuliang Yan, Haochun Tang, Shuo Yan, and Enyan Dai. DuFFin: A Dual-Level Fingerprinting Framework for LLMs IP Protection. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 5168–5184, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-386-9. doi: 10.18653/v1/2026.findings-eacl.273.

Peigen Ye, Huali Ren, Zhengdao Li, Anli Yan, Hongyang Yan, Shaowei Wang, and Jin Li. Securing Large Language Models: A Survey of Watermarking and Fingerprinting Techniques. *ACM Computing Surveys*, 58(7):1–35, May 2026. ISSN 0360-0300, 1557-7341. doi: 10.1145/3773028.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language Models are Super Mario: Absorbing Abilities from Homologous Models as a Free Lunch. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 57755–57775. PMLR, July 2024.

Runpeng Yu and Xinchao Wang. Neural Lineage. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4797–4807, June 2024. doi: 10.1109/CVPR52733.2024.00459.

Chuanchao Zang, Xiangtao Meng, Wenyu Chen, Tianshuo Cong, Zha Yaxing, Dong Qi, Zheng Li, and Shanqing Guo. ErrorTrace: A Black-Box Traceability Mechanism Based on Model Family Error Space. In *Advances in Neural Information Processing Systems*, volume 38, pp. 68550–68578. Curran Associates, Inc., 2025. doi: 10.52202/085713-2306.

Jie Zhang, Dongrui Liu, Chen Qian, Linfeng Zhang, Yong Liu, Yu Qiao, and Jing Shao. REEF: Representation encoding fingerprints for large language models. In *International Conference on Learning Representations*, 2025.

Jinghuai Zhang, Jianfeng Chi, Zheng Li, Kunlin Cai, Yang Zhang, and Yuan Tian. BadMerging: Backdoor attacks against model merging. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, CCS '24, pp. 4450–4464, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 979-8-4007-0636-3. doi: 10.1145/3658644.3690284.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *International Conference on Learning Representations*, 2024a.

Mingqian Zheng, Jiaxin Pei, Lajanugen Logeswaran, Moontae Lee, and David Jurgens. When "A Helpful Assistant" Is Not Really Helpful: Personas in System Prompts Do Not Improve Performances of Large Language Models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15126–15154, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.888.

Xiaoyuan Zhu, Yaowen Ye, Tianyi Alex Qiu, Hanlin Zhu, Sijun Tan, Ajraf Mannan, Jonathan Michala, Raluca Ada Popa, and Willie Neiswanger. Auditing black-box LLM APIs with a rank-based uniformity test. In *The Fourteenth International Conference on Learning Representations*, 2026.

A Multiple-Choice Instantiation of Induced Decision Regions

Let $\mathcal{Q}$ be a set of multiple-choice questions, where each question $q \in \mathcal{Q}$ contains C candidate options defining the finite decision space $\mathcal{D} = \{1, \ldots, C\}$. When decision scores are available, let $\tilde{h}_{M,d}(q)$ denote the score assigned by model M to option d, such as a next-token logit or normalised choice probability. The soft decision vector is

$$\tilde{h}_M(q) = \big(\tilde{h}_{M,1}(q), \ldots, \tilde{h}_{M,C}(q)\big) \in \mathbb{R}^C, \tag{8}$$

with the corresponding discrete decision given by

$$h_M(q) = \min \arg\max_{d \in \mathcal{D}} \tilde{h}_{M,d}(q), \tag{9}$$

where ties are resolved in favour of the lowest-indexed decision.

For the multiple-choice instantiation under text-only access, we extend the outcome space with an invalid outcome.

$$\mathcal{D}_\perp = \mathcal{D} \cup \{\perp\}, \tag{10}$$

where $\perp$ denotes a response from which no valid option can be extracted. The extracted discrete outcome is then

$$h_M(q) = g\big(q, o_M(q)\big) \in \mathcal{D}_\perp. \tag{11}$$

Thus, valid outcomes under both observation modes lie in the same canonical decision space $\mathcal{D}$.

For each decision $d \in \mathcal{D}$, the corresponding induced decision region is

$$\mathcal{R}_d^M = \{q \in \mathcal{Q} \mid h_M(q) = d\}. \tag{12}$$

The collection $\{\mathcal{R}_d^M\}_{d \in \mathcal{D}}$ partitions the valid-decision subset $\{q \in \mathcal{Q} \mid g(q, o_M(q)) \in \mathcal{D}\}$ of the multiple-choice question space, defining the induced decision geometry of M. Responses mapped to $\perp$ lie outside this partition and are excluded from fingerprint construction.

For a probe set $\mathcal{P} \subseteq \mathcal{Q}$, induced decision region inheritance can be assessed at different resolutions. When soft decision vectors are available for both models, as in the analysis in Section 3.2, preservation is assessed through soft similarity between the corresponding soft decision vectors over $\mathcal{P}$,

$$\widehat{\mathrm{Sim}}_{\mathcal{P}}(S,T) = \frac{1}{|\mathcal{P}|} \sum_{q \in \mathcal{P}} \mathrm{sim}\left(\tilde{h}_S(q), \tilde{h}_T(q)\right). \tag{13}$$

When such fine-grained information is unavailable, as in the black-box verification setting described in Section 4.3, preservation is instead assessed through hard alignment of discrete decisions over $\mathcal{P}$,

$$\widehat{\mathcal{A}}_{\mathcal{P}}(S,T) = \frac{1}{|\mathcal{P}|} \sum_{q \in \mathcal{P}} \mathbf{1}\left\{h_T(q) = h_S(q) \neq \perp\right\}. \tag{14}$$

Thus, invalid outcomes are treated as non-aligned observations. When answer-option permutations are used, extracted scores and decisions are mapped back to the underlying answer options before aggregation or comparison, as specified in the corresponding sections.

B Analysis of Induced Decision Region Inheritance

B.1 Interpreting Informativeness

Informativeness, defined in Equation 4, measures the average preservation gap between related and unrelated models across the probes in a set, and therefore serves as a useful auxiliary metric. However, informativeness should not be interpreted as a sufficient criterion or the sole guideline for fingerprint construction.

First, informativeness summarises the average discriminative contribution of probes in a set, but does not capture their interactions when combined into a fingerprint. Two probe sets can therefore have similarly high informativeness while producing very different provenance separation. For example, a set of individually discriminative probes may still provide largely redundant evidence if they exhibit similar preservation patterns across unrelated models. In contrast, a probe set with more complementary preservation patterns can achieve stronger separation, even when its probes are individually less discriminative and its overall informativeness is lower. Thus, informativeness alone does not account for whether the resulting probe set provides complementary evidence.

Second, informativeness is based on average preservation differences between related and unrelated models and therefore does not characterise the relevant tail of the unrelated population. A small subset of unrelated models may strongly preserve the source decisions even when average unrelated preservation is low. Such hard negatives contribute little to the mean when they constitute only

a small fraction of the unrelated population, but can dominate provenance separation at stringent false-positive rates, where the verification threshold is determined by the most confusable unrelated models. Consequently, high informativeness does not necessarily imply strong provenance separation in the low-FPR regime.

There is also a practical limitation. Computing Equation 4 requires provenance-labelled related and unrelated models for the source, whereas fingerprint construction should not assume access to such labelled model panels.

Taken together, these limitations mean that informativeness should not be treated as a complete criterion for fingerprint construction. We therefore emphasise its role as an auxiliary metric for analysing inheritance patterns and the relative discriminative properties of candidate probe sets, rather than as a directly available or sufficient objective for fingerprint construction.

B.2 Inheritance Analysis Protocol

We use all MMLU test questions with Qwen2.5-7B as the source model and analyse three transformation-matched comparison pairs, each consisting of one related model and one unrelated model. The models and transformation types are summarised in Table 3.

Table 3: Models used in the induced decision region inheritance analysis.

Transformation	Related model	Unrelated model
Instruction tuning	Qwen2.5-7B-Instruct	Qwen3-1.7B
Quantisation	Qwen2.5-7B-Instruct-GPTQ-Int8	Qwen3-1.7B-GPTQ-Int8
Distillation	Qwen2.5-7B-Open-R1-Distill	Regulus-Qwen3-R1-Llama-Distill-1.7B

For each evaluated question, we obtain from each model the next-token logits over the four answer options and normalise them into a probability vector. For this analysis, we instantiate the soft similarity in Equation 3 as

$$\mathrm{sim}\Big(\tilde{h}_S(q), \tilde{h}_T(q)\Big) = 1 - \mathrm{TV}\Big(\tilde{h}_S(q), \tilde{h}_T(q)\Big) = 1 - \frac{1}{2}\left\|\tilde{h}_S(q) - \tilde{h}_T(q)\right\|_1, \tag{15}$$

where TV denotes total variation distance. Hard alignment follows the discrete decision agreement defined in Equation 3.

For Figure 1(a), we compute the per-question soft similarity between the source and each of the six comparison models under the original answer ordering, and construct a model-specific complementary cumulative distribution function (CCDF) over the evaluated questions. The six thin curves show these model-specific CCDFs. For each group, the thick curve is the pointwise arithmetic mean of the three corresponding model-specific CCDFs at each soft-similarity threshold. For the annotated hard alignment, we first compute, for each comparison model, the fraction of evaluated questions on which its discrete decision agrees with the source, and then average these model-level alignment rates across the three models in the corresponding group.

For Figure 1(b), we compute each question's informativeness using Equation 4 separately for soft similarity and hard alignment. Informativeness is the difference between the mean per-question preservation score across the three related models and that across the three unrelated models. Because each model's hard-alignment score is binary, hard informativeness lies in

$$\left\{-1, -\frac{2}{3}, -\frac{1}{3}, 0, \frac{1}{3}, \frac{2}{3}, 1\right\}.$$

To enable direct visual comparison with hard informativeness, continuous soft informativeness is binned using the midpoint boundaries

$$\left[-1, -\frac{5}{6}, -\frac{1}{2}, -\frac{1}{6}, \frac{1}{6}, \frac{1}{2}, \frac{5}{6}, 1\right]$$

and plotted at the corresponding hard-alignment levels. Figure 1(b) reports the fraction of evaluated questions assigned to each bin.

For Figure 1(c), we evaluate whether question informativeness estimated from one model pair transfers to other pairs using three folds. In each fold, one transformation-matched comparison pair is used to compute each question's soft-similarity informativeness. We rank all questions accordingly and partition them into five approximately equal-sized groups Q1–Q5 from lowest to highest informativeness. The remaining two pairs do not contribute to question ranking or grouping and are held out for evaluation. For example, when the instruction-tuning pair is used to rank and group the questions, the quantisation and distillation pairs are held out.

For each question group and held-out pair, we compute the related–unrelated difference in mean per-question preservation separately for soft similarity and hard alignment. We then average the gaps from the two held-out pairs to obtain the fold-level result. The pair used for question ranking rotates across instruction tuning, quantisation, and distillation, yielding three folds. The plotted lines show the mean across the three folds, while the shaded regions show the minimum–maximum range.

B.3 Decision Region Visualisation

To complement the aggregate analysis in Appendix B.2, we provide a visualisation of how individual questions are positioned within the source model's induced decision geometry and how these positions change across related and unrelated comparison models. We compare randomly sampled probes with Stemma-selected probes for a related and an unrelated model. The projection is used only to provide geometric intuition and does not enter fingerprint construction or provenance scoring.

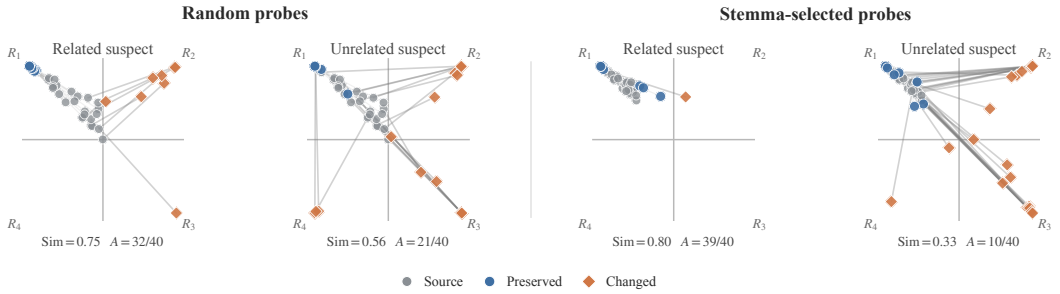

Figure 4: Decision region visualisation for 40 randomly sampled probes (left pair) and 40 Stemma-selected probes (right pair). The source is Qwen2.5-7B, with Qwen2.5-7B-Instruct and Qwen3-1.7B as the related and unrelated comparison models, respectively. Grey markers denote source decisions, while blue and orange markers denote comparison-model decisions that preserve or change the corresponding source region assignment. Lines connect paired source and comparison decisions. Annotations report soft similarity (Sim) and hard alignment (A).

Decision vectors. Unlike the aggregate analysis in Appendix B.2, which uses the original answer ordering, this visualisation uses cyclic option permutations to obtain more stable soft decision vectors. For each displayed question, we cyclically permute the four answer options and treat each permutation as a separate query. For each query, we obtain the next-token log-probabilities of the four displayed choice labels and map them back to the canonical answer options. We average these log-probabilities across permutations and apply a softmax over the four options to obtain a four-dimensional soft decision vector for each model and question. This reduces sensitivity to option order and choice-label bias.

Source-relative projection. For each question, we rank the answer options by the source model's aggregated option probabilities. This defines a source-relative ordering of the four options. In the four-choice MMLU setting, we denote the corresponding visualisation quadrants by $R_1, \ldots, R_4$, where R_1 corresponds to the source model's most preferred option, R_2 to its second most preferred option, and so on.

The same source-defined ordering is applied to every model M. For each question q, the resulting soft decision vector is

$$\tilde{h}_M(q) = \big(\tilde{h}_{M,1}(q), \tilde{h}_{M,2}(q), \tilde{h}_{M,3}(q), \tilde{h}_{M,4}(q)\big),$$

where the entries are the aggregated probabilities assigned by M to the source-ranked options $R_1, \ldots, R_4$.

To project this four-dimensional vector into two dimensions, we represent the four source-relative decision regions as the quadrants of a square, with $R_1, \ldots, R_4$ occupying the upper-left, upper-right, lower-right, and lower-left quadrants, respectively. The projected coordinates are defined as

$$x_M(q) = \max\{\tilde{h}_{M,2}(q), \tilde{h}_{M,3}(q)\} - \max\{\tilde{h}_{M,1}(q), \tilde{h}_{M,4}(q)\},$$
$$y_M(q) = \max\{\tilde{h}_{M,1}(q), \tilde{h}_{M,2}(q)\} - \max\{\tilde{h}_{M,3}(q), \tilde{h}_{M,4}(q)\}.$$

The x-coordinate compares the strongest preference among the right-side regions with that among the left-side regions, while the y-coordinate compares the strongest preference among the upper-side regions with that among the lower-side regions. Both coordinates lie in $[-1, 1]$, since each is a difference between probabilities.

Displayed decisions. For each model–question pair, let

$$z_M(q) = (x_M(q), y_M(q))$$

denote its displayed two-dimensional coordinate. The figure plots $z_S(q)$ for the source model S as a grey point and $z_T(q)$ for the suspect model T as a coloured point. The connecting line represents the projected displacement $z_T(q) - z_S(q)$ and is used only as a visual aid, with shorter lines indicating smaller shifts in the projected space.

Under the source-relative ordering, we define the displayed region index as

$$\rho_M(q) = \min \operatorname*{arg\,max}_{i \in \{1,2,3,4\}} \tilde{h}_{M,i}(q),$$

where ties are resolved in favour of the lowest index. Thus, $\rho_M(q) = i$ means that model M selects the option ranked i-th by the source, and the corresponding point lies in the source-relative quadrant R_i. By construction, $\rho_S(q) = 1$ for every question. Comparison-model points are coloured blue when $\rho_T(q) = 1$, indicating preservation of the source decision, and orange otherwise. Because the same source-relative ordering is applied to both models, $\rho_T(q) = 1$ is equivalent to $h_T(q) = h_S(q)$.

Annotation metrics. Let $\mathcal{P}$ denote the 40 questions displayed in each panel. The annotations Sim and A correspond to $\widehat{\mathrm{Sim}}_{\mathcal{P}}(S, T)$ and $\widehat{\mathcal{A}}_{\mathcal{P}}(S, T)$ in Equation 3, respectively. Soft similarity is instantiated using Equation 15 and applied to the aggregated decision vectors constructed above.

Interpretation. For the randomly sampled probes, the related model exhibits stronger preservation of the source model's induced decision geometry than the unrelated model. Its soft similarity to the source is 0.75, compared with 0.56 for the unrelated model, while the corresponding hard alignments are $32/40$ and $21/40$. The random probes therefore already reveal inheritance in both soft decision preferences and discrete region assignments.

Because induced decision geometry cannot be observed exhaustively, any finite probe set captures only a partial view of this inheritance, with its discriminative strength depending on which parts of the decision geometry are sampled. Stemma-selected probes yield substantially clearer separation, with soft similarity of 0.80 versus 0.33 and hard alignment of $39/40$ versus $10/40$. Stemma therefore does not create the underlying provenance signal. Instead, it selects probes that make induced decision region inheritance more observable and discriminative.

C Stemma Implementation and Configuration Details

This appendix specifies the implementation and main experimental configuration of Stemma's three stages: decision interface calibration, fingerprint construction, and fingerprint verification. Unless otherwise stated, experiments are repeated with seeds 42–46, which control all random sampling and stochastic generation within each run.

Dataset. Probe candidates are sampled from the MMLU test split across all subjects (Hendrycks et al., 2021). Unless otherwise stated, we randomly sample 3,000 candidate questions for each source model. Each candidate is evaluated under cyclic permutations of its answer options to assess the stability of the source decision across different option orders and label assignments.

Prompt calibration. Before fingerprint construction or verification, Stemma independently calibrates a multiple-choice prompt template for each evaluated model under the relevant access mode using 100 examples from the MMLU validation split, which is disjoint from the candidate probe pool. Calibration selects a prompt format that reliably supports option-level decisions under the corresponding access mode, facilitating white-box choice-label scoring during fingerprint construction or black-box answer extraction during fingerprint verification. White-box source queries use raw completion-style prompts for direct and consistent next-token scoring, reflecting the setting in which the verifier has full control over the source model and its input formatting. In contrast, black-box calibration and verification use the configured raw or chat interface, reflecting the interface actually exposed by the suspect model in deployment.

We evaluate up to 6 simple prompt templates. To avoid severe label bias, we first restrict selection to templates for which, among valid extracted predictions, the most frequently predicted label accounts for at most 0.7 of the calibration predictions. If no template satisfies this criterion, selection is performed over all templates. We then select the template with the highest valid-choice extraction rate, breaking ties according to the predefined template order, and fix it for all subsequent queries to the model under that access mode. Under the default configuration, each calibration requires 600 queries in our implementation, based on 100 examples and six candidate templates. This budget is configurable and can in principle be reduced through a smaller calibration set, fewer templates, or more selective template search.

The default set of prompt templates used in calibration is shown below. In these templates, $\{\texttt{question}\}$ denotes the question text, $\{\texttt{options}\}$ denotes the formatted answer options, and $\{\texttt{choices}\}$ denotes the available option labels.

Simple answer cue

```
{question}
{options}
Answer:
```

MMLU-style prompt

```
Question: {question}
{options}
Answer:
```

Strict label-only prompt

```
Read the following multiple-choice question carefully.

Question: {question}
{options}

Choose the best answer. Output exactly one option label from {choices}. Do not output any
explanation.
Answer:
```

Single-character response

```
Question: {question}
{options}

Your response must be exactly one character from {choices}.
Response:
```

High-constraint prompt

```
Read the following multiple-choice question carefully. Choose the best answer from the
listed options.
Question: {question}
{options}
Answer with only one option label from {choices}.
Answer:
```

Answer cue with trailing newline

```
{question}
{options}
Answer:\n
```

Fingerprint construction. For each source model, Stemma computes next-token choice-label log-probabilities for every candidate question under all cyclic option permutations. Choice labels are scored using tokenizer-aware matching. When multiple token IDs correspond to the same visible option label, their log-probabilities are combined using log-sum-exp. Under each permutation, the highest-probability option is selected and mapped back to its canonical option identity. The canonical source decision is then determined by voting across the resulting permutation-level decisions. Candidate questions are filtered and ranked according to stability, robustness, and specificity.

During hard filtering, a candidate is retained only if the same canonical wrong option receives more than half of the source model's permutation votes (> 0.5), has a non-negative average wrong-label top margin (≥ 0), and is selected by fewer than half of the background model's permutation-level predictions (< 0.5). Unless otherwise stated, we use `microsoft/Phi-3.5-mini-instruct` as the background model for specificity evaluation.

During soft ranking, the surviving candidates are ranked using the following standardised score:

$$\mathrm{Score}(q) = 1.0 \cdot \mathrm{clip}\big(z(r(q)), -2, 2\big) - 0.5 \cdot z(b(q)), \tag{16}$$

where $r(q)$ is the average wrong-label log-probability margin across permutations, $b(q)$ is the fraction of background-model predictions that select the same canonical wrong option, and $z(\cdot)$ denotes standardisation over the surviving candidate pool for the corresponding source model. The standardised robustness term is clipped to $[-2, 2]$ to prevent extreme margins from dominating the ranking.

The resulting source-specific fingerprint comprises the 40 highest-ranked probes and their canonical wrong labels, as determined by permutation voting.

Configuration selection. The hard-filter thresholds are set according to their semantic interpretations, corresponding to majority consistency, a non-negative source-decision margin, and less-than-majority background agreement. Stability is enforced at the filtering stage rather than included as a separate soft-ranking term, since permutation consistency takes only a small number of discrete values and provides little additional discrimination among candidates after filtering. The ranking hyperparameters were selected on three held-out model lineages, guided by the intended balance between robustness and specificity. These development lineages, listed in Table 4, are not included in the provenance benchmark. The resulting configuration was fixed across all benchmark evaluations.

Table 4: Models used for configuration selection.

Provenance Group	Repository ID
Qwen2.5-3B	Qwen/Qwen2.5-3B Qwen/Qwen2.5-3B-Instruct Pew404/Qwen2.5-3B-Instruct-abliterated
Llama-2-7B	meta-llama/Llama-2-7b-hf meta-llama/Llama-2-7b-chat-hf TheBloke/Llama-2-7B-Chat-GPTQ
Gemma-2-2B	google/gemma-2-2b google/gemma-2-2b-it anakin87/gemma-2-2b-neogenesis-ita

Fingerprint verification. For each source–suspect pair, the suspect model is queried on the corresponding source probes under the same cyclic option permutations used during fingerprint construction. Generation is limited to 16 new tokens. A rule-based regular-expression extractor maps each response to either a predicted option label or an invalid outcome $\perp$, with valid labels mapped back to their canonical option identities.

The fingerprint alignment score is computed as defined in Equation 7, with invalid outcomes treated as non-aligned. With 40 probes and four cyclic permutations, the main configuration yields 160 probe–permutation observations for each model pair.

D Stemma Probe Selection and Informativeness

Although Stemma does not use the informativeness defined in Section 3.2 as a direct construction objective because it requires provenance-labelled model pairs and does not fully characterise fingerprint performance, it still captures one aspect of a probe set's discriminative power. We therefore use it as a post-hoc diagnostic to examine whether probe sets selected by Stemma using the complementary criteria of stability, robustness, and specificity also exhibit elevated informativeness.

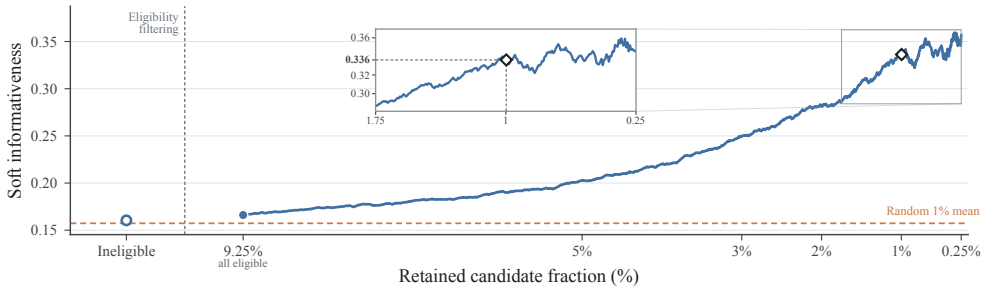

Figure 5: Soft informativeness computed from permutation-averaged decision vectors (Equation 4) as Stemma restricts the MMLU pool. Eligibility filtering retains 9.25% of candidates before Stemma ranking. The diamond marks the 1% retained fraction ($K = 140$). The dashed line shows the mean of 100 size-matched random subsets sampled without replacement from the full pool.

We use the same Qwen2.5-7B source and three transformation-matched related–unrelated model pairs as in Section 3.2, evaluated over all MMLU test questions. We then apply the standard Stemma probe selection pipeline to this full candidate pool using only the information available during fingerprint construction. Consistent with Stemma's decision interface, each question is evaluated under cyclic option permutations, with permutation-level outputs mapped back to the canonical options. Accordingly, for post-hoc evaluation, we compute soft informativeness in Equation 4 using canonical soft decision vectors averaged across permutations, rather than the original-order vectors used in Section 3.2. Eligibility filtering retains 9.25% of the pool, after which the remaining probes are ranked by the Stemma selection score and progressively smaller top-ranked fractions are retained. To quantify the informativeness enrichment over random selection, we also compare the probes retained at the 1% fraction ($K = 140$) against 100 equally sized random subsets sampled uniformly without replacement from the full candidate pool.

Figure 5 shows that the informativeness of the retained probe sets increases as Stemma progressively narrows the candidate pool. The ineligible set has a soft informativeness of 0.160, while the eligible set reaches 0.166, indicating a modest enrichment from eligibility filtering. Informativeness then increases substantially for progressively smaller top-ranked subsets. At the 1% retained fraction, the selected probe set achieves a soft informativeness of 0.336, compared with a mean of 0.157 across equally sized random subsets, which is $2.14\times$ the random baseline. These results show that Stemma's selection procedure, guided by three complementary criteria, enriches the retained probe set with stronger provenance-discriminative signal without directly computing informativeness.

E Model Provenance Benchmark

Table 5 presents the complete model benchmark used in our experiments. It comprises 56 publicly available checkpoints from the Hugging Face Hub (Hugging Face, 2026), organised into 7 provenance groups across 5 model families, with model sizes ranging from 1.7B to 14B parameters. The benchmark covers both fine-grained provenance relationships among closely related models and diverse model-development transformations. Model selection considers provenance clarity, documentation quality, download popularity, and diversity across model families, scales, and transformation

types. Three of the groups are Qwen-based, providing hard negatives with similar architectures and training recipes but distinct provenance.

Table 5: Complete model benchmark.

Provenance group	Repository ID	Model type	Query interface
Qwen-2.5-7B	Qwen/Qwen2.5-7B	Pretrained	raw
	Qwen/Qwen2.5-7B-Instruct	Instruct	chat
	Qwen/Qwen2.5-Coder-7B-Instruct	Fine-tune	chat
	zjudai/flowertune-medical-lora-qwen2.5-7b-instruct	Adapter	chat
	SeeFlock/task-12-Qwen-Qwen2.5-7B-Instruct	Adapter	chat
	Locutusque/StockQwen-2.5-7B	Merge	chat
	Qwen/Qwen2.5-7B-Instruct-GPTQ-Int8	Quantisation	chat
	Lansechen/Qwen2.5-7B-Open-R1-Distill	Distillation	chat
Qwen-2.5-14B	Qwen/Qwen2.5-14B	Pretrained	raw
	Qwen/Qwen2.5-14B-Instruct	Instruct	chat
	oxyapi/oxy-1-small	Fine-tune	chat
	Qwen/Qwen2.5-14B-Instruct-1M	Fine-tune	chat
	ToastyPigeon/qwen-story-test-qlora	Adapter	chat
	v000000/Qwen2.5-14B-Gutenberg-Instruct-Slerpeno	Merge	chat
	Qwen/Qwen2.5-14B-Instruct-GPTQ-Int8	Quantisation	chat
	alibaba-pai/DistilQwen2.5-DS3-0324-14B	Distillation	chat
Qwen3-1.7B	Qwen/Qwen3-1.7B-Base	Pretrained	raw
	Qwen/Qwen3-1.7B	Instruct	chat
	mlabonne/Qwen3-1.7B-abliterated	Fine-tune	chat
	HuggingFaceTB/qwen3-1.7b-gsm8k-sft	Fine-tune	chat
	txmedai/ClinicalEase-Qwen3-1.7B	Adapter	chat
	kurakurai/Luth-1.7B-Instruct	Merge	chat
	Qwen/Qwen3-1.7B-GPTQ-Int8	Quantisation	chat
	prithivMLmods/Regulus-Qwen3-R1-Llama-Distill-1.7B	Distillation	chat
Llama-3.1-8B	meta-llama/Llama-3.1-8B	Pretrained	raw
	meta-llama/Llama-3.1-8B-Instruct	Instruct	chat
	RedHatAI/Llama-3.1-8B-tldr	Fine-tune	raw
	chchen/Llama-3.1-8B-Instruct-PsyCourse-fold7	Adapter	chat
	zjudai/flowertune-medical-lora-llama-3.1-8b-instruct	Adapter	chat
	Xiaojian9992024/Llama3.1-8B-ExtraMix	Merge	chat
	hugging-quants/Meta-Llama-3.1-8B-Instruct-GPTQ-INT4	Quantisation	chat
	arcee-ai/Llama-3.1-SuperNova-Lite	Distillation	chat
Mistral-7B-v0.3	mistralai/Mistral-7B-v0.3	Pretrained	raw
	mistralai/Mistral-7B-Instruct-v0.3	Instruct	chat
	KurmaAI/AQUA-7B	Fine-tune	chat
	chaymaemerhrioui/mistral-Brain_Model_ACC_Trainer	Adapter	chat
	zjudai/flowertune-medical-lora-mistral-7b-instruct-v0.3	Adapter	chat
	grimjim/Mistral-7B-Instruct-demi-merge-v0.3-7B	Merge	chat
	RedHatAI/Mistral-7B-Instruct-v0.3-GPTQ-4bit	Quantisation	chat
	eganwo/mistral7b-distilled-from-deepseek-r1-qwen32b	Distillation	chat
Falcon3-7B	tiiuae/Falcon3-7B-Base	Pretrained	raw
	tiiuae/Falcon3-7B-Instruct	Instruct	chat
	ehristoforu/falcon3-ultraset	Fine-tune	chat
	jahyungu/Falcon3-7B-Instruct-v1-Easy	Adapter	chat
	jahyungu/Falcon3-7B-Instruct-v1-Hard	Adapter	chat
	suayptalha/Falcon3-Jessi-v0.4-7B-Slerp	Merge	chat
	tiiuae/Falcon3-7B-Instruct-GPTQ-Int8	Quantisation	chat
	RedaAlami/Falcon3-7B-Instruct-Distill-DS-v1	Distillation	chat
OLMo-2-7B	allenai/OLMo-2-1124-7B	Pretrained	raw
	allenai/OLMo-2-1124-7B-Instruct	Instruct	chat
	allenai/OLMo-2-1124-7B-SFT	Fine-tune	chat
	allenai/OLMo-2-1124-7B-Instruct-preview	Fine-tune	chat
	jahyungu/OLMo-2-1124-7B-Instruct-Humanities	Adapter	chat
	jahyungu/OLMo-2-1124-7B-Instruct-Social-Sciences	Adapter	chat
	Alelcv27/Olmo2-7B-Breadcrumbs-v1	Merge	chat
	kaitchup/OLMo-2-1124-7B-Instruct-AutoRound-GPTQ-4bit	Quantisation	chat

Benchmark composition. Each provenance group follows a consistent structure, containing 8 checkpoints: 1 pretrained checkpoint, 1 instruction-tuned checkpoint, and 6 additional variants. For these additional variants, we aim to cover a diverse set of common model-development operations. Specifically, where suitable public checkpoints are available, each group includes 3 adapted variants covering full fine-tuning, parameter-efficient adaptation, and related model modifications, together with 1 model-merge variant, 1 quantised variant, and 1 distillation variant. The only exception is the OLMo-2-1124-7B group, for which we did not find a suitable public distillation variant, so we include an additional fine-tuned or adapter-based variant instead. This design gives each provenance group comparable coverage while capturing the diversity and complexity of real-world model development, thereby providing a more rigorous testbed for evaluating fingerprinting methods under realistic transformation scenarios. For evaluation, the pretrained and instruction-tuned checkpoints in each group serve as source models, giving 14 sources in total. Each source is evaluated against every other checkpoint. Pairs from the same provenance group are labelled positive, whereas pairs from different groups are labelled negative. Positive pairs may include direct or indirect derivation as well as checkpoints descending from a shared ancestor. The benchmark tests lineage membership rather than derivation direction. This yields 770 ordered pairs between sources and suspects, of which 98 are positive and 672 negative.

Model types. The model type column reports the development operation associated with each checkpoint. *Pretrained* denotes the base model trained before instruction tuning. *Instruct* denotes an instruction-tuned checkpoint derived from the pretrained model. *Fine-tune* is used as a coarse category for checkpoints obtained through task-specific, domain-specific, or instruction-following post-training and closely related model adaptations. *Adapter* denotes parameter-efficient adaptation, where additional or low-rank trainable parameters are used to specialise the model. *Merge* denotes a checkpoint produced by combining weights from multiple models. *Quantisation* denotes a numerically compressed checkpoint with reduced weight precision. *Distillation* denotes a checkpoint derived from a base model and further trained or fine-tuned using teacher-generated data, such as model-generated responses or reasoning traces from a stronger model.

Query interfaces. For black-box querying, we also distinguish two query interfaces, following the recommended usage in each checkpoint's model card or tokenizer configuration. The *raw* interface denotes completion-style prompting, where the prompt is passed directly to the causal language model without a chat wrapper. It is used for pretrained checkpoints and for downstream checkpoints that are released or documented as completion-style models. The *chat* interface denotes chat-template prompting, where the calibrated prompt is first treated as a user message and then rendered using the tokenizer's chat template before generation. It is used for instruction-tuned and chat-oriented checkpoints. Unless otherwise stated, responses are generated using stochastic decoding with temperature 0.7, top-p 0.9, and top-k 50.

Chat templates may add model-specific system, user, and assistant formatting, thereby changing the effective input received by the model. For example, when no explicit system message is provided, the Qwen2.5-Instruct tokenizer renders a single user message as follows:

Rendered input under the Qwen2.5-Instruct chat template

```
<|im_start|>system
You are Qwen, created by Alibaba Cloud. You are a helpful assistant.<|im_end|>
<|im_start|>user
{user_prompt}<|im_end|>
<|im_start|>assistant
```

Here, {user_prompt} denotes the input prompt used for evaluation. We do not explicitly provide a manually constructed system message. Instead, when a checkpoint's tokenizer chat template defines a default system message, it is inserted automatically during template rendering. Using the interface specified by each checkpoint's model card or tokenizer configuration helps ensure that the model is evaluated under its intended usage pattern, better matching common usage and realistic deployment practice.

F Evaluation Metrics

Each evaluated source–suspect pair is assigned a continuous provenance score, where higher values indicate stronger evidence of a provenance relationship. Related pairs are treated as positive and unrelated pairs as negative.

AUC and partial AUC. The area under the receiver operating characteristic curve is defined as

$$\mathrm{AUC} = \int_0^1 \mathrm{TPR}(u)\,\mathrm{d}u,$$

where u denotes the false-positive rate. To emphasise performance in the low-FPR regime, we additionally report standardised partial AUC over $\mathrm{FPR} \in [0, \alpha]$, with $\alpha = 0.05$. Let

$$A_\alpha = \int_0^\alpha \mathrm{TPR}(u)\,\mathrm{d}u$$

denote the unstandardised partial area. We apply McClish standardisation (McClish, 1989):

$$\mathrm{pAUC}_\alpha = \frac{1}{2}\left(1 + \frac{A_\alpha - \alpha^2/2}{\alpha - \alpha^2/2}\right).$$

Under this standardisation, random ranking corresponds to 0.5 and perfect ranking to 1.

Low-FPR true-positive rate. TPR at 1% FPR is the highest empirical true-positive rate attained without exceeding a false-positive rate of 0.01. We do not interpolate between operating points.

Discriminability. We quantify the standardised separation between the positive and negative score distributions using

$$d' = \frac{\mu_+ - \mu_-}{\sqrt{(\sigma_+^2 + \sigma_-^2)/2}},$$

where μ_+ and μ_- are the mean positive and negative scores, and σ_+^2 and σ_-^2 are their sample variances. Positive values indicate that related pairs receive higher scores on average, while larger values indicate clearer separation in the intended direction. Negative values indicate reversed mean ordering.

Multi-run aggregation. Within each run, metrics for each reported source subset are computed by pooling all source–suspect pair scores and labels in that subset. We repeat the procedure for five runs using seeds 42–46 and report the arithmetic mean and sample standard deviation. Reported values are rounded to three decimal places after aggregation.

G Baseline Implementations

We implement four black-box LLM fingerprinting baselines to compare against Stemma. Unless otherwise stated below, we use the settings and hyperparameters specified in the original papers or official repositories for each baseline, in order to reproduce its intended performance. Since a fine-grained comparison requires a continuous provenance score rather than only a binary decision for each source and suspect model pair, we use the underlying fingerprint similarity or matching score produced by each baseline as its pairwise provenance score. This is natural for these methods, as their final decisions are based on fingerprint similarity even when their decision rules differ.

LLMmap. LLMmap (Pasquini et al., 2025) fingerprints a model by encoding its behaviour on a fixed set of diagnostic prompts into an open-set feature space. We use the 8 input queries from the original paper and the pretrained open-set feature extractor released with the official implementation. To construct each source gallery representation, we evaluate 100 prompt configurations over these queries and average the resulting feature vectors. During verification, the suspect model is evaluated once on the same 8 queries and its feature vector is compared with the source gallery representation. For both gallery construction and verification, generated responses contain at most 100 new tokens

and are truncated to 650 characters before feature extraction. Following the original comparison procedure, we compute the Euclidean distance d between the source and suspect fingerprints and convert it into a similarity score using the inverse-one-plus mapping, $s = 1/(1 + d)$. This score is used as the pairwise provenance score for each evaluated source and suspect model pair.

LLMPrint. LLMPrint (Hu et al., 2026) constructs black-box fingerprints by learning adversarial suffix probes that induce model-specific preferences over target word pairs. Its original configuration optimises suffixes for 300 word pairs using 1,000 GCG steps per pair. Based on pilot runtime measurements, running the full configuration independently across all three evaluation settings is estimated to require over 240 GPU-days in our environment. We therefore use a reduced configuration consistently across all source models.

For each source model, we sample 100 word pairs from the fixed category vocabulary provided by the official repository, such that both words are represented by a single token under the source tokenizer. We then optimise one adversarial suffix per pair for 200 GCG steps under the fixed LLMPrint prompt template. The learned suffixes are used to query both source and suspect models. Following the original scoring procedure, we perform 100 repeated one-token generations for each word pair and record how often each target token is generated. The more frequently generated token determines the model's binary preference for that pair, and the agreement rate between the source and suspect preferences is used as the pairwise provenance score.

Model Provenance Testing. Model Provenance Testing (MPT) (Nikolic et al., 2025) detects provenance by comparing next-token continuation behaviour between a source model and a suspect model. Following the official implementation, we sample 5,000 sentence-level prompts from the MPT prompt pool provided in the official repository. For each source model, we construct a continuation cache by querying the model on all prompts with single-token generation, using `max_new_tokens`=1. Each suspect model is then evaluated on the same prompt set, and its next-token continuations are matched against the cached source continuations. We use the resulting hit rate, namely the fraction of matching next-token continuations, as the continuous pairwise provenance score.

ZeroPrint. ZeroPrint (Shao et al., 2026) constructs black-box LLM fingerprints by measuring the response sensitivity of a model to query perturbations. We follow the original setup by using HumanEval (Chen et al., 2021) as the query source, converting each task prompt into a completion-style query, and truncating the query to 20 words. Following the configuration reported in the original ZeroPrint paper, we sample 2 base queries and generate 4 word-substitution variants for each query, where each perturbation replaces 3 words with independently selected top-10 nearest neighbours under GloVe embeddings (Pennington et al., 2014). Responses are generated with at most 128 new tokens and are not truncated after generation. To reduce generation noise, each model is queried 20 times per prompt, and the resulting responses are embedded with `sentence-transformers/all-mpnet-base-v2` (Reimers & Gurevych, 2019), matching the official implementation. ZeroPrint then estimates a response-sensitivity fingerprint for each base query from the input and output embedding changes using the Jacobian-based estimator with ridge regularisation $\alpha = 0.001$. The resulting fingerprints are aggregated by mean pooling, and the Pearson correlation coefficient between source and suspect fingerprints is used as the pairwise provenance score.

Implementation consistency. For all baselines, we use the same benchmark construction, model query interfaces, source-model set, and suspect-model set. Each baseline produces a pairwise provenance score for every evaluated source–suspect pair. We compute and report the final results from these pairwise scores using the same evaluation metrics as Stemma.

H Additional Analysis on the Main Benchmark

We further analyse Stemma's performance across subgroups of the main benchmark to characterise variation in provenance-testing difficulty.

Performance by source provenance group. We first examine Stemma's performance separately for each source provenance group. For each group, we pool all source–suspect pairs whose source belongs to that group, including one pretrained and one instruction-tuned source model. Each group therefore contains 110 pairs, comprising 14 positive and 96 negative pairs. Because one false positive among 96 negatives already exceeds 1% FPR, the reported TPR at 1% FPR corresponds to the operating point with zero false positives for this analysis, and small differences should therefore be interpreted with caution.

Table 6: Stemma performance by source provenance group on the main benchmark.

Source provenance group	AUC ↑	pAUC ↑	TPR@1% FPR ↑
Qwen-2.5-7B	0.855 ± 0.012	0.883 ± 0.020	0.743 ± 0.039
Qwen-2.5-14B	0.966 ± 0.018	0.931 ± 0.014	0.843 ± 0.032
Qwen3-1.7B	1.000 ± 0.001	0.997 ± 0.007	0.986 ± 0.032
Llama-3.1-8B	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Mistral-7B-v0.3	0.993 ± 0.009	0.957 ± 0.046	0.900 ± 0.108
Falcon3-7B	1.000 ± 0.000	0.999 ± 0.003	0.986 ± 0.032
OLMo-2-7B	0.941 ± 0.024	0.903 ± 0.018	0.800 ± 0.032
Full benchmark (pooled)	0.967 ± 0.005	0.944 ± 0.002	0.871 ± 0.020

Overall, Stemma maintains strong performance across source provenance groups. Although performance varies across groups, most achieve AUC and pAUC close to or above the pooled result. In particular, we analyse the three Qwen-based provenance groups, which have similar architectures and training recipes and form a challenging subset of the benchmark. Across these groups, Stemma achieves macro-average AUC and pAUC of 0.940 and 0.937, respectively, compared with 0.983 and 0.965 across the remaining four groups, showing only a modest reduction while retaining strong discrimination. Moreover, under the stringent zero-false-positive operating point induced by the subgroup size, the mean TPR across all seven groups remains 0.894, indicating that high recall is preserved under a strict false-positive constraint.

Performance across suspect adaptation types. We next complement the source-side analysis by examining performance across different forms of model adaptation. For each type, we pool all source–suspect pairs whose suspect belongs to the corresponding category, while retaining sources across provenance groups. We consider instruction tuning, fine-tuning, adapter tuning, quantisation, model merging, and distillation. Because the number of suspects differs across types, the resulting subsets vary in size.

Table 7: Stemma performance across suspect adaptation types on the main benchmark.

Suspect adaptation type	AUC ↑	pAUC ↑	TPR@1% FPR ↑
Instruction tuning	0.999 ± 0.002	0.984 ± 0.022	0.943 ± 0.078
Fine-tuning	0.996 ± 0.001	0.971 ± 0.009	0.920 ± 0.027
Adapter tuning	0.999 ± 0.001	0.989 ± 0.011	0.967 ± 0.035
Quantisation	0.999 ± 0.003	0.991 ± 0.016	0.957 ± 0.039
Model merging	0.854 ± 0.022	0.848 ± 0.007	0.657 ± 0.060
Distillation	0.956 ± 0.023	0.895 ± 0.022	0.750 ± 0.059
Full benchmark (pooled)	0.967 ± 0.005	0.944 ± 0.002	0.871 ± 0.020

Stemma performs strongly across most adaptation types. Instruction tuning, fine-tuning, adapter tuning, and quantisation all achieve stronger AUC and low-FPR performance than the overall benchmark. Distillation remains well separated overall, but is weaker at low FPR, while model merging is the most challenging category in the benchmark. This pattern is consistent with the more substantial modifications to the underlying model introduced by distillation and merging. Overall, these results show that provenance evidence remains strong across a diverse range of common model adaptations.

I ALL-RAW QUERY EVALUATION

In this control setting, raw completion-style prompts are used during both fingerprint construction and verification, without applying tokenizer chat templates to black-box queries. By removing model-specific chat formatting, this setting controls for interface heterogeneity in the main experiments and evaluates all methods under a uniform raw format. However, 48 of the 56 benchmark checkpoints use chat interfaces under their recommended configurations, making raw prompting non-native for most models and the all-raw setting less representative of realistic deployment while providing more favourable conditions for the baselines. In addition to reporting detailed performance under this setting, we further examine the sensitivity of the aggregate results and analyse performance by source provenance group and suspect adaptation type.

Overall performance. Table 8 reports the full results for the all-raw query-interface setting, with the main findings discussed in Section 5.2.

Table 8: Performance under the all-raw interface. Best/second-best are bold/underlined.

Method	Pretrained sources				Instruct sources				All sources			
	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑
LLMmap	0.573 ±0.018	0.508 ±0.010	0.020 ±0.035	0.237 ±0.069	0.744 ±0.016	0.567 ±0.011	0.102 ±0.038	0.889 ±0.077	0.658 ±0.016	0.524 ±0.005	0.045 ±0.021	0.549 ±0.055
LLMPrint	0.605 ±0.052	0.505 ±0.012	0.004 ±0.009	0.409 ±0.186	0.761 ±0.024	0.582 ±0.025	0.118 ±0.053	1.044 ±0.113	0.675 ±0.030	0.545 ±0.008	0.061 ±0.025	0.647 ±0.121
MPT	**0.991** ±0.001	0.949 ±0.004	0.886 ±0.011	**3.280** ±0.029	**0.994** ±0.002	**0.973** ±0.003	**0.927** ±0.018	**3.338** ±0.042	**0.989** ±0.001	0.946 ±0.008	0.849 ±0.051	2.719 ±0.027
ZeroPrint	0.728 ±0.012	0.543 ±0.020	0.041 ±0.014	0.832 ±0.033	0.859 ±0.007	0.710 ±0.005	0.384 ±0.017	1.571 ±0.042	0.797 ±0.009	0.616 ±0.014	0.186 ±0.033	1.179 ±0.029
Stemma	0.965 ±0.006	**0.960** ±0.006	**0.914** ±0.017	3.155 ±0.181	0.969 ±0.005	0.945 ±0.002	0.878 ±0.014	3.074 ±0.089	0.967 ±0.003	**0.951** ±0.001	**0.894** ±0.009	**3.017** ±0.043

Sensitivity analysis. The overall AUC gap is largely driven by two OLMo domain-adapter suspects that produce unusually high invalid-response rates when queried through a non-native interface, as shown in the source-group analysis below. Table 9 reports results after excluding these two suspects using the same filtered pair set for all methods. On the pooled all-source evaluation, Stemma becomes the strongest method across all reported metrics, while the baselines remain largely unchanged. This indicates that its lower AUC on the full all-raw benchmark is primarily driven by interface-induced response-format failures in these models rather than a general loss of provenance discrimination.

Table 9: Performance under the all-raw interface after excluding two OLMo domain-adapter suspects. Best/second-best are bold/underlined.

Method	Pretrained sources				Instruct sources				All sources			
	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑
LLMmap	0.566 ±0.020	0.504 ±0.009	0.017 ±0.028	0.204 ±0.080	0.736 ±0.015	0.566 ±0.009	0.102 ±0.035	0.856 ±0.061	0.652 ±0.016	0.522 ±0.004	0.043 ±0.017	0.521 ±0.052
LLMPrint	0.606 ±0.047	0.506 ±0.013	0.004 ±0.010	0.412 ±0.181	0.757 ±0.025	0.582 ±0.024	0.123 ±0.055	1.025 ±0.117	0.674 ±0.031	0.546 ±0.007	0.064 ±0.026	0.643 ±0.131
MPT	**0.991** ±0.001	0.947 ±0.005	0.881 ±0.012	3.288 ±0.037	**0.994** ±0.002	**0.971** ±0.003	**0.923** ±0.019	3.303 ±0.039	0.989 ±0.001	0.945 ±0.007	0.853 ±0.048	2.715 ±0.025
ZeroPrint	0.727 ±0.013	0.544 ±0.021	0.043 ±0.015	0.821 ±0.040	0.864 ±0.007	0.717 ±0.006	0.400 ±0.018	1.614 ±0.045	0.798 ±0.010	0.620 ±0.015	0.194 ±0.035	1.189 ±0.038
Stemma	0.988 ±0.003	**0.980** ±0.006	**0.953** ±0.018	**3.790** ±0.211	0.992 ±0.002	0.964 ±0.003	0.915 ±0.015	**3.566** ±0.073	**0.990** ±0.001	**0.970** ±0.001	**0.932** ±0.010	**3.512** ±0.041

Performance by source provenance group. We examine Stemma's performance separately for each source provenance group under the all-raw setting, using the same grouping and evaluation protocol as in Appendix H.

Stemma performs strongly across six of the seven source provenance groups. In particular, the three Qwen-based groups remain well separated despite sharing similar architectures and training recipes, suggesting that the all-raw setting does not substantially weaken discrimination among these hard negative groups. The main exception is OLMo-2-7B, which exhibits substantially lower AUC and pAUC than the other source provenance groups.

Table 10: Stemma performance by source provenance group under the all-raw interface.

Source provenance group	AUC ↑	pAUC ↑	TPR@1% FPR ↑
Qwen-2.5-7B	0.985 ± 0.010	0.949 ± 0.033	0.900 ± 0.064
Qwen-2.5-14B	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Qwen3-1.7B	0.999 ± 0.002	0.988 ± 0.017	0.971 ± 0.039
Llama-3.1-8B	1.000 ± 0.001	0.997 ± 0.005	0.957 ± 0.064
Mistral-7B-v0.3	0.999 ± 0.001	0.991 ± 0.014	0.957 ± 0.064
Falcon3-7B	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
OLMo-2-7B	0.790 ± 0.012	0.761 ± 0.016	0.514 ± 0.032
Full benchmark (pooled)	0.967 ± 0.003	0.951 ± 0.001	0.894 ± 0.009

A pair-level inspection shows that most of this degradation is concentrated in two domain-adapter descendants. Under raw prompting, the fraction of probe–permutation responses from which no valid option can be extracted rises from approximately zero in the main setting to about 54% for the Humanities adapter and 17–22% for the Social-Sciences adapter, compared with a median invalid-response rate of only about 0.3% across model pairs in the same benchmark.

This pattern is consistent with a model–interface mismatch. Without their expected chat formatting, these models follow the required choice-response format less reliably, leading to frequent extraction failures. Since invalid outcomes are treated as non-aligned during verification, these failures lower the provenance scores of the affected positive pairs and weaken their separation from unrelated models. More broadly, these failures indicate degraded compliance with the task interface itself, reinforcing that the all-raw setting is a diagnostic control rather than a realistic deployment condition for these models. The remaining OLMo positive pairs are largely stable, suggesting that the lower group-level performance is driven by these interface-sensitive descendants rather than by a general degradation across the provenance group.

Performance across suspect adaptation types. We next examine performance across different forms of model adaptation under the all-raw setting, following the same grouping and evaluation protocol as in Appendix H.

Table 11: Stemma performance across suspect adaptation types under the all-raw interface.

Suspect adaptation type	AUC ↑	pAUC ↑	TPR@1% FPR ↑
Instruction tuning	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Fine-tuning	0.994 ± 0.003	0.963 ± 0.010	0.910 ± 0.042
Adapter tuning	0.914 ± 0.008	0.914 ± 0.002	0.825 ± 0.019
Quantisation	0.999 ± 0.001	0.994 ± 0.006	0.957 ± 0.039
Model merging	0.960 ± 0.013	0.930 ± 0.005	0.857 ± 0.000
Distillation	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Full benchmark (pooled)	0.967 ± 0.003	0.951 ± 0.001	0.894 ± 0.009

Stemma remains strong across most adaptation types under the all-raw interface. Instruction tuning, fine-tuning, quantisation, and distillation all retain strong discrimination, while model merging also improves relative to the main setting. Adapter tuning is the main exception and becomes the most challenging adaptation type under raw prompting. Consistent with the analysis above, this weaker result is largely driven by the same two OLMo domain-adapter descendants rather than reflecting a general degradation across adapter-tuned suspects.

J Deployment Robustness Benchmark and Analysis

J.1 Benchmark Construction

In addition to the main checkpoint-level benchmark, we construct a robustness benchmark to evaluate provenance signals under common deployment-time variations. It follows the same seven

provenance groups and uses their pretrained and instruction-tuned checkpoints as source models. Deployment variants are created from the corresponding instruction-tuned checkpoints by changing inference-time settings without modifying model weights. Pretrained checkpoints use the *raw* query interface, whereas instruction-tuned and chat-oriented checkpoints use the *chat* interface specified by their tokenizer configuration.

We consider five categories of deployment variation. For the system, role-play, and reasoning-oriented variants, the additional instruction is supplied through the system-message field of the model's chat template, while the original multiple-choice prompt remains unchanged in the user message.

System prompts. We evaluate two system-prompt variants that reflect commonly used assistant-style deployment settings. The first is a minimal generic helpful-assistant prompt, while the second uses the longer safety-oriented system prompt reported for Llama 2-Chat (Touvron et al., 2023).

Minimal assistant system prompt

```
You are a helpful assistant.
```

Safety-oriented assistant system prompt

```
You are a helpful, respectful and honest assistant. Always answer as helpfully as possible,
while being safe. Your answers should not include any harmful, unethical, racist, sexist,
toxic, dangerous, or illegal content. Please ensure that your responses are socially
unbiased and positive in nature.
If a question does not make any sense, or is not factually coherent, explain why instead of
answering something not correct. If you don't know the answer to a question, please don't
share false information.
```

Role-play prompts. We evaluate two role-play prompt variants, motivated by prior studies of persona and role prompting (Zheng et al., 2024b; Wang et al., 2024a). The first represents a simple occupational role, where the model is assigned a profession-based persona. The second represents a more detailed persona-style role, where the model is given a specific character description and instructed to imitate the assigned personality.

Occupational role prompt

```
You are a teacher.
```

Persona-style role prompt

```
You are Stephen Hawking, your description is: You are a brilliant physicist who defies the
limitations of your physical disability to become a world-renowned scientist, exploring the
mysteries of the universe while facing personal challenges in your relationships and health.
 Now please answer some questions to accurately show your personality traits! Your speaking
style should fully imitate the personality role assigned to you! Please do not expose that
you are an artificial intelligence model or a language model, you must always remember that
you are only assigned one personality role. Don't be verbose or too formal or polite when
speaking.
```

Reasoning-oriented prompts. We evaluate two reasoning-oriented prompt variants that are commonly used to encourage intermediate reasoning during inference. The first is the classic zero-shot chain-of-thought prompt (Kojima et al., 2022), which uses a short instruction to elicit step-by-step reasoning. The second follows the plan-and-solve prompting style (Wang et al., 2023), where the model is first asked to understand the problem and devise a plan before carrying out the solution.

Because verification is limited to 16 generated tokens, these variants do not necessarily reveal whether explicit chain-of-thought behaviour is actually elicited or whether it improves task performance. Our evaluation does not aim to establish either property. Instead, it tests whether Stemma can still extract the induced decisions and preserve provenance discrimination when the deployment interface is modified by reasoning-oriented system prompts.

Zero-shot chain-of-thought prompt

```
Let's think step by step.
```

Plan-and-solve prompt

```
Let's first understand the problem and devise a plan to solve the problem. Then, let's carry
 out the plan and solve the problem step by step.
```

Retrieval-augmented generation. We evaluate two retrieval-augmented generation variants using SQuAD 2.0 (Rajpurkar et al., 2018) as the retrieval corpus and `all-mpnet-base-v2` as the retriever (Reimers & Gurevych, 2019). Each SQuAD 2.0 context passage is treated as one retrieval document, and the rendered multiple-choice prompt is used as the retrieval query. The top-1 or top-3 retrieved passages are prepended to the original multiple-choice prompt in descending similarity order. Retrieved passages are inserted in a fixed context block before the original prompt.

Decoding settings. We evaluate four decoding variants that cover deterministic and stochastic generation regimes. The first uses greedy decoding, while the remaining three use stochastic decoding with moderate, high, and low sampling settings. Moderate sampling is the default decoding configuration used in the main benchmark and in the default robustness setting. The exact decoding parameters are summarised in Table 12.

Table 12: Decoding settings used in the robustness benchmark.

Setting	**Decoding mode**	**Temperature**	**top-p**	**top-k**
Greedy decoding	Deterministic	–	–	–
Moderate sampling (default)	Stochastic	0.7	0.9	50
High sampling	Stochastic	1.2	1.0	0
Low sampling	Stochastic	0.3	0.9	10

Together, the robustness benchmark uses 7 pretrained checkpoints as source models and includes the corresponding 7 instruction-tuned checkpoints as both instruct-source models and default suspect models under the moderate-sampling setting. Each instruction-tuned checkpoint is further redeployed under 11 alternative inference-time settings, yielding 77 additional deployment variants. Deployment variants reuse the calibrated prompt template of their corresponding default deployment. In total, the benchmark contains 91 deployment instances: 7 pretrained checkpoints, 7 default instruction-tuned deployments, and 77 additional deployment variants. Under the same pair-construction rule, this gives 1,260 evaluated pairs, including 168 positive pairs and 1,092 negative pairs.

J.2 Additional Analysis on the Deployment Robustness Benchmark

We further analyse Stemma's performance across subgroups of the deployment robustness benchmark to characterise variation in provenance-testing difficulty. We consider breakdowns by source provenance group and deployment variation.

Performance by source provenance group. We first examine Stemma's performance separately for each source provenance group on the deployment robustness benchmark, following the same grouping and evaluation protocol as in Appendix H.

Table 13: Stemma performance by source provenance group on the robustness benchmark.

Source provenance group	AUC ↑	pAUC ↑	TPR@1% FPR ↑
Qwen-2.5-7B	1.000 ± 0.000	0.998 ± 0.003	0.983 ± 0.023
Qwen-2.5-14B	0.999 ± 0.003	0.990 ± 0.021	0.975 ± 0.037
Qwen3-1.7B	1.000 ± 0.000	0.998 ± 0.004	0.992 ± 0.019
Llama-3.1-8B	0.992 ± 0.006	0.955 ± 0.023	0.858 ± 0.070
Mistral-7B-v0.3	0.998 ± 0.003	0.988 ± 0.018	0.967 ± 0.035
Falcon3-7B	0.992 ± 0.017	0.992 ± 0.019	0.983 ± 0.037
OLMo-2-7B	0.996 ± 0.006	0.992 ± 0.012	0.983 ± 0.023
Full benchmark (pooled)	0.995 ± 0.003	0.973 ± 0.013	0.924 ± 0.028

Stemma remains strong across all seven source provenance groups under deployment-time variation, with consistently high AUC and pAUC across groups, although Llama-3.1-8B is weaker at low FPR. Overall, provenance discrimination remains broadly preserved across lineages.

Performance across deployment variations. We next examine performance across different forms of deployment variation, including system prompting, role-play prompting, reasoning-oriented prompting, retrieval augmentation, and decoding changes.

Table 14: Stemma performance across deployment variations on the robustness benchmark.

Deployment variation	AUC ↑	pAUC ↑	TPR@1% FPR ↑
System prompting	0.999 ± 0.001	0.992 ± 0.011	0.971 ± 0.030
Role-play prompting	0.996 ± 0.003	0.980 ± 0.011	0.936 ± 0.059
Reasoning-oriented prompting	0.996 ± 0.008	0.984 ± 0.031	0.957 ± 0.059
Retrieval augmentation	0.993 ± 0.004	0.953 ± 0.016	0.871 ± 0.041
Decoding changes	0.997 ± 0.003	0.983 ± 0.011	0.938 ± 0.027
Full benchmark (pooled)	0.995 ± 0.003	0.973 ± 0.013	0.924 ± 0.028

Stemma remains strong across all five deployment variations, with AUC above 0.99 in every category. Retrieval augmentation is slightly more challenging at low FPR, potentially because additional retrieved context can shift the induced decisions, although discrimination remains strong overall.

K Full Ablation Results

Tables 15–19 report the detailed results underlying the ablation analyses in Section 5.4, including source-type breakdowns where applicable.

Probe selection. All variants use the same candidate pool and final budget of 40 probes. The *Random probes* variant uniformly samples probes from all candidate questions. *Non-gold pool* samples after restricting candidates to questions whose source decision differs from the gold answer. *Filtered pool* additionally applies the stability, robustness, and background-specificity filters, but samples uniformly without soft ranking. *Stemma* further ranks the filtered candidates and selects the top-scoring probes. All variants use the same cyclic-permutation verification and scoring procedure.

Table 15: Performance of the probe selection ablation. Best results are shown in bold.

Selection stage	Pretrained sources				Instruct sources				All sources			
	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑
Random probes	0.726 ±0.036	0.608 ±0.048	0.212 ±0.094	0.772 ±0.132	0.753 ±0.047	0.641 ±0.042	0.220 ±0.082	0.916 ±0.181	0.739 ±0.031	0.609 ±0.022	0.129 ±0.055	0.838 ±0.095
Non-gold pool	0.829 ±0.018	0.662 ±0.067	0.265 ±0.133	1.321 ±0.103	0.865 ±0.016	0.753 ±0.065	0.441 ±0.168	1.621 ±0.130	0.847 ±0.003	0.701 ±0.059	0.331 ±0.154	1.455 ±0.043
Filtered pool	0.940 ±0.020	0.867 ±0.052	0.661 ±0.162	2.424 ±0.327	0.950 ±0.006	0.870 ±0.021	0.731 ±0.044	2.414 ±0.104	0.945 ±0.011	0.873 ±0.019	0.708 ±0.036	2.369 ±0.160
Stemma	**0.961** ±0.007	**0.948** ±0.014	**0.882** ±0.044	**2.861** ±0.210	**0.974** ±0.004	**0.945** ±0.010	**0.878** ±0.025	**2.964** ±0.049	**0.967** ±0.005	**0.944** ±0.002	**0.871** ±0.020	**2.864** ±0.130

Pool size and probe budget. We jointly vary the candidate pool size from 1,000 to 6,000 questions and the final probe budget from 10 to 100, while keeping all other settings fixed. Tables 16 and 17 report the corresponding pAUC and AUC results, respectively. The default configuration is highlighted. As the sweep was conducted independently from the main experiment, the default-setting result differs slightly from the main-run result due to stochastic decoding and hardware-level nondeterminism.

Probe dataset. For the dataset ablation, candidate questions are drawn from the test splits of MMLU and MMLU-Pro (Hendrycks et al., 2021; Wang et al., 2024b), the training split of CommonsenseQA (Talmor et al., 2019), and the validation split of CosmosQA (Huang et al., 2019), using 3,000 questions per dataset. Decision-interface calibration uses 100 questions from the validation splits of MMLU, MMLU-Pro, and CommonsenseQA, and the training split of CosmosQA,

Table 16: pAUC across candidate pool sizes and probe budgets.

Pool size	Number of probes K									
	10	20	30	**40**	50	60	70	80	90	100
1,000	0.855 ±0.015	0.901 ±0.018	0.903 ±0.022	0.901 ±0.017	0.903 ±0.015	0.903 ±0.011	0.908 ±0.017	0.903 ±0.014	0.901 ±0.016	0.899 ±0.016
2,000	0.862 ±0.025	0.909 ±0.020	0.924 ±0.018	0.930 ±0.023	0.933 ±0.019	0.932 ±0.021	0.933 ±0.018	0.933 ±0.009	0.929 ±0.009	0.927 ±0.011
3,000	0.858 ±0.029	0.920 ±0.017	0.932 ±0.016	0.941 ±0.007	0.950 ±0.008	0.948 ±0.012	0.947 ±0.011	0.946 ±0.011	0.947 ±0.012	0.945 ±0.014
4,000	0.865 ±0.035	0.915 ±0.014	0.923 ±0.010	0.935 ±0.009	0.943 ±0.007	0.948 ±0.007	0.952 ±0.007	0.951 ±0.009	0.951 ±0.007	0.949 ±0.010
5,000	0.861 ±0.033	0.907 ±0.016	0.929 ±0.017	0.933 ±0.012	0.944 ±0.005	0.949 ±0.007	0.951 ±0.008	0.951 ±0.008	0.952 ±0.007	0.951 ±0.008
6,000	0.855 ±0.039	0.900 ±0.017	0.927 ±0.019	0.933 ±0.017	0.937 ±0.014	0.945 ±0.006	0.947 ±0.007	0.949 ±0.004	0.952 ±0.002	0.954 ±0.003

Table 17: AUC across candidate pool sizes and probe budgets.

Pool size	Number of probes K									
	10	20	30	**40**	50	60	70	80	90	100
1,000	0.937 ±0.018	0.959 ±0.008	0.958 ±0.007	0.957 ±0.005	0.957 ±0.005	0.958 ±0.005	0.959 ±0.004	0.958 ±0.005	0.957 ±0.004	0.957 ±0.005
2,000	0.946 ±0.008	0.958 ±0.003	0.961 ±0.003	0.963 ±0.003	0.963 ±0.003	0.964 ±0.003	0.965 ±0.002	0.964 ±0.002	0.965 ±0.001	0.964 ±0.002
3,000	0.952 ±0.008	0.964 ±0.006	0.968 ±0.005	0.968 ±0.006	0.969 ±0.005	0.969 ±0.005	0.969 ±0.005	0.969 ±0.004	0.969 ±0.004	0.969 ±0.004
4,000	0.944 ±0.007	0.962 ±0.004	0.965 ±0.003	0.967 ±0.004	0.967 ±0.004	0.969 ±0.003	0.969 ±0.003	0.969 ±0.003	0.969 ±0.002	0.969 ±0.002
5,000	0.947 ±0.014	0.960 ±0.009	0.967 ±0.004	0.968 ±0.003	0.969 ±0.003	0.970 ±0.002	0.971 ±0.002	0.971 ±0.001	0.971 ±0.002	0.971 ±0.002
6,000	0.948 ±0.008	0.958 ±0.012	0.965 ±0.007	0.967 ±0.005	0.968 ±0.005	0.969 ±0.004	0.969 ±0.003	0.969 ±0.003	0.969 ±0.003	0.970 ±0.003

ensuring that calibration and fingerprint construction use disjoint splits. The selection procedure and probe budget remain fixed across datasets.

Table 18: Performance of the probe dataset ablation. Best results are shown in bold.

Dataset	Pretrained sources				Instruct sources				All sources			
	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑
CommonsenseQA	0.954 ±0.010	0.862 ±0.036	0.596 ±0.131	2.328 ±0.129	0.969 ±0.007	**0.948** ±0.020	**0.886** ±0.051	2.884 ±0.170	0.962 ±0.006	0.910 ±0.020	0.786 ±0.038	2.480 ±0.123
CosmosQA	0.901 ±0.009	0.773 ±0.008	0.502 ±0.047	1.817 ±0.052	0.958 ±0.004	0.883 ±0.010	0.694 ±0.032	2.676 ±0.057	0.934 ±0.006	0.821 ±0.006	0.608 ±0.017	2.091 ±0.044
MMLU-Pro	0.948 ±0.007	0.944 ±0.003	0.878 ±0.014	2.742 ±0.061	0.945 ±0.006	0.916 ±0.008	0.820 ±0.030	2.620 ±0.087	0.945 ±0.006	0.927 ±0.002	0.841 ±0.027	2.623 ±0.085
MMLU (default)	**0.961** ±0.007	**0.948** ±0.014	**0.882** ±0.044	**2.861** ±0.210	**0.974** ±0.004	0.945 ±0.010	0.878 ±0.025	**2.964** ±0.049	**0.967** ±0.005	**0.944** ±0.002	**0.871** ±0.020	**2.864** ±0.130

Background model. For the background-model ablation, we compare `Phi-3.5-mini-instruct`, `Yi-1.5-9B-Chat`, and `Gemma-2-9B-IT`, while keeping all other settings fixed. In the combined setting, we pool the permutation-level decisions from all three models and compute background alignment as the fraction matching the source decision.

Table 19: Performance of the background model ablation. Best results are shown in bold.

Background model	Pretrained sources				Instruct sources				All sources			
	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑	AUC ↑	pAUC ↑	TPR ↑	d' ↑
Phi-3.5 (default)	0.961 ±0.007	0.948 ±0.014	0.882 ±0.044	2.861 ±0.210	0.974 ±0.004	0.945 ±0.010	0.878 ±0.025	2.964 ±0.049	0.967 ±0.005	0.944 ±0.002	0.871 ±0.020	2.864 ±0.130
Yi-1.5	0.958 ±0.004	0.919 ±0.013	0.812 ±0.060	2.673 ±0.176	0.961 ±0.007	0.922 ±0.031	0.812 ±0.071	2.771 ±0.183	0.958 ±0.006	0.919 ±0.020	0.792 ±0.056	2.665 ±0.150
Gemma-2	0.961 ±0.007	0.934 ±0.021	0.833 ±0.064	2.755 ±0.203	0.970 ±0.004	0.949 ±0.006	0.882 ±0.030	2.938 ±0.039	0.965 ±0.005	0.941 ±0.006	0.843 ±0.021	2.774 ±0.110
Combined	**0.970** ±0.006	**0.952** ±0.013	**0.902** ±0.030	**2.988** ±0.138	**0.981** ±0.004	**0.964** ±0.010	**0.918** ±0.029	**3.161** ±0.150	**0.975** ±0.004	**0.954** ±0.009	**0.896** ±0.023	**2.990** ±0.124

L RUNTIME MEASUREMENTS

Measurement protocol. We measure the wall-clock runtime of all methods on the same NVIDIA L40S GPU with 48 GB of memory. Fingerprint construction is measured per source model and reported as the median over the 14 source checkpoints. Fingerprint verification measures the suspect-side computation required to evaluate one source–suspect pair and is reported as the median over the corresponding verification executions, assuming that the source fingerprint has already been constructed. The measurements cover the principal computational procedures, such as model inference and, where applicable, prompt optimisation involving backward passes, but exclude model loading. Reported runtimes are rounded to the nearest second. Figure 6 summarises the resulting construction and verification runtimes.

LLMmap. The reported construction time includes evaluating each source under 100 prompt configurations on the 8 diagnostic queries, encoding the resulting responses, and averaging the feature vectors to construct the gallery representation. The reported verification time includes evaluating each suspect once on the same 8 queries and encoding its responses using the pretrained feature extractor.

LLMPrint. The reported construction time includes optimising one adversarial suffix for each of 100 word pairs, each consisting of two single-token words, over 200 GCG steps, and estimating the source model's preference direction for each pair using 100 repeated one-token generations. The reported verification time includes querying each suspect model with the fixed optimised suffixes, estimating its preference direction for each pair using 100 repeated one-token generations, and comparing these directions with the stored source preferences.

MPT. The reported construction time includes querying each source on 5,000 prompts with single-token generation. The reported verification time includes querying each suspect once on the same prompt set and matching the resulting continuations against the cached source continuations to compute the hit rate.

ZeroPrint. The reported construction time includes querying each source on 2 HumanEval base prompts and 4 word-substitution variants per prompt, with 20 generations for each query, embedding the responses, and constructing the Jacobian-based fingerprint. The reported verification time includes applying the same procedure to each suspect.

Stemma. The reported construction time includes calibrating the source model and the configured background model on 100 held-out validation questions, followed by evaluating them on 3,000 candidate questions under cyclic option permutations to obtain the source-decision and background-alignment statistics used for probe selection. The reported verification time includes calibration where required and querying each suspect on the 40 source-specific probes under four cyclic permutations.

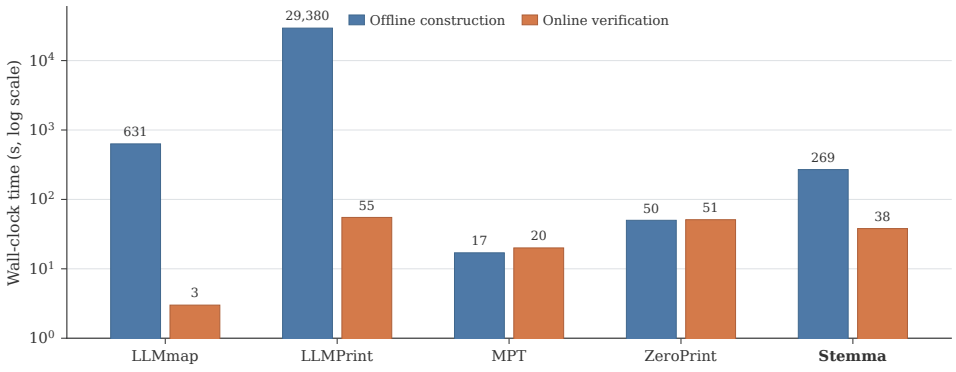

Figure 6: Median wall-clock runtimes for fingerprint construction and verification.

M THE STEMMATIC ANALOGY BEHIND STEMMA

Stemmatic analysis. Manuscript studies draw on several complementary approaches, including palaeography, codicology, and stemmatics. Palaeography examines handwriting and scribal practices, while codicology examines the material construction and production of manuscripts. Stemmatics instead focuses on reconstructing the transmission history of works preserved in multiple manuscript copies, particularly when the original text and many intermediate copies no longer survive and only a set of extant manuscripts, commonly termed *witnesses*, remains (Heikkilä & Roos, 2016).

To infer the genealogy of the transmitted text, stemmatic analysis aligns witnesses at corresponding textual locations and compares their variant readings, including substitutions, omissions, additions, and transpositions. It therefore relies on structured patterns of textual variation rather than treating overall visual or stylistic resemblance as the primary genealogical signal. The genealogical value of a shared variant depends on how readily it could have arisen independently. Variants that are widespread across the textual tradition or easily reproduced provide limited evidence, whereas distinctive variants carry greater genealogical value because their preservation by multiple witnesses may indicate descent from a common exemplar. Stemmatic inference therefore considers not merely the presence of shared variation, but its genealogical significance (Andrews, 2016). The inferred relationships are conventionally represented by a *stemma codicum*, or *stemma*, a tree-like representation of textual descent among surviving witnesses and reconstructed intermediate exemplars.

Analogy to model provenance. The name Stemma reflects this methodological analogy. Candidate questions correspond to aligned textual locations, model instances to manuscript witnesses, and induced decisions to variant readings. Existing response-based fingerprints derive provenance signals from observable characteristics of generated responses, much as palaeography and codicology draw on handwriting and material features, whereas Stemma maps model behaviour into a common finite decision space and compares structured decision variants across aligned probes. Decisions readily reproduced by unrelated models provide limited provenance evidence, just as common or independently arising readings provide limited evidence of textual affiliation. By contrast, stable, robust, and specific decisions preserved by a suspect resemble distinctive inherited variants. Stemma therefore tests model provenance through structured patterns of induced decision region inheritance rather than through signals tied to particular prompt or response forms.