Learning to Prove Theorems via Interacting with Proof Assistants

Kaiyu Yang, Jia Deng
Automated Theorem Proving (ATP)

\[ n \in \mathbb{N} \Rightarrow 1 + 2 + \cdots + n = \frac{(n + 1)n}{2} \]

Assumptions  Conclusion
Automated Theorem Proving (ATP)

\[ n \in \mathbb{N} \implies 1 + 2 + \ldots + n = \frac{(n + 1)n}{2} \]
Automated Theorem Proving (ATP)

\[ n \in \mathbb{N} \implies 1 + 2 + \cdots + n = \frac{(n + 1)n}{2} \]
Automated Theorem Proving (ATP) is Useful for

Computer-aided proofs in math
Automated Theorem Proving (ATP) is Useful for

Computer-aided proofs in math

Software verification
Automated Theorem Proving (ATP) is Useful for

Computer-aided proofs in math

Software verification

Hardware design
Automated Theorem Proving (ATP) is Useful for

- Computer-aided proofs in math
- Software verification
- Hardware design
- Cyber-physical systems
Drawbacks of State-of-the-art ATP

- Prove by resolution

\[ 1 + 2 + \cdots + n = \frac{(n + 1)n}{2} \]

Theorem

Conjunctive normal forms (CNFs)
Drawbacks of State-of-the-art ATP

- Prove by resolution

\[ 1 + 2 + \cdots + n = \frac{(n + 1)n}{2} \]

\[ p \lor \neg q \lor \neg r \lor s \]
\[ \neg x \lor y \lor z \lor q \]
\[ \neg x \lor y \lor z \lor p \lor \neg r \lor s \]

Theorem \hspace{1cm} Conjunctive normal forms (CNFs)
Drawbacks of State-of-the-art ATP

- Prove by resolution

\[ 1 + 2 + \cdots + n = \frac{(n + 1)n}{2} \]

\[ p \lor \neg q \lor \neg r \lor s \]
\[ \neg x \lor y \lor z \lor q \]
\[ \neg x \lor y \lor z \lor p \lor \neg r \lor s \]
\[ \cdots \]

Theorem

Conjunctive normal forms (CNFs)
Drawbacks of State-of-the-art ATP

- The CNF representation
  - Long and incomprehensible even for simple math equations
  - Unsuitable for human-like high-level reasoning

\[
1 + 2 + \cdots + n = \frac{(n + 1)n}{2}
\]

\[
p \lor \neg q \lor \neg r \lor s \\
\neg x \lor y \lor z \lor q \\
\neg x \lor y \lor z \lor p \lor \neg r \lor s \\
\vdots
\]

Theorem  Conjunctive normal forms (CNFs)
Interactive Theorem Proving

Human

Proof assistant
Interactive Theorem Proving

$1 + 2 + \cdots + n = \frac{n(n+1)}{2}$

$n \in \mathbb{N}$

Human

Proof assistant
Interactive Theorem Proving

**Goal**

**Assumptions**

**Conclusion**

\[1 + 2 + \cdots + n = \frac{n(n + 1)}{2}\]

*Human*

```
induction n.
```

*Proof assistant*
Interactive Theorem Proving

\[ n \in \mathbb{N} \]

\[ 1 + 2 + \cdots + n = \frac{n(n + 1)}{2} \]

\[ 1 = \frac{1 \times 2}{2} \]

\[ 1 + 2 + \cdots + (k - 1) = \frac{(k - 1)k}{2} \]

\[ 1 + 2 + \cdots + k = \frac{k(k + 1)}{2} \]

Human

induction n.

Proof assistant
Interactive Theorem Proving

\[ n \in \mathbb{N} \]

\[ 1 + 2 + \cdots + n = \frac{n(n + 1)}{2} \]

\[ 1 = \frac{1 \times 2}{2} \]

\[ 1 + 2 + \cdots + (k - 1) = \frac{(k - 1)k}{2} \]

\[ 1 + 2 + \cdots + k = \frac{k(k + 1)}{2} \]

Human

\[ \text{induction } n. \]
\[ + \text{ reflexivity} \]

Proof assistant
Interactive Theorem Proving

\[ n \in \mathbb{N} \]
\[ 1 + 2 + \cdots + n = \frac{n(n + 1)}{2} \]

\[ 1 = \frac{1 \times 2}{2} \]
\[ 1 + 2 + \cdots + (k - 1) = \frac{(k - 1)k}{2} \]
\[ 1 + 2 + \cdots + k = \frac{k(k + 1)}{2} \]
\[ \frac{(k - 1)k}{2} + k = \frac{k(k + 1)}{2} \]

induction n.
+ reflexivity
+ subst; reflexivity.
Interactive Theorem Proving

\[ n \in \mathbb{N} \]

\[ 1 + 2 + \cdots + n = \frac{n(n + 1)}{2} \]

Human

- induction \( n \)
- + reflexivity
+ subst; reflexivity.

Proof assistant

\[ 1 = \frac{1 \times 2}{2} \]

\[ 1 + 2 + \cdots + (k - 1) = \frac{(k - 1)k}{2} \]

\[ 1 + 2 + \cdots + k = \frac{k(k + 1)}{2} \]

\[ \frac{(k - 1)k}{2} + k = \frac{k(k + 1)}{2} \]

Labor-intensive, requires extensive training
Interactive Theorem Proving

\[ n \in \mathbb{N} \]
\[ 1 + 2 + \cdots + n = \frac{n(n+1)}{2} \]

\[
1 = \frac{1 \times 2}{2} \\
1 + 2 + \cdots + (k-1) = \frac{(k-1)k}{2} \\
1 + 2 + \cdots + k = \frac{k(k+1)}{2} \\
\frac{(k-1)k}{2} + k = \frac{k(k+1)}{2}
\]

- induction \( n \)
- + reflexivity
- + subst; reflexivity.
CoqGym: Dataset and Learning Environment

• Tool for interacting with the Coq proof assistant [Barras et al. 1997]
• 71K human-written proofs, 123 Coq projects
• Diverse domains
  • math, software, hardware, etc.
CoqGym: Dataset and Learning Environment

- Tool for interacting with the Coq proof assistant [Barras et al. 1997]
- 71K human-written proofs, 123 Coq projects
- Diverse domains
  - math, software, hardware, etc.
- Structured data
  - Proof trees
  - Abstract syntax trees

\[
1 + 2 + \cdots + n = \frac{n(n+1)}{2}
\]

Proof tree
ASTactic: Tactic Generation with Deep Learning

\[ n, k \in \mathbb{N} \]
\[ n = 2k \]
\[ n \geq k \]

Proof goal

Tactic

induction n.
ASTactic: Tactic Generation with Deep Learning

Abstract syntax trees (ASTs)

$n, k \in \mathbb{N}$

$n = 2k$

$n \geq k$

Proof goal
Proof goal

Abstract syntax trees (ASTs)

$n, k \in \mathbb{N}$

$n = 2k$

$n \geq k$

ASTactic: Tactic Generation with Deep Learning

TreeLSTM encoder

[Tai et al. 2015]

Feature vectors
ASTactic: Tactic Generation with Deep Learning

ASTactic can augment state-of-the-art ATP systems [Czajka and Kaliszyk, 2018] to prove more theorems.
Related Work

• CoqHammer [Czajka and Kaliszyk, 2018]
• SEPIA [Gransden et al. 2015]
• TacticToe [Gauthier et al. 2018]
• FastSMT [Balunovic et al. 2018]
• GamePad [Huang et al. 2019]
• HOList [Bansal et al. 2019] (concurrent work at ICML19)

Main differences:
• Our dataset is larger covers more diverse domains.
• Our model is more flexible, generating tactics in the form of ASTs.
Learning to Prove Theorems via Interacting with Proof Assistants

Kaiyu Yang, Jia Deng

Poster today @ Pacific Ballroom #247
Code: https://github.com/princeton-vl/CoqGym