Can large language models help people prove mathematical theorems?

We present **NaturalProver**, a language model that generates mathematical proofs by conditioning on background references (e.g. theorems and definitions that are either retrieved or human-provided), and optionally enforces their presence with constrained decoding.

**Grounding + references + constrained decoding**

NaturalProver is an instance of GPT-3 fine-tuned on NaturalProofs [Welleck et al., Neurips 2021]. NaturalProver adds two components on top of GPT-3:

- **In-context references**: retrieved or provided theorems/definitions relevant to a correct proof.
- **Constrained decoding**: samples multiple next-steps, retains steps in a beam based on constraints.

**Natural vs. formal theorem proving**

- Rigid
- Not much data
- Easy to verify

- Flexible
- Used in education, science, engineering
- Lots of language data
- Hard to verify!

Figure 1. Classical provers use rigid formal languages. Can LLMs prove in flexible natural language?

**Figure 2.** On theorems from the NaturalProofs benchmark, NaturalProver improves the quality of next-step suggestions and generated proofs over fine-tuned GPT-3, according to human evaluations from university-level mathematics students. NaturalProver is capable of proving some theorems that require short (2-6 step) proofs, and providing next-step suggestions that are rated as correct and useful over 40% of the time.

**Human-machine collaboration**

Figure 3. NaturalProver had > 40% correct and useful next-step predictions. These compound in full-proof generation. An exciting option is human-machine collaboration with multiple suggestions.

Towards verified natural proofs: come see us at the MathAI workshop!