A DISTRIBUTIONAL DISTANCE PERSPECTIVE ON EVALUATING LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Benchmark accuracy remains the primary criterion for evaluating large language models (LLMs). Even without exact test-set contamination, it may conflate task capability with input familiarity because LLMs are trained on heterogeneous, largely undisclosed corpora. Within a distributional-distance framework, we use perplexity and Min-K% as observable proxies for model-relative input surprise, given limited access to pretraining corpora. Controlled experiments with GPT-2 Small and SmolLM2-135M trained on four corpora show that both proxies correlate with a model-specific k-nearest-neighbor reference distance, reaching 0.749 and 0.847, respectively. Across nine LLMs and thirteen benchmarks, we observe predominantly negative model-level associations between accuracy and proxy distance, with mean Spearman correlations of -0.502 and -0.646. We therefore propose a two-dimensional diagnostic that jointly reports accuracy and distributional-distance proxies. Among 460 non-tied pairwise comparisons, 75.0% exhibit an accuracy–distance trade-off under Min-K%, indicating that accuracy-only rankings often admit an alternative explanation based on input familiarity. Distribution-aware reporting thus provides a more transparent and informative basis for interpreting benchmark performance and comparing LLM capabilities across heterogeneous evaluation settings. The code is available on `https://anonymous.4open.science/r/distributional-distance-iclr2027`.

1 INTRODUCTION

Large language models (LLMs) (Comanici et al., 2025; Yang et al., 2025; Adcock et al., 2026) have demonstrated remarkable performance in various real-world tasks, including reasoning tasks (Wang et al., 2026), code generation (Chen et al., 2025a; Tian & Chen, 2025), and question answering (Sui et al., 2025). Typically, such performance is measured by accuracy on public benchmarks, which is currently the standard evaluation criterion for demonstrating model capability (Wang et al., 2018; Gu et al., 2025). However, this accuracy-based evaluation suffers from a notable limitation due to variations in training data: data scale, data sources, and domain distributions vary substantially across different LLMs, which means that this standard evaluation can introduce evaluation bias and fail to genuinely reflect the generalization capability of the models.

A primary source of evaluation unfairness stems from data contamination, which is defined as test set samples being directly or indirectly included in the model's training data (Magar & Schwartz, 2022; Li & Flanigan, 2024; Balloccu et al., 2024; Liang et al., 2026). Such contamination causes the model to "recite" rather than "reason" during evaluation, artificially inflating its reported performance (Choi et al., 2025; Chen et al., 2025b). A surge of studies has proposed notable approaches to detect data contamination, and recent studies (Wang et al., 2024a; Xu et al., 2024; 2025) have taken data contamination into account in evaluating LLMs.

In this paper, we argue that accuracy-based evaluation may still be unfair even in the absence of data contamination. Our intuition is based on the distributional distance of inputs. As shown in Figure 1, generalization ability is essentially reflected by whether a model M can produce high-quality outputs for an input x that lies far from its training distribution (Liu et al., 2021; Yuan et al., 2023). This may suggest a strong correlation between model accuracy on a test set and the distributional distance between the test inputs and the corresponding training data. However, since the training data of most LLMs are typically unavailable, directly measuring this distance is impractical. We

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

therefore propose two surrogate methods to approximate the distributional distance between an input (without its response) and the training data. Our experimental results verify the effectiveness of these approximations. We further empirically demonstrate that the approximate distributional distance correlates strongly with accuracy. This finding supports the hypothesis that distributional distance due to different training data contributes to the unfairness in LLM evaluation.

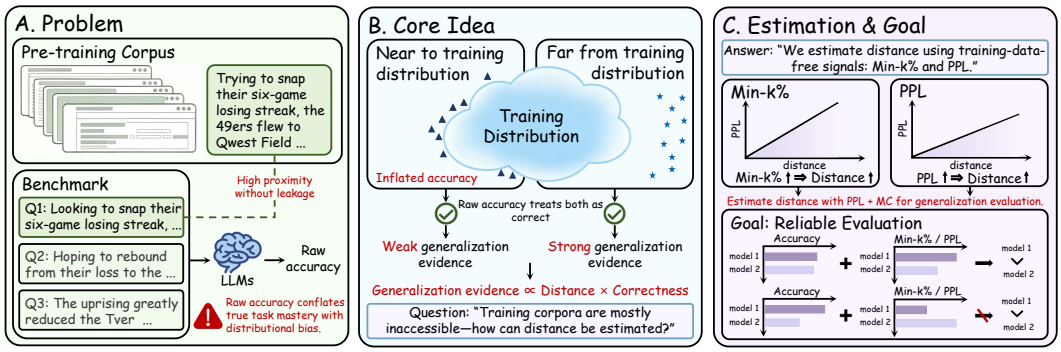


Figure 1: Overview of our evaluation framework. (A) Even without data contamination, raw accuracy conflates genuine generalization with proximity to the training distribution. (B) Correctness on far-distribution instances provides stronger generalization evidence. (C) With training corpora inaccessible, we estimate distance using Min-K% and PPL, then jointly consider accuracy and distance for fairer evaluation.

To this end, we propose a distance-aware evaluation criterion that combines accuracy with distributional distance. Specifically, when comparing two LLMs, the model with higher accuracy and larger distributional distance is considered superior. This criterion alleviates evaluation bias arising from differences in training data across models. Our main contributions are summarized as follows:

- To the best of our knowledge, we are the first to empirically demonstrate a strong correlation between distributional distance and LLM accuracy.
- We propose a distance-aware two-dimensional evaluation criterion with two simple implementations, and advocate a two-dimensional evaluation framework for assessing genuine generalization ability.
- We conduct a large-scale re-evaluation on 13 benchmarks with over 50,000 samples across nine representative LLMs, revealing how distributional distance shapes model performance and providing insights for fairer evaluation under distribution shift.

2 Distributional Distance Perspective

2.1 Motivation

Suppose we are given two large language models M_1 and M_2, and their pretraining datasets are respectively denoted by D_1 and D_2. If an input x is closer to $D_1 = \{x_1^1, \cdots, x_N^1\}$ than $D_2 = \{x_1^2, \cdots, x_N^2\}$, it is possible that the response y_1 from the model M_1 is more accurate than the response y_2 from M_2, which aligns with principles in statistical learning (Shimodaira, 2000). Motivated by this observation, we hypothesize that the distributional distance between the input x and training data may lead to evaluation bias under accuracy-based evaluation. To this end, we first formally define the distributional distance (§2.2) and then present two approximate methods to estimate it (§2.4). In §3, using these approximate methods, we verify the hypothesis and propose a distance-aware evaluation criterion.

2.2 Definition of Distributional Distance

Formally, suppose $D = \{x_1, \cdots, x_N\}$ denotes the pretraining data of a given LLM M, where x_i denotes a document and N is a super-large number. Such pretraining data generally specify an empirical distribution $P_{\text{emp}}(x) = \frac{1}{N}\sum_i \delta(x, x_i)$, where δ is a discrete-valued Dirac function which

equals one if and only if x coincides with x_i, and zero otherwise. Note that the empirical distribution is very sparse, and it is incapable of measuring which of the two samples x and x' lies closer to the training-set distribution if neither of them is in the pre-training data. Instead, we employ the K nearest-neighbor smoothed empirical distribution to define the **distributional distance** of x with respect to the distribution of the training data D:

$$P(x; D) \propto \exp\big(-\frac{1}{K}\sum_{x' \in \mathcal{N}(x,D)} d(x, x')\big)$$

where $\mathcal{N}(x, D)$ denotes the K nearest-neighboring documents of x in D according to a distance function $d(x, x')$ which can be defined by the Euclidean distance between two instances x and x' that are mapped into the Euclidean space via the LLM M.

Then, it is straightforward to extend the distributional distance of a sample to that of a set of samples $\{x'_1, \cdots, x'_m\}$ as follows:

$$P(x'_1, \cdots, x'_m; D) \propto \exp\big(-\frac{1}{K}\sum_{i}\sum_{x' \in \mathcal{N}(x'_i,D)} d(x'_i, x')\big)$$

Generally, given training data D, it involves two steps to calculate $P(x; D)$: 1) a distance function d to measure the distance between two instances; and 2) a search algorithm to obtain the K nearest-neighboring $\mathcal{N}(x, D)$ for x.

2.3 Challenge in Estimating Distributional Distance

Estimating the distributional distance $P(x; D)$ faces a fundamental challenge in LLM scenarios: the training data D of most LLMs are typically inaccessible. For most modern LLMs, the composition and distribution of training corpora remain unknown, making the distributional distance $P(x; D)$ fundamentally unobservable. Under this setting, the only accessible information usually consists of observable LLM behaviors on given inputs, such as logits, token-level probabilities, and generation likelihoods. Consequently, in the absence of explicit training data, a central challenge is how to effectively estimate the distributional distance $P(x; D)$ from observable model behaviors.

2.4 Approximate Estimation

To address this challenge, we employ two approximate methods that require no access to training data. Specifically, the proposed methods utilize model outputs and logits to estimate the relative proximity between benchmark samples and the training distribution. The first method, Perplexity (PPL), characterizes the overall likelihood of input sequences under the model distribution, while the second method, Min-K%, captures local low-confidence byte-level positions. Together, these two methods characterize training-distribution proximity from complementary perspectives.

Since both approximate methods are not explicitly dependent on the training data but on the LLM M, we adopt the consistent notation $\text{Dist}(x, M)$, where x is the input of an instance and M is a large language model. Similarly, Dist can be defined on a dataset $S = \{x'_1, \cdots, x'_m\}$ as follows:

$$\text{Dist}(S, M) = \frac{1}{|S|}\sum_{x \in S} \text{Dist}(x, M)$$

2.4.1 Perplexity (PPL) as a Proxy

PPL measures the average likelihood the model assigns to an input sequence. A lower PPL indicates higher confidence, suggesting that x lies in a high-density region of the training data D. We use PPL as an inverse proxy for distributional distance: the smaller $\text{PPL}(x)$, the smaller the distance $P(x; D)$.

We adopt the byte-normalized form. Let the tokenizer of M decompose x into L tokens $(x_1, \ldots, x_L)$; denote by b_t the number of bytes spanned by token x_t and by $B(x) = \sum_{t=2}^{L} b_t$ the total byte

length (excluding the first token, which has no preceding context for conditioning). We weight each log-probability by b_t:

$$\mathrm{Dist}_{\mathrm{PPL}}(x, M) = \exp\left(-\frac{1}{B(x)} \sum_{t=2}^{L} b_t \log P_M(x_t \mid x_{<t}) \right).$$

Here $P_M(x_t \mid x_{<t})$ is the probability M assigns to x_t given its preceding context; the formula aggregates these weighted log-likelihoods to capture global proximity to the training distribution.

2.4.2 MIN-K% AS A PROXY

Min-K% (Shi et al., 2024) is a local confidence measure based on model logits, focusing on positions where the model is least confident. To enable cross-tokenizer comparison, we compute Min-K% at the byte level: replicate each token's negative log-likelihood b_t times:

$$\big(\underbrace{-\log P_M(x_2 \mid x_{<2}), \ldots}_{b_2}, \underbrace{-\log P_M(x_3 \mid x_{<3}), \ldots}_{b_3}, \ \ldots\big),$$

and average the bottom $K\%$ of these $B(x)$ bytes:

$$\mathrm{Dist}_{\text{Min-K\%}}(x, M) = \frac{1}{\lceil K\% \cdot B(x) \rceil} \sum_{j \in \text{bottom-}K\% \text{ bytes}} \big(-\log P_M(x_{t(j)} \mid x_{<t(j)})\big),$$

where $t(j)$ indexes the source token of byte j.

In the experiments (see §4.2), we empirically verify that both approximate methods are strongly correlated with the true distributional distance $P(x; D)$ when the training data of a language model is available. Numerous studies (Shi et al., 2024; Yang et al., 2026) use PPL and Min-K% to detect whether a test example (x, y) is included in the training data D of an LLM; here we use them to measure the distance between the *input* x and the training data.

3 DISTANCE-AWARE EVALUATION CRITERION

3.1 REVISITING STANDARD EVALUATION

Accuracy-based Evaluation. Given two LLMs M_1 and M_2, and a benchmark test set $S = \{(x_j, y_j)\}_{j=1}^{n}$, where x_j denotes an input sample and y_j denotes the corresponding ground-truth output, the traditional approach directly computes the accuracy on S:

$$\mathrm{Acc}(M_i) = \frac{1}{n} \sum_{j=1}^{n} \Delta\big(M_i(x_j), y_j\big)$$

where Δ is a function that compares the consistency of the response $M_i(x_j)$ generated by M_i with the ground-truth answer y_j. For example, in multiple-choice question answering, Δ is an exact-match function which returns 1 if $M_i(x_j)$ is equal to y_j and 0 otherwise.

Under the standard evaluation criterion, candidate models are typically compared based on test-set accuracy, where a model with higher accuracy is considered superior.

Bias in Standard Evaluation. Recall our distributional distance hypothesis from §2: If an input x is closer to the training data D_1 of M_1 than the training data D_2 of M_2, it is possible that the response y_1 from M_1 is more accurate than the response y_2 by M_2.

If this hypothesis holds, it may imply that the standard evaluation based on accuracy is biased due to the different distributional distances between the inputs and training data.

To verify this hypothesis, we evaluate nine models across 13 benchmarks, using two approximate methods to compute distributional distance for each test set. As shown in Table 1, the correlation between distributional distance and accuracy reaches up to 0.9 on some datasets (see §4.3 for details), suggesting that ACC-based evaluation can be biased due to variations in distributional distance.

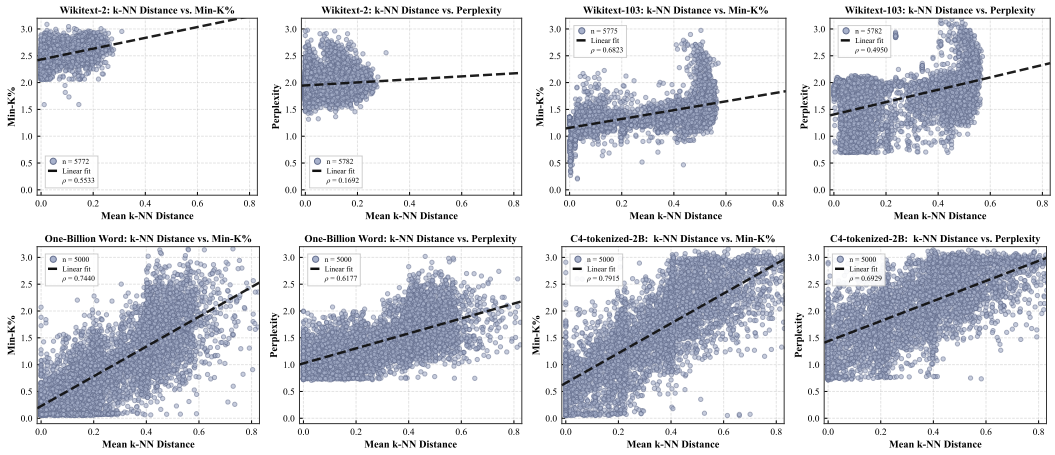


Figure 2: Controlled validation of distance proxies (PPL and Min-K%) against the distributional distance $P(x; D)$ on the GPT-2 Small model over WikiText-2, WikiText-103, One-Billion Word, and C4-tokenized-2B.

3.2 A Two-Dimensional Evaluation Criterion

Our goal is to compare two models M_1 and M_2 on the same benchmark S while controlling for differences in their training distributions. The core idea is that a model which achieves high accuracy on inputs far from the training distribution demonstrates stronger genuine generalization ability.

We propose the following distance-aware criterion. For two models M_1, M_2 and a test set S, let $\mathrm{Acc}(S, M_1) > \mathrm{Acc}(S, M_2)$ denote their accuracies, and let $\mathrm{dist}(S, M_i)$ denote their distributional proximities (e.g., Min-K% or PPL). The criterion is as follows:

- If $\mathrm{Acc}(S, M_1) > \mathrm{Acc}(S, M_2)$ and $\mathrm{dist}(S, M_1) > \mathrm{dist}(S, M_2)$, then M_1 has stronger overall generalization ability than M_2. In this case, M_1 achieves higher accuracy on test inputs that are further from its training distribution, so its advantage more likely comes from genuine reasoning rather than training-set familiarity.
- If $\mathrm{Acc}(S, M_1) > \mathrm{Acc}(S, M_2)$ yet $\mathrm{dist}(S, M_1) < \mathrm{dist}(S, M_2)$, we cannot conclude that M_1 is better. The higher accuracy of M_1 may be partly due to its training distribution being closer to the test set, not necessarily to superior generalization.

This criterion evaluates models along two dimensions: accuracy and distributional distance, and avoids unfairness from differences in training distributions that affect traditional evaluation methods.

3.2.1 Related to Data Contamination in LLM Evaluation

The proposed evaluation criterion originates from the perspective of distributional distance $P(x; D)$, defined over the input x in the test data and independent of its response y. This is fundamentally distinct from the well studied perspective of data contamination (Duarte et al., 2024; Xu et al., 2025): most of these studies focus on leakage of the response y and hence they focus on the distribution of y conditioned on the input x. Moreover, our scope is broader: even in the absence of data leakage, unfairness arising from distributional distance still persists.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We evaluate on 13 benchmarks covering knowledge (MMLU (Hendrycks et al., 2020), MMLU-Pro (Wang et al., 2024b), ARC-C (Clark et al., 2018), GPQA-Diamond (GPQA-D) (Rein et al., 2023)), reasoning (IFEval (Zhou et al., 2023), GSM8K (Cobbe et al., 2021), MGSM (Shi et al., 2022), MATH (Hendrycks et al., 2021), AIME2025 [1], BBH (Suzgun et al., 2023), DROP (Dua et al., 2019)), and code generation (HumanEval (Chen et al., 2021), MBPP (Austin et al., 2021)). For models, we include nine representative LLMs from three mainstream families, namely Qwen2.5 (Team, 2024), Llama3 (Grattafiori et al., 2024), and Ministral3 (Liu et al., 2026).

4.2 VALIDATION ON DISTRIBUTIONAL DISTANCE ESTIMATION

To validate the correlation between distributional distance and its two proxies, we train GPT-2 Small (Radford et al., 2019) (124M parameters) and SmolLM2-135M (Allal et al., 2025) from scratch on WikiText-2, WikiText-103 (Merity et al., 2017), One-Billion Word (Chelba et al., 2013), and C4-tokenized-2B[2], and construct a balanced query set by pairing each test instance with an equal number of training instances sampled without replacement. For each instance i, the true distance d_i is the average cosine distance to its $k = 45$ nearest neighbors in the training set (Appendix F), and the Min-K% proxy uses $K = 10$ (Appendix G). Figure 2 presents the results of controlled experiments on GPT-2 Small, with corresponding results on SmolLM2-135M reported in Appendix B. On the smaller WikiText-2 corpus, perplexity (PPL) exhibits almost no correlation ($\rho = 0.1692$), whereas Min-K% shows a moderate positive correlation ($\rho = 0.5533$). As the training corpus scales up, the correlation between both proxies and the true distance grows steadily. On the larger C4-tokenized-2B corpus, Min-K% and PPL achieve strong correlations of 0.7915 and 0.6929, respectively. This trend is consistent with prior observations (Carlini et al., 2022; Kocyigit et al., 2025; Kiyomaru et al., 2024) that scaling model and training data size increases the model's dependence on the training distribution, thereby strengthening the relationship between proxy metrics and true distributional distance. Consequently, in modern large-scale LLM pretraining settings, both PPL and Min-K% can serve as practical proxies for distributional distance without requiring access to the training corpus.

Benchmark	Acc vs. Min-K% Spearman ρ (mean $\pm$ std)	Acc vs. PPL Spearman ρ (mean $\pm$ std)
HumanEval	-0.9651 $\pm$ 0.0294	-0.9672 $\pm$ 0.0443
MMLU-Pro	-0.9494 $\pm$ 0.0201	-0.9515 $\pm$ 0.0111
AIME2025	-0.8320 $\pm$ 0.0445	-0.9371 $\pm$ 0.0201
MATH	-0.8277 $\pm$ 0.0686	-0.7838 $\pm$ 0.0367
GPQA-D	-0.6915 $\pm$ 0.0306	-0.8138 $\pm$ 0.0415
DROP	-0.7374 $\pm$ 0.0270	-0.7014 $\pm$ 0.0143
MMLU	-0.8655 $\pm$ 0.0344	-0.5247 $\pm$ 0.0442
MGSM	-0.5902 $\pm$ 0.0287	-0.7028 $\pm$ 0.0164
MBPP	-0.5769 $\pm$ 0.1281	-0.4593 $\pm$ 0.1032
ARC-C	-0.8212 $\pm$ 0.0321	-0.0955 $\pm$ 0.0581
GSM8K	-0.5585 $\pm$ 0.0998	-0.1415 $\pm$ 0.2407
BBH	-0.2927 $\pm$ 0.0951	-0.0449 $\pm$ 0.0935
IFEval	0.3134 $\pm$ 0.1229	0.5952 $\pm$ 0.0678
Average	**-0.6458 $\pm$ 0.0586**	**-0.5022 $\pm$ 0.0609**

Table 1: Spearman correlation between model accuracy and Min-K% / PPL across benchmarks.

4.3 CORRELATION BETWEEN DISTRIBUTIONAL DISTANCE AND ACCURACY

To quantitatively verify that distributional distance substantially affects model accuracy, we conduct a repeated sampling experiment. For each benchmark, we independently draw 1,000 test samples with replacement, repeating this 10 times to obtain reliable statistical estimates. Table 1 reports the mean and standard deviation of the Spearman correlation ρ across the 10 runs for each benchmark, separately for Min-K% and PPL. The results reveal strong negative correlations across most benchmarks. The average correlation is $\rho = -0.6458$ (Min-K%) and -0.5022 (PPL), indicating that test samples closer to the training distribution are systematically answered more correctly. This provides direct

[1]The AIME2025 dataset is maintained by OpenCompass and available at `https://huggingface.co/datasets/opencompass/AIME2025`.

[2]The C4-tokenized-2B dataset is maintained by Neel Nanda and available at `https://huggingface.co/datasets/NeelNanda/c4-tokenized-2b`.

Model	Metrics	Knowledge					Reasoning						Coding	
		MMLU	MMLU-Pro	ARC-C	GPQA-D	IFEval	GSM8K	MGSM	MATH	AIME2025	BBH	DROP	HumanEval	MBPP
Qwen2.5 (7B)	Acc (%)	74.23	56.55	92.05	32.32	71.72	80.52	56.80	75.20	6.67	54.88	80.46	82.32	62.40
	Min-k%	2.77	1.49	1.90	1.57	3.67	2.47	1.94	1.12	1.29	2.61	1.50	0.89	2.43
	PPL	1.50	1.60	3.82	1.93	2.22	1.47	2.18	1.98	1.57	2.47	1.56	0.87	1.97
Qwen2.5 (14B)	Acc (%)	79.82	65.25	92.16	45.96	79.11	82.41	77.35	73.40	16.67	73.98	78.79	84.76	73.34
	Min-k%	2.65	1.40	1.83	1.48	3.41	2.27	1.74	1.01	1.18	2.41	1.35	0.80	2.36
	PPL	1.52	1.53	3.77	1.77	1.84	1.33	1.97	1.63	1.43	2.49	1.39	0.77	1.70
Qwen2.5 (32B)	Acc (%)	82.94	69.32	93.94	44.95	79.11	83.17	80.87	77.40	10.00	77.07	82.03	88.41	79.77
	Min-k%	2.62	1.37	1.78	1.47	3.49	2.26	1.76	0.79	1.23	2.46	1.34	0.68	2.61
	PPL	1.46	1.48	3.63	1.77	1.75	1.26	1.98	1.53	1.46	2.49	1.39	0.69	1.84
Llama3 (8B)	Acc (%)	71.28	40.70	84.71	33.33	69.32	84.38	58.87	42.00	0.00	67.39	67.75	69.51	59.20
	Min-k%	3.29	1.72	2.29	1.71	3.78	2.32	2.13	1.87	1.54	2.77	1.83	1.68	3.15
	PPL	1.65	1.95	4.29	2.11	1.76	1.67	2.36	2.33	1.82	2.87	1.92	1.40	2.22
Llama3.1 (8B)	Acc (%)	71.21	47.90	75.95	22.22	75.23	84.31	65.38	45.00	0.00	64.95	75.08	70.12	61.62
	Min-k%	3.15	1.60	2.21	1.62	3.69	2.23	1.87	1.81	1.57	2.68	1.73	1.54	3.09
	PPL	1.51	1.80	3.44	1.88	1.73	1.45	2.11	2.13	1.83	2.76	1.83	1.34	2.12
Llama3.2 (8B)	Acc (%)	71.20	46.45	76.49	20.20	75.60	84.08	66.04	41.20	0.00	65.58	72.69	71.34	62.66
	Min-k%	3.15	1.59	2.22	1.62	3.70	2.23	1.87	1.81	1.58	2.68	1.73	1.54	3.10
	PPL	1.50	1.78	3.35	1.87	1.72	1.42	2.07	2.08	1.80	2.71	1.82	1.32	2.07
Ministral3 (3B)	Acc (%)	73.99	57.88	38.46	34.34	57.30	82.26	59.38	68.60	6.67	57.73	56.29	71.95	59.03
	Min-k%	3.07	1.53	2.09	1.61	3.43	2.11	2.31	1.74	1.43	2.50	1.84	1.41	2.73
	PPL	1.53	1.69	3.27	1.85	1.62	1.46	2.50	2.14	1.64	2.57	1.92	1.18	1.76
Ministral3 (8B)	Acc (%)	79.11	66.95	91.35	49.49	64.70	89.01	73.13	75.40	10.00	63.25	77.38	82.32	70.53
	Min-k%	2.90	1.43	1.93	1.52	3.37	1.97	2.05	1.62	1.33	2.39	1.68	1.27	2.65
	PPL	1.43	1.50	3.07	1.72	1.55	1.38	2.22	1.93	1.56	2.45	1.73	1.11	1.87
Ministral3 (14B)	Acc (%)	81.94	69.25	92.82	48.99	69.13	88.55	75.56	76.20	16.67	76.78	30.63	84.15	68.89
	Min-k%	2.81	1.39	1.86	1.49	3.29	1.94	1.98	1.57	1.31	2.37	1.60	1.12	2.55
	PPL	1.38	1.47	3.02	1.69	1.54	1.34	2.15	1.88	1.50	2.42	1.66	0.99	1.84

Table 2: Evaluating nine models on 13 benchmarks in terms of (two-dimension criterion) accuracy and distributional distance measured by Min-k% and PPL.

quantitative support for our core claim: distributional distance inflates evaluation accuracy, causing substantial overestimation of performance on tasks near the training distribution.

IFEval is the only notable exception, exhibiting positive correlations of $\rho = 0.3134$ (Min-K%) and 0.5952 (PPL). This anomaly is primarily driven by the Ministral family: within Qwen and Llama, accuracy and both metrics consistently follow the expected negative correlation, whereas Ministral models perform poorly even on distribution-matched samples, suggesting their instruction-following capability is not yet fully mature. This isolated exception does not undermine our overall conclusion.

Further, to fully eliminate the confounding interference caused by general model capability, we conduct a partial correlation analysis by controlling for each model's loss on a neutral held-out corpus (Appendix H). After removing the influence of general capability, the average partial correlation remains -0.6309, demonstrating only a minor attenuation and confirming that the effect of distributional distance is robust and independent of general model capability. Overall, our experimental results reveal a crucial insight: traditional evaluation paradigms that rely solely on raw accuracy while ignoring distributional discrepancies tend to produce unfair and misleading model comparisons.

4.4 Results on Distance-aware Evaluation

4.4.1 Results on Distributional Distance

First, Min-K% generally correlates well with PPL in Table 2, although some exceptions exist. For example, on GSM8K, Qwen2.5 14B achieves a higher Min-K% value than Ministral3 14B (2.27 vs. 1.94) but its PPL is lower than the latter (1.33 vs. 1.34); a similar exception occurs on MBPP between Ministral-3B and 8B, where their Min-K% comparison is 2.73 vs. 2.65 while their PPL

comparison is 1.76 vs. 1.87. As validated in §4.2, Min-K% is better than PPL as an approximate proxy for distributional distance, and thus we mainly draw conclusions according to Min-K%.

Second, the distributional distance varies substantially across models from different families. For instance, on MMLU, Qwen2.5-14B attains a Min-K% of 2.65 versus 2.81 for Ministral3-14B; on MBPP, Llama3-8B reaches 3.15 versus 2.65 for Ministral3-8B. As a result, it is more urgent to account for distributional distance when comparing the accuracy for models from different families.

Third, within the same model family, larger models consistently yield lower Min-K% values, indicating perceived proximity to the training distribution. For instance, Qwen2.5-32B generally produces smaller Min-K% than Qwen2.5-7B across all benchmarks except MBPP (Table 2). This likely stems from larger models being trained on more diverse corpora or having greater capacity to capture fine-grained regularities, making test inputs appear closer to training samples.

4.4.2 Two-Dimensional Evaluation Results

We first perform an aggregate pairwise diagnosis over Table 2. With nine models per benchmark, there are 36 pairs per benchmark, totaling 468 across 13 benchmarks. After excluding accuracy ties, 460 non-tied comparisons remain. For each non-tied pair, we label it as *distance-supported* if the higher-accuracy model also has a larger Min-K% value, indicating that its advantage comes from inputs farther from its training distribution and thus provides stronger evidence of generalization. Conversely, we label it as *distance-confounded* if the higher-accuracy model has a lower Min-K% value, suggesting that the accuracy advantage may partly reflect proximity to its training distribution.

Among the 460 non-tied comparisons, only 107 (23.3%) are distance-supported. For example, on IFEval, Llama3-8B outperforms Ministral3-8B in accuracy (69.32% vs. 64.70%) while also having a larger Min-K% value (3.78 vs. 3.37). In contrast, 345 comparisons (75.0%) are distance-confounded. For example, on MATH, Ministral3-14B substantially outperforms Llama3-8B in accuracy (76.20% vs. 42.00%), but it has a lower Min-K% value (1.57 vs. 1.87), suggesting that its raw accuracy advantage may be partly explained by the benchmark inputs appearing closer to its training distribution under the Min-K% proxy. The remaining 8 comparisons (1.7%) have tied Min-K% values.

We further re-rank models using a composite score $\mathrm{GS} = \mathrm{Acc} \times \widetilde{\mathrm{Dist}}$, where $\widetilde{\mathrm{Dist}}$ is the per-benchmark normalized Min-K% (Appendix I). As reported in Table 10, this re-ranking reveals substantial rank reversals: the Spearman correlation between GS-based and accuracy-only rankings averages $\rho = -0.4346$, with 66.5% of model pairs changing relative order. Overall, the majority of accuracy advantages are not supported by distributional distance evidence, and incorporating distance frequently reverses which model is deemed superior. These findings demonstrate that accuracy-only evaluation can systematically overestimate generalization ability and advocate for distance-aware assessment in fair model comparison.

Benchmark	Top-100	Bottom-100	Benchmark	Top-100	Bottom-100
GPQA-D	90	92	ARC-C	44	22
HumanEval	56	50	MATH	25	26
MMLU-PRO	46	48	DROP	12	35
IFEval	42	36	MMLU	6	32
BBH	18	52	GSM8K	20	14
MBPP	24	43	MGSM	1	17

Table 3: Consensus among nine models on the Top-100 and Bottom-100 samples for each benchmark.

4.5 Consensus Analysis on Distance

To assess cross-model consensus, each model ranks benchmark samples by Min-K%, with higher scores indicating greater distance, and selects the top-100 farthest and bottom-100 nearest samples. We then measure their intersections across the nine models.

4.5.1 Benchmark-dependent consensus

Model consensus varies dramatically across benchmarks, as shown in Table 3. On GPQA-D, the nine models reach an intersection size of 90 for far samples and 92 for proximal samples. In contrast, on MGSM, the intersection is only 1 and 17, respectively. This stark difference indicates that some benchmarks (e.g., GPQA-D) contain many samples that all models consistently perceive as distant from or close to their training distributions, while others (e.g., MGSM) exhibit very little cross-model consensus, suggesting that perceived sample distance varies substantially across models.

4.5.2 INTRA-FAMILY CONSENSUS IS STRONGER THAN CROSS-FAMILY CONSENSUS

Across all nine models, as shown in Table 4, the average intersection size is 32.6 for far and 38.1 for near samples. Within the same family, consensus increases substantially: Llama reaches 83.9 (far) and 83.7 (near); Ministral3 reaches 79.5 (far) and 80.5 (near); and Qwen2.5 reaches 64.8 (far) and 68.8 (near). To rule out the possibility that higher intra-family consensus is due to fewer models, we select three models from different LLM families and compute their intersections. The resulting averages, 43.4 (far) and 48.2 (near), are lower than the intra-family values. This confirms that models from the same family, sharing similar training corpora, have highly consistent perceptions of distributional distance.

Model Set	Average Top-100	Average Bottom-100
All nine models	32.6	38.1
Qwen2.5 family	64.8	68.8
Llama family	83.9	83.7
Ministral3 family	79.5	80.5
Cross-family	**43.4**	**48.2**

Table 4: Average intersection sizes across model sets.

5 RELATED WORK

5.1 BENCHMARK CONTAMINATION

Benchmark contamination (Chen et al., 2025b; Cheng et al., 2025) is a prominent issue in fair evaluation of LLMs. Detection methods include overlap-based heuristics (e.g., n-gram or lexical matching), corpus tracing, and membership-style signals (Balloccu et al., 2024; Ippolito et al., 2023; Chang et al., 2023; Shi et al., 2023; Li & Flanigan, 2024; Duarte et al., 2024), as well as black-box analyses based on similarity, temporal, alignment, or distributional cues (Magar & Schwartz, 2022; Roberts et al., 2023; Yang et al., 2023; Oren et al., 2023; Riddell et al., 2024). Beyond detection, recent studies propose contamination-aware score adjustments (Wang et al., 2024a; Xu et al., 2025). While these approaches represent an important step, they primarily target explicit test-sample leakage. In practice, even when no test samples appear in the training corpus, differences in distributional distance can still lead to unfair evaluation. This phenomenon is orthogonal to explicit contamination yet remains underexplored.

5.2 OUT-OF-DISTRIBUTION GENERALIZATION

Out-of-distribution (OOD) generalization has been extensively studied (Liu et al., 2021; Yang et al., 2024), recognizing that performance on the training distribution alone is insufficient for assessing true generalization. Related work falls into two categories: benchmark construction and robustness evaluation (Yuan et al., 2023; Wang et al., 2023; Verma et al., 2024; Zhang et al., 2024b; Oh et al., 2025; Huang et al., 2026), and OOD detection with contamination-aware calibration (Zhang et al., 2024a; Podolskiy et al., 2021; Jin et al., 2022; Uppaal et al., 2023; Gangal et al., 2020). However, most OOD studies rely on access to training corpora or model retraining, and typically focus on a single model's ability to handle unseen distributions. Little systematic work has addressed the fairness of comparing models trained on different corpora. To fill this gap, we propose a distance-aware evaluation framework that integrates accuracy with distributional distance, enabling fairer cross-model comparisons without requiring access to training data.

6 CONCLUSION

We investigate LLM evaluation from a distributional distance perspective and show that benchmark accuracy can be strongly affected by the distribution distance between the inputs in test data and the training data, even without benchmark contamination. To alleviate this issue, we propose a two-dimension evaluation criterion consisting of the accuracy and the distributional distance approximated by PPL and Min-K%. With this evaluation criterion, we empirically re-examine nine LLMs across 13 benchmarks, suggesting that current benchmark practices may systematically overestimate generalization ability. Our findings highlight the importance of incorporating distributional distance into future LLM evaluation. A discussion of the corresponding limitations and directions for future work is provided in Appendix J.

Ethics Statement

We affirm that this work adheres to the ICLR Code of Ethics. We do not collect, use, or release any human subject data; all experiments rely exclusively on established public benchmarks and standard model outputs. No personally identifiable information is collected or inferred, and our analysis requires no access to private user data or proprietary training corpora. All datasets, models, and artifacts are handled in accordance with their respective licenses and terms of service.

Reproducibility Statement

We have made significant efforts to ensure the reproducibility of our findings. The source codes for implementing our proposed distance-aware evaluation criterion and running the conducted experiments are included in the supplementary material. Detailed information on compute infrastructure and benchmark configurations is given in Appendix A; data construction, model training hyperparameters, and k-NN distance definition of the controlled validation in Appendix C; PPL scoring protocol in Appendix D; and Min-K% scoring protocol in Appendix E. All evaluation results can be reproduced using only publicly available benchmarks and open-source models.

AI Disclosure

In this work, we used generative AI tools to assist with language polishing and code formatting. We did not use AI for any other research tasks, including hypothesis development, experimental design, method implementation, data analysis, or result interpretation. All scientific claims, methodology, experiments, and conclusions are the original work of the authors. All AI-generated or AI-assisted content was reviewed and verified by the authors, who take full responsibility for the final content of this paper.

References

Aaron Adcock, Aayushi Srivastava, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pande, Abhinav Pandey, Abhinav Sharma, Abhishek Kadian, Abhishek Kumawat, Adam Kelsey, et al. The llama 4 herd: Architecture, training, evaluation, and deployment notes. *arXiv preprint arXiv:2601.11659*, 2026.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, et al. Smollm2: When smol goes big–data-centric training of a small language model. *arXiv preprint arXiv:2502.02737*, 2025.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

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.

Maosong Cao, Kai Chen, Haodong Duan, Yixiao Fang, Tong Gao, Ge Jiaye, Mo Li, Hongwei Liu, Junnan Liu, Yuan Liu, et al. Opencompass: A universal evaluation platform for large language models. *arXiv preprint arXiv:2605.19276*, 2026.

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.

Kent Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. Speak, memory: An archaeology of books known to chatgpt/gpt-4. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7312–7327, 2023.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*, 2013.

Huashan Chen, Zhenyu Qi, Haotang Li, Hong Chen, Jinfu Chen, Kebin Peng, In Kee Kim, Kyu Hyung Lee, and Sen He. Beyond single llms: Enhanced code generation via multi-stage performance-guided llm orchestration. *arXiv preprint arXiv:2510.01379*, 2025a.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Simin Chen, Yiming Chen, Zexin Li, Yifan Jiang, Zhongwei Wan, Yixin He, Dezhi Ran, Tianle Gu, Haizhou Li, Tao Xie, et al. Benchmarking large language models under data contamination: A survey from static to dynamic evaluation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 10091–10109, 2025b.

Yuxing Cheng, Yi Chang, and Yuan Wu. A survey on data contamination for large language models. *arXiv preprint arXiv:2502.14425*, 2025.

Hyeong Kyu Choi, Maxim Khanov, Hongxin Wei, and Yixuan Li. How contaminated is your benchmark? quantifying dataset leakage in large language models with kernel divergence. *arXiv preprint arXiv:2502.00678*, 2025.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2368–2378, 2019.

André V Duarte, Xuandong Zhao, Arlindo L Oliveira, and Lei Li. De-cop: Detecting copyrighted content in language models training data. *arXiv preprint arXiv:2402.09910*, 2024.

Varun Gangal, Abhinav Arora, Arash Einolghozati, and Sonal Gupta. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7764–7771, 2020.

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

Yuling Gu, Oyvind Tafjord, Bailey Kuehl, Dany Haddad, Jesse Dodge, and Hannaneh Hajishirzi. Olmes: A standard for language model evaluations. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 5005–5033, 2025.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

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

Shulin Huang, Linyi Yang, Yan Song, Shawn Chen, Leyang Cui, Ziyu Wan, Qingcheng Zeng, Ying Wen, Kun Shao, Weinan Zhang, et al. Thinkbench: Dynamic out-of-distribution evaluation for robust llm reasoning. *Advances in Neural Information Processing Systems*, 38, 2026.

Daphne Ippolito, Florian Tramer, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher Choquette Choo, and Nicholas Carlini. Preventing generation of verbatim memorization in language models gives a false sense of privacy. In *Proceedings of the 16th International Natural Language Generation Conference*, pp. 28–53, 2023.

Di Jin, Shuyang Gao, Seokhwan Kim, Yang Liu, and Dilek Hakkani-Tür. Towards textual out-of-domain detection without in-domain labels. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1386–1395, 2022.

Hirokazu Kiyomaru, Issa Sugiura, Daisuke Kawahara, and Sadao Kurohashi. A comprehensive analysis of memorization in large language models. In *Proceedings of the 17th International Natural Language Generation Conference*, pp. 584–596, 2024.

Muhammed Yusuf Kocyigit, Eleftheria Briakou, Daniel Deutsch, Jiaming Luo, Colin Cherry, and Markus Freitag. Overestimation in llm evaluation: A controlled large-scale study on data contamination's impact on machine translation. *arXiv preprint arXiv:2501.18771*, 2025.

Changmao Li and Jeffrey Flanigan. Task contamination: Language models may not be few-shot anymore. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18471–18480, 2024.

Zi Liang, Liantong Yu, Zhang Shiyu, Qingqing Ye, and Haibo Hu. How much do large language model cheat on evaluation? benchmarking overestimation under the one-time-pad-based framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 37636–37644, 2026.

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

Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Inbal Magar and Roy Schwartz. Data contamination: From memorization to exploitation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 157–165, 2022.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Changdae Oh, Zhen Fang, Shawn Im, Xuefeng Du, and Yixuan Li. Understanding multimodal llms under distribution shifts: An information-theoretic approach. *arXiv preprint arXiv:2502.00577*, 2025.

Yonatan Oren, Nicole Meister, Niladri S Chatterji, Faisal Ladhak, and Tatsunori Hashimoto. Proving test set contamination in black-box language models. In *The Twelfth International Conference on Learning Representations*, 2023.

Alexander Podolskiy, Dmitry Lipin, Andrey Bout, Ekaterina Artemova, and Irina Piontkovskaya. Revisiting mahalanobis distance for transformer-based out-of-domain detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 13675–13682, 2021.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

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

Martin Riddell, Ansong Ni, and Arman Cohan. Quantifying contamination in evaluating code generation capabilities of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14116–14137, 2024.

Manley Roberts, Himanshu Thakur, Christine Herlihy, Colin White, and Samuel Dooley. Data contamination through the lens of time. *arXiv preprint arXiv:2310.10628*, 2023.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, et al. Language models are multilingual chain-of-thought reasoners. *arXiv preprint arXiv:2210.03057*, 2022.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*, 2023.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *International Conference on Learning Representations*, volume 2024, pp. 51826–51843, 2024.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

Yuan Sui, Yufei He, Nian Liu, Xiaoxin He, Kun Wang, and Bryan Hooi. Fidelis: Faithful reasoning in large language models for knowledge graph question answering. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8315–8330, 2025.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, 2023.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL `https://qwenlm.github.io/blog/qwen2.5/`.

Zhao Tian and Junjie Chen. Aligning requirement for large language model's code generation. *arXiv preprint arXiv:2509.01313*, 2025.

Rheeya Uppaal, Junjie Hu, and Yixuan Li. Is fine-tuning needed? pre-trained language models are near perfect for out-of-domain detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12813–12832, 2023.

Aayush Atul Verma, Amir Saeidi, Shamanthak Hegde, Ajay Therala, Fenil Denish Bardoliya, Nagaraju Machavarapu, Shri Ajay Kumar Ravindhiran, Srija Malyala, Agneet Chatterjee, Yezhou Yang, et al. Evaluating multimodal large language models across distribution shifts and augmentations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5314–5324, 2024.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, 2018.

Jindong Wang, Xixu Hu, Wenxin Hou, Hao Chen, Runkai Zheng, Yidong Wang, Linyi Yang, Haojun Huang, Wei Ye, Xiubo Geng, et al. On the robustness of chatgpt: An adversarial and out-of-distribution perspective. *arXiv preprint arXiv:2302.12095*, 2023.

Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. Generalization vs memorization: Tracing language models' capabilities back to pretraining data. *arXiv preprint arXiv:2407.14985*, 2024a.

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

Ziqi Wang, Xingzhou Lou, Meiqi Wu, Zhengqi Wen, and Junge Zhang. Calibration-aware policy optimization for reasoning llms. *arXiv preprint arXiv:2604.12632*, 2026.

Cheng Xu, Nan Yan, Shuhao Guan, Changhong Jin, Yuke Mei, Yibing Guo, and Tahar Kechadi. Dcr: Quantifying data contamination in llms evaluation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 23013–23031, 2025.

Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. Benchmarking benchmark leakage in large language models. *arXiv preprint arXiv:2404.18824*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Chen Yang, Junyi Li, Shulin Lan, Yingchao Wang, Hongyang Du, Congcheng Gong, Xingshan Yao, Dusit Niyato, and Liehuang Zhu. Detecting training data for large language models: A survey. *ACM Computing Surveys*, 2026.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, 132(12):5635–5662, 2024.

Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E Gonzalez, and Ion Stoica. Rethinking benchmark and contamination for language models with rephrased samples. *arXiv preprint arXiv:2311.04850*, 2023.

Lifan Yuan, Yangyi Chen, Ganqu Cui, Hongcheng Gao, Fangyuan Zou, Xingyi Cheng, Heng Ji, Zhiyuan Liu, and Maosong Sun. Revisiting out-of-distribution robustness in nlp: Benchmarks, analysis, and llms evaluations. *Advances in Neural Information Processing Systems*, 36:58478–58507, 2023.

Andi Zhang, Tim Z Xiao, Weiyang Liu, Robert Bamler, and Damon Wischik. Your finetuned large language model is already a powerful out-of-distribution detector. *arXiv preprint arXiv:2404.08679*, 2024a.

Xingxuan Zhang, Jiansheng Li, Wenjing Chu, Junjia Hai, Renzhe Xu, Yuqing Yang, Shikai Guan, Jiazheng Xu, and Peng Cui. On the out-of-distribution generalization of multimodal large language models. *arXiv preprint arXiv:2402.06599*, 2024b.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A Experimental Details

All experiments were conducted on a server equipped with 8 $\times$ NVIDIA A100 GPUs (80GB).

A.1 Benchmarks Overview and Details

We evaluate on 13 public benchmarks covering knowledge, reasoning, and code generation. All benchmarks are loaded using the official OpenCompass dataset configurations and releases. For some benchmarks, we use the subset or variant specified by OpenCompass (e.g., MATH-500 instead of the full MATH test set). We adopt the standard evaluation split for each benchmark and apply no additional filtering or deduplication beyond the preprocessing already implemented in OpenCompass.

Benchmark	Task type	Split	Instances (N)
MMLU	knowledge (MCQ)	test	14,042
MMLU-Pro	knowledge (MCQ)	test	12,032
ARC-Challenge	knowledge (MCQ)	test	2,590
GPQA-Diamond	knowledge (MCQ)	test	198
IFEval	knowledge (instruction-following)	test	541
GSM8K	reasoning (short answer)	test	1,319
MGSM (11 langs)	reasoning (multilingual short answer)	test	2,750
MATH	reasoning (short answer)	test	500
AIME2025	reasoning (short answer)	test	30
BBH	reasoning (mixed)	test	6,511
DROP	reasoning/RC (short answer)	validation	9,536
HumanEval	coding (unit tests)	test	164
MBPP	coding (unit tests)	test	974

Table 5: Benchmarks, evaluation splits, and instance counts used in this paper.

Model	Model name	Context length	System prompt
Qwen2.5-7B	`Qwen/Qwen2.5-7B-Instruct`	128K	Shared
Qwen2.5-14B	`Qwen/Qwen2.5-14B-Instruct`	128K	Shared
Qwen2.5-32B	`Qwen/Qwen2.5-32B-Instruct`	128K	Shared
Llama3-8B	`meta-llama/Meta-Llama-3-8B-Instruct`	8K	Shared
Llama3.1-8B	`meta-llama/Llama-3.1-8B-Instruct`	128K	Shared
Llama3.2-8B	`avans06/Meta-Llama-3.2-8B-Instruct`	128K	Shared
Ministral3-3B	`mistralai/Ministral-3-3B-Instruct-2512`	128K	Shared
Ministral3-8B	`mistralai/Ministral-3-8B-Instruct-2512`	128K	Shared
Ministral3-14B	`mistralai/Ministral-3-14B-Instruct-2512`	128K	Shared

Table 6: Open-source models used in this work.

A.1.1 Unified metric reporting

In Table 2, we report Acc for all benchmarks, computed using the official OpenCompass evaluator for each category: option-letter extraction with exact match for multiple-choice tasks; benchmark-specific postprocessing followed by exact match for short-answer tasks; unit-test-based pass@1 for coding benchmarks; and the official compliance evaluator for IFEval. A summary of benchmarks, splits, and instance counts is provided in Appendix Table 5.

A.1.2 Benchmark-specific preprocessing

We follow the official OpenCompass prompting (Cao et al., 2026), answer extraction, and evaluation routines:

- **Multiple-choice (MMLU, MMLU-Pro, ARC-C, GPQA-D).** Outputs are post-processed to extract the final option letter and matched to the reference label.
- **Math short-answer (GSM8K, MGSM, MATH-500, AIME2025).** Outputs are post-processed to extract the final numeric answer (language-aware for MGSM) and exact-matched after normalization.
- **Coding (HumanEval, MBPP).** Predictions are post-processed into executable code and evaluated by unit tests; Acc is reported as pass@1.
- **IFEval.** Evaluation follows the official IFEval compliance protocol in OpenCompass.
- **BBH.** Mixed multiple-choice and free-form subsets; evaluation follows OpenCompass BBH evaluators and aggregation rules.
- **DROP.** Evaluation follows the OpenCompass DROP evaluator.

Benchmark	Prompt template used in this paper
MMLU	Answer the following multiple choice question. The last line should be in the format `ANSWER: $LETTER`, where LETTER is one of A/B/C/D. Then present `{input}` and options `A) {A}`, `B) {B}`, `C) {C}`, `D) {D}`.
MMLU-Pro	Answer the following multiple choice question. The last line should be in the format `ANSWER: $LETTER`, where LETTER is one of option labels. Then present `Question: {question}` and `Options: {options_str}`.
ARC-Challenge	Answer the following multiple choice question. The last line should be exactly `Answer: $LETTER`. Then present `Question: {question}`, `{options_str}`, and end with `Answer:`.
GPQA-Diamond	Answer the following multiple choice question. The last line should be in the format `ANSWER: $LETTER`, where LETTER is one of A/B/C/D. Then present `{question}` and options A to D.
IFEval	Direct instruction format with `{prompt}` as input.
GSM8K	Few-shot chain-of-thought template; the final query is `Question: {question}`, followed by `Let's think step by step` and `Answer:`.
MGSM	Language-specific instruction templates (11 languages). Each template asks for reasoning steps and a final single-line answer in `Answer:` format, then inserts `{question}`.
MATH	Few-shot chain-of-thought template with math exemplars; the final query is `Problem: {problem}` followed by `Solution:`.
AIME2025	`{question}` followed by: `Please reason step by step, and put your final answer within \boxed{}.`
BBH	Task-specific few-shot hint template: `Follow the given examples and answer the question.` Then `{hint}`, then `Q: {input}`, then `A: Let's think step by step.`
DROP	Read-and-answer template: read passage and question from `{prompt}`, do not request missing information, and end with exactly one final line `Answer: <your answer>`.
HumanEval	Code-only instruction: read function signature and docstring in `{prompt}` and output only the function implementation code.
MBPP	Python coding instruction with task `{text}` and tests `{test_list}`; output only Python code in a single code block format.

Table 7: Benchmark-level prompt templates used in our evaluation pipeline.

A.2 Model Overview and Details

We evaluate 9 models in total. All evaluations use a unified prompting protocol with a batch size of 64. As shown in Table 6, Llama3-8B uses its native 8K context length, while all other models are evaluated with a unified 128K context length for consistency.

A.3 Benchmark Prompt Templates

To ensure reproducibility, Table 7 summarizes the prompt templates used in our evaluation pipeline across 13 benchmarks. Placeholders such as `{question}`, `{input}`, `{options_str}`, and `{prompt}` are populated with benchmark-specific fields.

B Controlled Validation on SmolLM2-135M

This appendix presents the controlled validation results on the SmolLM2-135M model (Allal et al., 2025), complementing the GPT-2 Small results shown in Figure 2 of the main text. The SmolLM2-135M model is trained from scratch on the same four corpora (WikiText-2, WikiText-103, One-Billion Word, and C4-tokenized-2B); the query-set construction and distance-definition protocol follow Appendix C, and the model-specific training hyperparameters are also reported there.

B.1 Results and Analysis

Figure 3 presents the Spearman correlation ρ between the two proxy metrics (PPL and Min-K%) and the true distributional distance $P(x; D)$ across the four training corpora. The results reveal a

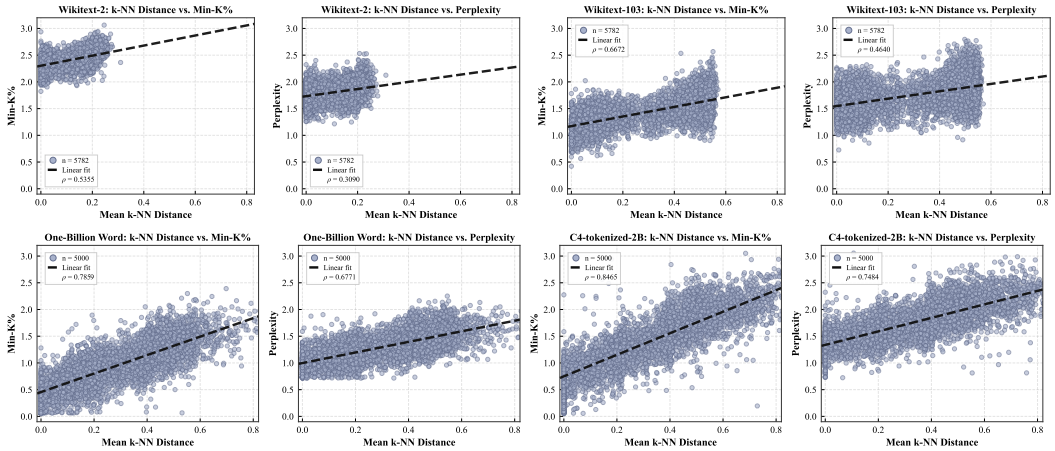


Figure 3: Controlled validation of distance proxies (PPL and Min-K%) against the distributional distance $P(x; D)$ on the SmolLM2-135M model over WikiText-2, WikiText-103, One-Billion Word, and C4-tokenized-2B.

consistent and clear trend: both proxy metrics exhibit stronger correlations with the true distance as the training corpus scales up, closely mirroring the pattern observed on GPT-2 Small.

On the smallest corpus (WikiText-2), PPL shows a weak correlation ($\rho = 0.3090$), while Min-K% demonstrates a moderate positive correlation ($\rho = 0.5355$). As the training corpus scales to WikiText-103, the correlation strengthens for both metrics: Min-K% rises to 0.6672 and PPL to 0.4640. The trend continues on the larger One-Billion Word corpus, where Min-K% reaches 0.7859 and PPL reaches 0.6771. The gap between the two proxies narrows compared to smaller corpora, suggesting that PPL benefits relatively more from increased training data volume, possibly because the model's overall likelihood estimates become more calibrated as exposure to distributional patterns grows. On the largest corpus (C4-tokenized-2B), both metrics achieve their strongest correlations: Min-K% at 0.8465 and PPL at 0.7484.

Throughout all four corpora, Min-K% consistently outperforms PPL by a substantial margin, which is consistent with the GPT-2 Small results and confirms that local low-confidence patterns are more informative about distributional distance than global perplexity. Furthermore, the SmolLM2-135M correlations broadly agree with those of GPT-2 Small reported in Figure 2, with both models showing the same monotonic improvement with corpus scale, which suggests that the proxy validation is robust across model architectures.

Taken together, these results on SmolLM2-135M independently validate the conclusions drawn from GPT-2 Small in the main text: (1) both PPL and Min-K% serve as effective proxies for distributional distance, with Min-K% consistently outperforming PPL; and (2) the correlation strengthens as the training corpus scales, confirming that in modern large-scale pretraining settings, these proxy metrics are reliable indicators of sample proximity to the training distribution.

C Controlled Validation Setup: Data, Models, and Distance

This appendix details the controlled validation used to verify the relationship between model-observable signals and sample-to-training distributional distance, covering both GPT-2 Small and SmolLM2-135M across all four corpora (WikiText-2, WikiText-103, One-Billion Word, and C4-tokenized-2B).

C.1 Data Construction

We use all four corpora (WikiText-2, WikiText-103, One-Billion Word, and C4-tokenized-2B). For corpora without an official test split, we hold out a portion of the data as the test set, while the remaining data serves as the training reference for distance computation. Since the test set is held

out from training by construction, a test-only query pool would consist entirely of far-from-training samples with high PPL and Min-K%, compressing the proxy-metric range and obscuring the true proxy–distance correlation. We therefore combine all test samples with an equal number of training samples drawn uniformly without replacement to obtain train-balanced query pools per corpus.

C.2 Model Training Details

We train GPT-2 Small (Radford et al., 2019) (124M) and SmolLM2-135M (Allal et al., 2025) (135M) on each of the four corpora using the standard causal language modeling objective, yielding eight controlled models. Both models share per-device batch size 16, gradient accumulation 2, learning rate 6×10^{-4}, bf16 mixed precision, and AdamW with a linear schedule. The only model-specific differences are the sequence length (1024 for GPT-2 Small, 2048 for SmolLM2-135M) and the tokenizer. Across corpora, we train for 15 epochs on WikiText-2 and 20 epochs on the three larger corpora to ensure convergence.

C.3 k-NN Distance Definition and Implementation

For each text sample, we compute an embedding from the corresponding model's hidden states (GPT-2 for the Figure 2 analysis and SmolLM2 for the Figure 3 analysis) using mean pooling over non-padding tokens, followed by L2 normalization:

$$\mathbf{e} = \frac{\mathrm{Pool}(\mathbf{H})}{\|\mathrm{Pool}(\mathbf{H})\|_2}$$

We then build a nearest-neighbor index over training-split embeddings and query it with the balanced query set. Distance is defined in cosine space:

$$d_{\cos}(\mathbf{e}_q, \mathbf{e}_t) = 1 - \langle \mathbf{e}_q, \mathbf{e}_t \rangle$$

where $\mathbf{e}_q$ is a query embedding and $\mathbf{e}_t$ is a training embedding. We use top-k retrieval with $k = 45$ and report the average cosine distance over the top-k neighbors as the distance statistic used in our controlled analysis.

D Perplexity (PPL) Scoring Protocol

This appendix specifies how PPL is computed in our experiments, covering prompt construction, truncation, and length control. All PPL computations are performed using the OpenCompass PPL inferencer with the same model and benchmark configurations as the main evaluation pipeline.

D.1 Task-wise PPL Input Construction

Let x_i denote the task input for sample i (e.g., question, passage-question pair, or code prompt). For all tasks, PPL is computed on the template-rendered input string under teacher-forced likelihood evaluation. No reference answer or completion is included in the input.

D.1.1 Multiple-choice tasks (MMLU, MMLU-Pro, ARC-C, GPQA-D)

For each input x_i, we construct the prompt as:

$$\pi_i = \mathrm{Template}(x_i)$$

where the template renders the question together with all candidate options. The PPL value computed on π_i serves as the proxy for distributional distance.

D.1.2 MATH AND SHORT-ANSWER TASKS (GSM8K, MGSM, MATH, AIME2025)

For each sample, we construct:

$$\pi_i = \text{Template}(x_i)$$

where the template renders the problem statement or mathematical expression.

D.1.3 CODE TASKS (HUMANEVAL, MBPP)

PPL scoring uses code-oriented templates that combine programming instructions or signatures:

$$\pi_i = \text{Template}(\text{code_prompt}_i)$$

D.1.4 INSTRUCTION-FOLLOWING TASKS (IFEVAL)

For IFEval, we use the instruction input as the prompt field:

$$\pi_i = \text{Template}(x_i)$$

where x_i is the instruction prompt.

D.1.5 MIXED REASONING TASKS (BBH)

BBH contains both multiple-choice-like and free-form subsets. For each sample, we construct the subset-specific template on the input x_i:

$$\pi_i = \text{Template}(x_i)$$

D.1.6 READING COMPREHENSION (DROP)

We follow the dataset-specific prompt format that combines passage and question context:

$$\pi_i = \text{Template}(\text{passage}_i, \text{question}_i)$$

D.2 ANSWER INCLUSION, TRUNCATION, AND LENGTH CONTROL

D.2.1 PPL SCORING WITHOUT ANSWER INCLUSION

Our PPL scoring uses only the prompt, not the answer. For multiple-choice tasks, the candidate options are treated as part of the prompt (i.e., the model receives the question along with all options). For short-answer, reading comprehension, and code tasks, the scoring prompt consists solely of the context and question (or task description), without any reference answer or completion. This ensures that the resulting PPL reflects the model's uncertainty about the input itself, uncontaminated by answer-specific likelihoods. By excluding the answer, we isolate the effect of input distributional distance from any potential answer leakage, allowing us to investigate how proximity of the input alone can bias evaluation even in the complete absence of benchmark contamination.

D.2.2 TRUNCATION POLICY UNDER CONTEXT CONSTRAINTS

Given the model's maximum sequence length, the prompt length is checked after template rendering. If the prompt exceeds the limit, OpenCompass iteratively truncates in-context demonstrations until the prompt fits. The task query and answer-bearing segment are preserved whenever possible, while auxiliary in-context exemplars are reduced first.

D.2.3 LENGTH ALIGNMENT WITHIN A SAMPLE

For multiple-choice scoring, all candidates for the same sample share the same context scaffold and differ only in the candidate filling at the answer slot. This design ensures within-sample comparability of candidate PPL values.

E MIN-K% SCORING PROTOCOL

This appendix details the Min-K% calculation method, including the extraction of token-level log-probabilities, their byte-level replication for cross-tokenizer comparison, and task-specific prompt construction. All Min-K% computations are performed using a single forward pass of the model on the input prompt, with the same model configurations as the main evaluation pipeline. Note that Min-K% is computed exclusively on the input prompt, without any reference answer or completion.

E.1 LOG-PROBABILITY EXTRACTION AND BYTE-LEVEL REPLICATION

Given a token sequence $x_1, x_2, \ldots, x_N$, the model outputs unnormalized logits $\ell_t \in \mathbb{R}^{|V|}$ at each position t (where $|V|$ is the vocabulary size). The log-probability of the target token x_t at position t is defined as:

$$\log p(x_t \mid x_{<t}) = \ell_t[x_t] - \log \sum_{v \in V} \exp(\ell_t[v])$$

For an input sequence, tokenization yields a token sequence $x = (x_1, x_2, \ldots, x_N)$ (where x_1 is typically a special start token). After a single forward pass, we extract the log-probabilities for all positions except the first token:

$$\mathcal{L} = \{\log p(x_i \mid x_1, \ldots, x_{i-1})\}_{i=2}^{N}$$

Each $\log p(x_i \mid x_{<i})$ is the conditional log-probability of token x_i given its preceding context.

To enable cross-tokenizer comparison (see §**??**), we replicate each token's negative log-likelihood by the number of bytes b_i spanned by token x_i. Let $B(x) = \sum_{i=2}^{N} b_i$ denote the total byte length of the prompt body, and let $t(j)$ index the source token of byte j. The token-level negative log-likelihoods are then expanded to a byte-level sequence:

$$\Big(\underbrace{-\log p(x_2 \mid x_{<2}), \ldots}_{b_2}, \underbrace{-\log p(x_3 \mid x_{<3}), \ldots}_{b_3}, \ \ldots, \ \underbrace{-\log p(x_N \mid x_{<N}), \ldots}_{b_N}\Big).$$

E.2 DEFINITION OF THE MIN-K% SCORE

Sort the byte-level negative log-likelihoods in ascending order and select the bottom $K\%$ as the hardest-to-predict subset.

The core idea of Min-K% is to select the $k \times 100\%$ bytes with the largest negative log-likelihoods (the "hardest-to-predict" subset) and compute their average. For a given percentage parameter $k \in (0, 1)$, let $\text{Min-K\%}(x)$ denote the $\lceil k \cdot B(x) \rceil$ bytes with the largest negative log-likelihoods in the byte-level replicated sequence. The Min-K% score is then defined as:

$$\text{Min-K\%PROB}(x) = \frac{1}{E} \sum_{j \in \text{Min-K\%}(x)} \big(-\log p(x_{t(j)} \mid x_{<t(j)})\big),$$

where $E = \lceil k \cdot B(x) \rceil$ is the number of bytes in the Min-K% subset and $t(j)$ indexes the source token of byte j.

In the main text, to facilitate comparison with PPL, we report this score directly: $\text{Dist}_{\text{Min-K\%}}(x, M) = \text{Min-K\%PROB}(x)$. A larger value indicates that more bytes of x fall outside the model's training distribution.

E.3 Prompt Construction and Task Adaptation

Min-K% scoring is computed solely on the input prompt, without any reference answer or completion. Prompt construction for each task follows the same protocol as the PPL scoring described in Section D:

- **Multiple-choice tasks (MMLU, MMLU-Pro, ARC-C, GPQA-D)**: the prompt includes the question and all candidate options.
- **Math and short-answer tasks (GSM8K, MGSM, MATH, AIME2025)**: the prompt contains only the problem statement.
- **Code tasks (HumanEval, MBPP)**: the prompt contains the code signature or task description.
- **Instruction-following tasks (IFEval)**: the prompt is the instruction text.
- **Mixed reasoning tasks (BBH)**: the subset-specific input text is used.
- **Reading comprehension (DROP)**: the prompt includes the passage and the question.

If the prompt exceeds the model's maximum sequence length, the truncation strategy described in Section D.2 is applied (in-context demonstrations are truncated first while preserving the core query).

F Discussion on the Choice of k in k-NN for True Distance Calculation

In §4.2 we computed the true distributional distance d_i for each test sample as the average cosine distance to its k nearest neighbors in the training set. The parameter k controls the locality of this estimate. To select an appropriate value of k for our main experiments, we conducted a sensitivity analysis on both the small-scale (WikiText-2) and large-scale (WikiText-103) controlled setups, varying k over a wide range and measuring the Spearman correlation between the true distance d_i and the proxy metric PPL, which serves as an inverse indicator of distance.

F.1 Results

Figure 4 illustrates the correlation ρ as a function of k. On WikiText-2 (small training corpus), the correlation remains relatively flat across different values of k. The highest correlation is observed at $k = 9$ ($\rho = 0.1725$), while at $k = 45$ the correlation is still comparable ($\rho = 0.1692$). In contrast, on WikiText-103 (large training corpus), the correlation varies considerably with k. The peak occurs at $k = 45$ ($\rho = 0.4961$), whereas at $k = 9$ the correlation is markedly lower ($\rho = 0.4594$).

F.2 Interpretation and choice of k

We attribute this discrepancy to the size of the training corpus. In a larger training corpus such as WikiText-103, a larger value of k provides a more stable and representative estimate of the local neighborhood density, because there are more relevant samples over which to average. In a small corpus such as WikiText-2, increasing k rapidly includes points that are far apart, rendering the distance estimate less discriminative. Nevertheless, because the overall correlation is low regardless of the specific value of k, the exact choice matters less in this regime.

For real-world large language models, the pretraining corpus is orders of magnitude larger than WikiText-103. Consequently, a larger value of k is likely to be more appropriate. Based on the clear peak at $k = 45$ observed on the larger corpus, and to ensure a consistent and principled setting across all analyses, we adopt $k = 45$ as the default value for all reported distance correlation experiments in this paper. All experiments involving the true distributional distance—including the correlation plots and the controlled validation—are performed with this setting.

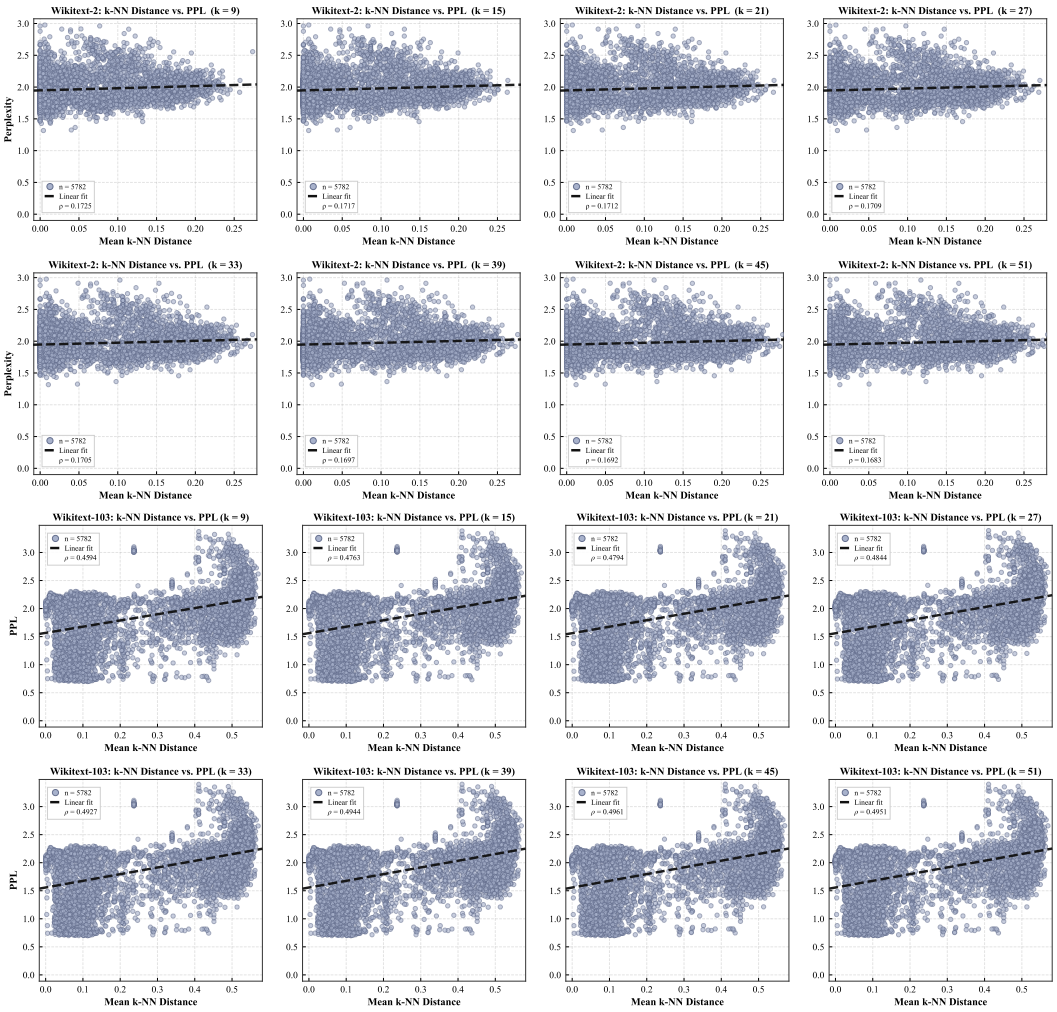


Figure 4: Spearman correlation between PPL and distributional distance for different k in k-NN on WikiText-2 and WikiText-103.

G Discussion on the Choice of K in Min-K%

The Min-K% score depends on the hyperparameter K, which determines the percentage of the lowest-probability bytes used to compute the average. To analyze its effect, we evaluate the correlation between the Min-K% score and the true distributional distance (mean k-NN cosine distance) under the same controlled setup described in §4.2. We vary K over $\{10, 20, 30, 40, 50, 60, 70, 80\}$ and compute the Spearman correlation coefficient for both the small-scale (WikiText-2) and large-scale (WikiText-103) corpora.

G.1 Results

Figure 5 presents the Spearman correlation as a function of K. On both corpora, the strongest correlation occurs at $K = 10$: $\rho = 0.5533$ on WikiText-2 and $\rho = 0.6823$ on WikiText-103. As K increases beyond 10, the correlation generally decreases, albeit with some minor fluctuations, and $K = 10$ consistently yields the highest correlation. This indicates that Min-K% is most effective at reflecting the true distance to the training distribution when it focuses on a very small fraction of the most uncertain bytes. Therefore, we set $K = 10$ as the default value in all experiments.

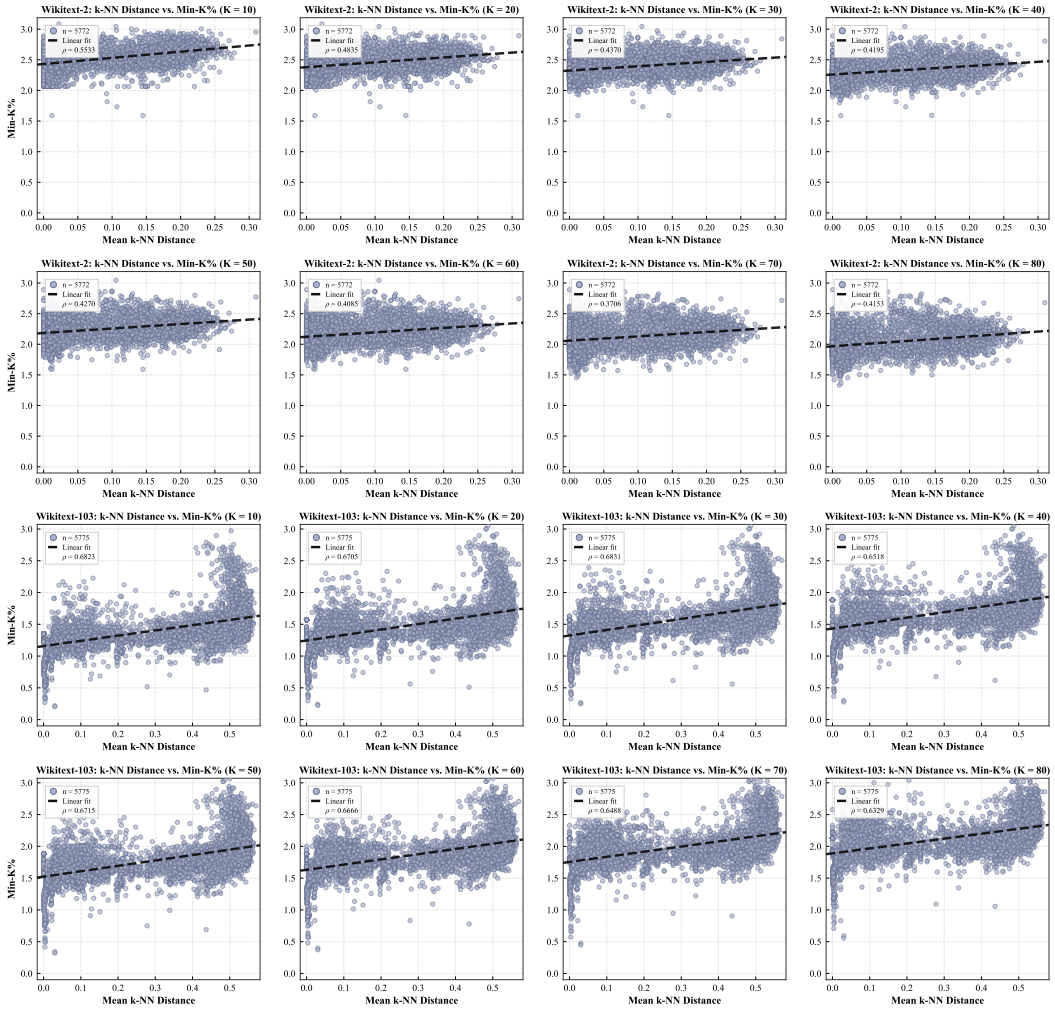


Figure 5: Spearman correlation between Min-K% and distributional distance for different K on WikiText-2 and WikiText-103.

G.2 Sample exclusion for very short sequences

We note that Min-K% cannot be computed for input sequences consisting of a single token, because the extraction of log-probabilities excludes the first token (typically a special start token) and at least one remaining token is required. Consequently, after byte-level replication there are no bytes left to average over. The number of valid samples for Min-K% is therefore slightly smaller than that for PPL, which can be computed on any non-empty sequence. In our controlled experiments, all such single-token samples are removed before calculating correlations. The same exclusion rule applies to all K values, ensuring fair comparison across different settings.

H Partial Correlation Analysis Controlling for General Model Capability

A potential confounding factor in our main analysis is general model capability: a stronger model assigns higher likelihood to almost any well-formed input and tends to answer more questions correctly, which would itself produce a negative accuracy–distance correlation even in the absence of any true distance effect. To rule out this confound, we re-estimate the correlations while partialling out each model's general capability.

H.1 Control Variable: LongBench Pro Loss

We select **LongBench Pro**[3] as a neutral held-out dataset. LongBench Pro was released after the training data cutoff dates of all evaluated models, ensuring it is not included in any model's training corpus. For each of the nine models, we compute the average negative log-likelihood loss on LongBench Pro and use this as the control variable Z for general model capability. A lower loss on LongBench Pro indicates stronger general capability.

H.2 Partial Correlation Computation

Let X denote Min-K% (larger values indicate greater distance from the training distribution) and Y denote accuracy. We compute the first-order partial correlation $\rho_{XY \cdot Z}$ between X and Y while controlling for Z:

$$\rho_{XY \cdot Z} = \frac{\rho_{XY} - \rho_{XZ}\rho_{YZ}}{\sqrt{(1 - \rho_{XZ}^2)(1 - \rho_{YZ}^2)}}$$

where ρ_{AB} denotes the Pearson correlation between variables A and B.

H.3 Results

Benchmark	ρ_{XY}	$\rho_{XY \cdot Z}$	$\rho_{XY} - \rho_{XY \cdot Z}$
HumanEval	−0.9664	−0.9492	−0.0172
MMLU-Pro	−0.9500	−0.9487	−0.0013
MMLU	−0.8619	−0.8107	−0.0512
AIME2025	−0.8330	−0.7928	−0.0402
MATH	−0.8201	−0.7940	−0.0261
ARC-C	−0.8167	−0.6755	−0.1411
DROP	−0.7280	−0.7128	−0.0152
GPQA-D	−0.6946	−0.8048	+0.1102
GSM8K	−0.5607	−0.5599	−0.0007
MBPP	−0.5833	−0.4208	−0.1626
MGSM	−0.5941	+0.0667	−0.6608
BBH	−0.2762	−0.1215	−0.1546
IFEval	+0.3013	+0.7074	−0.4062
Average	**−0.6458**	**−0.6309**	**−0.0149**

Table 8: Partial correlation between Min-K% and accuracy, controlling for general capability.

Table 8 reports the partial correlation results across all 13 benchmarks. After controlling for general capability, 11 of 13 benchmarks retain negative partial correlations, with an average of -0.6309—only marginally attenuated from the raw average of -0.6458. This attenuation suggests that general capability accounts for a small share of the observed correlation, while the bulk of the effect can be attributed to distributional distance itself.

Two partial-correlation estimates move against the general pattern. On GPQA-D, $\rho_{XY \cdot Z}$ is amplified from -0.6946 to -0.8048 (difference $+0.1102$). The correlation between Y and the control LongBench Pro loss is only $r(Y, Z) = -0.083$, indicating that the loss is not an effective confounder for GPQA-D; with little variance explained by Z to remove, partialling it out leaves the residual correlation largely intact. On MGSM, $\rho_{XY \cdot Z}$ flips sign from -0.5941 to $+0.0667$. Here X and Z show moderate collinearity (VIF $= 3.89$), so the denominator $\sqrt{(1 - \rho_{XZ}^2)(1 - \rho_{YZ}^2)}$ becomes small and the estimator becomes highly sensitive to noise; the $+0.0667$ reflects numerical instability under high collinearity rather than a genuine reversal and should not be overinterpreted.

Taken together, these results indicate that distributional distance contributes to evaluation accuracy as an independent factor beyond general model capability, consistent with our central claim that distributional distance inflates evaluation accuracy.

Note on statistics. ρ_{XY} and $\rho_{XY \cdot Z}$ are Pearson correlations computed directly on the nine-model observations used throughout this paper, and may differ slightly from the bootstrap Spearman means reported in Table 1 due to rank vs. linear statistics. All entries share the same nine-model sample as Table 2.

I Re-Ranking Analysis with the Generalization Score

To provide concrete evidence that accuracy-only rankings can be misleading, we apply the Generalization Score (GS) defined in §3 to re-rank the nine models on each of the 13 benchmarks.

[3] https://huggingface.co/datasets/caskcsg/LongBench-Pro

Model	MMLU		MMLU-Pro		ARC-C		GPQA-D		IFEval		GSM8K		MGSM	
	Acc	GS	Acc	GS	Acc	GS	Acc	GS	Acc	GS	Acc	GS	Acc	GS
Qwen2.5-32B	1	9	1	9	1	9	4	9	2	5	6	4	1	8
Ministral3-14B	2	6	2	8	2	7	2	7	7	9	2	9	3	4
Qwen2.5-14B	3	8	4	7	3	8	3	8	1	6	7	3	2	9
Ministral3-8B	4	5	3	6	5	4	1	6	8	8	1	8	4	3
Qwen2.5-7B	5	7	6	5	4	6	7	4	5	4	9	1	9	5
Ministral3-3B	6	4	5	4	9	5	5	2	9	7	8	7	7	1
Llama3-8B	7	1	9	1	6	1	6	1	6	1	3	2	8	2
Llama3.1-8B	8	2	7	2	8	3	8	3	4	3	4	5	6	7
Llama3.2-8B	9	3	8	3	7	2	9	5	3	2	5	6	5	6

Model	MATH		AIME2025		BBH		DROP		HumanEval		MBPP	
	Acc	GS	Acc	GS	Acc	GS	Acc	GS	Acc	GS	Acc	GS
Qwen2.5-32B	1	9	3	5	1	6	1	9	1	9	1	6
Ministral3-14B	2	3	2	1	2	9	9	7	3	6	4	7
Ministral3-8B	3	2	4	3	7	8	4	5	5	5	3	5
Qwen2.5-7B	4	7	5	4	9	4	2	6	4	7	6	8
Qwen2.5-14B	5	8	1	6	3	7	3	8	2	8	2	9
Ministral3-3B	6	1	6	2	8	5	8	4	6	4	9	4
Llama3.1-8B	7	4	8	8	6	3	5	2	8	3	7	3
Llama3-8B	8	5	7	7	4	1	7	1	9	1	8	1
Llama3.2-8B	9	6	9	9	5	2	6	3	7	2	5	2

Table 9: Accuracy rank vs. GS rank for each model across 13 benchmarks. GS = Acc $\times$ normalized Min-K%. 1 = best.

I.1 Methodology

Following the definition in §3, we use Min-K% as the distance proxy. For each benchmark b, we first collect the Min-K% values $\{\text{Min-K\%}(b, m)\}_{m=1}^{9}$ across all nine models. We then apply min-max normalization per benchmark to obtain $\widetilde{\text{Dist}}(b, m)$:

$$\widetilde{\text{Dist}}(b, m) = \frac{\text{Min-K\%}(b, m) - \min_m \text{Min-K\%}(b, m)}{\max_m \text{Min-K\%}(b, m) - \min_m \text{Min-K\%}(b, m)}, \tag{1}$$

which maps each model on that benchmark to $[0, 1]$, where a value closer to 1 indicates the benchmark inputs are farther from the model's training distribution. We then compute $\text{GS}(b, m) = \text{Acc}(b, m) \times \widetilde{\text{Dist}}(b, m)$. A higher GS indicates that a model's accuracy is more likely derived from inputs farther from its training distribution, providing stronger evidence of generalization.

I.2 Results

Table 9 presents the accuracy rank and GS rank for each model on all 13 benchmarks. GS $= \text{Acc} \times \widetilde{\text{Dist}}$ with $\widetilde{\text{Dist}}$ being the per-benchmark normalized Min-K%. The re-ranking reveals substantial shifts: models closer to training distribution (lower Min-K%) see their rankings drop under GS, while models farther from training distribution improve. For instance, on MMLU, Qwen2.5-32B ranks 1st by accuracy but drops to 9th under GS due to its extremely low Min-K% (2.62), whereas Llama3-8B rises from 7th to 1st with the highest Min-K% (3.29).

Table 10 further reveals a striking misalignment between accuracy-only rankings and our GS-based rankings that account for distributional distance. Across all 13 benchmarks, the Spearman correlation averages $\rho = -0.4346$, indicating substantial disagreement. Critically, 66.5% of model pairs undergo rank reversal, demonstrating that ignoring distributional distance can systematically misjudge which model truly generalizes better.

Benchmark	Spearman ρ (GS vs. Acc)	Rank reversal pairs	Reversal rate
HumanEval	-0.9667	34/36	94.4%
MMLU-Pro	-0.9500	33/36	91.7%
MMLU	-0.8500	30/36	83.3%
ARC-C	-0.7500	28/36	77.8%
GSM8K	-0.5500	25/36	69.4%
DROP	-0.5000	25/36	69.4%
GPQA-D	-0.5333	22/36	61.1%
MGSM	-0.5833	25/36	69.4%
MBPP	-0.5833	25/36	69.4%
BBH	-0.2667	20/36	55.6%
MATH	-0.1333	19/36	52.8%
IFEval	0.4167	15/36	41.7%
AIME2025	0.6000	10/36	27.8%
Average	**-0.4346**	**311/468**	**66.5%**

Table 10: GS-based vs. accuracy-only ranking: Spearman ρ and rank reversal statistics across benchmarks.

Notably, some benchmarks exhibit particularly high reversal rates. On HumanEval, 94.4% of pairs (34/36) change order under GS ranking. On MMLU-Pro, the reversal rate is 91.7% (33/36). On the other hand, IFEval and AIME2025 show the lowest reversal rates (41.7% and 27.8%, respectively). While these two benchmarks exhibit positive Spearman correlations ($\rho = 0.4167$ and $\rho = 0.6000$, respectively), this deviation is attributable to their limited sample sizes (IFEval: 541 instances; AIME2025: 30 instances), which introduces high variance in Min-K% estimation. Additionally, AIME2025 exhibits extremely low accuracy across all models (ranging from 0% to 16.67%), compressing the accuracy scale and making rankings sensitive to minor fluctuations. Nevertheless, these exceptions do not fundamentally undermine our conclusions.

These findings underscore the urgency of adopting distance-aware evaluation: relying solely on accuracy can misjudge which model truly generalizes better, with the error direction biased in favor of models whose training distributions happen to be closer to the test inputs. Therefore, we call for incorporating distributional distance in model evaluation.

J Limitations and Future Work

J.1 Limitations

Our work provides a distributional distance perspective for LLM evaluation and introduces a distance-aware two-dimensional evaluation criterion. Nevertheless, several limitations remain. First, the proposed proxy metrics require access to model logits or token-level probabilities. While many open-source models provide such outputs, some proprietary APIs expose only final answers, making our framework inapplicable without additional assumptions. Second, due to computational constraints, our controlled validation employs GPT-2 Small on WikiText-2 and WikiText-103; modern pretraining corpora are orders of magnitude larger, and it remains unclear whether the observed correlation patterns saturate or evolve at that scale. Third, our experiments cover three model families (Qwen2.5, Llama3, Ministral3) with up to 32B parameters. The observed scaling trends and cross-benchmark heterogeneity may not generalize to ultra-large models or other settings.

J.2 Future Work

Future work will address these limitations. First, we plan to scale up validation by training larger models on larger corpora, systematically characterizing the relationship between proxy metrics and true distributional distance. Second, we will explore logit-free alternatives, such as consistency across multiple generated samples or model self-explanations as proxy signals, though these remain exploratory and depend on API openness. Third, we intend to broaden the scope to more model families, larger parameter scales, and non-English benchmarks, testing the generality of our findings.