

When Large Language Models Know the Table: A Framework for Assessing Data Contamination in Tabular Datasets

Matteo Silvestri¹, Fabiano Veglianti¹

Flavio Giorgi^{1,2}, Fabrizio Silvestri¹, Gabriele Tolomei^{1,2}

¹Sapienza University of Rome, Italy; ²TellmewAI s.r.l.

Correspondence to {m.silvestri, gabriele.tolomei}@uniroma1.it

Abstract

Large language models (LLMs) are increasingly exposed to *data contamination*, i.e., performance gains driven by prior exposure of test datasets rather than generalization. However, in the context of tabular data, this problem is largely unexplored. Existing approaches primarily rely on memorization tests, which are too coarse to detect contamination. In contrast, we propose a framework for assessing contamination in tabular datasets by generating controlled queries and performing comparative evaluation. Given a dataset, we craft multiple-choice aligned queries that preserve task structure while allowing systematic transformations of the underlying data. These transformations are designed to selectively disrupt dataset information while preserving partial knowledge, enabling us to isolate performance attributable to contamination. We complement this setup with non-neural baselines that provide reference performance, and we introduce a statistical testing procedure to formally detect significant deviations indicative of contamination. Empirical results on eight widely used tabular datasets reveal clear evidence of contamination in four cases. These findings suggest that performance on downstream tasks involving such datasets may be substantially inflated, raising concerns about the reliability of current evaluation practices.

1 Introduction

Large language models (LLMs) are now used across many tasks and domains (Bommasani et al., 2022; Naveed et al., 2024), making reliable evaluation increasingly important. Benchmark results often guide deployment decisions and support claims of progress (Rogers & Luccioni, 2024). To be meaningful, these results must reflect a model’s ability to generalize to unseen data rather than familiarity with the evaluation set itself (Zhou et al., 2023).

A major threat to this validity is *data contamination* (Cheng et al., 2025), which we define here as overlap between training and test datasets. In this case, a model may perform well not because it learned the underlying task, but because it was previously exposed to the same dataset or highly similar instances during pretraining or post-training (Balloccu et al., 2024; Sainz et al., 2023). This can inflate benchmark scores and distort comparisons across models. Detecting contamination is therefore essential for trustworthy evaluation. Contamination is related to, but distinct from, *memorization*. Memorization asks whether a model stores and can reproduce training data, often in the context of privacy or data extraction (Carlini et al., 2019). Contamination, instead, is an evaluation problem: whether prior exposure to benchmark data renders performance an invalid measure of generalization (Xu et al., 2024). A model need not reproduce test data verbatim for contamination to matter. What matters is whether it has non-trivial prior knowledge of the dataset that can be exploited to achieve unexpectedly strong performance.

This issue is especially relevant for *tabular data*. LLMs are increasingly applied to structured inputs (Fang et al., 2024), in tasks such as table understanding (Wang et al., 2024), data generation (Borisov et al., 2022), and wrangling (Narayan et al., 2022), classification

(Gardner et al., 2024), and reasoning over structured attributes (Jiang et al., 2023b). At the same time, many tabular benchmarks are constructed from publicly accessible datasets¹ and are therefore plausible candidates for inclusion in model training corpora. Despite this risk, contamination in tabular settings remains poorly understood.

Existing approaches do not adequately address this problem. Prior contamination studies and detection frameworks have focused on text benchmarks (Golchin & Surdeanu, 2024; 2025), where the main signal is overlap between textual test examples and training data. By contrast, work on tabular data has mainly focused on verbatim memorization of individual rows (Bordt et al., 2024), rather than dataset-level contamination. This leaves a key methodological gap: there is still no principled framework to assess whether an LLM has prior knowledge of a public tabular dataset, nor a clear criterion for deciding when strong performance reflects contamination rather than ordinary task competence.

To address the need for reliable evaluation, we introduce a framework for assessing LLM contamination in tabular datasets using a dual-task multiple-choice benchmark. Unlike prior tabular approaches, which offer valuable memorization tests but no unified decision procedure, our framework grounds contamination assessment in a principled statistical testing procedure. Our method tests whether a model can recover masked attributes (completion) or verify an instance’s presence (existence). By evaluating performance across dataset variants and comparing it against non-neural baselines, we can finely separate general or statistical knowledge from dataset exposure. While prior approaches proved too coarse, our framework successfully detected contamination in four of eight tested datasets, highlighting a concrete risk in LLM evaluation. To summarize, our main contributions are:

- i) We introduce a framework for assessing contamination in tabular datasets used by LLMs. Given a dataset, we generate aligned multiple-choice queries across dataset variants to test a model’s prior knowledge. Our code is publicly available at https://github.com/hercolelab/tabular_contamination.
- ii) We propose non-neural baselines and a statistical testing procedure that provides a principled criterion for detecting contamination.
- iii) We empirically show that prior memorization-based frameworks are too coarse to detect contamination across models and datasets, whereas our framework yields a finer-grained, more informative contamination profile.

The remainder of the paper is structured as follows. Section 2 reviews related work. Section 3 presents our framework, including the problem overview, dataset variants, evaluation probes, metrics, baselines, and result interpretation. Section 4 describes the experimental setup and reports results for both prior memorization-based approaches and our method. Section 5 discusses limitations and implications, and Section 6 concludes.

2 Related Work

LLMs can memorize training data (Carlini et al., 2019; 2022) and reveal stored content under suitable prompting conditions (Carlini et al., 2021; Nasr et al., 2023). This shows that training information can persist in model parameters, raising concerns that benchmark performance may be inflated by prior exposure to evaluation data (Sainz et al., 2023).

This concern motivates the study of data contamination in benchmark evaluation (Xu et al., 2024). In NLP, contamination has been studied through methods such as n-gram overlap, token distribution analysis (Shi et al., 2024; Golchin & Surdeanu, 2024), and multiple-choice quizzes designed to test prior exposure (Golchin & Surdeanu, 2025). These works show that benchmark leakage can substantially bias reported performance. Similar concerns arise in other domains: Di Palma et al. (2025) show that LLMs memorize MovieLens (Harper & Konstan, 2015), a widely used recommendation benchmark, and that this prior knowledge can inflate evaluation results.

As LLMs are increasingly applied to tabular tasks like classification (Gardner et al., 2024; Hegselmann et al., 2023; Sun et al., 2024) and feature importance estimation (Kroeger

¹<https://www.kaggle.com/datasets> is a well-known source of tabular datasets

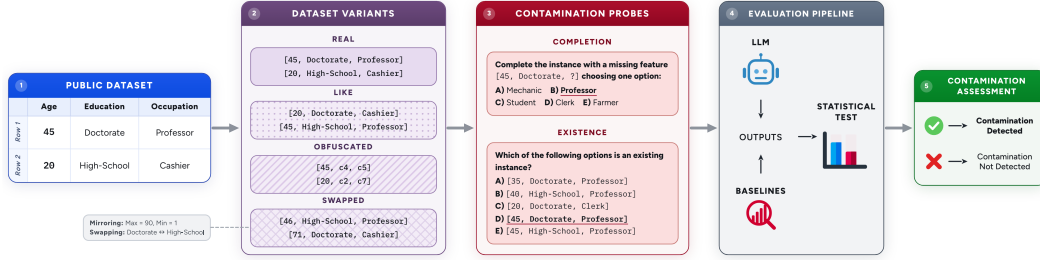


Figure 1: **Overview of the proposed framework for contamination assessment.** Given a public tabular dataset, we derive controlled dataset variants, construct contamination probes for completion and existence queries, evaluate LLMs against non-neural baselines, and use a statistical test to decide whether the observed behavior is indicative of contamination.

et al., 2023), contamination in structured data has become a critical yet underexplored concern. Existing detection methods rely heavily on broad dataset knowledge or verbatim memorization (Bordt et al., 2024; Ronval et al., 2025); however, these signals are often too coarse to reliably indicate inflated benchmark performance (see Section 4.2.1) or to help establish a standardized contamination registry, as claimed by Sainz et al. (2023). Bordt et al. (2024) rely on memorization tests that, as we show in Section 4.2.1, fail to detect contamination in smaller and open-source models, limiting their reach across the current model landscape. Ronval et al. (2025) instead propose a decision rule for tabular contamination, but it reduces exposure to a binary flag rather than a statistically grounded test, leaving no room to express degrees of evidence. To overcome these limitations, we evaluate prior knowledge using multiple-choice quizzes derived from original tabular instances and controlled variants, comparing model behavior against non-neural baselines within a statistical testing procedure to yield robust, graded evidence of contamination. Multiple-choice probing is itself an established paradigm in NLP contamination detection (Golchin & Surdeanu, 2025), but transferring it to tabular data is not a direct port. Tables are governed by column-level marginal distributions, feature correlations, and heterogeneous attribute types, so a probe that ignores this structure cannot tell dataset exposure apart from generic distributional knowledge about the domain. Our contribution is therefore twofold: probes designed around this structure, and a guessing-baseline construction that turns their outcome into a statistically grounded decision criterion for contamination.

3 Proposed Framework

In this section, we present our framework for assessing contamination in LLMs on tabular benchmarks (see Figure 1). Specifically, we focus on the problem overview, dataset variants, evaluation probes, metrics, baselines, and result interpretation.

3.1 Problem Overview

Because prior exposure to benchmarks threatens methodological integrity even without exact memorization, tabular datasets serve as an ideal testbed: their structured format limits verbatim recall while remaining semantically recognizable. To address this, our framework’s goal is to investigate row-level contamination by evaluating an LLM’s ability to recognize or reconstruct instance information under controlled transformations.

Let $D = \{r_1, \dots, r_N\}$ be a tabular dataset, whose rows share a fixed schema. We evaluate row-level contamination using five-way multiple-choice probes, specifically, *completion* tasks (predicting masked cells) and *existence* tasks (identifying an authentic row among distractors), to ensure cross-model comparability and establish explicit chance baselines. Ultimately, our framework assesses contamination by comparing this structured performance on the original dataset against transformed variants that isolate specific data characteristics.

3.2 Dataset Variants

For each dataset, we construct four variants: real, like, swapped, and obfuscated. These variants are generated from the same source table and are aligned so that the same underlying candidate rows are probed across conditions (more details in Appendix A.2).

Real. The real variant is the unmodified cleaned dataset. It preserves the original feature names, original values, marginal distributions, and joint row-level dependencies. This is the reference condition in which contamination, if present, should be most directly expressed.

Like. The like variant independently resamples each column from its empirical marginal distribution. It preserves univariate marginals but destroys row-wise dependence, so success here reflects column-level regularities rather than authentic row knowledge.

Obfuscated. The obfuscated variant removes surface-level semantic cues. All feature names are renamed sequentially, e.g., to $f_1, f_2, \dots$, the target column is renamed to target, and categorical values are replaced with anonymous codes. This variant preserves row structure and value distributions while suppressing interpretable semantics, helping distinguish contamination from world knowledge or familiarity with the dataset name.

Swapped. The swapped variant preserves schema and row structure but applies a within-column bijection so no value maps to itself. For categorical features, we map each element to another so that no value maps to itself; for numerical features with values in the interval $[N_{min}, N_{max}]$, we apply the map $\pi(x) := N_{max} - x - N_{min}$. This variant preserves row structure and column co-occurrence patterns.

3.3 Task Generation

Tasks involve dataset-dependent queries that are aligned across dataset variants. We define a *probe* as a structured multiple-choice query designed to test for dataset-specific knowledge. In the remainder of the paper, we use query and probe interchangeably to refer to these evaluation instances. This alignment ensures that differences in accuracy across variants are attributable to the transformation itself, rather than to changes in query difficulty. Details on probe construction and examples are provided in Appendix A.1.

Completion Task. In a completion probe, a real row is partially observed, and the selected columns are replaced by question marks; then the model is asked to choose the correct values for the masked cells among five options. The correct option contains the true values, while the four distractors are sampled from the empirical supports of the masked columns and filtered to differ from the truth, from one another, and from original instances. Specifically, let $M = \{c_{j_1}, \dots, c_{j_m}\}$ denote the masked columns, then the prompt displays the row with $c_{j_\ell} = ?$ for all ℓ and each option is formatted as a comma-separated assignment list, such as

$$c_{j_1} = v_{j_1}, c_{j_2} = v_{j_2}, \dots, c_{j_m} = v_{j_m}.$$

The position of the correct option in a prompt is randomly chosen. The random choice allows us to overcome potential position bias in the model (see Figure 2). This task assesses whether the model can recover missing attribute values from the observed portion of a row.

Existence Task. In an existence probe, the model is shown five complete rows and asked to identify which one genuinely occurs in the dataset. Similarly to the completion task, the correct option is the true row, while the four distractors are generated by replacing the $M = \{c_{j_1}, \dots, c_{j_m}\}$ target column values with alternative values sampled from the empirical support of the corresponding columns, ensuring that they differ from the true row and that distractors are unique. The resulting options are fully serialized rows, and the five candidate rows are randomly shuffled. This task measures whether the model can distinguish authentic row-level combinations from plausible but fabricated near-matches.

3.4 Metrics and Baselines

The framework is expressly targeted at contamination, hence it uses metrics and statistical tests against baselines to assert contamination.

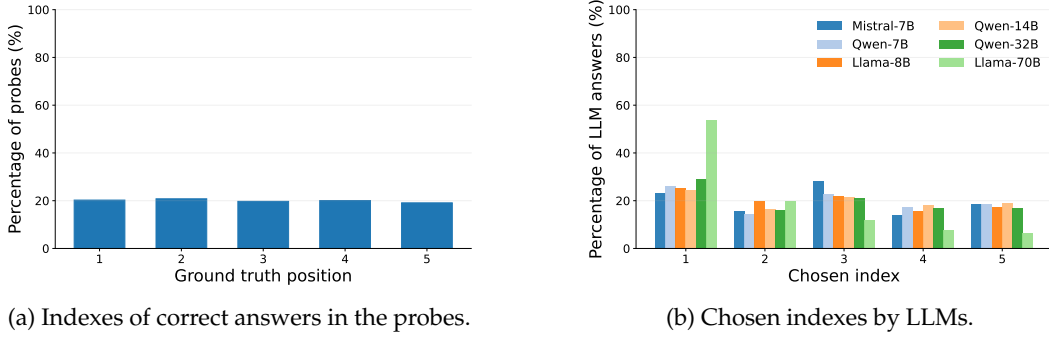


Figure 2: **Analysis of positional bias.** We report the average position across datasets and its variants of the correct answers in the contamination probes (Figure 2a), and the chosen indexes of LLMs during experiments (Figure 2b).

Evaluation Metrics. For a set of probes $\{1, \dots, n\}$, let $\hat{y}_i$ denote the model’s chosen option index and y_i the correct option index. The per-probe correctness indicator is $z_i = \mathbb{I}[\hat{y}_i = y_i]$.

By denoting $\mathcal{C}$ and $\mathcal{E}$ the completion and existence probe subsets, respectively, and $|\mathcal{C}| = n_{\text{comp}}$, $|\mathcal{E}| = n_{\text{exist}}$, the framework reports task-specific accuracies:

$$\mathcal{A}_{\text{comp}} = \frac{1}{n_{\text{comp}}} \sum_{i \in \mathcal{C}} z_i, \quad \mathcal{A}_{\text{exist}} = \frac{1}{n_{\text{exist}}} \sum_{i \in \mathcal{E}} z_i,$$

Beyond absolute accuracy, the framework performs paired comparisons between the model and each baseline using McNemar’s test (McNemar, 1947). Let b denote the number of probes where the model is correct and the baseline is wrong, and c the number where the model is wrong and the baseline is correct. The test is applied to these discordant pairs (b, c) . We use a one-sided alternative $H_1 : \{\text{model is better than baseline}\}$, and compute separate p -values for completion, and existence probes. This paired design is appropriate because both model and baseline are evaluated on the same set of probes, controlling for instance-level variability.

Baselines. The framework includes three explicit baselines: a *random* baseline, a *deterministic* baseline, and a *stochastic* baseline. The former selects one of the five options uniformly at random. This is the minimal reference level and captures pure chance performance under the multiple-choice protocol. The deterministic and stochastic baselines rely on the probability distribution of options in a probe; specifically, the deterministic baseline selects the option with the highest probability, while the stochastic baseline samples one option from the distribution. Details on how the probability distribution of options in a probe is derived are provided in Appendix C.

3.5 Interpretation of the Framework

Hereinafter, we consider a model contaminated if its statistical test against the random baseline is positive on real. This is a deliberate choice: we define contamination at the lowest detectable signal level, so that a significant test against the random baseline is read as evidence that the model has access to exploitable information about the original dataset beyond uninformed guessing, i.e., that the benchmark can no longer be treated as fully unseen data for that model. Beating the random baseline while failing to beat the deterministic or stochastic baseline carries weaker evidential weight than beating all three, but the two cases are not conflated: the random baseline is a necessary, not sufficient, condition for a stronger contamination claim, and success against the marginal-informed baselines signals a more pronounced form of contamination within this hierarchical interpretation of evidence. Additionally, the framework supports a structured interpretation of model behavior. Strong performance on real but not on like indicates reliance on authentic row-level structure rather than marginals. Retained performance on swapped suggests robustness to categorical relabeling and reliance on relational structure. Retained performance on obfuscated rules

out dependence on semantic cues (e.g., feature names or labels). Finally, a comparison with the deterministic and stochastic baselines shows how statistical information can be leveraged to solve the tasks.

4 Experiments

In this section, we present the experimental results. Section 4.1 describes the experimental setup, including the evaluated models, datasets, and research goals. Section 4.2 then addresses these goals through the empirical analysis.

4.1 Experimental Setup

Models. We evaluate seven large language models spanning different families and parameter scales. The open-weight models are Mistral 7B (Jiang et al., 2023a), Qwen-{7B, 14B, 32B} (Bai et al., 2023; Qwen et al., 2025) and Llama-8B (Grattafiori et al., 2024). In addition, we include GPT-OSS (OpenAI et al., 2025) and Llama 70B (Grattafiori et al., 2024). This model set allows us to study whether contamination signals are consistent across architectures, parameter scales, and model families.

Datasets. We evaluate our framework on eight public tabular datasets: adult (Becker & Kohavi, 1996), blood (Yeh, 2008), credit (Yeh, 2009), diabetes (Smith et al., 1988), gamma (Bock, 2004), iris (Fisher, 1936), mushroom (dat, 1981), titanic²; these datasets span multiple domains and feature types. Additionally, we include a synthetic dataset as a controlled negative reference point. Details on its construction are provided in Appendix A.3.

Framework Hyperparameters. We use 5 answer options per query and set the decoding temperature to 0 to ensure deterministic outputs. In the completion task, we mask 20% of the columns, a trade-off between test difficulty and reliability of the estimated accuracy; Section D.1 reports an ablation at 50% coverage. Statistical significance for both completion and existence tests is set at $\alpha = 0.01$. An ablation study on temperature and masking ratio is reported in the Appendix D.1.

Research Goals. Our experiments are organized around four primary research goals: (RG1) validating our framework’s ability to reliably detect tabular dataset contamination in LLMs; (RG2) introducing a comprehensive contamination registry across various datasets and models; (RG3) evaluating whether the extent of contamination varies across different model scales; and (RG4) demonstrating that our approach reveals subtle contamination signals missed by existing, memorization-based approaches.

4.2 Results

We first examine the limitations of existing memorization-based frameworks, then we present results from our framework, and summarize them in a contamination registry.

4.2.1 Memorization-based Framework Results

We compare our approach with the state-of-the-art framework of Bordt et al. (2024), reporting all four of its memorization tests. *Row completion* requires reconstructing a full row given n_p preceding rows and n_e few-shot examples, while *feature completion* predicts a masked feature given the remaining features and n_e examples. The *header* test shows the model the first n_p rows of the dataset and asks it to predict the next m rows. Finally, the *first token* (FT) test is identical to row completion, but accuracy is computed only on the first token of the completed row. We use the default hyperparameters.

Table 1 reports results on three widely used datasets. Memorization-based tests fail to detect contamination on canonical benchmarks such as adult and titanic, where all models achieve near-zero reconstruction accuracy. Given the prevalence of these datasets, this absence of detectable memorization is unexpected and is contradicted by the results in the

²<https://www.kaggle.com/datasets/yasserh/titanic-dataset>

Table 1: Performance of the memorization-based tests of Bordt et al. (2024) across models and three famous datasets: adult, iris, and titanic.

Datasets	Task	Models					
		<i>Mistral-7B</i>	<i>Qwen-7B</i>	<i>Llama-8B</i>	<i>Qwen-14B</i>	<i>Qwen-32B</i>	<i>GPT-OSS</i>
adult	Row	0/250	0/250	0/250	0/250	0/250	0/250
	Feature	0/250	0/250	0/250	0/250	0/250	0/250
	Header	0/5	0/5	0/5	0/6	0/5	0/0
	FT	0/250	0/250	0/250	0/250	0/250	0/250
iris	Row	52/250	109/250	5/250	1/250	182/250	48/250
	Feature	2/250	17/250	6/250	5/250	9/250	58/250
	Header	0/16	1/19	1/17	0/18	3/17	0/17
	FT	0/250	0/250	0/250	0/250	0/250	0/250
titanic	Row	10/250	0/250	0/250	10/250	12/250	0/250
	Feature	10/250	6/250	2/250	0/250	3/250	13/250
	Header	0/19	0/18	0/19	0/16	0/19	0/1
	FT	0/250	0/250	0/250	0/250	0/250	0/250

next section. More critically, these tests lack a criterion to determine when memorization implies contamination. This is evident on *iris*: some models (e.g., *Qwen-32B*, 182/250 correct row completions) show clear memorization, while others (e.g., *Llama-8B*, 5/250) do not. Their framework provides no statistical guidance to interpret such variability, leaving contamination ambiguous at both the dataset and model levels. Overall, verbatim memorization is a coarse and unreliable proxy for tabular data contamination, underscoring the need for a more controlled, fine-grained framework.

4.2.2 Proposed Framework Results

We analyze the results obtained with our framework. Figure 3 illustrates representative behaviors, while complete results are reported in Appendix D.

Detecting contamination on titanic. The completion results on *titanic* (Figure 3a) provide a clear example of the effectiveness of our framework. On the *real* variant, all models achieve accuracy significantly above the random baseline according to our statistical test. Under the decision rule introduced in Section 3, this constitutes sufficient evidence of contamination for this dataset-model setting. Importantly, this addresses **RG1**, demonstrating that our framework can detect contamination in tabular datasets within LLMs.

The comparison across variants further clarifies the nature of the detected signal. Performance remains relatively high on *like*, indicating that part of the advantage can be explained by knowledge of dataset-specific marginals. However, accuracy drops substantially on *swapped* and *obfuscated*, suggesting that the signal expressed on the *real* dataset is not merely due to generic tabular reasoning. Rather, the pattern is consistent with prior knowledge tied to the original dataset distribution and, at least in part, to its original feature-value semantics. Lastly, the comparison with the deterministic and stochastic baselines shows that the contamination effect is not solely due to statistical knowledge of the dataset.

Completion versus existence probes. The existence results on *titanic* (Figure 3b) reveal a complementary pattern. In this case, only the larger models are significantly above the random baseline on the *real* variant. This suggests that the existence task is intrinsically more difficult than the completion task. The same tendency is observed on other datasets (see Figure 8), indicating that completion probes are generally the more sensitive instrument for contamination detection. At the same time, the existence task remains informative. Even though it is less sensitive, it still identifies contamination signals on *titanic* for the larger models, whereas the memorization-based framework reports near-zero row reconstruction accuracy on this dataset (see Section 4.2.1). This is even more evident

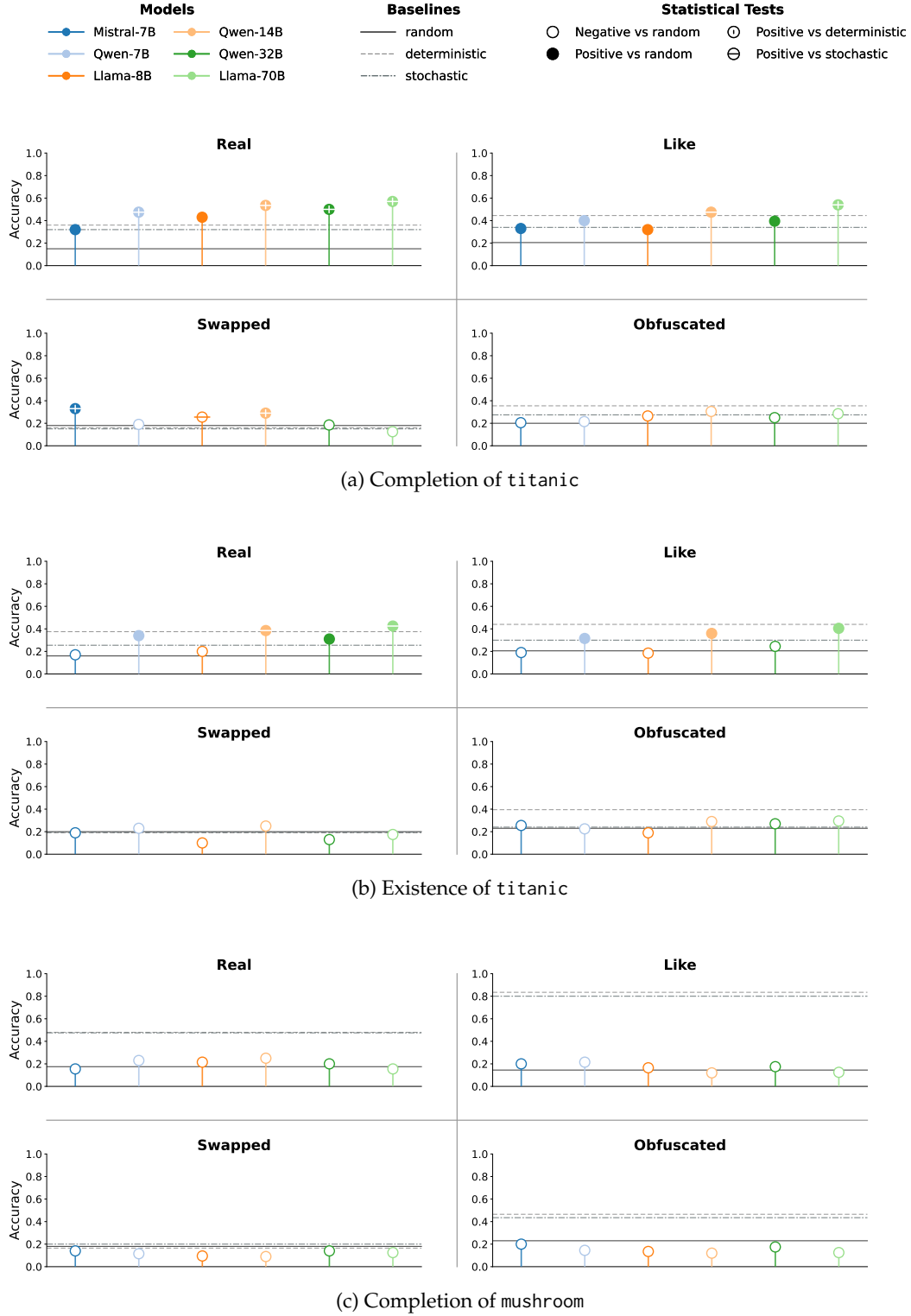


Figure 3: **Task performance across dataset variants and models.** Baselines are represented as horizontal lines, and statistical test results using filled dots for detected contamination (positive against random), and horizontal/vertical bars on dots for superiority against statistical baselines.

for the adult dataset (Figures 4a and 4b). This addresses **RG4**: our framework captures contamination phenomena that are not revealed by existing methodologies.

A negative case: mushroom. Figure 3c provides an equally important negative example. For mushroom, completion accuracy on the real variant remains close to chance and does not show a stable advantage over the transformed variants. In other words, our framework does not detect a contamination signal for this dataset. The result is important methodologically: it shows that the framework is not trivially flagging every public benchmark as contaminated, but can also return negative findings when the statistical evidence is insufficient. This is further validated by the synthetic dataset (see Figure 5).

Research Goals. Taken together, these results address all the research goals of this section. First, the titanic case demonstrates that our framework can reliably detect contamination, thereby addressing **RG1**. Second, the full set of results in Appendix D naturally supports the construction of a dataset-model contamination registry, depicted in Table 2, which addresses **RG2**. Third, a clear scale-related trend emerges from Figures 3a and 3b and from the additional results in the appendix: contamination signals generally become stronger as model size increases. This directly addresses **RG3** and suggests that larger models are more likely to express the consequences of prior dataset exposure. Finally, the contrast between our results and those of the memorization-based framework shows that contamination signals can be present even when verbatim reconstruction fails, addressing **RG4**.

Table 2: **Contamination registry across datasets and models.** A checkmark (✓) indicates that our framework detects contamination for the corresponding dataset-model pair, while a cross (×) indicates no detected contamination under our evaluation protocol. The results highlight contamination consistently detected on titanic, adult, credit, and iris.

Datasets	Models					
	<i>Mistral-7B</i>	<i>Qwen-7B</i>	<i>Llama-8B</i>	<i>Qwen-14B</i>	<i>Qwen-32B</i>	<i>Llama-70B</i>
adult	✓	✓	✓	✓	✓	✓
blood	×	×	×	×	×	×
credit	×	✓	✓	✓	✓	✓
diabetes	×	×	×	×	×	×
gamma	×	×	×	×	×	×
iris	×	✓	×	✓	✓	✓
mushroom	×	×	×	×	×	×
synthetic	×	×	×	×	×	×
titanic	✓	✓	✓	✓	✓	✓

5 Discussion and Limitations

Our results show that tabular data contamination in LLMs is a real methodological concern, challenging the common assumption that public tabular benchmarks are unseen at evaluation time. If a model has prior knowledge of a benchmark, evaluation reflects both generalization and exposure, inflating scores and weakening claims of progress. Crucially, this concern does not depend on observing a measurable gain: contamination is problematic even when it yields no detectable increase in downstream accuracy, because it already violates the assumption that benchmark examples are unseen at evaluation time. The primary harm is the loss of the guarantee that makes benchmark numbers interpretable, not the size of the inflation they exhibit. We therefore argue that contamination assessment should be standard in studies evaluating LLMs on public tabular datasets. Contaminated datasets need not be discarded, but their use should be reported transparently and their results interpreted as potentially biased upward, and claims about model generalization should be correspondingly softened. Like dataset documentation or significance testing, contamination assessment does not replace evaluation; it makes it interpretable. A second implication concerns model scale: contamination tends to be more visible in larger models, suggesting that they are better able to exploit prior exposure. As a result, contamination can bias not only absolute performance estimates but also comparisons across model sizes,

making scale-related gains appear larger than they would on truly unseen data. This is particularly important because benchmark leaderboards are often used to support claims about the superiority of larger models. Our results indicate that part of this superiority may, in contaminated settings, reflect a greater ability to exploit contamination rather than a cleaner improvement in generalization.

We acknowledge that our work has some limitations. First, *it can be used to show the presence of data contamination, but never to show its absence*. A negative result in our framework must be interpreted as an absence of detected evidence, not as evidence that the model has never been exposed to the dataset. This is especially relevant for datasets such as mushroom, where we detect no contamination despite the dataset being old and publicly available. A second limitation is sensitivity: although our framework is more sensitive than prior approaches, it may still miss contamination. This is plausible for old, widely circulated benchmarks and suggests that contamination detection remains, in part, a problem of test sensitivity. We conjecture that contamination detectability is inversely related to probe difficulty: verbatim memorization is the hardest and least sensitive signal, row existence is intermediate, and row completion is the easiest and most sensitive in practice. This interpretation is consistent with our coverage ablation: higher coverage makes row completion probes harder and contamination harder to detect. A more principled way to calibrate the framework’s sensitivity and specificity would be to establish ground-truth contamination through controlled injection, e.g., via continued pretraining on post-release or synthetic tabular datasets. Future work should therefore investigate even more sensitive protocols, calibrated against such ground-truth contamination, while preserving enough control to avoid false positives driven by superficial dataset regularities. A third limitation concerns the granularity of the knowledge that contamination induces. Even when a model is contaminated, it is not obvious *what* exactly the model knows about the dataset. Our crafted variants provide a first step toward discovering the kind of knowledge. However, this analysis is still limited. It does not yet fully characterize which parts of the dataset the model is confident about, which attributes are more strongly affected by contamination, or how contamination interacts with task formulation. A more fine-grained understanding of contaminated knowledge remains an important direction for future work. A final limitation is that we do not quantify how the detected contamination translates into inflated downstream performance. Our probes establish that dataset knowledge is present, not how much of a benchmark score it accounts for. A systematic study of downstream inflation requires a dedicated controlled setup, and must account for the fact that leakage need not surface as higher accuracy if the retained information is not elicited by the prompting and evaluation protocol in use. We leave this to future work.

6 Conclusion

In this work, we propose a framework for assessing contamination in tabular datasets by crafting multiple-choice queries and performing comparative evaluation using dataset variants and non-neural baselines. The main message of this paper is twofold. First, contamination in tabular benchmarks is real and methodologically consequential, so evaluations that ignore it risk overstate both absolute performance and relative differences between models. Second, the contamination assessment itself remains an open problem: our framework provides a useful, more sensitive tool, but not a complete solution. We believe that this work will motivate the community to treat contamination analysis as a necessary part of rigorous LLM evaluation, and to develop stronger methods for identifying not only whether a model has seen a dataset before, but also which properties of that dataset continue to influence its behavior.

Ethical Statement

This work does not involve personal data or sensitive information. All experiments are conducted on publicly available benchmark datasets commonly used in prior work. We do not identify any direct ethical risks or broader societal harms arising from this study. On the

contrary, by improving the rigor of contamination detection, our work contributes to a more reliable and transparent evaluation of large language models.

Acknowledgments

This work was partially supported by the following projects: GHOST – Protecting User Privacy from Community Detection in Social Networks (B83C24007070005) and SEEDS – Sustainable Ecosystems in Evolving Digital Societies, funded by Sapienza University of Rome through the “Progetti di Ricerca Grandi” and “Progetti Dipartimentali” programs, respectively.

References

- Mushroom. UCI Machine Learning Repository, 1981. DOI: <https://doi.org/10.24432/C5959T>.
- Jinze Bai, Shuai Bai, and et al. Qwen technical report, 2023. URL <https://arxiv.org/abs/2309.16609>.
- Simone Balloccu, Patrícia Schmidtová, Mateusz Lango, and Ondřej Dušek. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source llms. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 67–93, 2024.
- Debangshu Banerjee, Tarun Suresh, Shubham Ugare, Sasa Misailovic, and Gagandeep Singh. Crane: Reasoning with constrained llm generation. *arXiv preprint arXiv:2502.09061*, 2025.
- Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: <https://doi.org/10.24432/C5XW20>.
- R. Bock. MAGIC Gamma Telescope. UCI Machine Learning Repository, 2004. DOI: <https://doi.org/10.24432/C52C8B>.
- Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, and et al. On the opportunities and risks of foundation models, 2022. URL <https://arxiv.org/abs/2108.07258>.
- Sebastian Bordt, Harsha Nori, Vanessa Rodrigues, Besmira Nushi, and Rich Caruana. Elephants never forget: Memorization and learning of tabular data in large language models. *arXiv preprint arXiv:2404.06209*, 2024.
- Vadim Borisov, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. Language models are realistic tabular data generators. *arXiv preprint arXiv:2210.06280*, 2022.
- Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX security symposium (USENIX security 19)*, pp. 267–284, 2019.
- Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pp. 2633–2650, 2021.
- Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*, 2022.
- Yuxing Cheng, Yi Chang, and Yuan Wu. A survey on data contamination for large language models, 2025. URL <https://arxiv.org/abs/2502.14425>.

- Dario Di Palma, Felice Antonio Merra, Maurizio Sfilio, Vito Walter Anelli, Fedelucio Narducci, and Tommaso Di Noia. Do llms memorize recommendation datasets? a preliminary study on movielens-1m. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, pp. 2582–2586. ACM, July 2025. doi: 10.1145/3726302.3730178. URL <http://dx.doi.org/10.1145/3726302.3730178>.
- Xi Fang, Weijie Xu, Fiona Anting Tan, Jiani Zhang, Ziqing Hu, Yanjun Qi, Scott Nickleach, Diego Socolinsky, Srinivasan Sengamedu, and Christos Faloutsos. Large language models (llms) on tabular data: Prediction, generation, and understanding—a survey. *arXiv preprint arXiv:2402.17944*, 2024.
- R. A. Fisher. Iris. UCI Machine Learning Repository, 1936. DOI: <https://doi.org/10.24432/C56C76>.
- Josh Gardner, Juan C Perdomo, and Ludwig Schmidt. Large scale transfer learning for tabular data via language modeling. *Advances in Neural Information Processing Systems*, 37: 45155–45205, 2024.
- Shahriar Golchin and Mihai Surdeanu. Time travel in llms: Tracing data contamination in large language models, 2024. URL <https://arxiv.org/abs/2308.08493>.
- Shahriar Golchin and Mihai Surdeanu. Data contamination quiz: A tool to detect and estimate contamination in large language models. *Transactions of the Association for Computational Linguistics*, 13:809–830, 2025.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, and et al. The llama 3 herd of models, 2024. URL <https://arxiv.org/abs/2407.21783>.
- F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4), December 2015. ISSN 2160-6455. doi: 10.1145/2827872. URL <https://doi.org/10.1145/2827872>.
- Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. Tabllm: Few-shot classification of tabular data with large language models, 2023. URL <https://arxiv.org/abs/2210.10723>.
- Albert Q. Jiang, Alexandre Sablayrolles, and et al. Mistral 7b, 2023a. URL <https://arxiv.org/abs/2310.06825>.
- Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Wayne Xin Zhao, and Ji-Rong Wen. Structgpt: A general framework for large language model to reason over structured data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9237–9251, 2023b.
- Nicholas Kroege, Dan Ley, Satyapriya Krishna, Chirag Agarwal, and Himabindu Lakkaraju. In-context explainers: Harnessing llms for explaining black box models. *arXiv preprint arXiv:2310.05797*, 2023.
- Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.
- Avanika Narayan, Ines Chami, Laurel Orr, Simran Arora, and Christopher Ré. Can foundation models wrangle your data? *arXiv preprint arXiv:2205.09911*, 2022.
- Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A Feder Cooper, Daphne Ippolito, Christopher A Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*, 2023.
- Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models, 2024. URL <https://arxiv.org/abs/2307.06435>.

- OpenAI, Sandhini Agarwal, Lama Ahmad, and et al. gpt-oss-120b and gpt-oss-20b model card, 2025. URL <https://arxiv.org/abs/2508.10925>.
- Qwen, An Yang, Baosong Yang, and et al. Qwen2.5 technical report, 2025. URL <https://arxiv.org/abs/2412.15115>.
- Anna Rogers and Alexandra Sasha Luccioni. Position: Key claims in llm research have a long tail of footnotes. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 42647–42665, 2024.
- Benoît Ronval, Pierre Dupont, and Siegfried Nijssen. Detection of large language model contamination with tabular data. In *International Symposium on Intelligent Data Analysis*, pp. 234–245. Springer, 2025.
- Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10776–10787, 2023.
- David W Scott. On optimal and data-based histograms. *Biometrika*, 66(3):605–610, 1979.
- Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models, 2024. URL <https://arxiv.org/abs/2310.16789>.
- Jack W. Smith, J. E. Everhart, W. C. Dickson, W. C. Knowler, and R. S. Johannes. Using the adap learning algorithm to forecast the onset of diabetes mellitus. *Proceedings of the Annual Symposium on Computer Application in Medical Care*, pp. 261–265, 1988. URL <https://pmc.ncbi.nlm.nih.gov/articles/PMC2245318/>.
- Yiming Sun, Xumeng Wen, Shun Zheng, Xiaowei Jia, and Jiang Bian. Scaling generative tabular learning for large language models. In *NeurIPS 2024 Third Table Representation Learning Workshop*, 2024.
- Zhi Rui Tam, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung-yi Lee, and Yun-Nung Chen. Let me speak freely? a study on the impact of format restrictions on performance of large language models. *arXiv preprint arXiv:2408.02442*, 2024.
- Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, et al. Chain-of-table: Evolving tables in the reasoning chain for table understanding. *arXiv preprint arXiv:2401.04398*, 2024.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. Benchmarking benchmark leakage in large language models. *arXiv preprint arXiv:2404.18824*, 2024.
- I-Cheng Yeh. Blood Transfusion Service Center. UCI Machine Learning Repository, 2008. DOI: <https://doi.org/10.24432/C5GS39>.
- I-Cheng Yeh. Default of Credit Card Clients. UCI Machine Learning Repository, 2009. DOI: <https://doi.org/10.24432/C55S3H>.
- Kun Zhou, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. Don’t make your llm an evaluation benchmark cheater, 2023. URL <https://arxiv.org/abs/2311.01964>.

A Details of Probes and Datasets Construction

A.1 Probes Construction

A key methodological aspect is that candidate rows and masking patterns are *aligned across variants*. The same underlying row indices are selected for the real, like, swapped, and obfuscated conditions, and the same masked column subsets are reused after mapping names appropriately. This paired construction is crucial because it ensures that differences in accuracy across variants are attributable to the transformation itself rather than to changes in probe difficulty. For each dataset variant, the framework generates 400 probes in total, split equally between completion and existence tasks.

A.1.1 Row Selection

Probe generation is not uniform over rows. Instead, the framework preferentially samples *rare* or distinctive rows, which are more informative for contamination analysis. Each row receives a rarity score computed from inverse empirical frequencies over randomly selected small subsets of columns. Specifically, in several trials, the algorithm selects a small subset of columns and computes an inverse-frequency product for each row; the final rarity score is the average of the log-transformed products across trials.

Intuitively, rows composed of uncommon value combinations receive higher scores. The generator ranks rows by this rarity measure, retains a pool of highly ranked candidates, shuffles that pool, and then constructs probes from it. This design biases the benchmark toward rows whose correct identification is less likely to be explained by coarse statistical regularities alone, and strengthens the reliability of the contamination-detected cases.

A.1.2 Column Selection

The framework allows evaluating contamination at different *coverage* levels (e.g., 20%, 50%), corresponding to the proportion of non-target columns that are involved in the probe. Let p denote the number of non-target columns in the dataset. The number of masked or mutated columns, denoted m , is computed as

$$m = \max(1, \min(p, \lfloor p \cdot \text{coverage} + 0.5 \rfloor)).$$

Hence, coverage is operationalized as a rounded proportion of the available non-target attributes. The target column is excluded from masking. This choice is methodologically important: it prevents the task from degenerating into a direct label-prediction problem and keeps the focus on contamination of feature configurations and row identity.

For each probe, the set of masked or mutated columns is chosen with a mixed strategy intended to cover both categorical and numerical information:

- categorical columns are preferred according to empirical entropy, with a preference for non-binary variables when possible;
- numerical columns are preferred according to variance;
- when only one column is selected, binary categorical variables are avoided if non-binary or numerical alternatives exist.

This procedure avoids trivially easy probes and encourages questions that depend on informative attributes.

A.2 Dataset Variants

Real. The real variant is the unmodified cleaned dataset. It preserves the original feature names, original values, marginal distributions, and joint row-level dependencies. This is the reference condition in which contamination, if present, should be most directly expressed.

Like. The like variant is obtained by independently resampling each column from its empirical marginal distribution. Formally, if the original dataset has columns $c_1, \dots, c_p$,

then the transformed dataset D^{like} is generated by sampling each entry of column c_j independently from the observed non-missing values of c_j in D . This transformation preserves univariate marginals but destroys row-wise joint structure. Accordingly, success on this condition reflects knowledge compatible with column-level regularities, but not necessarily contamination of authentic rows.

Obfuscated. The obfuscated variant removes surface-level semantic cues. All feature names are renamed sequentially, e.g., to $f_1, f_2, \dots$, with the target column, when known, renamed to target. Categorical values are also replaced by anonymous codes such as c_0, c_1 , and so on. For binary targets, the target is converted to a binary encoding, typically $\{0, 1\}$. Thus, the obfuscated condition preserves the row-level structure and the distributional pattern of values, but suppresses human-interpretable semantics associated with feature names, categorical feature values, and category labels. This is a particularly stringent condition for distinguishing contamination from ordinary world knowledge or familiarity with the dataset name. Note that this variant has a stronger effect on datasets composed entirely of categorical features.

Swapped. The swapped variant preserves the table schema but applies a strict within-column derangement. For each categorical feature, the set of unique observed values is permuted so that no category maps to itself. If a column has categories $\mathcal{V}_c$, the method constructs a bijection

$$\pi_c : \mathcal{V}_c \rightarrow \mathcal{V}_c$$

such that $\pi_c(v) \neq v$ for every $v \in \mathcal{V}_c$. For numerical features whose values fall within the interval $\mathcal{I}_n := [N_{\min}, N_{\max}]$, the bijection used is

$$\pi_n : \mathcal{I}_n \rightarrow \mathcal{I}_n, \quad \pi_n(x) := N_{\max} - x + N_{\min}$$

This operation preserves the row structure and the pattern of co-occurrence across columns.

A.3 Synthetic Dataset

We construct a synthetic dataset as a negative control to validate that the framework does not detect contamination when contamination is absent. By design, this dataset has no prior exposure in LLM training data; thus, any detected signal would indicate a failure mode of the method. Importantly, the framework is not intended to certify the absence of contamination, but to provide evidence of its presence.

The dataset is generated deterministically (fixed random seed set to 42) using *make_classification*³, a predefined function of *scikit-learn*, with 50,000 samples, 10 initial continuous features, and a binary target (class imbalance 75/25, 1% label noise, moderate separability). Five categorical features are then derived from the continuous space: (i) quantile binning (9 bins), (ii) quantile binning with 10% random reassignment (6 bins), (iii) probabilistic sampling with weak target dependence (10 categories), (iv) KMeans clustering on a 2D subspace (4 clusters), and (v) quantile binning with 20% random flips (5 bins). All categorical features are explicitly cast to discrete types. The original continuous features used to construct categorical variables are removed. The final dataset contains 15 predictors (10 continuous, 5 categorical) and one binary target.

A.4 Probe Examples

In this subsection, we provide example contamination probes for the *adult* dataset. Since the real, like, and swapped variants cannot be distinguished when inspecting an individual probe in isolation, we include only examples from the real variant (Table 3) and the obfuscated variant (Table 4).

³https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_classification.html

Table 3: **Example probe of the completion and existence tasks for the adult dataset (real variant, coverage= 20%).** The upper block reports the probe for the completion task, while the lower block reports the probe for the existence task.

COMPLETION

Complete the masked columns in this row with '?' marks. Choose the correct values.

ROW: age=?, workclass=Private, fnlwgt=198654, education=Prof-school, educational-num=15, marital-status=Married-civ-spouse, occupation=?, relationship=Husband, race=Asian-Pac-Islander, gender=Male, capital-gain=0, capital-loss=?, hours-per-week=67, native-country=India, income=>50K

OPTIONS:

1. occupation=Adm-clerical, capital-loss=1669, age=82
 2. occupation=Farming-fishing, capital-loss=2282, age=84
 3. occupation=Prof-specialty, capital-loss=2415, age=39
 4. occupation=Farming-fishing, capital-loss=1421, age=37
 5. occupation=Farming-fishing, capital-loss=1411, age=62
-

EXISTENCE

Which of the following complete rows exists in the dataset? Choose the index only.

OPTIONS:

1. age=81, workclass=Federal-gov, fnlwgt=424340, education=Bachelors, educational-num=8, marital-status=Widowed, occupation=Prof-specialty, relationship=Not-in-family, race=White, gender=Female, capital-gain=0, capital-loss=1668, hours-per-week=3, native-country=Hungary, income<=50K
 2. age=81, workclass=Self-emp-not-inc, fnlwgt=123959, education=Bachelors, educational-num=13, marital-status=Widowed, occupation=Prof-specialty, relationship=Not-in-family, race=White, gender=Female, capital-gain=0, capital-loss=1668, hours-per-week=3, native-country=Hungary, income<=50K
 3. age=81, workclass=State-gov, fnlwgt=174778, education=Bachelors, educational-num=8, marital-status=Widowed, occupation=Prof-specialty, relationship=Not-in-family, race=White, gender=Female, capital-gain=0, capital-loss=1668, hours-per-week=3, native-country=Hungary, income<=50K
 4. age=81, workclass=Federal-gov, fnlwgt=258761, education=Bachelors, educational-num=15, marital-status=Widowed, occupation=Prof-specialty, relationship=Not-in-family, race=White, gender=Female, capital-gain=0, capital-loss=1668, hours-per-week=3, native-country=Hungary, income<=50K
 5. age=81, workclass=Local-gov, fnlwgt=353281, education=Bachelors, educational-num=16, marital-status=Widowed, occupation=Prof-specialty, relationship=Not-in-family, race=White, gender=Female, capital-gain=0, capital-loss=1668, hours-per-week=3, native-country=Hungary, income<=50K
-

Table 4: **Example probe of the completion and existence tasks for the adult dataset (obfuscated variant, coverage= 20%).** The upper block reports the probe for the completion task, while the lower block reports the probe for the existence task.

COMPLETION
Complete the masked columns in this row with '?' marks. Choose the correct values.
ROW: f1=?, f2=c2, f3=198654, f4=c14, f5=15, f6=c2, f7=?, f8=c0, f9=c1, f10=c1, f11=0, f12=?, f13=67, f14=c18, target=1
OPTIONS:
1. f7=c8, f12=1669, f1=82
2. f7=c9, f12=2282, f1=84
3. f7=c1, f12=2415, f1=39
4. f7=c3, f12=1421, f1=37
5. f7=c3, f12=1411, f1=62
EXISTENCE
Which of the following complete rows exists in the dataset? Choose the index only.
OPTIONS:
1. f1=81, f2=c000, f3=424340, f4=c009, f5=8, f6=c006, f7=c009, f8=c001, f9=c004, f10=c000, f11=0, f12=1668, f13=3, f14=c017, target=0
2. f1=81, f2=c004, f3=123959, f4=c009, f5=13, f6=c006, f7=c009, f8=c001, f9=c004, f10=c000, f11=0, f12=1668, f13=3, f14=c017, target=0
3. f1=81, f2=c005, f3=174778, f4=c009, f5=8, f6=c006, f7=c009, f8=c001, f9=c004, f10=c000, f11=0, f12=1668, f13=3, f14=c017, target=0
4. f1=81, f2=c000, f3=258761, f4=c009, f5=15, f6=c006, f7=c009, f8=c001, f9=c004, f10=c000, f11=0, f12=1668, f13=3, f14=c017, target=0
5. f1=81, f2=c001, f3=353281, f4=c009, f5=16, f6=c006, f7=c009, f8=c001, f9=c004, f10=c000, f11=0, f12=1668, f13=3, f14=c017, target=0

B LLM Interrogation Protocol

Each probe is converted into a standardized prompt template. The model is told that it will receive a multiple-choice question about a tabular dataset, that the options are numbered from 1 to 5, and that it should first reason step by step and then produce a final answer consisting of a single integer.

To bypass the limitations of constrained decoding while supporting intermediate reasoning (Tam et al., 2024; Wei et al., 2022), models are prompted to generate free-form text before outputting their final integer choice within a programmatically parsed tag (Banerjee et al., 2025). Therefore, the final answer must be enclosed within explicit delimiters to allow reliable post-processing. This yields a robust, machine-parseable evaluation pipeline while still allowing the model to produce an unconstrained rationale.

Tables 5 and 6 illustrate the interrogation protocol through a prompt example for the real and obfuscated variant, respectively. Notice that in the real variant (the same holds for like and swapped), we explicitly specify the dataset name, since we want the model to leverage prior knowledge about it, if it has any. In contrast, in the obfuscated variant, we name it as “a tabular dataset”, describing only the number and type of features.

Table 5: **Example prompt for the completion task on the iris dataset (real variant, coverage=20%).** The prompt specifies the response format, provides a row with one masked feature, and asks the model to identify the correct value from five multiple-choice candidates.

You will be given a MULTIPLE-CHOICE question about the Iris dataset.
Each question is MULTIPLE-CHOICE with options numbered from 1 to 5.

IMPORTANT INSTRUCTIONS:

1. First, reason step-by-step about the question and each option. Think through which option is most plausible.
2. After your reasoning, write your final answer as a single integer between 1 and 5 (the number of the chosen option).
3. You MUST wrap your final answer between these exact tags: [FINAL_ANSWER_start] < number> [FINAL_ANSWER_end]
Example: [FINAL_ANSWER_start] 3 [FINAL_ANSWER_end]
4. If you are uncertain, still choose the single most plausible option.

QUESTION:

Complete the masked columns in this row with '?' marks. Choose the correct values.

ROW: sepal_length=?, sepal_width=3.3, petal_length=1.7, petal_width=0.5, species=Iris-setosa

OPTIONS:

- 1: sepal_length=6.1
- 2: sepal_length=5.1
- 3: sepal_length=5.0
- 4: sepal_length=6.8
- 5: sepal_length=7.0

First reason about the question, then provide your final answer between the tags.

Table 6: **Example prompt for the completion task on the iris dataset (obfuscated variant, coverage=20%).** The prompt specifies anonymized information about the dataset, the response format, provides a row with one masked feature, and asks the model to identify the correct value from five multiple-choice candidates.

You will be given a MULTIPLE-CHOICE question about a tabular dataset. This dataset contains 4 features in total. They are all numerical features: f1, f2, f3, f4. The target variable is categorical with 3 classes.

Each question is MULTIPLE-CHOICE with options numbered from 1 to 5.

IMPORTANT INSTRUCTIONS:

[...]

QUESTION:

Complete the masked columns in this row with '?' marks. Choose the correct values.\nROW: f1=?, f2=3.3, f3=1.7, f4=0.5, target=c0

OPTIONS:

- 1: f1=6.1
- 2: f1=5.1
- 3: f1=5.0
- 4: f1=6.8
- 5: f1=7.0

First reason about the question, then provide your final answer between the tags.

C Probability Distribution of Options in Probes

C.1 Deterministic Marginal Log-Likelihood Baseline

The deterministic baseline scores each option using column-wise empirical marginals estimated from the original cleaned dataset. For each categorical column c , if a value v appears $\text{count}_c(v)$ times, the baseline estimates

$$p(v | c) = \frac{\text{count}_c(v) + 1}{N_c + |\mathcal{V}_c| + 1},$$

where N_c is the number of non-missing observations in column c and $|\mathcal{V}_c|$ is the number of distinct observed values. This corresponds to Laplace smoothing. Similarly, for the numerical columns c , we apply the *binning* approach. Specifically, we compute the number of bins using Scott’s rule (Scott, 1979), thus obtaining a set of intervals $\{I_1, \dots, I_B\}$. Next, the baseline estimate

$$p(v | c) = p(\text{bin}(v) | c) = \frac{\text{count}_{\text{bin}}(v) + 1}{N_c + B}$$

where B is the number of bins and $\text{count}_{\text{bin}}(v)$ returns the number of observations which are within the bin of v . Finally, given an option o , represented as a set of column-value assignments, the score is defined as

$$S(o) = \sum_{(c,v) \in o} \log p(v | c).$$

Unseen values receive a large negative default score.

The deterministic baseline selects the option with the maximal score:

$$\hat{y} = \arg \max_o S(o),$$

with random tie-breaking when necessary. For both completion and existence probes, o includes only the masked columns. In the obfuscated variant, options are first translated back into the original feature/value space via stored metadata, and only then scored against the real-data marginals. In contrast, in swapped and like variants, we compute the score directly against the real-data marginals. This makes the baseline comparable across variants.

C.2 Stochastic Marginal Log-Likelihood Baseline

The stochastic baseline uses the same score $S(o)$ as above, but converts scores into a softmax distribution:

$$P(\hat{y} = o_i) = \frac{\exp(S(o_i) - \max_j S(o_j))}{\sum_j \exp(S(o_j) - \max_k S(o_k))}.$$

An option is then sampled according to this distribution. This baseline captures what can be achieved by exploiting marginal column statistics without making a purely greedy decision.

D Additional Results

In this section, we report additional figures, i.e., Figures 4 to 7. In Section D.1, we conduct an ablation on the hyperparameters of the framework, i.e., the temperature of the model and the level of coverage. Lastly, in Section D.2 we report raw results for all settings.

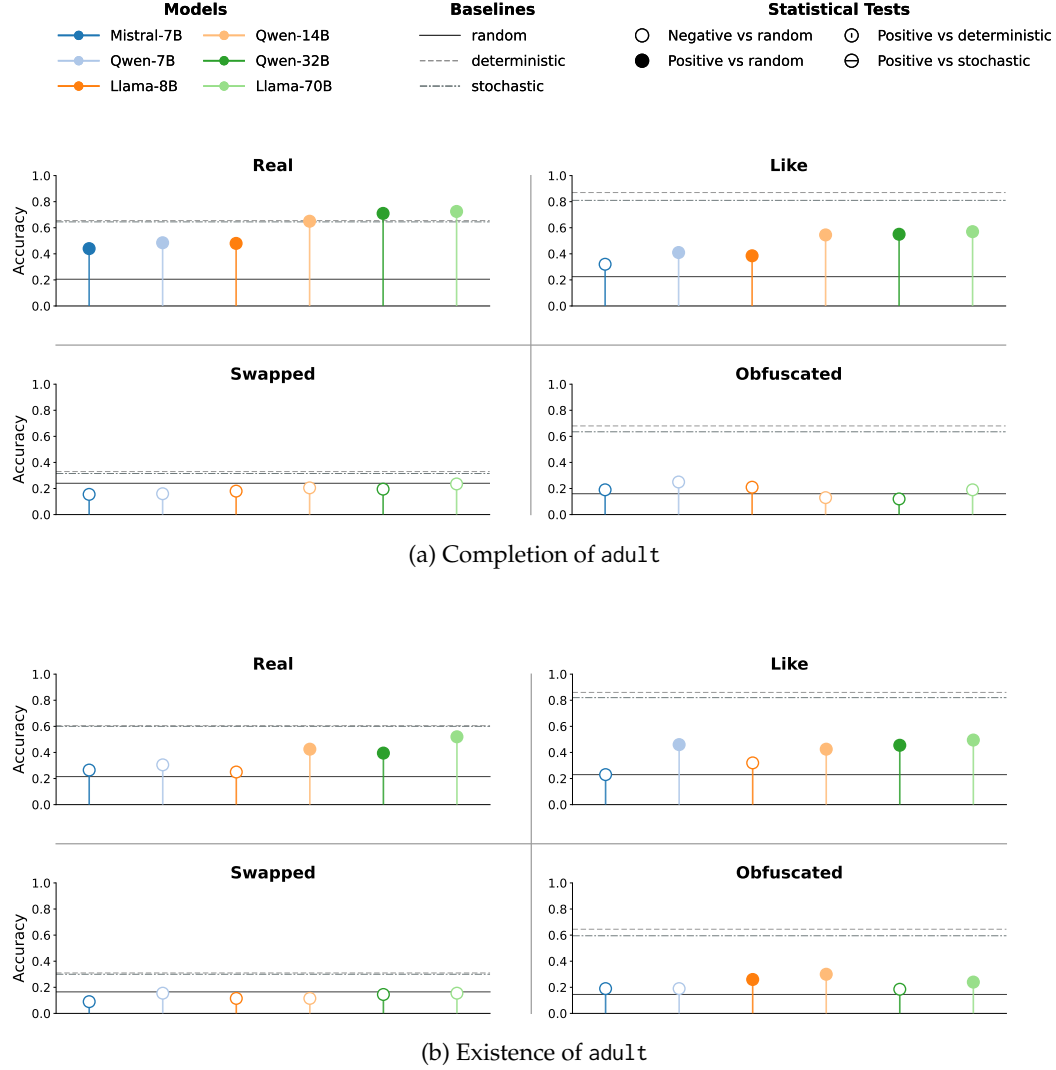
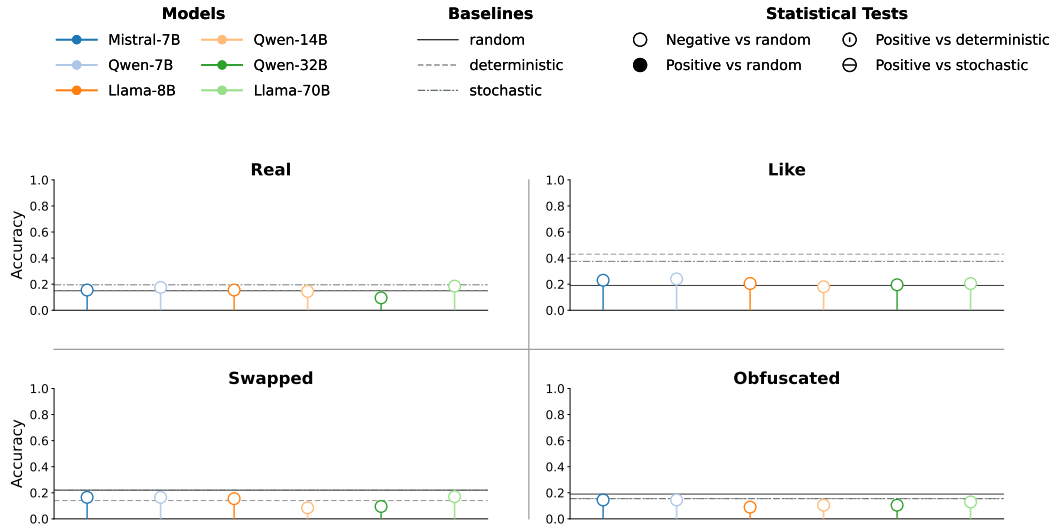
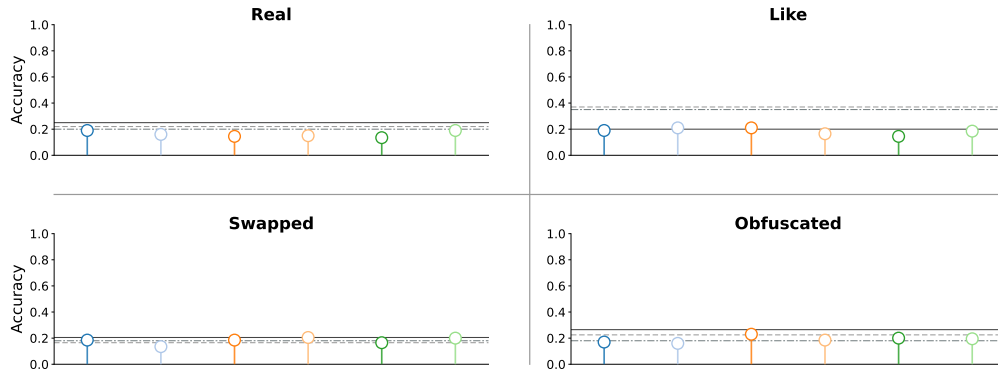


Figure 4: **Task performance for adult across dataset variants and models.** We represent baselines as horizontal lines, and statistical test results using filled dots for detected contamination (positive against random), and horizontal/vertical bars on dots for superiority against statistical baselines.

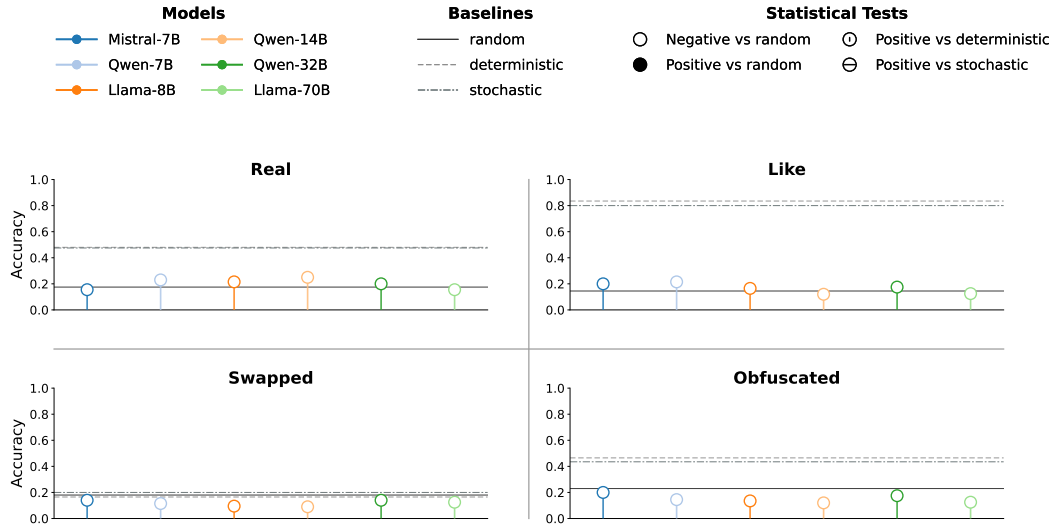


(a) Completion of synthetic

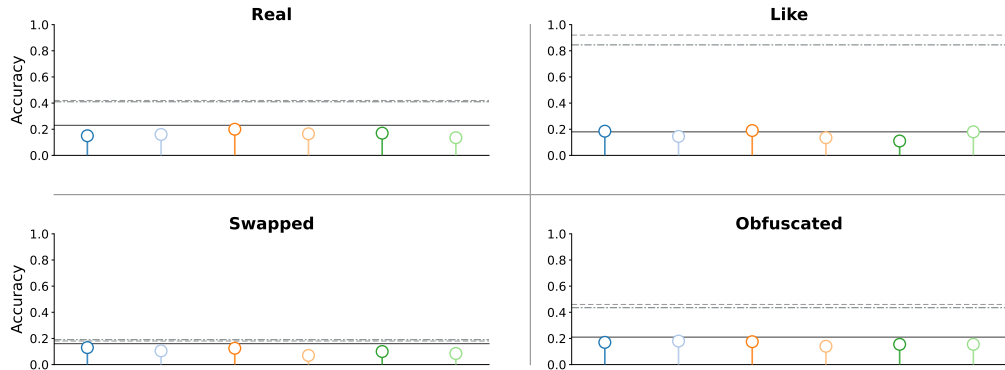


(b) Existence of synthetic

Figure 5: Task performance for synthetic across dataset variants and models. We represent baselines as horizontal lines, and statistical test results using filled dots for detected contamination (positive against random), and horizontal/vertical bars on dots for superiority against statistical baselines.

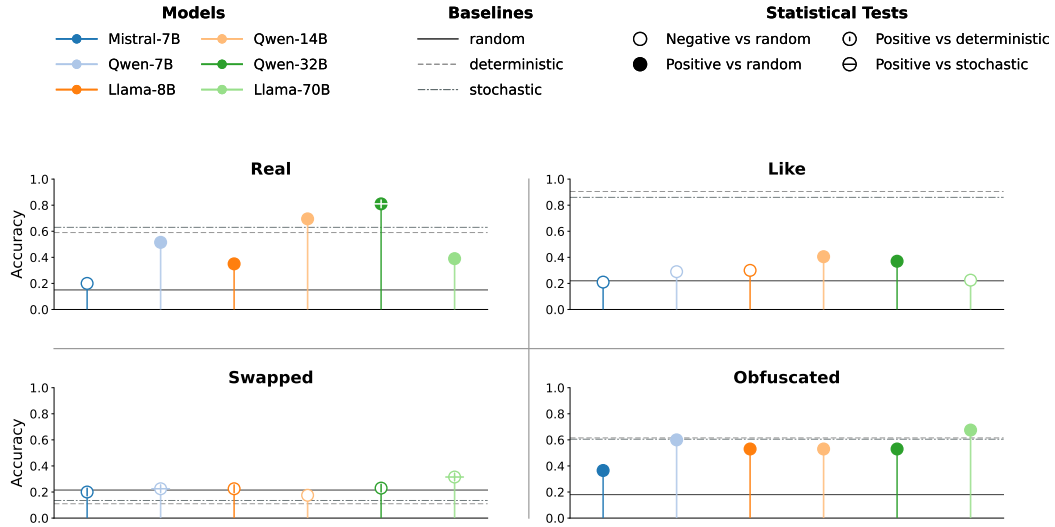


(a) Completion of mushroom

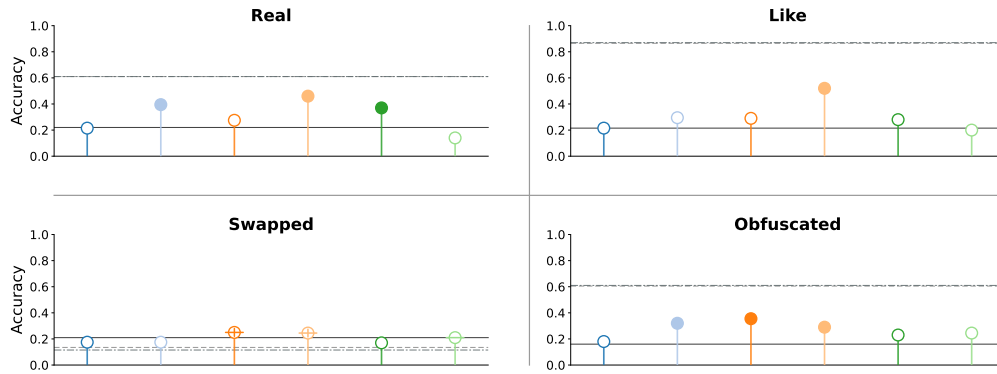


(b) Existence of mushroom

Figure 6: **Task performance for mushroom across dataset variants and models.** We represent baselines as horizontal lines, and statistical test results using filled dots for detected contamination (positive against random), and horizontal/vertical bars on dots for superiority against statistical baselines.



(a) Completion of credit



(b) Existence of credit

Figure 7: Task performance for credit across dataset variants and models. We represent baselines as horizontal lines, and statistical test results using filled dots for detected contamination (positive against random), and horizontal/vertical bars on dots for superiority against statistical baselines.

D.1 Ablation Study

The goal of this section is to analyze how the outcomes of our framework change as two main hyperparameters vary: the LLM’s sampling temperature and the percentage of columns masked or perturbed during probe construction, which we refer to as *coverage*.

D.1.1 Impact of Temperature

We first study the effect of the decoding temperature. As shown in the right panels of Figure 8, changing the temperature has little effect on contamination detection on the real dataset variant. In most model–dataset pairs, the test outcome remains unchanged when moving from deterministic decoding to a more stochastic setting. This indicates that the observed results are not driven by sampling-induced response variability and suggests that the framework is robust to this factor.

The same pattern is confirmed by the aggregate counts reported in the right panels of Figure 10, which summarize transitions across all dataset variants for each model. Nearly all models exhibit only a small number of outcome changes, and in some cases, such as *Llama-8B*, the effect is negligible. Overall, these results show that temperature is not a critical factor in the framework’s behavior.

D.1.2 Impact of Coverage

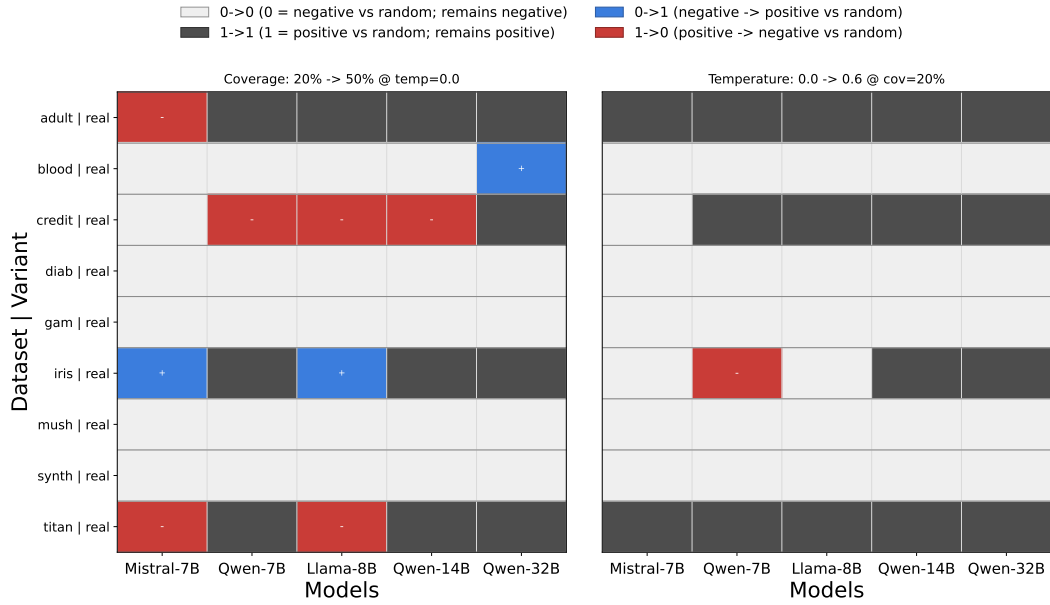
We next analyze the effect of coverage, i.e., the fraction of columns that are masked or perturbed in the generated probes. Unlike temperature, coverage directly affects the task’s difficulty and, therefore, performance on the real variant.

For the completion task, the left panels of Figures 8a and 9a show that most transitions occur from contamination detected to no contamination detected as coverage increases. This behavior is expected: masking a larger fraction of columns makes the completion problem harder, reducing the information available to recover the correct answer.

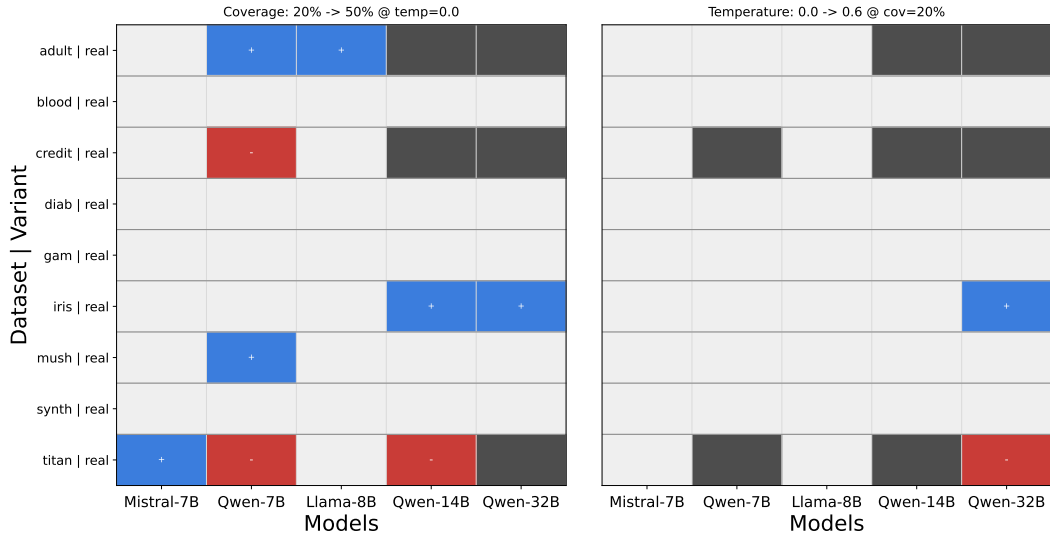
For the existence task, the trend is different. As shown in the left panels of Figures 8b and 9b, increasing coverage often makes the task easier rather than harder, with several transitions in the opposite direction. Adding more information to the distractor options makes the true row easier to identify. This suggests that LLMs can exploit prior knowledge effectively in this setting and, somewhat surprisingly, may benefit from stronger perturbations applied to the negatives.

D.1.3 Summary

Overall, this ablation study shows that temperature has a limited influence on the conclusions of our framework, whereas coverage plays a substantial role, and a more principled treatment of this hyperparameter would further strengthen the framework. One direction is to run probes at multiple coverage levels and aggregate the resulting outcomes, so that the contamination verdict depends on a model’s behavior across the entire coverage range rather than on a single, fixed choice. We leave this multi-coverage analysis, along with the design of perturbation strategies that reduce the sensitivity of performance to coverage, to future work.

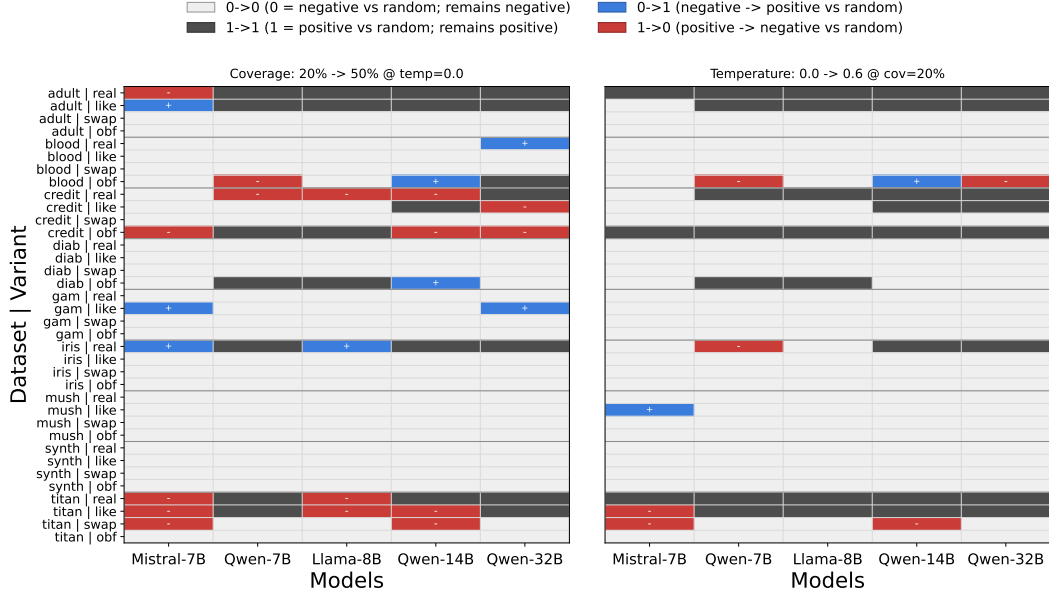


(a) Completion

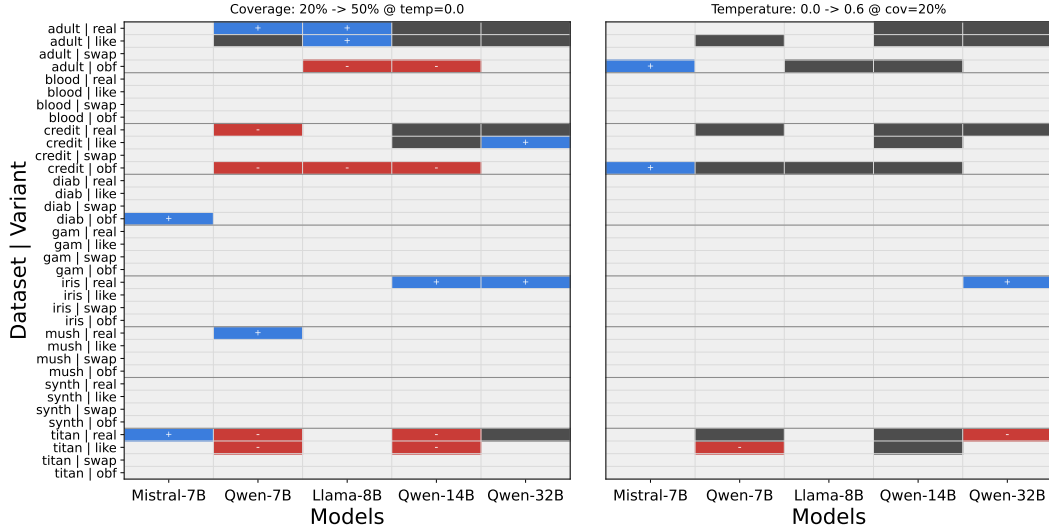


(b) Existence

Figure 8: **Ablation results on the real dataset variant for the completion and existence task.** The figure shows transitions in contamination-test outcomes for each model–dataset pair when increasing coverage from 20% to 50% at fixed temperature $t = 0$ (left), or temperature from 0 to 0.6 at 20% coverage (right) Colors denote stable outcomes and positive/negative flips, as reported in the legend.

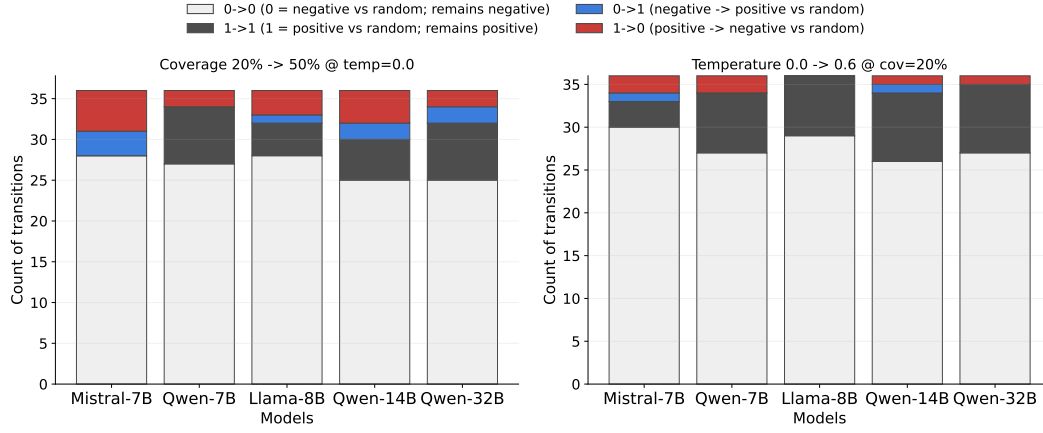


(a) Completion

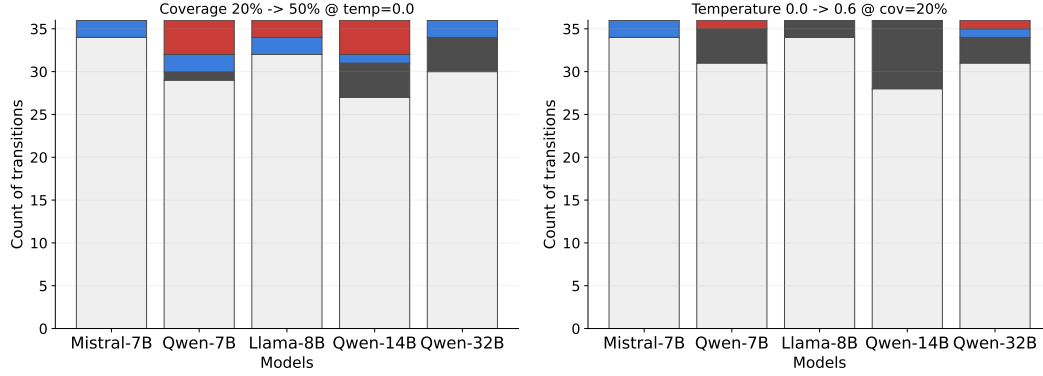


(b) Existence

Figure 9: **Ablation results on dataset variants for the completion and existence task.** The figure shows transitions in contamination-test outcomes for each model–dataset pair when increasing coverage from 20% to 50% at fixed temperature $t = 0$ (left), or temperature from 0 to 0.6 at 20% coverage (right). Colors denote stable outcomes and positive/negative flips, as reported in the legend.



(a) Completion



(b) Existence

Figure 10: **Aggregate transition counts across all dataset variants for the completion and existence task.** For each model, the bars report how many contamination-test outcomes remain unchanged or switch sign when increasing coverage from 20% to 50% at fixed temperature $t = 0$ (left), or temperature from 0 to 0.6 at 20% coverage (right). Colors follow the transition categories defined in the legend.

D.2 Detailed Contamination Results

In this section, we report the complete raw results across all models and datasets used throughout the paper, including those summarized in the main text and discussed in the previous section. Specifically, Table 7 presents results with temperature = 0 and coverage = 20%, Table 8 reports results with temperature = 0.6 and coverage = 20%, and Table 9 shows results with temperature = 0 and coverage = 50%.

Table 7: **Summary of contamination results across datasets and models (temperature = 0.0, coverage = 20%).** $\mathcal{A}_{\text{comp}}$ denotes completion accuracy, and $\mathcal{A}_{\text{exist}}$ denotes existence accuracy. Bold values indicate performances that are statistically significant above the random baseline and have higher accuracy than the random baseline ($\alpha = 0.01$); * indicates significance against the deterministic baseline, the stochastic baseline, or both ($\alpha = 0.01$).

Dataset	Variant	Metric	Model					
			Mistral-7B	Qwen-7B	Llama-8B	Qwen-14B	Qwen-32B	Llama-70B
adult	real	$\mathcal{A}_{\text{comp}}$	0.44	0.48	0.48	0.65	0.71	0.72
		$\mathcal{A}_{\text{exist}}$	0.27	0.30	0.25	0.42	0.40	0.52
	like	$\mathcal{A}_{\text{comp}}$	0.32	0.41	0.39	0.55	0.55	0.57
		$\mathcal{A}_{\text{exist}}$	0.23	0.46	0.32	0.42	0.46	0.49
	swap	$\mathcal{A}_{\text{comp}}$	0.15	0.16	0.18	0.20	0.20	0.23
		$\mathcal{A}_{\text{exist}}$	0.09	0.15	0.12	0.12	0.14	0.15
	obf	$\mathcal{A}_{\text{comp}}$	0.19	0.25	0.21	0.13	0.12	0.19
		$\mathcal{A}_{\text{exist}}$	0.19	0.19	0.26	0.30	0.18	0.24
	real	$\mathcal{A}_{\text{comp}}$	0.27	0.21	0.17	0.24	0.25	0.27
		$\mathcal{A}_{\text{exist}}$	0.18	0.21	0.20	0.28	0.28	0.28
	like	$\mathcal{A}_{\text{comp}}$	0.23	0.22	0.20	0.21	0.21	0.20
		$\mathcal{A}_{\text{exist}}$	0.18	0.23	0.23	0.21	0.23	0.26
blood	swap	$\mathcal{A}_{\text{comp}}$	0.20	0.20*	0.28*	0.17	0.23*	0.23*
		$\mathcal{A}_{\text{exist}}$	0.15	0.23*	0.26*	0.19*	0.16	0.20*
	obf	$\mathcal{A}_{\text{comp}}$	0.23	0.30	0.23	0.25	0.28	0.31
		$\mathcal{A}_{\text{exist}}$	0.21	0.23	0.27	0.22	0.17	0.19
	real	$\mathcal{A}_{\text{comp}}$	0.20	0.52	0.35	0.69	0.81*	0.39
		$\mathcal{A}_{\text{exist}}$	0.21	0.40	0.28	0.46	0.37	0.14
	like	$\mathcal{A}_{\text{comp}}$	0.21	0.29	0.30	0.41	0.37	0.23
		$\mathcal{A}_{\text{exist}}$	0.21	0.29	0.29	0.52	0.28	0.20
	swap	$\mathcal{A}_{\text{comp}}$	0.20*	0.23*	0.23*	0.17	0.23*	0.32*
		$\mathcal{A}_{\text{exist}}$	0.17	0.17	0.25*	0.24*	0.17	0.21*
	obf	$\mathcal{A}_{\text{comp}}$	0.36	0.60	0.53	0.53	0.53	0.68
		$\mathcal{A}_{\text{exist}}$	0.18	0.32	0.35	0.29	0.23	0.24
credit	real	$\mathcal{A}_{\text{comp}}$	0.21	0.17	0.17	0.21	0.23	0.18
		$\mathcal{A}_{\text{exist}}$	0.20	0.19	0.16	0.28	0.24	0.17
	like	$\mathcal{A}_{\text{comp}}$	0.17	0.19	0.15	0.23	0.20	0.18
		$\mathcal{A}_{\text{exist}}$	0.30	0.15	0.15	0.21	0.20	0.18
	swap	$\mathcal{A}_{\text{comp}}$	0.15	0.18	0.20	0.15	0.14	0.14
		$\mathcal{A}_{\text{exist}}$	0.11	0.21	0.14	0.12	0.14	0.17
	obf	$\mathcal{A}_{\text{comp}}$	0.19	0.41	0.36	0.29	0.22	0.42
		$\mathcal{A}_{\text{exist}}$	0.23	0.22	0.24	0.21	0.20	0.18
	real	$\mathcal{A}_{\text{comp}}$	0.21	0.13	0.17	0.20	0.14	0.20
		$\mathcal{A}_{\text{exist}}$	0.17	0.17	0.17	0.14	0.11	0.18
	like	$\mathcal{A}_{\text{comp}}$	0.20	0.20	0.20	0.19	0.17	0.17
		$\mathcal{A}_{\text{exist}}$	0.17	0.23	0.19	0.18	0.20	0.23
gamma	swap	$\mathcal{A}_{\text{comp}}$	0.20	0.14	0.23	0.17	0.17	0.22

Continued on next page

Table 7: Summary of contamination results across datasets and models (continued)

Dataset	Variant	Metric	Model					
			Mistral-7B	Qwen-7B	Llama-8B	Qwen-14B	Qwen-32B	Llama-70B
iris	obf	$\mathcal{A}_{\text{exist}}$	0.14	0.15	0.18	0.20	0.15	0.20
		$\mathcal{A}_{\text{comp}}$	0.21	0.17	0.21	0.14	0.18	0.23*
		$\mathcal{A}_{\text{exist}}$	0.15	0.14	0.15	0.15	0.14	0.15
	real	$\mathcal{A}_{\text{comp}}$	0.15	0.40*	0.35*	0.47*	0.47*	0.53*
		$\mathcal{A}_{\text{exist}}$	0.21	0.24	0.28	0.35	0.36	0.36
	like	$\mathcal{A}_{\text{comp}}$	0.24	0.33	0.31	0.19	0.32	0.25
		$\mathcal{A}_{\text{exist}}$	0.20	0.31	0.15	0.20	0.29	0.20
	swap	$\mathcal{A}_{\text{comp}}$	0.17	0.16	0.08	0.15	0.12	0.12
		$\mathcal{A}_{\text{exist}}$	0.19	0.15	0.19	0.25	0.12	0.20
	obf	$\mathcal{A}_{\text{comp}}$	0.23	0.24	0.24	0.19	0.20	0.17
		$\mathcal{A}_{\text{exist}}$	0.17	0.19	0.11	0.28	0.16	0.21
	mushroom	real	$\mathcal{A}_{\text{comp}}$	0.15	0.23	0.21	0.25	0.20
$\mathcal{A}_{\text{exist}}$			0.15	0.16	0.20	0.17	0.17	0.14
like		$\mathcal{A}_{\text{comp}}$	0.20	0.21	0.17	0.12	0.17	0.12
		$\mathcal{A}_{\text{exist}}$	0.18	0.14	0.19	0.14	0.11	0.18
swap		$\mathcal{A}_{\text{comp}}$	0.14	0.12	0.10	0.09	0.14	0.12
		$\mathcal{A}_{\text{exist}}$	0.13	0.10	0.12	0.07	0.10	0.09
obf		$\mathcal{A}_{\text{comp}}$	0.20	0.14	0.14	0.12	0.17	0.12
		$\mathcal{A}_{\text{exist}}$	0.17	0.18	0.17	0.14	0.15	0.15
synthetic	real	$\mathcal{A}_{\text{comp}}$	0.15	0.17	0.15	0.14	0.10	0.18
		$\mathcal{A}_{\text{exist}}$	0.19	0.16	0.14	0.15	0.14	0.19
	like	$\mathcal{A}_{\text{comp}}$	0.23	0.24	0.20	0.18	0.20	0.20
		$\mathcal{A}_{\text{exist}}$	0.19	0.21	0.21	0.17	0.14	0.18
	swap	$\mathcal{A}_{\text{comp}}$	0.17	0.17	0.15	0.09	0.10	0.17
		$\mathcal{A}_{\text{exist}}$	0.18	0.14	0.18	0.20	0.17	0.20
	obf	$\mathcal{A}_{\text{comp}}$	0.14	0.14	0.09	0.10	0.10	0.13
		$\mathcal{A}_{\text{exist}}$	0.17	0.16	0.23	0.18	0.20	0.20
titanic	real	$\mathcal{A}_{\text{comp}}$	0.32	0.47*	0.43	0.54*	0.50*	0.57*
		$\mathcal{A}_{\text{exist}}$	0.17	0.34	0.20	0.39*	0.31	0.42*
	like	$\mathcal{A}_{\text{comp}}$	0.33	0.40	0.32	0.47*	0.40	0.54*
		$\mathcal{A}_{\text{exist}}$	0.19	0.32	0.18	0.36	0.24	0.41
	swap	$\mathcal{A}_{\text{comp}}$	0.33*	0.19	0.26*	0.29*	0.18	0.12
		$\mathcal{A}_{\text{exist}}$	0.19	0.23	0.10	0.25	0.13	0.17
	obf	$\mathcal{A}_{\text{comp}}$	0.20	0.21	0.27	0.30	0.25	0.28
		$\mathcal{A}_{\text{exist}}$	0.26	0.23	0.19	0.29	0.27	0.29

Table 8: **Summary of contamination results across datasets and models (temperature = 0.6, coverage = 20%).** $\mathcal{A}_{\text{comp}}$ denotes completion accuracy, and $\mathcal{A}_{\text{exist}}$ denotes existence accuracy. Bold values indicate performances that are statistically significant above the random baseline and have higher accuracy than the random baseline ($\alpha = 0.01$); * indicates significance against the deterministic baseline, the stochastic baseline, or both ($\alpha = 0.01$).

Dataset	Variant	Metric	Model				
			Mistral-7B	Qwen-7B	Llama-8B	Qwen-14B	Qwen-32B
adult	real	$\mathcal{A}_{\text{comp}}$	0.35	0.49	0.54	0.65	0.69
		$\mathcal{A}_{\text{exist}}$	0.31	0.32	0.23	0.44	0.43
	like	$\mathcal{A}_{\text{comp}}$	0.33	0.46	0.37	0.49	0.56
		$\mathcal{A}_{\text{exist}}$	0.30	0.45	0.28	0.46	0.46
	swap	$\mathcal{A}_{\text{comp}}$	0.17	0.17	0.16	0.15	0.20
		$\mathcal{A}_{\text{exist}}$	0.18	0.17	0.18	0.10	0.10
	obf	$\mathcal{A}_{\text{comp}}$	0.20	0.24	0.24	0.14	0.09
		$\mathcal{A}_{\text{exist}}$	0.28	0.22	0.29	0.24	0.17
blood	real	$\mathcal{A}_{\text{comp}}$	0.21	0.18	0.19	0.21	0.28
		$\mathcal{A}_{\text{exist}}$	0.21	0.28	0.16	0.28	0.23
	like	$\mathcal{A}_{\text{comp}}$	0.26	0.22	0.24	0.20	0.23
		$\mathcal{A}_{\text{exist}}$	0.22	0.20	0.25	0.27	0.24
	swap	$\mathcal{A}_{\text{comp}}$	0.17	0.20*	0.17	0.23*	0.21*
		$\mathcal{A}_{\text{exist}}$	0.14	0.23*	0.21*	0.17*	0.20*
	obf	$\mathcal{A}_{\text{comp}}$	0.20	0.25	0.18	0.28	0.23
		$\mathcal{A}_{\text{exist}}$	0.17	0.19	0.18	0.19	0.14
credit	real	$\mathcal{A}_{\text{comp}}$	0.24	0.53	0.38	0.66	0.76*
		$\mathcal{A}_{\text{exist}}$	0.22	0.39	0.23	0.46	0.39
	like	$\mathcal{A}_{\text{comp}}$	0.23	0.32	0.23	0.41	0.37
		$\mathcal{A}_{\text{exist}}$	0.28	0.28	0.27	0.45	0.27
	swap	$\mathcal{A}_{\text{comp}}$	0.21*	0.20	0.18	0.15	0.19
		$\mathcal{A}_{\text{exist}}$	0.18	0.20	0.19	0.23*	0.19
	obf	$\mathcal{A}_{\text{comp}}$	0.35	0.66	0.50	0.56	0.57
		$\mathcal{A}_{\text{exist}}$	0.30	0.34	0.31	0.30	0.26
diabetes	real	$\mathcal{A}_{\text{comp}}$	0.19	0.21	0.26	0.19	0.23
		$\mathcal{A}_{\text{exist}}$	0.26	0.18	0.20	0.24	0.27
	like	$\mathcal{A}_{\text{comp}}$	0.20	0.18	0.23	0.24	0.17
		$\mathcal{A}_{\text{exist}}$	0.31	0.14	0.17	0.23	0.21
	swap	$\mathcal{A}_{\text{comp}}$	0.18	0.23*	0.15	0.17	0.12
		$\mathcal{A}_{\text{exist}}$	0.12	0.16	0.17	0.15	0.14
	obf	$\mathcal{A}_{\text{comp}}$	0.22	0.41	0.39	0.27	0.21
		$\mathcal{A}_{\text{exist}}$	0.23	0.24	0.23	0.22	0.20
gamma	real	$\mathcal{A}_{\text{comp}}$	0.18	0.19	0.24*	0.17	0.17
		$\mathcal{A}_{\text{exist}}$	0.18	0.20*	0.19	0.11	0.14
	like	$\mathcal{A}_{\text{comp}}$	0.19	0.16	0.20	0.19	0.20
		$\mathcal{A}_{\text{exist}}$	0.18	0.20	0.18	0.17	0.19
	swap	$\mathcal{A}_{\text{comp}}$	0.18	0.14	0.21	0.15	0.17
		$\mathcal{A}_{\text{exist}}$	0.17	0.19	0.20	0.23	0.17
	obf	$\mathcal{A}_{\text{comp}}$	0.20	0.20	0.18	0.15	0.21
		$\mathcal{A}_{\text{exist}}$	0.19	0.14	0.18	0.16	0.19
iris	real	$\mathcal{A}_{\text{comp}}$	0.28	0.32	0.33*	0.43*	0.56*
		$\mathcal{A}_{\text{exist}}$	0.20	0.21	0.23	0.33	0.47*
	like	$\mathcal{A}_{\text{comp}}$	0.20	0.23	0.24	0.24	0.28
		$\mathcal{A}_{\text{exist}}$	0.28	0.19	0.20	0.21	0.20

Continued on next page

Table 8: Summary of contamination results across datasets and models (continued)

Dataset	Variant	Metric	Model				
			Mistral-7B	Qwen-7B	Llama-8B	Qwen-14B	Qwen-32B
mushroom	swap	$\mathcal{A}_{\text{comp}}$	0.15	0.20	0.11	0.15	0.11
		$\mathcal{A}_{\text{exist}}$	0.23	0.24	0.12	0.28	0.23
	obf	$\mathcal{A}_{\text{comp}}$	0.17	0.20	0.27	0.20	0.12
		$\mathcal{A}_{\text{exist}}$	0.27	0.23	0.17	0.29	0.29
	real	$\mathcal{A}_{\text{comp}}$	0.18	0.18	0.17	0.17	0.21
		$\mathcal{A}_{\text{exist}}$	0.20	0.14	0.17	0.17	0.12
	like	$\mathcal{A}_{\text{comp}}$	0.26	0.23	0.20	0.12	0.19
		$\mathcal{A}_{\text{exist}}$	0.20	0.16	0.17	0.13	0.10
synthetic	swap	$\mathcal{A}_{\text{comp}}$	0.14	0.14	0.12	0.04	0.10
		$\mathcal{A}_{\text{exist}}$	0.14	0.15	0.10	0.08	0.12
	obf	$\mathcal{A}_{\text{comp}}$	0.18	0.10	0.15	0.13	0.12
		$\mathcal{A}_{\text{exist}}$	0.15	0.18	0.17	0.20	0.15
	real	$\mathcal{A}_{\text{comp}}$	0.15	0.15	0.17	0.11	0.14
		$\mathcal{A}_{\text{exist}}$	0.17	0.17	0.17	0.15	0.15
	like	$\mathcal{A}_{\text{comp}}$	0.16	0.21	0.23	0.20	0.20
		$\mathcal{A}_{\text{exist}}$	0.24	0.17	0.21	0.20	0.14
titanic	swap	$\mathcal{A}_{\text{comp}}$	0.17	0.13	0.20	0.10	0.14
		$\mathcal{A}_{\text{exist}}$	0.19	0.17	0.17	0.20	0.17
	obf	$\mathcal{A}_{\text{comp}}$	0.10	0.16	0.18	0.12	0.14
		$\mathcal{A}_{\text{exist}}$	0.20	0.17	0.25	0.17	0.15
	real	$\mathcal{A}_{\text{comp}}$	0.29	0.47*	0.38	0.49*	0.43*
		$\mathcal{A}_{\text{exist}}$	0.20	0.30	0.21	0.40*	0.26
	like	$\mathcal{A}_{\text{comp}}$	0.28	0.40	0.32	0.41	0.43
		$\mathcal{A}_{\text{exist}}$	0.25	0.28	0.20	0.35	0.28
titanic	swap	$\mathcal{A}_{\text{comp}}$	0.23	0.20	0.20	0.24	0.18
		$\mathcal{A}_{\text{exist}}$	0.22	0.28	0.14	0.28	0.19
	obf	$\mathcal{A}_{\text{comp}}$	0.24	0.21	0.26	0.24	0.26
		$\mathcal{A}_{\text{exist}}$	0.28	0.23	0.17	0.28	0.22

Table 9: **Summary of contamination results across datasets and models (temperature = 0.0, coverage = 50%).** $\mathcal{A}_{\text{comp}}$ denotes completion accuracy, and $\mathcal{A}_{\text{exist}}$ denotes existence accuracy. Bold values indicate performances that are statistically significant above the random baseline and have higher accuracy than the random baseline ($\alpha = 0.01$); * indicates significance against the deterministic baseline, the stochastic baseline, or both ($\alpha = 0.01$).

Dataset	Variant	Metric	Model				
			Mistral-7B	Qwen-7B	Llama-8B	Qwen-14B	Qwen-32B
adult	real	$\mathcal{A}_{\text{comp}}$	0.28	0.55	0.47	0.76	0.79
		$\mathcal{A}_{\text{exist}}$	0.23	0.36	0.39	0.45	0.49
	like	$\mathcal{A}_{\text{comp}}$	0.36	0.51	0.47	0.65	0.65
		$\mathcal{A}_{\text{exist}}$	0.24	0.54	0.34	0.53	0.52
	swap	$\mathcal{A}_{\text{comp}}$	0.15	0.10	0.09	0.19	0.12
		$\mathcal{A}_{\text{exist}}$	0.14	0.16	0.23	0.09	0.15
	obf	$\mathcal{A}_{\text{comp}}$	0.21	0.21	0.21	0.07	0.07
		$\mathcal{A}_{\text{exist}}$	0.17	0.17	0.21	0.21	0.23
	real	$\mathcal{A}_{\text{comp}}$	0.17	0.24	0.20	0.28	0.38
		$\mathcal{A}_{\text{exist}}$	0.20	0.22	0.20	0.35	0.26
blood	like	$\mathcal{A}_{\text{comp}}$	0.23	0.25	0.26	0.27	0.28
		$\mathcal{A}_{\text{exist}}$	0.23	0.20	0.17	0.34	0.24
	swap	$\mathcal{A}_{\text{comp}}$	0.19*	0.20*	0.23*	0.14	0.17
		$\mathcal{A}_{\text{exist}}$	0.15*	0.18*	0.22*	0.09	0.11
	obf	$\mathcal{A}_{\text{comp}}$	0.25	0.24	0.23	0.28	0.33
		$\mathcal{A}_{\text{exist}}$	0.26	0.19	0.26	0.28	0.23
	real	$\mathcal{A}_{\text{comp}}$	0.19	0.20	0.14	0.20	0.28
		$\mathcal{A}_{\text{exist}}$	0.14	0.21	0.21	0.35	0.34
	like	$\mathcal{A}_{\text{comp}}$	0.23	0.19	0.17	0.41	0.28
		$\mathcal{A}_{\text{exist}}$	0.22	0.32	0.33	0.56	0.51
credit	swap	$\mathcal{A}_{\text{comp}}$	0.24*	0.20*	0.28*	0.23*	0.24*
		$\mathcal{A}_{\text{exist}}$	0.21*	0.27*	0.23*	0.22*	0.18*
	obf	$\mathcal{A}_{\text{comp}}$	0.21	0.46	0.49	0.26	0.30
		$\mathcal{A}_{\text{exist}}$	0.16	0.27	0.26	0.24	0.18
	real	$\mathcal{A}_{\text{comp}}$	0.23	0.17	0.23	0.20	0.28
		$\mathcal{A}_{\text{exist}}$	0.21	0.17	0.15	0.24	0.24
	like	$\mathcal{A}_{\text{comp}}$	0.21	0.18	0.18	0.21	0.24
		$\mathcal{A}_{\text{exist}}$	0.26	0.23	0.11	0.30	0.16
	swap	$\mathcal{A}_{\text{comp}}$	0.17	0.23*	0.23*	0.17	0.12
		$\mathcal{A}_{\text{exist}}$	0.16	0.15	0.18	0.16	0.12
diabetes	obf	$\mathcal{A}_{\text{comp}}$	0.23	0.36	0.33	0.30	0.25
		$\mathcal{A}_{\text{exist}}$	0.31	0.21	0.27	0.23	0.17
	real	$\mathcal{A}_{\text{comp}}$	0.17	0.17*	0.14	0.18*	0.19*
		$\mathcal{A}_{\text{exist}}$	0.15	0.14	0.18	0.14	0.13
	like	$\mathcal{A}_{\text{comp}}$	0.30	0.20	0.19	0.21	0.27
		$\mathcal{A}_{\text{exist}}$	0.23	0.18	0.21	0.18	0.14
	swap	$\mathcal{A}_{\text{comp}}$	0.15	0.14	0.23	0.20	0.10
		$\mathcal{A}_{\text{exist}}$	0.15	0.19	0.21	0.18	0.16
	obf	$\mathcal{A}_{\text{comp}}$	0.21*	0.10	0.20	0.17	0.20
		$\mathcal{A}_{\text{exist}}$	0.17	0.18*	0.18*	0.17*	0.15
gamma	real	$\mathcal{A}_{\text{comp}}$	0.37	0.39	0.37	0.61*	0.69*
		$\mathcal{A}_{\text{exist}}$	0.27	0.36	0.29	0.48*	0.51*
	like	$\mathcal{A}_{\text{comp}}$	0.19	0.24	0.20	0.16	0.20
		$\mathcal{A}_{\text{exist}}$	0.09	0.15	0.25	0.15	0.17

Continued on next page

Table 9: Summary of contamination results across datasets and models (continued)

Dataset	Variant	Metric	Model				
			Mistral-7B	Qwen-7B	Llama-8B	Qwen-14B	Qwen-32B
mushroom	swap	$\mathcal{A}_{\text{comp}}$	0.13	0.21	0.20	0.21	0.24
		$\mathcal{A}_{\text{exist}}$	0.16	0.25	0.12	0.16	0.23
	obf	$\mathcal{A}_{\text{comp}}$	0.19	0.13	0.19	0.16	0.28
		$\mathcal{A}_{\text{exist}}$	0.16	0.20	0.15	0.16	0.17
	real	$\mathcal{A}_{\text{comp}}$	0.08	0.18	0.13	0.12	0.12
		$\mathcal{A}_{\text{exist}}$	0.12	0.29	0.14	0.20	0.14
	like	$\mathcal{A}_{\text{comp}}$	0.20	0.17	0.12	0.10	0.10
		$\mathcal{A}_{\text{exist}}$	0.06	0.20	0.12	0.11	0.10
	swap	$\mathcal{A}_{\text{comp}}$	0.16*	0.12	0.09	0.04	0.07
		$\mathcal{A}_{\text{exist}}$	0.10	0.14	0.10	0.14	0.14
synthetic	obf	$\mathcal{A}_{\text{comp}}$	0.17	0.09	0.15	0.07	0.11
		$\mathcal{A}_{\text{exist}}$	0.13	0.16	0.14	0.16	0.18
	real	$\mathcal{A}_{\text{comp}}$	0.15	0.12	0.16	0.07	0.12
		$\mathcal{A}_{\text{exist}}$	0.12	0.15	0.17	0.19	0.17
	like	$\mathcal{A}_{\text{comp}}$	0.26	0.23	0.23	0.29	0.24
		$\mathcal{A}_{\text{exist}}$	0.21	0.23	0.20	0.23	0.18
	swap	$\mathcal{A}_{\text{comp}}$	0.17	0.10	0.16	0.13	0.13
		$\mathcal{A}_{\text{exist}}$	0.14	0.17	0.17	0.15	0.14
	obf	$\mathcal{A}_{\text{comp}}$	0.15	0.12	0.17	0.11	0.14
		$\mathcal{A}_{\text{exist}}$	0.18	0.20	0.22	0.19	0.20
titanic	real	$\mathcal{A}_{\text{comp}}$	0.28	0.43	0.28	0.50	0.60
		$\mathcal{A}_{\text{exist}}$	0.26	0.20	0.22	0.24	0.31
	like	$\mathcal{A}_{\text{comp}}$	0.26	0.35	0.28	0.33	0.40
		$\mathcal{A}_{\text{exist}}$	0.23	0.26	0.13	0.23	0.18
	swap	$\mathcal{A}_{\text{comp}}$	0.11	0.21*	0.22*	0.13*	0.12*
		$\mathcal{A}_{\text{exist}}$	0.05	0.06	0.10	0.08	0.04
	obf	$\mathcal{A}_{\text{comp}}$	0.14	0.20	0.20	0.17	0.07
		$\mathcal{A}_{\text{exist}}$	0.11	0.14	0.08	0.07	0.14