The Measure of All Measures: Quantifying LLM Benchmark Quality

Anonymous Author(s)
Affiliation
Address
email

Abstract

The development of Large Language Models (LLMs) is advancing at a fast pace, and choosing the right benchmarks has become central to understanding and characterizing real progress. The community now faces an abundance of benchmarks. We often lack a systematic way to tell which benchmark is harder, which provides cleaner separations between models, or which offers sufficient topical and linguistic coverage for a developer's use case. This paper proposes a principled and quantitative answer. We introduce three metrics for benchmark quality, *hardness*, *separability*, and *diversity*, each with explicit mathematical definitions suitable for automated evaluation pipelines. We further derive a difficulty–aware leaderboard index that rewards solving genuinely hard items. We instantiate the framework across math, coding, knowledge, instruction following and agentic evaluation suites. Together, these metrics enable systematic comparison and selection of the right benchmarks for model developers.

1 Introduction

LLM development is extraordinarily fast, and picking the right benchmarks to track is now core to understanding, comparing, and steering progress of LLMs. The ecosystem of benchmarks has exploded across capabilities, spanning knowledge (MMLU [Hendrycks et al., 2020], MMLU-Pro [Wang et al., 2024], GPQA [Rein et al., 2024], SimpleQA [Wei et al., 2024], HLE [Phan et al., 2025], Gaokao 2023 [Zhang et al., 2023]), math (AIME 2024/2025 [AIME, 2025], HMMT Feb25 [Balunović et al., 2025], Math 500 [Hendrycks et al., 2021], MathOdyssey [Fang et al., 2025], OlympiadBench [He et al., 2024a]), instruction-following (ComplexBench [Wen et al., 2024], FollowBench [Jiang et al., 2023], IF-Bench [Pyatkin et al., 2025], IF-Eval [Zhou et al., 2023], InfoBench [Qin et al., 2024], MultiChallenge [Sirdeshmukh et al., 2025], Multi-IF [He et al., 2024b]), agent tasks (ACEBench [Chen et al., 2025], BFCL [Patil et al.], ComplexFuncBench [Zhong et al., 2025], DrafterBench [Li et al., 2025], MultiChallenge [Sirdeshmukh et al., 2025], NexusBench [team, 2024], τ-Bench [Yao et al., 2024], τ^2-Bench [Barres et al., 2025], ToolSandbox [Lu et al., 2024]), and code (LiveCodeBench v5/v6 [Jain et al., 2024], OJBench [Wang et al., 2025], Terminal-Bench [Team, 2025], SWE-bench [Jimenez et al., 2023]).

Earlier broad suites such as BIG-bench [Srivastava et al., 2023], GSM8K [Cobbe et al., 2021], MATH [Hendrycks et al., 2021] and HumanEval [Chen et al., 2021] etc. established the foundation. The growth of new benchmarks in the recent years makes it difficult to determine which benchmarks are genuinely hard, which provide clean separability among models, and which ensure sufficient diversity. Furthermore, recent work has revealed significant shortcomings in measurement quality across existing benchmarks, e.g. inconsistent leaderboard rankings [Zhou et al., 2025] and poor model separability among top performers [Ni et al., 2024]. Our work introduces a set of quantitative

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

criteria—hardness, separability, and diversity—for systematic comparison across this expanding ecosystem.

- **Hardness**—evaluating each prompt's difficulty for differentiating models, quantified using established psychometric modeling through Item Response Theory (IRT) [Verhelst and Glas, 1995, Cai et al., 2016].
- **Separability—**capturing how well a benchmark spreads model scores (between-model variance) relative to sampling noise (within-model variance), evaluated by adjacent ranking stability.
- **Diversity**—ensuring broad semantic coverage among prompts, leveraging embedding-based dispersion measures [Zhang et al., 2019].

We conducted experiments on 34 benchmarks and 12 recent LLMs, including GPT-4O-MINI, GPT-4O, GPT-4.1, O3-HIGH, O4-MINI-HIGH, DEEPSEEK-V3, DEEPSEEK-R1, CLAUDE 4 SONNET, CLAUDE 4 SONNET (think), KIMI-K2-INSTRUCT, QWEN3-235B-THINKING, and QWEN3-235B-INSTRUCT. We calculated the hardness, separability and diversity score for each benchmark. We also proposed a new method which incorporates difficulty for model ranking and produced a new LLM leaderboard based on difficulty-aware ranking method.

2 Related Work

Metrics for Benchmarks Evaluation. Recent work has developed various metrics to assess benchmark quality across multiple dimensions. For hardness and difficulty measurement, [Zhou et al., 2025] applied PSN-IRT to analyze 11 LLM benchmarks, while [Hempstead et al., 2004] used Item Response Theory to select efficient benchmark subsets. Separability metrics have been formalized through signal-to-noise frameworks [Heineman et al., 2025] and confidence interval analysis in Arena-Hard-Auto [Li et al., 2024]. Diversity measures have been explored through comprehensive embedding evaluation frameworks [Zhang et al., 2019, Muennighoff et al., 2022] and text diversity measurement tools [Shaib et al., 2024]. Some optimization approaches have shown promise for quality-diversity balancing in various domains [Liu et al., 2025, Shypula et al., 2025], though their application to benchmark curation remains underexplored. However, most existing approaches address individual quality dimensions in isolation rather than providing unified optimization frameworks.

Benchmarking Benchmarks. Systematic analyses have revealed significant limitations in current LLM benchmarks. [McIntosh et al., 2025] comprehensively evaluated 23 state-of-the-art benchmarks, uncovering biases, measurement inconsistencies, and cultural oversight. Data contamination has emerged as a critical concern, with [Sainz et al., 2023, Balloccu et al., 2024] demonstrating that benchmark leakage leads to unreliable performance estimation. Benchmark reconstruction approaches like MixEval [Ni et al., 2024] achieved high correlation with human preferences through strategic benchmark mixing, while Arena-Hard [Li et al., 2024] introduced automated curation from crowd-sourced data. Dynamic evaluation methods have been proposed to address benchmark saturation [Kiela et al., 2021, White et al., 2024], with studies showing that traditional benchmarks like MMLU suffer from rapid ceiling effects [Hendrycks et al., 2020]. Despite these advances, previous work lacks proactive design principles, and limited theoretical foundations that fail to jointly optimize multiple benchmark quality criteria.

3 Methodology

3.1 Preliminaries

Let $\mathcal{B} = \{1, \ldots, N\}$ be the prompts and $\mathcal{M} = \{1, \ldots, M\}$ the reference models (humans may be included). Denote by $a_{mi} \in [0, 1]$ the accuracy of model m on prompt i and by $s_m = \frac{1}{N} \sum_i a_{mi}$ its mean score. Unless otherwise stated, expectations are taken over the uniform distribution on prompts.

3.2 Hardness Metric

We derive hardness for each prompt from Item Response Theory (IRT). The **one–parameter logistic (1PL)** model is a principled way to place prompts and models on a common latent scale. For model

m on prompt i:

$$P(a_{mi} = 1) = \sigma(\theta_m - \beta_i), \qquad \sigma(x) = \frac{1}{1 + e^{-x}}. \tag{1}$$

- θ_m — *ability* of model m.
- β_i — *difficulty* of prompt i (what we want).

Higher β_i implies a lower success probability for a fixed θ_m. Given the binary response matrix $\boldsymbol{A} = [a_{mi}]$ we can fit Equation (1) directly to get a numeric hardness score $\hat{\beta}_i$ for every prompt. We average the hardness score in the same benchmark to derive the hardness score for each benchmark. It also gives a scalar θ_m for each model m as its capability metric. We fit Equation (1) on each category to derive per-category LLM ranking.

3.3 Separability Metric

Intuitively, a good benchmark spreads model scores widely while keeping each model's sampling noise small. We define **adjacent ranking stability** specifically as a measure of separability.

Assume the M models are sorted by their scores such that $s_1 \geq s_2 \geq \cdots \geq s_M$. For each pair (m, n), the probability of a rank reversal under binomial uncertainty is

$$P^{\text{flip}}_{mn} = \Phi\Big(-\frac{|s_m - s_n|}{\sqrt{\sigma^2_{W,m} + \sigma^2_{W,n}}}\Big), \tag{2}$$

where Φ is the standard normal CDF and $\sigma^2_{W,m}$ is the binomial noise

$$\sigma^2_{W,m} = \frac{s_m\,(1 - s_m)}{N}. \tag{3}$$

Increasing N drives $\sigma^2_{W,m} \to 0$ but at higher annotation cost. We define the **Adjacent Ranking Stability** (R_{adj}) as:

$$R_{\text{adj}} = 1 - \frac{1}{M-1}\sum_{m=1}^{M-1} P^{\text{flip}}_{m,m+1}$$

where $P^{\text{flip}}_{m,m+1}$ is the probability of a rank reversal between the model at rank m and the model at rank $m + 1$.

3.4 Diversity Metric

Diversity ensures that solving the benchmark demands breadth rather than narrow skill, and that it is not a simple permutation of existing prompts so that the dependency is strong between prompts. Let $f(\cdot)$ be a sentence or code encoder and $\boldsymbol{e}_i = f(i)$. We define the semantic dispersion as

$$C_{\text{sem}} = \frac{2}{N(N-1)}\sum_{i<j}\big[1 - \cos(\boldsymbol{e}_i, \boldsymbol{e}_j)\big] \in [0, 1]. \tag{4}$$

Values near 1 indicate a wide semantic spread and good coverage around diverse topics.

4 Experiments

We present the evaluation results in Table 1. For further detailed discussion and the difficulty-aware leaderboard, please refer to Appendix A.

Hardness. Hardness Analysis. Among the five core capabilities we evaluate, knowledge and instruction following exhibit the largest performance gaps between the hardest and easiest datasets, with gaps of 3.401 and 3.129 respectively. In contrast, agent and code capabilities show relatively consistent difficulty levels across datasets. Notably, many widely-used benchmarks such as IF-Eval, Math 500, and MMLU appear to be too easy for current state-of-the-art LLMs. Consequently, evaluation results on these benchmarks may fail to adequately expose model limitations, potentially hindering pushing forward the frontier. More detailed hardness analysis is in Appendix A.

Capability	Benchmark	Hardness ↑	Separability ↑	Diversity ↑
Knowledge	MMLU [Hendrycks et al., 2020]	-0.590	0.778	0.837
	MMLU-Pro [Wang et al., 2024]	-0.203	0.799	0.830
	GPQA [Rein et al., 2024]	0.370	0.712	0.750
	SimpleQA [Wei et al., 2024]	1.977	**0.908**	**0.840**
	HLE [Phan et al., 2025]	**2.808**	0.830	0.809
	Gaokao 2023 [Zhang et al., 2023]	-0.248	0.728	0.702
Math	AIME 2024 [AIME, 2025]	0.894	0.661	0.630
	AIME 2025 [AIME, 2025]	1.298	0.653	0.600
	HMMT Feb25 [Balunović et al., 2025]	**1.876**	0.642	0.633
	Math 500 Hendrycks et al. [2021]	-0.842	0.733	0.661
	MathOdyssey [Fang et al., 2025]	1.231	0.757	**0.672**
	OlympiadBench [He et al., 2024a]	0.523	**0.758**	0.637
Instruction Following	ComplexBench Wen et al. [2024]	0.322	0.680	0.835
	FollowBench [Jiang et al., 2023]	1.326	0.697	0.834
	IF-Bench [Pyatkin et al., 2025]	2.378	0.748	0.820
	IF-Eval [Zhou et al., 2023]	-0.028	0.720	0.808
	InfoBench [Qin et al., 2024]	-0.320	0.608	**0.857**
	MultiChallenge [Sirdeshmukh et al., 2025]	**2.847**	0.725	0.846
	Multi-IF [He et al., 2024b]	0.033	**0.794**	0.800
Agent	ACEBench [Chen et al., 2025]	-0.608	0.655	0.828
	BFCL [Patil et al.]	-0.230	0.738	0.780
	ComplexFuncBench [Zhong et al., 2025]	0.520	0.822	0.625
	DrafterBench [Li et al., 2025]	-0.826	0.707	0.474
	MultiChallenge [Sirdeshmukh et al., 2025]	0.839	0.750	**0.844**
	NexusBench [team, 2024]	**1.412**	0.707	0.799
	τ-Bench [Yao et al., 2024]	0.504	0.637	0.237
	τ^2-Bench [Barres et al., 2025]	0.769	0.724	0.366
	ToolSandbox [Lu et al., 2024]	0.856	**0.836**	0.352
Code	LiveCodeBench v5 [Jain et al., 2024]	-0.519	**0.891**	0.623
	LiveCodeBench v6 [Jain et al., 2024]	-0.251	0.854	**0.631**
	OJBench [Wang et al., 2025]	1.211	0.799	0.595
	Terminal-Bench [Team, 2025]	**1.327**	0.695	0.593
	SWE-bench-verified [Jimenez et al., 2023] (mini-swe-agent)	0.567	0.831	0.414
	SWE-bench-verified [Jimenez et al., 2023] (swe-agent)	0.512	0.839	0.528

Table 1: Hardness, separability and diversity scores for each dataset. Best scores for each capability are in **bold**. Hardness scores are calculated relative to other benchmarks within the same capability area. Knowledge datasets show the largest hardness gap, instruction following benchmarks show highest diversity and dataset with more samples show higher separability.

Seperability. Benchmarks like SimpleQA, LiveCodeBench, and ToolSandbox provides high separability due to both high number of prompts and wide spread of scores. Benchmarks like AIME 2024 and 2025 are weaker in separability due to small amount of prompts covered, making it harder to separate models confidently.

Diversity. For diversity, we use QWEN3-EMBEDDING-8B [Zhang et al., 2025] as a text encoder to embed each benchmark prompt and compute benchmark-level semantic dispersion (4). Benchmarks in Instruction-Following and Knowledge generally exhibit the highest diversity, while most Math and Coding benchmarks show relatively lower diversity, reflecting more specialized domain knowledge and templated problem formats. Agent benchmarks are bimodal, with some high and others clearly low. The low-diversity group usually pairs long system prompts with short user prompts, which reduces diversity.

References

AIME. Aime problems and solutions. `https://artofproblemsolving.com/wiki/index.php/AIME_Problems_and_Solutions`, 2025. Accessed: 2025-04-20.

Simone Balloccu, Patrícia Schmidtová, Mateusz Lango, and Ondřej Dušek. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source llms. *arXiv preprint arXiv:2402.03927*, 2024.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions. *arXiv preprint arXiv:2505.23281*, 2025.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-bench: Evaluating conversational agents in a dual-control environment. *arXiv preprint arXiv:2506.07982*, 2025.

Li Cai, Kilchan Choi, Mark Hansen, and Lauren Harrell. Item response theory. *Annual Review of Statistics and Its Application*, 3(1):297–321, 2016.

Chen Chen, Xinlong Hao, Weiwen Liu, Xu Huang, Xingshan Zeng, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Yuefeng Huang, et al. Acebench: Who wins the match point in tool learning? *arXiv e-prints*, pages arXiv–2501, 2025.

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.

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.

Meng Fang, Xiangpeng Wan, Fei Lu, Fei Xing, and Kai Zou. Mathodyssey: Benchmarking mathematical problem-solving skills in large language models using odyssey math data. *Scientific Data*, 12(1):1392, 2025.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*, 2024a.

Yun He, Di Jin, Chaoqi Wang, Chloe Bi, Karishma Mandyam, Hejia Zhang, Chen Zhu, Ning Li, Tengyu Xu, Hongjiang Lv, et al. Multi-if: Benchmarking llms on multi-turn and multilingual instructions following. *arXiv preprint arXiv:2410.15553*, 2024b.

David Heineman, Valentin Hofmann, Ian Magnusson, Yuling Gu, Noah A Smith, Hannaneh Hajishirzi, Kyle Lo, and Jesse Dodge. Signal and noise: A framework for reducing uncertainty in language model evaluation. *arXiv preprint arXiv:2508.13144*, 2025.

Mark Hempstead, Matt Welsh, and David Brooks. Tinybench: The case for a standardized benchmark suite for tinyos based wireless sensor network devices. In *29th Annual IEEE International Conference on Local Computer Networks*, pages 585–586. IEEE, 2004.

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.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. Followbench: A multi-level fine-grained constraints following benchmark for large language models. *arXiv preprint arXiv:2310.20410*, 2023.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*, 2023.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, et al. Dynabench: Rethinking benchmarking in nlp. *arXiv preprint arXiv:2104.14337*, 2021.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. *arXiv preprint arXiv:2406.11939*, 2024.

Yinsheng Li, Zhen Dong, and Yi Shao. Drafterbench: Benchmarking large language models for tasks automation in civil engineering. *arXiv preprint arXiv:2507.11527*, 2025.

Fengze Liu, Weidong Zhou, Binbin Liu, Zhimiao Yu, Yifan Zhang, Haobin Lin, Yifeng Yu, Bingni Zhang, Xiaohuan Zhou, Taifeng Wang, et al. Quadmix: Quality-diversity balanced data selection for efficient llm pretraining. *arXiv preprint arXiv:2504.16511*, 2025.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Felix Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, et al. Toolsandbox: A stateful, conversational, interactive evaluation benchmark for llm tool use capabilities. *arXiv preprint arXiv:2408.04682*, 2024.

Timothy R McIntosh, Teo Susnjak, Nalin Arachchilage, Tong Liu, Dan Xu, Paul Watters, and Malka N Halgamuge. Inadequacies of large language model benchmarks in the era of generative artificial intelligence. *IEEE Transactions on Artificial Intelligence*, 2025.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022.

Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures. *Advances in Neural Information Processing Systems*, 37:98180–98212, 2024.

Shishir G Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E Gonzalez. The berkeley function calling leaderboard (bfcl): From tool use to agentic evaluation of large language models. In *Forty-second International Conference on Machine Learning*.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. Generalizing verifiable instruction following. *arXiv preprint arXiv:2507.02833*, 2025.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. Infobench: Evaluating instruction following ability in large language models. *arXiv preprint arXiv:2401.03601*, 2024.

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. In *First Conference on Language Modeling*, 2024.

Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. *arXiv preprint arXiv:2310.18018*, 2023.

Chantal Shaib, Joe Barrow, Jiuding Sun, Alexa F Siu, Byron C Wallace, and Ani Nenkova. Standardizing the measurement of text diversity: A tool and a comparative analysis of scores. *arXiv preprint arXiv:2403.00553*, 2024.

Alexander Shypula, Shuo Li, Botong Zhang, Vishakh Padmakumar, Kayo Yin, and Osbert Bastani. Evaluating the diversity and quality of llm generated content. *arXiv preprint arXiv:2504.12522*, 2025.

Ved Sirdeshmukh, Kaustubh Deshpande, Johannes Mols, Lifeng Jin, Ed-Yeremai Cardona, Dean Lee, Jeremy Kritz, Willow Primack, Summer Yue, and Chen Xing. Multichallenge: A realistic multi-turn conversation evaluation benchmark challenging to frontier llms. *arXiv preprint arXiv:2501.17399*, 2025.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adri Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on machine learning research*, 2023.

Nexusflow.ai team. Nexusbench: Fc and agent benchmarking suite, 2024. URL `https://github.com/nexusflowai/NexusBench`.

The Terminal-Bench Team. Terminal-bench: A benchmark for ai agents in terminal environments, Apr 2025. URL `https://github.com/laude-institute/terminal-bench`.

Norman D Verhelst and Cees AW Glas. The one parameter logistic model. In *Rasch models: Foundations, recent developments, and applications*, pages 215–237. Springer, 1995.

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

Zhexu Wang, Yiping Liu, Yejie Wang, Wenyang He, Bofei Gao, Muxi Diao, Yanxu Chen, Kelin Fu, Flood Sung, Zhilin Yang, et al. Ojbench: A competition level code benchmark for large language models. *arXiv preprint arXiv:2506.16395*, 2025.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Bosi Wen, Pei Ke, Xiaotao Gu, Lindong Wu, Hao Huang, Jinfeng Zhou, Wenchuang Li, Binxin Hu, Wendy Gao, Jiaxing Xu, et al. Benchmarking complex instruction-following with multiple constraints composition. *Advances in Neural Information Processing Systems*, 37:137610–137645, 2024.

Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Ben Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Siddartha Naidu, et al. Livebench: A challenging, contamination-free llm benchmark. *arXiv preprint arXiv:2406.19314*, 4, 2024.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. Evaluating the performance of large language models on gaokao benchmark. *arXiv preprint arXiv:2305.12474*, 2023.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

Lucen Zhong, Zhengxiao Du, Xiaohan Zhang, Haiyi Hu, and Jie Tang. Complexfuncbench: exploring multi-step and constrained function calling under long-context scenario. *arXiv preprint arXiv:2501.10132*, 2025.

Hongli Zhou, Hui Huang, Ziqing Zhao, Lvyuan Han, Huicheng Wang, Kehai Chen, Muyun Yang, Wei Bao, Jian Dong, Bing Xu, et al. Lost in benchmarks? rethinking large language model benchmarking with item response theory. *arXiv preprint arXiv:2505.15055*, 2025.

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 Analysis

A.1 Hardness Distribution

We present the results of hardness distribution from Figure 1 to Figure 5. The hardness distributions exhibit markedly different characteristics across the five capabilities. **Instruction following** datasets predominantly cluster at low difficulty levels, though MultiChallenge shows a more balanced distribution. **Math** datasets display the most varied patterns: while commonly-used benchmarks like Math-500 and OlympiadBench show a long-tailed distribution, specialized competitions (AIME25, HMMT) extend into higher difficulty ranges, showing near uniformal or normal distribution. **Knowledge** datasets either concentrate at very low difficulty or shows a distinctive peak at high difficulty levels. **Code** datasets generally exhibit bimodal distributions with peaks at both the lowest and highest difficulty levels, revealing substantial intra-dataset difficulty variance. **Agent** capabilities display the most consistent uniform distributions across datasets. This analysis reveals that benchmark difficulty varies dramatically not only between datasets but also within each dataset, highlighting the inadequacy of relying on popular but easy benchmarks for comprehensive model evaluation.

A.2 Model Capabilities

We present the model capabilities calculated by the IRT models in Table 2. This analysis reveals that model evaluation should move beyond simple average accuracy metrics but should consider performance across varying prompt hardness levels such as model capabilities learned form IRT models.

B Future Work: Core–Set Selection via Submodular Optimization

As part of the future work, we plan to develope core-set selection algorithm for the entire benchmark prompt dataset with submodular optimization. Let $g(S)$ measure the quality of subset S (e.g. a combination of difficulty and separability) and $d(S)$ its diversity (e.g. C_{sem}). We choose

$$f(S) = g(S) + \alpha\, d(S), \quad 0 \le \alpha \le 1. \tag{5}$$

We would like to choose both g and d as monotone submodular surrogates. Under a cardinality constraint $|S| \le k$ the greedy algorithm obtains a $(1 - 1/e)$ approximation to

$$\max_{S \subseteq \mathcal{B},\, |S| \le k} f(S). \tag{6}$$

Empirically, $k = 100$ balances evaluation cost with fidelity to the full benchmark (rank–correlation > 0.95).

Model	Knowledge	Math	Instruction Following	Agent	Code	Overall
GPT-4O-MINI	0.663	0.061	3.092	0.249	-2.170	1.311
GPT-4O	1.560	0.157	3.189	1.251	-1.696	0.537
GPT-4.1	1.905	1.240	4.725	1.421	-0.884	1.588
O3-HIGH	2.420	2.821	6.232	1.404	1.139	2.220
O4-MINI-HIGH	1.709	3.062	5.431	0.943	0.980	1.596
DEEPSEEK-V3	1.735	1.737	4.047	0.629	-0.889	1.324
DEEPSEEK-R1	1.981	4.163	3.569	0.622	0.228	1.578
CLAUDE 4 SONNET	1.671	1.695	4.629	0.829	-0.155	1.511
CLAUDE 4 SONNET (think)	1.839	2.993	4.908	1.096	-0.767	1.561
KIMI-K2-INSTRUCT	1.850	2.742	4.956	0.965	-0.359	1.649
QWEN3-235B	1.691	4.247	0.621	0.395	-0.186	1.079
QWEN3-235B-INSTRUCT	2.085	3.428	4.509	0.498	-0.459	1.600

Table 2: Model capabilities θ_m computed by IRT models. θ_m gives a more hardness-aware ranking than accuray.

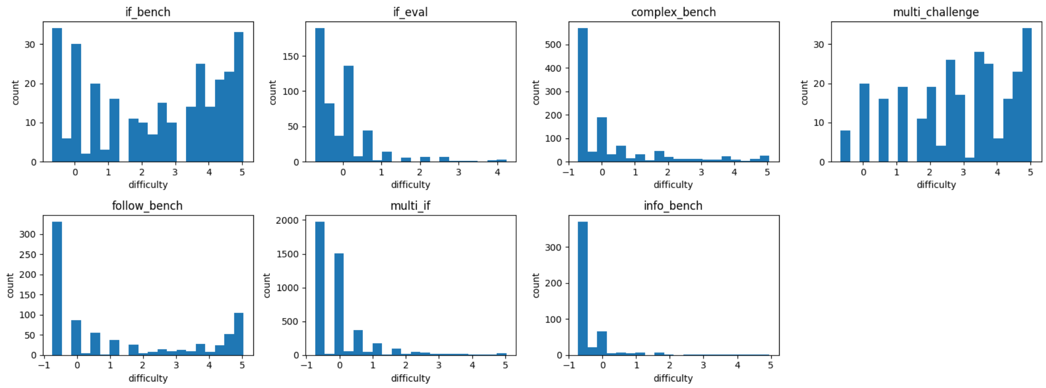

Figure 1: Hardness distribution on instruction following datasets.

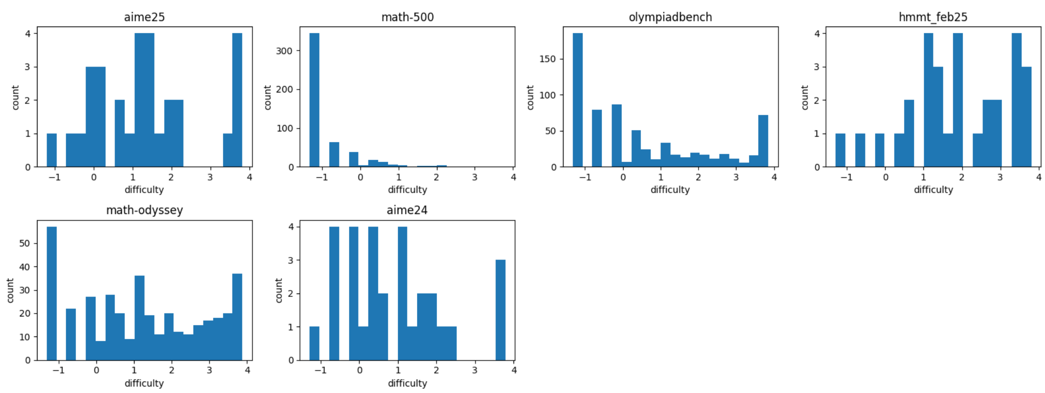

Figure 2: Hardness distribution on math datasets.

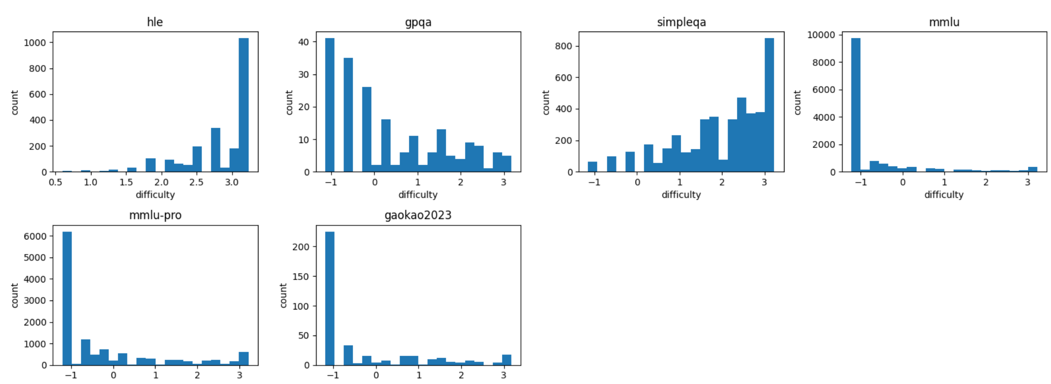

Figure 3: Hardness distribution on knowledge datasets.

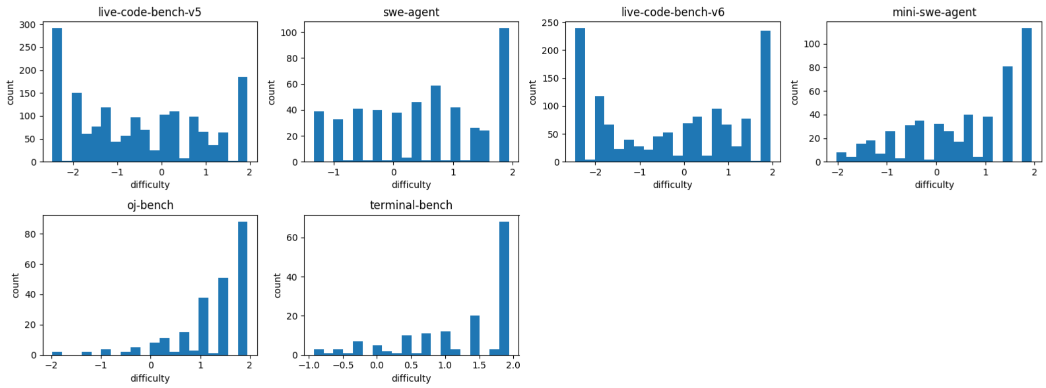

Figure 4: Hardness distribution on code datasets.

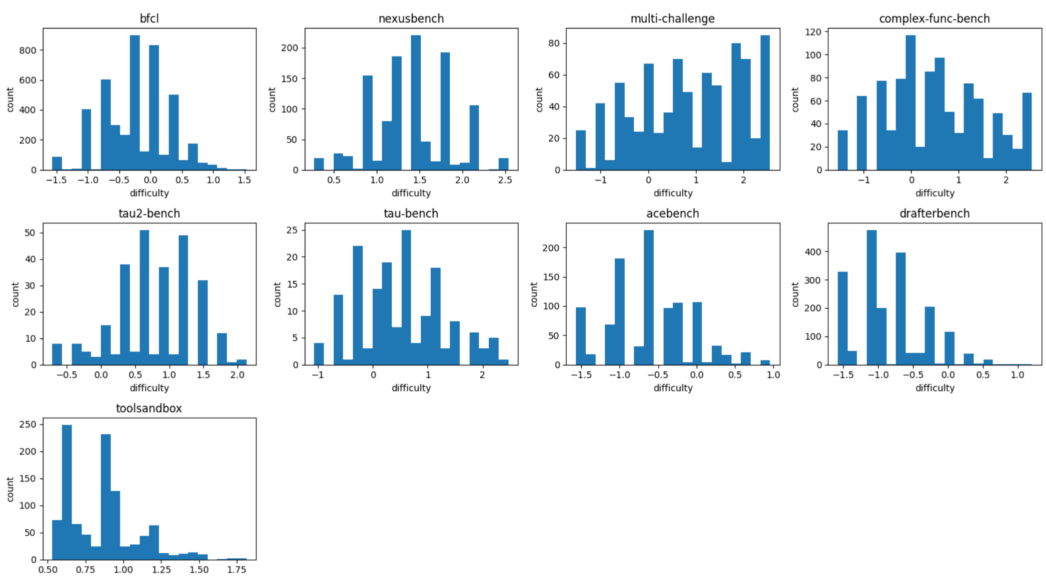

Figure 5: Hardness distribution on agent datasets.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]