

Learning to Prove Theorems via Interacting with Proof Assistants



Kaiyu Yang, Jia Deng

Department of Computer Science, Princeton University



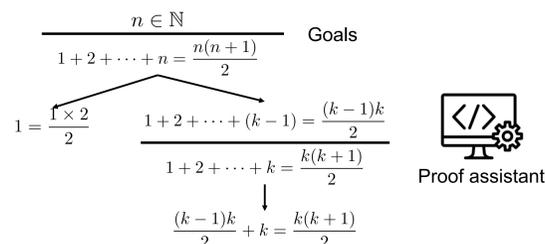
1. Introduction

Can machine learning agents learn high-level, human-like mathematical reasoning?

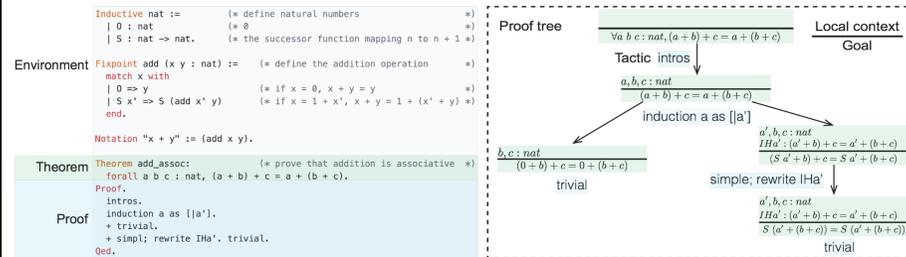
- Problem**
 - Automated theorem proving (ATP) manipulates theorems using a low-level representation, making it difficult to benefit from the high-level abstraction common to humans.
 - Interactive theorem proving (ITP) is close to human mathematical reasoning. But it is labor-intensive, requiring humans to interact with a software system (proof assistant).
- Proposed solution**
 - ITP's high-level formalism + ATP's proof automation
 - Develop machine learning agents to imitate humans for interacting with a proof assistant.
- Contributions**
 - CoqGym: a large-scale dataset and learning environment for theorem proving via interacting with a proof assistant.
 - ASTactic: a deep learning model for this task that can prove theorems not provable by existing methods.

2. Interactive Theorem Proving

- Humans interact with proof assistants**
 - The user sees the **goal** and enters a **tactic** representing a high-level manipulation, such as induction.
 - The proof assistant executes the tactic, decomposing the goal into multiple sub-goals.
 - The process starts with the original theorem as the initial goal, and ends when there is no goal left.
- We train machine learning agents to replace humans in this task.



3. CoqGym: Data and Environment

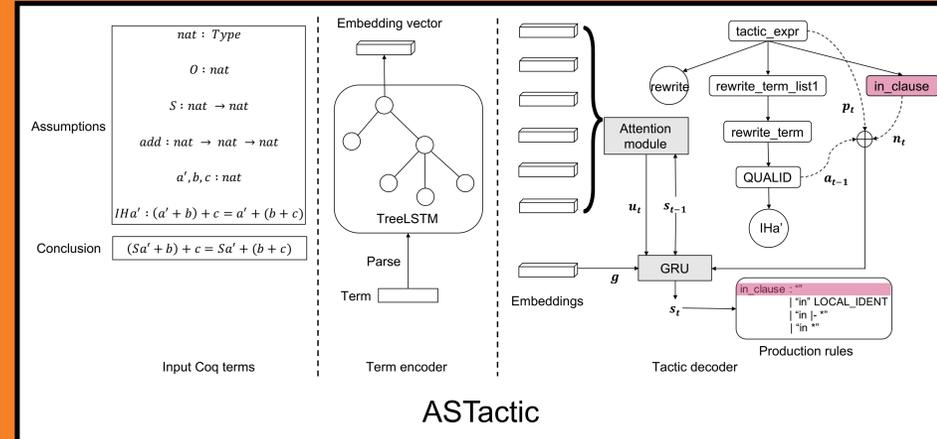


- A tool for interacting with the Coq proof assistant [1]
- Large**
 - 71K human-written proofs
 - Over an order of magnitude larger than existing datasets
 - Sufficient for data-hungry machine learning models
- Diverse:**
 - From 123 Coq projects
 - Cover a broad spectrum of domains (math, software, etc.)
- Structured data**
 - Proof represented by proof trees
 - Abstract syntax trees of the expressions in the proof

4. ASTactic: Deep Tactic Generation

Given the current proof goal, an encoder-decoder architecture generates a tactic in the form of abstract syntax tree (AST).

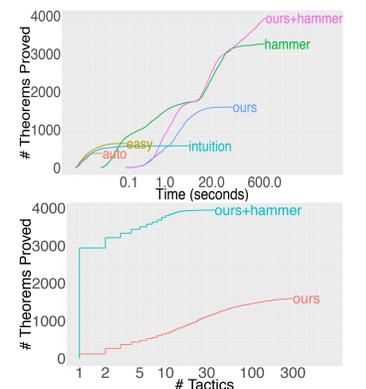
- Term encoder**
 - Input terms (multiple assumptions and one conclusion) are parsed into ASTs and embedded by a TreeLSTM network [2].
- Tactic decoder**
 - To generate a tactic AST conditioned on the embeddings, the decoder sequentially grows a partial tree by selecting production rules and terminal tokens.
- Search for complete proofs**
 - At each step, ASTactic outputs 20 tactics via beam search.
 - Treating them as possible actions, we search for a proof via depth-first search.



5. Experiments

- Task:** Proving the 13,137 testing theorems in CoqGym. Each proof within 300 tactics AND a wall time of 10 minutes.
- Baselines**
 - Coq's built-in automated tactics: *trivial*, *auto*, *intuition*, *easy*
 - State-of-the-art ATP systems: *hammer* [3]
- Results**
 - Our system proves 12.2% of the theorems, outperforming Coq's built-in tactics (4.9%).
 - Proves 30.0% of the theorems when combined with *hammer*, a large improvement of 5.2% over using *hammer* alone.

Method	Success rate (%)
trivial	2.4
auto	2.9
intuition	4.4
easy	4.9
hammer	24.8
ours	12.2
ours + auto	12.8
ours + hammer	30.0



- Barras, Bruno, et al. The Coq proof assistant reference manual: Version 6.1. Diss. Inria, 1997.
- Tai, Kai Sheng, et al. "Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks." ACL. 2015.
- Czajka, Łukasz, and Cezary Kaliszyk. "Hammer for Coq: Automation for dependent type theory." Journal of automated reasoning. 2018.
- Gauthier, Thibault, Cezary Kaliszyk, and Josef Urban. "TacticToe: Learning to Reason with HOL4 Tactics." LPAR. 2017.
- Huang, Daniel, et al. "Gamepad: A learning environment for theorem proving." ICLR. 2019.
- Bansal, Kshitij, et al. "HOList: An Environment for Machine Learning of Higher Order Logic Theorem Proving." ICML. 2019.