SCORE: A Model Equality Test for Auditing Black-Box LLM APIs

Anonymous authors
Paper under double-blind review

Abstract

Users increasingly rely on black-box inference APIs as their primary interface to large language models (LLMs), for both closed- and open-weight models (e.g., Llama models are now widely accessed through Amazon Bedrock and Azure AI Studio), with little insight into which model is actually deployed. To reduce costs or maliciously alter model behavior, API providers may silently deploy quantized models or substitute entirely different ones, potentially degrading model performance and compromising safety. Detecting such substitutions is challenging because users cannot inspect model weights and, in most cases, lack access to internal model signals such as output logits, leaving generated text as the only observable. To address this problem, we propose **S**emantic **C**ontrast for **O**utput-only **R**ank-based Model **E**quality Testing (SCORE), an output-only auditing method that tests whether a target endpoint and a trusted reference endpoint induce the same prompt–response joint distribution, thereby assessing whether the black-box API behaves consistently with the model it claims to serve. SCORE measures the change in semantic recovery induced by a control prefix, converts each target score into a randomized rank calibrated against reference scores, and tests the resulting ranks for uniformity at significance level α using only generated text. We evaluate SCORE across multiple threat scenarios, including model quantization, supervised fine-tuning, and full model replacement, as well as on real-world commercial inference API endpoints. Across these settings, SCORE consistently achieves higher statistical power than existing methods under strict output-only constraints.

1 Introduction

Hosted APIs Li et al. (2026) have become a primary interface to large language models (LLMs) across consumer applications Zhou et al. (2026), enterprise tools, and scientific workflows Ren et al. (2025). Users submit prompts and receive generated text, but cannot inspect model weights, hidden states, or token-level log probabilities. They therefore have limited means of verifying whether an API actually serves the model claimed by its provider. Meanwhile, high deployment costs and pressure to reduce time to first token create incentives for providers to serve smaller or quantized model variants. When undisclosed, such modifications may shift the model's response distribution, degrade performance, or introduce new safety risks Gao et al. (2025); Lin et al. (2026).

These risks motivate LLM API auditing: assessing whether a target endpoint is behaviorally consistent with the model it claims to serve. This task is difficult under strict output-only access, where the auditor observes only a limited number of text outputs and cannot access model weights, internal architectures, or intermediate representations. Behavioral-fingerprint-based audits may also be vulnerable to evasion. A dishonest provider could recognize probe queries with distinctive distributional patterns and selectively route them to the claimed model. Even without knowing the precise auditing strategy, the provider could mix models

Table 1: Comparison of LLM API auditing methods by *accuracy* (Acc.), *strict output-only applicability* (O-Only), and *robustness to adversarial providers* (Rob.).

Method	Acc.	O-Only	Rob.
SCORE (Ours)	✓	✓	✓
RUT Zhu et al. (2026)	✓	✗	✓
MMD Gao et al. (2025)	✓	✓	✗
PTS Shi et al. (2026)	✓	✓	✗
Classifier Sun et al. (2025)	✗	✓	✗
Identity-prompting Huang et al. (2025)	✗	✓	✗
Benchmark Chen et al. (2023)	✗	✓	✗

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.

or switch between them across requests, weakening the stability and identifiability of any single behavioral fingerprint. Reliable and difficult-to-evade model auditing using only generated text therefore remains an important open problem.

We argue that a practical model-equality test for LLM API auditing should satisfy three requirements: accuracy, output-only applicability, and robustness to evasive behavior. Accuracy requires reliable decisions, including type I error control and sufficient detection power, under a practical query budget. Output-only applicability requires the method to operate solely on text returned by the target and reference endpoints, without model weights, output probabilities, or other privileged information. Robustness requires the audit to remain informative under query recognition, dynamic routing, and request-level model switching.

Existing model-equality tests fall short of one or more of these requirements, as summarized in Table 1. Existing approaches include the rank-based uniformity test Zhu et al. (2026), maximum mean discrepancy (MMD) tests Gao et al. (2025), trained text classifiers Sun et al. (2025), identity prompting Huang et al. (2025), and benchmark-performance comparisons Chen et al. (2023). Trained classifiers can require prohibitively large query budgets, while identity prompting may fail to distinguish changes in model scale, version, or quantization Cai et al. (2025). Implementations based on MMD Gao et al. (2025) or benchmark comparisons rely on prescribed or recognizable query distributions, which an adversarial provider may detect and handle through selective routing or prompt caching Gu et al. (2025). RUT Zhu et al. (2026) offers black-box access only on the target side and requires privileged access to the reference model. These limitations motivate an auditing method that treats both endpoints as black boxes while retaining statistically valid detection.

To address this gap, we propose Semantic Contrast for Output-only Rank-based Model Equality Testing (SCORE), an output-only framework for testing whether a target endpoint and a trusted reference endpoint induce the same prompt–response joint distribution. For each probe, Semantic Contrast constructs paired prompts at held-out lexical positions, with and without a fixed control prefix, and measures the resulting change in semantic recovery. This procedure produces one target score and a collection of independently generated reference scores for each probe. Under the model-equality null hypothesis, conditional score exchangeability makes the reference-calibrated randomized rank exactly $\mathrm{Uniform}[0, 1]$. Independent probes and generation blocks therefore produce independent and identically distributed uniform ranks, reducing model auditing to a one-sample uniformity test at significance level α. The entire procedure uses generated text alone and requires no access to model weights, logits, token probabilities, or hidden states from either endpoint.

Our main contributions are summarized as follows:

- **A Novel Strictly Output-only Test for Auditing LLM APIs.** We propose SCORE, a model-equality test that uses only generated text from the target and reference endpoints. It transforms semantic-contrast scores into reference-calibrated randomized ranks, enabling a level-α audit without access to model internals.
- **Evaluation across multiple threat scenarios.** We evaluate SCORE under model quantization, supervised fine-tuning (SFT), and full model replacement. Across these settings, SCORE consistently achieves higher detection power than the evaluated baselines.
- **Auditing real-world commercial endpoints.** We apply SCORE to multiple publicly accessible commercial LLM APIs and compare it with existing auditing baselines, demonstrating its practical utility under strict output-only access.

2 Related Work

Auditing LLM APIs. Early work monitors deployed LLMs through longitudinal benchmarks or task-level behavioral measurements (Chen et al., 2023; Eyuboglu et al., 2024), but aggregate performance may miss subtle distributional changes or partial model routing. Statistical audits instead compare endpoint behavior directly. Gao et al. (2025) formalize Model Equality Testing and apply a kernel-MMD two-sample test to sampled completions. Shi et al. (2026) represent the reference endpoint's output distribution with a frequency-weighted prefix tree and construct statistics sensitive to distributional deviations for auditing LLM APIs. Zhu et al. (2026) transform target scores into reference-calibrated ranks and test their uniformity; however, their primary log-rank instantiation evaluates generations using token-level statistics from a fully accessible local reference

model. More recent black-box audits exploit knowledge-boundary recall, pseudo-random output biases, or repeated constrained generations to detect model substitution and routing dilution (Fang et al., 2026; Zhang et al., 2026; Wu et al., 2026). Cai et al. (2025) analyze quantization, randomized substitution, and benchmark-aware evasion, and investigate hardware-backed verification through trusted execution environments. Complementary inference-integrity methods use reference replay with synchronized sampling or verifiable computation (Karvonen et al., 2026; Guo et al., 2026), but assume a faithful local implementation, controllable sampling, or provider-side support. Unlike these approaches, SCORE tests model equality by measuring control-prefix-induced changes in semantic recovery and converting the resulting semantic-contrast scores into reference-calibrated randomized ranks, thereby enabling sensitive and statistically valid model-equality testing under strict output-only access to both endpoints. Additional related work is provided in the Appendix B.

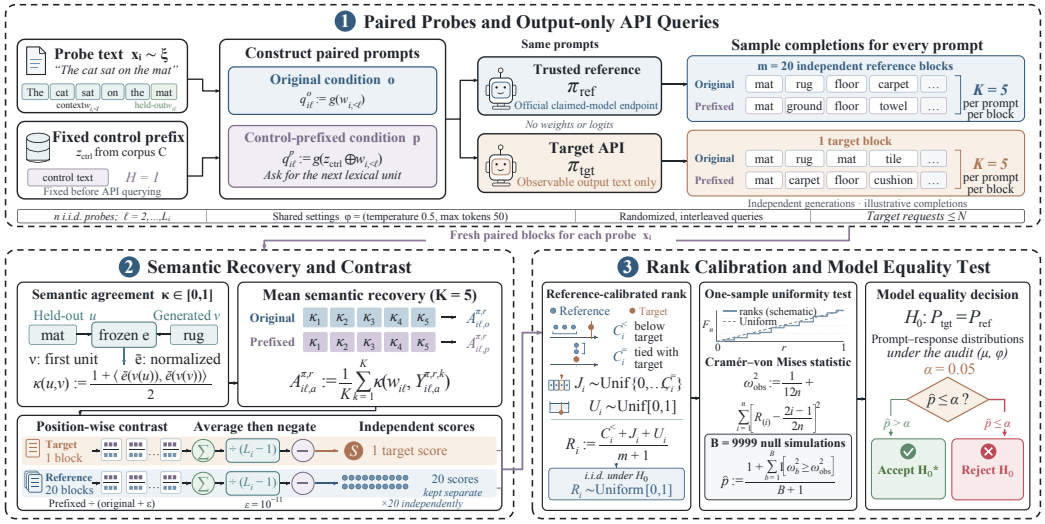


Figure 1: **Overview of the SCORE framework for LLM API auditing.** We query the target and reference APIs with paired original and control-prefixed prompts, then compute contrast scores from semantic recovery ratios. These scores are converted into randomized ranks and tested for uniformity with Monte Carlo calibration to determine whether to reject model equality at significance level α.

3 Problem Formulation

Models. We formulate LLM API auditing as model equality testing (Gao et al., 2025), comparing the conditional response kernels of a target endpoint and a trusted reference endpoint for the claimed model. For each prompt $q \in \mathcal{Q}$ and decoding configuration ϕ, the reference and target endpoints induce conditional response kernels $\pi_{\mathrm{ref}}(\cdot \mid q; \phi)$ and $\pi_{\mathrm{tgt}}(\cdot \mid q; \phi)$ on $\mathcal{Y}$. For proprietary models such as GPT and Gemini, the reference is the official provider endpoint under a preregistered model identifier. The shared configuration ϕ contains only generation controls supported by both APIs, and the auditor observes exclusively their generated text.

Audit distribution. Let $X \sim \xi$ be a random probe text in $\mathcal{X}$, with $L(X) \geq 2$ lexical units. Conditional on $X = x$, draw

$$\Lambda \mid X = x \sim \mathrm{Unif}\{2, \ldots, L(x)\}, \qquad A \mid X = x \sim \mathrm{Unif}\{o, p\}, \tag{1}$$

independently, and set

$$Q := T(X, \Lambda, A), \qquad \mu := \mathcal{L}(Q), \tag{2}$$

where T is the prespecified deterministic prompt-construction map instantiated in Eq. equation 7. Thus, ξ determines the weight of each probe, while eligible positions and prompt conditions are weighted uniformly within each probe. The support of μ defines the scope of the equality test. In the finite audit, we draw $X_1, \ldots, X_n \overset{\mathrm{iid}}{\sim} \xi$ and enumerate all position–condition pairs for each probe rather than sampling them. This stratifies the conditional design exactly, while the probes X_i remain the independent units used by the final test.

Hypotheses. We use samples from the reference endpoint and the target endpoint to test whether they induce the same distribution over prompt–response pairs under μ and ϕ, distinguishing between

$$\begin{aligned} H_0 &: \quad P_{\mathrm{ref}} = P_{\mathrm{tgt}}, \\ H_1 &: \quad P_{\mathrm{ref}} \neq P_{\mathrm{tgt}}. \end{aligned} \tag{3}$$

Throughout, $P_s(q, y) := \mu(q)\pi_s(y \mid q; \phi)$ denotes the corresponding joint distribution for $s \in \{\mathrm{ref}, \mathrm{tgt}\}$. The estimand is equality of the observable prompt–response joint distribution under (μ, ϕ).

Threat model and testing objective. Each target completion at q follows an arbitrary alternative with prompt-dependent probability $\rho(q)$. Thus,

$$\pi_{\mathrm{tgt}}(\cdot \mid q; \phi) = \big[1 - \rho(q)\big]\pi_{\mathrm{ref}}(\cdot \mid q; \phi) + \rho(q)\pi_{\mathrm{alt}}(\cdot \mid q; \phi), \qquad \rho(q) \in [0, 1], \tag{4}$$

where π_{alt} is arbitrary and $\rho(q)$ is the probability of substitution. This model includes randomized, partial, and prompt-selective routing. The routing policy is memoryless and depends on the current prompt. The auditor observes a transcript $D_N = (D_N^{\mathrm{tgt}}, D^{\mathrm{ref}})$ collected using at most N target requests. Reference requests are reported separately. A level-α test returns $\delta_\alpha(D_N; \zeta) \in \{0, 1\}$, where ζ collects all auxiliary randomization and 1 denotes rejection of H_0. For every common distribution $P = P_{\mathrm{tgt}} = P_{\mathrm{ref}}$, validity requires

$$\Pr_P[\delta_\alpha(D_N; \zeta) = 1] \leq \alpha. \tag{5}$$

For fixed P_{ref} and $P_{\mathrm{tgt}} \neq P_{\mathrm{ref}}$, the power is

$$\beta(P_{\mathrm{tgt}}; P_{\mathrm{ref}}) := \Pr_{P_{\mathrm{tgt}}, P_{\mathrm{ref}}}[\delta_\alpha(D_N; \zeta) = 1]\,. \tag{6}$$

We seek high power under Eq. equation 5 and the target-query budget. Both probabilities integrate over probes, endpoint generations, and auxiliary randomization. Target and reference requests are issued in randomized, interleaved order within a prespecified audit window. Both response kernels remain stationary throughout the window. Conditional on the prompts, all completions are independent across endpoints, positions, conditions, sample indices, probes, and score replicates.

4 Method

To test whether the target and reference endpoints induce the same prompt–response joint distribution, we propose SCORE, a black-box model equality test for API auditing, as illustrated in Fig. 1. Section 4.1 constructs a probe-level semantic-contrast score from paired completions, while Section 4.2 calibrates each target score against independently generated reference scores. Under H_0, conditional score exchangeability makes each reference-calibrated randomized rank exactly $\mathrm{Uniform}[0, 1]$. Independence across probes yields a valid one-sample uniformity test at level $\alpha = 0.05$.

4.1 Semantic Contrast

Paired lexical-unit probes. A fixed, endpoint-independent tokenizer segments each x_i into $L_i = L(x_i)$ lexical units $(w_{i1}, \ldots, w_{iL_i})$, each a word or punctuation mark. For $\ell = 2, \ldots, L_i$, let $w_{i,<\ell}$ be the exact textual prefix preceding the held-out unit $w_{i\ell}$. Before sampling responses, we select $H = 1$ text from a control corpus $\mathcal{C}$ by a fixed rule to form the control prefix z_{ctrl}. Let $\oplus$ denote concatenation with a fixed boundary delimiter. A deterministic prompt template g requests the next lexical unit and constructs

$$q_{i\ell}^o := g(w_{i,<\ell})\,, \qquad q_{i\ell}^p := g(z_{\mathrm{ctrl}} \oplus w_{i,<\ell})\,. \tag{7}$$

For each condition, both APIs receive the same serialized prompt and common decoding settings. The resulting contrast measures the change in semantic recovery induced by the control prefix.

Semantic recovery and contrast. For endpoint $\pi \in \{\pi_{\mathrm{ref}}, \pi_{\mathrm{tgt}}\}$ and score replicate r, we draw $K = 5$ independent completions from each prompt:

$$Y_{i\ell,o}^{\pi,r,k} \sim \pi(\cdot \mid q_{i\ell}^o; \phi)\,, \qquad Y_{i\ell,p}^{\pi,r,k} \sim \pi(\cdot \mid q_{i\ell}^p; \phi)\,, \quad k = 1, \ldots, K. \tag{8}$$

The operator ν trims whitespace, strips tokenizer boundary markers and takes the first lexical unit. For a frozen endpoint-independent encoder e ($e(z) \neq 0$), let $\bar{e}(z) := e(z)/\|e(z)\|_2$ and define

$$\kappa(u, v) := \frac{1 + \langle \bar{e}(\nu(u)), \bar{e}(\nu(v)) \rangle}{2}. \tag{9}$$

The rescaled cosine $\kappa \in [0, 1]$ measures graded semantic agreement. For $a \in \{o, p\}$, its empirical mean is

$$A^{\pi,r}_{i\ell,a} := \frac{1}{K} \sum_{k=1}^{K} \kappa\Big(w_{i\ell}, Y^{\pi,r,k}_{i\ell,a}\Big). \tag{10}$$

We average the negated prefix-to-original recovery ratios across positions:

$$S^{\pi,r}_i := -\frac{1}{L_i - 1} \sum_{\ell=2}^{L_i} \frac{A^{\pi,r}_{i\ell,p}}{A^{\pi,r}_{i\ell,o} + \varepsilon}, \qquad \varepsilon = 10^{-11}. \tag{11}$$

The constant ε ensures a strictly positive denominator, while negation makes larger scores indicate a lower mean recovery ratio. Each score uses a fresh block of $2K(L_i - 1)$ completions.

Instantiation and exchangeability. We implement ν with NLTK word tokenization and e with the frozen encoder. Every eligible position uses the semantic agreement rule in Eq. equation 9. Both endpoints receive identical messages and settings: $\phi = (0.5, 50)$ (temperature, maximum output tokens). Endpoint-specific serving instructions are incorporated into the corresponding response kernel. The control prefix and score map are fixed before querying, preserving null calibration. For ξ-almost every x_i, H_0 makes the target and reference generation blocks conditionally i.i.d. Applying the same deterministic score map therefore yields exchangeable scores.

4.2 Output-only Ranks for Model Equality

Reference-calibrated randomized ranks. For each probe x_i, mutually independent generation blocks yield one target score and $m = 20$ reference scores:

$$S^{\mathrm{tgt}}_i := S^{\pi_{\mathrm{tgt}},0}_i, \qquad S^{\mathrm{ref}}_{ij} := S^{\pi_{\mathrm{ref}},j}_i, \quad j = 1, \ldots, m. \tag{12}$$

Let $C^{<}_i$ and $C^{=}_i$ count reference scores strictly below and equal to S^{tgt}_i, respectively. Conditional on the scores, draw the J_i independently with

$$J_i \sim \mathrm{Uniform}\{0, \ldots, C^{=}_i\}.$$

Independently of the scores and $J_{1:n}$, draw $U_i \overset{\text{i.i.d.}}{\sim} \mathrm{Uniform}[0, 1]$ and define

$$R_i := \frac{C^{<}_i + J_i + U_i}{m + 1}. \tag{13}$$

Under H_0, the target and reference scores are i.i.d. conditional on x_i. Exchangeability and J_i make $C^{<}_i + J_i$ uniform on $\{0, \ldots, m\}$, and U_i fills the corresponding interval of width $1/(m + 1)$. Consequently,

$$R_i \mid x_i, H_0 \sim \mathrm{Uniform}[0, 1]. \tag{14}$$

Thus, H_0 induces i.i.d. uniform ranks, yielding a one-sample uniformity test (Zhu et al., 2026).

Uniformity test. Let $R_{(1)} \leq \cdots \leq R_{(n)}$ denote the ordered ranks. Their departure from uniformity is measured by the Cramér–von Mises statistic, which accumulates discrepancies across the unit interval (Cramér, 1928):

$$\omega^2_{\mathrm{obs}} := \frac{1}{12n} + \sum_{i=1}^{n} \left(R_{(i)} - \frac{2i - 1}{2n} \right)^2. \tag{15}$$

Larger values indicate stronger aggregate nonuniformity. To calibrate the observed departure, draw $B = 9999$ independent null samples of n i.i.d. $\mathrm{Uniform}[0, 1]$ ranks, independently of the observed ranks, and let ω^2_b be their statistics. The plus-one Monte Carlo p-value is

$$\widehat{p} := \frac{1 + \sum_{b=1}^{B} \mathbb{I}\big[\omega^2_b \geq \omega^2_{\mathrm{obs}}\big]}{B + 1}. \tag{16}$$

We reject H_0 when $\widehat{p} \leq \alpha$ and choose B such that $(B + 1)^{-1} \leq \alpha$, ensuring an attainable rejection threshold. Under H_0, the observed and simulated statistics are exchangeable. Consequently, the plus-one procedure controls the type I error rate at level α for every common joint distribution under the stated assumptions. Finite-sample validity and power guarantees are provided in Appendix G.

Query complexity. Let $W := \sum_{i=1}^{n}(L_i - 1)$ denote the number of evaluated positions. The audit uses $2KW$ target and $2mKW$ reference completions; the final uniformity test contains n independent ranks. Each target request returns $c_{\mathrm{tgt}} = 1 \in \mathbb{N}_+$ independent completions, and each reference request returns $c_{\mathrm{ref}} = 1 \in \mathbb{N}_+$ independent completions. Within each (i, ℓ, a) prompt, reference completions are pooled and split into m disjoint groups of K. The resulting request counts are

$$N_{\mathrm{tgt}} = 2W \left\lceil \frac{K}{c_{\mathrm{tgt}}} \right\rceil, \qquad N_{\mathrm{ref}} = 2W \left\lceil \frac{mK}{c_{\mathrm{ref}}} \right\rceil. \tag{17}$$

The design imposes $L_i \le L_{\max} = 50$ almost surely under ξ and chooses n such that the worst-case target request count, $2n(L_{\max} - 1)\lceil K/c_{\mathrm{tgt}} \rceil$, does not exceed N. This ensures $N_{\mathrm{tgt}} \le N$. Excess completions per prompt are discarded. Target and reference blocks are collected in randomized, interleaved order within the same prespecified audit window.

5 Experiments

We evaluate SCORE under three controlled model-deviation scenarios: model quantization 5.2, supervised fine-tuning (SFT) 5.3, and full model replacement 5.4. We further assess its generalization across diverse query domains 5.5 and evaluate its effectiveness on live commercial API endpoints 5.6. Power AUC serves as the primary evaluation metric throughout our experiments.

5.1 Experimental Setup

To evaluate detection under adversarial routing, we simulate probabilistic model substitution. A fraction $q \in [0, 1]$ of API requests is routed to an alternative model, while the remaining requests are served by the reference model. The alternative may be a quantized or fine-tuned variant, or an entirely different model. For each $q > 0$, we estimate statistical power as the probability of correctly rejecting H_0 at significance level α. The case $q = 0$ corresponds to the null and is used to assess type I error calibration. We summarize detection performance by the area under the power–substitution-probability curve over $q \in [0, 1]$. The resulting power AUC lies between 0 and 1 and measures average detection power across substitution probabilities. Higher values indicate greater robustness to partial model substitution within this threat model. We compute 95% confidence intervals for power AUC using bootstrap resampling and use 95% Wilson intervals for pointwise power estimates. Full power curves and additional implementation details are provided in Appendix D.

Data We use WildChat(Zhao et al., 2024) as the primary probe-text distribution because it contains naturally occurring user conversations and provides a realistic approximation to ordinary API traffic. To evaluate cross-domain generalization, we additionally use multilingual Wikipedia for text continuation and MATH(Hendrycks et al., 2021) for mathematical reasoning. We use BigCodeBench(Zhuo et al., 2025) for code generation and UltraChat(Ding et al., 2023) for instruction-following dialogue. All examples are deduplicated after Unicode normalization and partitioned by document identifier into mutually disjoint development, test, control, and online-audit splits. This partitioning prevents source-document overlap across experimental stages.

(a) Power AUC for detecting quantized variants. **Bold** = best method; gray = none reliable.

Model	SCORE	MMD	RUT	PTS	KS
Llama–4-bit	**0.916** $_{528^{-}\,572^{+}}$	0.625 $_{394^{-}\,356^{+}}$	0.642 $_{362^{-}\,408^{+}}$	0.746 $_{461^{-}\,435^{+}}$	0.474 $_{264^{-}\,304^{+}}$
Llama–8-bit	0.101 $_{64^{-}\,58^{+}}$	0.008 $_{04^{-}\,06^{+}}$	0.032 $_{20^{-}\,18^{+}}$	0.044 $_{24^{-}\,28^{+}}$	0.006 $_{05^{-}\,03^{+}}$
Gemma–4-bit	**0.796** $_{449^{-}\,507^{+}}$	0.214 $_{132^{-}\,124^{+}}$	0.392 $_{219^{-}\,251^{+}}$	0.568 $_{355^{-}\,327^{+}}$	0.017 $_{09^{-}\,11^{+}}$
Gemma–8-bit	0.088 $_{55^{-}\,51^{+}}$	0.008 $_{04^{-}\,06^{+}}$	0.041 $_{26^{-}\,24^{+}}$	0.046 $_{27^{-}\,29^{+}}$	0.009 $_{06^{-}\,04^{+}}$
Ministral–4-bit	**0.813** $_{454^{-}\,522^{+}}$	0.489 $_{305^{-}\,281^{+}}$	0.539 $_{307^{-}\,339^{+}}$	0.632 $_{402^{-}\,356^{+}}$	0.493 $_{287^{-}\,305^{+}}$
Ministral–8-bit	0.093 $_{58^{-}\,54^{+}}$	0.039 $_{22^{-}\,24^{+}}$	0.068 $_{43^{-}\,39^{+}}$	0.056 $_{33^{-}\,35^{+}}$	0.003 $_{03^{-}\,01^{+}}$
Qwen–4-bit	**0.783** $_{446^{-}\,494^{+}}$	0.523 $_{333^{-}\,295^{+}}$	0.662 $_{385^{-}\,409^{+}}$	0.704 $_{452^{-}\,392^{+}}$	0.388 $_{224^{-}\,242^{+}}$
Qwen–8-bit	0.102 $_{65^{-}\,57^{+}}$	0.028 $_{16^{-}\,18^{+}}$	0.032 $_{20^{-}\,18^{+}}$	0.045 $_{26^{-}\,28^{+}}$	0.005 $_{04^{-}\,02^{+}}$

Table 2: Power AUCs. All confidence interval ranges are reported in $\times 10^{-4}$.

Baselines. We compare SCORE with Maximum Mean Discrepancy (MMD) (Gao et al., 2025) and PTS (Shi et al., 2026). We also include the RUT and its Kolmogorov–Smirnov (KS) baseline (Zhu et al., 2026). More details on the baselines and experimental settings are provided in Appendix C.

5.2 Auditing Model Quantization

We consider the setting in which an API provider substitutes the claimed model with a quantized variant. We evaluate Llama-3.1-70B-Instruct Grattafiori et al. (2024), Ministral-3-14B-Instruct Liu et al. (2026), Gemma-3-27B-it Team (2025), and Qwen2.5-72B-Instruct Yang et al. (2024), comparing each model against its 4-bit and 8-bit quantized variants. As shown in Table 2, all baselines exhibit limited power against 8-bit substitutions, whereas SCORE retains a clear detection advantage. Across the remaining settings, SCORE consistently outperforms all baselines, including RUT, demonstrating greater sensitivity to quantization-induced shifts in the output distribution. Full statistical-power curves and corresponding power AUCs are provided in Appendix D.1.

5.3 Auditing Supervised Fine-Tuning

We consider the setting in which an API provider serves a model fine-tuned on instruction-following data. We fine-tune Llama-3.1-8B-Instruct, Ministral-3-8B-Instruct, and Qwen2.5-7B-Instruct on both benign and harmful datasets. Alpaca (Taori et al., 2023) provides benign instruction-following data, whereas BeaverTails (Ji et al., 2023) provides harmful question-answering examples. For each dataset, we sample 1,000 examples and fine-tune every model for five epochs. Fine-tuning uses LoRA (Hu et al., 2021) with rank $r = 64$, scaling factor $\alpha_{\text{LoRA}} = 16$, batch size 32, and learning rate 1×10^{-4} on NVIDIA H100 GPUs. For each checkpoint, we compute the power AUC of every detection method. As shown in Figure 2, SCORE achieves a higher power AUC than all baselines across every fine-tuning configuration. Notably, SCORE detects behavioral changes at the first-epoch checkpoint, demonstrating sensitivity to early fine-tuning-induced distribution shifts. Although all methods improve with additional epochs, SCORE retains the highest power AUC across models and datasets. Full power AUCs are provided in Appendix D.2.

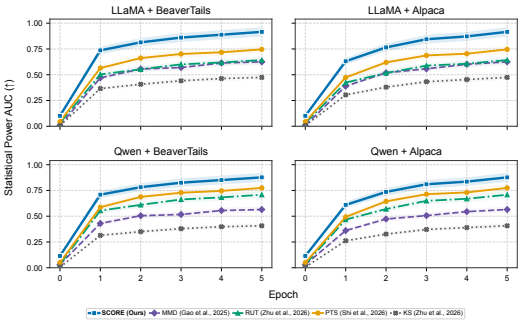


Figure 2: Power AUC of SFT across epochs.

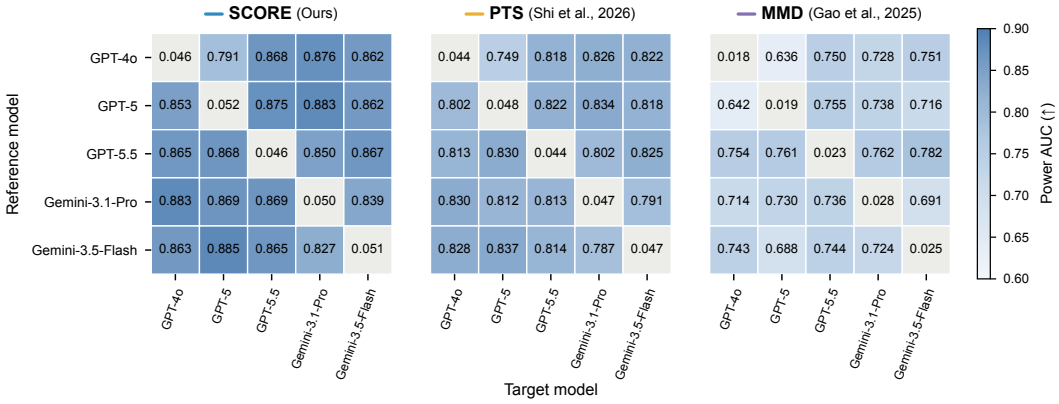


Figure 3: Power AUC for detecting full model replacement. Each cell shows the power AUC between reference and target endpoints. Diagonal values represent self-comparisons.

5.4 Auditing Full Model Replacement

We consider the setting in which an API provider replaces the claimed model with a different model. We first conduct pairwise comparisons among five open-weight models: Llama-3.1-70B-Instruct, Ministral-3-8B-Instruct, Qwen2.5-7B/72B-Instruct, and Gemma-3-27B-it. We then extend the evaluation to live commercial APIs from the GPT and Gemini families. The claimed model's official provider endpoint serves as the reference endpoint, while another proprietary model's endpoint

serves as the target endpoint. This setup evaluates cross-model substitution under strict output-only access. RUT and KS are inapplicable because their statistics require reference-side token probabilities or rankings, which output-only commercial APIs do not expose. We therefore compare SCORE only against MMD and PTS, which satisfy the same output-only access constraint. As shown in Figure 3, SCORE achieves the highest power AUC (AUC_q) across all evaluated open-weight model pairs and commercial API pairs, outperforming every applicable baseline. These results show that SCORE reliably detects distributional shifts caused by full model replacement and generalizes to live commercial endpoints. Full statistical-power curves and pairwise AUC_q results are provided in Appendix D.3.

5.5 Auditing Robustness Across Query Domains

We further evaluate the robustness of SCORE to shifts in the probe-text distribution. Beyond WildChat Zhao et al. (2024), which approximates general conversational traffic, we consider BigCodeBench (Zhuo et al., 2025) for code generation, MATH (Hendrycks et al., 2021) for mathematical reasoning, and multilingual Wikipedia spanning English, German, Spanish, French, and Russian. We retain the quantization setup from Section 5.2 and vary only the source of the audit prompts, thereby isolating the effect of probe-text distribution on detection performance. As quantization often induces subtle response-distribution shifts, this setting provides a stringent test of cross-domain robustness. As shown in Table 3, SCORE maintains consistent detection performance across conversational, code, mathematical, and multilingual queries, following the trends observed in Section 5.2. In the more challenging 8-bit setting, statistical power decreases for all methods, but SCORE still achieves the highest or near-highest power. Across the remaining settings, SCORE consistently outperforms MMD, RUT, and KS. These results demonstrate that SCORE is not specific to the WildChat probe-text distribution and generalizes effectively to code, mathematics, and multilingual text. Full results are provided in Appendix D.4. A temperature-sensitivity analysis for SCORE is reported in Appendix E.

(a) BigCodeBench Zhuo et al. (2025)

Model	SCORE	MMD	RUT	PTS	KS
Llama–4-bit	**0.846** $^{462^-\ 554^+}$	0.353 $^{237^-\ 187^+}$	0.593 $^{303^-\ 409^+}$	0.689 $^{454^-\ 372^+}$	0.524 $^{270^-\ 358^+}$
Llama–8-bit	0.294 $^{197^-\ 155^+}$	0.143 $^{73^-\ 99^+}$	0.093 $^{62^-\ 50^+}$	0.128 $^{66^-\ 88^+}$	0.021 $^{14^-\ 12^+}$
Gemma–4-bit	**0.576** $^{141^-\ 191^+}$	0.170 $^{112^-\ 92^+}$	0.136 $^{71^-\ 93^+}$	0.397 $^{129^-\ 107^+}$	0.013 $^{07^-\ 09^+}$
Gemma–8-bit	0.103 $^{68^-\ 56^+}$	0.052 $^{27^-\ 35^+}$	0.048 $^{32^-\ 26^+}$	0.054 $^{28^-\ 36^+}$	0.001 $^{02^-\ 01^+}$
Ministral–4-bit	**0.725** $^{374^-\ 496^+}$	0.339 $^{221^-\ 185^+}$	0.480 $^{253^-\ 323^+}$	0.564 $^{389^-\ 287^+}$	0.190 $^{103^-\ 125^+}$
Ministral–8-bit	0.084 $^{35^-\ 29^+}$	0.056 $^{30^-\ 38^+}$	0.040 $^{28^-\ 20^+}$	0.033 $^{18^-\ 22^+}$	0.011 $^{08^-\ 06^+}$
Qwen–4-bit	**0.671** $^{355^-\ 451^+}$	0.448 $^{309^-\ 229^+}$	0.567 $^{306^-\ 374^+}$	0.603 $^{413^-\ 311^+}$	0.333 $^{182^-\ 218^+}$
Qwen–8-bit	0.119 $^{82^-\ 60^+}$	0.033 $^{18^-\ 22^+}$	0.037 $^{25^-\ 19^+}$	0.053 $^{29^-\ 35^+}$	0.006 $^{05^-\ 03^+}$

(b) MATH Hendrycks et al. (2021)

Model	SCORE	MMD	RUT	PTS	KS
Llama–4-bit	**0.736** $^{508^-\ 376^+}$	0.454 $^{245^-\ 299^+}$	0.516 $^{353^-\ 267^+}$	0.598 $^{327^-\ 391^+}$	0.463 $^{311^-\ 245^+}$
Llama–8-bit	0.130 $^{448^-\ 548^+}$	0.084 $^{125^-\ 95^+}$	0.063 $^{144^-\ 172^+}$	0.062 $^{243^-\ 191^+}$	0.023 $^{63^-\ 85^+}$
Gemma–4-bit	**0.633** $^{433^-\ 327^+}$	0.249 $^{136^-\ 162^+}$	0.312 $^{209^-\ 165^+}$	0.452 $^{230^-\ 312^+}$	0.062 $^{41^-\ 33^+}$
Gemma–8-bit	0.099 $^{54^-\ 64^+}$	0.064 $^{43^-\ 33^+}$	0.046 $^{24^-\ 32^+}$	0.052 $^{34^-\ 28^+}$	0.002 $^{01^-\ 02^+}$
Ministral–4-bit	**0.559** $^{375^-\ 295^+}$	0.222 $^{113^-\ 153^+}$	0.371 $^{245^-\ 201^+}$	0.435 $^{224^-\ 298^+}$	0.165 $^{108^-\ 90^+}$
Ministral–8-bit	0.077 $^{39^-\ 53^+}$	0.057 $^{37^-\ 31^+}$	0.056 $^{29^-\ 39^+}$	0.046 $^{31^-\ 25^+}$	0.018 $^{10^-\ 12^+}$
Qwen–4-bit	**0.617** $^{407^-\ 333^+}$	0.412 $^{212^-\ 282^+}$	0.522 $^{341^-\ 285^+}$	0.555 $^{293^-\ 373^+}$	0.306 $^{212^-\ 156^+}$
Qwen–8-bit	0.116 $^{60^-\ 80^+}$	0.032 $^{21^-\ 17^+}$	0.036 $^{19^-\ 25^+}$	0.051 $^{36^-\ 26^+}$	0.006 $^{03^-\ 05^+}$

Table 3: Power AUC for detecting quantized variants across query domains. **Bold** denotes the best method; gray denotes none reliable. All confidence interval ranges are reported in $\times 10^{-4}$.

5.6 Auditing Commercial APIs

We apply SCORE to 31 commercial inference endpoints from nine providers, covering four Meta Llama models. For each target endpoint, Meta's full-precision and FP16 weights define two candidate reference configurations. Let P_1 and P_2 denote their response distributions, and let Q denote the target distribution. Following Gao et al. (2025), we test

$$\begin{aligned} H_0 &: \quad Q = P_1 \ \text{ or } \ Q = P_2, \\ H_1 &: \quad Q \neq P_1 \ \text{ and } \ Q \neq P_2. \end{aligned} \tag{18}$$

Running SCORE on $(\mathcal{D}_{P_1}, \mathcal{D}_Q)$ and $(\mathcal{D}_{P_2}, \mathcal{D}_Q)$ yields p-values p_1 and p_2, respectively. The composite decision rule is

$$\delta(\mathcal{D}_{P_1}, \mathcal{D}_{P_2}, \mathcal{D}_Q) = \mathbb{I}[p_1 \leq \alpha \wedge p_2 \leq \alpha] = \mathbb{I}[\max\{p_1, p_2\} \leq \alpha]\,. \tag{19}$$

Table 4: Audit results for 31 API endpoints spanning nine providers and four language models. ✗ indicates that an endpoint failed the audit (rejection rate $\geq 50\%$).

Provider	Wikipedia				HumanEval		UltraChat	
	3 8B	3.1 8B	3 70B	3.1 70B	3 8B	3.1 8B	3 8B	3.1 8B
Amazon	✓	✗	✓	✗	✓	✗	✗	✗
Openlux	✓	✓	✗	✗	✓	✓	✓	✓
Azure	✓	✓	✓	✓	✓	✓	✓	✓
DeepInfra	✓	✓	✓	✓	✓	✓	✓	✓
Fireworks	✓	✓	✓	✓	✗	✗	✓	✗
Groq	✓	✓	✗	✗	✓	✗	✓	✓
Perplexity	✗	✗	✗	✗	✗	✗	✗	✗
Replicate	✓	—	✗	—	✓	—	✓	—
Together	✓	✓	✓	✓	✓	✓	✓	✓

This rule rejects H_0 only when both component tests are significant and controls the type I error at α. For example, when $Q = P_1$,

$$\Pr_{Q=P_1}[\delta = 1] \leq \Pr_{Q=P_1}[p_1 \leq \alpha] \leq \alpha, \tag{20}$$

and the same argument applies when $Q = P_2$. We audit each endpoint using Wikipedia, HumanEval, and UltraChat queries. Wikipedia uses 25 prompts with a 50-token generation limit. HumanEval and UltraChat each use 20 prompts with a 250-token limit and are evaluated only on 8B models. All endpoints are sampled at $\mathrm{temperature} = 1$, with $N = 10m$ completions per audit-prompt distribution, where m is the number of prompts. Following the decision protocol of Gao et al. (2025), we repeat each test ten times at $\alpha = 0.01$ and flag an endpoint–query pair as inconsistent when its mean rejection rate is at least 0.5. As shown in Table 4, 15 of the 31 endpoints are inconsistent with both reference configurations. Amazon and Perplexity contain the most flagged endpoints, with every evaluated Perplexity configuration rejected. In contrast, Azure AI Studio, DeepInfra, and Together AI yield no rejections. HumanEval and UltraChat detect more inconsistencies than Wikipedia, showing that audit outcomes depend on the audit-prompt distribution. Overall, commercial endpoints advertised as serving the same model can exhibit statistically detectable behavioral differences, motivating audits across multiple audit-prompt distributions.

6 Conclusion

The increasing reliance on opaque commercial APIs for LLM deployment makes testing prompt–response joint-distribution equality essential for service reliability and trust. We introduce SCORE, which formulates LLM API auditing under strict output-only access to both endpoints as a model equality testing problem. By combining semantic-contrast scores with reference-calibrated randomized ranks, SCORE reduces endpoint comparison to a uniformity test with type I error controlled at level α. Experiments spanning model quantization, supervised fine-tuning, full model replacement, cross-domain queries, and real commercial APIs show that SCORE effectively detects response-distribution shifts and achieves stronger overall statistical power than existing baselines.

Limitations and Future Work SCORE has not been systematically evaluated against adaptive providers that know the auditing procedure and may selectively route audit queries to the claimed model. Moreover, its calibration assumes that the reference endpoint remains stable and trustworthy. Promising directions include more reference-efficient calibration, time-varying audits under rolling model updates, broader query probes, improved statistical power, and extensions to other modalities such as image generation. Overall, SCORE provides a principled framework for model equality testing under strict output-only access and enables reliable and scalable auditing of increasingly cloud-hosted and proprietary LLM services.

AI USE STATEMENT

We did not use generative AI tools for generating synthetic data, developing theoretical models or conceptual frameworks, formulating mathematical claims or proofs, proposing or refining hypotheses, designing the methodology or experiments, implementing methods, or interpreting results. Generative AI tools were used only to assist with manuscript writing and language polishing. All AI-assisted text was carefully reviewed and verified by the authors to ensure that it accurately reflects the intended meaning. The authors take full responsibility for the final content of this work.

REFERENCES

Saeif Alhazbi, Ahmed Hussain, Gabriele Oligeri, and Panos Papadimitratos. Llms have rhythm: Fingerprinting large language models using inter-token times and network traffic analysis. *IEEE Open Journal of the Communications Society*, 6:5050–5071, 2025.

Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

Will Cai, Tianneng Shi, Xuandong Zhao, and Dawn Song. Are you getting what you pay for? auditing model substitution in llm apis. *arXiv preprint arXiv:2504.04715*, 2025.

Lingjiao Chen, Matei Zaharia, and James Zou. How is chatgpt's behavior changing over time? *arXiv preprint arXiv:2307.09009*, 2023.

Harald Cramér. On the composition of elementary errors: First paper: Mathematical deductions. *Scandinavian Actuarial Journal*, 1928(1):13–74, 1928.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3029–3051, 2023.

Sabri Eyuboglu, Karan Goel, Arjun Desai, Lingjiao Chen, Mathew Monfort, Chris Ré, and James Zou. Model changelists: Characterizing updates to ml models. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, pp. 2432–2453, 2024.

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*, 2026.

Zhiyuan Fu, Junfan Chen, Lan Zhang, Ting Yang, Jun Niu, Hongyu Sun, Ruidong Li, Peng Liu, Jice Wang, Fannv He, et al. Fdllm: A dedicated detector for black-box llms fingerprinting. In *2025 IEEE 24th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pp. 1374–1379. IEEE, 2025.

Irena Gao, Percy Liang, and Carlos Guestrin. Model equality testing: Which model is this api serving? In *International Conference on Learning Representations*, volume 2025, pp. 86369–86382, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Chenchen Gu, Xiang Lisa Li, Rohith Kuditipudi, Percy Liang, and Tatsunori Hashimoto. Auditing prompt caching in language model APIs. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=gUj2fxQcLZ.

Yanpei Guo, Wenjie Qu, Linyu Wu, Shengfang Zhai, Lionel Z Wang, Ming Xu, Yue Liu, Binhang Yuan, Dawn Song, and Jiaheng Zhang. Immaculate: A practical llm auditing framework via verifiable computation. *arXiv preprint arXiv:2602.22700*, 2026.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Yangsibo Huang, Milad Nasr, Anastasios Angelopoulos, Nicholas Carlini, Wei-Lin Chiang, Christopher A Choquette-Choo, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Ken Ziyu Liu, et al. Exploring and mitigating adversarial manipulation of voting-based leaderboards. *arXiv preprint arXiv:2501.07493*, 2025.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36:24678–24704, 2023.

Adam Karvonen, Daniel Reuter, Roy Rinberg, Luke Marks, Adrià Garriga-Alonso, and Keri Warr. DiFR: Inference verification despite nondeterminism. In *ICML Workshop on Technical AI Governance Research*, 2026. URL https://openreview.net/forum?id=OtKAcP46lB.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International conference on machine learning*, pp. 17061–17084. PMLR, 2023.

Haorui Li, Zhenghui He, Xuanzi Liu, Yang Xu, Dongsheng Liu, Jiakang Ma, Lupan Wu, Yangjie Wu, Xiongchao Tang, and Tianhui Shi. When is the same model not the same service? a measurement study of hosted open-weight llm apis. *arXiv preprint arXiv:2605.02821*, 2026.

Guanjie Lin, Yinxin Wan, Shichao Pei, Ting Xu, Kuai Xu, and Guoliang Xue. Behavioral consistency and transparency analysis on large language model api gateways. *arXiv preprint arXiv:2604.21083*, 2026.

Alexander H Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026.

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

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

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Jie Ren, Han Xu, Yiding Liu, Yingqian Cui, Shuaiqiang Wang, Dawei Yin, and Jiliang Tang. A robust semantics-based watermark for large language model against paraphrasing. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 613–625, 2024.

Shuo Ren, Can Xie, Pu Jian, Zhenjiang Ren, Chunlin Leng, and Jiajun Zhang. Towards scientific intelligence: A survey of llm-based scientific agents. *arXiv preprint arXiv:2503.24047*, 2025.

Gurvan Richardeau, Gohar Dashyan, Erwan Le Merrer, and Gilles Trédan. Flips: Instance-fingerprinting for llms via pseudo-random sequences. *arXiv preprint arXiv:2606.03330*, 2026.

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

Kai-Yuan Shi, Fang-Qi Li, Zhuo-Meng Zhang, and Shi-Lin Wang. Model equality testing of black-box llm apis via prefix tree statistics. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2431–2435. IEEE, 2026.

Mingjie Sun, Yida Yin, Zhiqiu Xu, J Zico Kolter, and Zhuang Liu. Idiosyncrasies in large language models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=FCZ3jVzmTZ.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.

Gemma Team. Gemma 3. 2025. URL https://goo.gle/Gemma3Report.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024.

Xiangfan Wu, Zonghao Ying, Huiyu Wu, Xing Zheng, Huangsheng Cheng, Xiaorong Shi, and Jing Guo. Ventor-qtest: Threat-model-driven verification of vendor-hosted llm apis. *arXiv preprint arXiv:2608.16391*, 2026.

Jiashu Xu, Fei Wang, Mingyu Ma, Pang Wei Koh, Chaowei Xiao, and Muhao Chen. Instructional fingerprinting of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3277–3306, 2024.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Yuewei Zhang, Zhi-Hai Zhang, and Hanzhang Qin. Which model is actually serving you? iris: Budgeted black-box auditing of model substitution and routing dilution in llm gateways. *arXiv preprint arXiv:2607.20860*, 2026.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m chatgpt interaction logs in the wild. In *International Conference on Learning Representations*, volume 2024, pp. 34590–34605, 2024.

Jijie Zhou, Niloofar Mireshghallah, and Tianshi Li. Operationalizing data minimization for privacy-preserving llm prompting. In *International Conference on Learning Representations*, volume 2026, pp. 56919–56943, 2026.

Xiaoyuan Zhu, Yaowen Ye, Tianyi Qiu, Hanlin Zhu, Sijun Tan, Ajraf Mannan, Jonathan Michala, Raluca Popa, and Willie Neiswanger. Auditing black-box llm apis with a rank-based uniformity test. In *International Conference on Learning Representations*, volume 2026, pp. 54232–54261, 2026.

Terry Yue Zhuo, Minh Chien Vu, Jenny Chim, Han Hu, Wenhao Yu, Ratnadira Widyasari, Imam Nur Bani Yusuf, Haolan Zhan, Junda He, Indraneil Paul, et al. Bigcodebench: Benchmarking code generation with diverse function calls and complex instructions. In *International Conference on Learning Representations*, volume 2025, pp. 66602–66656, 2025.

A Algorithm

This section presents the computational procedures underlying score-function evaluation and SCORE. Algorithm 1 estimates the prompt-averaged AUROC of each candidate score function, providing a prompt-balanced measure of its ability to distinguish target and reference outputs. Algorithm 2 specifies the complete SCORE audit, including semantic-contrast scoring, randomized-rank calibration, and the final uniformity test. Together, these algorithms formalize the score-evaluation and model-auditing stages of our framework.

Algorithm 1: Estimating prompt-averaged AUROC for candidate score functions

Input : Prompt corpus $\mathcal{D}$; reference and target response distributions π_{ref} and π_{tgt}; decoding configuration $\phi = (\tau = 0.5, \ell_{\max} = 50)$; number of prompts $n = 100$; number of completions per prompt and endpoint $m = 20$; candidate scalar score functions $\Delta = \{\delta_k : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}\}_{k=1}^{K}$.

Output Empirical prompt-averaged AUROC $\widehat{\mu}_{\mathrm{AUROC}}(\delta_k)$ for every $\delta_k \in \Delta$.

:

Sample n distinct prompts $x_1, \ldots, x_n$ uniformly without replacement from $\mathcal{D}$;

for $i \leftarrow 1$ **to** n **do**

  $y_{\mathrm{ref},i}^{(1)}, \ldots, y_{\mathrm{ref},i}^{(m)} \overset{\text{i.i.d.}}{\sim} \pi_{\mathrm{ref}}(\cdot \mid x_i; \phi)$;

  $y_{\mathrm{tgt},i}^{(1)}, \ldots, y_{\mathrm{tgt},i}^{(m)} \overset{\text{i.i.d.}}{\sim} \pi_{\mathrm{tgt}}(\cdot \mid x_i; \phi)$;

  $\mathbf{y}_i \leftarrow \mathrm{concat}\Big((y_{\mathrm{ref},i}^{(j)})_{j=1}^{m}, (y_{\mathrm{tgt},i}^{(j)})_{j=1}^{m}\Big)$;

  $\mathbf{z}_i \leftarrow \Big(\underbrace{0, \ldots, 0}_{m}, \underbrace{1, \ldots, 1}_{m}\Big)$;

  for $k \leftarrow 1$ **to** K **do**

    $\mathbf{s}_{ik} \leftarrow (\delta_k(x_i, [\mathbf{y}_i]_r))_{r=1}^{2m}$;

    $A_{ik} \leftarrow \mathrm{AUROC}(\mathbf{s}_{ik}, \mathbf{z}_i)$;

for $k \leftarrow 1$ **to** K **do**

  $\widehat{\mu}_{\mathrm{AUROC}}(\delta_k) \leftarrow \dfrac{1}{n} \sum_{i=1}^{n} A_{ik}$;

Note. $\mathrm{AUROC}(\mathbf{s}, \mathbf{z})$ denotes the empirical binary area under the receiver operating characteristic curve (Bradley, 1997), with target-endpoint responses treated as the positive class. AUROC is computed separately for each prompt and then macro-averaged across prompts.

Algorithm 2: SCORE model equality test

Input : Probe distribution ξ; reference and target endpoints π_{ref} and π_{tgt}; fixed prompt and score maps; decoding configuration ϕ; $n = 100, m = 20, K = 5, B = 9999$, and $\alpha = 0.05$, where $(B+1)^{-1} \le \alpha$.

Output Monte Carlo p-value $\widehat{p}$ and decision $\delta_\alpha \in \{0, 1\}$, where $\delta_\alpha = 1$ rejects H_0.

:

Draw $x_1, \ldots, x_n \overset{\text{i.i.d.}}{\sim} \xi$

for $i = 1, \ldots, n$ **do**

  Construct $\{q_{i\ell}^{o}, q_{i\ell}^{p}\}_{\ell=2}^{L_i}$ using Eq. equation 7

  Using fresh, mutually independent response blocks, compute S_i^{tgt} and $\{S_{ij}^{\mathrm{ref}}\}_{j=1}^{m}$ using Eqs. equation 8–equation 11 and Eq. equation 12

  $C_i^{<} \leftarrow \sum_{j=1}^{m} \mathbb{I}[S_{ij}^{\mathrm{ref}} < S_i^{\mathrm{tgt}}], \quad C_i^{=} \leftarrow \sum_{j=1}^{m} \mathbb{I}[S_{ij}^{\mathrm{ref}} = S_i^{\mathrm{tgt}}]$

  Draw $J_i \sim \mathrm{Unif}\{0, \ldots, C_i^{=}\}$ and $U_i \sim \mathrm{Unif}[0, 1]$ independently

  Compute R_i using Eq. equation 13

Order $R_{1:n}$ and compute ω_{obs}^2 using Eq. equation 15

for $b = 1, \ldots, B$ **do**

  Draw $V_{b,1}, \ldots, V_{b,n} \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}[0, 1]$

  Order $V_{b,1:n}$ and compute ω_b^2 using Eq. equation 15, replacing $R_{(i)}$ with $V_{b,(i)}$

Compute $\widehat{p}$ using Eq. equation 16

$\delta_\alpha \leftarrow \mathbb{I}[\widehat{p} \le \alpha]$

return $(\widehat{p}, \delta_\alpha)$

B Further Related Work

LLM fingerprinting. LLM fingerprinting infers model identity, provenance, or ownership from characteristic behavior. Active approaches either implant keyed behaviors through instruction tuning or output watermarking (Xu et al., 2024; Kirchenbauer et al., 2023; Ren et al., 2024), or use non-invasive probes designed to elicit model-specific responses (Pasquini et al., 2025). Passive approaches instead extract naturally occurring signatures from generated text or serving dynamics, including learned textual fingerprints, lexical idiosyncrasies, and inter-token timing (Fu et al., 2025; Sun et al., 2025; Alhazbi et al., 2025). Recent work systematizes this space, studies robustness to post-training transformations and adaptive spoofing, and extends attribution from model families to deployed model instances (Shao et al., 2025; Nasery et al., 2025; Richardeau et al., 2026). Although both fingerprinting and model equality testing rely on observable behavior, they address different inferential targets. Fingerprinting assigns an unknown endpoint to a source and may favor signatures that persist across deployment changes. Model equality testing instead asks whether a target endpoint and a reference endpoint induce the same observable prompt–response joint distribution under a prespecified audit setting. Accordingly, SCORE neither implants a fingerprint nor trains a provenance classifier, but directly tests equality between the target endpoint and the reference endpoint.

C Baseline Details

Baselines.

Maximum Mean Discrepancy (MMD). MMD (Gao et al., 2025) applies a kernel two-sample test to completions sampled from the target and reference endpoints. It uses a character-level Hamming kernel to measure discrepancies between their empirical output distributions.

Prefix Tree Statistics (PTS). PTS (Shi et al., 2026) summarizes surface-form sequence counts using a prefix-tree statistic, with individual contributions weighted by their observed frequencies. The resulting statistic is used to test API consistency.

Rank-based Uniformity Test (RUT). RUT (Zhu et al., 2026) computes response-level log-rank scores using token rankings from the reference model. It converts each target score into a reference-calibrated randomized rank and tests whether the resulting ranks follow a uniform distribution.

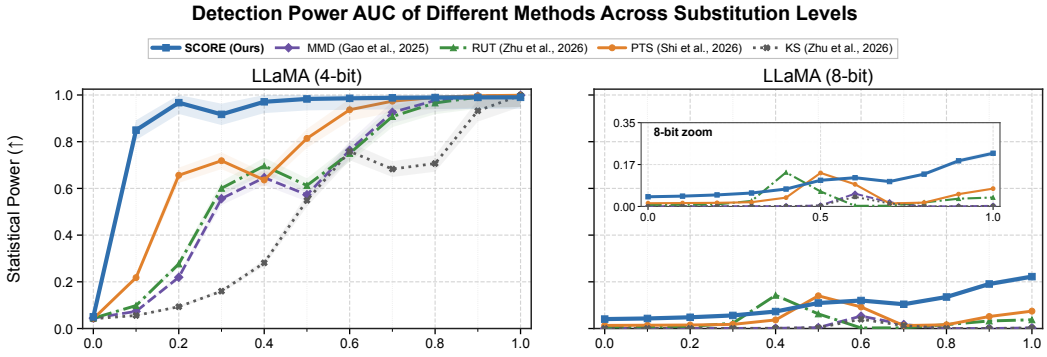
Kolmogorov–Smirnov (KS). The KS baseline (Zhu et al., 2026) uses the same reference-model log-rank scores as RUT but does not perform randomized-rank calibration. Instead, it directly applies a two-sample KS test to the target and reference score distributions. Because RUT and KS require reference-side token rankings, we treat both as privileged-reference baselines.

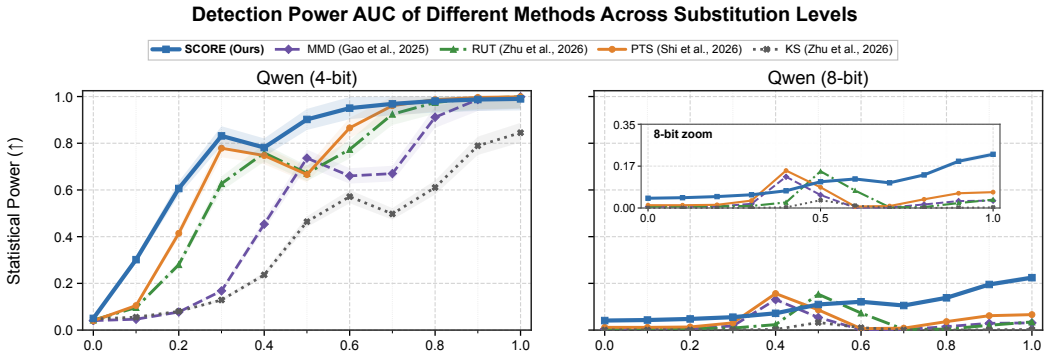
Evaluation protocol. All methods use a sampling temperature of 0.5 and a maximum generation length of 50 tokens. Reference-model inference is performed in bfloat16 using NVIDIA H100 80GB GPUs under a fixed software environment. For each substitution probability q, we collect samples according to the corresponding test protocol and reject H_0 when its p-value is at most $\alpha = 0.05$. We repeat this procedure over 500 independent Monte Carlo trials and estimate statistical power by the proportion of trials that reject H_0 at significance level $\alpha = 0.05$.

D Further details of the experimental results

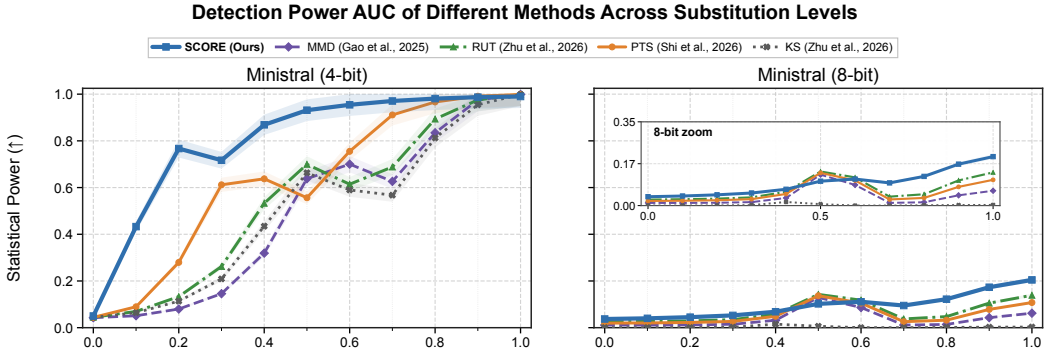
D.1 Full Statistical Power Curves for Auditing Model Quantization

We present the full statistical-power curves underlying the quantization results in Section 5.2. We evaluate Llama-3.1-70B-Instruct Grattafiori et al. (2024), Ministral-3-14B-Instruct Liu et al. (2026), Gemma-3-27B-it Team (2025), and Qwen2.5-72B-Instruct Yang et al. (2024) against their respective 4-bit and 8-bit quantized variants. For each model pair, we simulate probabilistic substitution by independently routing each target request to the quantized variant with probability $q \in [0, 1]$ and to the original model otherwise. Thus, $q = 0$ corresponds to no substitution, whereas $q = 1$ corresponds to complete substitution. The curves compare SCORE with MMD, PTS, RUT, and KS under



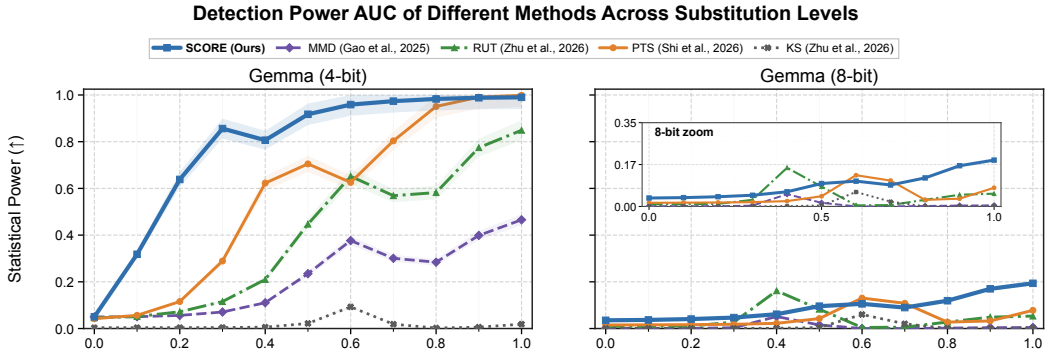


the common evaluation protocol. Pointwise uncertainty is reported using 95% Wilson confidence intervals. Integrating each power curve over q yields the power AUC values, with 95% confidence intervals estimated by bootstrap. The complete curves show that 8-bit substitution produces weaker audit signals than 4-bit substitution across all methods. Nevertheless, SCORE maintains consistently stronger detection power across the evaluated models and quantization levels.



D.2 Full Statistical Power Curves for Auditing Supervised Fine-Tuning

We present the full statistical-power curves underlying the supervised fine-tuning results in Section 5.3. We evaluate Llama-3.1-8B-Instruct, Ministral-3-8B-Instruct, and Qwen2.5-7B-Instruct after fine-tuning on Alpaca (Taori et al., 2023) and BeaverTails (Ji et al., 2023), representing benign and harmful instruction-following data, respectively. For each dataset, we fine-tune every model on 1,000 examples for five epochs using LoRA (Hu et al., 2021), with rank 64, a scaling parameter of 16, batch size 32, and learning rate 1×10^{-4}. Each epoch checkpoint is treated as a separate alternative to the corresponding base model. For every model–checkpoint pair, we simulate probabilistic substitution by routing each target request to the fine-tuned model with probability $q \in [0, 1]$ and



to the base model otherwise. The curves compare SCORE with MMD, PTS, RUT, and KS under the common evaluation protocol. Pointwise uncertainty is reported using 95% Wilson confidence intervals. Integrating each curve over q yields the power AUC values reported in the main paper, with 95% confidence intervals estimated by bootstrap. The complete curves show that SCORE detects behavioral changes from the first fine-tuning epoch and maintains consistently stronger detection power across models, datasets, and subsequent checkpoints.

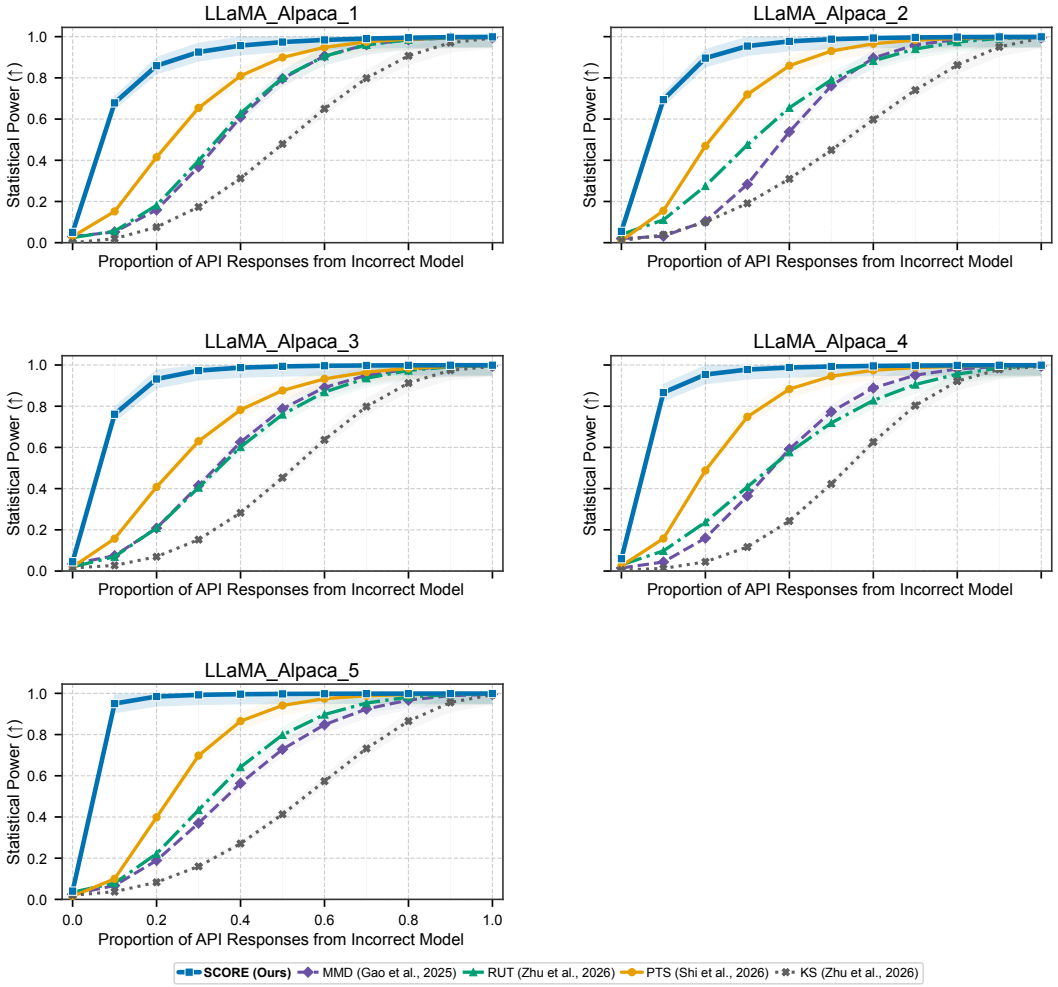


Figure 4: Full Statistical Power Curves for Detecting SFT on Llama.

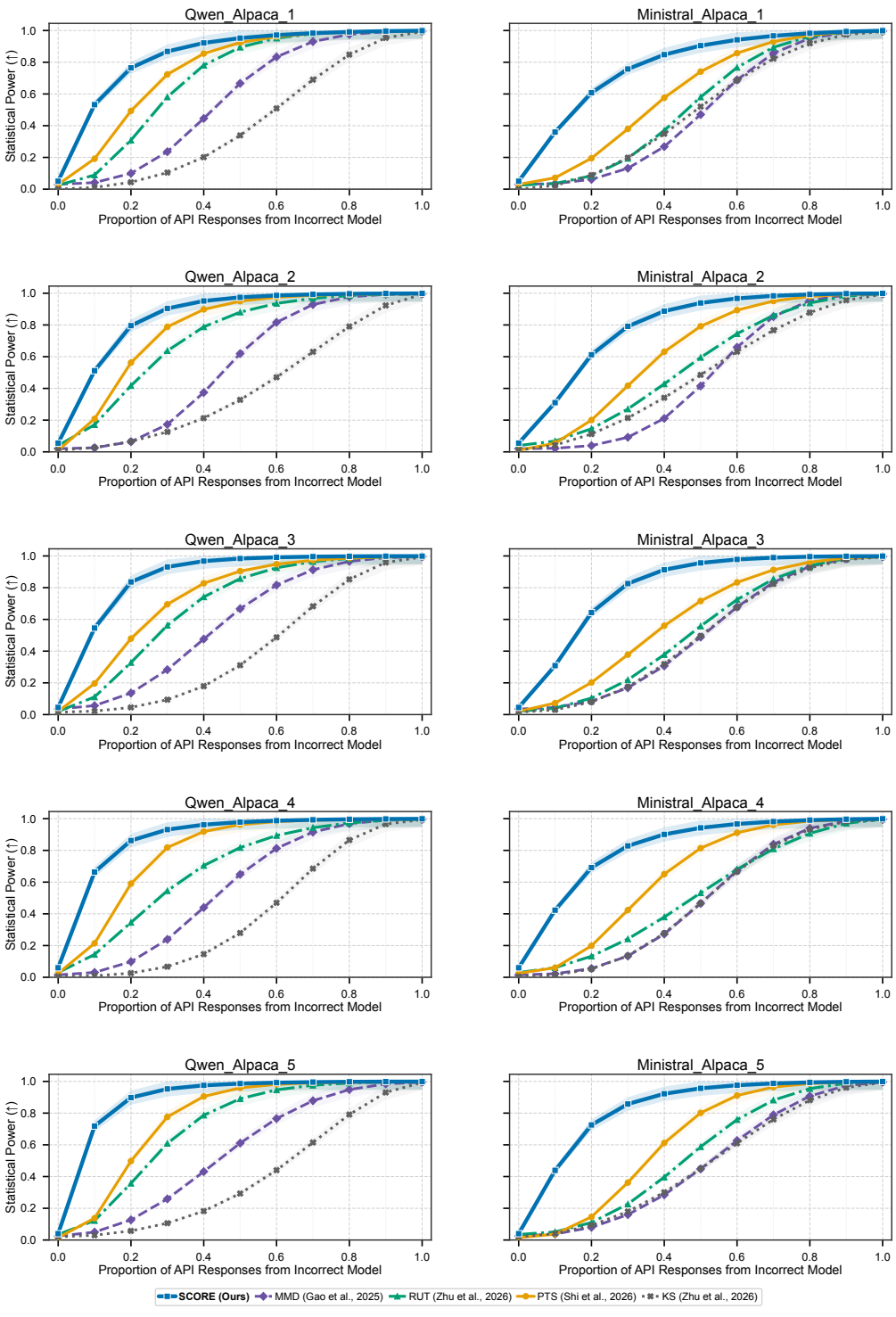


Figure 5: Full Statistical Power Curves for Detecting SFT across models and datasets.

D.3 Additional Experimental Results for Auditing Full Model Replacement

We present the full statistical-power curves underlying the model-replacement results in Section 5.4. For open-weight models, we evaluate pairwise substitutions among Llama-3.1-70B-Instruct,

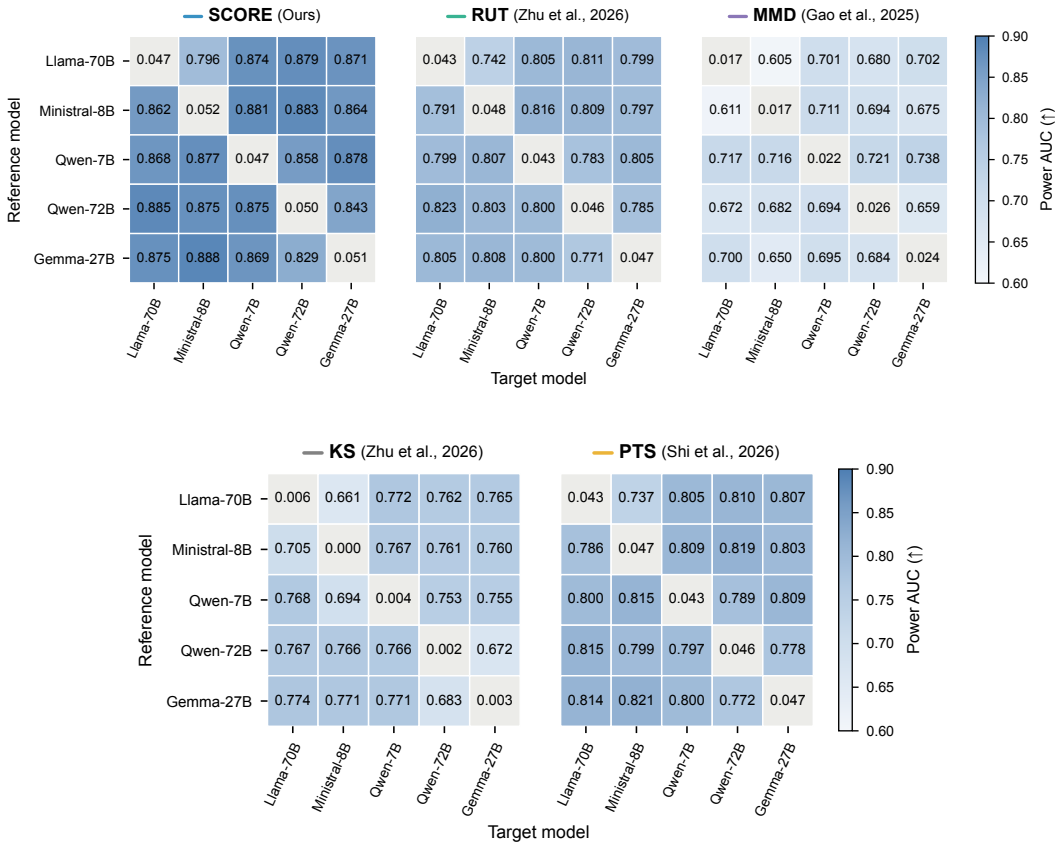


Figure 6: Power AUC for detecting full model replacement. Each cell shows the power AUC between reference and target endpoints. Diagonal values represent self-comparisons.

Ministral-3-8B-Instruct, Qwen2.5-7B-Instruct, Qwen2.5-72B-Instruct, and Gemma-3-27B-it. For each evaluated pair, one model serves as the claimed model and the other as the alternative model. Probabilistic substitution is simulated by routing each target request to the alternative model with probability $q \in [0, 1]$ and to the claimed model otherwise. Thus, the curves characterize detection power from partial routing to full model replacement. For open-weight models, we compare SCORE with MMD, PTS, RUT, and KS. Because RUT and KS require reference-side token rankings, the commercial-API settings include only SCORE, MMD, and PTS. Pointwise uncertainty is reported using 95% Wilson confidence intervals. Integrating each curve over q yields the power AUC values reported in the main paper, with 95% confidence intervals estimated by bootstrap. Across the evaluated open-weight and commercial model pairs, SCORE achieves the strongest overall detection power and consistently identifies the distributional shifts introduced by full model replacement.

D.4 Full Statistical Power Curves for Auditing Robustness Across Query Domains

We present the full statistical-power curves underlying the cross-domain results in Section 5.5. Across query domains, detection power generally increases with the substitution probability, while SCORE maintains a stronger power profile than the baselines over most of the evaluated range. This advantage is most pronounced for 4-bit quantization, where SCORE reaches high detection power more consistently. Although the subtler distributional shifts induced by 8-bit quantization reduce the power of all methods, SCORE remains the strongest or among the strongest across domains. Additional evaluations on MMLU-Pro Wang et al. (2024) and GPQA Rein et al. (2023) show the same pattern, with SCORE outperforming all existing baselines on both reasoning benchmarks. These trends yield higher power AUCs and demonstrate that the detection advantage of SCORE generalizes beyond any single probe-text distribution.

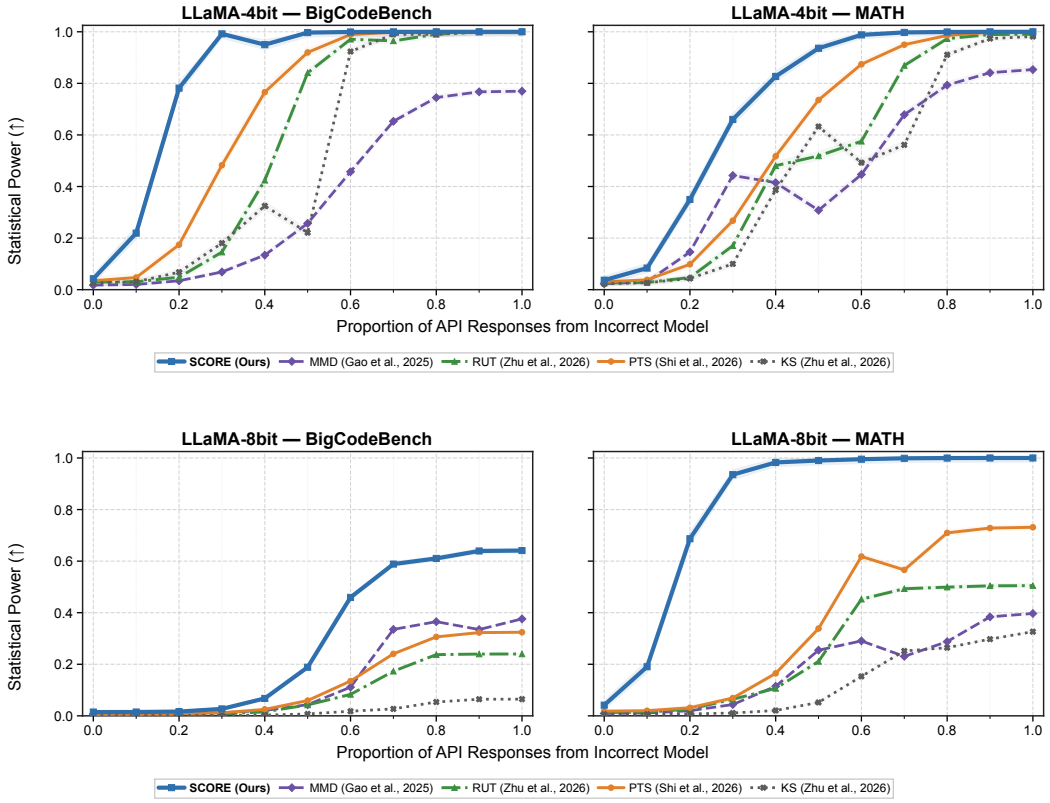
(a) MMLU-Pro Wang et al. (2024)

Model	SCORE	MMD	RUT	PTS	KS
Llama–4-bit	**0.880** $480^{-}\,576^{+}$	0.360 $242^{-}\,190^{+}$	0.575 $293^{-}\,397^{+}$	0.717 $473^{-}\,387^{+}$	0.498 $257^{-}\,341^{+}$
Llama–8-bit	0.285 $192^{-}\,150^{+}$	0.134 $68^{-}\,92^{+}$	0.095 $63^{-}\,51^{+}$	0.124 $64^{-}\,84^{+}$	0.022 $14^{-}\,12^{+}$
Gemma–4-bit	**0.290** $148^{-}\,200^{+}$	0.179 $118^{-}\,96^{+}$	0.128 $66^{-}\,88^{+}$	0.201 $132^{-}\,110^{+}$	0.013 $07^{-}\,09^{+}$
Gemma–8-bit	0.099 $65^{-}\,53^{+}$	0.050 $26^{-}\,34^{+}$	0.050 $33^{-}\,27^{+}$	0.051 $27^{-}\,35^{+}$	0.001 $02^{-}\,01^{+}$
Ministral–4-bit	**0.747** $385^{-}\,511^{+}$	0.349 $228^{-}\,190^{+}$	0.461 $244^{-}\,310^{+}$	0.592 $408^{-}\,302^{+}$	0.179 $96^{-}\,118^{+}$
Ministral–8-bit	0.151 $34^{-}\,28^{+}$	0.055 $29^{-}\,37^{+}$	0.041 $29^{-}\,21^{+}$	0.032 $17^{-}\,21^{+}$	0.012 $08^{-}\,06^{+}$
Qwen–4-bit	**0.698** $369^{-}\,469^{+}$	0.475 $328^{-}\,242^{+}$	0.556 $301^{-}\,367^{+}$	0.621 $425^{-}\,321^{+}$	0.320 $175^{-}\,209^{+}$
Qwen–8-bit	0.115 $79^{-}\,59^{+}$	0.031 $17^{-}\,21^{+}$	0.039 $26^{-}\,20^{+}$	0.052 $28^{-}\,34^{+}$	0.006 $05^{-}\,03^{+}$

(b) GPQA Rein et al. (2023)

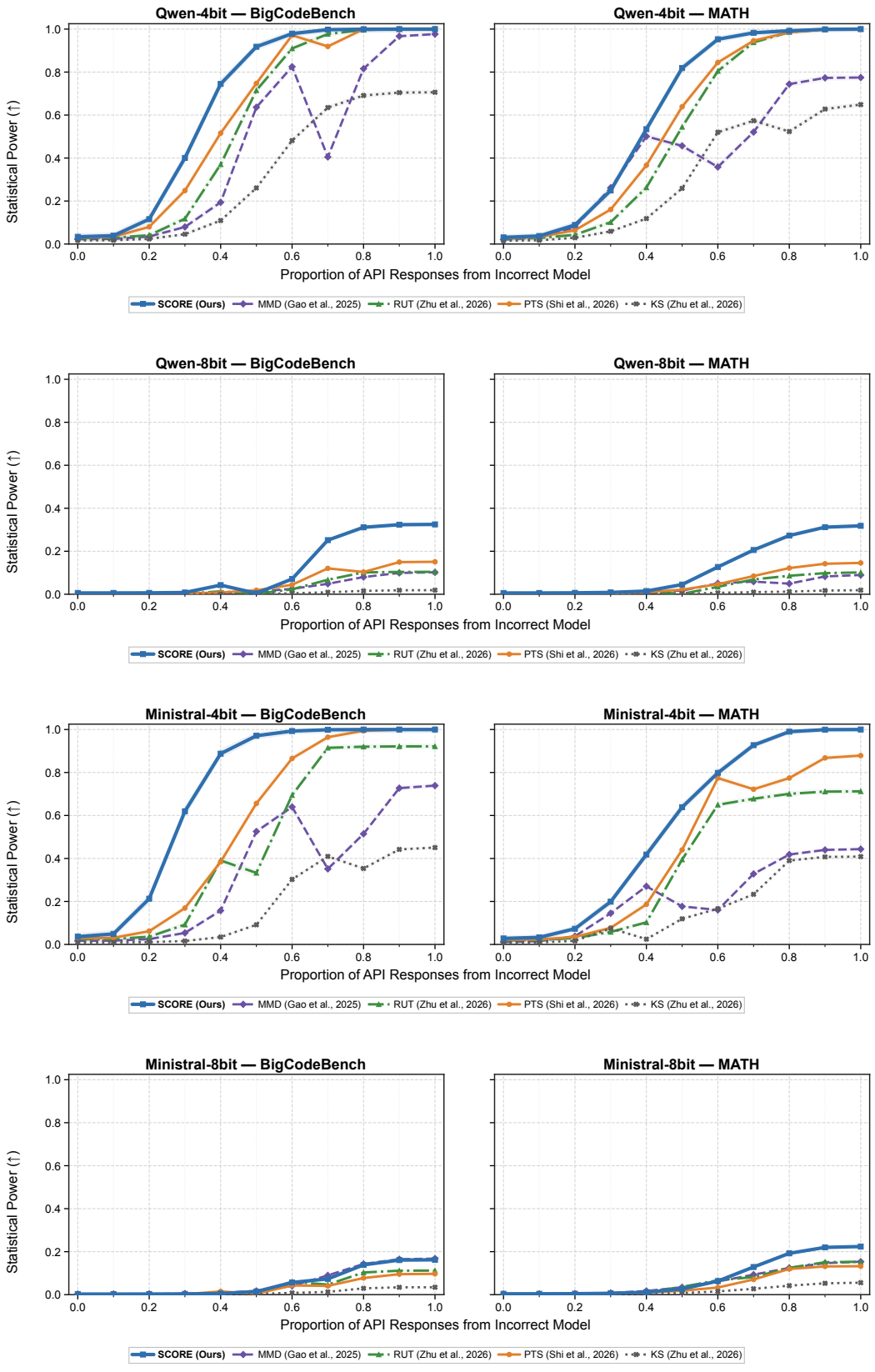
Model	SCORE	MMD	RUT	PTS	KS
Llama–4-bit	**0.699** $482^{-}\,356^{+}$	0.436 $236^{-}\,288^{+}$	0.542 $371^{-}\,279^{+}$	0.562 $307^{-}\,367^{+}$	0.472 $317^{-}\,249^{+}$
Llama–8-bit	0.163 $466^{-}\,570^{+}$	0.090 $130^{-}\,98^{+}$	0.052 $137^{-}\,165^{+}$	0.080 $255^{-}\,201^{+}$	0.016 $59^{-}\,81^{+}$
Gemma–4-bit	**0.614** $420^{-}\,316^{+}$	0.244 $133^{-}\,159^{+}$	0.321 $216^{-}\,170^{+}$	0.434 $221^{-}\,299^{+}$	0.065 $43^{-}\,35^{+}$
Gemma–8-bit	0.104 $56^{-}\,68^{+}$	0.068 $46^{-}\,36^{+}$	0.045 $23^{-}\,31^{+}$	0.054 $35^{-}\,29^{+}$	0.002 $01^{-}\,02^{+}$
Ministral–4-bit	**0.537** $361^{-}\,283^{+}$	0.211 $108^{-}\,146^{+}$	0.393 $260^{-}\,212^{+}$	0.426 $220^{-}\,292^{+}$	0.170 $111^{-}\,93^{+}$
Ministral–8-bit	0.079 $40^{-}\,54^{+}$	0.059 $39^{-}\,31^{+}$	0.053 $28^{-}\,36^{+}$	0.049 $32^{-}\,26^{+}$	0.018 $10^{-}\,12^{+}$
Qwen–4-bit	**0.586** $387^{-}\,317^{+}$	0.400 $206^{-}\,274^{+}$	0.543 $355^{-}\,297^{+}$	0.527 $278^{-}\,354^{+}$	0.324 $223^{-}\,165^{+}$
Qwen–8-bit	0.121 $63^{-}\,83^{+}$	0.033 $22^{-}\,18^{+}$	0.035 $18^{-}\,24^{+}$	0.053 $37^{-}\,27^{+}$	0.006 $03^{-}\,05^{+}$

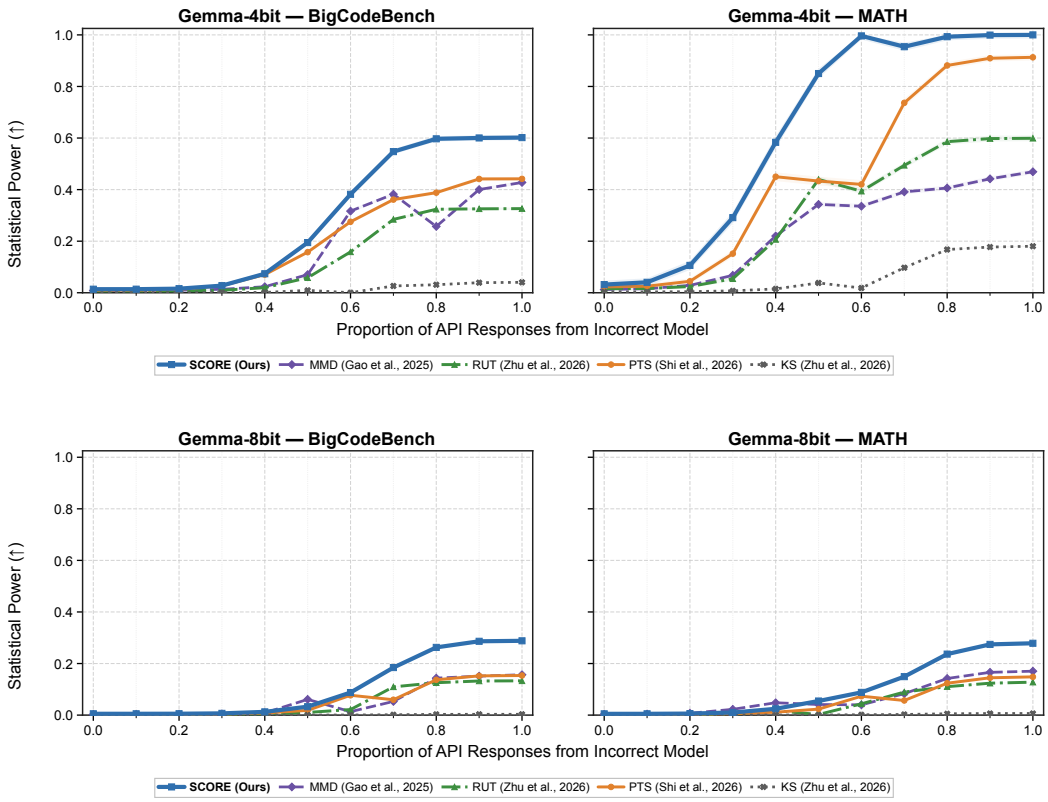
Table 5: Power AUC for detecting quantized variants across query domains. **Bold** denotes the best method; gray denotes none reliable. All confidence interval ranges are reported in $\times 10^{-4}$.



E Auditing the Impact of Decoding-Parameter Variations

We evaluate whether auditing methods can detect deviations from declared decoding configurations, focusing on sampling temperature and the top-p parameter of nucleus sampling. We compare alternative configurations against the default setting $(\text{temperature}, \text{top-}p) = (0.5, 1.0)$ while holding the model checkpoint, audit-prompt distribution, generation length, and all other inference settings fixed. We evaluate Llama-3.1-8B-Instruct and Qwen2.5-7B-Instruct through output-only interfaces and extend the analysis to the Gemini-3.1 Pro commercial black-box API using MATH (Hendrycks et al., 2021) and WildChat (Zhao et al., 2024). We compare SCORE only with MMD (Gao et al., 2025) and PTS (Shi et al., 2026), as both baselines satisfy the same strict output-only access constraint across all endpoints. As shown in Fig. 7 and 8, SCORE achieves higher power AUC than both baselines across most decoding configurations. These results demonstrate that SCORE can audit


Qwen-4bit — BigCodeBench
Qwen-4bit — MATH
Qwen-8bit — BigCodeBench
Qwen-8bit — MATH
Ministral-4bit — BigCodeBench
Ministral-4bit — MATH
Ministral-8bit — BigCodeBench
Ministral-8bit — MATH
Statistical Power (↑)
Proportion of API Responses from Incorrect Model
SCORE (Ours)
MMD (Gao et al., 2025)
RUT (Zhu et al., 2026)
PTS (Shi et al., 2026)
KS (Zhu et al., 2026)



decoding-parameter variations in both open-weight and commercial black-box models, even when the underlying model remains unchanged.

F Ablation Studies under Model Replacement

We conduct controlled ablations under model replacement to evaluate three design choices in SCORE: paired prompting, semantic recovery, and ratio scoring. We additionally examine sensitivity to the control prefix. All variants retain the same probe construction, randomized-rank calibration, and CvM test; within each comparison, all factors not required by the contrast are held fixed. Throughout Tables 6 and 7, *Difference* denotes semantic similarity with difference scoring, whereas SCORE denotes the same semantic similarity with ratio scoring. The corresponding rows therefore represent identical configurations across the two tables.

Experimental setup. We evaluate Llama-3.1-70B-Instruct, Ministral-3-8B-Instruct, Qwen2.5-7B-Instruct, Qwen2.5-72B-Instruct, and Gemma-3-27B-it, abbreviated as Llama-70B, Ministral-8B, Qwen-7B, Qwen-72B, and Gemma-27B. Each ordered pair of distinct models defines one replacement setting, yielding $5 \times 4 = 20$ comparisons. Let $A_{a\to b,v}$ denote the power AUC obtained when reference model M_a is replaced by M_b under variant v. We summarize performance as

$$\bar{A}_{a,v} = \frac{1}{4}\sum_{b\neq a} A_{a\to b,v}, \qquad \bar{A}_{v} = \frac{1}{20}\sum_{a\neq b} A_{a\to b,v}. \tag{21}$$

Each model column therefore averages over four replacement models, while Macro AUC assigns equal weight to all 20 ordered pairs. Self-comparisons are excluded, although $q = 0$ remains part of every power curve to assess null calibration.

Unless explicitly varied, all configurations use $n = 100$, $K = 5$, $m = 20$, $\varepsilon = 10^{-11}$, temperature 0.5, a 50-token generation limit, and $r_v = 0.05$. Every variant uses the same substitution-rate grid, 500 Monte Carlo trials per value of q, and the same AUC computation. The probe split,

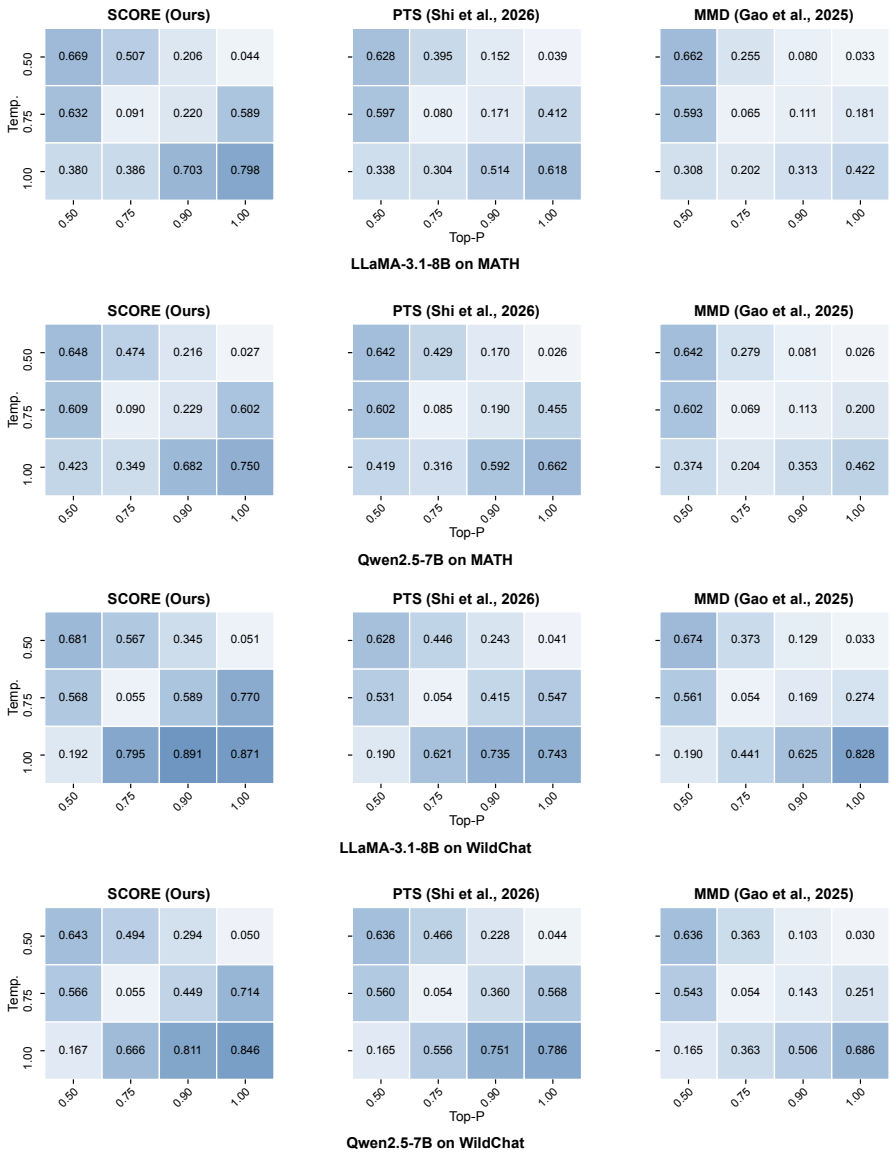


Figure 7: Power AUC for Detecting Decoding Parameter Variations (Temperature and Top-p) across models and datasets.

model versions, tokenizer, semantic encoder, response normalization, and prompt templates are fixed before evaluation. Compatible variants reuse the same generated responses to support paired comparisons. Within each audit, the target score, reference-score replicates, and scores from different probes are constructed from mutually independent response blocks. The randomized-rank CvM test uses $B = 9{,}999$ null simulations at significance level $\alpha = 0.05$. Tables report AUCs to four decimal places; all differences, ranges, and percentages below are computed from the corresponding unrounded estimates.

F.0.1 Paired Prompting and Ratio Scoring

We first examine whether the paired-contrast design provides information beyond allocating the same generation budget to a single prompt condition. Using semantic recovery throughout, we compare original-only recovery, prefix-only recovery, difference scoring, and the ratio score used by SCORE.

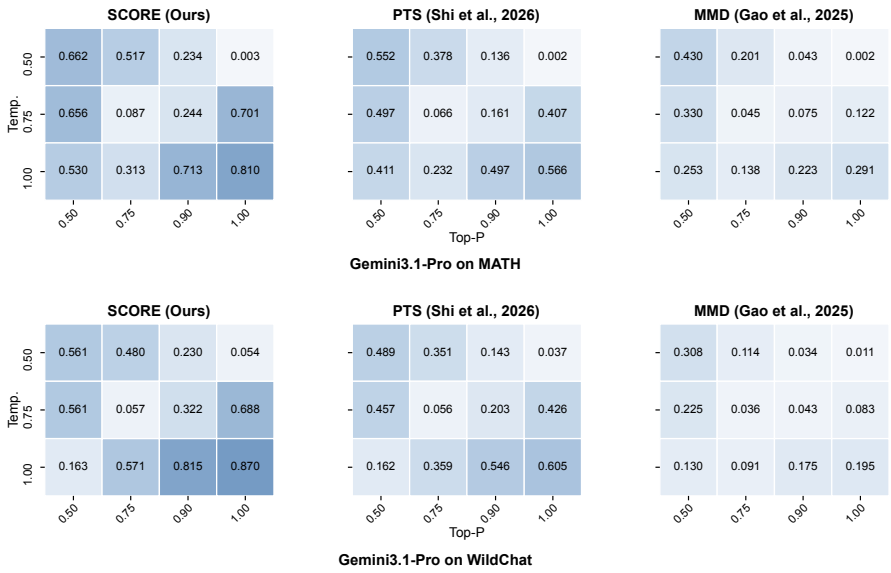


Figure 8: Power AUC for Detecting Decoding Parameter Variations (Temperature and Top-p) across Datasets on Gemini-3.1 Pro.

Table 6: Ablation study of score construction. For each reference model, the reported power AUC is averaged over the four replacement models. The final two columns present r_v and Macro AUC, respectively, with Macro AUC computed across all 20 ordered model pairs. Higher AUC values indicate stronger performance.

Variant	Llama-70B	Ministral-8B	Qwen-7B	Qwen-72B	Gemma-27B	r_v	Macro AUC
Original-only ($K = 5$)	0.5406	0.5530	0.5496	0.5500	0.5474	0.05	0.5481
Prefix-only ($K = 5$)	0.5089	0.5205	0.5173	0.5178	0.5153	0.05	0.5160
Original-only ($K = 10$)	0.6197	0.6341	0.6301	0.6307	0.6276	0.05	0.6284
Prefix-only ($K = 10$)	0.5881	0.6017	0.5979	0.5984	0.5955	0.05	0.5963
Semantic + Difference ($K = 5 + 5$)	0.7463	0.7639	0.7591	0.7597	0.7560	0.05	0.7570
SCORE (Semantic + Ratio)	**0.8413**	**0.8613**	**0.8558**	**0.8565**	**0.8523**	0.05	**0.8534**

With the overline denoting an average over evaluated lexical positions, these scores are

$$S_o = \overline{A_o}, \qquad S_p = \overline{A_p}, \qquad S_{\text{diff}} = \overline{A_o - A_p}, \qquad S_{\text{ratio}} = -\overline{\frac{A_p}{A_o + \varepsilon}}. \tag{22}$$

The two paired configurations use $K = 5$ generations per condition. Single-condition variants with $K = 5$ reuse the corresponding response blocks, whereas the $K = 10$ variants match the total generation budget of the paired configurations. At $m = 20$, a single-condition $K = 5$ variant uses 5 target and 100 reference completions per evaluated position; the paired configurations and $K = 10$ controls use 10 target and 200 reference completions.

At $K = 5$, Original-only and Prefix-only attain Macro AUCs of 0.5481 and 0.5160, respectively. Increasing their single-condition budget to $K = 10$ raises these values to 0.6284 and 0.5963. For Original-only, this increase accounts for 26.3% of its initial gap to SCORE, leaving a Macro-AUC difference of 0.2250 under the matched completion budget. Additional sampling therefore explains only part of the observed performance difference.

Under the matched budget, Semantic + Difference reaches a Macro AUC of 0.7570, exceeding Original-only and Prefix-only by 0.1285 and 0.1607, respectively. This comparison evaluates the paired semantic-contrast design as a whole, because it changes both the number of prompt conditions and the allocation of samples across them. With the generated responses and semantic-recovery values held fixed, replacing difference scoring with ratio scoring further increases Macro AUC by 0.0964 to 0.8534. This latter contrast isolates the effect of post-generation score construction within the evaluated design.

Table 7: Ablation study of semantic recovery. The 2×2 design varies the recovery representation and scoring rule while keeping the generated responses fixed. The final two columns report r_v and Macro AUC, respectively, with Macro AUC computed over all 20 ordered model pairs. Higher AUC values indicate better performance.

Variant	Llama-70B	Ministral-8B	Qwen-7B	Qwen-72B	Gemma-27B	r_v	Macro AUC
Exact + Difference	0.5485	0.5611	0.5576	0.5581	0.5554	0.05	0.5561
Semantic + Difference	0.7463	0.7639	0.7591	0.7597	0.7560	0.05	0.7570
Exact + Ratio	0.5010	0.5124	0.5093	0.5097	0.5073	0.05	0.5079
SCORE (Semantic + Ratio)	**0.8413**	**0.8613**	**0.8558**	**0.8565**	**0.8523**	0.05	**0.8534**

Relative to budget-matched Original-only, the total descriptive difference of 0.2250 can be traced along the specified comparison sequence as 0.1285 from the paired-contrast configuration and 0.0964 from ratio scoring, corresponding to 57.1% and 42.9%, respectively. This accounting is specific to the chosen comparison path and should not be interpreted as a general causal decomposition.

The two scoring rules encode different recovery changes. For $A_o > 0$ and $\varepsilon \ll A_o$,

$$-\frac{A_p}{A_o + \varepsilon} \approx \frac{A_o - A_p}{A_o} - 1. \tag{23}$$

Difference scoring measures absolute recovery changes, whereas ratio scoring expresses each change relative to its original recovery level. This position-dependent normalization can alter probe-score ordering and, consequently, the randomized ranks used by the final test.

F.0.2 Semantic Recovery

We next examine whether semantic similarity retains information that exact lexical agreement discards. We cross semantic versus exact recovery with difference versus ratio scoring, yielding a 2×2 design. All four configurations use identical generated responses, normalization, and randomized-rank calibration. Exact recovery is defined as

$$\kappa_{\text{exact}}(u, v) = \mathbb{1}[\nu(u) = \nu(v)]\,. \tag{24}$$

Both ratio variants use $\varepsilon = 10^{-11}$ and the same randomized tie-breaking rule. The Semantic + Difference and Semantic + Ratio configurations are identical to the corresponding rows in Table 6.

Under difference scoring, replacing exact recovery with semantic similarity increases Macro AUC from 0.5561 to 0.7570, an observed contrast of 0.2009. Because the responses and scoring rule remain fixed, this comparison directly evaluates the recovery representation within the difference-scoring design. Under ratio scoring, the corresponding semantic–exact contrast is 0.3455, from 0.5079 to 0.8534. The effect of ratio scoring depends on the recovery representation. Ratio scoring increases Macro AUC by 0.0964 with semantic similarity but decreases it by 0.0482 with exact matching, yielding a descriptive difference-in-differences of 0.1446. Because exact matching also changes score discreteness, denominator behavior, and tie frequency, the 0.3455 contrast under ratio scoring cannot be attributed to semantic representation alone. The observed interaction instead favors the joint use of graded semantic recovery and ratio scoring, without identifying a specific numerical mechanism.

F.0.3 Control-Prefix Sensitivity

Finally, we assess whether SCORE depends strongly on a particular control prefix. We compare the default prefix z_1 with four alternatives drawn from a disjoint control split and matched in language and lexical-unit length. Each configuration uses one fixed prefix ($H = 1$), $r_v = 0.05$, and the same sampling budget. The prefix set is selected independently of target audit outcomes. Original-condition responses are shared where compatible, whereas each distinct prefix condition requires separate generations.

According to the unrounded estimates, Macro AUC ranges from 0.8507 to 0.8552 across the five prefixes, with a mean of 0.8531 and a range of 0.0046. The highest-Macro-AUC prefix is z_2, whereas the best model-specific prefix varies across reference models. For Qwen-72B, z_1 and z_4 both round

Table 8: Ablation study of control-prefix sensitivity. The final two columns report r_v and Macro AUC, respectively. Boldface indicates the highest AUC in each column; ties at the displayed precision are both highlighted in bold.

Variant	Llama-70B	Ministral-8B	Qwen-7B	Qwen-72B	Gemma-27B	r_v	Macro AUC
z_1 (default)	0.8413	0.8613	0.8558	**0.8565**	0.8523	0.05	0.8534
z_2	**0.8516**	0.8599	**0.8701**	0.8539	0.8407	0.05	**0.8552**
z_3	0.8450	**0.8747**	0.8454	0.8510	0.8549	0.05	0.8542
z_4	0.8372	0.8481	0.8563	**0.8565**	**0.8612**	0.05	0.8519
z_5	0.8259	0.8670	0.8666	0.8487	0.8451	0.05	0.8507

to 0.8565, with z_4 exceeding z_1 by only 1.613×10^{-5} before rounding. Model-specific ranges span 0.0078–0.0267, showing that macro averaging can conceal larger variation for individual reference models. The macro-level prefix range is approximately 3.5% of the matched-budget paired-design contrast and 2.3% of the semantic-recovery contrast under difference scoring. Variation across the selected control texts is therefore substantially smaller than the principal component contrasts. Nevertheless, the variation in prefix rankings across models indicates that no single prefix is uniformly optimal. The comparison distinguishes the value of introducing a control condition from sensitivity to its specific text. Table 6 evaluates the former through matched-budget variants, whereas Table 8 evaluates the latter over five prespecified prefixes. Prespecifying the prefix and applying the same score map to both endpoints prevents data-dependent selection from the audit responses. Null exchangeability additionally relies on the endpoint-equality and block-sampling assumptions. The results therefore establish limited sensitivity within the evaluated prefix set, but not robustness to arbitrary or adaptively selected controls.

F.0.4 Summary of Component Effects

The complete Semantic + Ratio configuration achieves a Macro AUC of 0.8534 and outperforms every score-construction and semantic-recovery ablation in Tables 6 and 7. Collectively, the controlled comparisons favor retaining the components jointly: paired prompting exposes control-induced recovery changes, semantic similarity preserves graded agreement, and ratio scoring expresses recovery changes relative to their original levels. The Macro-AUC range of 0.0046 and model-specific ranges of 0.0078–0.0267 further characterize sensitivity to the selected control prefixes. These findings support the complete design without claiming that the aggregate contrasts identify independent causal effects for its individual components.

G Theoretical Analysis of SCORE

Following the rank-distribution perspective of RUT (Zhu et al., 2026), we analyze the finite-sample validity and power of SCORE through its probe-level randomized ranks. The analysis uses the semantic-contrast score and independent sampling protocol in Sections 3 and 4; it requires neither continuous scores nor a parametric model of their distribution. Let $\mathcal{D}$ denote the fixed audit specification and $\mathcal{G} = (\mathcal{D}, X_{1:n})$ also include the realized probes. All score distributions below are conditional on $\mathcal{D}$, which we suppress from the notation. The per-probe specification is held fixed as n varies. For a fixed probe x, write $S = f_x(\mathcal{Y})$, where $\mathcal{Y}$ contains the $2K(L(x) - 1)$ completions entering Eq. equation 11. The same measurable map f_x is used for the target and every reference replicate. Since $\kappa \in [0, 1]$ and $\varepsilon > 0$, this score satisfies $-1/\varepsilon \leq S \leq 0$ whenever the response normalization and encoder are well-defined.

Assumption G.1 (Prespecified independent sampling)**.** The probe distribution, prompt construction, control prefix, tokenizer, encoder, score map, decoding configuration, and sample sizes are fixed before drawing the audit probes and responses. The score map includes prespecified handling of all possible outputs. Conditional on the probes, completions are mutually independent across endpoints, positions, conditions, sample indices, probes, and score replicates, with stationary response kernels throughout the audit. All random seeds for rank randomization and null simulation are mutually independent and independent of the responses and probes.

Under H_0, the response kernels agree for μ-almost every prompt. Every enumerated position–condition pair has positive conditional weight in Eq. equation 2, so this equality holds at every such pair for ξ-almost every probe. Assumption G.1 then makes the target and reference completion blocks conditionally i.i.d.; applying f_x preserves this property. Equality of single-completion kernels alone would not establish this conclusion for arbitrarily correlated API batches. The stated protocol uses $c_{\mathrm{tgt}} = c_{\mathrm{ref}} = 1$.

For later use, define the conditional reference-score CDFs

$$F_x^-(s) := \Pr(S^{\mathrm{ref}} < s \mid X = x), \qquad F_x(s) := \Pr(S^{\mathrm{ref}} \le s \mid X = x). \tag{25}$$

These distributions may depend on x and may contain atoms. To make the tie randomization precise, take independent $Z_i, U_i \sim \mathrm{Uniform}[0,1)$ and set $J_i = \lfloor (C_i^= + 1) Z_i \rfloor$. Thus, J_i is conditionally uniform on $\{0, \dots, C_i^=\}$; it is the seed Z_i, rather than J_i itself, that is independent of the scores.

G.1 Exact Finite-Sample Validity

Theorem G.2 (Exact finite-reference ranks)**.** *Under H_0 and Assumption G.1, for every $m \ge 1$,*

$$C_i^< + J_i \mid \mathcal{G} \sim \mathrm{Unif}\{0, \dots, m\}, \qquad R_i \mid \mathcal{G} \sim \mathrm{Uniform}[0,1]. \tag{26}$$

Moreover, $R_1, \dots, R_n$ are conditionally i.i.d. The result holds for arbitrary score distributions, including distributions with ties.

Proof. Fix a probe and condition on $\mathcal{G}$ and the unordered multiset of its $m+1$ scores. Let the distinct values, in increasing order, have multiplicities $d_1, \dots, d_g$, and set $D_h = \sum_{a=1}^{h} d_a$, with $D_0 = 0$. Exchangeability assigns the target to group h with probability $d_h/(m+1)$. On this event, $C_i^< = D_{h-1}$ and $C_i^= = d_h - 1$, so $C_i^< + J_i$ is uniform on $\{D_{h-1}, \dots, D_h - 1\}$. Each integer in $\{0, \dots, m\}$ therefore has probability $1/(m+1)$. The independent U_i spreads this mass uniformly over an interval of width $1/(m+1)$. Independence across probes and random seeds proves the joint claim. □

Theorem G.3 (Finite-sample type I error control)**.** *For $n, m, B \ge 1$ and $0 < \alpha < 1$, the Monte Carlo p-value in Eq. equation 16 satisfies*

$$\Pr_{H_0}(\widehat{p} \le \alpha \mid \mathcal{G}) \le \frac{\lfloor \alpha(B+1) \rfloor}{B+1} \le \alpha. \tag{27}$$

Consequently, the unconditional test is also valid. The condition $(B+1)^{-1} \le \alpha$ is needed only to make rejection attainable.

Proof. By Theorem G.2, the observed rank sample and the B independently simulated uniform samples have the same conditional distribution. Their CvM statistics are exchangeable. Ranking the observed statistic among these $B+1$ values gives the plus-one p-value; counting ties with the upper tail can only reduce the rejection probability. Averaging the conditional bound over $\mathcal{G}$ gives unconditional validity. □

Here, B counts null simulations for one p-value, not the repeated audit trials used to estimate experimental power. The theorem also applies to a fixed probe set; i.i.d. probe sampling is used below to characterize power through a common marginal rank distribution.

G.2 Effect of the Reference Sample Size

Let $Z \sim \mathrm{Uniform}[0,1)$ be independent of the scores. The population randomized reference quantile is

$$R_x^\infty := F_x^-(S^{\mathrm{tgt}}) + Z\big[F_x(S^{\mathrm{tgt}}) - F_x^-(S^{\mathrm{tgt}})\big]. \tag{28}$$

This definition accommodates atoms in the reference-score distribution.

Proposition G.4 (Finite-reference limit)**.** *Under Assumption G.1, let $R_{m,x}$ denote the rank conditional on $X = x$. Then*

$$R_{m,x} \xrightarrow{\mathrm{d}} R_x^{\infty} \qquad \text{as } m \to \infty. \tag{29}$$

If F_x is continuous and $V_x = F_x(S^{\mathrm{tgt}})$, then

$$\mathbb{E}[R_{m,x} \mid V_x] = \frac{mV_x + 1/2}{m+1}, \tag{30}$$

$$\mathrm{Var}(R_{m,x} \mid V_x) = \frac{mV_x(1-V_x) + 1/12}{(m+1)^2}, \tag{31}$$

$$\mathbb{E}[(R_{m,x} - V_x)^2 \mid V_x] = \frac{mV_x(1-V_x) + 1/12 + (1/2 - V_x)^2}{(m+1)^2}. \tag{32}$$

Proof. Given $S^{\mathrm{tgt}} = s$, the law of large numbers gives

$$(C^{<}/m, C^{=}/m) \xrightarrow{\text{a.s.}} \big(F_x^-(s), F_x(s) - F_x^-(s)\big).$$

Couple all reference sample sizes using an infinite i.i.d. reference sequence and the same independent Z, with $J = \lfloor (C^{=}+1)Z \rfloor$. Then $(C^{<} + J + U)/(m+1)$ converges almost surely to the right-hand side of Eq. equation 28. Integrating over the target score proves convergence in distribution. For continuous F_x, $C^{=} = J = 0$ almost surely and $C^{<} \mid V_x \sim \mathrm{Binomial}(m, V_x)$; the three identities follow from independence of $C^{<}$ and U conditional on V_x. □

Thus, finite-reference error vanishes as m grows, while null validity is already exact at every finite m. This approximation result does not imply that detection power increases monotonically with m.

G.3 Power Against SCORE-Detectable Alternatives

With the audit configuration fixed and $X_i \overset{\text{i.i.d.}}{\sim} \xi$, define

$$H_{m,x}(u) := \Pr(R_i \le u \mid X_i = x), \qquad H_m(u) := \int H_{m,x}(u)\, \xi(\mathrm{d}x), \tag{33}$$

and

$$\Delta_m^2 := \int_0^1 [H_m(u) - u]^2 \, \mathrm{d}u, \qquad \Delta_m := \sqrt{\Delta_m^2}. \tag{34}$$

An alternative is *SCORE-detectable at reference size* m if $\Delta_m^2 > 0$. Under the alternative, the ranks are i.i.d. after averaging over the independent probes; their conditional distributions need not agree. The marginal definition matters because probe-specific deviations can cancel.

Theorem G.5 (Consistency and its boundary)**.** *Under Assumption G.1 and i.i.d. probes, fix m and B with $(B+1)^{-1} \le \alpha$. If $\Delta_m^2 > 0$, then*

$$\Pr(\widehat{p} \le \alpha) \longrightarrow 1 \qquad \text{as } n \to \infty. \tag{35}$$

If $\Delta_m^2 = 0$, the rejection probability is at most $\lfloor \alpha(B+1) \rfloor/(B+1)$, even if H_1 holds.

Proof. Let $\widehat{H}_n$ be the empirical rank CDF. The integral representation of the CvM statistic and the Glivenko–Cantelli theorem give

$$\frac{\omega_{\mathrm{obs}}^2}{n} = \int_0^1 [\widehat{H}_n(u) - u]^2 \, \mathrm{d}u \xrightarrow{\text{a.s.}} \Delta_m^2. \tag{36}$$

Under the null, $\mathbb{E}[\omega_b^2] = 1/6$ for every n, since $n \,\mathrm{Var}(\widehat{H}_n(u)) = u(1-u)$. Hence each null statistic is $O_{\Pr}(1)$. For fixed B and positive Δ_m^2, the observed statistic exceeds all simulated statistics with probability tending to one. On this event, $\widehat{p} = (B+1)^{-1} \le \alpha$. If $\Delta_m^2 = 0$, right continuity implies $H_m(u) = u$ everywhere, so the i.i.d. ranks have exactly their null distribution. □

Finite reference size can itself conceal score differences. For continuous F_x, the probability of rank interval $[j/(m+1), (j+1)/(m+1))$ is

$$\mathbb{E}\left[\binom{m}{j} V_x^j (1-V_x)^{m-j}\right], \qquad j = 0, \dots, m. \tag{37}$$

These probabilities depend only on the first m moments of V_x. Consequently, a nonuniform V_x sharing those moments with a uniform variable also induces uniform finite-reference ranks. For example, when $m = 1$, $S^{\mathrm{ref}} \sim \mathrm{Uniform}[-1, 0]$ and $S^{\mathrm{tgt}} \equiv -1/2$ produce a uniform rank despite different score laws. Thus, score differences alone do not guarantee $\Delta_m^2 > 0$.

Proposition G.6 (Finite-sample power of the Monte Carlo test)**.** *Under the assumptions of Theorem G.5, suppose $\Delta_m^2 > 0$ and define $k_{\alpha,B} := \lfloor \alpha(B+1) \rfloor \geq 1$ and $\vartheta_{n,m} := \min\{1, 2/(3n\Delta_m^2)\}$. Then*

$$\Pr(\widehat{p} \leq \alpha) \geq 1 - 2e^{-n\Delta_m^2/2} - \Pr\{\mathrm{Binomial}(B, \vartheta_{n,m}) \geq k_{\alpha,B}\}. \tag{38}$$

For any $0 < \gamma < 1$, the conservative sufficient condition

$$n > \frac{1}{\Delta_m^2} \max\left\{2 \log \frac{4}{\gamma}, \frac{4B}{3k_{\alpha,B}\gamma}\right\} \tag{39}$$

guarantees power at least $1 - \gamma$ for the implemented finite-B test.

Proof. The Dvoretzky–Kiefer–Wolfowitz inequality implies

$$\Pr\left(\|\widehat{H}_n - H_m\|_\infty > \Delta_m/2\right) \leq 2e^{-n\Delta_m^2/2}.$$

On the complementary event, the reverse triangle inequality in $L^2[0,1]$ gives $\omega_{\mathrm{obs}}^2 \geq t := n\Delta_m^2/4$. For each simulated null statistic, Markov's inequality gives

$$p_0(t) := \Pr(\omega_b^2 \geq t) \leq \min\{1, 1/(6t)\} = \vartheta_{n,m}.$$

The number of simulated statistics at least t is $\mathrm{Binomial}(B, p_0(t))$. If this count is below $k_{\alpha,B}$ and $\omega_{\mathrm{obs}}^2 \geq t$, the Monte Carlo test rejects. A union bound and binomial stochastic monotonicity prove Eq. equation 38. Applying Markov's inequality once more bounds its binomial tail by $2B/(3k_{\alpha,B}n\Delta_m^2)$. Eq. equation 39 makes each error term at most $\gamma/2$. □

For fixed α, B, γ, this sufficient probe count is $O(\Delta_m^{-2})$. Combining it with Eq. equation 17 and $L_i \leq L_{\max}$ gives sufficient request budgets of orders

$$\begin{aligned} N_{\mathrm{tgt}} &= O\left(\frac{(L_{\max}-1)\lceil K/c_{\mathrm{tgt}}\rceil}{\Delta_m^2}\right), \\ N_{\mathrm{ref}} &= O\left(\frac{(L_{\max}-1)\lceil mK/c_{\mathrm{ref}}\rceil}{\Delta_m^2}\right). \end{aligned} \tag{40}$$

These are sufficient budgets, not lower bounds or claims of optimality. Here n counts independent probes, not completions. Since Δ_m depends on m, K, the score map, and the probe distribution, the bound does not imply a universal cost–power ordering across these choices.

G.4 Prompt-Dependent Substitution

Under the stated one-completion-per-request protocol, a target score for probe x uses $M_x = 2K(L(x)-1)$ independently routed requests. Index their prompts by q_{xt}, allowing repetitions, and let P_{xt} and Q_{xt} be the reference and alternative response laws. Set $\rho_\eta(q) = \eta r(q)$, where $0 \leq \eta, r(q) \leq 1$. The target-score law is

$$G_{x,\eta} = (f_x)_{\#} \bigotimes_{t=1}^{M_x} \{[1 - \eta r(q_{xt})]P_{xt} + \eta r(q_{xt}) Q_{xt}\}, \tag{41}$$

where $(f_x)_{\#}$ denotes the distribution induced by applying f_x. Because one score aggregates multiple requests, this is generally not a two-component mixture of an all-reference and an all-alternative score law.

Corollary G.7 (Local prompt-dependent substitution)**.** *Fix $m, K, L_{\max}$ and the audit design. Under independent, memoryless routing, the marginal rank CDF satisfies*

$$H_{m,\eta}(u) = u + \eta h_m(u) + O_\infty(\eta^2), \tag{42}$$

where the remainder is uniform over $u \in [0,1]$. More explicitly, let $H^{(t)}_{m,x}$ denote the conditional rank CDF when only target request t uses Q_{xt} and all other target requests use their reference laws. Then

$$h_m(u) = \int \sum_{t=1}^{M_x} r(q_{xt})[H^{(t)}_{m,x}(u) - u]\, \xi(\mathrm{d}x). \tag{43}$$

If $\|h_m\|_2 > 0$, then

$$\Delta^2_m(\eta) = \eta^2 \|h_m\|_2^2 + O(\eta^3). \tag{44}$$

For $\eta_n = c/\sqrt{n}$ with fixed $c \geq 0$,

$$\omega^2_{\mathrm{obs}} \Longrightarrow T_c := \int_0^1 [\mathbb{B}_0(u) + c h_m(u)]^2 \, \mathrm{d}u, \tag{45}$$

where $\mathbb{B}_0$ is a standard Brownian bridge.

Proof. The product in Eq. equation 41 is a polynomial in η of degree at most $M_x \leq 2K(L_{\max} - 1)$. At $\eta = 0$, the rank CDF is u. Differentiating replaces exactly one reference request by its alternative, giving Eq. equation 43. The bounded degree and $r \in [0,1]$ bound the remaining terms uniformly after applying the score and rank maps and averaging over probes. Squaring and integrating proves Eq. equation 44.

For the local limit, $H_{m,\eta_n} \to \mathcal{U}$ uniformly, where $\mathcal{U}(u) = u$. Representing each row of i.i.d. ranks by the quantile transform of i.i.d. uniforms, the uniform empirical-process limit gives $\sqrt{n}(\widehat{H}_n - H_{m,\eta_n}) \Longrightarrow \mathbb{B}_0$. Eq. equation 42 also gives $\sqrt{n}(H_{m,\eta_n} - \mathcal{U}) \to c h_m$ uniformly. The integral representation of CvM now yields Eq. equation 45. □

For the finite-B implementation, the corresponding local limiting power is

$$\Pr\left\{ \sum_{b=1}^{B} \mathbb{I}[T^0_b \geq T_c] \leq k_{\alpha,B} - 1 \right\}, \qquad T^0_b := \int_0^1 \mathbb{B}_b(u)^2 \, \mathrm{d}u, \tag{46}$$

where $\mathbb{B}_0, \ldots, \mathbb{B}_B$ are independent bridges. Their continuous statistic distributions justify passing to the rejection event. Thus $\eta = \Theta(n^{-1/2})$ describes a local detection scale, not a guarantee of power tending to one. If $h_m \neq 0$ and $n\eta_n^2 \to \infty$ with $\eta_n \to 0$, the finite-sample bound gives power tending to one. For fixed desired power, it provides a sufficient $n = O(\eta^{-2})$ scaling, without claiming a matching lower bound. If $h_m = 0$, first-order effects cancel and no universal η^{-2} guarantee follows.

Scope of the guarantees. The score map can discard response-distribution differences; finite-reference calibration can discard score-distribution differences; and mixing over probes can cancel conditional rank deviations. Consistency therefore applies exactly to alternatives with $\Delta^2_m > 0$, not every pair of unequal endpoint kernels. Failure to reject does not establish model equality. Nor does this analysis prove that semantic ratio scoring is universally more powerful than other scores; such comparisons remain empirical. The guarantees require the prespecified sampling protocol and exclude unmodeled dependence, endpoint drift, history-dependent routing, or score selection using the same audit responses without independent data splitting or a separate valid calibration. The error guarantee concerns one prespecified audit; aggregating repeated decisions requires an additional error analysis.