Auto-formalizations with LLMs

Students: Siyuan (James) Ge, Henry Adams, Attila Jamilov, Elizabeth Wang
Mentors: Zihong Lin, Vasily Ilin

Overview

LLM-powered A*-search of formalised Lean proofs

- Towards an artificial mathematician
- Example
- Evaluation on the minif2f dataset
- Experiments with search heuristics
- Experiments with LLM parameters

Language models + Lean = 💙

Language models are creative but prone to hallucination.

Lean is tedious but does not allow mistakes.

Can "ground" LM's thinking by using feedback from Lean.

Goal: create an artificial mathematician.

Tree-based search

- Each step of the proof is a node
- Steps are suggested by an LLM (GPT-4o, Claude 3.5, Grok 3)
- A **heuristic** determines which nodes to pursue further and which to abandon

A look at the tactic suggestions

To prove: `theorem mathd_algebra_171 (f : ℝ → ℝ) (h₀ : ∀x, f x = 5 * x + 4) : f 1 = 9,`

we used the LeanDojo model. This model takes the current goal state as input, and provides a number of tactics out.

The benefit of this model is that it runs locally and is trained on Lean.

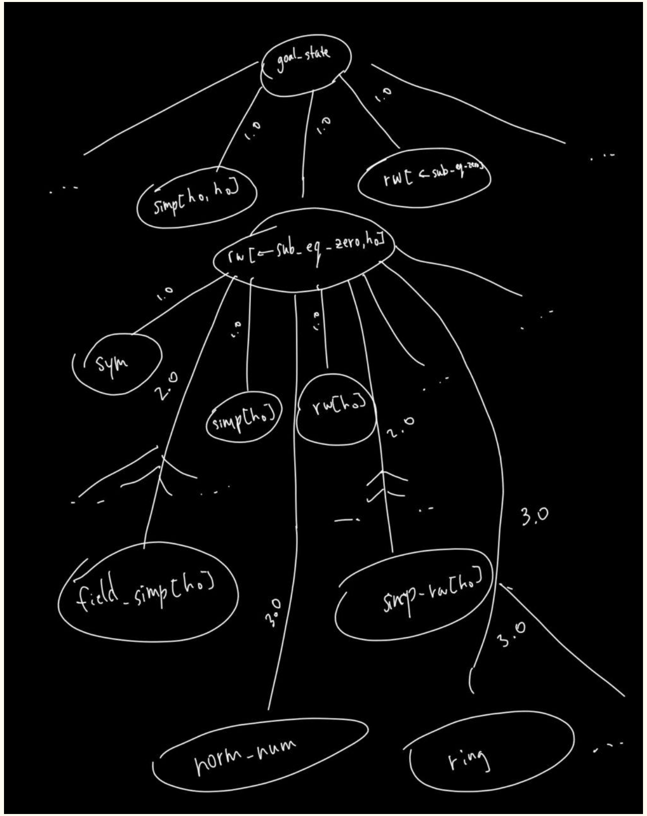
A look at the tactic suggestions

The weights represent how many turns it took to generate this tactic to make progress upon the goal state.

It took about 4 calls to the model to prove this example. The tactics the model chose are:

```
rw [← sub_eq_zero, h₀]
```

```
norm_num
```



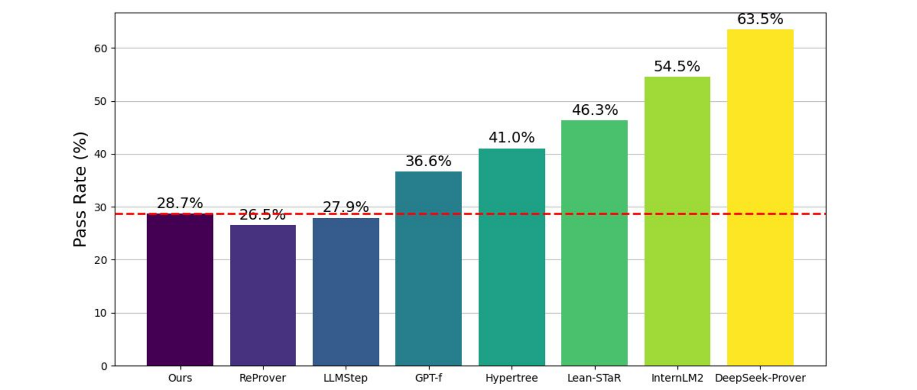
Evaluations on the miniF2F dataset

- miniF2F: formalized Olympic-level benchmark in algebra and number theory
- Our results: we did pass some problems but ...
  - Out of Steps = Reached the maximum search step
  - Out of States = No more search states to explore

	Count	Pass	Pass Rate	Out of Steps	Out of States	Others
IMO	20	0	0.00%	14	6	0
AIME	15	1	6.67%	12	2	0
AMC12	45	4	8.89%	34	4	3
MATH_algebra	70	32	45.71%	35	3	0
MATH_numbertheory	60	33	55.00%	21	5	1
Custom_algebra	18	0	0.00%	12	6	0
Custom_numbertheory	8	0	0.00%	7	1	0
Custom_induction	8	0	0.00%	5	2	1
Total	244	70	28.69%	140	29	5

Evaluations on the miniF2F dataset

- Current search is short-view and can only solve simple problems
  - With only 1 or 2 lines of proof
  - Relying on built-in goal-solving tactics: norm_num, linarith …



Search Step	Count
1	58
2	5
3	2
4	4
5	1

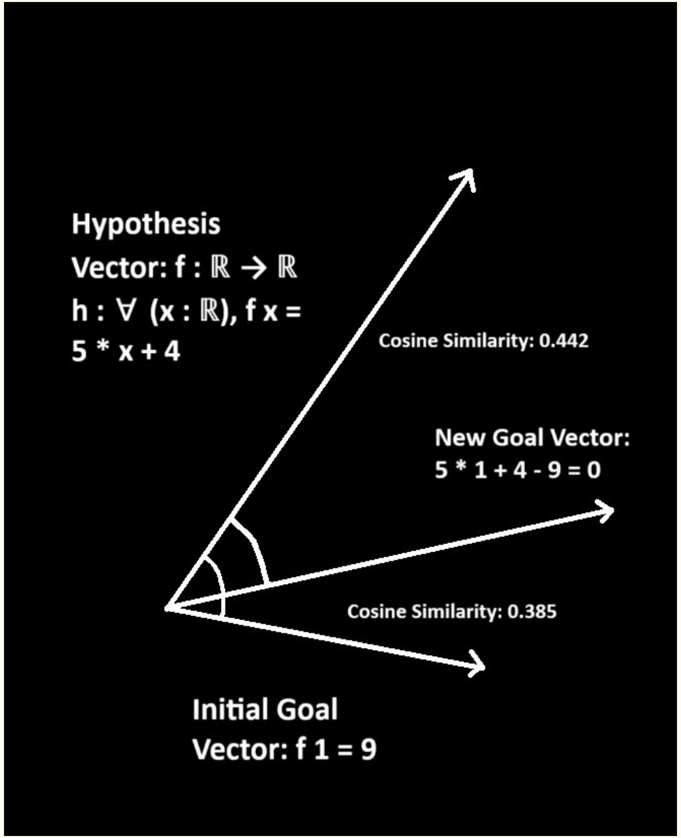
Proof Line	Count
1	58
2	12

Tactic	Count
norm_num	30
linarith	15
omega	11
simp	6
ring	5
nlinarith	3
field_simp	3
rfl	3
rw	3
symm	1
subst	1
have	1

Experimenting with heuristics in tree-search

When the search tree has a high branching factor, we need to be able to evaluate which proof states are "better" than others. We have a few options:

- Trivial Heuristic: If a proof state contains many goals, it should be harder to prove
- Log Probabilities: We can take the probabilities of different LLM suggestions as a measure of confidence
- Cosine Similarity: We embed hypotheses and goals in a high dimensional space, then compare them as vectors



Experiments with LLMs

```
theorem mathd_algebra_171
  (f : ℝ → ℝ)
  (h₀ : ∀x, f x = 5 * x + 4) :
  f 1 = 9 := by
```

Model	Temperature	Predicted Next Steps
deepseek-reasoner	0.5	rw [h_0]
deepseek-reasoner	0.7	rw [h_0]
deepseek-reasoner	1	rw [h_0]

Model	Temperature	Predicted Next Steps
gpt-4o-mini	1.2	rw [h_0 1] simp [h_0 1] rewrite h_0 1
gpt-4o-mini	1.4	rw [h_0 1] rw [h_0 1] rewrite h_0 1
gpt-4o-mini	1.5	rw h_0 1 rw [h_0 1] rewrite h_0 1

Model	Temperature	Predicted Next Steps
o3-mini	0.5	rw [h_0] rw [h_0 1] rw [h_0]
o3-mini	1.2	rw [h_0] rw [h_0] rw [h_0 1]
o3-mini	2	rw [h_0] rw [h_0 1] rw [h_0]

Model	Temperature	Predicted Next Steps
gemini-2.0-flash	0.8	rw [h_0] rw [h_0] rw [h_0]
gemini-2.0-flash	1.3	specialize h_0 1 rw [h_0] rw [h_0]
gemini-2.0-flash	1.8	exact h_0 1 rw [h_0 1] rw [h_0]

Experiments with LLMs

```
theorem mathd_algebra_171
  (f : ℝ → ℝ)
  (h₀ : ∀x, f x = 5 * x + 4) :
  f 1 = 9 := by
```

Previous tactics:
rw [h_0]

Model	Temperature	Predicted Next Steps
deepseek-reasoner	0.5	norm_num
deepseek-reasoner	1	norm_num
deepseek-reasoner	1.5	norm_num

Model	Temperature	Predicted Next Steps
gpt-4o-mini	0.7	norm_num norm_num linarith
gpt-4o-mini	1.3	linarith linarith norm_num
gpt-4o-mini	1.4	linarith simp [h_0 1] norm_num

Model	Temperature	Predicted Next Steps
o3-mini	0.5	norm_num norm_num norm_num
o3-mini	1.2	norm_num norm_num norm_num
o3-mini	2	norm_num norm_num norm_num

Model	Temperature	Predicted Next Steps
gemini-2.0-flash	1	simp simp norm_num
gemini-2.0-flash	1.4	norm_num simp simp
gemini-2.0-flash	1.8	norm_num simp simp

Future Plans

- More experiments:
  - Better LLM prompting
  - RAG
  - Diffusion-based LLMs
  - More sophisticated heuristics
  - More compute: 2000 steps instead of 20
- Sketching, drafting
- RL fine tuning