SCHOOL OF COMPUTING & DATA SCIENCE
The University of Hong Kong

HKU Musketeers Foundation Institute of Data Science
香港大學同心基金數據科學研究院

深圳河套學院
Shenzhen Loop Area Institute

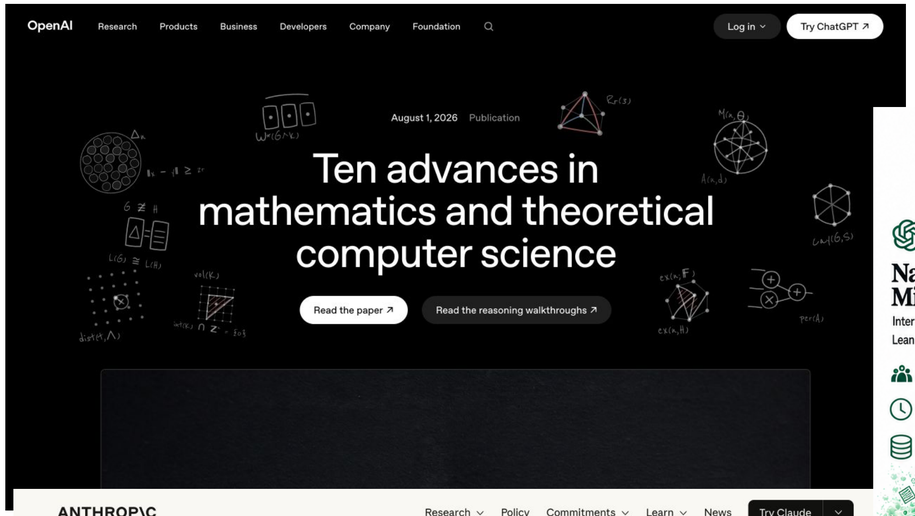
LLM for ML Theory at Scale: The VALG Agent and OpenTheoryBench

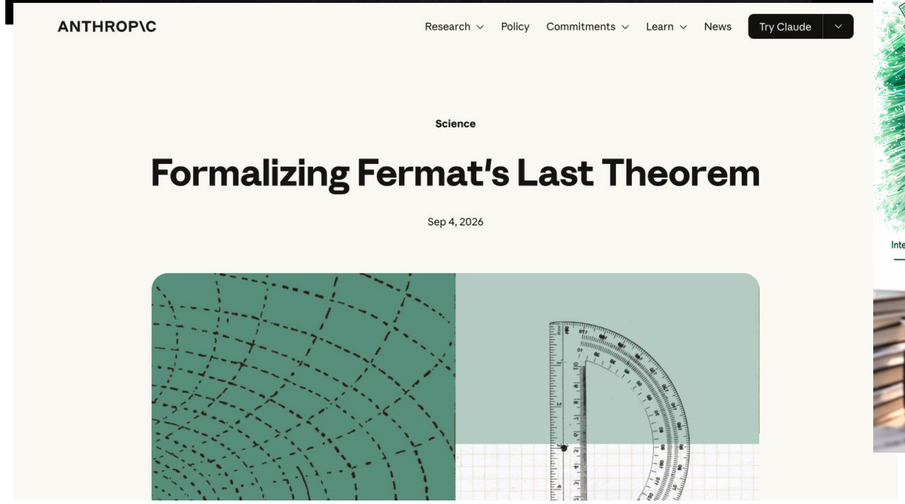
Difan Zou

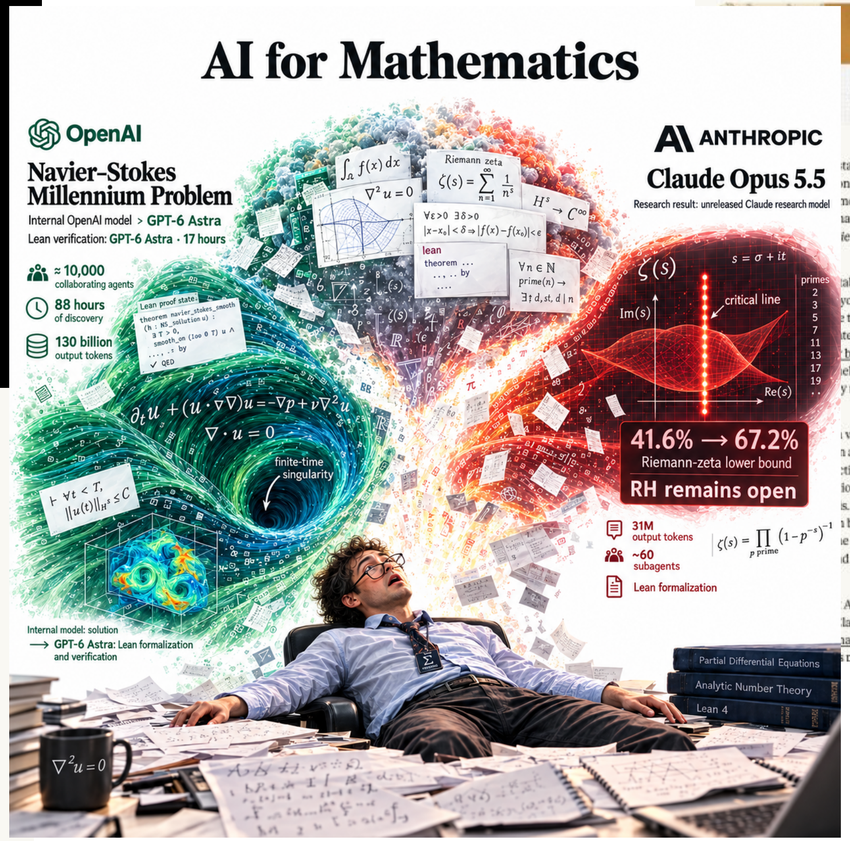
10.17, 2026, X-AGI 2026

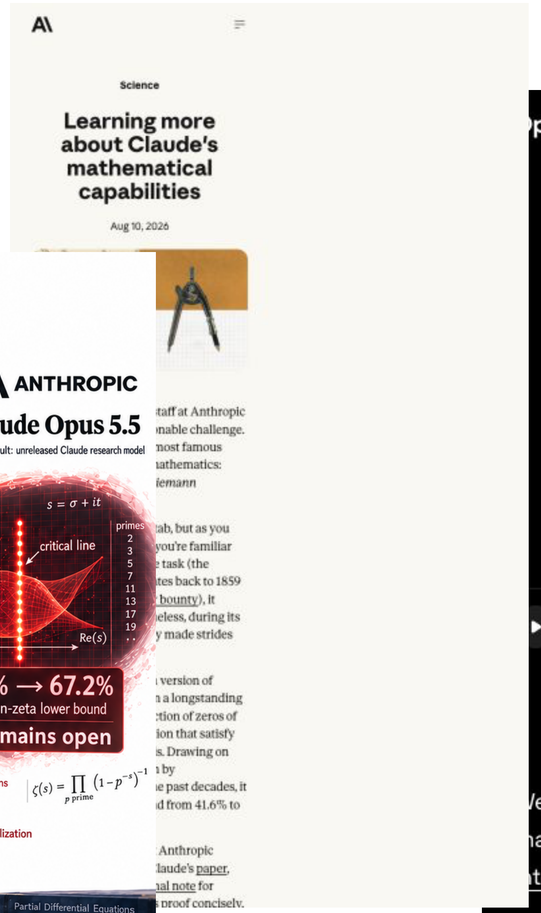
Joint work with many talented students (or incoming students) and colleagues in HKU ML Fundamental groups.

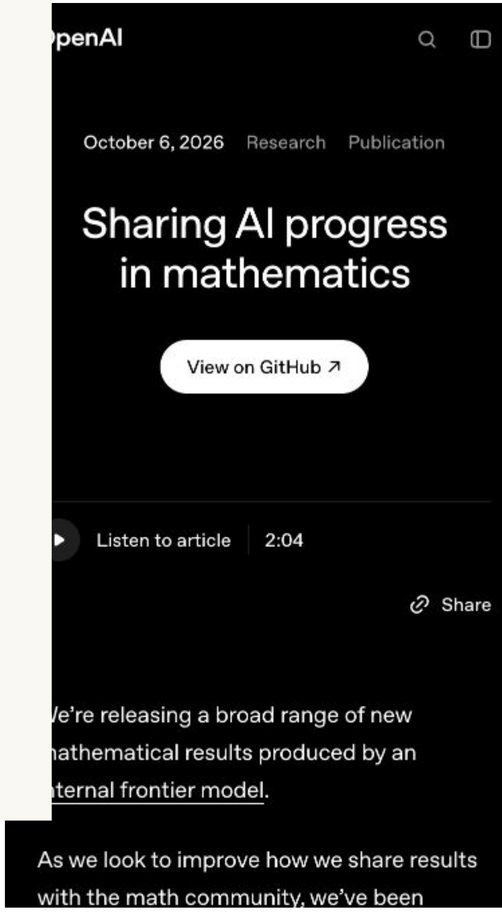
Recent Development of AI4Math











We have witnessed the “astonishingly” rapid development of AI4math in recent few months .

Research Paradigm of AI4math

- Researchers focus on multiple directions of AI4math
  - Building auto-formalizing techniques that translates natural language proof/ problem statement to formalized language, e.g., Lean.
  - Building Lean library for multiple areas of math problems.
  - Solving frontier math problems with the help of LLM, e.g., building agents.
  - Improving mathematical reasoning
  - Math education using AI
  - …

Currently, in the math area, most of the current works are working on solving problems either in natural language or using formal language.

Machine Learning Theory Research

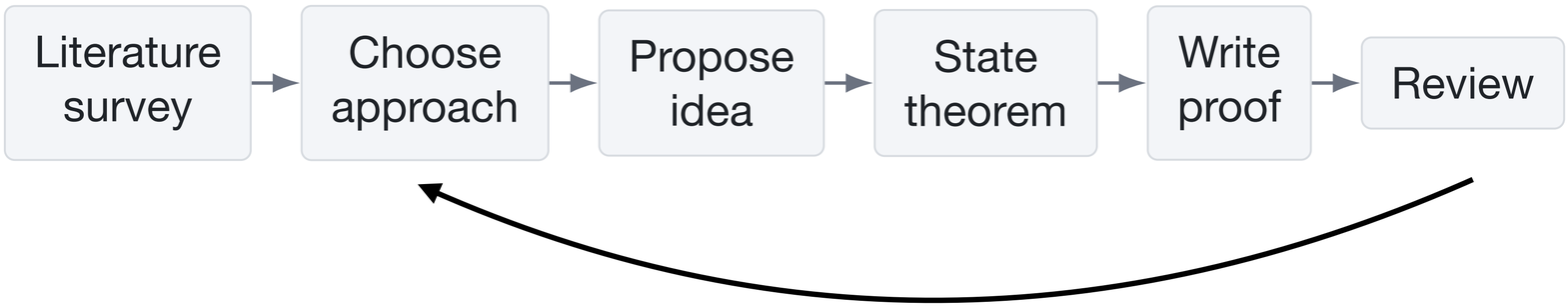
- In AI and ML, the theoretical research is similar to math research:
  - We formulate AI/ML problems using mathematical language, and develop rigorous proof to deliver guarantees on the algorithms, models, oracles, etc.
  - But ML theory is not only the proof, it has the following paradigm



Under [assumptions], for [model] on [task],
what is the [guarantee] (upper or lower bound), and is it tight?

Manual ML Theory Research

- In the past years, the pipeline of manual theoretical research is as follows:



  - Human cannot try different directions/approaches/ideas in short time.
  - Typically, if a researcher find that the formulated problem is difficult to solve, one may try to relax the setting rather than keep solving the problem.
  - When deriving the theoretical proof, human can have certain short-cut approaches to quickly verify whether the proof is correct or not.

As the LLM becomes more powerful, these aspects can be performed by LLM and we can build an agent to automatically perform the above procedures.

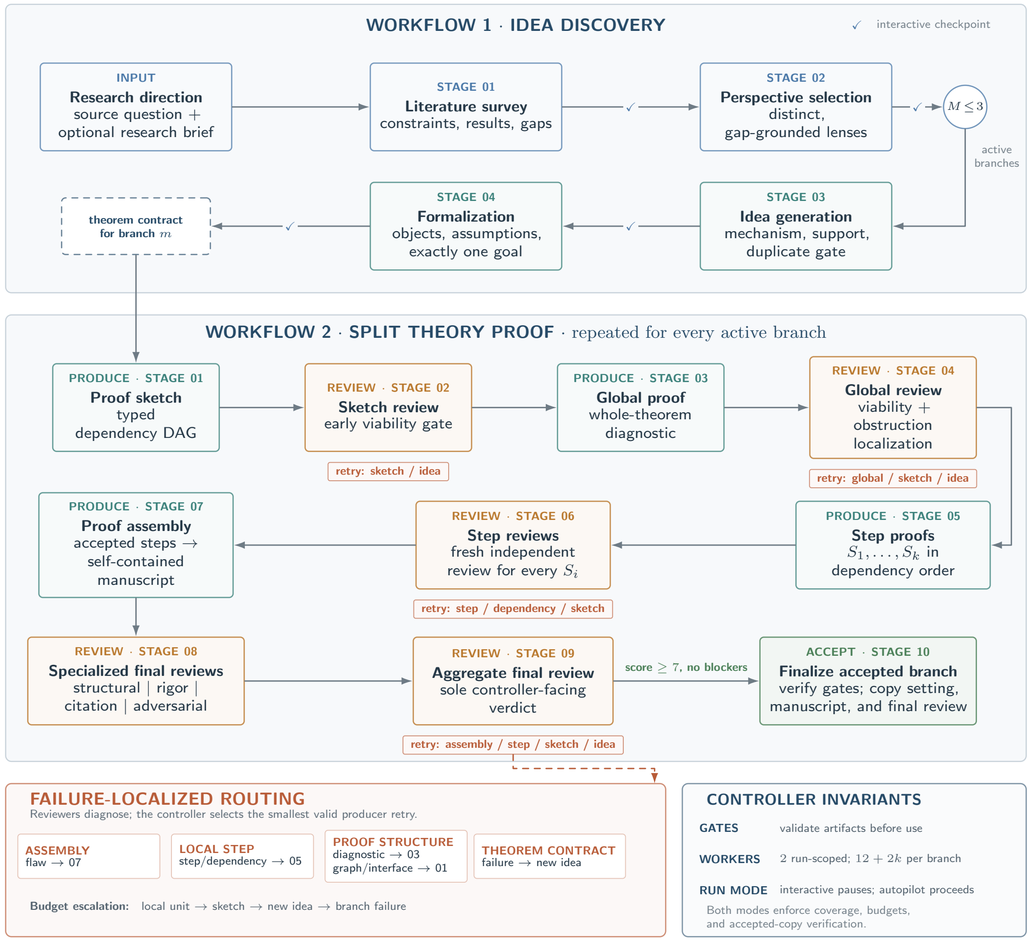
LLM for ML Theory Research

- Goal: start from an open problem/or a rough problem design, deliver the final setting, theorem, and proof.
  - For instance, *“I would like to study the how quantization-aware training affects the performance after fine-tuning. You may try to use a low-rank matrix factorization setting and develop the convergence or generalization guarantees”*
- In many cases, we only seek theory that is “feasible and explainable” rather than requiring the bound to be extremely sharp.

We can accept partial results as for ML theory research, the problem is allowed to be addressed under certain assumptions, which can be left for further exploration or verified via empirical results.

The VALG Agent

- Following the manual theoretical research pipeline, we develop VALG: the agent with verification, adaptive formulation for learning theory problem.

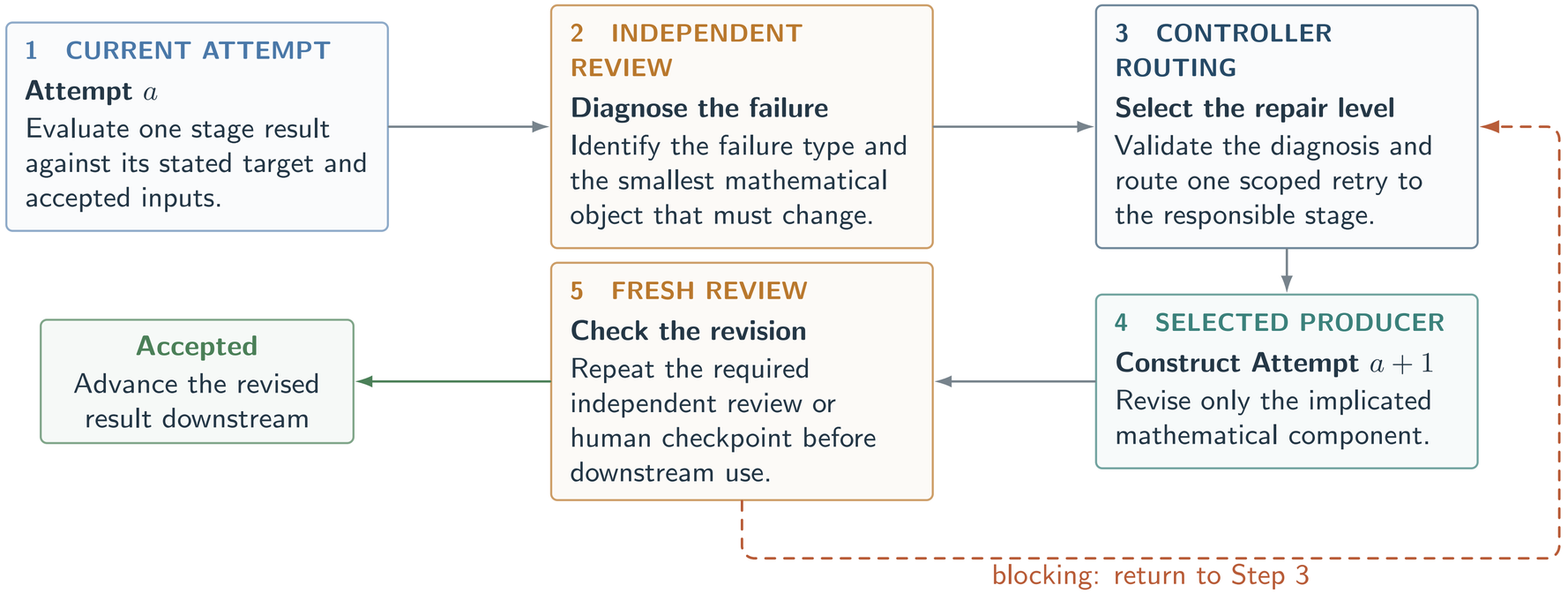


- Workflow 1: find the idea and problem formulation
  - Literature survey provide multiple perspectives (Target, model, data assumption, regime, algorithm)
  - Depth-focus idea generation in parallel.
  - Formalize the theorem statement with notation, assumptions, and goal.
- Workflow 2: complete the proof
  - first deliver the proof sketch and create a DAG
  - check the sketch to see if this problem can be solved
  - Perform the step proof and then assemble them
  - Final reviewers: structure, rigor, reference, adversarial.
  - Accept if all reviews pass with high confidence.

Zhang et al., VALG: An Agentic System for ML Theory Research, arXiv 2026

The VALG Agent

- When the proof fails, a human researcher would try to go back to modify the problem setup, to see if other model, algorithm, or assumption can give feasible results.



The proof failure is sent to multiple reviewers, they will find the smallest repair targets of the problem using an attempt-revise loop.

Results on COLT 2026 Open Problems

Problem	Subproblem	Persp.	Ideas tried	Branch outcome	Elapsed (h)	Progress	Rank
P1: Tensor ALS	Upper bound	1	7	**Idea 7**	76	6.50	1
		2	14	Exhausted	–	0.00	2
	Lower bound	1	2	**Idea 2**	8	2.25	2
		2	2	**Idea 2**	11	3.00	1
		3	3	**Idea 3**	38	3.00	3
P2: 1-bit mean estimation	Non-adaptive	1	1	**Idea 1**	13	**10.00**	2
		2	2	Interrupted	–	0.00	3
		3	1	**Idea 1**	9	**10.00**	1
P3: Deep vs. linear learning	SGD	1	3	**Idea 3**	12	2.75	2
		2	2	**Idea 2**	10	2.25	3
		3	2	**Idea 2**	12	5.25	1
	SQ	1	3	**Idea 3**	15	2.75	2
		2	2	**Idea 2**	8	3.00	1
		3	2	**Idea 2**	9	2.25	3
P4: Private PAC learning	Sample complexity	1	3	**Idea 3**	48	6.50	1
		2	4	**Idea 4**	91	6.25	2
		3	1	**Idea 1**	28	5.25	3
	Class existence	1	5	**Idea 5**	28	3.00	2
		2	2	**Idea 2**	25	5.50	1
		3	6	Exhausted	–	0.00	3
P5: Online optimization	Polynomial	1	1	**Idea 1**	12	5.75	2
		2	1	**Idea 1**	24	7.00	1
	Pfaffian	1	1	**Idea 1**	30	**10.00**	1
		2	1	**Idea 1**	40	**10.00**	2
		3	1	**Idea 1**	31	6.75	3

- 9 subproblems from 5 papers, we proposed 22 finalized theorem candidates and solved 2 subproblems.
- For other theorems, even though the problem has not been fully solved, but we can solve the relaxed version.
- We also find out that the fully solved problems are typically addressed within **one iteration**.

Examples

- Original question

Open question. Is there a universal $c > 0$ such that, for $r < k \leq r^{1+c}$, ALS, gradient descent, or another iterative method converges with constant probability to a strictly positive objective value?

$$\left\| T - \sum_{i=1}^{k} x_i \otimes y_i \otimes z_i \right\|_F$$

- Our solution

Assumption 4.9 (Fixed ambient dimension). $n \geq 8r^{5/4}$.

Assumption 4.10 (Superlinear algorithmic rank). $r < k \leq r^{5/4}$; *equivalently, the exponent is* $\alpha = 1/4$.

Assumption 4.11 (Uniform arbitrary deterministic bases). *The deterministic base triple is unrestricted, and the claim is pointwise uniform over all such triples.*

Assumption 4.12 (Independent Gaussian smoothing). $q > 0$ *is fixed,* $\rho = r^{-q}$, *and the* $3r$ *smoothing vectors are mutually independent with covariance* $(\rho^2/n)I_n$.

Assumption 4.13 (Shared target and independent starts). *The two initialization triples are mutually independent and independent of smoothing, while both methods use the same realized* T.

Theorem 4.2 (Fixed-span positive limiting objective). *Under Assumptions 4.9–4.13, with probability at least* $1/4$ *over smoothing and both starts, simultaneously for* $M \in \{\mathrm{cALS}, \mathrm{cGD}\}$,

$$\|(I - P_{\mathcal{H}_M})T\|_F^2 \geq \frac{3}{4}\|T\|_F^2, \qquad F_M(t) \geq \frac{3}{8}\|T\|_F^2 \quad \text{for every } t \geq 0,$$

and $F_M(t)$ *has a finite scalar limit satisfying*

$$\lim_{t\to\infty} F_M(t) \geq \frac{3}{8}\|T\|_F^2.$$

The statement holds for every admissible r, n, k *and is pointwise uniform in the deterministic base triple.*

Relaxed part: the original question asks lower bound for all iterative methods, our theory can only hold for a certain class of algorithms including constrained ALS and gradient descent.

Examples

- Original question

Open question. *Is there a constant C such that for all $\mathcal{H} \subseteq \{\pm 1\}^{\mathcal{X}}$ over $\mathcal{X} = \{\pm 1\}^n$, and $\epsilon < 1/4$, if there exists a fully connected ReLU network with S parameters in total, stepsize η and number of steps T, such that for every input distribution $\mathcal{D}$, every $h^* \in \mathcal{H}$, SGD yields expected error $\mathbb{E}\mathcal{L}_{\mathcal{D},h^*}(\hat{h}) \leq \epsilon$ (expectation over the initialization and SGD sampling), then* $\mathrm{dc}(\mathcal{H}) \leq C \cdot TS$.

$\mathrm{dc}(\cdot)$ denotes the dimension complexity.

- Our solution

Assumption 4.32 (**Antipodally odd target class**). *Every $h \in \mathcal{H}$ satisfies $h(-x) = -h(x)$ for all $x \in \mathcal{X}$.*

Assumption 4.33 (**Strict high-accuracy regime**). *The accuracy obeys the strict inequality $2\varepsilon < 1/(n+1)$.*

Assumption 4.34 (Universal source success). *The width, stepsize, and horizon are fixed before the distribution and target, and for every $\mathcal{D}$ and $h \in \mathcal{H}$,*

$$\mathbb{E}_\omega\left[\mathcal{L}_{\mathcal{D},h}(G_\omega)\right] \leq \varepsilon,$$

where expectation is joint over the independent Gaussian initialization and fresh one-sample SGD draws.

Theorem 4.7 (Exact identity representation in the odd high-accuracy regime). *Under Assumptions 4.32, 4.33, and 4.34, the identity map exactly represents the class:*

$$\forall h \in \mathcal{H}\ \exists w_h \in \mathbb{R}^n\ \forall x \in \mathcal{X}, \qquad \mathrm{sign}_{s_0}(\langle w_h, x\rangle) = h(x).$$

Hence the distribution- and target-independent law $\mathcal{P}_{\mathrm{id}} = \delta_{\varphi_{\mathrm{id}}}$ succeeds with probability one for every $(\mathcal{D}, h)$, and

$$\mathrm{dc}^{1/2}(\mathcal{H}) \leq \mathrm{dc}(\mathcal{H}) \leq n \leq S \leq TS.$$

Relaxed part: the original question is proposed for a general class of h, our solution requires h to be within an odd target class.

Beyond Agentic Research

- Building agent for ML theory is becoming less helpful when base model becomes stronger:
  - Our finding is that GPT 5.6 sol or GPT 6 astra generally do not need multi-loop structure to solve the well-defined problem.
  - Our result also shows that, most of the full-resolved problems are addressed within one iteration.
- Besides, solving single problems or several open problems cannot fully reveal the “intelligence” or capability of the LLM.

We need to build a benchmark to systematically evaluate the capability of LLM for frontier ML theory research.

Starting Point: Human-written Papers

- Our starting point lies from extracting recent ML theory papers from arXiv

arXiv category	Papers	Share
stat.ML	1,583	47.7%
cs.LG (cross-listed)	1,448	43.6%
math.OC (ML-relevant)	289	8.7%
Total	**3,320**	100%

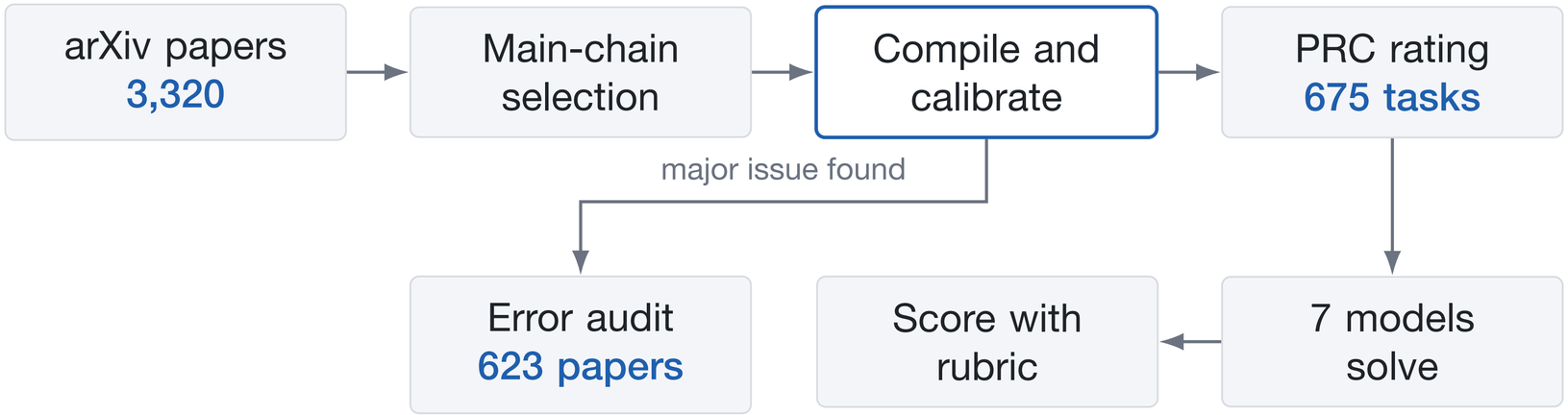
- Papers from 1 January to 27 July 2026.
- Recent papers plus closed-book, tool-free solving reduce contamination.
- Final benchmark: **675 tasks**, each with difficulty assigned in advance.

These papers provide human-written proofs for well-defined problems, which at least can help evaluate the gap between LLM and human researchers.

Zhou et al., xxxx, xxxx (TBD)

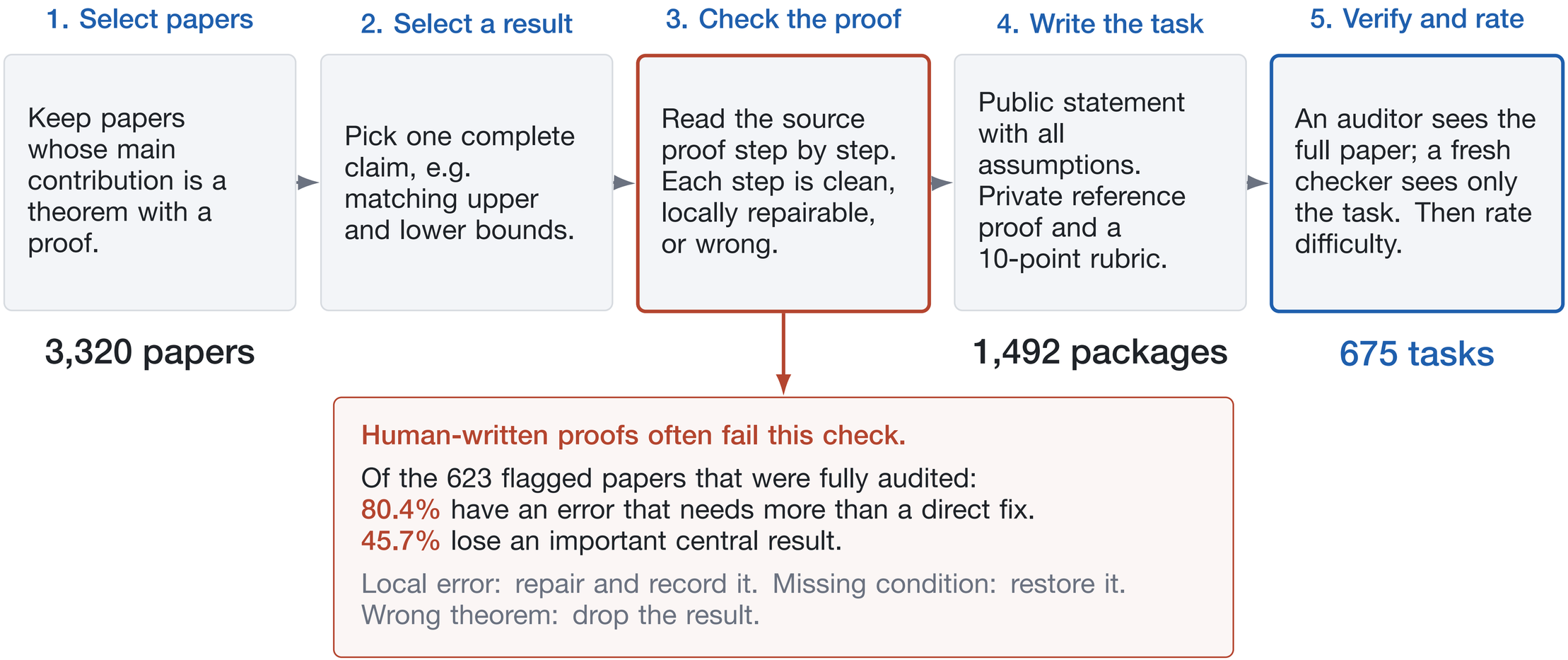
Pipeline of the Benchmarking

- Key pipeline of building the benchmark from the raw arXiv papers



- We use separate models to perform the data filtering, data labeling, and scoring.
- Every decision stage was tuned on new sets of 10 papers until model judgments largely agreed with human judgements.

Data Pipeline: From Paper to Task



arXiv paper cannot be fully trusted and many of them have proof flaws and issues.

Problems in Human-Written Papers

- We perform some simple statistical analysis on the flawed papers.

Error mechanism

	All	Level ≥ 2
Invalid inference step	50.5%	61.0%
Calculation error	20.2%	8.9%
Missing proof step	15.1%	19.7%
Invalid object in statement	14.2%	10.4%
Findings	3,919	1,658

Repair difficulty, per finding

Fix = the smallest valid correction that keeps the most of the original result.

Level	Fix requires	Share
1 Direct correction	recompute a formula, add the condition the proof already needs, or restrict to the proved range	57.7%
2 Local repair	a new local lemma or case analysis; same proof strategy	37.7%
3 Core revision	a new argument for a key step	4.6%
4 Reconstruction	rebuild the main mechanism or withdraw	0.0%

80.4% of these papers have at least one finding at level ≥ 2, i.e., an error that needs a new argument.

Problems in Human-Written Papers

- We perform some simple statistical analysis on the flawed papers.

Paper impact: after all fixes are applied, how much of the main contribution (abstract, flagship results) is still supported?

Level	What is still supported	Papers	Share
1 Preserved	main guarantees still hold	75	12.0%
2 Qualified	core result kept; some scope or a secondary result lost	263	42.2%
3 Central result	a central result loses support; other contributions remain	221	35.5%
4 Main claim	the claim in the title or abstract loses support	64	10.3%
Total		623	100%

Remark: impact is separate from repair difficulty: changing a wrong $O(n)$ to $O(n^2)$ is a level-1 repair, but can remove the paper's main advantage.

Difficulty Evaluation of the Problem

- We perform the difficulty evaluation based on **Proof Reconstruction Complexity** (PRC), which is designed by evaluating the hardness of roadmap, subgoals, techniques, and combinations.

  Assigned before any model is run.

  1. List 3–7 substantive proof obligations.
  2. Score four dimensions, 0–3 each:
     - S: strategy discovery
     - D: subgoal and dependency structure
     - T: technical execution
     - C: combining different techniques
  3. Subtract for exposure: if the task reveals part of the proof route, lower the score or rewrite the task.

Level	Score (0–12)	Tasks	Share
PRC0	0–3	41	6.1%
PRC1	4–6	259	38.4%
PRC2	7–8	182	27.0%
PRC3	9–10	85	12.6%
PRC4	$\geq$11, S=3, T=3	108	16.0%
Total		675	100%

All of the evaluations are performed by LLM-as-a-judge with expert-written prompts. Human judgement is applied on a subset of problems to ensure the feasibility of LLM judgement.

The Ability of Natural Language Proof Check

- In most ML theory domain, there lacks auto formalization tool as well as the formal verification tool, yet, we found that natural language proof check is sufficient.
- We testify 20 ML-theory results with Lean-verified proofs (AI4SLT, TestingLowerBounds), translated to English. We then consider 3 versions with one inserted error (missing step, invalid inference, calculation), to verify that whether they can be identified.

	GPT-6 Astra	DeepSeek V4 Pro	GLM 5.3
Inserted errors found (/60)	60	58	57
with correct category and reason	60	54	55
Correct proofs called wrong (/20)	0	0	0
Typos called real errors (/20)	1	0	1
Full match with ground truth (/100)	99	87	70

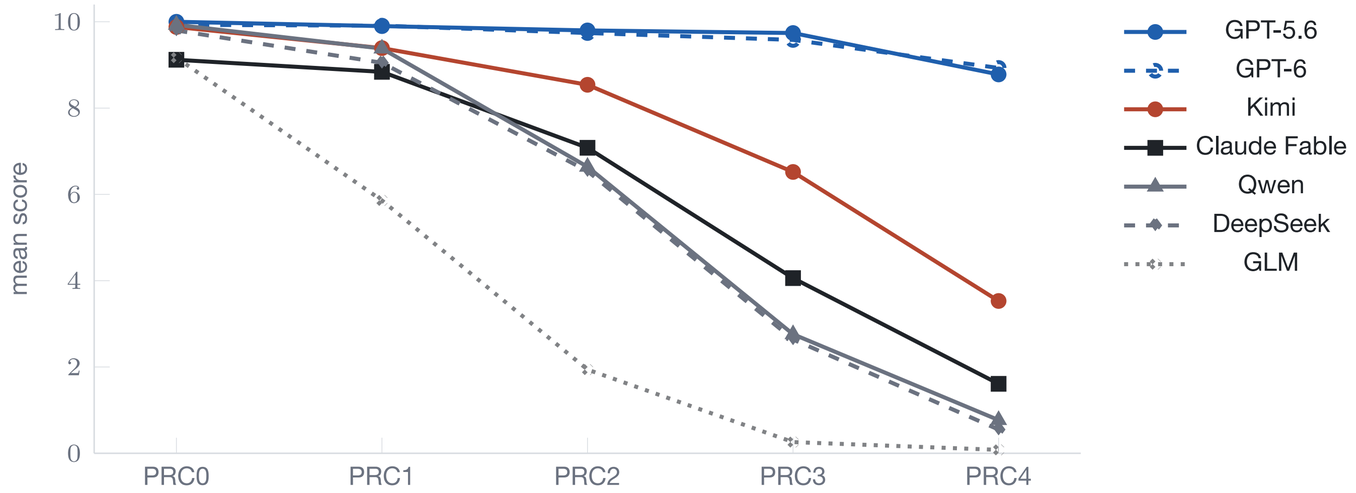
With Lean as ground truth, GPT-6 Astra agrees on 99 among 100 submissions. We believe that this support the usage of LLM-as-a-judge for proof check, from an engineering perspective.

Benchmarking LLM's Capability

Average performance over all tasks

Model	Mean score (/10)	Score ≥ 8	No final answer
GPT-5.6	9.68	94.8%	0.1%
GPT-6	9.67	93.9%	0.1%
Kimi K3	7.89	73.0%	10.2%
Claude Fable 5	6.62	58.8%	20.9%
Qwen 3.8 Max	6.46	62.4%	29.5%
DeepSeek V4 Pro	6.25	61.3%	31.9%
GLM 5.3	3.38	34.5%	62.5%

Performance for tasks with different difficulties



We perform the simple Q&A, with carefully written prompt approach, to get the solution, no agentic approach is leveraged.

We also set the limit of reasoning token length to be 48K for fair comparison.

Benchmarking LLM's Capability

We further consider the hardest set, i.e., RPC4, and allow the reasoning token to be 256K.

Model	Budget	Answers	Mean score (/10)	Score ≥ 8
GPT-6 (original run)	48k	30/30	**8.47**	21/30
GPT-5.6 (original run)	48k	30/30	**8.03**	19/30
Qwen	256k	30/30	4.80	3/30
Kimi	256k	30/30	4.73	4/30
DeepSeek	256k	30/30	3.57	1/30
Claude Fable	128k	20/30	3.10	5/30
GLM	256k*	1/30	0.10	0/30

GPT models consistently outperform other models, and some of other models can achieve higher performance with more reasoning budget.

What's Beyond?

- Good news:
  - Difficulties can be roughly set in advance via LLM, our PRC metric could be useful for identifying the difficulty of the problem.
  - PRC4 can serve as a strong metric to separate different LLM models.
  - Using natural language to verify the correctness of the proof can be done reliably (I am not saying this way is 100% confidently).
- Challenges:
  - Human-written data is not sustainable, some older ones have already been used in LLM training.
  - Human written data cleaning is costly.
  - It's kind of difficult to get harder problems.

OpenTheoryBench: ML Theory

We need a source of new, verified, sustainable theory/math problems that do not depend on human publication.

- This is different from many existing works

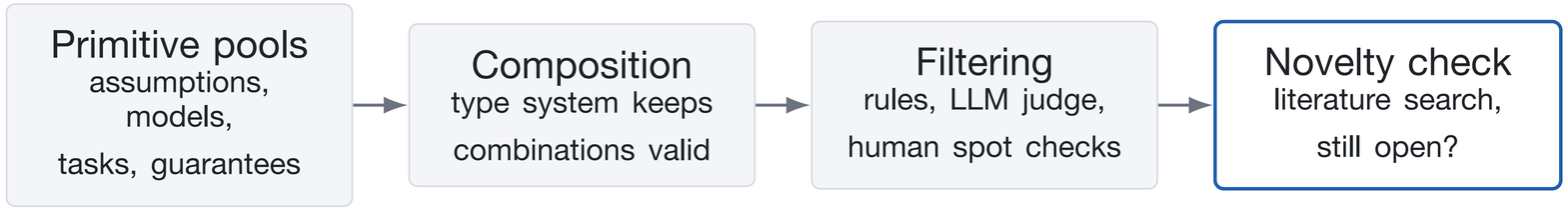
Work	Problem source	How ours differs
FrontierMath[21]	open problems	Curated by mathematicians	About 50 problems, all checkable
TCSBench[13], FormalTCS[14]	Theorems proved in papers	Answers known; reproduction
Bolzano[15]	Extracted from papers	Extraction only; corpus-bound
Moonshine[18]	One deep problem	Single-point, no difficulty scale
LiveMathematicianBench[19]	New papers after cutoff	Novelty prevents contamination
Stellar Colosseum[16], Cogentic[17]	Source is not a concern	Orchestrates solving only

Our approach is to develop methods to construct the problems automatically at scale.

https://github.com/OpenMLTheory/OpenMLTheoryData (under construction)

ML Theory Problems can be Synthesized

Under [assumptions], for [model] on [task],
what is the [guarantee] (upper or lower bound), and is it tight?

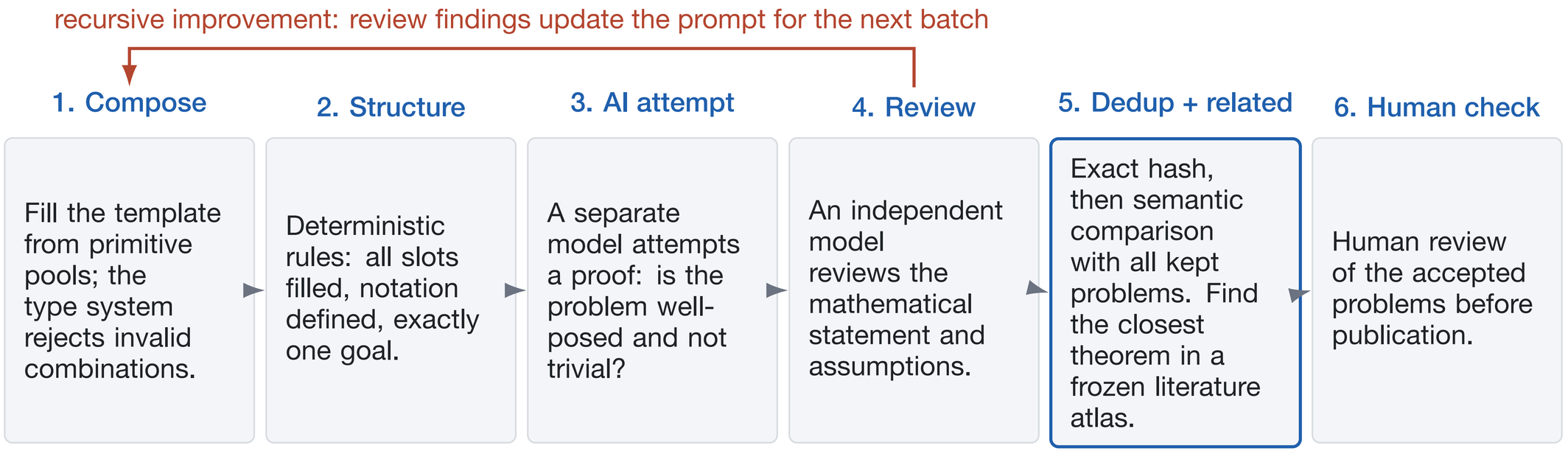


Slots and primitive pools

Slot	Example primitives (sampling)
Geometry	strongly log-concave, log-Sobolev, …
Algorithm	Langevin MC, MALA, HMC, …
Metric	TV, KL, Wasserstein-2, …
Theorem type	upper bound, lower bound, minimax

- Most ML theory problems follow a small number of templates, so they can be generated in batches.
- Published open problems are used for verification and deduplication.

The Synthesis Pipeline



Output: one record per problem, fixed format

Metadata	Problem setup	Assumption system	Key theorem	Most related work
slot values, hash, status	objects, algorithm, objective	numbered assumptions	the goal to prove	closest theorem, location, difference

Two phases: **phase 1** using recursive improvement to revise the workflow (i.e., prompts and skills); **phase 2** will directly perform simple modification of the generated problem.

What Can We Do with Synthesized Data

- Evaluation
  - We can use the problems to perform fair evaluation (no model has seen these problems before).
  - We can potentially set the difficulties of the problem in advance.
- Training
  - Large and structured problem set can help train better models
  - ML problems have certain advantages and we don’t need ground-truth answers to provide reward signals.
  - Given a large batch of problem and its generation trajectories, we may further explore how LLM learns to design problems, with few human labeling.

Our initial estimate is that we can generate at least million-level ML theory problems.

OpenMLTheory Bench: Current Progress

Direction	Published	Status
Sampling	3,610	bounded novelty check against a frozen literature atlas
PAC learning	1,000	first release; novelty and difficulty not yet assessed
RL and bandits	795	accepted from 1,404 audited candidates (218 on hold, 391 rejected)
Learning neural networks	1,000	records uploaded; not yet in the registry
Registry total	**5,405**	

We have started problem generation in 4 directions and generated about 5405 problems (subset of the problems have been reviewed and passed by human). Other directions are under construction for now.

Example data

Metadata	strongly log-concave · Lipschitz gradient · overdamped Langevin MC · constant step size · Gaussian initialization · total variation · last iterate · finite-time lower bound
Problem setup	Target $\pi_f \propto e^{-f}$ on $\mathbb{R}^d$. Start $\vartheta^{(0)} \sim \mathcal{N}(x_f^\star, L^{-1} I_d)$ and run $\vartheta^{(k+1)} = \vartheta^{(k)} - h\nabla f(\vartheta^{(k)}) + \sqrt{2h}\,\xi^{(k+1)}$, $\xi^{(k)} \sim \mathcal{N}(0, I_d)$, for $k < K$. Objective: $\mathrm{TV}(\mu_K, \pi_f)$, where μ_K is the law of $\vartheta^{(K)}$.
Assumption system	A1 (strong log-concavity): $f(y) \geq f(x) + \langle \nabla f(x), y - x\rangle + \frac{m}{2}\|y - x\|^2$. A2 (Lipschitz gradient): $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$.
Key theorem (goal)	Prove a non-asymptotic lower bound on $\mathrm{TV}(\mu_K, \pi_f)$ with explicit dependence on d, m, L, h, K, and give a nonempty region of (h, K) where it holds. Minimax quantifiers are not allowed.
Most related work	Dalalyan (2017), JRSS-B 79(3), Theorem 2 and Corollary 1, §3. Difference: the cited result is an upper bound; this problem asks for a lower bound.

Take Aways and Future Work

- Take Away
  - GPT is strongest in ML Theory problem (I believe no one doubt that), agentic development may not be necessary for most ML theory problem.
  - Building benchmark is necessary and promising (rather than solving individual problems).
- Future Works
  - Expanding the domain of OpenTheoryBench to other subjects, e.g., optimization, TCS, quantum theory, theoretical physics, etc.
  - Building a more systematic benchmark using the synthesized data.
  - Build the problem generate-solve loop for model training.
  - …

Thanks
Any questions?