REFERENCE-BASED DISTILLATION DETECTION IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Model distillation—training on outputs from stronger third-party models—is widely used to boost performance, but raises concerns about unfair advantages and policy violations. This motivates a fundamental question: *can we detect whether a model was distilled from another?* We show that, while identifying a teacher model from a student in isolation is highly challenging, it becomes tractable in a **reference-based setting**: given a model and an earlier-generation checkpoint from the same lineage, we can identify the teacher model used to train the later checkpoint. We introduce a distillation detection method based on reference-based membership inference. By comparing how strongly a student model preferentially aligns with outputs from different candidate teachers relative to a reference checkpoint, our method identifies the most likely teacher and detects evidence of distillation. To handle unknown distillation pipelines such as hidden prompts, we infer proxy prompt templates directly from model outputs. We additionally identify a distinctive glyph-level signal specific to o1/o3 models. Evaluating distillation detection is challenging because modern model lineages are already heavily entangled. To address this, we develop a hybrid evaluation spanning both controlled distillation experiments and real-world models. Across both settings, our approach recovers the true teacher with near-perfect accuracy in single-teacher distillation scenarios, even when the underlying distillation pipeline is largely unknown. We further introduce statistical tests for both teacher attribution and distillation detection, and extend our framework to open-world settings where no teacher is guaranteed to be present among the candidates. Applying our method to contemporary models yields new evidence regarding potential distillation relationships involving QwQ, DeepSeek-R1, and GPT-OSS.

1 INTRODUCTION

Large language models (LLMs) are increasingly trained via model distillation, where a student is fine-tuned on outputs from a stronger teacher to transfer capabilities such as reasoning and instruction following (Hinton et al., 2015; Wang et al., 2022; Toshniwal et al., 2024; Muennighoff et al., 2025). This creates ecosystem-level risks: a competitor can query a proprietary model at scale and train on its outputs, cheaply acquiring capabilities that would otherwise require new architectures or datasets. Major providers have responded with defenses based on request metadata (News, 2026; Seetharaman & Arámburo, 2026; Anthropic, 2026), but these require usage logs and infrastructure access and do not address detection from model behavior.

This motivates a fundamental problem in model auditing: *given a model, can we determine whether a teacher was used to train it?* This is challenging due to both the difficulty of membership inference in modern LLMs and the opacity of distillation pipelines. We find that a reference-based setting is more tractable: *given a model and an earlier-generation checkpoint of itself, can we determine whether a teacher was used to train the later checkpoint, and if so, which one?*

In this work, we propose a teacher identification method based on contrastive, reference-normalized likelihoods (§3; Figure 1(a)): we score each candidate teacher by how strongly the student prefers its outputs relative to a reference model. The key insight is that likelihoods taken relative to the earlier checkpoint—not absolute ones—carry the signal: standard membership inference metrics (e.g., raw likelihood, ZLIB, Min-K) fail to reliably recover the true teacher, whereas our formulation remains

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.

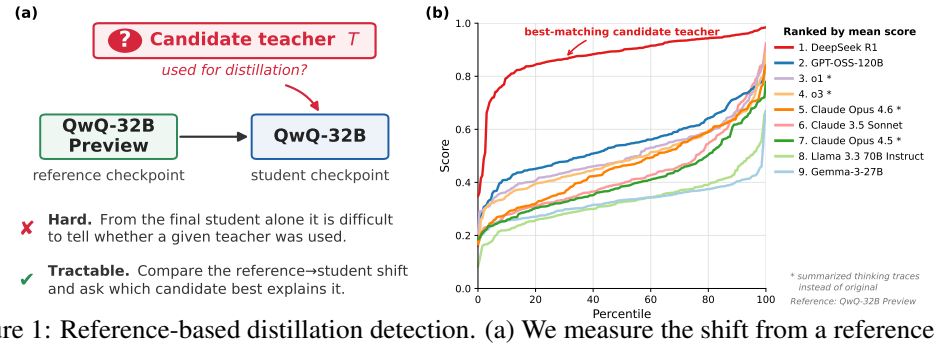


Figure 1: Reference-based distillation detection. (a) We measure the shift from a reference model to the student, rather than attributing from the student alone. (b) Scoring candidate teachers for QwQ-32B identifies DeepSeek R1 (red).

robust. The method requires only black-box access to candidate teachers, extends to hidden-prompt settings via proxy prompts, and is paired with a per-probe significance test. We additionally identify a glyph-level signature of o1/o3 models—unusually frequent non-ASCII Unicode characters—which yields a complementary indicator of o1/o3 distillation.

Because model lineage is already deeply entangled across modern LLMs, we design a hybrid evaluation combining controlled experiments, which ensure clean ground truth, and real-world models, which reflect practical conditions (§4), focusing on single-teacher, SFT-style distillation. Across both, our method recovers the true teacher with near-perfect accuracy, significantly outperforming baselines, and remains robust under domain shift; preliminary results also recover all teachers under multi-teacher distillation (§B.5). We pair the detection threshold with a significance test and cross-prompt agreement; cross-validation suggests the threshold generalizes to unseen students.

Finally, we apply our framework in an open-world setting, where no teacher is guaranteed to be among the candidates, through case studies on QwQ, DeepSeek-R1, and GPT-OSS (§7; Figure 1(b)). Together, these findings establish the feasibility of model-level auditing for distillation in modern LLMs, and we hope they motivate a more thorough treatment of attribution under complex, multi-stage regimes.

2 Related Work

Distillation in LLMs Model distillation was introduced for compressing knowledge into smaller models in image models (Hinton et al., 2015). Recent work has adopted distillation to obtain strong LLMs (Kim & Rush, 2016; Sanh et al., 2019; Jiao et al., 2020), such as Phi (Abdin et al., 2024) and Alpaca (Taori et al., 2023), with gains improving when the teacher is a reasoning LLM (Muennighoff et al., 2025). While effective and widely adopted, distillation creates ecosystem-level risks: model developers may query competitor models at scale and train systems on their outputs to replicate capabilities at a fraction of the original development cost, an increasingly imminent concern repeatedly reported by major AI providers (News, 2026; Anthropic, 2026; Seetharaman & Arámburo, 2026).

Distillation detection in LLMs The growing use of distillation has consequently prompted major AI providers to deploy detection systems, which primarily rely on request-side monitoring and infrastructure signals rather than model behavior itself, such as request metadata and IP address correlation (Anthropic, 2026). Another line of work studies watermark-based detection, where the auditor controls the teacher model and embeds detectable signals into generated outputs (Sander et al., 2024); these require strong assumptions about provider-side access, and such watermarks can often be weakened or removed during downstream training (Pan et al., 2025). Zhang et al. (2025) studies a different but related problem: detecting which prompts are used for generating continuations used in a model's distillation. These approaches all target absolute, provider-level detection; to the best of our knowledge, there is no prior work on reference-based LLM detection from model behavior alone, and we present this work as a first step in that direction.

Distillation detection in non-LLM settings Distillation detection has also been studied outside the LLM setting, particularly for image models. Prior work detects distilled image classifiers and generative models using synthesized inputs and statistical scoring (Shi et al., 2025), prompt-level

fingerprints in text-to-image models (Zhang et al., 2024), and shadow-model-based provenance verification comparing generalization behavior against suspicious models (Xie et al., 2025).

Membership Inference Our approach relies on membership inference, which determines whether a particular example was used during model training (Shokri et al., 2017), according to the intuition that training examples receive lower loss than non-training examples. Comparing against a reference output distribution has been shown to yield stronger membership inference signal (Carlini et al., 2022; Watson et al., 2022), and recent work extends membership inference to decide whether a model is trained on a dataset (Maini et al., 2024). We build on this literature but study a different problem: whether a model is trained on another model's generations.

3 METHOD

LLM Distillation We consider supervised fine-tuning (SFT)-based distillation. Let S_0 be a base student model and T a teacher model. Given inputs $\mathcal{X} = \{x_i\}_{i=1}^N$, where each x_i includes the prompt template used to query the teacher, the teacher produces outputs $y_i = T(x_i)$, yielding the distillation dataset $\mathcal{D}_T = \{(x_i, y_i)\}_{i=1}^N$; fine-tuning S_0 on $\mathcal{D}_T$ produces the distilled student S.

Distillation detection Given a subject model S and candidate teachers $\mathcal{T} = \{T_1, \ldots, T_K\}$, our goal is to determine whether S was distilled from any teacher in $\mathcal{T}$, and if so, identify the teacher. As an initial step, we assume S was distilled from at most one teacher in $\mathcal{T}$. We assume the detector has logit access to S, only black-box (sampling) access to $\mathcal{T}$, and no access to the prompts or templates used for distillation.

A natural first attempt is to detect distillation in an *absolute* sense: scoring candidate teacher outputs directly under S, via raw likelihood or standard membership-inference metrics. We find this unreliable, as log-likelihood can favor teacher outputs that are easy to model in general rather than specifically aligned with S (Table 1). We instead adopt a *reference-based* formulation: rather than asking whether S explains a teacher's outputs in absolute terms, we ask whether it explains them *better than* an earlier-generation reference checkpoint R does.

3.1 TEACHER IDENTIFICATION

We first consider the simpler setting of **teacher identification**, assuming a single true teacher exists in $\mathcal{T}$, as also studied in prior computer vision work (Shi et al., 2025): the task maps a student S given a candidate set $\{T_1, \ldots, T_K\}$ to the index of the true teacher: $(S, \{T_1, \ldots, T_K\}) \mapsto y \in [K]$. We construct proxy inputs $\hat{\mathcal{X}}$ from open-source data, which may differ from the original inputs $\mathcal{X}$, query each teacher T_k on every $x_i \in \hat{\mathcal{X}}$, and score each output with an alignment function $\mathbf{f}(T_k(x_i), S)$. In principle $\mathbf{f}$ may be any measure of alignment, including lexical or semantic similarity between $S(x_i)$ and $T_k(x_i)$ (n-gram overlap, embedding cosine); we find likelihood-based scoring more reliable and instantiate

$$\mathbf{f}(T_k(x_i), S) = -\big(\ell_S(x_i, y_i^{(k)}) - \ell_R(x_i, y_i^{(k)})\big),$$

where $\ell_M(x, y)$ is the average token-level negative log-likelihood of y given x under model M. Following reference-based membership inference (Watson et al., 2022), this calibrates the likelihood under S against a reference R that was ideally not exposed to any teacher candidate—an earlier same-family checkpoint, a prior training stage, or a smaller variant—and measures how much better the student explains a candidate teacher's output than the reference does. We ablate the choice of R in §B.1.

We identify the teacher using two complementary decision rules: (i) the highest mean alignment score across $\hat{\mathcal{X}}$, and (ii) sorting per-teacher scores and selecting the teacher that wins the largest fraction of rank-matched comparisons. We report **Per-sample** as the fraction of rank-matched comparisons won by the true teacher, and **Agg.** as the better of the two rules; formal definitions are in §A.

3.2 DISTILLATION DETECTION

Beyond identifying a single teacher, we consider the binary *distillation detection* problem: predict $y \in \{0, 1\}$ indicating whether S was distilled from any teacher in $\mathcal{T}$. We define the **margin** $\delta = \bar{s}_1 - \bar{s}_2$, where $\bar{s}_k = \frac{1}{N}\sum_{i=1}^N \mathbf{f}(T_k(x_i), S)$ and $\bar{s}_1 \geq \bar{s}_2$ are the top two mean scores among

candidate teachers. If the true teacher is present, it should dominate the pool, yielding a large δ; if absent, scores are more uniform and δ stays small. We select a threshold τ on a calibration set by trying each observed training margin as a cutpoint and choosing the value that maximizes F1, using only training examples and never held-out margins.

3.3 Statistical Significance

Both tasks output a point estimate, so for each we add a test that checks whether the signal is real rather than noise. For **teacher identification**, with the true teacher $t^\star$ known in advance, we count the probes on which $t^\star$ outscores every other candidate and report the win rate $\hat{\rho}$. If ranking were random $t^\star$ would top a probe with probability $1/K$, so we test $H_0 : \rho \leq 1/K$ against $H_1 : \rho > 1/K$ with a one-sided exact binomial test; because $t^\star$ is fixed in advance rather than selected as the best of K, this needs no multiple-comparison correction.

For **distillation detection**, let t_1 and t_2 denote the top- and second-ranked candidates by mean score, and define the paired per-probe difference $d_i = \mathbf{f}(T_{t_1}(x_i), S) - \mathbf{f}(T_{t_2}(x_i), S)$, whose sample mean recovers δ. We test $H_0 : \delta \leq \tau^*$ against $H_1 : \delta > \tau^*$ with a one-sided t-test, falling back to a Wilcoxon signed-rank test on $d_i - \tau^*$ when normality is rejected (Shapiro $p < 0.05$); we test against the calibrated τ^* rather than zero because at $N \approx 200$ even a negligible margin is significant against zero. Since t_1 is the best of K, we Bonferroni-correct as $p^* = \min(1,\, p \cdot (K-1))$, and call a target distilled from teacher T only if $p^* < 0.05$ on *both* prompt sets and both name the same teacher. Full definitions and justifications are in §C.

4 Experiment: Teacher Identification

We first validate our teacher identification method.

Disclaimer. Our main evaluation focuses on a canonical setting—single-teacher, SFT-style final-stage distillation. Real-world pipelines may involve multiple teachers, multi-stage use, and diverse roles (e.g., generation, rewriting); we report preliminary results relaxing the first two in §B.5 and leave a fuller treatment to future work.

4.1 Experimental Setup

Because modern LLM lineages are deeply entangled, ground-truth attribution is often ambiguous. We therefore use a hybrid evaluation: **controlled experiments**, where we distill models ourselves to isolate key variables, and **models in the wild**, where real-world distilled models test robustness under unknown prompts, instructions, filtering, and training pipelines.

Controlled Experiments Since most state-of-the-art LLMs may already have been distilled from each other, we select teacher–student pairs from distinct U.S.-organization model families, where legal constraints limit cross-company distillation (Philipp, 2026), and additionally include Qwen models to broaden coverage—though we acknowledge that prior distillation is harder to rule out in these cases.

We consider four base student models (S_0): Llama-3.2-3B-Instruct (Meta, a), Gemma-3-4B-PT (Gemma Team, 2025), Qwen2.5-1.5B, and Qwen2.5-3B (Qwen Team et al., 2025). We distill each from three teachers: Qwen3-8B (Qwen Team, 2025), Llama-3.3-70B-Instruct (Meta, b), and GPT-OSS-120B (OpenAI, 2025), using prompts from OpenMathInstruct-2 (OMI-2) and s1 (Toshniwal et al., 2024; Muennighoff et al., 2025), yielding $4 \times 3 \times 2 = 24$ distilled students. We retain the 19 for which distillation improves utility over the base model. At detection time, we additionally include Gemma-3-27B-IT (Gemma Team, 2025) as a candidate teacher, forming a four-way classification task.

Real-World Distilled Models We consider (1) six DeepSeek-R1 distill models officially distilled by DeepSeek;[1] (2) `s1.1-32B`,[2] distilled from R1-generated datasets; and (3) X-Coder-SFT-Qwen3-

[1] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-7B#deepseek-r1-distill-models

[2] https://huggingface.co/simplescaling/s1.1-32B

Method	Llama 3.3 70B Instruct		Qwen-3-8B		GPT-OSS-120B	
	Agg.	Per-sample	Agg.	Per-sample	Agg.	Per-sample
Random	25.0	25.0	25.0	25.0	25.0	25.0
Length Matching Baseline	42.9	58.5	33.3	19.1	33.3	23.2
Non Reference-Based Methods						
String-based						
Multi n-gram Jaccard	100.0	87.6	33.3	27.3	0.0	0.5
Symmetric Coverage	85.7	77.0	33.3	40.4	0.0	0.2
Embedding Cosine	100.0	95.3	16.7	11.2	0.0	3.4
Logit-based						
Raw likelihood	28.6	37.6	0.0	2.4	0.0	0.0
ZLIB-normalized	100.0	100.0	0.0	0.8	0.0	0.0
Min-k	28.6	45.0	0.0	4.1	0.0	0.0
Reference-Based Methods						
String-based						
Multi n-gram Jaccard	87.5	61.0	66.7	54.6	16.7	31.6
Symmetric Coverage	75.0	58.4	83.3	53.5	16.7	37.0
Embedding Cosine	50.0	43.3	66.7	59.2	0.0	22.9
Raw likelihood (Ours)	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**99.9**

Table 1: **Teacher identification accuracy (%) in controlled experiments**. Accuracy is broken down by true teacher model, including both aggregated and per-sample metrics. Our method, which is the reference-based variant of likelihood, achieves near-perfect accuracy.

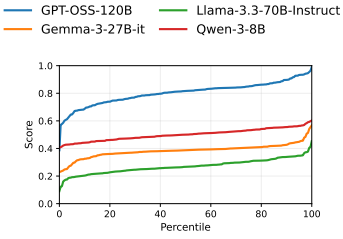


Figure 2: **Reference-based MIA CDFs.** Qwen-2.5-3B distilled from GPT-OSS-120B on s1, probed with OMI.

True teacher	n	p range vs chance	Sig.
GPT-OSS-120B	6	3.9×10^{-121}–2.3×10^{-118}	6/6
Qwen-3-8B	6	3.9×10^{-121}–1.4×10^{-113}	6/6
Llama-3.3-70B-Instruct	7	3.9×10^{-121}–3.5×10^{-71}	7/7
Overall	19	—	19/19

Table 2: **Per-probe top-1 significance in controlled experiments.** p ranges are across students for each teacher; n varies because we exclude distilled students that underperform their base model. All 19 are significant.

8B,[3] a code-focused model trained via SFT on synthetic code data from multiple reasoning models (including `DeepSeek-R1` and `Qwen-3-235B-A22B`) with prompts from `o3-mini`.

For the pool of teacher candidates, we consider 10 models: Claude Opus 4.5, Claude Opus 4.6, Claude 3.5 Sonnet, DeepSeek R1, GPT-OSS-120B, Gemma-3-27B-it, Llama-3.3-70B-Instruct, QwQ-32B-Preview, o1, and o3—forming a ten-way classification task.

Detection method details. We use the pre-distilled base model of each student as the reference R (ablated in §B.1), and truncate all teacher outputs to 2,048 tokens so that output length cannot trivialize detection. We assume no access to the original distillation inputs $\mathcal{X}$ and probe with disjoint prompts $\hat{\mathcal{X}}$ (s1 for OMI-distilled students, and vice versa); for the R1-distilled models, whose prompts are unknown, we use both. We compare against length-matching, string-based (multi n-gram Jaccard, symmetric coverage, embedding cosine), and membership-inference baselines (raw likelihood, ZLIB-normalized, Min-k, and reverse variants); §A gives full configurations, distillation hyperparameters, and baseline definitions.

4.2 Results from Controlled Experiments

Table 1 reports results across the 19 retained distilled students. Our reference-based method achieves near-perfect accuracy in every setting, while baselines are inconsistent—performing well only when Llama 3.3 70B Instruct is the teacher and failing on Qwen-3-8B and GPT-OSS-120B.

[3] https://huggingface.co/IIGroup/X-Coder-SFT-Qwen3-8B

Method	Default instructions		Highly customized instructions	
	Agg.	Per-sample	Agg.	Per-sample
Random	25.0	25.0	25.0	25.0
Ours (Reference-based Raw likelihood)	100.0	100.0	50.0	65.9
+ In-context exemplars from S	N/A	N/A	75.0	67.1

Table 3: Comparison of reference-based teacher identification under highly customized instructions during distillation (unknown to the detector), with and without in-context exemplars.

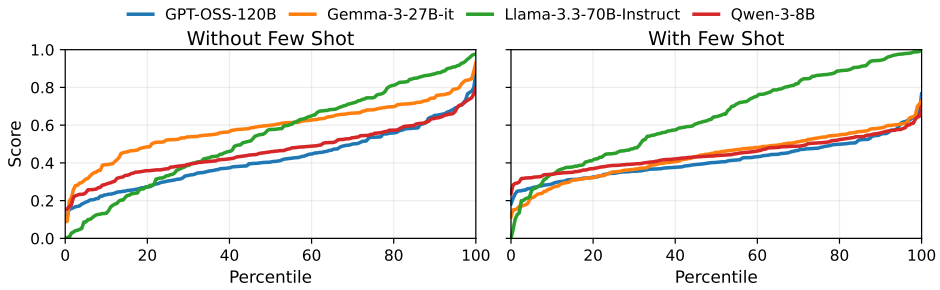


Figure 3: Results on a model distilled from Llama 3.3 70B Instruct, using highly customized instructions (unknown to the detector). While the default method struggles to identify the true teacher (left), few-shot prompting improves teacher identification (right).

Figure 2 shows a representative controlled example: a Qwen-2.5-3B-Base student distilled from GPT-OSS-120B on s1 prompts. The true teacher is clearly separated under the non-training data probes (OMI), and we observe the same qualitative pattern across all controlled student–teacher pairs.

Statistical significance testing in Table 2 confirms that the true teacher's per-probe advantage is not attributable to sampling noise: across all 19 students, the true teacher significantly exceeds the $1/K = 0.25$ chance baseline ($p = 3.5 \times 10^{-71}$ to 3.9×10^{-121}).

Effect of highly customized instructions in distillation detection Toshniwal et al. (2024) introduce a specialized prompt format (OMI COT) that they show substantially alters Llama Instruct response style, making this a challenging setting where prompt-induced stylistic artifacts may obscure the teacher signal. Following Toshniwal et al. (2024), we prompt Llama 3.3 70B Instruct with the OMI COT template, distill students on this data, and test whether reference-based MIA recovers the teacher using s1 as probing data (since OMI was used for training). Detection accuracy degrades substantially under this distribution shift, falling from 100.0 to 50.0 Agg. (65.9 Per-sample).

To address this, we introduce a **simple few-shot prompting strategy**: for each candidate teacher, we prepend a small number of student-generated (prompt, response) examples under the default template to better align teacher outputs with the student's style. As shown in the last row of Table 3, this improves accuracy, recovering the true teacher for three of four student models (the exception being Gemma-3-4B-pt). Figure 3 illustrates the improved separation between the true teacher and alternatives, with additional CDFs shown in §D.

§B reports additional ablations on reference-model choice and proxy-prompt mismatch. Performance is generally robust to reference choice, though for instruction-tuned students, instruction-tuned references are more effective than base ones; identification is also unaffected even when the reference comes from a different family and organization. It also remains stable when proxy prompts differ substantially from the original distillation inputs, including cross-domain settings such as math versus code.

4.3 Results on Real-World Distilled Models

Results on DeepSeek-R1 distill models and s1.1-32B As reported in Table 4, we find our method correctly identifies the true teacher among ten candidates for seven out of the seven evaluated models, both over OMI and s1 probing sets, showing that our method works when distillation data and training procedure are largely unknown. The same holds when all candidates are restricted to final answers only, showing our method does not require access to reasoning traces.

Method	Avg. (7 models)	
	Agg.	Per-samp.
Random	–	10.0
Non Reference-Based Methods		
Multi n-gram Jaccard	33.3	49.5
Symmetric Coverage	16.7	40.3
Reference-Based Methods		
Multi n-gram Jaccard	100.0	56.8
Symmetric Coverage	83.3	41.1
Raw likelihood (final answers only)	100.0	86.8
Raw likelihood (Ours)	**100.0**	**97.2**

Table 4: **Teacher identification accuracy (%) for real-world distilled models (OMI probe set).** Averaged over the six R1-Distill models plus s1.1-32B. The final-answers-only row omits QwQ-32B-Preview (no clean trace/answer split).

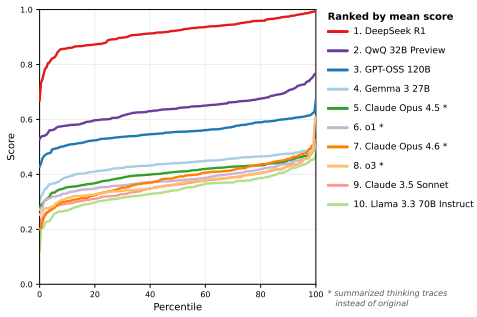


Figure 4: Results for `s1.1-32B`, a real-world model trained by Muennighoff et al. (2025). The true teacher model, R1, is ranked at the top by a large margin, with the other candidates clustered below. The separation holds across the full percentile range.

The same per-probe test confirms this at the level of individual probes (Table 5): for all seven targets, on both prompt sets, DeepSeek-R1's per-probe win rate is statistically significantly above the $1/K$ chance baseline.

True teacher	p range vs chance
DeepSeek-R1	$1.0 \times 10^{-200} - 3.4 \times 10^{-138}$

Table 5: **Per-probe top-1 significance on real-world distilled models.** p ranges are across the seven targets and both prompt sets, all of which are significant.

The CDF graph for `s1.1-32B` is provided in Figure 4 for reference, showing that the true teacher (R1) is clearly separated from other candidates.

Additional CDFs for the remaining real-world models, including the `X-Coder-SFT-Qwen3-8B` results in Figure 15, are provided in §E.

5 Experiment: Distillation Detection

This section evaluates how the margin-threshold detector from §3.2 generalizes beyond the models used to calibrate it. Selecting τ to maximize controlled accuracy yields $\tau = 0.067$, which attains 85.5% $(65/76)$ on the controlled set but only 53.6% $(15/28)$ on the real-world cells—a single fixed threshold does not transfer cleanly between settings. We therefore isolate two forms of generalization: transfer to an unseen *student* whose teacher was seen during calibration, and transfer to an unseen *teacher*.

5.1 Experimental Setup

Each evaluation cell is $(S, \{T_1, \ldots, T_K\}, y)$, where $y = 1$ if the true teacher is present in the candidate pool and $y = 0$ otherwise. We use the 19 controlled students from §4.1, each evaluated under OMI and s1 prompts with the true teacher included (K=4) or excluded (K=3), yielding $19 \times 2 \times 2 = 76$ cells; and six DeepSeek-R1 distilled models plus s1.1-32B, evaluated the same way with 10 or 9 candidates, yielding $7 \times 2 \times 2 = 28$ held-out cells.

Cross-Validation. In every cross-validation fold we refit τ on the training cells only, selecting the value that maximizes F1. For the controlled models we use two splits: *leave-one-student-out*, which holds out one student and leaves all teachers represented, and *leave-one-teacher-out*, which holds out every student distilled from one teacher. The real-world models all share DeepSeek-R1 as their teacher, so we perform only leave-one-student-out. For each held-out cell we report margin-prediction accuracy ($\delta \geq \tau$) and apply the detection test from §3.3 with the fold's τ.

True teacher	Detected	Detection p^*	Acc. (S-out)	Acc. (T-out)	FP (S-out)	FP (T-out)
GPT-OSS-120B	$6/6$	$<10^{-26}$	$24/24$	$15/24$	$0/6$	$4/6$
Qwen-3-8B	$6/6$	$<10^{-20}$	$22/24$	$15/24$	$0/6$	$1/6$
Llama-3.3-70B-Instruct	$1/7$	6×10^{-34}–1.0	$17/28$	$16/28$	$0/7$	$0/7$
Overall	$13/19$	—	$63/76$	$46/76$	$0/19$	$5/19$

Table 6: **Controlled cross-validation by true teacher.** *Detected* and *Detection* p^* are the detection test on the true-teacher-present cells, using the leave-one-student-out fold τ; *FP* is a substitute falsely called when the true teacher is removed, under the S-out τ (FP S-out) or T-out τ (FP T-out). *Acc.* is the margin prediction ($\delta \geq \tau$) per cell with the corresponding fold τ. S-out = leave-one-student-out, T-out = leave-one-teacher-out.

5.2 Results

Generalization to unseen students versus unseen teachers. Table 6 breaks the controlled cross-validation down by true teacher. The two reasoning teachers (GPT-OSS-120B and Qwen-3-8B) are detected for all their students under both splits, whereas the non-reasoning Llama-3.3-70B-Instruct is detected for only $1/7$ regardless of split. The splits diverge in specificity, not detection: leaving a student out produces no false detections, whereas leaving a teacher out produces them on the reasoning teachers' teacher-removed cells ($4/6$ and $1/6$), and overall accuracy falls from 82.9% to 60.5%. Detection seems to generalize better to unseen students when their teacher is represented in calibration, and is less reliable across unseen teachers, though our setup is limited to three controlled teachers.

Generalization across real-world students. The real-world experiment reproduces the student-side pattern. With τ refit on the other six R1-distilled models in each fold, leave-one-student-out attains 92.9% $(26/28)$ accuracy (Table 15), calling six of seven held-out students distilled from DeepSeek-R1; when DeepSeek-R1 is removed, no alternative ever clears the threshold, giving zero false detections. The single missed positive is s1.1-32B, where R1 ranks first on both prompts but the OMI margin falls below the fold-fit threshold, so the both-prompts rule withholds the call (§F).

6 OpenAI o1 Detection Methods

Method. We serialize each `o1` output in two ways: Unicode, preserving raw UTF-8 characters, and ASCII, escaping each non-ASCII codepoint. We then run our reference-MIA method with all other hyperparameters fixed and define $\Delta_{\text{ASCII}} = \overline{\mathcal{L}}_{\text{ASCII}} - \overline{\mathcal{L}}_{\text{Unicode}}$, where positive values indicate that ASCII normalization raises reference-normalized loss. We evaluate four `o1`-distilled students trained on s1 using 100 held-out OMI probes, and compare against real U.S.-organization control models not expected to be distilled from `o1`/`o3`. To assess significance we reuse the statistical test (§3.3) with the null placed at zero; additional details, controls, and wild-model results are in §G.

Results. Figure 5 and Table 16 show that `o1`-distilled students have consistently positive ASCII gaps ($\Delta_{\text{ASCII}} \in [0.0573, 0.0732]$; $p < 10^{-15}$), while controls mostly remain near zero. Three Gemma-3 controls also reach significance, but with much smaller gaps; with only 100 probes a p-value can resolve even a tiny positive Δ_{ASCII} from zero, so what matters is the size of the gap rather than significance alone: the smallest distilled gap ($+0.0573$) exceeds the largest control gap ($+0.0373$) by over 50%. Thus, Unicode features inherited from `o1` can serve as a complementary membership signal under controlled distillation, though with only 100 OMI probes from a single prompt set we view it as a cue rather than standalone evidence.

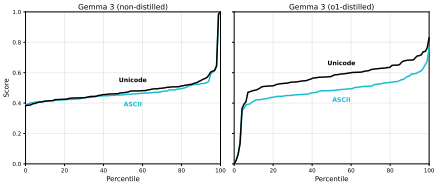


Figure 5: **Per-probe CDFs.** The non-distilled `gemma-3-27b-pt` control (left) has nearly overlapping ASCII and Unicode curves; the `o1`-distilled `gemma-3-4b-pt` student (right) separates consistently.

7 Open Questions

We explore the applicability of reference-based MIA to open-ended settings where it is unknown whether a model was distilled from a third-party model. Specifically, we ask whether **QwQ-32B**,

GPT-OSS, and **DeepSeek-R1** show evidence of distillation from any of our candidate teachers (Llama, QwQ, o1/o3, Claude, DeepSeek-R1, etc).

Disclaimer. Our validation relies on limited evaluations and on simplifying assumptions—primarily single-teacher, SFT-style final-stage distillation—though preliminary results suggest the method may extend to multi-teacher and multi-stage settings (§B.5). The candidate teachers we consider are also themselves likely deeply connected, often building on one another through complex dependency chains (Adhikesaven et al., 2026). Results here should therefore be read as preliminary exploration rather than definitive evidence.

Candidate pools and references. For each target, we use the same candidate-teacher pool as in §4.3, excluding near-variants of the target to avoid trivial same-family matches: we exclude `GPT-OSS-120B` when evaluating `GPT-OSS-20B`, and `QwQ-32B-Preview` when evaluating `QwQ-32B`. We include one representative figure per experiment; additional plots and statistical analyses are in §H.

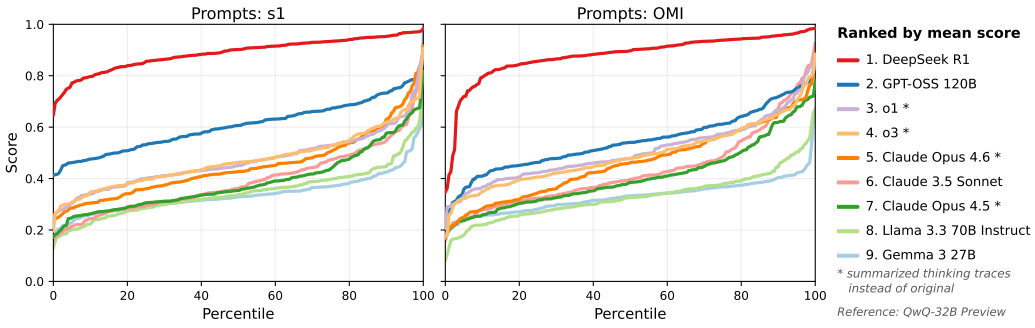


Figure 6: Our reference-based method applied to **`QwQ-32B`**, using `QwQ-32B-Preview` as a reference. Results may suggest `QwQ-32B` was distilled from R1 after the release of `QwQ-32B-Preview`.

QwQ-32B. We evaluate `QwQ-32B` using `QwQ-32B-Preview` as the reference model. As shown in Figure 6, `DeepSeek-R1` ranks highest across both probing sets, with a pronounced gap from the next candidate. Combined with the release timeline—`QwQ-32B-Preview` in November 2024, `DeepSeek-R1` in January 2025, and `QwQ-32B` in March 2025—this raises the possibility that `DeepSeek-R1` outputs were used to train `QwQ-32B` following the release of `QwQ-32B-Preview`.

DeepSeek-R1 and GPT-OSS. Neither target yields a clean attribution. For `DeepSeek-R1`, using `DeepSeek-MoE-16B` as the reference model, our method ranks `o1` and `QwQ-32B-Preview` near the top (Figure 17). For `GPT-OSS`, which unlike `QwQ` and `DeepSeek-R1` has no suitable same-family checkpoint to serve as a reference, `DeepSeek-R1` and `QwQ-32B-Preview` are ranked highest with `GPT-2 XL` as the reference (Figure 18). Both rankings are hard to interpret, as these candidates may themselves share lineage. Our ASCII–Unicode diagnostic gives a clearer signal: both targets show large Unicode–ASCII gaps ($\delta_{\text{Uni}-\text{ASCII}} = +0.937$ and $+0.83$; $p < 10^{-15}$; Table 19), suggesting possible `o1`-style signal. Full results are in §H.

8 Conclusion

We introduce reference based distillation detection: identifying the teacher used to train a model given an earlier-generation checkpoint from the same lineage. By comparing normalized likelihood differences between a student and reference model, our approach identifies candidate teachers whose outputs are preferentially aligned with the student. Across controlled and real-world settings—including DeepSeek-R1 distills and s1.1—our method consistently identifies the true teacher and outperforms similarity-based baselines, while remaining effective under cross-domain evaluation. Overall, our work provides an initial step toward practical auditing of distillation in modern LLMs.

AI USE STATEMENT

We used generative AI tools to help write code for training, evaluation, and running our experiments; our code and its outputs were reviewed by the authors afterward. Generating synthetic data with LLMs is also part of our method: the distillation training sets and the candidate-teacher outputs used for reference-based detection were generated by language models. We additionally used generative AI tools to help build figures and tables from measured experimental results, to help outline the paper, and to polish writing after the authors wrote initial drafts. We did not use generative AI to propose hypotheses, design the research methodology, or interpret results. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide full details of our method, datasets, and evaluation setup in the main paper, with additional reproducibility details (base/teacher/reference models, distillation hyperparameters, output truncation, significance tests, etc.) in §A and §C.

REFERENCES

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Sanjay Adhikesaven, Haoxiang Sun, and Sewon Min. Which models are our models built on? auditing invisible dependencies in modern llms, 2026. URL https://arxiv.org/abs/2606.12385.

Anthropic. Detecting and preventing distillation attacks, 2026. URL https://www.anthropic.com/news/detecting-and-preventing-distillation-attacks.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy*, 2022.

Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. URL https://arxiv.org/abs/2503.19786.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Xiaoqi Jiao, Yichun Yin, Shang Lifeng, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

Pratyush Maini, Hengrui Jia, Nicolas Papernot, and Adam Dziedzic. Llm dataset inference: Did you train on my dataset?, 2024. URL https://arxiv.org/abs/2406.06443.

Meta. meta-llama/llama-3.2-3b-instruct. https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct, a. Hugging Face model card, accessed 2026-04-05.

Meta. Llama 3.3 | model cards and prompt formats. https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_3/, b. Model card for Llama 3.3 70B Instruct, accessed 2026-04-05.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

NBC News. Google says attackers used 100,000+ prompts to try to clone ai chatbot gemini, 2026. URL https://www.nbcnews.com/tech/security/google-gemini-hit-100000-prompts-cloning-attempt-rcna258657.

Team Olmo, :, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2026. URL https://arxiv.org/abs/2512.13961.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025. URL https://arxiv.org/abs/2508.10925.

Leyi Pan, Aiwei Liu, Shiyu Huang, Yijian Lu, Xuming Hu, Lijie Wen, Irwin King, and Philip S. Yu. Can llm watermarks robustly prevent unauthorized knowledge distillation?, 2025. URL https://arxiv.org/abs/2502.11598.

Claudia Philipp. From prompt to clone: Copyright challenges in ai model distillation. *UC Law Science and Technology Journal*, 17(1):49, 2026.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025. URL https://arxiv.org/abs/2412.15115.

Tom Sander, Pierre Fernandez, Alain Durmus, Matthijs Douze, and Teddy Furon. Watermarking makes language models radioactive, 2024. URL https://arxiv.org/abs/2402.14904.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Deepa Seetharaman and Fabiola Arámburo. Openai says china's deepseek trained its ai by distilling us models, memo shows, 2026. URL https://www.reuters.com/world/china/openai-accuses-deepseek-distilling-us-models-gain-advantage-bloomberg-news-2026-02-12/.

Qin Shi, Amber Yijia Zheng, Qifan Song, and Raymond A Yeh. Knowledge distillation detection for open-weights models. *arXiv preprint arXiv:2510.02302*, 2025.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy*, 2017.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7, 2023.

Shubham Toshniwal, Wei Du, Ivan Moshkov, Branislav Kisacanin, Alexan Ayrapetyan, and Igor Gitman. Openmathinstruct-2: Accelerating ai for math with massive open-source instruction data. *arXiv preprint arXiv:2410.01560*, 2024.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

Lauren Watson, Chuan Guo, Graham Cormode, and Alex Sablayrolles. On the importance of difficulty calibration in membership inference attacks, 2022. URL https://arxiv.org/abs/2111.08440.

Yuechen Xie, Jie Song, Huiqiong Wang, and Mingli Song. Training data provenance verification: Did your model use synthetic data from my generative model for training?, 2025. URL https://arxiv.org/abs/2503.09122.

Hengxiang Zhang, Hyeong Kyu Choi, Sharon Li, and Hongxin Wei. Detecting distillation data from reasoning models, 2025. URL https://arxiv.org/abs/2510.04850.

Likun Zhang, Hao Wu, Lingcui Zhang, Fengyuan Xu, Jin Cao, Fenghua Li, and Ben Niu. Training data attribution: Was your model secretly trained on data created by mine?, 2024. URL https://arxiv.org/abs/2409.15781.

A EXPERIMENTAL DETAILS

Base student models (S_0) We consider Llama-3.2-3B-Instruct (Meta, a), Gemma-3-4B-PT (Gemma Team, 2025), and Qwen2.5-1.5B and Qwen2.5-3B (Qwen Team et al., 2025) as base student models.

Teacher models ($\mathcal{T}$) Our teacher candidate set includes Llama-3.3-70B-Instruct (Meta, b),[4] Qwen3-8B (Qwen Team, 2025), Gemma-3-27B (Gemma Team, 2025), and GPT-OSS-120B (OpenAI, 2025).

These four models together form the pool of candidate teachers for the teacher identification task, allowing each to act as both a positive and a negative candidate. Consequently, the task is formulated as a four-way classification task. To prevent our task from becoming trivial due to substantial variation in teacher output lengths, we standardize teacher outputs prior to distillation by truncating each output to at most 2,048 tokens.

Inputs ($\mathcal{X}$) We use two open-source prompt datasets: OpenMathInstruct-2 (OMI-2), a dataset of math problems and reasoning questions, and s1, another set of reasoning-heavy prompts used for test-time scaling (Toshniwal et al., 2024; Muennighoff et al., 2025).

We create distilled student models by querying each teacher with input prompts, and fine-tuning each student on the resulting teacher-generated data. In total, this setup produces 4 base student models $\times$ 3 teacher models $\times$ 2 prompt datasets, yielding 24 distilled models. We then retain models only when distillation leads to performance improvements, outperforming their corresponding base models,[5] resulting in 19 models.

Baseline detection methods We consider the following baselines. (1) **Raw likelihood**, which averages the token-level log-likelihood of $T_k(x_i)$ under S. (2) **ZLIB-normalized likelihood**, which normalizes likelihood using compression-based length estimates. (3) **Min-k% probability**, which focuses on the lowest-probability tokens in the sequence. (4) **Similarity-based scoring**, which compares teacher and student outputs using lexical overlap (e.g., n-gram Jaccard) and embedding-based similarity. Finally, we compare to our primary method, **reference-based MIA**, which normalizes likelihood using a reference model. We also evaluate a **reverse MIA** variant, where we instead generate outputs from the student $S(x_i)$ and score them under each candidate teacher model T_k. Results are shown in Table 12.

Evaluation Metrics We evaluate teacher identification using two complementary decision rules. First, we report a sorted per-sample score: for each candidate teacher, we sort its alignment scores over the probe set and compare candidates at matched ranks, measuring how often the true teacher obtains the highest alignment score (Per-sample). Formally, letting $\mathbf{f}^{\downarrow}_{T_j,r}$ denote the r-th largest alignment score for teacher T_j over $\hat{\mathcal{X}}$:

$$\hat{T}_{\text{sorted}} = \arg\max_{T_k \in \mathcal{T}} \frac{1}{N} \sum_{r=1}^{N} \mathbf{1}\left[T_k = \arg\max_{T_j \in \mathcal{T}} \mathbf{f}^{\downarrow}_{T_j,r} \right].$$

This captures whether the true teacher is favored distributionally, rather than requiring it to win on the same prompt. Second, we report aggregate identification accuracy by applying both decision rules at the model level: selecting the teacher with the highest mean alignment score,

$$\hat{T}_{\text{mean}} = \arg\max_{T_k \in \mathcal{T}} \frac{1}{N} \sum_{i=1}^{N} \mathbf{f}(T_k(x_i), S),$$

and selecting the teacher that wins most often under the sorted per-sample comparison. We then report the higher accuracy between these two aggregate decision rules (Agg.).

We report the models used in the main paper for our controlled experiments (Table 7), including only settings where distillation improves utility over the base model on both GSM8K and MATH500. When using our highly customized prompt template (OMI COT), we include all distilled models, observing gains in at least one of GSM8K or MATH500.

[4] We use the quantized version from `nvidia/Llama-3.3-70B-Instruct-NVFP8`.

[5] We use GSM8K and MATH 500 to evaluate whether distillation improves performance. Detailed results are reported in Table 7..

ID	Student	Teacher	Data	Template	Trace	GSM8K	MATH500
B1	Qwen-2.5-1.5B (Base)	Baseline	–	–	–	67.25	32.6
	Qwen-2.5-1.5B	Nvidia-Llama-3.3-70B-Instruct	OMI-2 (1K)	None	No	68.68	34.0
	Qwen-2.5-1.5B	Nvidia-Llama-3.3-70B-Instruct	s1 (1K)	None	No	69.44	34.0
	Qwen-2.5-1.5B	Nvidia-Llama-3.3-70B-Instruct	OMI-2 (918)	OMI COT	No	69.00	38.0
	Qwen-2.5-1.5B	Qwen-3-8B	OMI-2 (1K)	None	Yes	68.40	37.4
	Qwen-2.5-1.5B	Qwen-3-8B	s1 (1K)	None	Yes	69.75	36.8
	Qwen-2.5-1.5B	GPT-OSS-120B	OMI-2 (1K)	None	Yes	69.59	39.8
	Qwen-2.5-1.5B	GPT-OSS-120B	s1 (1K)	None	Yes	69.67	40.6
B2	Qwen-2.5-3B (Base)	Baseline	–	–	–	75.82	39.8
	Qwen-2.5-3B	Nvidia-Llama-3.3-70B-Instruct	OMI-2 (1K)	None	No	76.87	43.2
	Qwen-2.5-3B	Nvidia-Llama-3.3-70B-Instruct	s1 (1K)	None	No	76.40	42.6
	Qwen-2.5-3B	Nvidia-Llama-3.3-70B-Instruct	OMI-2 (918)	OMI COT	No	77.00	47.6
	Qwen-2.5-3B	Qwen-3-8B	OMI-2 (1K)	None	Yes	79.98	47.6
	Qwen-2.5-3B	Qwen-3-8B	s1 (1K)	None	Yes	79.37	45.6
	Qwen-2.5-3B	GPT-OSS-120B	OMI-2 (1K)	None	Yes	79.15	51.0
	Qwen-2.5-3B	GPT-OSS-120B	s1 (1K)	None	Yes	79.90	53.4
B3	Llama-3.2-3B-Instruct (Base)	Baseline	–	–	–	76.57	42.0
	Llama-3.2-3B-Instruct	Nvidia-Llama-3.3-70B-Instruct	OMI-2 (1K)	None	No	77.25	54.2
	Llama-3.2-3B-Instruct	Nvidia-Llama-3.3-70B-Instruct	OMI-2 (918)	OMI COT	No	77.71	40.8
	Llama-3.2-3B-Instruct	Nvidia-Llama-3.3-70B-Instruct	s1 (1K)	None	No	76.88	48.0
B4	Gemma-3-4B-pt (Base)	Baseline	–	–	–	38.13	23.6
	Gemma-3-4B-pt	GPT-OSS-120B	s1	None	Yes	53.37	29.4
	Gemma-3-4B-pt	GPT-OSS-120B	OMI-2 (1K)	None	Yes	51.17	27.6
	Gemma-3-4B-pt	Nvidia-Llama-3.3-70B-Instruct	s1	None	No	50.57	27.4
	Gemma-3-4B-pt	Nvidia-Llama-3.3-70B-Instruct	OMI-2 (918)	OMI COT	No	50.27	23.0
	Gemma-3-4B-pt	Qwen-3-8B	s1	None	Yes	51.78	28.05
	Gemma-3-4B-pt	Qwen-3-8B	OMI-2 (1K)	None	Yes	48.59	28.2

Table 7: Student model training results for the controlled distillation setups.

Distillation hyperparameters For each (student, teacher, prompt dataset) combination, we fine-tune the base student S_0 on the teacher-generated dataset $\mathcal{D}_T$ using supervised fine-tuning. We use identical hyperparameters across all controlled distillation runs: 3 epochs, learning rate $1e{-5}$ with cosine schedule and 5% warmup, per-device batch size 4 with gradient accumulation 4 (effective batch size 16), block size 4,096, bf16 precision, and gradient checkpointing. Loss is computed on response tokens only; prompt tokens are masked with -100.

For Llama-3.2-3B-Instruct, we apply the model's native chat template (`apply_chat_template`) to format each (question, response) pair, and identify the prompt span via the prompt-only tokenization with `add_generation_prompt=True` so that supervision is restricted to the assistant turn. For the non-instruct base students (Gemma-3-4B-PT, Qwen-2.5-1.5B, Qwen-2.5-3B), we use a simple `Problem:\n{question}\n\nSolution:\n` template. Teacher responses are truncated to 2,048 tokens prior to training to prevent variation in teacher output length making our detection task easier, and the same truncation is applied when computing reference-based MIA scores at evaluation time.

Compute resources We ran all training and evaluation on H200 GPUs. Each controlled distillation run uses 2×H200, as do all reference-based MIA evaluations and baselines (similarity-based, logit-based, reverse MIA, ablations, and the o1 ASCII-vs-Unicode diagnostic). Larger candidate teachers were run on the same 2×H200 setup, except for DeepSeek-R1, which we ran locally on 8×H200 due to its size. Outputs from API-only models were generated through their respective APIs and did not require local GPUs.

We additionally report descriptive statistics for teacher output lengths in Table 8. These statistics provide context for our controlled distillation setup, since teacher models differ substantially in output verbosity, especially when full reasoning traces are included. For models with explicit reasoning traces, we also report final-output-only lengths to separate reasoning verbosity from the final answer length.

B Additional Experimental Results

B.1 Ablations on the choice of reference model (R)

We ablate the choice of reference model R to assess its impact on teacher identification (Table 9). Using Llama 3.2-3B-Instruct students, we vary R across related base and instruction-tuned models.

Model	Avg Words (with s1 prompts)	Avg Words (with OMI prompts)
Llama-3.3-70B-Instruct (Final output only)	524.6	402.7
Gemma-3-27B-it (Final output only)	565.5	419.5
Claude-3.5-Sonnet (Final output only)	289.4	229.7
Claude Opus 4.5	2776.7	1838
Claude Opus 4.6	1646	1398.1
GPT-OSS-120B	1800.1	1032.5
o1	840.2	687.1
o3	658.6	758.9
DeepSeek R1	5085.7	3577.6
DeepSeek R1 (Final output only)	284.1	225.0
Qwen-3-8B	8369.9	5708.1
Qwen-3-8B (Final output only)	517.0	385.3
QwQ-32B Preview	4701.0	3301.0

Table 8: **Average output length in words across teacher models.** We report the average number of words generated by each model under s1 prompts and OMI prompts.

Reference model R	**Llama 3.3-70B-Instruct as teacher**		**GPT-OSS-120B as teacher**	
	Agg.	**Per-sample**	**Agg.**	**Per-sample**
Llama 3.2-3B-Instruct (default)	100.0	100.0	100.0	100.0
Llama 3.2-3B	100.0	100.0	0.0	1.5
Llama 3.1-8B-Instruct	100.0	99.5	100.0	100.0
Llama 3.1-8B	100.0	100.0	0.0	17.5

Table 9: **Effect of the reference model R on teacher identification**, when the base student model is Llama 3.2 3B Instruct. We note that using GPT-OSS-120B as a teacher did not lead to major utility, but we include it for diversity in this ablation.

Overall, performance is reasonably robust to the choice of R; however, using base (non-instruction-tuned) models degrades accuracy. This suggests that although the method is reasonably stable, matching the instruction-tuning of the student is important for reliable identification.

The ablation above varies R only within the Llama family. To test whether the reference must share the student's lineage at all, we re-ran the method on all 19 controlled students using a reference from a different family *and* a different organization: Gemma-3-4B-IT for Llama-3.2-3B-Instruct students, Llama-3.2-3B for Gemma-3-4B-PT, Falcon3-3B-Base for Qwen2.5-3B, and SmolLM2-1.7B for Qwen2.5-1.5B. Everything else is unchanged, including cross-domain probing and the four-way candidate pool. Teacher identification is essentially unaffected (Table 11): 100.0 Agg. for every teacher, with Per-sample at or near 100. The method therefore requires a *reasonable* reference, not a same-family one.

B.2 Ablations on proximity in input prompts ($\hat{\mathcal{X}}$)

To quantify the effect of input prompt approximation—i.e., the mismatch between the true distillation prompts $\mathcal{X}$ and the prompts that a detector has access to $\hat{\mathcal{X}}$—we consider three settings:

1. **Identical**: $\mathcal{X}$ and $\hat{\mathcal{X}}$ are identical, e.g., both are s1 prompts, or both are OMI prompts.
2. **Similar-domain**: $\mathcal{X}$ and $\hat{\mathcal{X}}$ are not identical but are from similar domains, e.g., ($\mathcal{X}$ =s1, $\hat{\mathcal{X}}$ =OMI), or ($\mathcal{X}$ =OMI, $\hat{\mathcal{X}}$ =s1). This was the default setting in our main experiments.
3. **Different-domain**: $\mathcal{X}$ and $\hat{\mathcal{X}}$ are from substantially different domains, e.g., $\mathcal{X}$ is s1 or OMI (math), while $\hat{\mathcal{X}}$ consists of prompts for code generation.[6]

We follow the same setup as in the main experiments: models are distilled using four base students, four candidate teachers, and two prompt sets (s1 or OMI), and the detection task is to identify the true teacher among four candidates. Table 10 reports the results across these three proxy-input settings. Our method remains accurate even when the detector uses prompts from a different domain (code

[6]We took input prompts from MBPP (Austin et al., 2021)

Method	Identical		Similar-domain		Different-domain	
	Agg.	Per-sample	Agg.	Per-sample	Agg.	Per-sample
Random	25.0	25.0	25.0	25.0	25.0	25.0
Length Matching Baseline	36.8	32.8	36.8	34.9	42.1	42.2
String-based methods						
Multi n-gram Jaccard	47.4	43.5	47.4	41.1	47.4	42.2
Symmetric Coverage	47.4	46.4	42.1	41.2	52.6	44.3
Embedding Cosine	42.1	41.8	42.1	39.7	36.8	30.8
Logit-based methods						
Raw likelihood	15.8	18.4	10.5	14.6	36.8	36.8
ZLIB-normalized	36.8	40.3	36.8	37.1	36.8	36.8
Min-k	21.1	19.9	10.5	17.9	36.8	36.8
Reverse raw likelihood	47.4	43.7	47.4	40.3	52.6	45.3
Reverse ZLIB-normalized	42.1	43.1	42.1	41.7	42.1	45.4
Reverse Min-k	42.1	41.6	36.8	39.1	57.9	45.6
Ours (Reference-based)	100.0	100.0	100.0	100.0	100.0	98.8

Table 10: **Impact of the choice of proxy input prompts** ($\hat{\mathcal{X}}$). 'Identical' indicates $\mathcal{X}$ and $\hat{\mathcal{X}}$ are identical; 'Similar domain' indicates $\mathcal{X}$ and $\hat{\mathcal{X}}$ are not identical but are from similar domains; and 'Different domain' indicates $\mathcal{X}$ and $\hat{\mathcal{X}}$ are from substantially different domains. Detection accuracy is relatively stable across all settings.

Method	Llama 3.3 70B Instruct		Qwen-3-8B		GPT-OSS-120B	
	Agg.	Per-sample	Agg.	Per-sample	Agg.	Per-sample
Random	25.0	25.0	25.0	25.0	25.0	25.0
Ours (cross-family reference)	**100.0**	**100.0**	**100.0**	**99.7**	**100.0**	**99.8**

Table 11: **Cross-family, cross-organization reference models.** All 19 controlled students, scored with a reference from a different family and organization ($n = 7, 6, 6$ per teacher).

probing data), suggesting that the teacher-identification signal is not limited to the exact prompt distribution used during distillation.

B.3 Ablations on the number of distillation examples

We additionally ablate how many teacher-generated examples are needed for the reference-based signal to emerge. Starting from the same controlled setup, we vary the number of training examples used to distill the student while keeping the probing and candidate teacher pool fixed. Checkpointing a Qwen-2.5-1.5B student every 16 training examples and re-running reference-based MIA at each stage—using the base Qwen-2.5-1.5B as the reference model—we find the signal emerges almost immediately: after just 32 examples, the true teacher is correctly identified for all three teachers (GPT-OSS-120B, Qwen-3-8B, and Nvidia-Llama-3.3-70B-Instruct), as shown in Figure 7. The preceding checkpoint (16 examples) yields an identically zero reference-normalized loss, since under our cosine schedule with 5% warmup the first step is taken at a near-zero learning rate, leaving the student unchanged from the base reference. Identification then remains stable through the end of training, indicating that only a few dozen teacher-generated examples suffice for the true teacher to dominate.

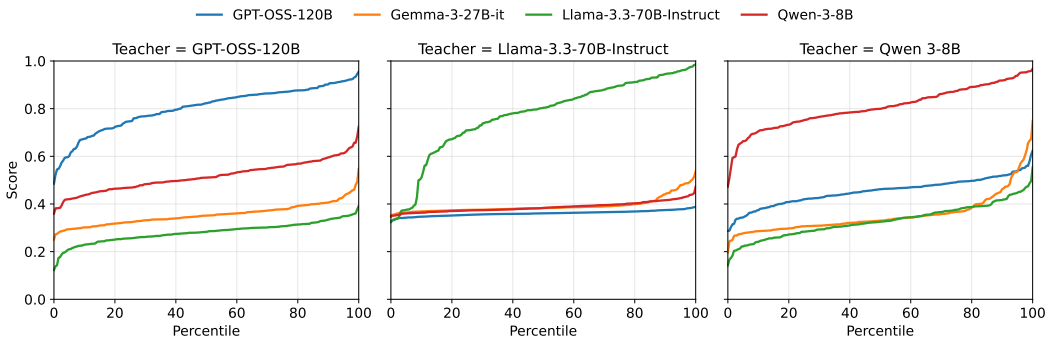


Figure 7: Reference-based MIA after only 32 training examples (checkpoint 2). A Qwen-2.5-1.5B student distilled from each teacher (GPT-OSS-120B, Llama-3.3-70B-Instruct, Qwen-3-8B) on OMI prompts, probed with s1 and referenced against the base Qwen-2.5-1.5B. In every panel the true teacher is the top-ranked candidate, showing that the teacher signal is already identifiable after only a few dozen examples.

B.4 Ablations for Reverse MIA methods

Method	Llama 3.3 70B Instruct		Qwen-3-8B		GPT-OSS-120B	
	Agg.	Per-sample	Agg.	Per-sample	Agg.	Per-sample
Random	25.0	25.0	25.0	25.0	25.0	25.0
Reverse logit-based methods						
Reverse raw likelihood	100.0	89.1	33.3	23.7	0.0	0.0
Reverse ZLIB-normalized	100.0	93.7	16.7	22.8	0.0	0.1
Reverse Min-k	100.0	97.2	0.0	10.3	0.0	0.2
Ours (Reference-based)	100.0	100.0	100.0	100.0	100.0	99.9

Table 12: **Teacher identification accuracy (%) for reverse logit-based baselines**. Accuracy is broken down by true teacher model, including both aggregated and per-sample metrics. Reverse likelihood variants perform well for Llama 3.3 70B Instruct but fail to reliably identify Qwen-3-8B and GPT-OSS-120B, while our reference-based method remains accurate across teachers.

B.5 Beyond the Canonical Setting

Our main evaluation assumes single-teacher, SFT-style final-stage distillation. We report preliminary results relaxing both assumptions.

Multiple teachers. We evaluate two models with publicly documented multi-teacher provenance, Olmo-3-7B-Think-SFT and Olmo-3-32B-Think-SFT, each distilled from traces produced by DeepSeek-R1, QwQ-32B, and Qwen-3-235B (Olmo et al., 2026). We use the corresponding pre-think checkpoint as the reference and score each true teacher against the nine other candidates in our pool, excluding the other two true teachers. As shown in Table 13, all three true teachers rank above every other candidate by mean alignment score, on both students and both probe sets. Per-sample, the signal is strongest on s1, where all six teacher–student pairs exceed 92.0. OMI probing is weaker but still recovers all three. The lowest cell, DeepSeek-R1 with the 32B student on OMI (44.0), is still the clear winner among ten candidates: the nearest competitors reach only 28.5 and 27.5, every other candidate is at zero, and chance is 10.0. The method therefore surfaces all contributing teachers at the top of the ranking rather than committing to a single one.

Beyond SFT. The Olmo-3 Think models include a DPO stage trained with delta learning, where chosen completions come from Qwen-3-32B and rejected ones from Qwen-3-0.6B. We run reference-based MIA on the DPO checkpoints using each model's own SFT checkpoint as the reference, isolating the DPO stage; the true teacher, Qwen-3-32B, is added to our pool, giving eleven candidates. Results are mixed (Table 14). For the 7B model, Qwen-3-32B attains the lowest mean reference-normalized loss on both probe sets and a high per-sample score on s1; for the 32B model the signal is much weaker and the true teacher is not recovered. Preference-tuning signal from a teacher is therefore detectable, but less reliably than under SFT—expected, since DPO shapes likelihood only

True teacher	Olmo-3-7B-Think-SFT		Olmo-3-32B-Think-SFT	
	OMI	s1	OMI	s1
Random	10.0	10.0	10.0	10.0
QwQ-32B	97.0	100.0	100.0	99.5
Qwen-3-235B	93.5	100.0	88.5	99.5
DeepSeek-R1	79.5	92.0	44.0	99.5

Table 13: **Multi-teacher identification (per-sample %).** References are Olmo-3-1025-7B and Olmo-3-1125-32B respectively. Each true teacher is scored against nine other candidates, excluding the other two true teachers.

through a relative preference gradient rather than by maximizing the likelihood of teacher text directly. This is a preliminary result on two models and we leave a fuller study to future work.

Target	OMI	s1
Olmo-3-7B-Think-DPO	57.5	98.5
Olmo-3-32B-Think-DPO	0.0	37.0

Table 14: **Identification under a DPO stage (per-sample %).** True teacher is Qwen-3-32B among eleven candidates; reference is each model's own SFT checkpoint.

C Statistical Significance Details

This section expands on the statistical tests summarized in §3.3.

Per-probe top-1 consistency (identification). With the true teacher $t^\star$ known in advance, we ask the most direct question: on how many *individual* probes is $t^\star$ the top-scoring candidate? We count a win on probe x_i when $t^\star$ outscores every other candidate,

$$X_i = \mathbb{1}\left[\arg\max_k \mathbf{f}(T_k(x_i), S) = t^\star\right],$$

and report the win rate $\hat{\rho} = \frac{1}{N}\sum_{i=1}^{N} X_i$; a *majority* means $\hat{\rho} > 0.5$. To rule out chance, note that if ranking were random $t^\star$ would top a probe with probability $1/K$, making the N wins Binomial; we test $H_0 : \rho \leq 1/K$ against $H_1 : \rho > 1/K$ with a one-sided exact binomial test. Because the test is exact, it needs no normality check or Wilcoxon fallback, and because $t^\star$ is fixed in advance (not picked as the best of K), it needs no Bonferroni correction—which is why this is an identification test, complementing the detection test below.

Margin significance (detection). The margin δ is a point estimate; to decide whether an observed δ reflects distillation rather than sampling noise, we attach an inferential test. Let t_1 and t_2 denote the top- and second-ranked candidates by mean score, and define the paired per-probe difference

$$d_i = \mathbf{f}(T_{t_1}(x_i), S) - \mathbf{f}(T_{t_2}(x_i), S).$$

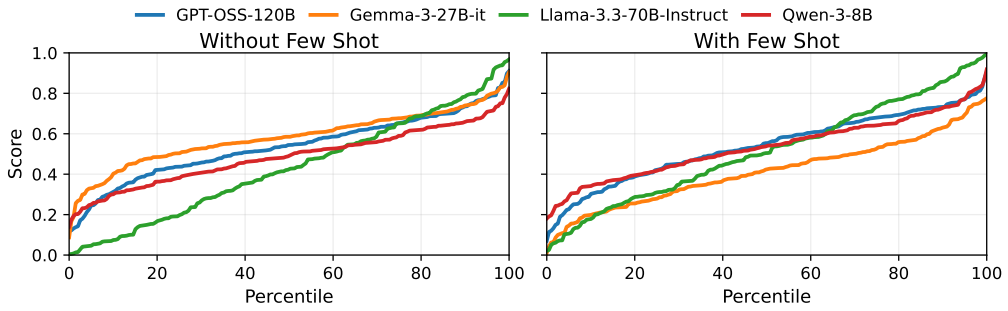
Pairing within a probe cancels per-prompt difficulty shared across candidates, and the sample mean of d recovers δ. We test against the calibrated threshold τ^* rather than against zero: at $N \approx 200$, even a negligible margin can be significant against zero, including for non-distilled models. Concretely, for $H_0 : \delta \leq \tau^*$ against $H_1 : \delta > \tau^*$, we use a one-sided t-test with statistic $t = (\delta - \tau^*)/(s_d/\sqrt{N})$ on $N-1$ degrees of freedom, and report a 95% confidence interval for δ from 10,000 probe-level bootstrap resamples as an effect-size diagnostic where shown (Table 16). Since d is occasionally non-normal (Shapiro–Wilk), we additionally compute a one-sided Wilcoxon signed-rank test on the shifted differences $d_i - \tau^*$—a distribution-free check of the same hypothesis—and base the decision on this test whenever the normality assumption is rejected (Shapiro $p < 0.05$). The decision uses the margin over the runner-up t_2: since t_2 is the closest competitor, clearing τ^* against it would generally imply clearing τ^* against every weaker candidate T_k ($k \neq t_1$).

Because t_1 is selected as the best of K candidates, we Bonferroni-correct the selection by reporting $p^* = \min(1,\, p \cdot (K-1))$, where p is the p-value of the governing test (t or Wilcoxon). We call a target distilled from teacher T only if the corrected test rejects H_0 ($p^* < 0.05$) on *both* prompt sets and both name the same teacher; requiring cross-prompt agreement makes the call teacher-specific rather than family-generic.

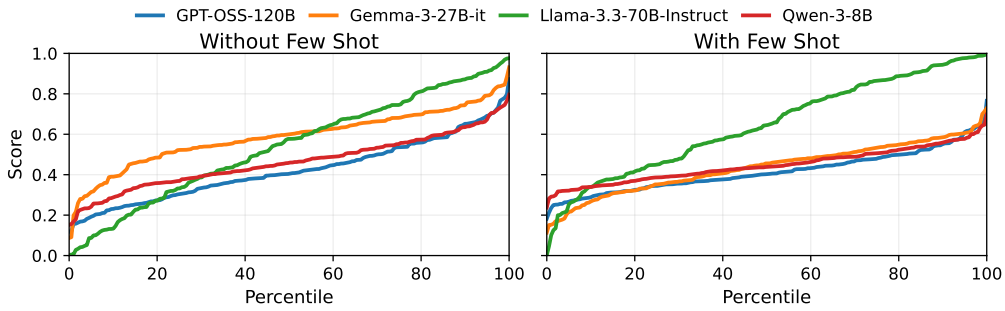
D DETECTING DISTILLATION WITH A HIGHLY CUSTOMIZED PROMPT TEMPLATE

We provide additional visualization of the few-shot prompting strategy described in the main paper (Table 3). While the table reports aggregate detection accuracy, the plots below illustrate how few-shot prompting affects the distribution of alignment scores across candidate teachers.

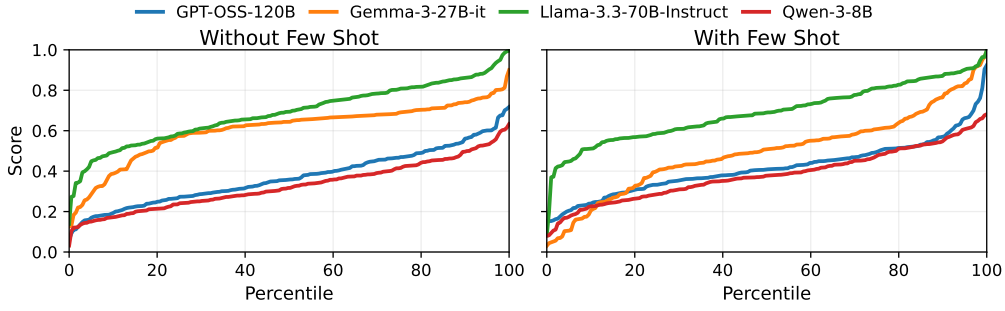
Each subplot corresponds to a different distilled student model, and shows how candidate teachers behave when prompted with a small number of student-generated examples. We observe that few-shot prompting generally sharpens the separation between the true teacher and alternative candidates, improving identification performance. However, for the Gemma-3-4B-pt student, the separation remains weaker, consistent with the failure case observed in Table 3.



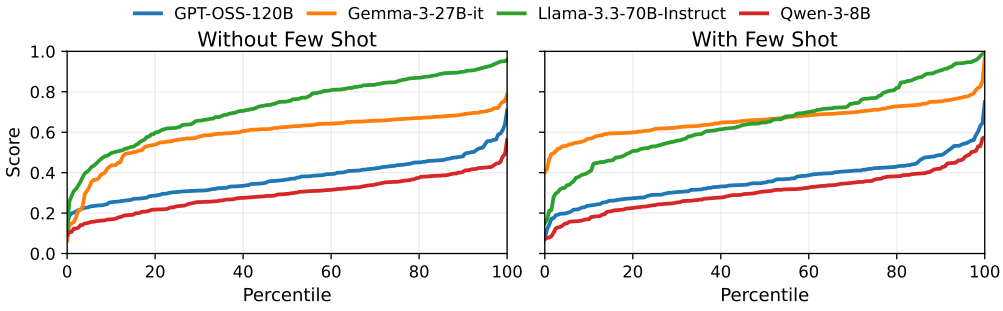
(a) Qwen-2.5-1.5B



(b) Qwen-2.5-3B



(c) Llama-3.2-3B-Instruct



(d) Gemma-3-4B-pt

Figure 8: Results of prompting each candidate teacher model with few-shot examples drawn from the outputs of a distilled student model. Each subplot corresponds to a different student and evaluates how well candidate teachers can match the student's behavior under in-context learning.

E Model in the Wild Plots

We evaluate the DeepSeek-R1 distill models, `s1.1-32B`, and XCoder using reference-based membership inference (reference-based MIA). For each distilled model, we use as the reference model the corresponding base model from which the distill model was derived prior to distillation. We run the method on both OMI and s1 probe data, allowing us to test whether teacher-identification signals persist across related math-domain prompt distributions.

Overall, our method successfully identifies the true teacher across the evaluated candidates, showing that the reference-normalized likelihood signal remains informative even for real-world distilled models whose exact training data and distillation pipelines are unknown. This is encouraging because these models are not produced under our controlled setup: the prompts, filtering procedures, decoding parameters, and post-training details are all unobserved. Nevertheless, the true teacher generally receives the strongest alignment signal.

We further evaluate XCoder, a code-specialized model trained on code data, while still probing it using math data from s1 and OMI. The method continues to show a teacher-specific signal in this out-of-domain setting, suggesting that the attribution signal is not limited to models trained directly on math data.

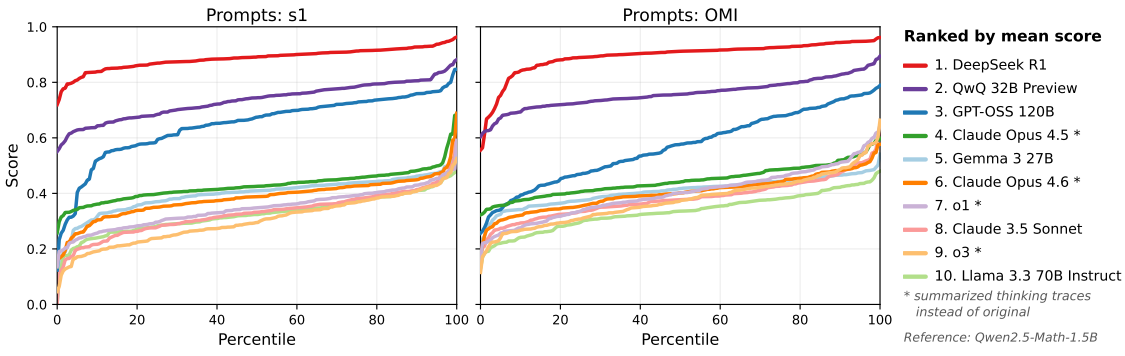


Figure 9: Reference-based MIA results for **DeepSeek-R1-Distill-Qwen-1.5B**, using **Qwen2.5-Math-1.5B** as the reference model.

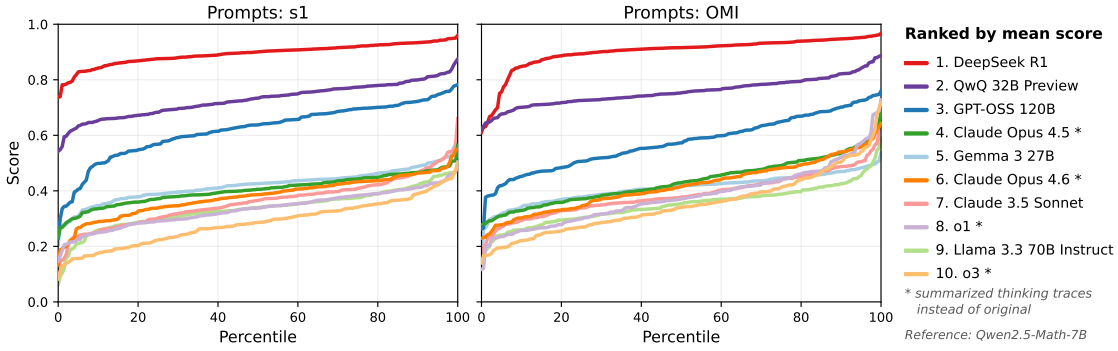


Figure 10: Reference-based MIA results for **DeepSeek-R1-Distill-Qwen-7B**, using **Qwen2.5-Math-7B** as the reference model.

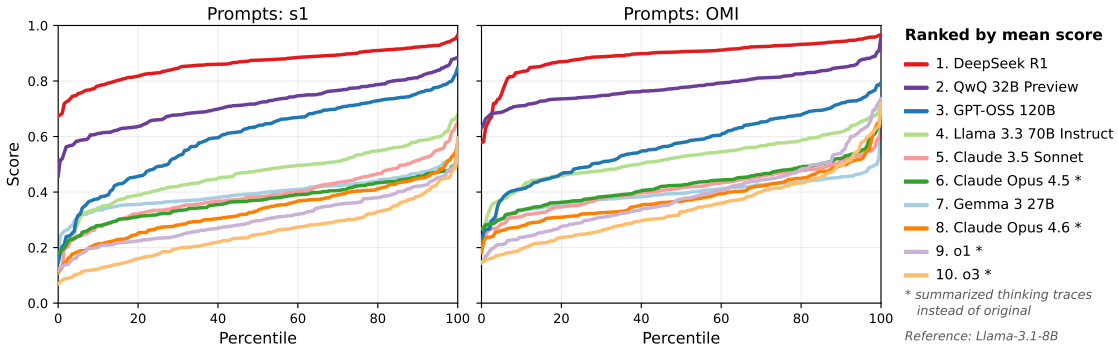


Figure 11: Reference-based MIA results for **DeepSeek-R1-Distill-Llama-8B**, using **Llama-3.1-8B** as the reference model.

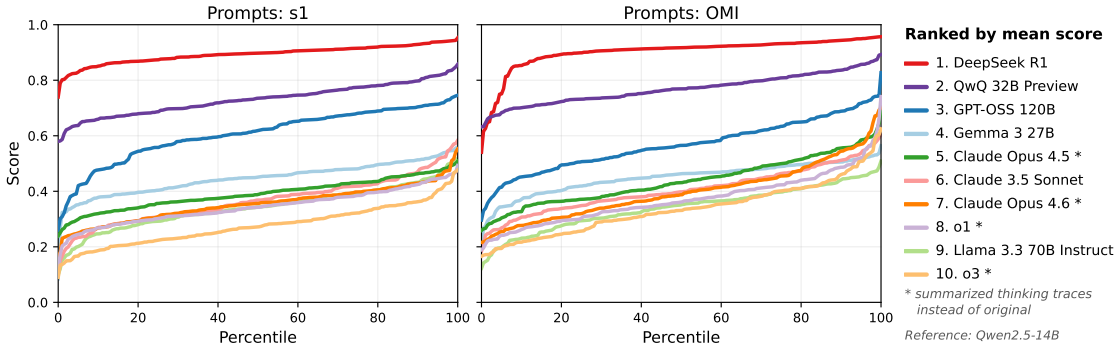


Figure 12: Reference-based MIA results for **DeepSeek-R1-Distill-Qwen-14B**, using **Qwen2.5-14B** as the reference model.

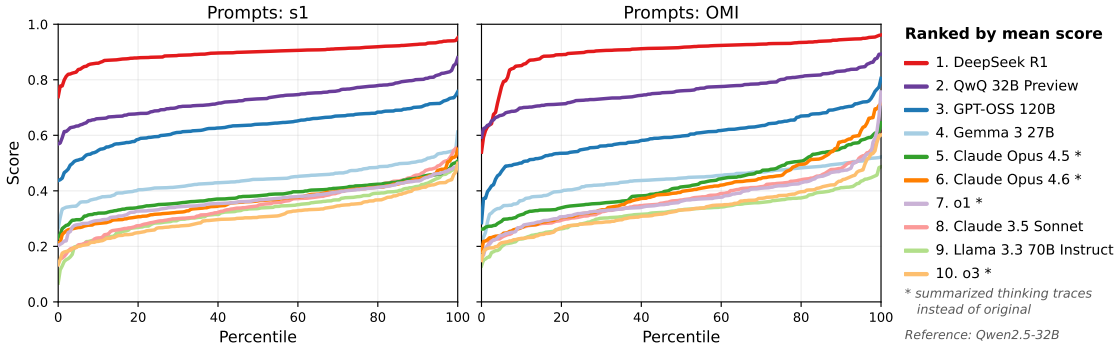


Figure 13: Reference-based MIA results for **DeepSeek-R1-Distill-Qwen-32B**, using **Qwen2.5-32B** as the reference model.

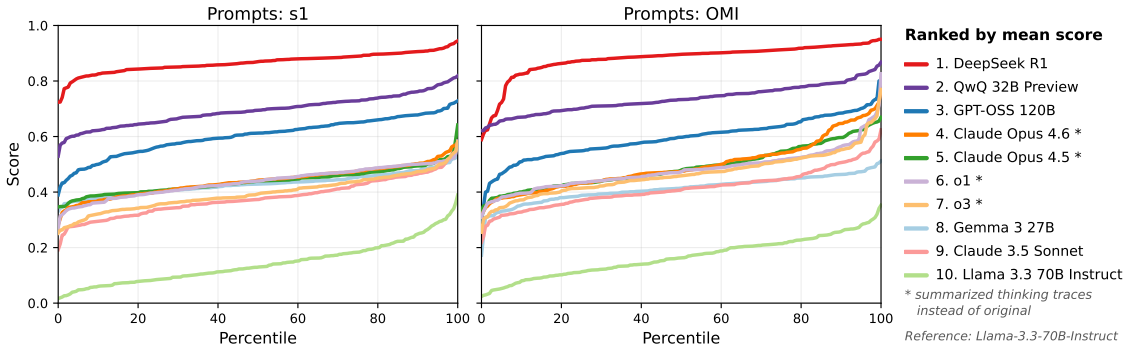


Figure 14: Reference-based MIA results for **DeepSeek-R1-Distill-Llama-70B**, using **Llama-3.3-70B-Instruct** as the reference model.

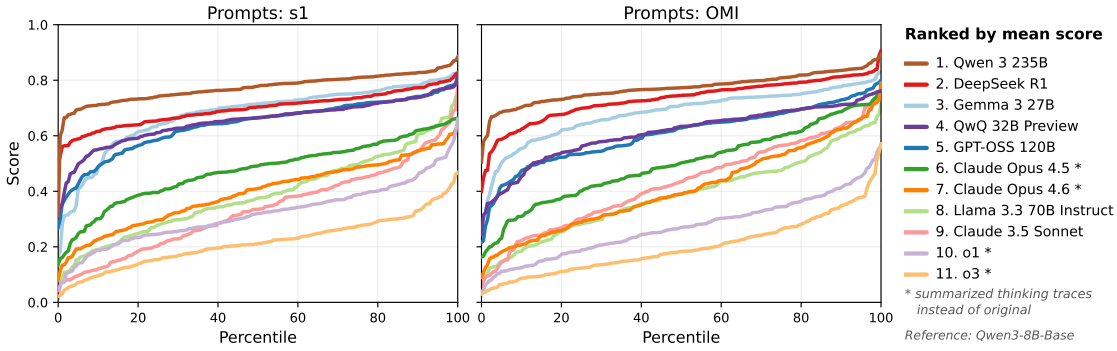


Figure 15: Reference-based MIA results on **XCoder**. Although XCoder was trained on code data, we probe it using math datasets (s1 and OMI) to test whether teacher-specific likelihood signals remain detectable out of domain.

F Additional Distillation Detection Methods

This section provides additional results for the distillation detection experiments in §5.

Leave-one-student-out on real-world models. Table 15 reports the real-world cross-validation summarized in §5.2. With τ refit on the other six R1-distilled models in each fold, the detection test calls six of seven held-out students distilled from DeepSeek-R1 (both prompts, p^* between 1.2×10^{-26} and 8.0×10^{-3}); when DeepSeek-R1 is removed, no alternative ever clears the threshold

($p^* \geq 0.078$), giving zero false detections. The single missed positive is s1.1-32B: DeepSeek-R1 ranks first on both prompts, but its OMI margin (0.105) falls below the fold-fit threshold ($\tau = 0.176$), so OMI is non-significant even though s1 is overwhelming ($p^* = 8.5\times10^{-34}$), and the both-prompts rule withholds the call.

Held-out condition	Accuracy	Detected	Detection p^*
DeepSeek-R1 included	13/14	6/7	1.2×10^{-26}–8.0×10^{-3}
DeepSeek-R1 removed	13/14	0/7	≥ 0.078 (n.s.)
Overall	**26/28** (92.9%)	—	—

Table 15: **Leave-one-student-out (real-world).** Margin accuracy and significant detections (both prompts, same teacher); with DeepSeek-R1 removed, no alternative is significant.

G Additional o1 Detection Details

Background: ASCII and Unicode. ASCII is a 7-bit encoding covering 128 characters; Unicode assigns codepoints to over 140,000 characters spanning nearly all writing systems and mathematical notation. ASCII is a strict subset of Unicode, so any Unicode string can be losslessly rendered as ASCII by escaping each non-ASCII codepoint as its literal `\uXXXX` sequence.

Methodology. We generate each `o1` output once and serialize it two ways: a Unicode form (`ensure_ascii=False`) preserving raw UTF-8 glyphs, and an ASCII form (`ensure_ascii=True`) in which each non-ASCII codepoint becomes its six-character escape (e.g. "•" → `\u2022`). For each target, we run reference-based MIA twice with all other hyperparameters fixed and define $\Delta_{\text{ASCII}} = \overline{\mathcal{L}}_{\text{ASCII}} - \overline{\mathcal{L}}_{\text{Unicode}}$ on the reference-normalized loss. Under the null that the detector is not exploiting glyph-level features, $\Delta_{\text{ASCII}} \approx 0$; a positive gap indicates that ASCII normalization *raises* the target's loss relative to the Unicode baseline—i.e., that the encoding contributes to the membership signal.

Controlled positives and probing data. To establish a ground-truth positive condition, we supervised-fine-tune four students—`Qwen2.5-1.5B`, `Qwen2.5-3B`, `Llama-3.2-3B-Instruct`, and `gemma-3-4b-pt`—with `o1` as the teacher on 1,000 questions drawn from the s1 dataset. Because the OpenAI API exposes only a *summarized* reasoning trace, each training target is the concatenation of the summarized trace and the final answer. The MIA reference for each student is its own pre-SFT base checkpoint, isolating any signal acquired during distillation. As negative controls, we use real U.S.-organization models that are not expected to be distilled from `o1`/`o3`. We use 100 held-out OMI questions as probing data, providing a realistic setting where the detector does not have access to the original training inputs.

Target	Δ_{ASCII}	95% CI	p	Sig.
`gemma-3-4b-pt` (SFT)	+0.0732	[+0.064, +0.083]	$<10^{-15}$	✓
`Qwen2.5-1.5B` (SFT)	+0.0594	[+0.050, +0.069]	$<10^{-15}$	✓
`Qwen2.5-3B` (SFT)	+0.0573	[+0.048, +0.067]	$<10^{-15}$	✓
`Llama-3.2-3B-Instruct` (SFT)	+0.0647	[+0.055, +0.075]	$<10^{-15}$	✓
`gemma-3-12b-it`	+0.0373	[+0.028, +0.047]	5.0×10^{-11}	✓
`gemma-3-27b-it`	+0.0072	[−0.001, +0.015]	2.7×10^{-2}	✓
`gemma-3-27b-pt`	+0.0070	[+0.003, +0.011]	5.1×10^{-4}	✓
`gemma-2-9b`	+0.0016	[−0.005, +0.008]	0.28	✗
`Llama-3.3-70B`	+0.0007	[−0.005, +0.006]	0.66	✗
`Llama-3.1-8B`	−0.0024	[−0.007, +0.002]	0.63	✗
`Llama-3.1-70B`	−0.0050	[−0.008, −0.002]	1.00	✗
`Llama-3.1-8B-Inst.`	−0.0078	[−0.017, +0.001]	0.93	✗
`gemma-2-9b-it`	−0.0733	[−0.105, −0.043]	1.00	✗

Table 16: **ASCII vs. Unicode serialization with significance.** Targets ranked by Δ_{ASCII}; positive values indicate higher loss under ASCII normalization. We report Δ_{ASCII} with 95% bootstrap CIs; ✓ denotes statistical significance. All four `o1`-distilled students are significant, as are three Gemma-3 controls—but with much smaller gaps. Red: `o1`-distilled positives; blue: controls.

Additional control and wild-model results. We additionally evaluate a broader set of within-family controls, controlled o1-distilled positives, and wild models with large observed ASCII gaps. Table 17 reports the reference model used for each target and the resulting Δ_{ASCII}.

Target	Reference	Δ_{ASCII}
Within-family controls (not distilled from o1/o3)		
Llama-3.1-8B-Instruct	Llama-3-8B-Instruct	−0.0078
Llama-3.3-70B-Instruct	Llama-3.1-70B-Instruct	+0.0007
Llama-3.1-8B	Llama-3-8B	−0.0024
Llama-3.1-70B	Llama-3-70B	−0.0050
gemma-2-9b-it	gemma-1.1-7b-it	−0.0733
gemma-3-12b-it	gemma-2-9b-it	+0.0373
gemma-3-27b-it	gemma-2-27b-it	+0.0072
gemma-2-9b	gemma-7b	+0.0016
gemma-3-27b-pt	gemma-2-27b	+0.0070
o1-distilled positives		
gemma-3-4b-pt (SFT)	gemma-3-4b-pt	+0.0732
Qwen2.5-1.5B (SFT)	Qwen2.5-1.5B	+0.0594
Qwen2.5-3B (SFT)	Qwen2.5-3B	+0.0573
Llama-3.2-3B-Instruct (SFT)	Llama-3.2-3B-Instruct	+0.0647
Models in the wild		
DeepSeek-R1	DeepSeek-MoE-16B-Base	+0.937
GPT-OSS-120B	GPT-OSS-20B	+0.3207
GPT-OSS-120B	GPT-2 XL	+0.8310
GPT-OSS-20B	GPT-2 XL	+0.5103

Table 17: **ASCII gap Δ_{ASCII} on controlled and wild models.** For each target, we report the reference model used and $\Delta_{\text{ASCII}} = \overline{\mathcal{L}}_{\text{ASCII}} - \overline{\mathcal{L}}_{\text{Unicode}}$ on the reference-normalized loss, computed on 100 OMI probes (see §6). Within-family controls cluster near zero, controlled o1-distilled positives show consistently positive gaps, and DeepSeek-R1 and both GPT-OSS models exhibit large positive gaps.

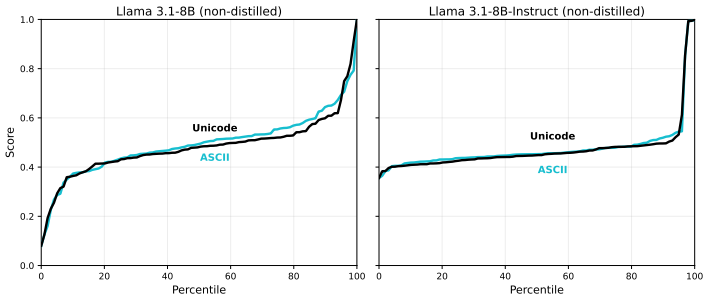


Figure 16: **ASCII vs. Unicode CDFs for non-o1-distilled Llama controls.** For Llama models not expected to be distilled from o1, the ASCII and Unicode reference-normalized loss distributions nearly overlap, indicating little to no encoding-specific separation.

The within-family controls—Llama and Gemma checkpoints not expected to be distilled from o1 or o3—produce ASCII gaps concentrated near zero, with most pairs falling within ± 0.01 and the largest deviation at -0.0733 for gemma-2-9b-it. This is also visible in Figure 16: for non-o1-distilled Llama controls, the ASCII and Unicode CDFs lie almost on top of each other, showing no substantial difference in reference-normalized loss under the two serializations. This is consistent with the hypothesis that no o1-specific encoding signal is present in models not exposed to o1 during training, and matches the pattern reported in the main text.

In contrast, the controlled o1-distilled positives all exhibit positive gaps ($+0.0573$ to $+0.0732$), while the real-world models exhibit much larger positive gaps. DeepSeek-R1, with DeepSeek-MoE-16B-Base as reference, produces $\Delta_{\text{ASCII}} = +0.937$. Using GPT-2 XL

as the reference, GPT-OSS-20B produces $\Delta_{\text{ASCII}} = +0.5103$ and GPT-OSS-120B produces $\Delta_{\text{ASCII}} = +0.8310$; using GPT-OSS-20B as a within-family reference for GPT-OSS-120B still yields $\Delta_{\text{ASCII}} = +0.3207$. Thus, the real-world gaps are not only positive but also substantially larger than those of our controlled o1-distilled students, often by nearly an order of magnitude. This mirrors the per-probe CDF separation for GPT-OSS and DeepSeek-R1 reported in §7.

Surface-Level Stylistic Markers as Complementary Signals We additionally observe that certain teacher models exhibit highly consistent surface-level generation habits. Specifically, Llama-3.3-70B-Instruct responses follow a structured Markdown format using headings of the form `## Step` N, which Toshniwal et al. (2024) document as the characteristic "Llama CoT" format. We similarly observe that claude-3.5-sonnet responses frequently contain the phrase "Let's solve this step by step," and QwQ-32B-Preview responses frequently open with "I've got this problem." Notably, Toshniwal et al. (2024) show that a highly customized prompt template can suppress these stylistic artifacts during distillation, which may limit their utility as attribution signals. We leave a more systematic study—including robustness to prompt variation and integration with likelihood-based scores—to future work.

Results. Table 18 reports occurrence rates across the first 200 OMI responses per teacher. The Llama marker is especially strong, appearing in every response with an average of 7.26 occurrences per response, while the Claude and QwQ markers occur at nontrivial but lower rates.

Table 18: Surface-level stylistic marker occurrence rates across the first 200 OMI responses per teacher model.

Teacher Model	Marker	Responses Hit	Hit Rate (%)	Avg. per Response
Claude 3.5 Sonnet	"Let's solve this step by step"	66 / 200	33.0	0.33
QwQ-32B-Preview	"I've got this problem"	35 / 200	17.5	0.18
Llama-3.3-70B-Instruct	`## Step` N heading	200 / 200	100.0	7.26

H Additional Open Question Plots

This section provides additional reference-based MIA plots and statistical analysis for the open-question analyses in §7. These settings involve target models whose training data, prompts, instructions, filtering strategies, and distillation procedures are not publicly known, so the results should be interpreted as diagnostic rather than conclusive evidence of distillation.

DeepSeek R1. We evaluate DeepSeek-R1, using DeepSeek-MoE-16B as the reference model. Results are reported in Figure 17. Our reference-based method ranks o1 and QwQ-32B-Preview near the top, raising the possibility that DeepSeek-R1 was influenced by or distilled from o1 or QwQ-32B-Preview.

In addition, the o1 diagnostic in Figure 17 (right) shows a clear Unicode–ASCII gap. We test this gap one-sided against zero (95% bootstrap CI over 100 probes; Wilcoxon, as the differences are non-normal): it is $\delta_{\text{Uni}-\text{ASCII}} = +0.937$ ($[+0.746, +1.141]$, $p = 8.5 \times 10^{-16}$; Table 19), confirming the separation is resolvable rather than a sampling artifact. This suggests possible o1-style signal in DeepSeek-R1.

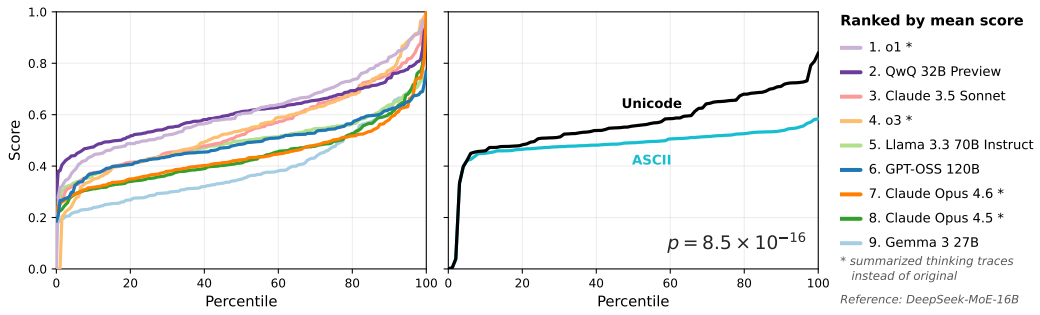


Figure 17: Our reference-based method and ASCII–Unicode diagnostic applied to `DeepSeek-R1` using `DeepSeek-MoE-16B` as a reference.

GPT-OSS-20B & GPT-OSS-120B. Unlike `QwQ` and `DeepSeek-R1`, `GPT-OSS` does not have a suitable same-family checkpoint that can serve as a natural reference model. We therefore use `GPT-2 XL`, an older open-source LM released by OpenAI, while noting that it is unlikely to be a reliable reference given the substantial temporal and architectural gap. We include this evaluation primarily to illustrate how our method behaves in the absence of an appropriate reference model; results with alternative reference models are reported below.

As shown in Figure 18, `DeepSeek-R1` and `QwQ-32B-Preview` are ranked highest. However, these results are difficult to interpret, as our earlier analyses suggest that `DeepSeek-R1` and `QwQ-32B-Preview` may themselves be influenced by `o1`, which is also a plausible teacher for `GPT-OSS`.[7] An additional `o1` diagnostic (Figure 18, right) shows that the Unicode CDF lies consistently above the ASCII CDF, and this gap is statistically resolvable across all reference models for `GPT-OSS` (Table 19). Together, these observations raise the possibility that `GPT-OSS` is influenced by the same `o1`/`DeepSeek-R1`/`QwQ` lineage. However, because `GPT-OSS` lacks a suitable reference model, this conclusion remains highly uncertain. We leave a more systematic analysis to future work that relaxes the assumption that an appropriate reference model is available.

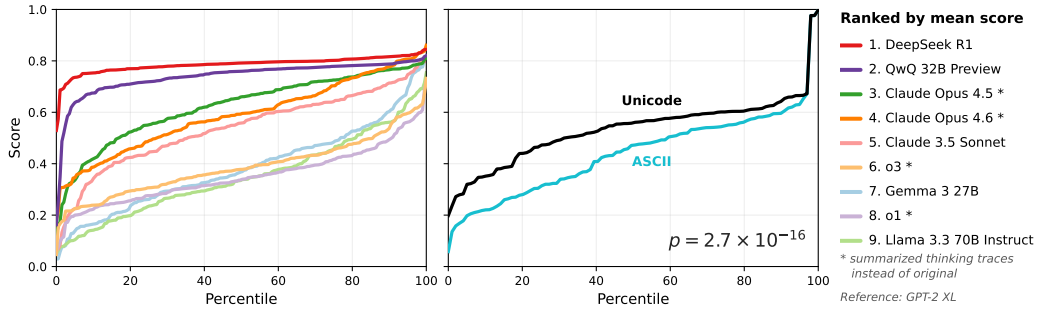


Figure 18: Our reference-based method and ASCII–Unicode diagnostic applied to `GPT-OSS-120B` using `GPT-2-XL` as the reference.

[7]Note that `o1` is included in our candidate teacher models but ranks low. However, this does not rule out the possibility that `GPT-OSS` was influenced by `o1`, because `o1` does not expose its original thinking traces, and this systematically underestimates the likelihood assigned to `o1`.

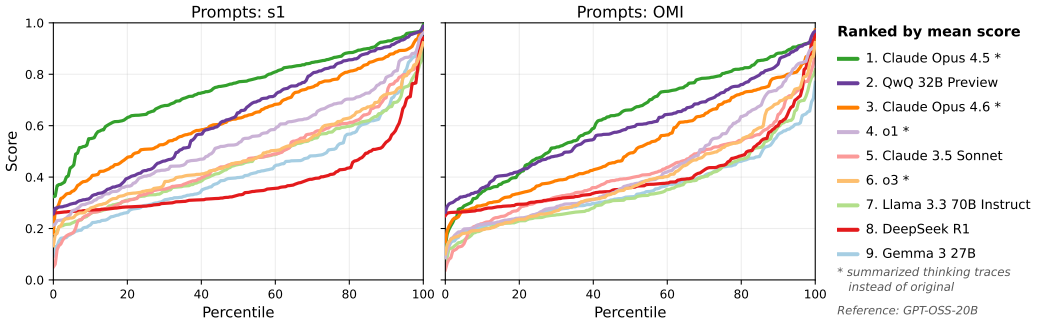


Figure 19: We apply reference-based MIA to `GPT-OSS-120B`, using `GPT-OSS-20B` as the reference model.

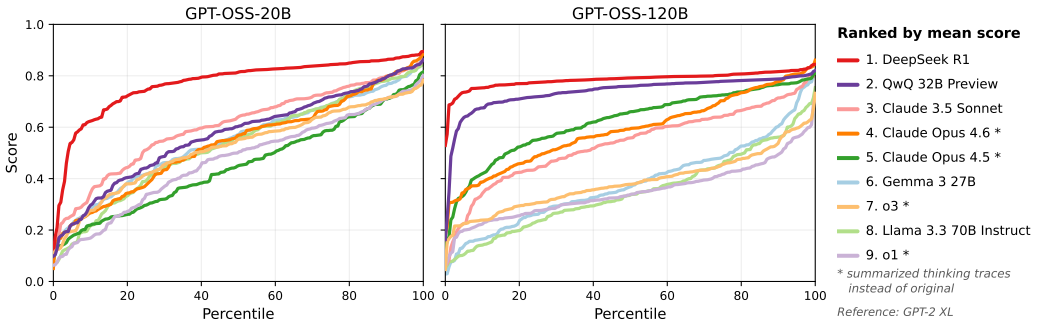


Figure 20: We apply reference-based MIA to `GPT-OSS-120B` and `GPT-OSS-20B`, using `GPT-2 XL` as the reference model and OMI as probing data.

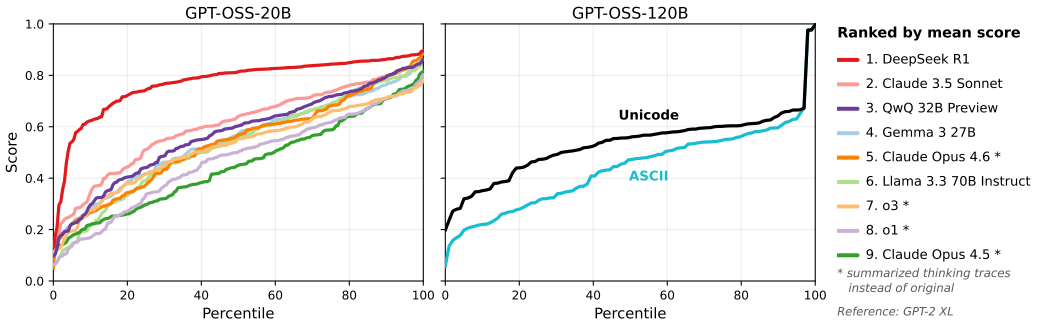


Figure 21: **Results for the GPT-OSS family.** We evaluate these models with both reference-based MIA (left) and our ASCII–Unicode diagnostic (right).

Target	Reference	Probe	$\delta_{\text{Uni}-\text{ASCII}}$ [95% CI]	Test	p	Sig.
DeepSeek-R1	DeepSeek-MoE-16B-Base	OMI	$+0.937$ $[+0.746, +1.141]$	Wilcoxon	8.5×10^{-16}	✓
GPT-OSS-120B	GPT-OSS-20B	OMI	$+0.32$ $[+0.21, +0.44]$	Wilcoxon	3.5×10^{-8}	✓
GPT-OSS-120B	GPT-2 XL	OMI	$+0.83$ $[+0.69, +0.98]$	Wilcoxon	2.7×10^{-16}	✓
GPT-OSS-20B	GPT-2 XL	OMI	$+0.51$ $[+0.39, +0.64]$	Wilcoxon	1.3×10^{-13}	✓

Table 19: **Statistical significance of the o1 Unicode–ASCII diagnostic.** The within-model gap $\delta_{\text{Uni}-\text{ASCII}}$ tested one-sided against 0 (Wilcoxon where the differences are non-normal, t-test otherwise); ✓ marks $p < 0.05$. A significant gap means the separation is resolvable rather than sampling noise—suggestive of o1-style signal, not a confirmed distillation call.