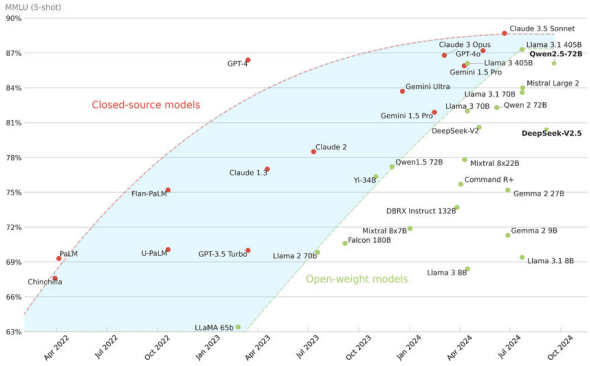
HPS

Niclas Unger

LLM Benchmarking Frameworks and their limitations

Motivation

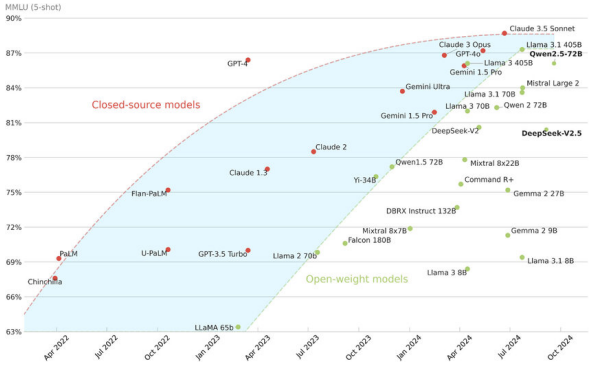
- Best Large Language Model?



Source: Riedemann, Labonne, and Gilbert (2024), Figure 2

Motivation

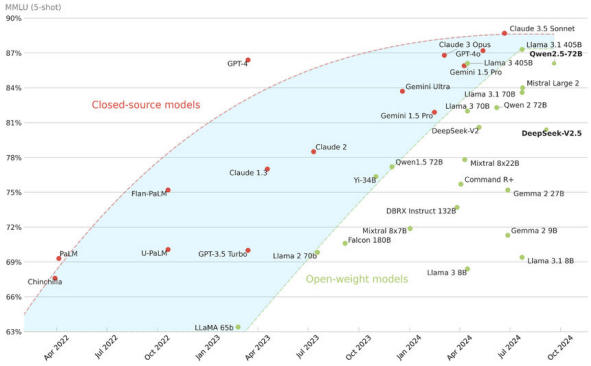
- Best Large Language Model?
  - Measure performance
    $\rightarrow$ benchmarks



Source: Riedemann, Labonne, and Gilbert (2024), Figure 2

Motivation

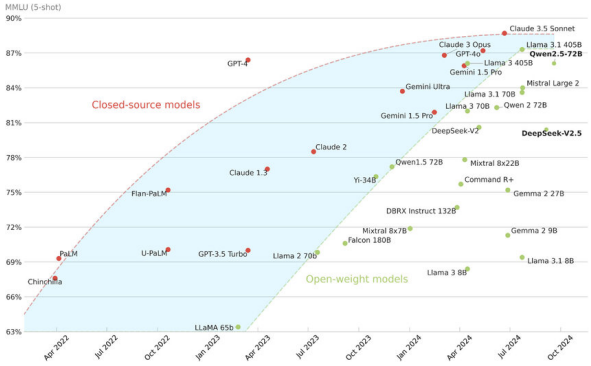
- Best Large Language Model?
  - Measure performance
    $\rightarrow$ benchmarks
- Structure, reproducibility, comparability, automation, scalability
  $\rightarrow$ benchmarking frameworks



Source: Riedemann, Labonne, and Gilbert (2024), Figure 2

Motivation

- Best Large Language Model?
  - Measure performance
    $\rightarrow$ benchmarks
- Structure, reproducibility, comparability, automation, scalability
  $\rightarrow$ benchmarking frameworks
- Can we truly trust the scores from benchmarking frameworks?



Source: Riedemann, Labonne, and Gilbert (2024), Figure 2

Table of contents

What is an LLM benchmark

- **Dataset**
  - Set of **tasks**/ subjects
  - Fixed sets of **questions** for each task

What is an LLM benchmark

- **Dataset**
  - Set of **tasks**/ subjects
  - Fixed sets of **questions** for each task
- Performance evaluation **metric**
  - E.g.: Accuracy, efficiency, robustness, bias, toxicity

What is an LLM benchmark

- **Dataset**
  - Set of **tasks**/ subjects
  - Fixed sets of **questions** for each task
- Performance evaluation **metric**
  - E.g.: Accuracy, efficiency, robustness, bias, toxicity
- Categories
  - Knowledge
  - Reasoning
  - Language
  - Bias, toxicity, safety
  - Coding
  - ...

Why we need good benchmarks

- Allows **comparison** of different models
  - Main **indicator for progress** of models
- **fairness** and **competitiveness** for developers

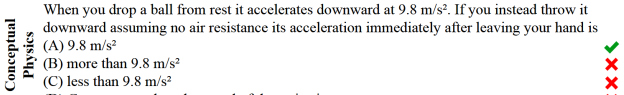
Example: MMLU

- Massive Multitask Language Understanding
- 16.000 multiple choice questions
  - development, validation, test sets
  - 57 tasks/ subjects
- Scored via accuracy
  (choose highest probability
  answer token)

Hendrycks, Burns, Basart, et al., *Measuring Massive Multitask Language Understanding*, 2021

Example: MMLU

- Massive Multitask Language Understanding
- 16.000 multiple choice questions
  - development, validation, test sets
  - 57 tasks/ subjects
- Scored via accuracy
  (choose highest probability
  answer token)



Source: Hendrycks, Burns, Basart, et al. (2021), Figure 4

Hendrycks, Burns, Basart, et al., *Measuring Massive Multitask Language Understanding*, 2021

Example: MMLU

- Massive Multitask Language Understanding
- 16.000 multiple choice questions
  - development, validation, test sets
  - 57 tasks/ subjects
- Scored via accuracy (choose highest probability answer token)

Few Shot Prompt and Predicted Answer

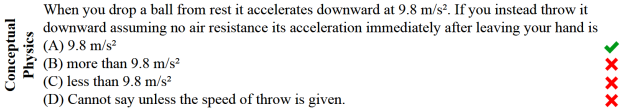
The following are multiple choice questions about high school mathematics.

How many numbers are in the list 25, 26, ..., 100?
(A) 75 (B) 76 (C) 22 (D) 23
Answer: B

Compute $i + i^2 + i^3 + \cdots + i^{258} + i^{259}$.
(A) -1 (B) 1 (C) i (D) $-i$
Answer: A

If 4 daps = 7 yaps, and 5 yaps = 3 baps, how many daps equal 42 baps?
(A) 28 (B) 21 (C) 40 (D) 30
Answer: C

Source: Hendrycks, Burns, Basart, et al. (2021), Figure 1 (a)



Source: Hendrycks, Burns, Basart, et al. (2021), Figure 4

Hendrycks, Burns, Basart, et al., *Measuring Massive Multitask Language Understanding*, 2021

Example: HellaSwag

- ∼70.000 common sense multiple choice language inference questions
- Improvement of **Swag** dataset
  - Source text, generator, discriminator
  - Adversarial filtering

A woman is outside with a bucket and a dog. The dog is running around trying to avoid a bath. She…

A. rinses the bucket off with soap and blow dry the dog's head.
B. uses a hose to keep it from getting soapy.
C. gets the dog wet, then it runs away again.
D. gets into a bath tub with the dog.

Source: Zellers et al. (2019), Figure 1

Source: Zellers et al., *HellaSwag: Can a Machine Really Finish Your Sentence?*, 2019

Other benchmark examples

- GSM8K
- ARC
- CommonsenseQA
- MATH
- Modified versions of MMLU
  - MMLU-Pro
  - MMMLU
  - MMLU-Redux
- ...

*Where on a **river** can you hold a cup upright to catch water on a sunny day?*

✓ **waterfall**, ✗ bridge, ✗ valley, ✗ pebble, ✗ mountain

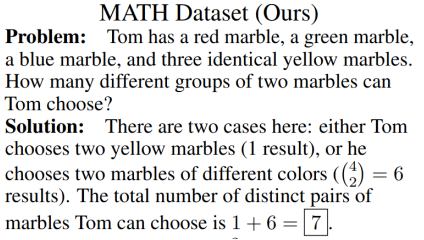
*Where can I stand on a **river** to see water falling without getting wet?*

✗ waterfall, ✓ **bridge**, ✗ valley, ✗ stream, ✗ bottom

*I'm crossing the **river**, my feet are wet but my body is dry, where am I?*

✗ waterfall, ✗ bridge, ✓ **valley**, ✗ bank, ✗ island

CommonsenseQA, Source: Talmor et al. (2019), Figure 1



MATH Dataset (Ours)

Problem: Tom has a red marble, a green marble, a blue marble, and three identical yellow marbles. How many different groups of two marbles can Tom choose?

Solution: There are two cases here: either Tom chooses two yellow marbles (1 result), or he chooses two marbles of different colors ($\binom{4}{2} = 6$ results). The total number of distinct pairs of marbles Tom can choose is $1 + 6 = \boxed{7}$.

Source: Hendrycks, Burns, Kadavath, et al. (2021), Figure 1

Outline

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs
- Executes **one or multiple benchmarks** on **one or multiple models**

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs
- Executes **one or multiple benchmarks** on **one or multiple models**
- Defines **evaluation protocols**

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs
- Executes **one or multiple benchmarks** on **one or multiple models**
- Defines **evaluation protocols**
  - Prompt templates (zero-shot, few-shot, Chain-of-Thought)

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs
- Executes **one or multiple benchmarks** on **one or multiple models**
- Defines **evaluation protocols**
  - Prompt templates (zero-shot, few-shot, Chain-of-Thought)
  - Decoding settings

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs
- Executes **one or multiple benchmarks** on **one or multiple models**
- Defines **evaluation protocols**
  - Prompt templates (zero-shot, few-shot, Chain-of-Thought)
  - Decoding settings
  - Output scoring

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs
- Executes **one or multiple benchmarks** on **one or multiple models**
- Defines **evaluation protocols**
  - Prompt templates (zero-shot, few-shot, Chain-of-Thought)
  - Decoding settings
  - Output scoring
- Ensures **comparability and reproducibility** across models and runs

What is an LLM Benchmarking Framework

- (Software) system that **standardizes and/or automates** the evaluation of LLMs
- Executes **one or multiple benchmarks** on **one or multiple models**
- Defines **evaluation protocols**
  - Prompt templates (zero-shot, few-shot, Chain-of-Thought)
  - Decoding settings
  - Output scoring
- Ensures **comparability and reproducibility** across models and runs
- Tends to support **scalable execution** (e.g. batching, parallelism)

Types of LLM Benchmarking Frameworks

- **Classical**
  - Fixed datasets, metrics, models
  - provide code
  - Some: Focus on prompt sensitivity and instability
- **Holistic**
- **Human-preference evaluation platform**
  - Ranking via user feedback

Example: LM Evaluation Harness

- Classical
- "Unified framework to test generative language models on a large number of different evaluation tasks"
- Supports:
  - >60 benchmarks
  - Many different LLMs
  - Commercial APIs, e.g. OpenAI
  - Local models and benchmarks
  - Custom prompts and evaluation metrics
  - Multi GPU parallelism

Gao et al., *A framework for few-shot language model evaluation*, 2021

Example: LM Evaluation Harness

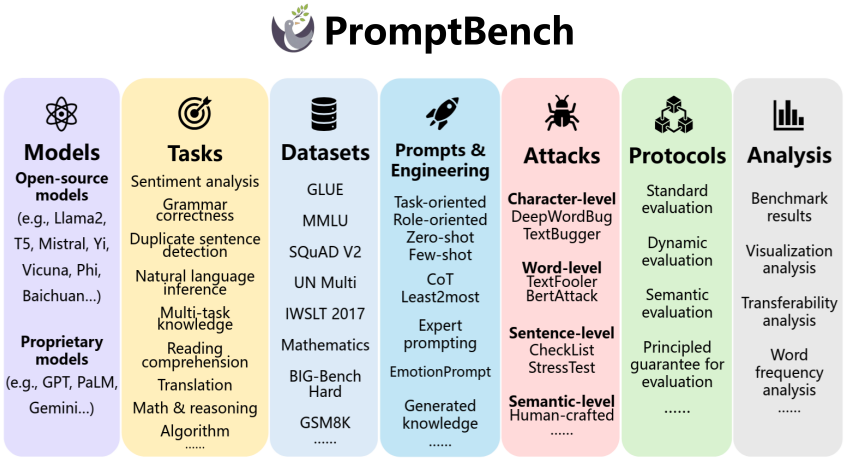
Evaluate a model "hosted" on the HuggingFace Hub (e.g. GPT-J-6B) on HellaSwag

bash

```
lm_eval --model hf \
    --model_args pretrained=EleutherAI/gpt-j-6B \
    --tasks hellaswag \
    --device cuda:0 \
    --batch_size 8
```

Gao et al., *A framework for few-shot language model evaluation*, 2021, Github README

Example: PromptBench



Source: Zhu et al., *PromptBench: A Unified Library for Evaluation of Large Language Models*, 2024, Figure 1

Other Examples

- **OpenCompass** (classical)
- **HELM** (holistic)
- **Chatbot Arena**
  (human-preference evaluation platform)

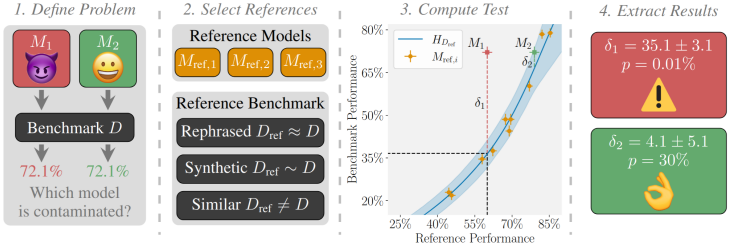
Outline

Benchmark limitations Overview

- Data Contamination
- Prompt Sensitivity
- Evaluation process
  - Metric sensitivity
  - Handling of outputs
- Static nature
  - Changing capability of LLMs $\rightarrow$ outdated
  - Increased risk of data contamination
- Correctness

ConStat

- Data Contamination
  - Syntax-Specific
  - Sample-Specific
  - Benchmark-Specific
- Reference models
- Reference benchmarks



Source: Dekoninck, Müller, and Vechev (2024), Figure 1

Model	Benchmark	Type	Perf. [%]	p [%]	$\hat{\delta}$ [%]	$\hat{\delta}_{0.95}$ [%]
LLAMA-3-70b	ARC	S	69.03	0.03	6.61	3.21
MISTRAL-7b-v0.1	GSM8k	S	39.04	0.15	8.25	4.48
MISTRAL-7b-v0.1	Hellaswag	S	83.65	0.24	3.14	1.27
LLAMA-2-INSTRUCT-70b	Hellaswag	S	85.55	0.41	3.37	1.29
MISTRAL-INSTRUCT-7b-v0.2	ARC	B	62.46	0.04	10.62	5.95
MISTRAL-INSTRUCT-7b-v0.2	Hellaswag	B	84.55	0.18	3.52	1.56
PHI-2	GSM8k	B	58.91	$< 10^{-2}$	36.42	26.46
PHI-3-MINI	GSM8k	B	76.65	0.29	16.30	6.33
OLMO-INSTRUCT-7b	GSM8k	B	11.75	$< 10^{-2}$	8.86	4.99

Source: Dekoninck, Müller, and Vechev (2024), Table 2

Dekoninck, Müller, and Vechev, *ConStat: Performance-Based Contamination Detection in Large Language Models*, 2024

ConStat Results

Model	Benchmark	Type	Perf. [%]	p [%]	$\hat{\delta}$ [%]	$\hat{\delta}_{0.95}$ [%]
QWEN-INSTRUCT-1.5-14b	ARC	B	56.91	0.71	11.77	5.94
QWEN-INSTRUCT-1.5-72b	ARC	B	64.68	0.01	12.59	7.61
QWEN-INSTRUCT-1.5-110b	ARC	B	69.45	0.12	9.33	4.47
QWEN-INSTRUCT-1.5-4b	GSM8k	S	6.52	$< 10^{-2}$	5.35	4.01
QWEN-INSTRUCT-1.5-7b	Hellaswag	B	78.65	0.74	7.46	3.00
QWEN-INSTRUCT-1.5-14b	Hellaswag	B	82.15	0.07	6.48	3.74
QWEN-INSTRUCT-1.5-72b	Hellaswag	B	86.35	$< 10^{-2}$	8.24	6.05
YI-34b	ARC	S	63.99	0.20	5.00	2.12
YI-34b	Hellaswag	S	86.15	$< 10^{-2}$	6.51	4.40
YI-34b	Hellaswag	B	86.15	0.14	3.96	1.89
INTERNLM-2-7b	GSM8k	B	62.09	0.43	19.27	7.98
INTERNLM-2-MATH-7b	GSM8k	B	72.93	$< 10^{-2}$	39.40	27.15
INTERNLM-2-7b	Hellaswag	B	80.10	0.41	6.58	3.19
INTERNLM-2-MATH-7b	Hellaswag	B	77.65	0.90	8.55	3.11
INTERNLM-2-MATH-BASE-7b	Hellaswag	B	79.65	0.40	11.41	5.51
STABLELM-2-12b	ARC	S	59.47	0.68	4.61	1.59
STABLELM-2-1.6b	GSM8k	B	18.88	$< 10^{-2}$	16.56	12.95
STABLELM-2-INSTRUCT-1.6b	GSM8k	B	42.00	$< 10^{-2}$	27.79	19.27
STABLELM-2-ZEPHYR-3b	GSM8k	B	51.63	$< 10^{-2}$	48.78	44.34
STABLELM-2-12b	GSM8k	B	58.00	0.39	17.92	6.96
STABLELM-2-INSTRUCT-12b	GSM8k	B	68.84	$< 10^{-2}$	32.93	21.09
STABLELM-2-INSTRUCT-12b	Hellaswag	B	86.25	$< 10^{-2}$	7.04	4.88

Source: Dekoninck, Müller, and Vechev, *ConStat: Performance-Based Contamination Detection in Large Language Models*, 2024, Table 4

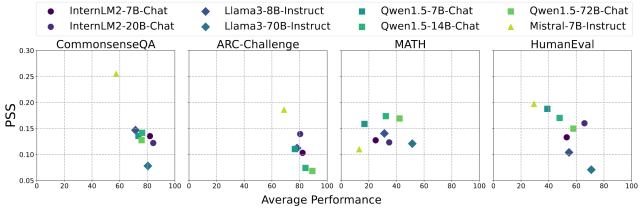
Data contamination - why this matters

- Contamination **skews reported model performance** between older and newer models
- For frameworks: Benchmarking frameworks provide only tools
  - **Results** can **lack validity**
- User vigilance and awareness necessary

ProSA

- Framework designed to evaluate **prompt sensitivity**
  - PromptSensiScore

Simple Input	Problem:\n{problem}\nSolution:\n
Emotional Support	Please provide a solution to the following problem:\n{problem}\n
Role Player	You are a very helpful mathematical problem solver. Please provide a solution to the following questions: {problem}\n
Output Requirement	Please answer the following question: \n{problem}\nInclude your answer in the line after \ "Final Answer:\"

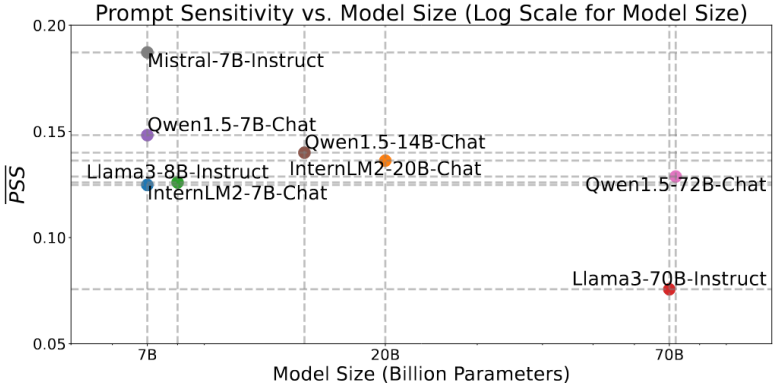


Source: Zhuo et al. (2024), Figure 3

Source: Zhuo et al. (2024), Figure 1

Zhuo et al., *ProSA: Assessing and Understanding the Prompt Sensitivity of LLMs*, 2024

ProSA Results



Source: Zhuo et al., *ProSA: Assessing and Understanding the Prompt Sensitivity of LLMs*, 2024, Figure 4

Prompt sensitivity - why this matters

- **Skews benchmark results** for certain prompting methods
- For frameworks: Same as with data contamination
  - Benchmarking frameworks provide only tools
    - **Results** can **lack validity**
  - User vigilance and awareness necessary

Framework Limitations

- Directly inherit limitations of benchmarks
  $\rightarrow$ potentially unreliable scores
- Closed models
- Closed weights
- No log-likelihood
- Coding challenges
  - Version compatibility
  - Reliance on external libraries and APIs

Outline

My plans for the full report

- Focus on LM Eval Harness
  - Scalability and technical aspects of the framework
- Expand on benchmark limitations
  - More papers and quantification, wider coverage
  - How limitations affect specifically new, state of the art models and benchmarks

Key Takeaways

- Frameworks structure benchmarking
  - Give LLM benchmark evaluation structure, guidelines (and automation)
  - Types: **Classical**, **Holistic**, **Human-preference**
- Interpret benchmark scores with caution
  - **Data contamination** $\rightarrow$ newer models have unfair advantage
  - **Prompt sensitivity** $\rightarrow$ prompt wording skews results
  - Sponsors of benchmarks?
- Still more research required on reliability and improvement of benchmarks
  - How reliable are benchmark scores exactly?

References

Dekoninck, Jasper, Mark Niklas Müller, and Martin Vechev. *ConStat: Performance-Based Contamination Detection in Large Language Models*. 2024. arXiv: 2405.16281 [cs.CL]. URL: https://arxiv.org/abs/2405.16281.

Gao, Leo et al. *A framework for few-shot language model evaluation*. Version v0.0.1. Sept. 2021. DOI: 10.5281/zenodo.5371628. URL: https://doi.org/10.5281/zenodo.5371628.

Hendrycks, Dan, Collin Burns, Steven Basart, et al. *Measuring Massive Multitask Language Understanding*. 2021. arXiv: 2009.03300 [cs.CY]. URL: https://arxiv.org/abs/2009.03300.

Hendrycks, Dan, Collin Burns, Saurav Kadavath, et al. *Measuring Mathematical Problem Solving With the MATH Dataset*. 2021. arXiv: 2103.03874 [cs.LG]. URL: https://arxiv.org/abs/2103.03874.

Riedemann, Lars, Maxime Labonne, and Stephen Gilbert. "The path forward for large language models in medicine is open". In: *npj Digital Medicine* 7 (Nov. 2024). DOI: 10.1038/s41746-024-01344-w.

Talmor, Alon et al. *CommonsenseQA: A Question Answering Challenge Targeting Commonsense Knowledge*. 2019. arXiv: 1811.00937 [cs.CL]. URL: https://arxiv.org/abs/1811.00937.

Zellers, Rowan et al. *HellaSwag: Can a Machine Really Finish Your Sentence?* 2019. arXiv: 1905.07830 [cs.CL]. URL: https://arxiv.org/abs/1905.07830.

Zhu, Kaijie et al. *PromptBench: A Unified Library for Evaluation of Large Language Models*. 2024. arXiv: 2312.07910 [cs.AI]. URL: https://arxiv.org/abs/2312.07910.

Zhuo, Jingming et al. *ProSA: Assessing and Understanding the Prompt Sensitivity of LLMs*. 2024. arXiv: 2410.12405 [cs.CL]. URL: https://arxiv.org/abs/2410.12405.