KNOWLEDGE DISTILLATION AS DECONTAMINATION? REVISITING THE "DATA LAUNDERING" CONCERN

Anonymous authors
Paper under double-blind review

ABSTRACT

Concerns have been raised that knowledge distillation may transfer test-set knowledge from a contaminated teacher to a clean student—a "data laundering" effect that potentially threatens evaluation integrity. In this paper, we assess the severity of this phenomenon. If these concerns regarding data laundering are minor, then distillation could be used to mitigate risks of direct data exposure. Across eight benchmarks, we find that substantial laundering is the exception rather than the rule: unlike the large performance gains from direct contamination, any accuracy inflation from laundering is consistently smaller and statistically insignificant in all but two cases. More broadly, using sample-level analysis, we find that the two phenomena are weakly correlated, suggesting that laundering is not simply a diluted form of contamination but a distinct effect that arises primarily when benchmarks exhibit large train–test distribution gaps. Motivated by this, we conduct controlled experiments that systematically enlarge the train–test distance on two benchmarks where laundering was initially negligible, and observe that laundering becomes more significant as the gap widens. Taken together, our results indicate that knowledge distillation, despite rare benchmark-specific residues, can be expected to function as an effective decontamination technique that largely mitigates test-data leakage.

1 INTRODUCTION

Proprietary models have been proven to, perhaps inadvertently, learn from leaked benchmark data, raising questions about the reliability of closed-source models (Magar & Schwartz, 2022; Balloccu et al., 2024). One particularly subtle form of contamination is **data laundering**, where test-set knowledge leaks to a student model via a contaminated teacher, compromising evaluation integrity (Mansurov et al., 2025). While prior work highlighted this phenomenon, the prevalence, magnitude, and mechanisms of laundering remain largely unexplored. To assess the critical risk of this phenomenon, we ask: is data laundering a pervasive threat that undermines current benchmarking practices? Our extensive experiments on classification tasks suggest that data laundering is often much weaker than direct contamination and can even mitigate some of its harmful effects. This work contributes a potential foundation for establishing **safer empirical research environments**.

First, we conduct a large-scale assessment across eight benchmarks to determine the prevalence and magnitude of data laundering. We find that significant laundering is a rare phenomenon: while it does occur, its effect on model accuracy is substantially smaller than that of direct contamination, and in many cases the difference is not statistically significant. This initial finding suggests that knowledge distillation may indeed function as an effective decontamination method.

Given that data laundering effects are substantially weaker than those of direct contamination, we investigate whether it is simply a watered-down form of direct contamination or a distinct phenomenon. Using sample-level analysis across the same benchmarks, we examine whether a sample's sensitivity to direct contamination predicts its sensitivity to laundering. Our analysis reveals only a weak correlation between the two; samples highly susceptible to direct contamination are not necessarily the ones most affected by laundering. This suggests that laundering is indeed a distinct phenomenon.

Finally, we explore the conditions under which data laundering emerges. Controlled experiments show that systematically widening the train-test distributional gap increases the effects of laundering, suggesting a causal connection. Consistent with these experiments, benchmarks with naturally larger

train-test gaps tend to exhibit stronger laundering effects, although the magnitude and statistical significance of the effect vary across datasets. Our findings indicate that both the characteristics of the benchmark and the degree of train-test shift influence the extent of data laundering, highlighting that its occurrence depends on benchmark-specific factors such as dataset domain and the degree of train–test distributional shift, rather than being a universal consequence of knowledge distillation.

In summary, this paper systematically disentangles the role of data laundering in model evaluation. We argue that while the concern is valid, its practical impact is often minimal. Our findings indicate that laundering is generally limited in scope, substantially smaller than direct contamination, and tied to particular conditions such as train–test distributional shifts. By contextualizing these risks and identifying the conditions under which they arise, we provide a pathway for more responsible and reliable model evaluation including a principled use of KD in the era of ubiquitous large models.

2 Related Work

Data contamination in evaluation. A growing number of work has shown that benchmark integrity can be compromised when test material leaks into pretraining or fine-tuning corpora, artificially inflating scores without corresponding generalization. Early red flags already appeared with large web-scale LMs such as GPT-3 (Brown et al., 2020) and in corpora audits like C4 (Dodge et al., 2021), while Magar & Schwartz (2022) provided a controlled, task-level analysis linking memorization to performance inflation. Subsequent studies proposed black-box and white-box detectors—e.g., guided-instruction "time travel" tests (Golchin & Surdeanu, 2024) and distributional peakedness checks (Dong et al., 2024)—and documented practical challenges for closed models (OpenAI, 2023). Broader surveys and empirical audits emphasize that overlap can be subtle (paraphrases, partial spans, synthetic rephrasings) and uneven across benchmarks, motivating routine, benchmark-specific contamination checks (Sainz et al., 2023) and calls for provenance transparency (e.g., to report train–test overlap) (Zhang et al., 2024). Recent work also targets modern LLM benchmarks directly, offering methods tailored to both open and proprietary models (Deng et al., 2024).

Knowledge distillation and data laundering. Knowledge distillation (KD) is a standard tool for compression and transfer (Hinton et al., 2015; Sanh et al., 2019), but it also opens a distinct vector for leakage. Mansurov et al. (2025) formally introduced *data laundering* showing that a contaminated teacher (exposed to test data) can pass benchmark-specific knowledge to a student trained only on clean data via KD, inflating evaluation without direct access to the test set. However, their study had limitations: the experiments relied on a bert-base-uncased student trimmed down to just 2 layers, rather than using a pretrained 2-layer model, making results difficult to disentangle from near-random baselines. Additionally, the study did not compare laundering against direct contamination or systematically explore when it arises. As a result, the prevalence, magnitude, and mechanisms of laundering remain unclear, motivating our more systematic analyses. Complementary evidence from ranking distillation shows that even tiny teacher exposure (e.g., $< 0.1\%$ of training) can yield inflated student effectiveness, especially with pairwise/order-based objectives (Suresh Kalal et al., 2024). Security-oriented studies further show that KD can transmit non-benign artifacts (e.g., backdoor behaviors), particularly in data-free settings (Hong et al., 2023). Together these results underscore procedural defenses such as transparent training histories and contamination-aware KD protocols (Zhang et al., 2024).

3 Methodology

3.1 Experimental Setup and Data

Models and Distillation Process To isolate and measure data laundering, we use a controlled, two-stage process. Our setup involves fine-tuning eight models in total for each benchmark. First, we train teacher models using (`bert-base-uncased`) (Devlin et al., 2019). A **clean teacher** (T_{clean}) is fine-tuned on the original training set, while a **dirty teacher** (T_{dirty}) is fine-tuned on a training set contaminated with test data.

Second, we use these teachers to distill knowledge into smaller student models (`distilbert-base-uncased`) (Sanh et al., 2019) using soft-label distillation with forward

KL divergence (Wang et al., 2024). Crucially, the distillation process for both student types (S_{clean} distilled from T_{clean} and S_{dirty} from T_{dirty}) is always performed using the original, clean training set. This ensures that any test-set knowledge is transferred exclusively via the teacher model, not through direct data exposure during the student's training.

To serve as a control group and contextualize the results, we also establish baseline models that share the same architecture as the students. A **Clean Baseline** (B_{clean}) and a **Dirty Baseline** (B_{dirty}) are created by fine-tuning the student architecture directly on the clean and contaminated datasets, respectively.

Benchmarks and Contamination Protocol We selected eight public classification benchmarks, with diverse domains ranging from topic classification, sentiment analysis, and intent recognition to emotion detection, and NLI (details in Appendix B.2, Table 4). This diversity ensures that our findings generalize across tasks with varying difficulty, number of labels, and dataset sizes. For each benchmark, we create a contaminated dataset to train B_{dirty} and T_{dirty} with a **replacement strategy**: we contaminate training data by injecting the full test set and removing an equal number of original training samples to keep size constant. All experiments are repeated with five different random seeds to ensure robustness.

3.2 Evaluation Metrics

We employ a set of metrics to quantify data leakage, distinguishing between those that operate on the entire benchmark and those that apply to individual samples. Let the test set be denoted by $\mathcal{C} = \{x_1, x_2, \ldots, x_n\}$. Metrics computed over $\mathcal{C}$ capture the overall impact of data leakage on model accuracy at the benchmark level, and sample-level metrics evaluate performance on individual examples x_i.

Sample-Level Leakage Scores To analyze leakage mechanisms at a finer granularity, we measure how much each test sample x_i becomes easier or harder under different training conditions. The *difficulty* of a sample x_i under a model M is defined as

$$D(x_i, M) = 1 - P(y_i \mid x_i; M),$$

that is, the probability of the model assigning the wrong label. Intuitively, higher $D(x_i, M)$ means the model finds x_i more difficult.

We then define two sample-level leakage effects:

$$\Delta_{\text{laund}}(x_i) = D(x_i, S_{\text{dirty}}) - D(x_i, S_{\text{clean}}),$$
$$\Delta_{\text{contam}}(x_i) = D(x_i, B_{\text{dirty}}) - D(x_i, B_{\text{clean}}).$$

Here Δ_{laund} captures how much a student model changes when trained on dirty vs. clean teachers, while Δ_{contam} captures how much a baseline model changes when directly trained on dirty vs. clean data. These sample-level scores provide a natural way to capture the effect of laundering and contamination on individual examples. In practice, we typically expect these Δ values to be negative, since the presence of laundering or contamination generally reduces the sample difficulty.

Laundering vs. Contamination Correlation To assess whether data laundering and direct contamination are mechanistically related, we compute a benchmark-level correlation from the sample-level scores. For a given test set $\mathcal{C}$, we first construct two vectors of leakage effects:

$$\mathbf{l} = [\Delta_{\text{laund}}(x_1), \ldots, \Delta_{\text{laund}}(x_n)], \quad \mathbf{c} = [\Delta_{\text{contam}}(x_1), \ldots, \Delta_{\text{contam}}(x_n)].$$

We then calculate the Pearson correlation coefficient

$$r(\mathcal{C}) = \frac{\text{cov}(\mathbf{l}, \mathbf{c})}{\sigma_{\mathbf{l}}\, \sigma_{\mathbf{c}}}. \tag{1}$$

This metric is advantageous as it is scale-invariant, allowing us to compare the directional agreement of the two phenomena across all samples, irrespective of the absolute magnitude of their effects.

4 RESULTS AND ANALYSIS

4.1 KNOWLEDGE DISTILLATION AS AN EFFECTIVE DECONTAMINATION TECHNIQUE

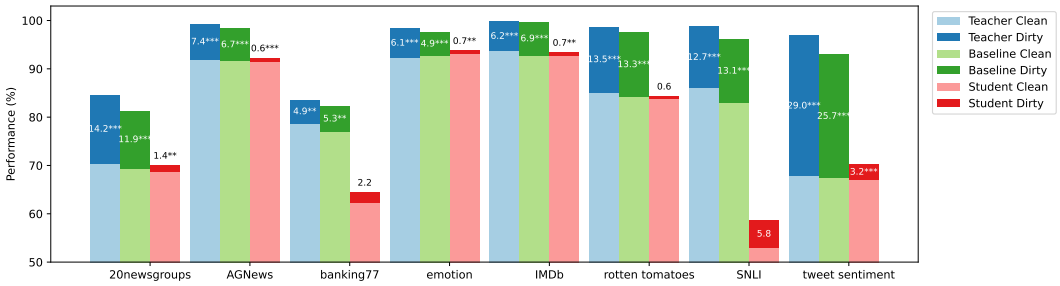


Figure 1: Performance on clean and contaminated benchmarks for Teacher, Baseline, and Student models. Bars show clean (lighter) and contaminated/dirty (darker) accuracies, with ΔAcc_M values annotated. The statistical significance of the differences are marked with *p<0.05, **p<0.005, ***p<0.001.

We establish an upper bound on potential data leakage by comparing the aggregate impact of contamination using benchmark-level metrics. Specifically, we contrast the gains from direct contamination in clean and dirty baseline (B) and teacher (T) models with the gains from data laundering in clean and dirty student (S) models. For a model M and benchmark $\mathcal{C}$, the performance gain due to test-set leakage at training time is defined as

$$\Delta\text{Acc}_M = \text{Acc}(M_{\text{dirty}}, \mathcal{C}) - \text{Acc}(M_{\text{clean}}, \mathcal{C}). \tag{2}$$

When $M = B$ or $M = T$, Equation (2) measures the direct contamination effects, whereas for $M = S$ it measures data laundering gains.

Figure 1 presents the performance gains over the baselines for all models (the full results are reported in Appendix C, Table 7). Comparing the gains of baseline and student models allows us to quantify the relative impact of data laundering on observed performance improvements, and contrasting the direct contamination gains of baseline and teacher models highlights the impact of model capacity.[1]

The results reveal a clear and consistent pattern: directly training on a test-set-contaminated dataset results in substantial and highly significant performance gains across all eight benchmarks for all Teacher and Baseline models. Performance gains for the Baseline models range from 4.89% on `emotion` to 25.66% on `tweet_sentiment`, confirming the well-documented effects of direct data contamination and providing a crucial reference point for evaluating the impact of knowledge distillation.

By contrast, the gains observed due to data laundering (ΔAcc_S) are noticeably smaller, highlighting the mediating effect of knowledge distillation. For example, on `tweet sentiment`, the Baseline's 25.66% gain is reduced to 3.25% after distillation through a contaminated Teacher. Similarly, on `20newsgroups`, an 11.91% direct gain shrinks to just 1.42%. This trend is consistent across all benchmarks: distillation acts as a strong bottleneck, significantly mitigating the performance inflation caused by direct contamination. Overall, these results suggest that knowledge distillation, rather than solely propagating leakage, may act as a *effective decontamination mechanism*, substantially mitigating the performance inflation caused by direct contamination.

The gains from data laundering are clearly more modest than those from direct contamination, with significant increases on datasets like `agnews` (0.65%) and `tweet_sentiment` (3.25%). However, for three benchmarks (`banking77`, `rotten tomatoes`, and `SNLI`) the difference in performance is not statistically significant. This detailed observation allows us to refine our perspective. While the concern about data laundering is not unfounded, as the phenomenon does occur, its practical impact

[1] An exception is `SNLI`, where the performance gap between the clean student and the clean baseline is unusually large. This result may be explained by two factors: (i) all models were trained for only 3 epochs without early stopping to ensure fair comparison with the teacher and baseline, which may have left the student undertrained when relying solely on teacher signals; and (ii) `SNLI` is inherently more challenging than typical classification tasks, making it harder for the student to achieve strong performance under distillation.

is the exception rather than the rule. It is a rare and mild effect, with a smaller magnitude and limited influence on overall model evaluation.

4.2 Laundering and Contamination: A Tale of Two Mechanisms

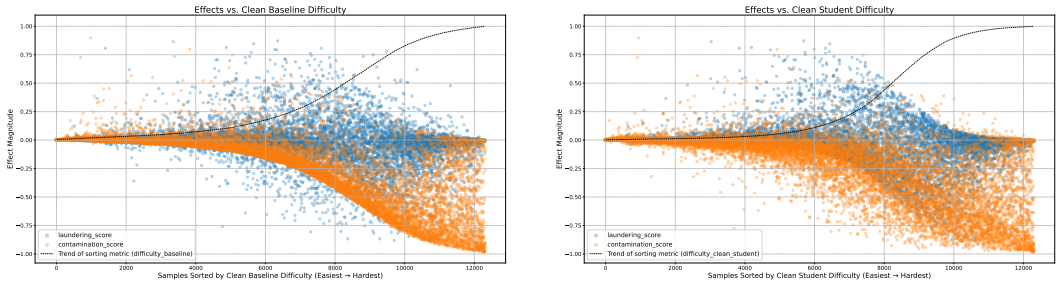
Having established that data laundering is a rare and much weaker effect than direct contamination, a crucial question arises regarding its nature: is laundering merely a weaker, scaled-down version of direct contamination, or is it a distinct phenomenon with its own underlying mechanism? If it were simply "contamination-lite," we would expect the samples most affected by both phenomena to be highly correlated.

$\mathcal{C}$	**20newsgroups**	**agnews**	**banking77**	**emotion**	**imdb**	**rotten tomatoes**	**snli**	**tweet sentiment**
$r(\mathcal{C})$	0.30(03)***	0.32(02)***	0.13(08)	0.26(12)***	0.30(02)***	0.12(06)*	-0.03(17)***	0.31(01)***

Table 1: Benchmark correlations between laundering and contamination effects. Shows baseline accuracy (B_{clean}) and Pearson correlations of sample-level effects. Statistical significance was assessed using bootstrapping-based tests, with detailed procedures provided in Appendix B.4.

In Table 1, we show the Pearson correlation between the sample-level laundering effect and contamination effect scores. We find that on the benchmarks most susceptible to laundering, `agnews` and `tweet_sentiment`, the correlations are as small as 0.32 and 0.31, respectively. While statistically significant, these values are far below the commonly accepted threshold of 0.7 for a strong relationship (see e.g., Rickert et al., 2023; Kjell et al., 2022)), indicating only a weak linear association. The connection is weaker still on other benchmarks, and even becomes slightly negative for `snli`.

This weak correlation suggests that the two phenomena impact samples differently. A sample that is highly vulnerable to being "memorized" through direct training is not necessarily the same sample whose knowledge is indirectly transferred through a teacher model. This observation motivates a deeper exploration into the nature of these mechanisms. To explore this further, we conduct a granular, sample-level analysis on the `tweet_sentiment` benchmark, where the laundering effect was most pronounced. Figure 2 visualizes the laundering and contamination effects for each test sample, sorted by their difficulty as perceived by both the clean baseline and clean student models.



(a) Samples sorted by clean baseline difficulty

(b) Samples sorted by clean student difficulty

Figure 2: Laundering and contamination effects on the `tweet_sentiment` benchmark, with samples sorted by difficulty. The contamination effect (orange) shows a strong, monotonic downward trend as samples become harder. In contrast, a laundering effect (blue) is more dispersed, has a much weaker trend, and is less correlated with initial sample difficulty.

By sorting samples by the difficulty perceived by the clean model, we can visualize that some samples are more inherently prone to contamination than others: those that the clean model finds more difficult have a larger potential for improvement when seen during training. As a result, the contamination effect (in orange) exhibits a strong and relatively monotonic downward trend—a more negative Δ_{contam}—on harder samples. This is an expected signature of direct test-set exposure. In stark contrast, the data laundering effect (in blue) does not share this strong monotonic relationship with sample difficulty. Its overall magnitude is smaller, its trend is weaker, and it exhibits significant volatility, with benefits appearing for both difficult and some easy samples.

Together, the weak sample-level correlations and non-monotonic laundering trends show laundering is not "scaled-down contamination" but a distinct mechanism. Instead, it is an independent mechanism triggered by different conditions. This discovery leads to the conclusion that data laundering, when it occurs, is a more elusive mechanism than direct contamination and likely possesses its own unique set of enabling conditions.

5 Discussion: The Role of Distributional Gaps

Our analysis in the previous section established that data laundering is a rare, mild, and mechanically distinct phenomenon from direct contamination. This prompts the next logical question: what specific characteristics of a benchmark make it more susceptible to this elusive effect? Our results revealed the impact of data laundering to be highly benchmark-specific, with datasets such as `tweet_sentiment` exhibiting comparatively more significant effects. This naturally leads us to hypothesize that some intrinsic property of these datasets may be at play.

	Jaccard		TF-IDF		Avg Emb Sim		Average Max Semantic Sim		Average Pattern Conformity	
Dataset	**Micro**	**Macro**	**Micro**	**Macro**	**Micro**	**Macro**	**Micro**	**Macro**	**Micro**	**Macro**
agnews	0.1686	0.1463	0.9881	0.9661	0.9984	0.9965	0.6682	0.6554	0.4942	0.5468
tweet_sentiment	**0.0887**	**0.0733**	**0.6310**	**0.6371**	**0.7115**	**0.7681**	**0.5102**	**0.4913**	**0.4437**	**0.4523**
20newsgroups	0.1787	0.1513	0.9848	0.8559	0.9921	0.9599	0.5985	0.5709	0.4871	0.5431
emotion	0.0941	0.0666	0.9820	0.9072	0.9984	0.9877	0.6403	0.5954	0.5504	0.5622
imdb	0.2218	0.2116	0.9978	0.9961	0.9994	0.9990	0.6919	0.6837	0.6669	0.6707
rotten_tomatoes	0.0702	0.0606	0.9198	0.8609	0.9987	0.9975	0.6331	0.6229	0.5920	0.5958
snli	0.1359	0.1552	0.9906	0.9877	0.9979	0.9975	0.7011	0.6668	0.5606	0.5605
banking77	0.2874	0.2217	0.9810	0.9103	0.9963	0.9887	0.8985	0.8940	0.7520	0.8627

Table 2: Similarity metrics between the test set and the training subsets. "Micro" refers to the global similarity, calculated across all samples without referencing labels. "Macro" refers to the unweighted average of similarities computed on a per-label basis. Benchmarks more vulnerable to data laundering, such as `tweet_sentiment`, happen to appear lower similarity scores.

To investigate this, we first characterize the intrinsic relationship between the training and test sets in a model-agnostic way. We compute a suite of similarity metrics, all normalized to the range $[0, 1]$, to help identify inherent data properties that might make a benchmark susceptible to leakage. The metrics used are: Jaccard Similarity, TF-IDF Cosine Similarity, Average Embedding Similarity, Average Max Semantic Similarity, and Average Pattern Conformity. The detailed mathematical formulations for these metrics are provided in Appendix B.5.

An analysis of these metrics across the benchmarks (detailed in Table 2) revealed a notable pattern. The benchmarks most affected by laundering—`tweet_sentiment`—consistently exhibit lower similarity scores across several metrics. This indicates a larger distributional gap between their training and testing sets. This observation provides us with a plausible hypothesis: **data laundering is more likely to occur, and its effects are more pronounced, when there is a significant distributional distance between a benchmark's training and test sets.**

The intuition behind this is that when a teacher model is contaminated with test data that deviates from the training data's dominant semantic patterns, it is exposed to alternative, test-specific regularities. These regularities different from those emphasized in the training distribution can then be systematically learned by the teacher. Crucially, such patterns may be particularly advantageous for the test set, and can subsequently be passed on to the student during distillation. Based on this hypothesis, we designed a series of controlled experiments to systematically validate this relationship.

5.1 Experimental Design

To test our hypothesis rigorously, we designed a controlled experimental setup to systematically vary the train-test distribution gap. This section outlines the creation of our stratified datasets and the experimental protocol.

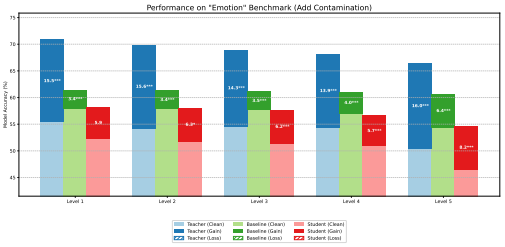
Creating Controlled Distributional Gaps The core of our methodology involves partitioning the training data of the `emotion` and `rotten_tomatoes` benchmarks into distinct subsets. This is achieved through a **stratified splitting** process: for each class, we first identify its test-set centroid, and then partition the training samples belonging to that class into five equal-sized quintiles based

on their semantic similarity to this centroid. These quintiles are then aggregated across all classes to yield five global training sets, denoted as Levels 1 through 5. **Level 1** contains samples most similar to the test set (smallest gap), while **Level 5** contains those least similar (largest gap). This method successfully creates the intended gradient of distributional gaps. The effectiveness of this partitioning is validated by a systematic decrease in cosine similarity from Level 1 to 5, as well as a consistent monotonic decrease across five other similarity metrics (see Appendix B.6 for detailed visualizations).

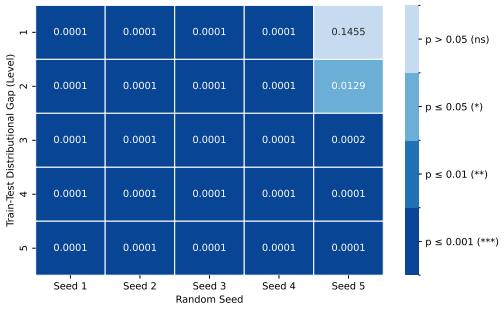
Training and Contamination Protocol For each of the five data levels, we conduct a consistent training and distillation protocol. We train a **clean teacher** (T_{clean}) on the original training quintile and a **dirty teacher** (T_{dirty}) on its contaminated counterpart. We use an `add` mode (as opposed to `replace` mode) for contamination in this setup; because each training level is significantly smaller than the original dataset, this approach ensures the model has sufficient data for robust training and mitigates the impact of the reduced training set size. Knowledge from both teachers is then distilled into respective student models, S_{clean} and S_{dirty}. The distillation process itself always uses the clean training data of that level, isolating the transferred knowledge as the primary variable.

5.2 Results: Laundering Effect Intensifies with Wider Distributional Gaps

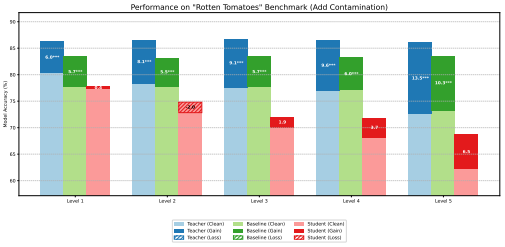
With the controlled gaps established and verified, we now turn to the central result of our experiment. As shown in Figure 3, our findings demonstrate a clear, positive relationship between the train-test distributional gap and the magnitude of the data laundering effect. The key insight lies in comparing the accuracy of the clean student (S_{clean}) and the dirty student (S_{dirty}) at each level, as we did with ΔAcc_S using Equation (2).



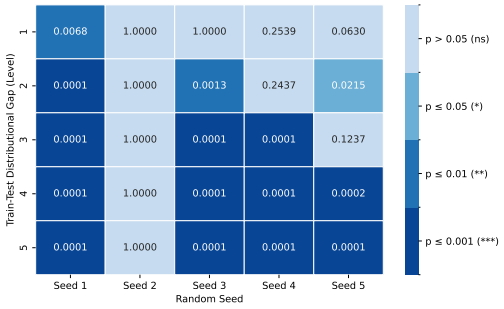
(a) `emotion` accuracies



(b) `emotion` significance heatmap



(c) `rotten_tomatoes` accuracies



(d) `rotten_tomatoes` significance heatmap

Figure 3: Data laundering effects across controlled distributional gaps for `emotion` and `rotten_tomatoes`. **Left panels:** model accuracy comparison, where the performance gain for the student model (red bars) represents the laundering effect. **Right panels:** heatmaps of p-values, verifying that the distributional gaps between train and test sets are statistically significant across levels and random seeds. Statistical significance for performance gains ($*p < 0.05, **p < 0.01, ***p < 0.001$) was assessed using bootstrapping-based tests.

Direct Contamination Effect Remains Stable First, we observe that for both the `emotion` and `rotten_tomatoes` datasets, a significant direct contamination effect is present from Level 1 to 5, evidenced by the performance differences between the dirty and clean teachers, as well as between the dirty and clean baselines. However, this performance gap does not show significant fluctuation or a clear trend as the distributional gap widens. This can be observed visually in Figure 3, where the gaps between clean and dirty models (for both teachers and baselines) remain largely stable across the levels. Our experiment thus indicates that, under this setup, the magnitude of the direct contamination effect is not significantly correlated with the distributional gap between the training and test sets.

Analysis of the `emotion` Dataset We then turn our attention to the data laundering effect. For the `emotion` dataset, the significance heatmap in Figure 3(b) reveals that the p-values generally decrease as we move from Level 1 to Level 5. This trend confirms that the data laundering effect becomes more statistically apparent as the distributional gap grows. It is important to note, however, that while the statistical certainty of the effect grows, the performance gap between the clean and dirty students does not systematically widen. This suggests that the primary effect observed here is an increase in statistical confidence rather than a systematic increase in the magnitude of the laundering effect itself.

Analysis of the `rotten_tomatoes` Dataset In contrast, the `rotten_tomatoes` benchmark exhibits a similar and more pronounced pattern. Following the trend seen in the `emotion` dataset, the data laundering effect becomes more statistically significant as the distributional gap increases. As depicted in the heatmap (Figure 3(d)), the p-value consistently decreases with higher levels for most random seeds, with the notable exception of the second seed. Moreover, the performance delta between the clean and dirty students also tends to widen as the level increases, suggesting that the magnitude of the laundering effect strengthens with a larger distributional gap. We therefore conclude that a clear relationship between data laundering and the distributional gap exists for this benchmark.

Summary and Implications In summary, our experimental results indicate that test data is less susceptible to laundering when the training data is distributionally close, whereas the laundering effect becomes more pronounced as the distributional gap widens. At the same time, we confirm that the effect of direct contamination is not significantly correlated with this gap. This finding offers potential insights for benchmark designers and researchers. On the one hand, a test set is expected maintain a certain distributional distance from the training set to properly evaluate a model's generalization capabilities. On the other hand, an excessively large distributional gap may increase the risk of data laundering, even if the overall effect remains weak.

Therefore, the challenge of striking a balance between a "too close" and a "too far" test set reveals a potential limitation in current benchmark design paradigms. This inherent tension suggests that relying on a single test set with a fixed distributional distance is insufficient. Rather than seeking a single "optimal" balance, a more robust approach may be to employ multiple test sets at varying distributional distances. This would enable a more comprehensive assessment, simultaneously evaluating a model's generalization power across different gaps and ensuring resilience to data laundering.

6 Conclusion

This paper set out to investigate the severity of **data laundering** and its implications for evaluation integrity. Our comprehensive investigation across eight benchmarks offers a reassuring but also nuanced, conclusion: the concerns of data laundering as a pervasive threat appear largely overstated. Instead, we find that **knowledge distillation generally functions as a promising decontamination technique**, dramatically attenuating the performance inflation caused by direct test-set exposure.

While distillation acts as a strong buffer, it is not a perfect one. We confirm that residual leakage can occur, but these instances of significant laundering are the exception, not the rule. Crucially, our analysis reveals that this leakage is not simply a diluted form of direct contamination but a **mechanically distinct phenomenon**. We identified the **train-test distributional gap** as a key driver, a hypothesis confirmed through controlled experiments where systematically widening this gap induced a significant laundering effect.

In summary, knowledge distillation, far from being a liability, is a robust defense against test-data leakage. The rare, benchmark-specific instances of laundering are not an indictment of the method itself but are a predictable consequence of large distributional shifts between training and test data—a factor that benchmark designers may need to consider. Our findings thus help clarify the risks associated with distillation and provide a path toward more reliable and responsible model evaluation.

7 Limitations and Future Work

While our study provides a comprehensive analysis, its scope has several limitations that also outline promising directions for future research.

Limitations Our work is primarily constrained by its focus on BERT-family encoder-only models, leaving the effects on larger-scale models and different architectures, such as decoder-only LMs, unexplored. Additionally, the analysis was confined to classification tasks; data laundering in generative contexts, where evaluation and distillation strategy can be more diverse and complex, remains an open question. Finally, our experiments used only English datasets, so the findings may not generalize to multilingual or domain-specific scenarios.

Future Work These limitations suggest several research avenues. Future work should extend this investigation to broader architectures and tasks, including developing appropriate metrics for generative models. It is also crucial to examine data laundering on multilingual and domain-specific benchmarks to test the generality of our findings and to further probe the mechanistic distinctions between laundering and direct contamination. Besides, a particularly promising direction involves leveraging KD for decontamination use, and even act as a diagnostic tool to infer the contamination level of teacher models.

References

Ibtihel Amara, Nazanin Sepahvand, Brett H. Meyer, Warren J. Gross, and James J. Clark. Bd-kd: Balancing the divergences for online knowledge distillation, 2024. URL https://arxiv.org/abs/2212.12965.

Simone Balloccu, Patrícia Schmidtová, Mateusz Lango, and Ondrej Dusek. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source LLMs. In Yvette Graham and Matthew Purver (eds.), *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 67–93, St. Julian's, Malta, March 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.eacl-long.5. URL https://aclanthology.org/2024.eacl-long.5/.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Xinyu Deng, Zhengyan Liu, Zihan Du, Jiaao Bai, Maosong Sun, Yuxuan He, and Zhiyuan Liu. Investigating data contamination in modern benchmarks for large language models. In *Proceedings of NAACL*. Association for Computational Linguistics, 2024. URL https://aclanthology.org/2024.naacl-long.482/.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Proceedings of EMNLP*, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.98/.

Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. *Findings of ACL*, pp. 12039–12050, 2024. URL https://aclanthology.org/2024.findings-acl.716.pdf.

Shahriar Golchin and Mihai Surdeanu. Time travel in LLMs: Tracing data contamination in large language models. In *Proceedings of ICLR*, 2024. URL https://openreview.net/forum?id=2Rwq6c3tvr.

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

Mingyuan Hong, Yicheng Li, Zijian Yang, Yisen Wang, Zhangyang Wang, et al. Revisiting data-free knowledge distillation with poisoned teachers. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, 2023. URL https://proceedings.mlr.press/v202/hong23c/hong23c.pdf.

Oscar N. E. Kjell, Sverker Sikström, Katarina Kjell, and H. Andrew Schwartz. Natural language analyzed with ai-based transformers predict traditional subjective well-being measures approaching the theoretical upper limits in accuracy. *Scientific Reports*, 12(1), March 2022. ISSN 2045-2322. doi: 10.1038/s41598-022-07520-w. URL http://dx.doi.org/10.1038/s41598-022-07520-w.

Inbal Magar and Roy Schwartz. Data contamination: From memorization to exploitation. In *Proceedings of ACL (Short)*, pp. 157–165, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.18. URL https://aclanthology.org/2022.acl-short.18/.

Jonibek Mansurov, Akhmed Sakip, and Alham Fikri Aji. Data laundering: Artificially boosting benchmark results through knowledge distillation. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8332–8345, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.407. URL https://aclanthology.org/2025.acl-long.407/.

OpenAI. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023. URL https://arxiv.org/abs/2303.08774.

Carolin A. Rickert, Manuel Henkel, and Oliver Lieleg. An efficiency-driven, correlation-based feature elimination strategy for small datasets. *APL Machine Learning*, 1(1):016105, 02 2023. ISSN 2770-9019. doi: 10.1063/5.0118207. URL https://doi.org/10.1063/5.0118207.

Oscar Sainz, Inbal Magar, Tianyi Zhang, Mikel Artetxe, and Jon Rodriguez. On the need to measure LLM data contamination for each benchmark. In *Findings of EMNLP*. Association for Computational Linguistics, 2023. URL https://aclanthology.org/2023.findings-emnlp.722.pdf.

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.

Vishakha Suresh Kalal, Andrew Parry, and Sean MacAvaney. Training on the test model: Contamination in ranking distillation. In *arXiv preprint arXiv:2411.02284*, 2024. URL https://arxiv.org/abs/2411.02284.

Bichen Wang, Yuzhe Zi, Yixin Sun, Yanyan Zhao, and Bing Qin. Rkld: Reverse kl-divergence-based knowledge distillation for unlearning personal information in large language models, 2024. URL https://arxiv.org/abs/2406.01983.

Taiqiang Wu, Chaofan Tao, Jiahao Wang, Runming Yang, Zhe Zhao, and Ngai Wong. Rethinking Kullback-Leibler divergence in knowledge distillation for large language models. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5737–5755, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.383/.

Junteng Zhang, He He, Chiyuan Zhang, et al. Language model developers should report train–test overlap. arXiv preprint arXiv:2410.17164, 2024. URL https://arxiv.org/abs/2410.17164.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pp. 19–27, USA, 2015. IEEE Computer Society. ISBN 9781467383912. doi: 10.1109/ICCV.2015.11. URL https://doi.org/10.1109/ICCV.2015.11.

A Statements

A.1 Ethics Statement

The datasets used in this work do not involve any sensitive or personally identifiable information, nor do they raise copyright concerns. Based on the experiments we conducted, we did not find evidence of systematic bias against different genders, languages, or regions. To the best of our knowledge, the experiments reported in this paper don't raise ethical concerns.

A.2 Reproducibility Statement

We are committed to ensuring the reproducibility of our results. All code, data, and preprocessing scripts will be released publicly after the anonymous review period. Detailed hyperparameters, hardware specifications, and other experimental settings are documented in Appendix B. Together, these materials should allow independent researchers to fully reproduce our results.

A.3 LLM Usage Disclosure

Large language models (LLMs) were used in a limited capacity during this work. First, AI tools were employed for minor English polishing, such as improving grammar and selecting more accurate word usage to convey ideas precisely. Second, AI tools were occasionally used to assist in debugging code during experiments. All scientific ideas, experimental designs, and contributions remain the authors' own.

B Experimental and Implementation Details

This section provides supplementary details on our experimental setup, models, datasets, and implementation.

B.1 Model Setup

Our experimental setup involves a comprehensive suite of models, detailed in Table 3. A key principle of our design is the direct comparability between baseline and student models: they share the same architecture, training hyperparameters, and number of training samples. The only distinction lies in their supervisory signal—baselines learn from ground-truth labels, while students learn from a teacher's outputs.

To fully explore the characteristics of each distillation method, we deviate from the standard paradigm. Specifically, we investigate setups both with and without the cross-entropy loss term on ground-truth labels for student training. This is controlled by a weighting coefficient, α. When $\alpha = 1$, students learn exclusively from the teacher's supervisory signal. This ensures that any observed performance gain in a dirty student is attributable purely to the laundered knowledge from the teacher. When $\alpha = 0.5$, the student learns from a balanced mix of the teacher's signal and the ground-truth labels.

For the sake of brevity, we primarily present the results from the Soft Forward distillation with $\alpha = 1$ in the main body of the paper. However, our experiments comprehensively cover all three core distillation methods: (i) soft-label distillation with forward KL divergence (SoftFwd), (ii) soft-label distillation with reverse KL divergence (SoftRev) (Amara et al., 2024), which uses an alternative divergence measure, and (iii) hard-label distillation (Hard) (Hinton et al., 2015). Each distillation

method is evaluated in settings both with and without the Ground Truth Cross-Entropy loss. Detailed results across all these configurations can be found in appendix C.

Model	Description	Training Data	Supervisory Signal	Loss Function	α
B_{clean}	Clean Baseline	Clean Train Set	Ground-truth labels	$\mathcal{L}_{CE}(y, \sigma(z_m))$	0
B_{dirty}	Dirty Baseline	Contaminated Train Set	Ground-truth labels	$\mathcal{L}_{CE}(y, \sigma(z_m))$	0
S^{hard}_{clean}	Student from T_{clean} (pure)	Clean Train Set	Hard labels from T_{clean}	$\mathcal{L}_{CE}(\hat{y}_t, \sigma(z_s))$	1
S^{hard}_{dirty}	Student from T_{dirty} (pure)	Clean Train Set	Hard labels from T_{dirty}	$\mathcal{L}_{CE}(\hat{y}_t, \sigma(z_s))$	1
$S^{hard,mix}_{clean}$	Student from T_{clean} (mixed)	Clean Train Set	GT + Hard labels from T_{clean}	$0.5 \cdot \mathcal{L}_{CE}(y, \sigma(z_s)) + 0.5 \cdot \mathcal{L}_{CE}(\hat{y}_t, \sigma(z_s))$	0.5
$S^{hard,mix}_{dirty}$	Student from T_{dirty} (mixed)	Clean Train Set	GT + Hard labels from T_{dirty}	$0.5 \cdot \mathcal{L}_{CE}(y, \sigma(z_s)) + 0.5 \cdot \mathcal{L}_{CE}(\hat{y}_t, \sigma(z_s))$	0.5
$S^{soft,fwd}_{clean}$	Student from T_{clean} (pure)	Clean Train Set	Soft labels from T_{clean}	$\text{KL}(\sigma(z_t/\tau) \parallel \sigma(z_s/\tau))$	1
$S^{soft,fwd}_{dirty}$	Student from T_{dirty} (pure)	Clean Train Set	Soft labels from T_{dirty}	$\text{KL}(\sigma(z_t/\tau) \parallel \sigma(z_s/\tau))$	1
$S^{soft,fwd,mix}_{clean}$	Student from T_{clean} (mixed)	Clean Train Set	GT + Soft labels from T_{clean}	$0.5 \cdot \mathcal{L}_{CE}(y, \sigma(z_s)) + 0.5 \cdot \text{KL}(\sigma(z_t/\tau) \parallel \sigma(z_s/\tau))$	0.5
$S^{soft,fwd,mix}_{dirty}$	Student from T_{dirty} (mixed)	Clean Train Set	GT + Soft labels from T_{dirty}	$0.5 \cdot \mathcal{L}_{CE}(y, \sigma(z_s)) + 0.5 \cdot \text{KL}(\sigma(z_t/\tau) \parallel \sigma(z_s/\tau))$	0.5
$S^{soft,rev}_{clean}$	Student from T_{clean} (pure)	Clean Train Set	Soft labels from T_{clean}	$\text{KL}(\sigma(z_s/\tau) \parallel \sigma(z_t/\tau))$	1
$S^{soft,rev}_{dirty}$	Student from T_{dirty} (pure)	Clean Train Set	Soft labels from T_{dirty}	$\text{KL}(\sigma(z_s/\tau) \parallel \sigma(z_t/\tau))$	1
$S^{soft,rev,mix}_{clean}$	Student from T_{clean} (mixed)	Clean Train Set	GT + Soft labels from T_{clean}	$0.5 \cdot \mathcal{L}_{CE}(y, \sigma(z_s)) + 0.5 \cdot \text{KL}(\sigma(z_s/\tau) \parallel \sigma(z_t/\tau))$	0.5
$S^{soft,rev,mix}_{dirty}$	Student from T_{dirty} (mixed)	Clean Train Set	GT + Soft labels from T_{dirty}	$0.5 \cdot \mathcal{L}_{CE}(y, \sigma(z_s)) + 0.5 \cdot \text{KL}(\sigma(z_s/\tau) \parallel \sigma(z_t/\tau))$	0.5

Table 3: Overview of the models in our experimental setup. T_{clean} and T_{dirty} denote the clean and dirty teachers. z_m, z_s, z_t are the logits from the main model, student, and teacher, respectively. σ is the softmax function, y is the ground-truth label, $\hat{y}_t$ is the teacher's hard prediction, and τ is the temperature. Hard distillation uses Cross-Entropy (CE) loss. Soft distillation uses KL Divergence; forward KL is "mean-seeking," while reverse KL is "mode-seeking" (Wu et al., 2025). The α parameter controls the weight of the ground-truth loss term; $\alpha = 1$ indicates pure distillation, while $\alpha = 0.5$ indicates a mixed objective.

B.2 Datasets

We use eight public classification benchmarks, detailed in Table 4. We adopt BERT-base-uncased for teachers and DistilBERT-base-uncased for students/baselines.

Benchmark	Task Type	Classes	Original Train Size	Original Test Size	Train Subset Ratio	Effective Train/Test Ratio
20newsgroups	Topic Classification	20	11,314	7,532	1.0	1.50
agnews	Topic Classification	4	120,000	7,600	0.1	1.58
banking77	Intent Classification	77	10,003	3,080	1.0	3.25
emotion	Emotion Classification	6	16,000	2,000	1.0	8.00
imdb	Sentiment Analysis	2	25,000	25,000	1.0	1.00
rotten_tomatoes	Sentiment Analysis	2	8,530	1,066	1.0	8.00
snli	Natural Language Inference	3	550,152	9,824	0.1	5.60
tweet_sentiment	Sentiment Analysis	3	45,615	12,284	0.5	1.86

Table 4: Details of the benchmark datasets used in our experiments. The Train Subset Ratio adjusts the training set size to control the relative influence of training versus injected test data.

Benchmark	Classes	Original Train Size	Size per Stratified Quintile	Original Test Size	Effective Train/Test Ratio
emotion	6	16,000	3,200	2,000	1.60
rotten_tomatoes	2	8,530	1,706	1,066	1.60

Table 5: Dataset details for the controlled distribution gap experiments. The training set for each benchmark was partitioned into five stratified quintiles based on semantic similarity to the test set. The table shows the resulting size of each quintile and the corresponding effective train/test ratio.

B.3 Hyperparameters and Computational Resources

The same set of hyperparameters was used for training all baseline, teacher, and student models to ensure a fair and controlled comparison. We trained all models for a fixed number of epochs and did not use a development set for early stopping. The specific hyperparameters are detailed in Table 6.

All experiments in Section 4 were conducted on NVIDIA A100 GPUs. All experiments in section 5 were conducted on AMD MI250X GPUs.

Hyperparameter	Value
Learning Rate	2e-5
Batch Size	32
Training Epochs	3
Distillation Temperature	2.0 (for soft distillation only)
Max Sequence Length	128 (512 for IMDB & 20newsgroups)
Random Seeds	1, 42, 86, 358, 1024

Table 6: Hyperparameters for all model training.

B.4 Significance Testing Details

To ensure the reliability of our findings, we employed bootstrapping-based statistical tests. The detailed procedures for assessing the significance of accuracy gains and correlation coefficients are outlined below.

Accuracy Gains (Clean vs. Dirty) To determine if the accuracy of a "dirty" model was significantly higher than its "clean" counterpart, we used a one-sided paired bootstrap test. For each of the five random seeds, we first drew 10,000 bootstrap samples (with replacement) from the test set predictions of the clean and dirty models. Then, for each bootstrap sample, we calculated the difference in accuracy between the dirty and clean models. Finally, the p-value was estimated as the proportion of bootstrap samples where the clean model's accuracy was greater than or equal to the dirty model's accuracy. To maintain a conservative assessment, we report the maximum p-value observed across the five random seeds for each benchmark comparison.

Correlation Coefficients To confirm the stability and significance of the Pearson correlation coefficient between the laundering effect and the contamination effect, we used a two-sided bootstrap test. For each random seed, we performed 10,000 bootstrap resamples of the test set samples. For each resample, we re-calculated the Pearson correlation. The p-value was then derived from the distribution of these bootstrapped correlation coefficients to test the null hypothesis that the true correlation is zero. We report the maximum p-value across the five seeds.

B.5 Model-Agnostic Data Characteristics

To characterize the relationship between the training and test sets in a model-agnostic way, we compute a suite of similarity metrics. These metrics, all normalized to the range $[0, 1]$, help identify inherent data properties that might make a benchmark more susceptible to leakage. Let $\mathcal{C}_{\text{train}}$ and $\mathcal{C}_{\text{test}}$ denote the training and test corpora, respectively. For some metrics, the similarity is a direct comparison between corpora, while for others, it is an aggregation of sample-level calculations.

- **Jaccard Similarity**: Measures lexical overlap based on the set of unique n-grams (N_g) present in each corpus. This is a direct corpus-level comparison.

$$\text{Jaccard}(\mathcal{C}_{\text{train}}, \mathcal{C}_{\text{test}}) = \frac{|N_g(\mathcal{C}_{\text{train}}) \cap N_g(\mathcal{C}_{\text{test}})|}{|N_g(\mathcal{C}_{\text{train}}) \cup N_g(\mathcal{C}_{\text{test}})|}$$

- **TF-IDF Cosine Similarity**: Measures similarity by comparing the centroids of the corpora in the TF-IDF vector space. First, each corpus $\mathcal{C}$ is represented by its mean TF-IDF vector, $\vec{v}_{\text{tfidf}}(\mathcal{C})$, which is an aggregation of individual sample vectors.

$$\vec{v}_{\text{tfidf}}(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{x \in \mathcal{C}} \vec{v}_{\text{tfidf}}(x)$$

  The final similarity is the cosine distance between these two mean vectors.

$$\text{Sim}_{\text{TF-IDF}}(\mathcal{C}_{\text{train}}, \mathcal{C}_{\text{test}}) = \text{cos\_sim}(\vec{v}_{\text{tfidf}}(\mathcal{C}_{\text{train}}), \vec{v}_{\text{tfidf}}(\mathcal{C}_{\text{test}}))$$

- **Average Embedding Similarity**: Similar to TF-IDF, this metric computes the cosine similarity between the mean Sentence-BERT embedding vectors of the corpora. The mean

embedding vector for a corpus, $\vec{e}(\mathcal{C})$, is derived by averaging the embeddings of all its samples.

$$\vec{e}(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{x \in \mathcal{C}} \text{emb}(x)$$

The similarity is then calculated between these two corpus-level representations.

$$\text{Sim}_{\text{AvgEmb}}(\mathcal{C}_{\text{train}}, \mathcal{C}_{\text{test}}) = \text{cos\_sim}(\vec{e}(\mathcal{C}_{\text{train}}), \vec{e}(\mathcal{C}_{\text{test}}))$$

- **Average Max Semantic Similarity**: Quantifies how well each test sample is represented in the training set. This metric is explicitly an aggregation of sample-level scores. For each test sample x_{test}, we find its highest cosine similarity to any sample in the training set, and then average these maximum similarity scores.

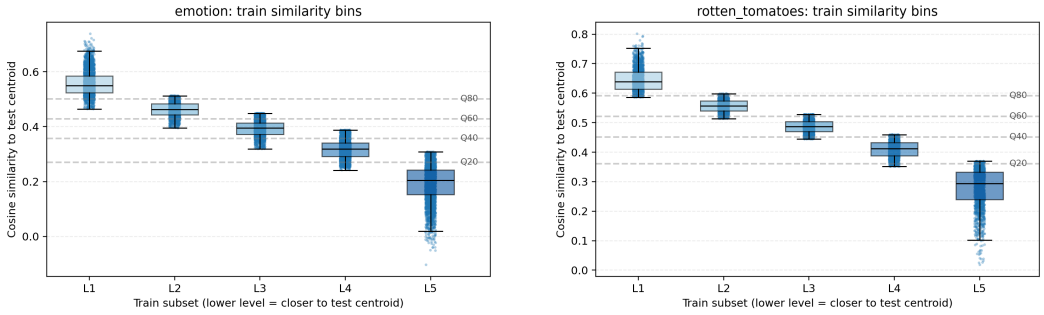
$$\text{Sim}_{\text{AvgMax}} = \frac{1}{|\mathcal{C}_{\text{test}}|} \sum_{x_{\text{test}} \in \mathcal{C}_{\text{test}}} \left(\max_{x_{\text{train}} \in \mathcal{C}_{\text{train}}} \text{cos\_sim}(\text{emb}(x_{\text{test}}), \text{emb}(x_{\text{train}})) \right)$$

- **Average Pattern Conformity**: Assesses how well test samples align with the dominant semantic patterns of the training set. We first run k-Means on the training embeddings to find k centroids $\{c_i\}_{i=1}^{k}$. The metric is the average of each test sample's maximum cosine similarity to any of these centroids, making it a clear aggregation of sample-level conformity scores.

$$\text{PatternConformity} = \frac{1}{|\mathcal{C}_{\text{test}}|} \sum_{x_{\text{test}} \in \mathcal{C}_{\text{test}}} \left(\max_{i \in \{1,\ldots,k\}} \text{cos\_sim}(\text{emb}(x_{\text{test}}), c_i) \right)$$

B.6 Verification of Controlled Distributional Gaps

To verify that our stratified splitting method effectively created controlled distributional gaps, we visualized the similarity between each training data level and the test set. Figure 4 shows the distribution of cosine similarities, confirming a systematic shift where Level 1 is most similar to the test set and Level 5 is least similar. Figure 5 further corroborates this by demonstrating a monotonic decrease across five different lexical and semantic similarity metrics, validating the integrity of our experimental setup.



(a) `emotion` dataset

(b) `rotten_tomatoes` dataset

Figure 4: Similarity bins for the `emotion` (left) and `rotten_tomatoes` (right) datasets. Both plots show the distribution of cosine similarities between training samples in each level and the test set centroid, confirming that Level 1 is most similar and Level 5 is least similar. It is worth noting that the levels are not perfectly discrete, which is a natural consequence of our stratified splitting strategy, as samples are partitioned within each class before being aggregated into global levels.

C Detailed Evaluation Statistics and Sanity Checks

To further validate the robustness of our main findings, we conducted a series of auxiliary experiments. These checks were designed to test our conclusions against alternative methodological choices and

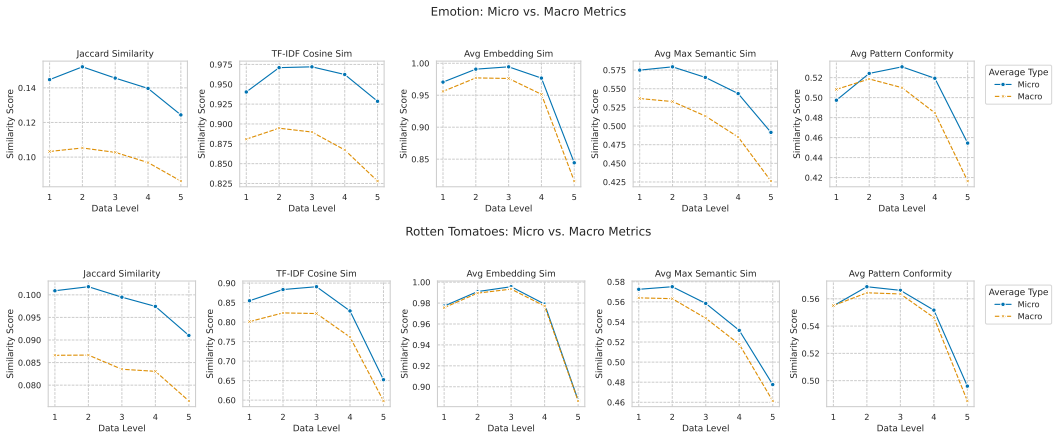


Figure 5: Verification of controlled distributional gaps using five similarity metrics for the `emotion` (top panel) and `rotten_tomatoes` (bottom panel) datasets. Both panels show a consistent decline across all metrics as the level increases from 1 to 5, validating the successful creation of a widening train-test gap.

potential confounding factors, ensuring that the observed decontamination effect of knowledge distillation is a genuine and reliable phenomenon. This appendix details the methodology and results of these investigations.

C.1 Full Performance Data for Baseline, Teacher, and Student Models

Benchmark	Teacher			Baseline (Supervised)			Student (Soft Fwd)		
	Clean	Dirty	Δ	Clean	Dirty	Δ	Clean	Dirty	Δ
20newsgroups	70.37 ± 0.19	84.62 ± 0.36	$14.25 \pm 0.60^{***}$	69.38 ± 0.28	81.29 ± 0.20	$11.91 \pm 0.35^{***}$	68.60 ± 0.30	70.02 ± 0.22	$1.42 \pm 0.45^{**}$
AGNews	91.89 ± 0.07	99.25 ± 0.05	$7.36 \pm 0.13^{***}$	91.64 ± 0.12	98.31 ± 0.12	$6.67 \pm 0.23^{***}$	91.59 ± 0.08	92.24 ± 0.12	$0.65 \pm 0.14^{***}$
banking77	78.65 ± 1.29	83.60 ± 1.36	$4.95 \pm 2.02^{**}$	76.97 ± 1.06	82.29 ± 0.58	$5.32 \pm 1.56^{**}$	62.21 ± 3.60	64.44 ± 1.55	2.23 ± 2.89
emotion	92.29 ± 0.29	98.35 ± 0.17	$6.06 \pm 0.33^{***}$	92.70 ± 0.26	97.59 ± 0.17	$4.89 \pm 0.25^{***}$	93.10 ± 0.36	93.76 ± 0.38	$0.66 \pm 0.31^{**}$
IMDb	93.63 ± 0.08	99.81 ± 0.03	$6.18 \pm 0.08^{***}$	92.78 ± 0.02	99.65 ± 0.02	$6.87 \pm 0.03^{***}$	92.72 ± 0.11	93.39 ± 0.08	$0.67 \pm 0.19^{**}$
rotten tomatoes	85.14 ± 0.55	98.65 ± 0.30	$13.51 \pm 0.52^{***}$	84.20 ± 0.32	97.52 ± 0.17	$13.32 \pm 0.18^{***}$	83.77 ± 0.36	84.33 ± 0.34	0.56 ± 0.59
SNLI	86.00 ± 0.49	98.72 ± 0.14	$12.73 \pm 0.58^{***}$	82.95 ± 0.20	96.10 ± 0.16	$13.14 \pm 0.35^{***}$	52.91 ± 11.55	58.67 ± 8.36	5.77 ± 14.99
tweet sentiment	67.81 ± 0.29	96.82 ± 0.30	$29.01 \pm 0.13^{***}$	67.33 ± 0.50	92.99 ± 0.09	$25.66 \pm 0.60^{***}$	66.99 ± 0.42	70.24 ± 0.30	$3.25 \pm 0.56^{***}$

Table 7: Baseline, teacher, and student performance across benchmarks. Values are percentage points; uncertainties denote standard deviation over seeds.

Table 7 presents the complete dataset corresponding to the figure in the main body that illustrates the performance of the baseline, teacher, and student models. This table provides a detailed breakdown of the results.

C.2 Robustness Across Distillation Strategies

To ensure our conclusions are robust, we tested them across various distillation methods and loss formulations. The results, presented in Tables 8, 9, and 10, show that the choice of distillation strategy does not alter our core findings. Specifically, these tables compare pure distillation with a mixed-loss approach for Soft Forward, Soft Reverse, and Hard-label methods, respectively.

Across all configurations, a consistent conclusion emerges: knowledge distillation effectively serves as a decontamination technique, significantly reducing the performance gains from direct data contamination. Furthermore, while data laundering can occasionally occur in specific scenarios, it is not a widespread or primary concern. This consistency across different methods validates our main conclusion. As an additional observation, the results across the three primary distillation methods (Soft Forward, Soft Reverse, and Hard) show minimal differences, further strengthening the claim that our findings are generalizable and not tied to a specific distillation technique.

Benchmark	Soft Fwd Clean	Soft Fwd Dirty	Soft Fwd Δ	Soft Fwd Mix Clean	Soft Fwd Mix Dirty	Soft Fwd Mix Δ
20newsgroups	68.60 ± 0.30	70.02 ± 0.22	$1.42 \pm 0.45^{**}$	69.04 ± 0.28	70.13 ± 0.20	$1.09 \pm 0.16^{***}$
AGNews	91.59 ± 0.08	92.24 ± 0.12	$0.65 \pm 0.14^{***}$	91.62 ± 0.11	92.26 ± 0.10	$0.64 \pm 0.15^{***}$
banking77	62.21 ± 3.60	64.44 ± 1.55	2.23 ± 2.89	77.91 ± 1.16	78.28 ± 0.37	0.37 ± 1.11
emotion	93.10 ± 0.36	93.76 ± 0.38	$0.66 \pm 0.31^{**}$	92.86 ± 0.32	93.17 ± 0.34	0.31 ± 0.45
IMDb	92.72 ± 0.11	93.39 ± 0.08	$0.67 \pm 0.19^{**}$	92.71 ± 0.12	93.76 ± 0.06	$1.05 \pm 0.18^{***}$
rotten tomatoes	83.77 ± 0.36	84.33 ± 0.34	0.56 ± 0.59	84.07 ± 0.27	84.65 ± 0.60	$0.58 \pm 0.59^{*}$
SNLI	52.91 ± 11.55	58.67 ± 8.36	5.77 ± 14.99	68.52 ± 10.65	74.18 ± 4.54	5.66 ± 11.24
tweet sentiment	66.99 ± 0.42	70.24 ± 0.30	$3.25 \pm 0.56^{***}$	67.20 ± 0.41	69.63 ± 0.19	$2.43 \pm 0.37^{***}$

Table 8: Student accuracy for Soft Forward and Soft Forward Mix distillation strategies. "Mix" refers to a mixed-loss objective with $\alpha = 0.5$.

Benchmark	Soft Rev Clean	Soft Rev Dirty	Soft Rev Δ	Soft Rev Mix Clean	Soft Rev Mix Dirty	Soft Rev Mix Δ
20newsgroups	68.90 ± 0.39	70.20 ± 0.32	$1.30 \pm 0.68^{*}$	69.16 ± 0.24	70.19 ± 0.24	$1.03 \pm 0.19^{***}$
AGNews	91.47 ± 0.14	92.09 ± 0.19	$0.63 \pm 0.14^{***}$	91.56 ± 0.17	92.26 ± 0.18	$0.70 \pm 0.17^{***}$
banking77	61.68 ± 4.19	63.34 ± 1.54	1.66 ± 3.76	78.68 ± 1.17	78.87 ± 0.62	0.19 ± 0.90
emotion	93.11 ± 0.21	93.48 ± 0.43	0.37 ± 0.50	92.87 ± 0.20	93.25 ± 0.31	0.38 ± 0.43
IMDb	92.68 ± 0.11	93.29 ± 0.07	$0.60 \pm 0.18^{**}$	92.69 ± 0.09	93.70 ± 0.03	$1.01 \pm 0.08^{***}$
rotten tomatoes	83.71 ± 0.48	84.09 ± 0.15	0.38 ± 0.57	84.15 ± 0.36	84.28 ± 0.31	0.13 ± 0.34
SNLI	52.84 ± 11.49	58.54 ± 8.44	5.70 ± 15.02	65.84 ± 12.19	71.76 ± 5.49	5.91 ± 13.17
tweet sentiment	67.29 ± 0.40	70.20 ± 0.23	$2.92 \pm 0.48^{***}$	67.28 ± 0.37	69.77 ± 0.24	$2.49 \pm 0.36^{***}$

Table 9: Student accuracy for Soft Reverse and Soft Reverse Mix distillation strategies.

Benchmark	Hard Clean	Hard Dirty	Hard Δ	Hard Mix Clean	Hard Mix Dirty	Hard Mix Δ
20newsgroups	69.10 ± 0.37	69.57 ± 0.26	$0.48 \pm 0.50^{*}$	69.27 ± 0.27	70.18 ± 0.20	$0.90 \pm 0.19^{***}$
AGNews	91.72 ± 0.16	91.80 ± 0.17	0.08 ± 0.21	91.66 ± 0.23	92.16 ± 0.14	$0.49 \pm 0.18^{**}$
banking77	66.61 ± 2.75	66.42 ± 2.05	-0.19 ± 2.06	70.87 ± 1.29	71.00 ± 1.40	0.13 ± 2.75
emotion	92.88 ± 0.25	93.57 ± 0.22	$0.69 \pm 0.35^{*}$	92.90 ± 0.32	93.34 ± 0.15	$0.44 \pm 0.33^{*}$
IMDb	92.79 ± 0.15	93.06 ± 0.07	$0.27 \pm 0.22^{*}$	92.82 ± 0.04	93.58 ± 0.05	$0.76 \pm 0.06^{***}$
rotten tomatoes	84.15 ± 0.33	84.11 ± 0.31	-0.04 ± 0.34	84.18 ± 0.36	84.45 ± 0.22	0.26 ± 0.54
SNLI	52.69 ± 11.55	58.17 ± 8.12	5.47 ± 14.88	69.36 ± 8.00	74.23 ± 4.10	4.87 ± 8.98
tweet sentiment	67.16 ± 0.27	69.33 ± 0.24	$2.17 \pm 0.18^{***}$	67.26 ± 0.29	69.01 ± 0.24	$1.75 \pm 0.31^{***}$

Table 10: Student accuracy for Hard and Hard Mix distillation strategies.

C.3 Pretraining Corpus Overlap Audit

A potential concern is that the base models used in our study, BERT and DistilBERT, might have been inadvertently exposed to benchmark test data in their original pretraining corpora. To ensure that our findings originate from our explicit contamination protocols rather than from such pre-existing issues, we conducted a corpus-level overlap audit. According to their respective documentation, both BERT and DistilBERT were pretrained on a combination of English Wikipedia and the BookCorpus (Zhu et al., 2015). We created a surrogate pretraining corpus composed of recent snapshot of Wikipedia ('20231101.en')[2] and the BookCorpusOpen[3]. Subsequently, we performed an exhaustive search to determine if any sentence from the test sets of our eight benchmarks appeared verbatim within this extensive corpus.

The audit revealed zero exact 13-gram matches between any of our benchmark test sets and the surrogate pretraining data. While this does not preclude more subtle forms of semantic overlap, it provides strong evidence that our results are not confounded by the most direct form of test set leakage into the pretraining pipeline of the models we used. This finding increases our confidence that the leakage effects studied in this paper are indeed a consequence of our controlled experiments.

[2]https://huggingface.co/datasets/wikimedia/wikipedia

[3]https://huggingface.co/datasets/lucadiliello/bookcorpusopen