Sareh Nabi

Stanford University · CodeX Center for Legal Informatics
Joint work with Roland Vogl and Marzieh Nabi
Paper: arXiv:2609.34460

sarehnabi.com
X: @SarehNabi
linkedin: sarehnabi

Project page

When Does Structured Knowledge Help Neural Theorem Proving?

Keywords: neural theorem proving, Lean 4, Mathlib, knowledge graphs, retrieval augmentation, LLM agents, controlled ablation, complementarity

Birds-eye view

- **Setting.** LLM provers write machine-checked Lean 4 proofs from what they learned in training; Mathlib's 285K+ theorems carry only syntactic dependencies.
- **Question.** The semantic layer mathematicians use (analogies, generalizations, cross-domain bridges) is implicit. Does supplying it at inference time help, for which models?
- **MathKG.** 364 Mathlib-verified theorems and definitions linked by 9,434 typed edges in 8 relation types, inferred by an LLM over all node pairs.
- **MathAgent.** An Explorer gathers context from MathKG and from Mathlib retrieval; a Prover writes Lean 4 proofs with error-guided refinement (pass@32).
- **Ablation.** 4 augmentation modes × 5 models: general vs. Lean-specialized provers at 8B and 32B, plus Sonnet 4.6, on miniF2F (and two harder benchmarks for Sonnet).
- **Specialization dominates augmentation.** Lean fine-tuning adds 33–38 points of solve rate in every mode; no augmentation mode adds more than 3.
- **Augmentation is capability-conditioned.** KG context helps 8B models and hurts 32B models and Sonnet.
- **The modes are complementary.** Picking the best mode per problem solves 6–58% more than the bare prover (32% more on PutnamBench), motivating adaptive augmentation.

More about Sareh Nabi

- AI researcher at Stanford University; previously Research Scientist at Amazon and Applied Research Scientist at Microsoft.
- PhD in Operations Research, University of Washington; MSc in Mathematics and in Economics, Simon Fraser University; BSc in Mathematics, Sharif University of Technology.
- Research interests: AI for science, reinforcement learning, LLM reasoning and agents.

Stanford University
Human-Centered
Artificial Intelligence

LAUDE INSTITUTE

Compute provided by the Laude Institute. Code and data: github.com/sarehnabi/mathagent

MathKG and the controlled ablation

The missing layer in Mathlib

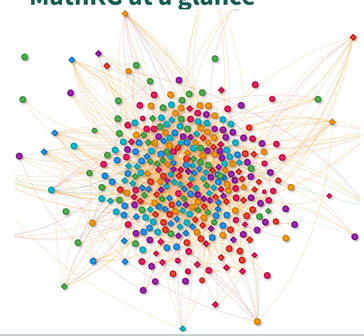
What Mathlib has

✓ 285K+ verified theorems
✓ Syntactic dependencies
✓ Type-checked correctness

What it lacks

✗ Cross-domain analogies
✗ Generalization relationships
✗ Strategic proof guidance

MathKG at a glance



364 nodes: 229 theorems, 135 definitions, 7 domains (e.g., number theory, analysis, geometry)

9,434 typed edges in 8 relation types: cross-domain-bridge, prerequisite, applied-in, analogous-to, instance-of, generalizes, specializes, equivalent-to

Built by LLM relation extraction with Sonnet 4.6 over all 132,132 node pairs, anchored to verified Mathlib declarations

Augmentation modes

prover_only: no external context (baseline)

with_kg: MathKG semantic search

with_mathlib: lexical Mathlib retrieval

full_system: KG + Mathlib

Oracle: best mode per problem = union of the four modes

Models

Qwen3-8B / 32B: general-purpose

Goedel-Prover-V2-8B / 32B: Lean-specialized, fine-tuned from the Qwen3 bases

Sonnet 4.6: stronger reference

A matched 2×2 (general vs. specialized, 8B vs. 32B) plus a reference model.

Benchmarks

miniF2F: 488 problems, all five models

MathOlympiadBench: 360 problems, Sonnet

PutnamBench: 672 problems, Sonnet

Protocol: pass@32 with sequential, error-guided refinement

MathAgent

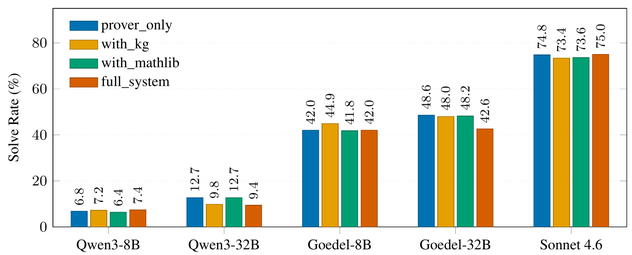
Explorer: builds the Prover's context in two layers, MathKG semantic search and lexical Mathlib retrieval

Prover: writes a Lean 4 proof, then refines it on compiler errors, up to 32 attempts

Every LLM call in a run uses the evaluated model.

Results: three findings from the controlled ablation

Finding 1: Specialization dominates augmentation



+33–38 points
from Lean fine-tuning, in every mode

8B beats 32B
Goedel-8B leads Qwen3-32B by 29–35 points

≤ 3 points
the most any augmentation mode adds

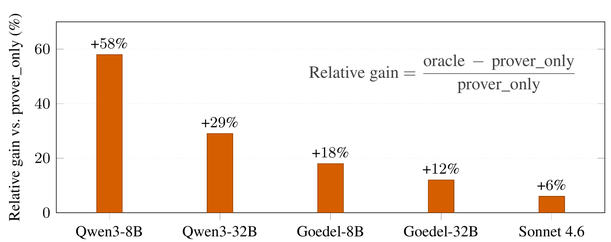
Finding 2: Augmentation is capability-conditioned

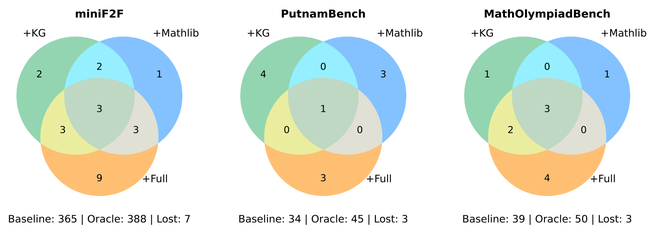
Δ solved, miniF2F	Qwen3 8B	Qwen3 32B	Goedel 8B	Goedel 32B	Sonnet 4.6
KG helps? (kg − PO)	+2	−14	**+14**	−3	−7
Mathlib helps? (ml − PO)	−2	0	−1	−2	−6
Synergy? (full − best)	+1	−16	−14	−27	+7
Best fixed mode	full	PO / ml	KG	PO	full

Scale sets the sign. KG context helps at 8B (+2, +14) but hurts at 32B (−14, −3) and Sonnet (−7).

Specialization sets the magnitude. The Lean-tuned model gains more than its general base at both scales.

Finding 3: The augmentation modes are complementary





Baseline: 365 | Oracle: 388 | Lost: 7

Baseline: 34 | Oracle: 45 | Lost: 3

Baseline: 39 | Oracle: 50 | Lost: 3

Per-problem selection beats the unaugmented prover for every model, even the strongest, where no fixed mode improves the aggregate. The gain shrinks as capability grows.

For Sonnet, each augmented mode recovers different problems beyond the prover_only baseline, and few are lost (solved only by prover_only): the oracle gain grows from +6% on miniF2F to +28% on MathOlympiadBench and +32% on PutnamBench.