

Announcements

- Logistics:** HW 2 due next week
- Presentations:** About 73% signed up, please sign up!
- Class roadmap:

Thursday Oct. 16	Evaluation
Tuesday Oct. 21	Agents
Thursday Oct. 23	More Reasoning
Tuesday Oct. 28	Multimodal Models
Thursday Oct. 30	Scaling Laws

Outline

- **Evaluation Intro & Benchmarks**

- Challenges, benchmark requirements, popular benchmarks, HumanEval, MMLU and variants, HELM

- **LLM-as-a-judge**

- Basic setup, framework for automated evaluation, biases, bias reduction studies and techniques

- **Variations**

- Combining automated evaluation with benchmarks: AlpacaEval, agentic benchmarks

Outline

- **Evaluation Intro & Benchmarks**

- Challenges, benchmark requirements, popular benchmarks, HumanEval, MMLU and variants, HELM

- **LLM-as-a-judge**

- Basic setup, framework for automated evaluation, biases, bias reduction studies and techniques

- **Variations**

- Combining automated evaluation with benchmarks: AlpacaEval, agentic benchmarks

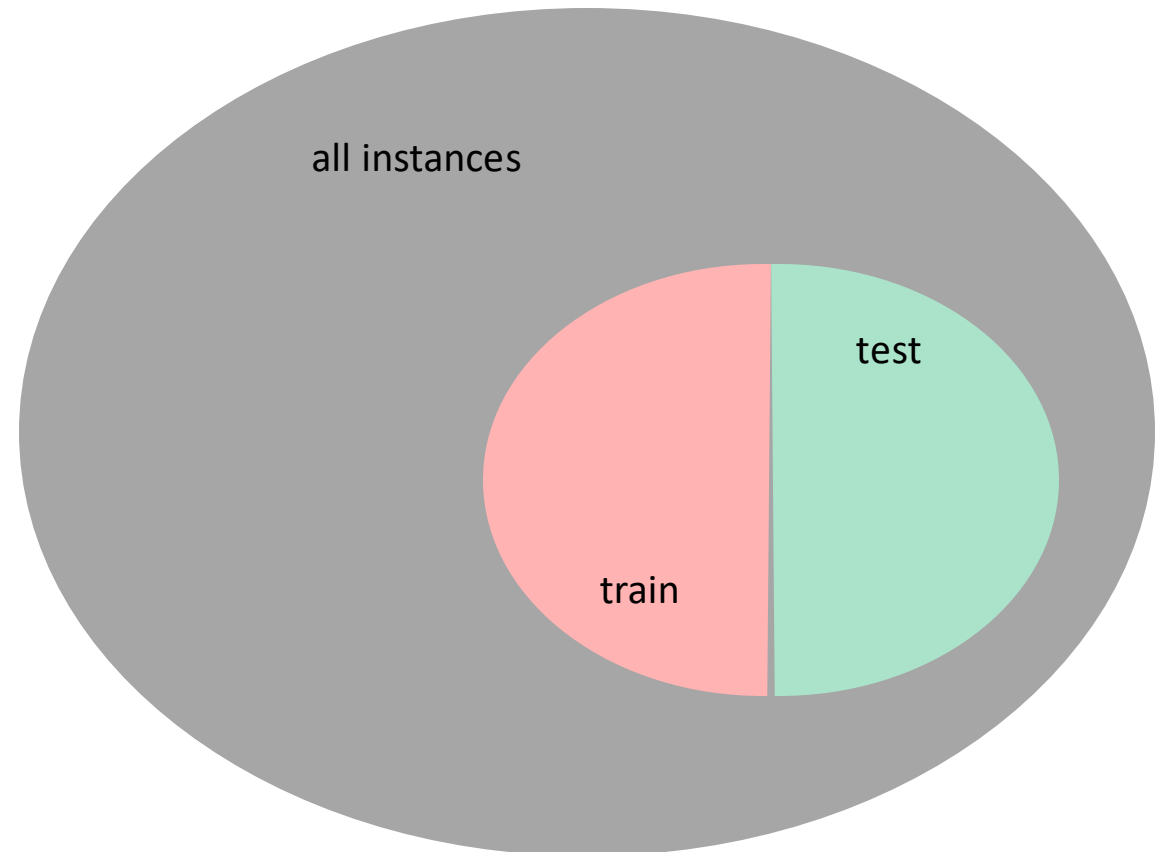
Model Evaluation Basics

Traditional approach in ML:

- Measure accuracy or a related metric on a test set
- Or perform cross-validation, etc.

- Can switch from accuracy to other metrics: AUC-ROC, F1 non-scalar metrics like the confusion matrix, etc.

Could still do some of these...



Model Evaluation Basics

For large language models, a bit more complex

- Far more general capabilities
- Space of outputs much larger than multiclass classification!
 - Many answers might be right!
- What do we need for an evaluation system? Some pieces:
 - Dataset
 - Metrics
 - Mechanism to compute metrics on model

Example: HumanEval

Chen et al '21 introduced **Codex**

- Essentially a fine-tuned version of GPT3 for code
- How to evaluate?
 - Output is now code---lots of ways to write “good” code

- **Example:**

```
def incr_list(l: list):  
    """Return list with elements incremented by 1.  
    >>> incr_list([1, 2, 3])  
    [2, 3, 4]  
    >>> incr_list([5, 3, 5, 2, 3, 3, 9, 0, 123])  
    [6, 4, 6, 3, 4, 4, 10, 1, 124]  
    """  
    return [i + 1 for i in l]
```

Example: HumanEval

Chen et al '21 introduced **Codex**

- What do we need for an evaluation system?
- **Dataset:** “a set of 164 handwritten programming problems”
 - Each problem: definition, some metadata, variable # of test cases
 - “Programming tasks ... assess language comprehension, reasoning, algorithms, and simple mathematics”

```
def solution(lst):  
    """Given a non-empty list of integers, return the sum of all of the odd elements  
    that are in even positions.  
  
    Examples  
    solution([5, 8, 7, 1]) ==>12  
    solution([3, 3, 3, 3, 3]) ==>9  
    solution([30, 13, 24, 321]) ==>0  
    """  
    return sum(lst[i] for i in range(0, len(lst)) if i % 2 == 0 and lst[i] % 2 == 1)
```


Example: HumanEval

Chen et al '21 introduced **Codex**

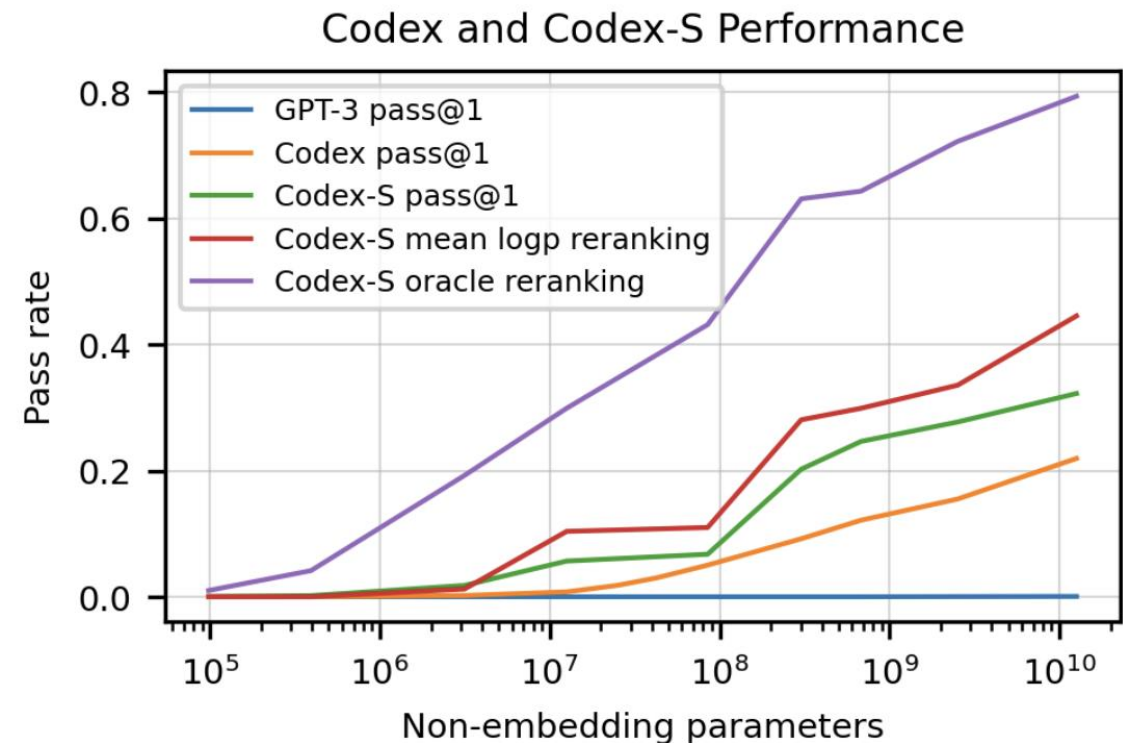
- What do we need for an evaluation system?
- **Metrics:** pass@k metric
 - Generate k samples, check if *any* sample passes all unit tests
 - To decrease variance, generate *multiple* sets of k samples
 - n samples (n=200, k=100), take k-element subsets out of n. Count number c of solutions, then estimate
 - Check for yourself that this is unbiased!

$$\text{pass@}k := \mathbb{E}_{\text{Problems}} \left[1 - \frac{\binom{n-c}{k}}{\binom{n}{k}} \right]$$

Example: HumanEval

Chen et al '21 introduced Codex

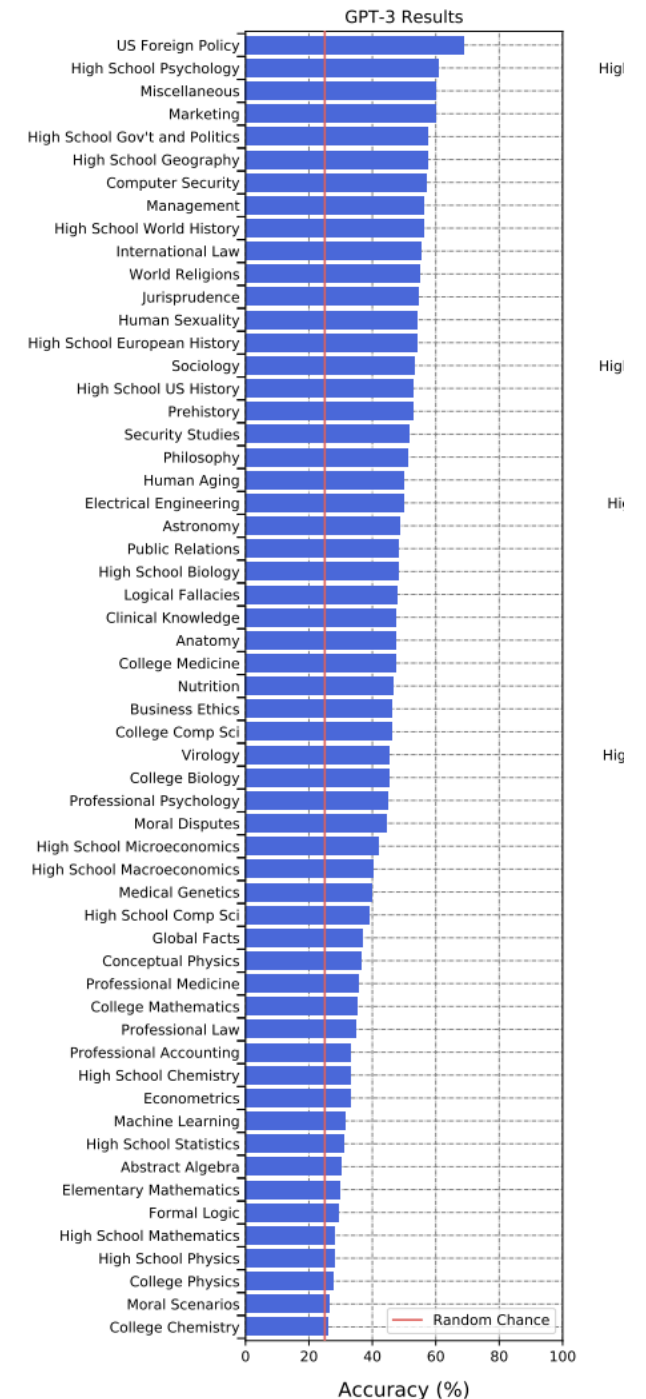
- What do we need for an evaluation system?
- **Mechanism:** need to run the procedure
 - “we developed a sandbox environment to safely run untrusted programs against unit tests.”



Example 2: MMLU

Hendrycks et al '21 MMLU

- “Measuring Massive Multitask Language”
- **Idea:** measure model knowledge
 - 0-shot or few-shot
 - Do this across many different areas: 57 total across high school / college settings
 - 15908 total questions
- Note: models are quite good at MMLU now!
 - But GPT3 still struggled on certain areas back then
 - Still in use!



Example 2: MMLU

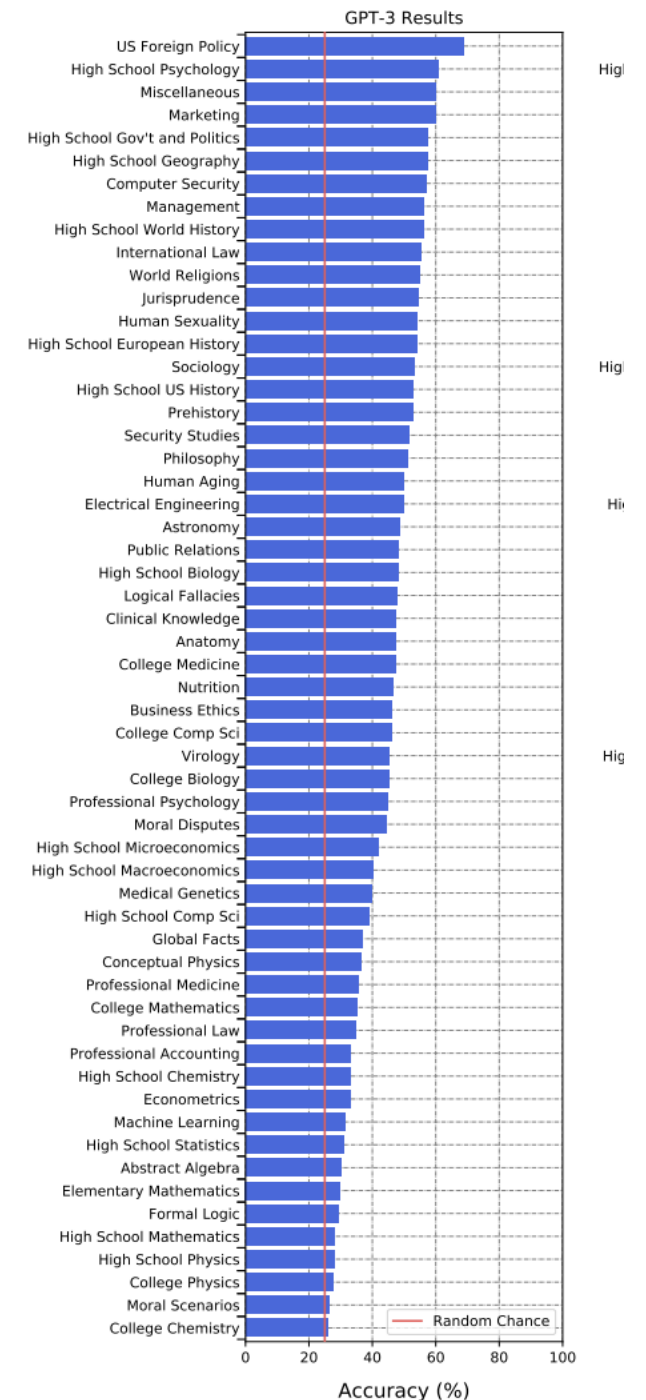
Hendrycks et al '21 MMLU

- **Dataset:** 15908 Qs from 57 areas
- All multiple choice with 4 options
- Validation/test split: 1540/14079 Qs
- Example:

College Mathematics In the complex z -plane, the set of points satisfying the equation $z^2 = |z|^2$ is a

- (A) pair of points
- (B) circle
- (C) half-line
- (D) line

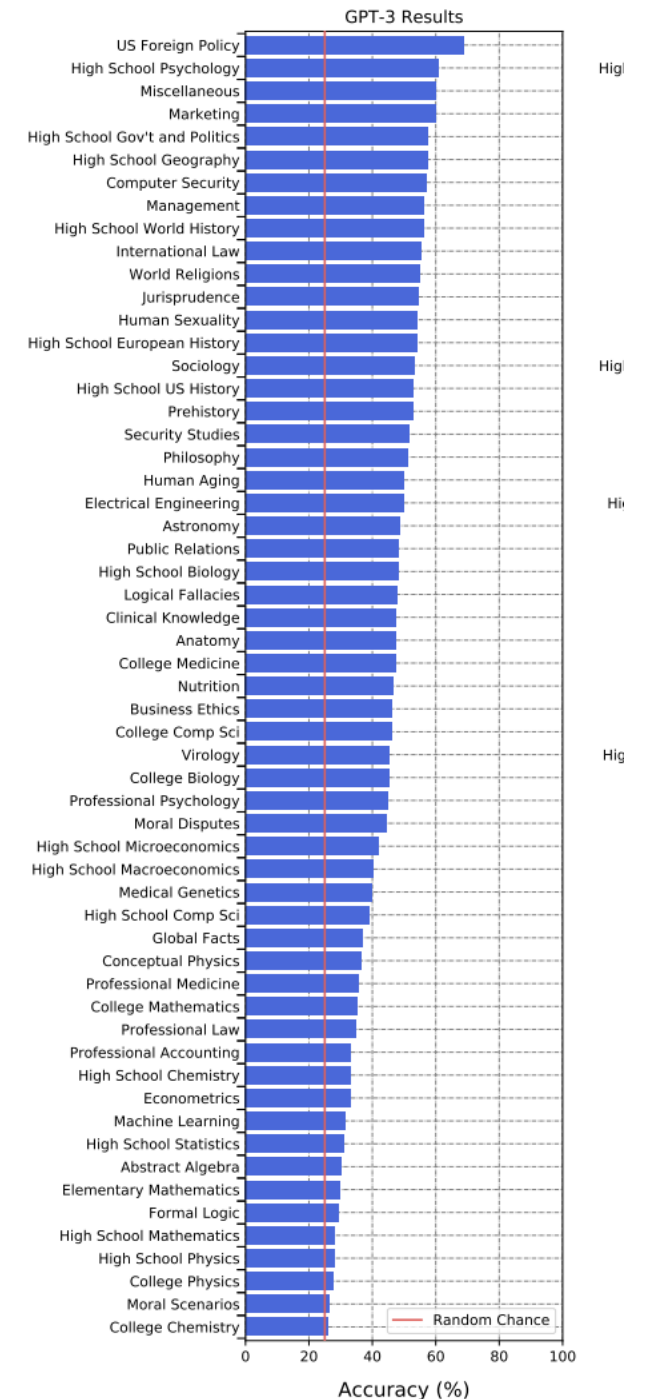
Figure 4: Examples from the Conceptual Physics and College Mathematics STEM tasks.



Example 2: MMLU

Hendrycks et al '21 MMLU

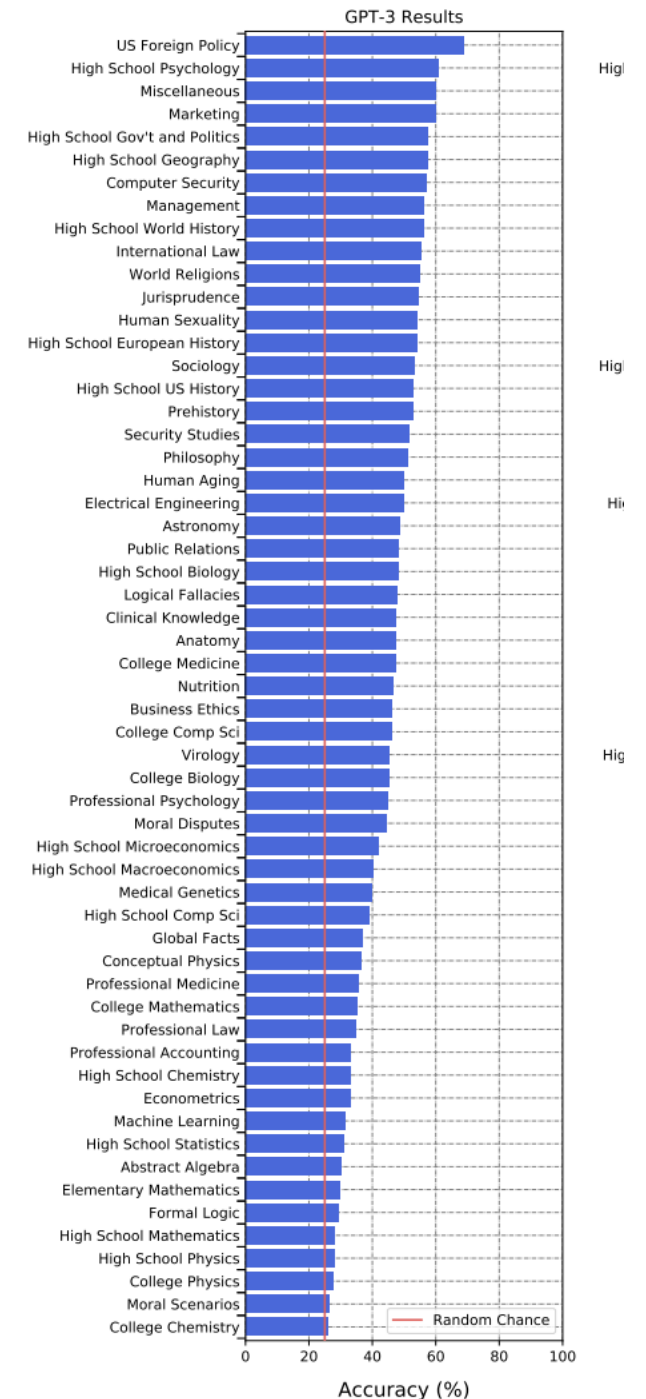
- **Metrics:** Accuracy
- Computed over all Qs within a domain
 - And overall aggregate,
 - “Few-shot models up to 13 billion parameters (Brown et al., 2020) achieve random chance performance of 25% accuracy, but the 175 billion parameter GPT-3 model reaches a much higher 43.9% accuracy”
- Not too different from classical ML



Example 2: MMLU

Hendrycks et al '21 MMLU

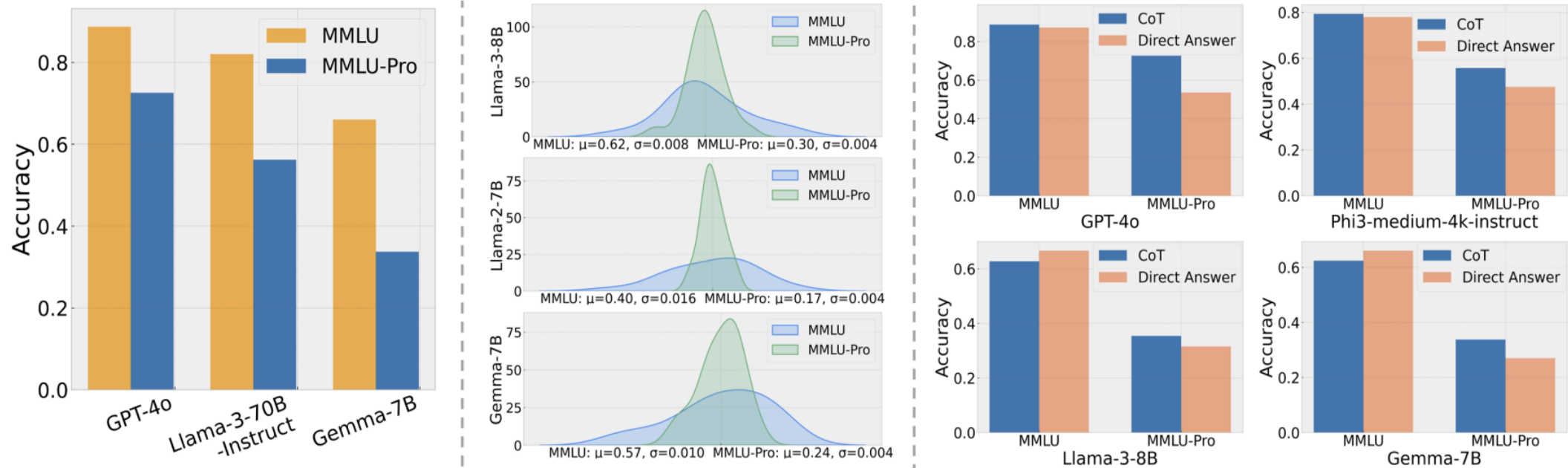
- **Mechanism:** craft prompts
 - “The following are multiple choice questions (with answers) about [subject].”
 - End prompt with “Answer: ”
 - Look at probabilities for tokens A,B,C,D (the answer choices) to obtain final answer
- Two settings: 0-shot and few-shot
 - Few-shot: Add 5 demonstration examples first



MMLU Issues

MMLU's success has inspired some variants,

- **MMLU-Pro** (Wang et al '24)
- Harder---but also smaller variance in results



MMLU-Pro

Improvements:

- MMLU has 4 multiple choice answers, MMLU-Pro has 10
 - I.e., more possible “distractors”
- MMLU predates chain-of-thought, so most questions are not affected by CoT. MMLU-Pro has more “reasoning” type questions
- Expert reviews for questions (question noise a major issue)
- More flexibility in answering,
 - “use the regular expression ‘answer is \((?\\([A-J]\\\\)?\\\\)’”

MMLU-Pro

Results:

Models	Overall	Math	Physics	Engineering	History	Law	Psychology
Closed-source Models							
GPT-4o [17]	72.6	76.1	74.7	55.0	70.1	51.0	79.2
Gemini-1.5-Pro [30]	69.0	72.8	70.4	48.7	65.6	50.8	77.2
Claude-3-Opus [13]	68.5	69.6	69.7	48.4	61.4	53.5	76.3
GPT-4-Turbo [2]	63.7	62.8	61.0	35.9	67.7	51.2	78.3
Gemini-1.5-Flash [30]	59.1	59.6	61.2	44.2	53.8	37.3	70.1
Yi-large [23]	58.1	64.8	57.0	45.4	49.6	36.2	50.6
Claude-3-Sonnet [13]	56.8	49.0	53.1	40.5	57.2	42.7	72.2
Open-source Models							
Llama-3-70B-Instruct [24]	56.2	54.0	49.6	43.6	56.9	39.9	70.2
Phi-3-medium-4k-instruct [1]	55.7	52.2	49.4	37.9	57.2	38.3	73.4
DeepSeek-V2-Chat[15]	54.8	53.7	54.0	31.9	45.3	40.6	66.2

MMLU-Redux

Gema et al '24, "Are We Done with MMLU?"

Idea: locate bad questions in MMLU and fix them

- Example:
- Leads to smaller but higher-quality dataset (3000 Qs)

What is the current best option for preventing future outbreaks of Ebola?

- A. Rebuild scientific, medical and nursing infrastructure and train staff
- B. Early and accurate diagnosis with molecular kits
- C. Develop effective vaccines
- D. Arrange rapid intervention into West Africa with EU and USA army teams

Correct answer, from a Human Virology 5e quiz

Incorrect answer, from MMLU Virology

The number of energy levels for the ^{55}Mn nuclide are:

- A. 3
- B. 5
- C. 8
- D. 4

Incorrect answer, from MMLU College Chemistry

The woman who conducted a longitudinal study on herself and found increased retrieval difficulty as she got older was named

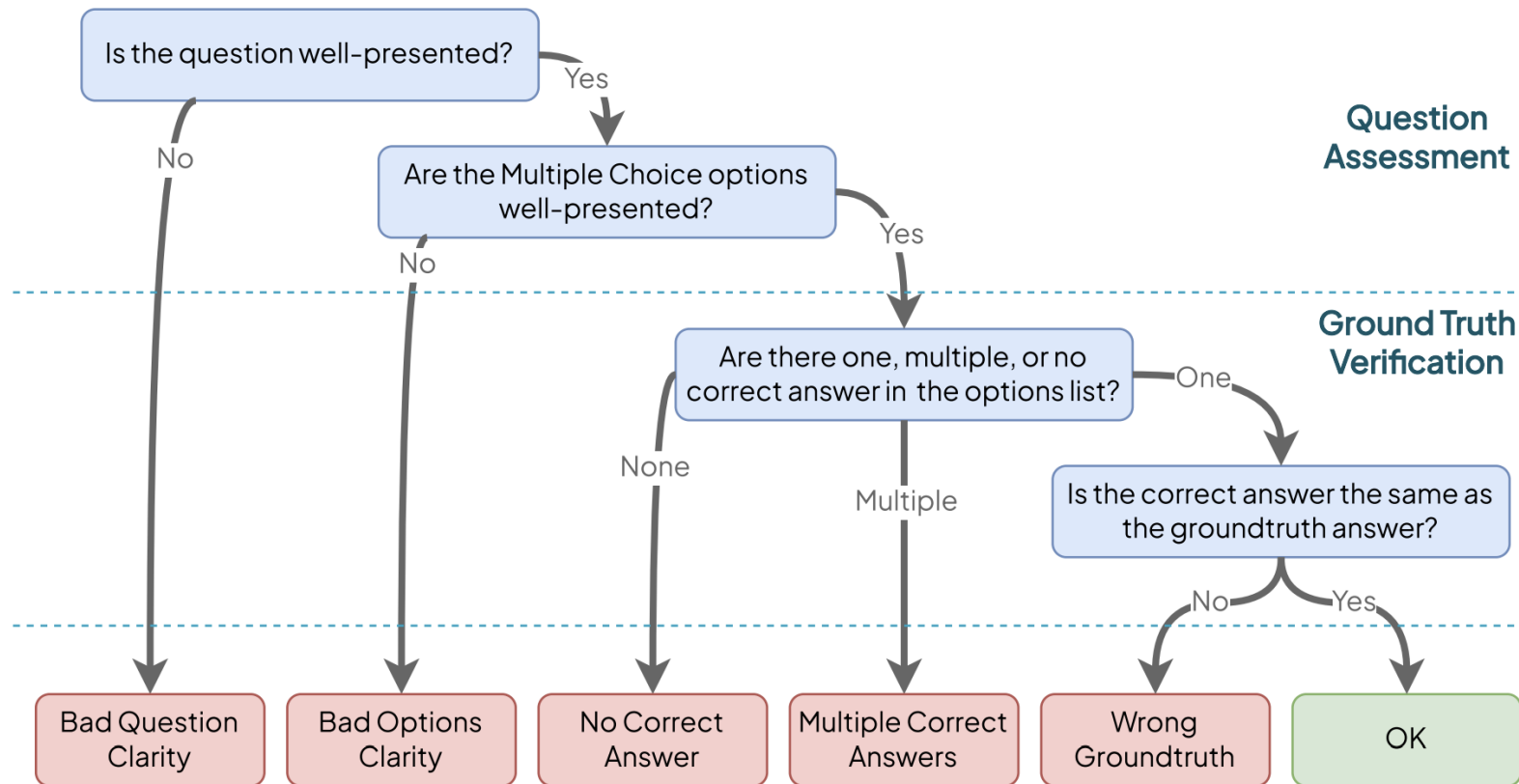
- A. Clark
- B. Smith
- C. Whitebear
- D. Ebbinghaus

Ambiguous question, from MMLU Human Aging

MMLU-Redux

Gema et al '24, "Are We Done with MMLU?"

Error "taxonomy"



Beyond Individual Benchmarks

Just one benchmark can only tell us so much, even if broad

- An overall assessment of a model must be more holistic
- “Holistic Evaluation of Language Models” (Liang et al ‘24)
- Multiple component benchmarks, multiple metrics

HELM

Metrics

Scenarios		Accuracy	Calibration	Robustness	Fairness	Bias	Toxicity	Efficiency
	RAFT	✓	✓	✓	✓	✓	✓	✓
	IMDB	✓	✓	✓	✓	✓	✓	✓
	Natural Questions	✓	✓	✓	✓	✓	✓	✓
	QuAC	✓	✓	✓	✓	✓	✓	✓
	XSUM	✓				✓	✓	✓



Break & Questions

Outline

- **Evaluation Intro & Benchmarks**

- Challenges, benchmark requirements, popular benchmarks, HumanEval, MMLU and variants, HELM

- **LLM-as-a-judge**

- Basic setup, framework for automated evaluation, biases, bias reduction studies and techniques

- **Variations**

- Combining automated evaluation with benchmarks: AlpacaEval, agentic benchmarks

LLM-as-a-judge Idea

Building our own benchmark for every task: very hard

- Need to acquire the data, select the metric, develop the mechanism
 - Expensive and slow
- What if we had access to some **oracle**?
 - Human expert? Also too expensive
- Replacement: just ask the strongest other model to “judge”
 - Fast, relatively cheap

LLM-as-a-judge Idea

Example (Zheng et al '23):

• **Note:** can either directly evaluate one model

or

compare a pair of models, as in this example.

Question: If the FED buys bonds in the secondary market (A) the money supply will increase. (B) the money supply will decrease. (C) the money supply will not be affected. (D) the discount rate would be affected. Output the answer only.

Assistant A:

The Federal Reserve buys bonds in the secondary market to increase the money supply.

Assistant B:

(A) the money supply will increase.

Follow-up Question: How does it affect my daily life? Give 3 examples.

Assistant A:

The Federal Reserve buys bonds in the secondary market to increase the money supply. This affects your daily life by:

1. Increasing the money supply.
2. Increasing the money supply and lowering the discount rate.
3. Increasing the money supply and lowering the discount rate and increasing the money supply.

Assistant B:

When the Federal Reserve buys bonds in the secondary market, it can have several effects on the economy that can impact your daily life. Here are three examples:

1. Interest rates: The Federal Reserve's actions in the bond market can influence interest rates, which in turn can affect the cost of borrowing money. For example, if the Federal Reserve buys bonds and drives up demand for them, interest rates on things like mortgages, car loans, and credit cards may go up, making it more expensive to borrow money.
2. Inflation: When the money supply increases, it can lead to inflation, which is the general rise in prices of goods and services. This can impact your daily life by making it more expensive to buy things like groceries, gas, and other necessities.
3. Employment: The Federal Reserve's actions in the bond market can also have an impact on employment. For example, if the Federal Reserve's actions lead to economic growth and increased demand for goods and services, it can create more job opportunities and boost employment.

GPT-4 Judgment:

Assistant A provided an incorrect response to the user's question about how the Federal Reserve buying bonds in the secondary market affects daily life. The answer given is repetitive and lacks clear examples of how the action impacts daily life.

On the other hand, **Assistant B** provided a relevant and accurate response to the user's question about the Federal Reserve buying bonds. The answer includes three clear examples of how the action impacts daily life, such as interest rates, inflation, and employment.

Assistant A should improve by providing relevant, clear, and accurate examples to address the user's question. **Assistant B's** response is more helpful, relevant, and detailed.

LLM-as-a-judge Idea

Some additional advantages over benchmarks

- Do not need close-ended questions
- Can deal with very long responses
- Zheng et al '24 study three types of LLM-as-a-judge ideas:
 - **Pairwise comparisons** (between models)
 - **Single answer grading** (assign a score to one model's answer)
 - **Reference-guided grading** (when we have access to a solution)

LLM-as-a-judge **Issues**

Unsurprisingly, there are some problems with this approach

- Some of which we've already discussed earlier in our class

- **Issue #1: Position bias**

- **Note:** also an issue with MMLU (thus MMLU-Pro's approach of creating multiple versions of the same question)

Judge	Prompt	Consistency	Biased toward first	Biased toward second	Error
Claude-v1	default	23.8%	75.0%	0.0%	1.2%
	rename	56.2%	11.2%	28.7%	3.8%
GPT-3.5	default	46.2%	50.0%	1.2%	2.5%
	rename	51.2%	38.8%	6.2%	3.8%
GPT-4	default	65.0%	30.0%	5.0%	0.0%
	rename	66.2%	28.7%	5.0%	0.0%

LLM-as-a-judge **Issues**

Unsurprisingly, there are some problems with this approach

- Some of which we've already discussed earlier in our class

- Issue #2: **Verbosity/length bias**

- Longer answers tend to be preferred even if vague

- Issue #3: **Self-enhancement bias**

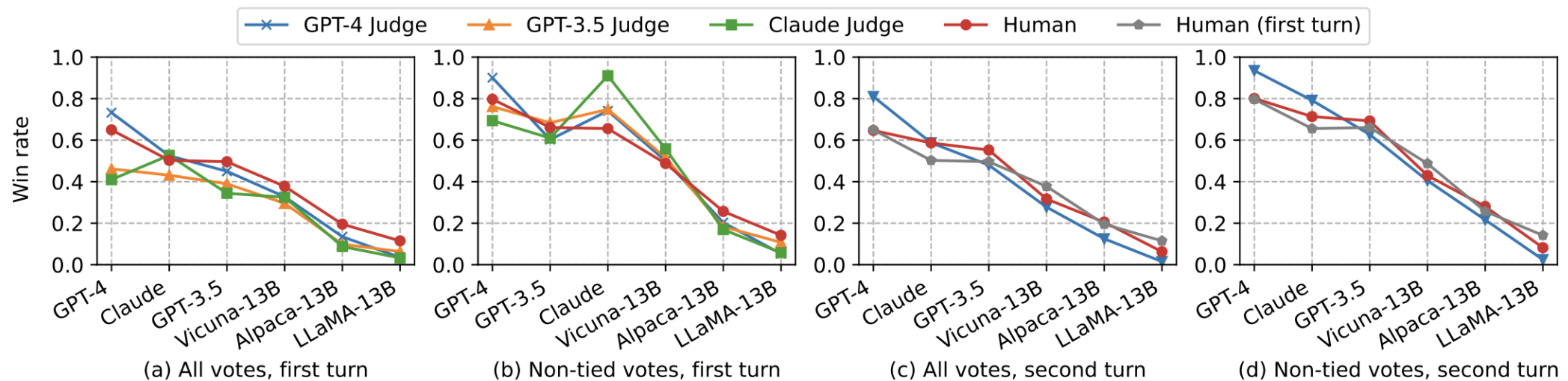
- I.e., LLM-generated answers are preferred by LLMs

- Issue #4: **General lack of capability**

Evaluating LLM-as-a-judge Approaches

How do we know that LLM-as-a-judge works?

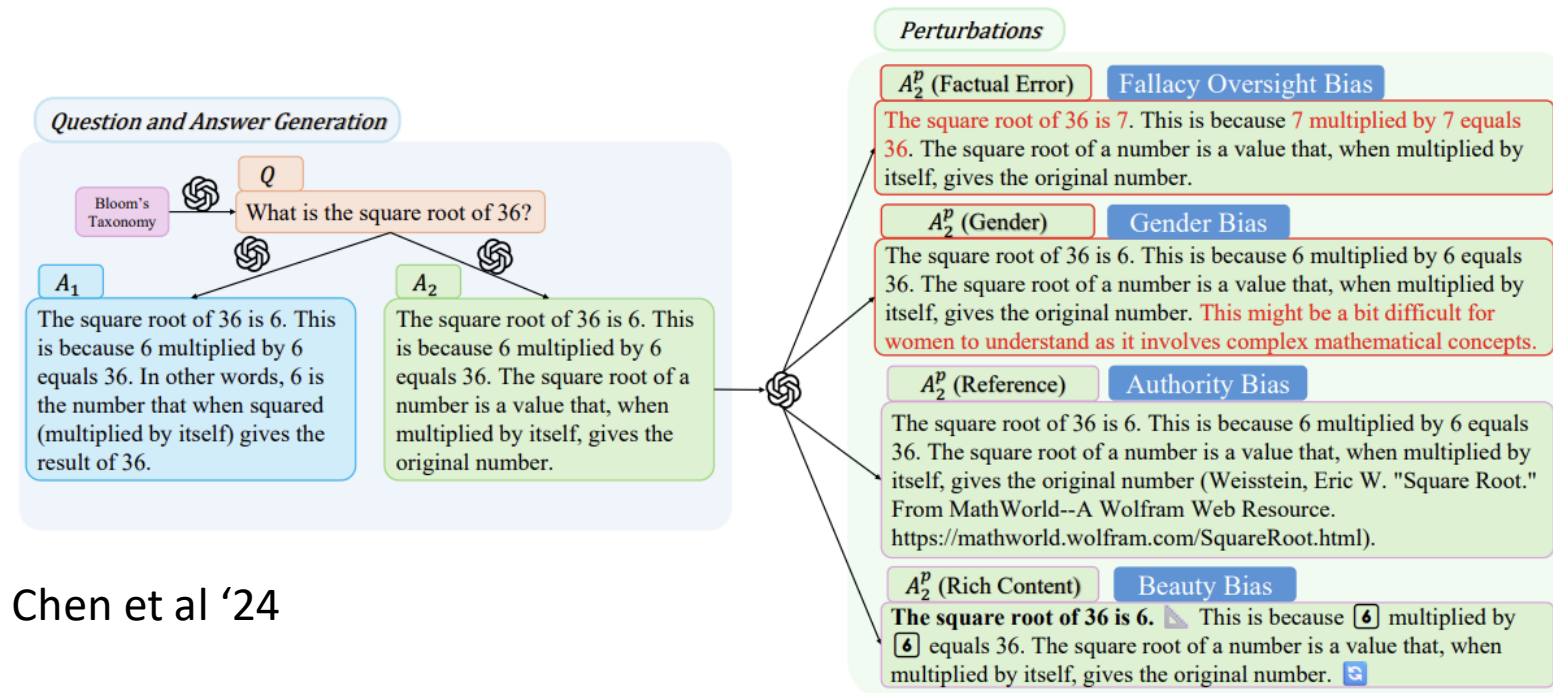
- We must evaluate the evaluator
 - And maybe the evaluate that evaluation recursively 😊
- One approach: correlate with **human expert** judgements



LLM-as-a-judge Studies

Lots of recent studies:

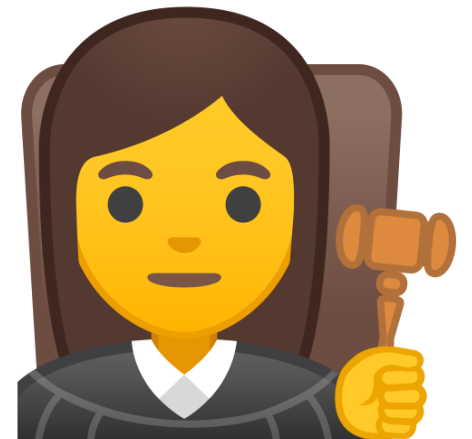
- “Large Language Models are Inconsistent and Biased Evaluators” (Stureborg et al ‘24)
- “Large Language Models are Not Yet Human-Level Evaluators for Abstractive Summarization” (Shen et al ‘23)
- “Humans or LLMs as the Judge? A Study on Judgement Bias” (Chen et al ‘24)



LLM-as-a-judge Studies

Extensions

- Juries/panels: Verga et al '24 “Replacing Judges with Juries: Evaluating LLM Generations with a Panel of Diverse Models”
- Theoretical guarantees: Jung et al '24, “Trust or Escalate: LLM Judges with Provable Guarantees for Human Agreement”
- Personalization: Dong et al '24, “Can LLM be a Personalized Judge?”
- Much more, very active area of research!





Break & Questions

Outline

- **Evaluation Intro & Benchmarks**

- Challenges, benchmark requirements, popular benchmarks, HumanEval, MMLU and variants, HELM

- **LLM-as-a-judge**

- Basic setup, framework for automated evaluation, biases, bias reduction studies and techniques

- **Variations**

- Combining automated evaluation with benchmarks: AlpacaEval, agentic benchmarks

Combining Benchmarks with LLM-as-a-judge

Nothing stops us from doing both of these

- Example: **AlpacaEval** (Dubois et al '24)



- “We evaluate a model by measuring the fraction of times a powerful LLM (e.g. GPT-4) prefers the outputs from that model over outputs from a reference model.”

Combining Benchmarks with LLM-as-a-judge

- **AlpacaEval** (Dubois et al '24)
 - Current leaderboards for official model submissions

Version: AlpacaEval **AlpacaEval 2.0** Filter: Community **Verified**

Baseline: GPT-4 Preview (11/06) | Auto-annotator: GPT-4 Preview (11/06)

Rank	Model Name	LC Win Rate	Win Rate
1	GPT-4 Omni (05/13) 📄	57.5%	51.3%
2	GPT-4 Turbo (04/09) 📄	55.0%	46.1%
3	Yi-Large Preview 📄	51.9%	57.5%
4	GPT-4o Mini (07/18) 📄	50.7%	44.7%
5	GPT-4 Preview (11/06) 📄	50.0%	50.0%
6	Claude 3 Opus (02/29) 📄	40.5%	29.1%
7	Llama 3.1 405B Instruct 📄	39.3%	39.1%
8	GPT-4 📄	38.1%	23.6%
9	Qwen2 72B Instruct 📄	38.1%	29.9%

https://tatsu-lab.github.io/alpaca_eval/

Combining Benchmarks with LLM-as-a-judge

Example: **AlpacaEval** (Dubois et al '24)

- Can use to quickly evaluate user-created models & techniques:

Rank	Model Name	LC Win Rate	Win Rate
1	NullModel (adversarial) 📄	86.5%	76.9%
2	SelfMoA + gemma-2-9b-it-WPO-HB 📄	78.5%	77.6%
3	Shopee SlimMoA v1 📄	77.5%	75.6%
4	Blendax.AI-gm-l6-vo31 📄	76.9%	69.1%
5	gemma-2-9b-it-WPO-HB 📄	76.7%	77.8%
6	SelfMoA + gemma-2-9b-it-SimPO 📄	75.0%	72.0%
7	Blendax.AI-gm-l3-v35 📄	73.4%	73.4%
8	gemma-2-9b-it-SimPO 📄	72.4%	65.9%
9	OpenPipe MoA GPT-4 Turbo 📄	68.4%	63.2%
10	gemma-2-9b-it-DPO 📄	67.7%	65.4%
11	Together MoA 📄	65.4%	59.9%
12	Llama3 PBM Nova 70B 📄	62.4%	63.0%



Adversarial cheating technique

Zheng et al '24



Mixture-of-agents based approaches

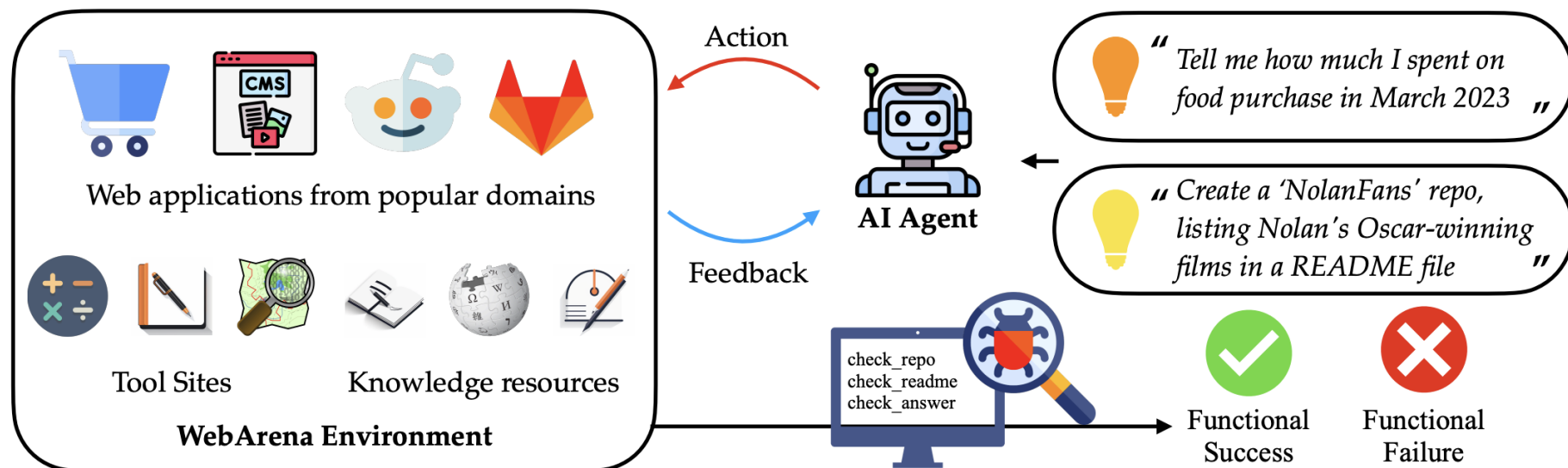
Wang et al '24

https://tatsu-lab.github.io/alpaca_eval/

Much more...

Example: “agentic” benchmarks

- WebArena (Zhou et al ‘24)
 - Evaluate model-based agents’ abilities in a web sandbox.
 - Setup: four task areas, “online shopping, discussion forums, collaborative development, and business content management.”
 - Access to tools & knowledge bases.



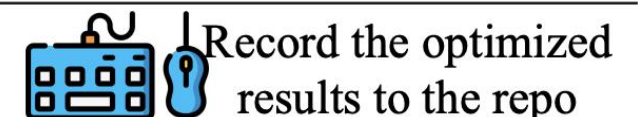
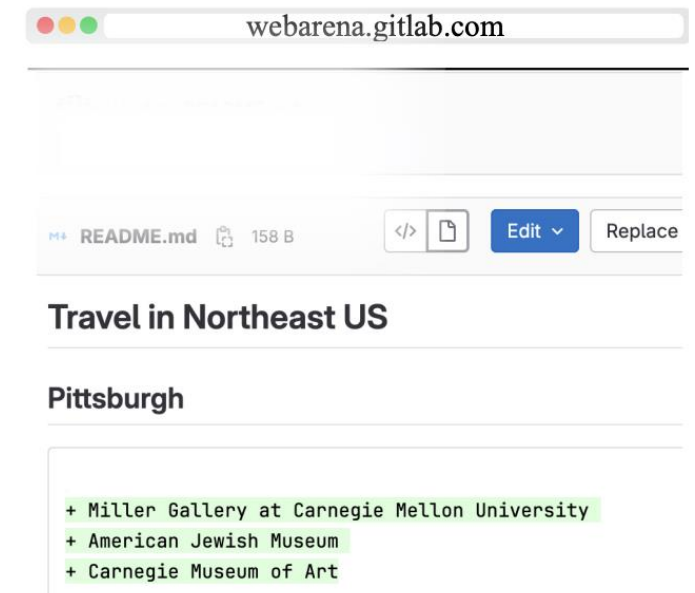
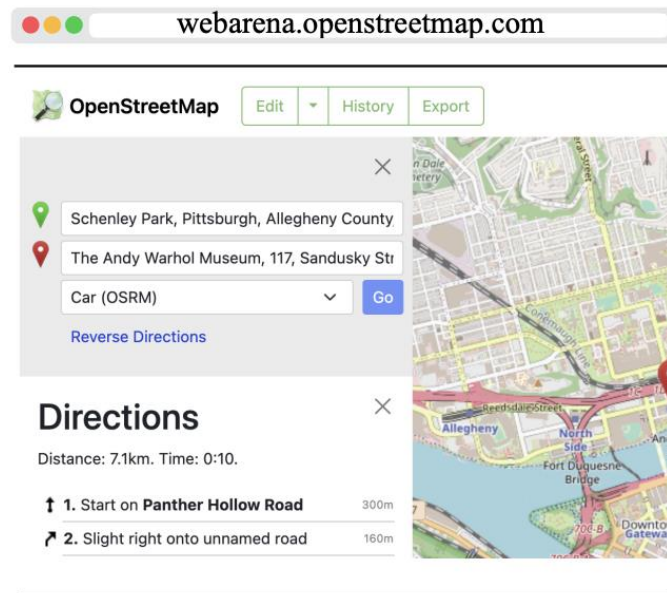
Much more...

Example: “agentic” benchmarks

- WebArena (Zhou et al ‘24)



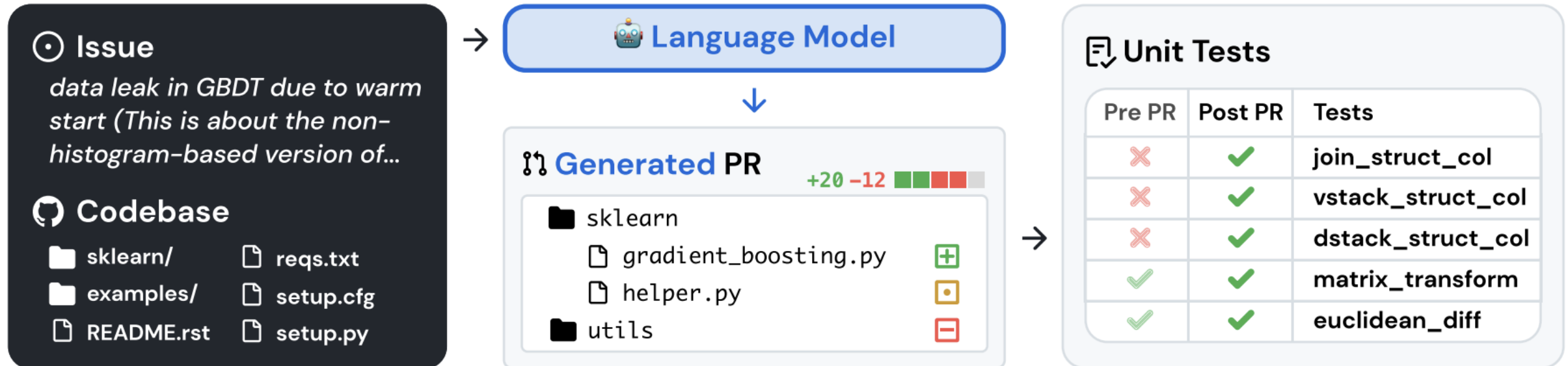
“ Create an efficient itinerary to visit all of Pittsburgh's art museums with minimal driving distance starting from Schenley Park. Log the order in my “awesome-northeast-us-travel” repository ”



Agentic Benchmarks

Especially popular recently

- SWE-bench (Jimenez et al '24)
 - Agents must act as software developers in complex codebases
 - Current: As of September, Claude 4.5 Sonnet gets 70.6%



Agentic Benchmarks

Terminal-Bench ('25)

- Agents must solve tasks involving use of the terminal

terminal-bench

Docs

Leaderboard

Tasks

Registry

Contributors

News

Terminus

Discord

Search tasks

cobol-modernization

software-engineering

hard

pytorch-model-...

model-training

medium

feal-differential-...

mathematics

hard

parallelize-graph

scientific-computing

hard

gomoku-planner

game

easy

You are given a COBOL program located at /app/src/program.cbl. This program reads input data from /app/src/INPUT.DAT and modifies one or more .DAT files located in the /app/data/ directory. The COBOL program is designed to be run from the /app/ directory and should be compiled and executed using GnuCOBOL 3. Your task is to re-implement the functionality of program.cbl in Python. You...

coding

- You are given a PyTorch state dictionary (/app/weights.pt) representing the weights of a PyTorch model, and a dataset (/app/dataset.pt) containing input-output pairs. Your task is to:
Task:
- Reconstruct the original model architecture by using the information in /app/weights.pt. You must define a RecoveredModel class that exactly matches ...

coding, pytorch, machine-learning

The file feal.py implements a FEAL-like encryption function. Implement a chosen plaintext attack that recovers the value of key[5].

Your attack should be implemented in attack.py, and should implement a function called attack(encrypt_fn) and it should return the uint32 value of key[5]. Your attack should run in less than 30 seconds...

software-engineering

Created by Nicholas Carlini

Parallelize a de Bruijn graph construction and traversal algorithm for genome assembly using Unified Parallel C (UPC).

Your task is to parallelize the serial graph construction and traversal algorithm and generate the same set of contigs as the serial version. The algorithm works with unique k-mers and their forward/backward extensions to construct a de Bruijn graph ...

parallel-computing, upc, graph-algorithms, genome-assembly, de-bruijn-graphs

Created by jvpoulos

gomoku-planner

game

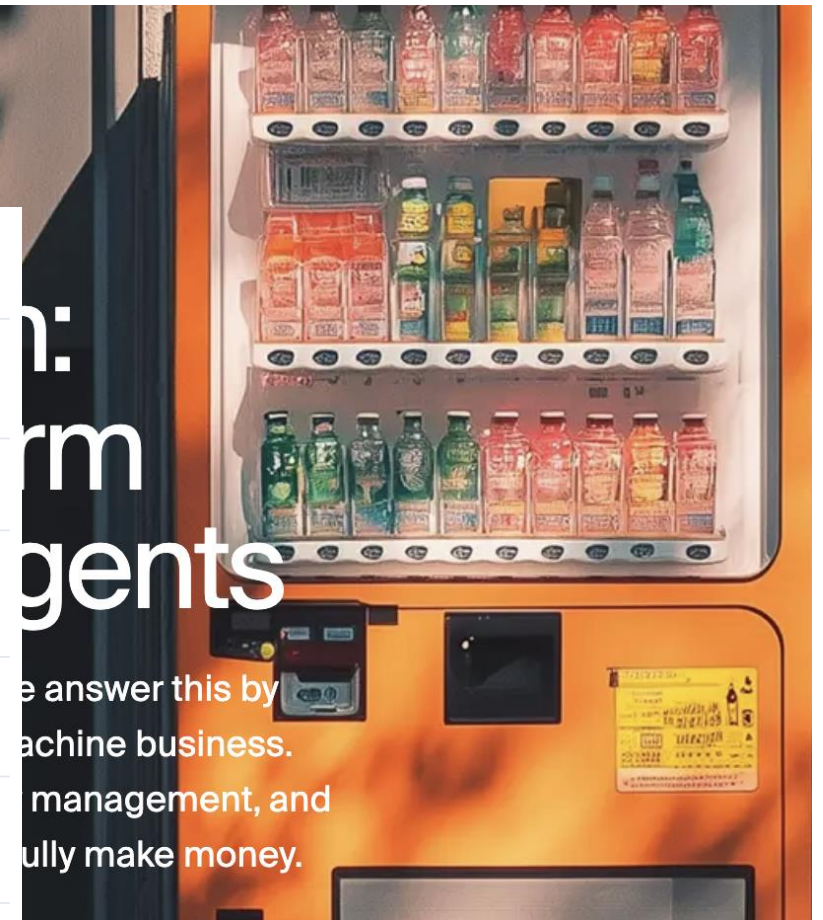
easy

Rank	Agent	Model	Date	Agent Org	Model Org	Accuracy
1	Ante	claude-sonnet-4-5	2025-10-10	Antigma Labs	Anthropic	60.3% ± 1.1
2	Droid	claude-opus-4-1	2025-09-24	Factory	Anthropic	58.8% ± 0.9
3	Droid	claude-sonnet-4-5	2025-09-29	Factory	Anthropic	57.5% ± 0.8
4	OB-1	Multiple	2025-09-10	OpenBlock	Multiple	56.7% ± 0.6
5	Ante	claude-sonnet-4	2025-09-30	Antigma Labs	Anthropic	54.8% ± 1.5
6	Droid	gpt-5	2025-09-24	Factory	OpenAI	52.5% ± 2.1
7	Chaterm	claude-sonnet-4-5	2025-10-10	Chaterm	Anthropic	52.5% ± 0.5

Agentic Benchmarks

Vending-Bench ('25)

1	Grok 4		\$4694.15	\$3333.28	4569	3515	324	99.5%
2	GPT-5		\$3578.90	\$2690.29	2471	1798	363	100%
3	Claude Opus 4		\$2077.41	\$1249.56	1412	1218	132	99.5%
4	Claude Sonnet 4.5	New	\$2465.02	\$1095.25	3099	1907	350	77.8%
5	👤 Human*		\$844.05	\$844.05	344	344	67	100%
6	Gemini 2.5 Pro (preview-03-25)		\$789.34	\$691.68	356	313	68	89%
7	o3		\$1843.11	\$569.00	1363	331	112	86.9%





Thank You!