LeanDojo

Yifeng He

Sep 19, 2023

Section 1

Background

Proof Assistant Languages

- Originated from logical programming, which was the first attempt to develop **Artificial Intelligence**

Terminology

- state: current state of the theorem
- tactic: a function that takes a state and returns a new state
- primise: proved theorems/pre-defined functions to be used in tactics

Curry-Howard Correspondence

Logic	Programming
Proposition	Type
Theorem	Function Definition
Proof	Implementation of Function

$\implies$ LLM theorem proving $\approx$ code generation in a Dependent Type Language.

Section 2

Approach

LeanDojo Benchmark/Dataset

1. *File dependencies*: directed acyclic graph, nodes are files and edges are relations between files
2. *ASTs* of all files
3. *States and tactics*: process all proofs to (PrevState, Tactic, NextState) triples
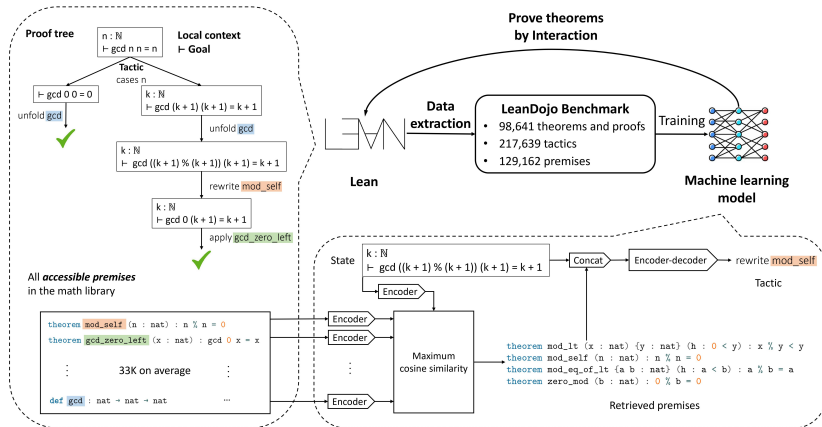4. *Premises*: recored where is defined and used on ASTs.

ReProver

Figure 1: ReProver

Section 3

Evaluation

Premise Selection

Given current state, predict the tactic to use.

Table 1: Premise selection testing performance. `random` and `novel_premises` are different data splits (Sec. 4). R@k is the recall for the top k retrieved premises, and MRR is the mean reciprocal rank metric (higher is better). Our retriever outperforms BM25 and ablations.

Method	random			novel_premises		
	R@1	R@10	MRR	R@1	R@10	MRR
BM25	5.6	16.9	0.14	7.8	21.0	0.19
Ours	**12.9**	**38.4**	**0.31**	**9.3**	**28.5**	**0.24**
w/ all premises	11.6	36.0	0.26	7.4	23.3	0.21
w/o in-file negatives	10.5	33.2	0.26	8.0	25.5	0.22

Figure 2: Premise selection

Theorem Proving

Table 2: Theorem proving Pass@1 (%) on the testing data of LeanDojo Benchmark. Our ReProver model outperforms tidy, GPT-4, and a baseline that generates tactics directly without retrieval.

Method	random	novel_premises
tidy	23.8	5.4
GPT-4	28.8	7.5
ReProver (ours)	**51.4**	**26.2**
w/o retrieval	47.5	22.9

Figure 3: Theorem proving