Can Language Models Falsify?
The Need for Inverse Benchmarking

Shiven Sinha[I], Shashwat Goel[E,M], Ponnurangam Kumaraguru[I],
Jonas Geiping[E,M,T], Matthias Bethge[T,U,†], Ameya Prabhu[T,U,†]

I INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD
E ellis | INSTITUTE TÜBINGEN
M MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS
T Tübingen AI Center
U EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Oral

Science Progresses by Iteratively Improving Hypotheses...

Generate Solution... → **...Falsify Incorrect Solution**

Most current benchmarks measure this e.g. FrontierMath, RE-Bench

Benchmarks for LM Falsification capability are neglected...

But isn't verification easier than generation anyway?
For algorithmic coding problems, we find NO!

Traditional Benchmark

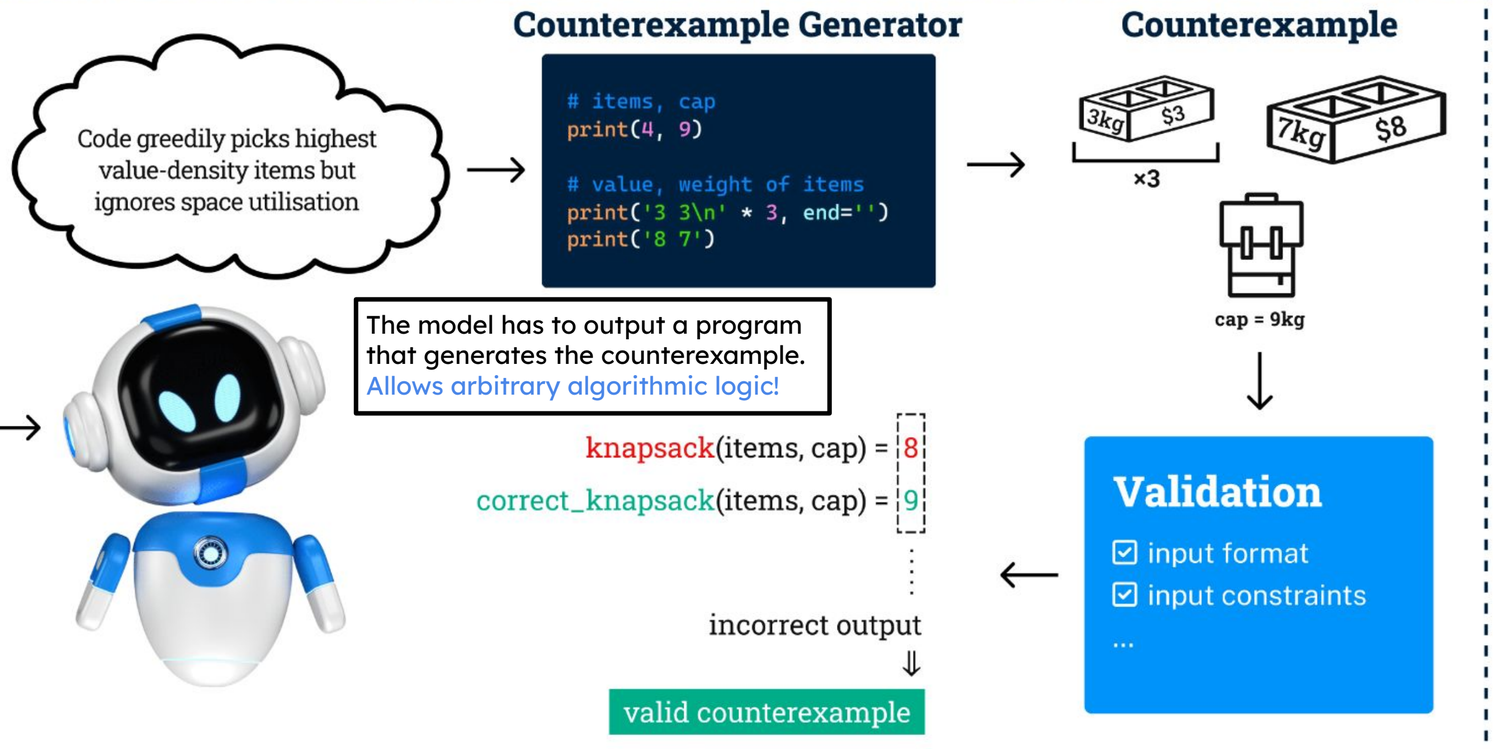
0-1 Knapsack
Given item values and weights, pick items to maximize total value without exceeding weight `cap`.

Input constraints: [...]

```python
def knapsack(items, cap):
  a = sorted(
    items,
    key=lambda x: x[0] / x[1],
    reverse=True)

  tot_v = 0
  for v, w in a:
    if cap ≥ w:
      cap -= w
      tot_v += v
  return tot_v
```

Inverse Benchmark



Our Benchmark
REFUTE**i**ng
Erroneous
Findings
Using
Targeted
Examples

- ✅ Fast Automated Evaluation
- ✅ Dynamically Updated (Live) Benchmark
- ✅ Challenging for Reasoning Agents
- ✅ Real-world Human Debugging Use-case
- ✅ Necessary for self-improvement

Main Results!

SOTA Reasoners Falsify <10% ...

... even if they can solve almost 50% problems

Method and Scaffold

Chat → Reasoner → Agent → +RandSearch → +Correct Solution

Prompting — **Tool-Use** — **Oracle**

Model	Solve Est.*	Prompt	ReAct Agent	Rand Search	w/ Correct Solution	Search w/ Soln
Deepseek V3	10.8	2.7	3.7	4.0	3.7	**15.1**
Claude 3.5	6.6	4.6	3.0	$$$	2.2	$$$
Claude 3.7	?	8.0	7.5	$$$	3.4	$$$
Flash 2.0 (Think)	?	2.1	2.5	3.7	2.5	7.7
Deepseek R1	44.0	5.8	**8.6**	4.0	4.6	9.9
O3-mini (high)	48.7	**9.3**	**8.6**	**8.3**	**9.3**	$$$

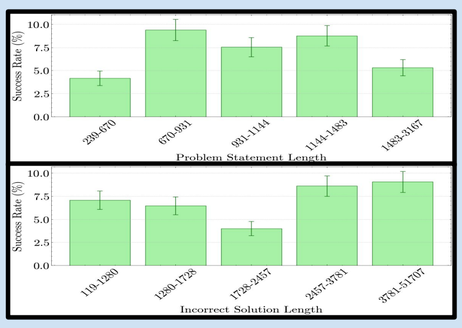
*computed based on reported elo

Reasoning helps

Knowing the solution is not enough

Early reasoners do not benefit from tool-use

Not a long-context length issue

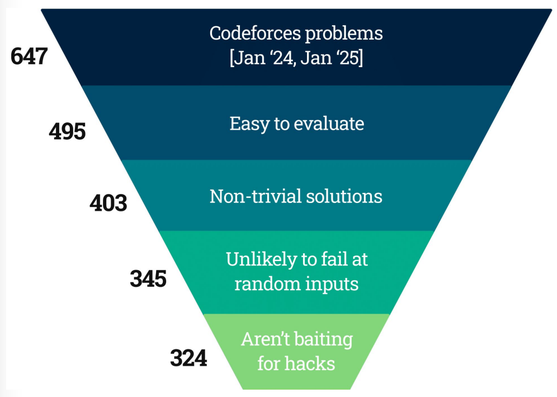


Explicitly prompting it to *"search against the correct solution using an input generator"* helps!

▶▶ Check out the preprint for **more details!**

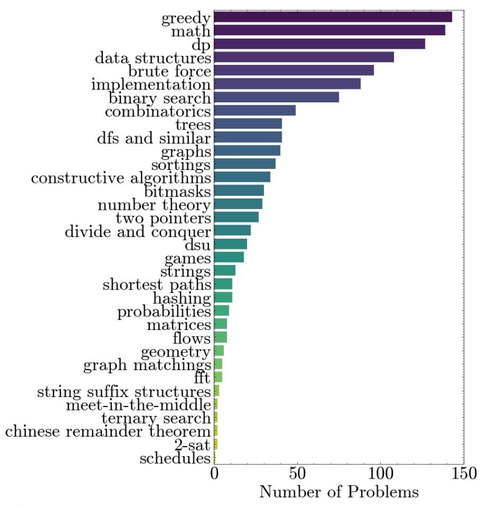
Methodology

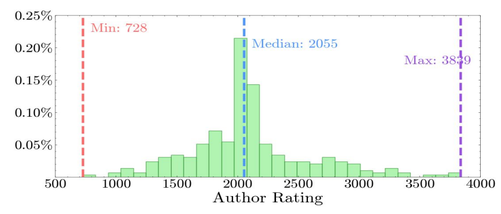
Data Curation Process

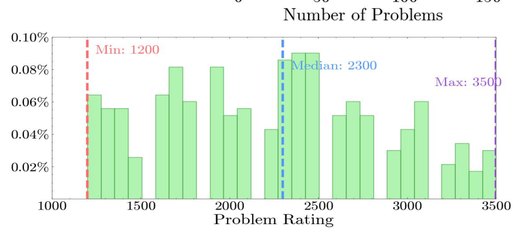


All Steps can be done automatically and quality-checked by Codeforces Grandmaster

Concepts Tested

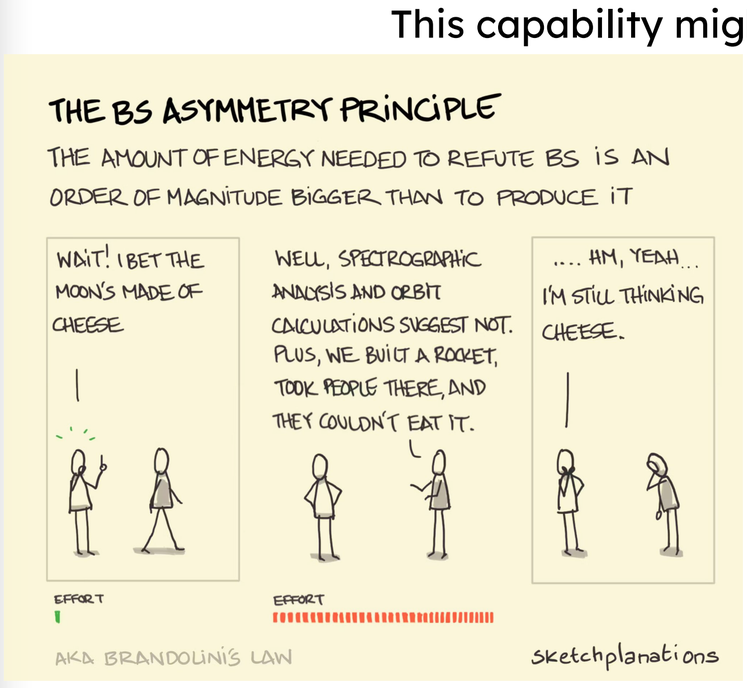






So in summary...

In the era of LLM-generated papers...
LLM Falsification will be a necessity...
This capability might be lagging behind!



Want to evaluate your model?
Our data, code is public!
Coming soon on LiveCodeBench