

HOList: An Environment for Machine Learning of Higher-Order Theorem Proving

Kshitij Bansal, Christian Szegedy

Can we create a human level AI to reason about mathematics?

Can we create a human level AI to reason about mathematics?

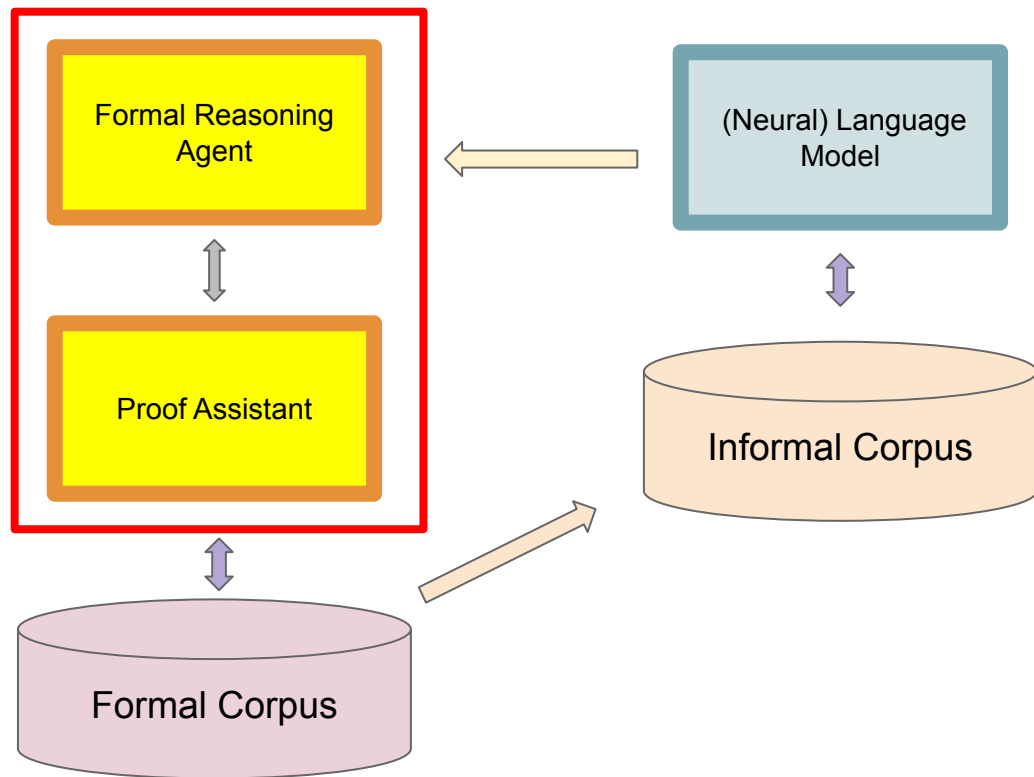
Without relying on informal human mathematics

- No need for autoformalization (requires high level of natural language understanding)
- Need to formalize the notion of “interestingness”.
- User needs to learn an “alien” language just to communicate a theorem to it
- Can’t communicate its discoveries
- May be hard to bootstrap (little training data)

Relying on informal human mathematics

- Needs auto-formalization
- Requires no formalization on user side
- Could learn the human notion of “interestingness”.
- Lot of training data to bootstrap from

Vision of joint proving and auto-formalization



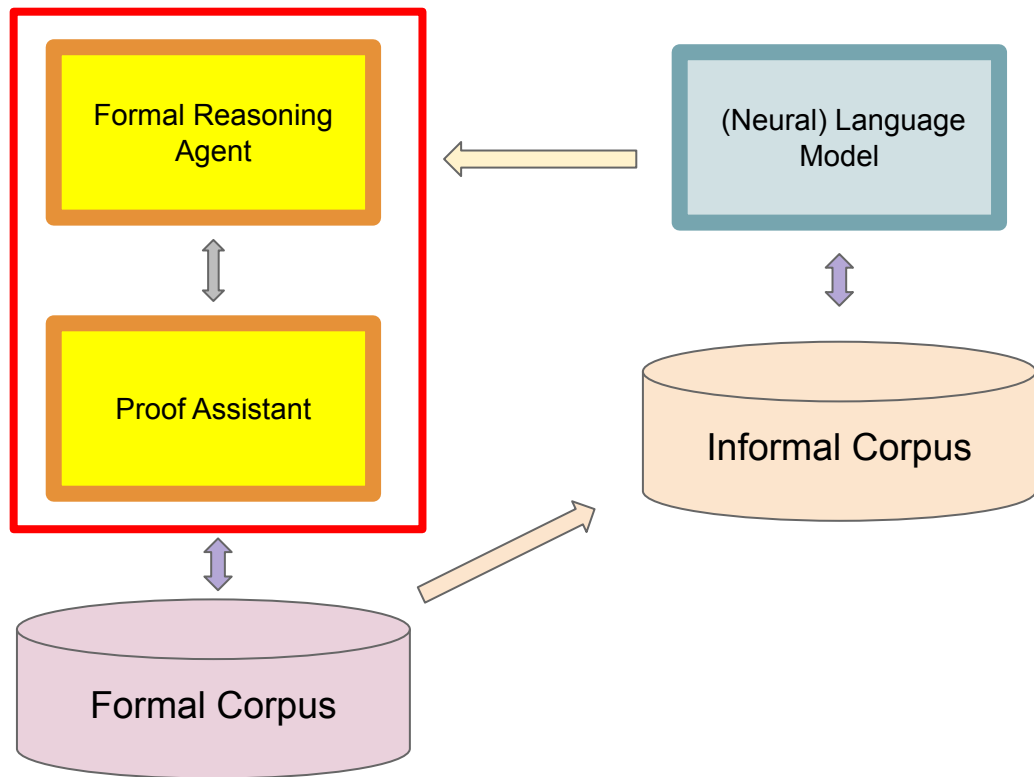
Which Proof Assistant?

- Coq
- Lean
- Isabelle
- HOL4
- HOL Light
- Mizar

Which Proof Assistant?

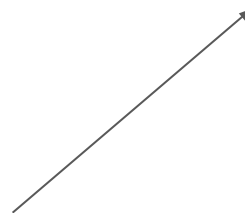
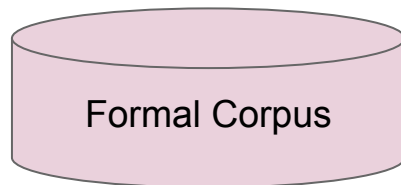
- Coq
- Lean
- Isabelle
- HOL4
- **HOL Light**
- Mizar

Vision of joint proving and auto-formalization



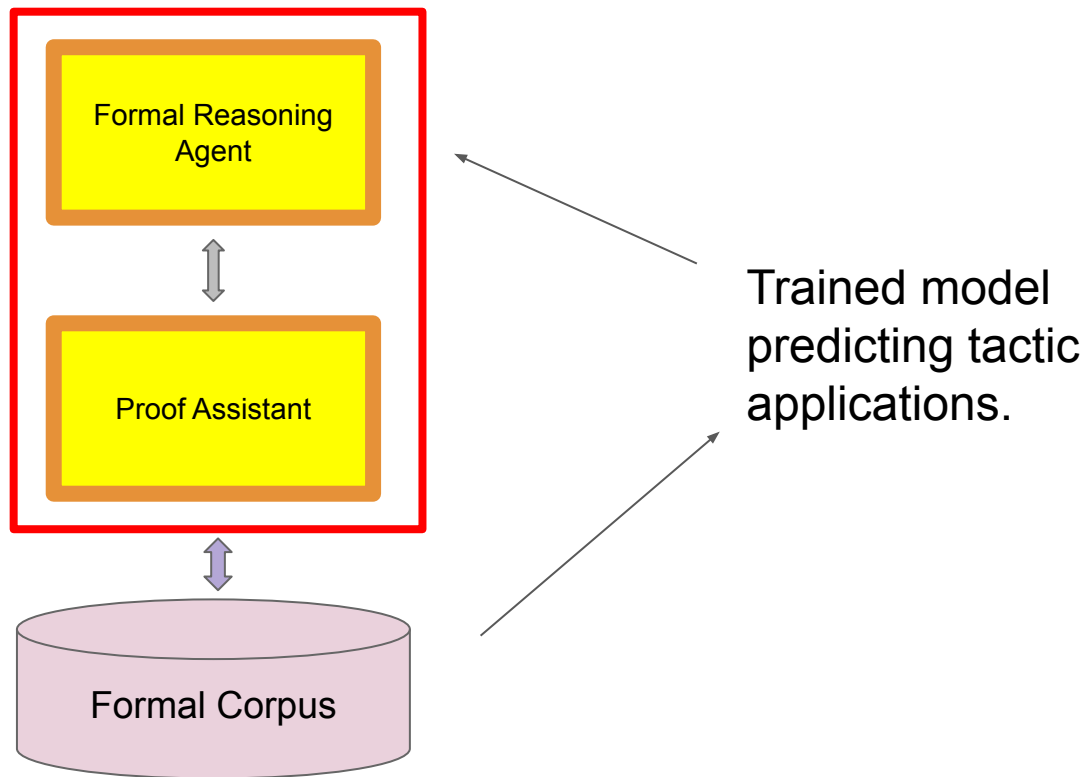
AITP'18

- Theorems
- Proofs: tree of (goal, tactic) to (subgoals)



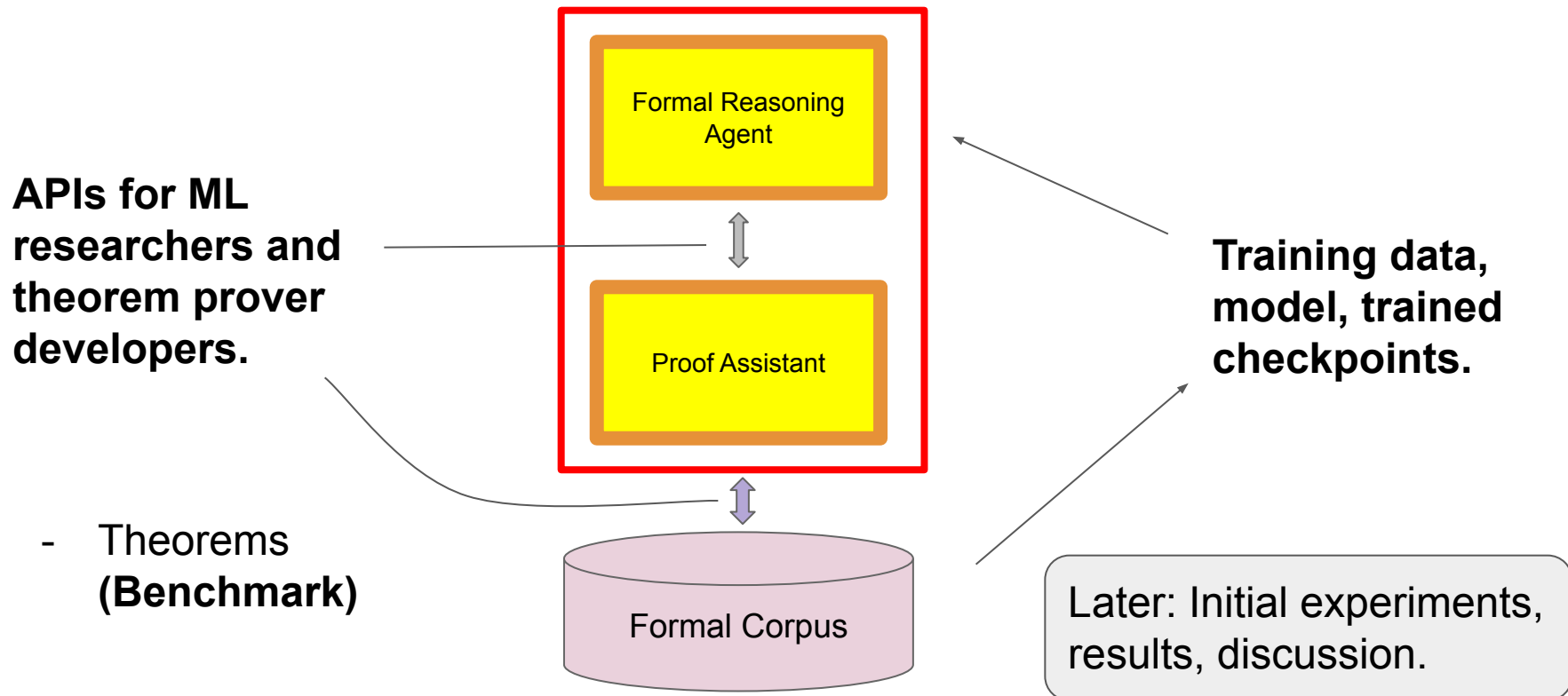
Trained model
predicting tactic
applications.

- Theorems
- Proofs: tree of (goal, tactic) to (subgoals)

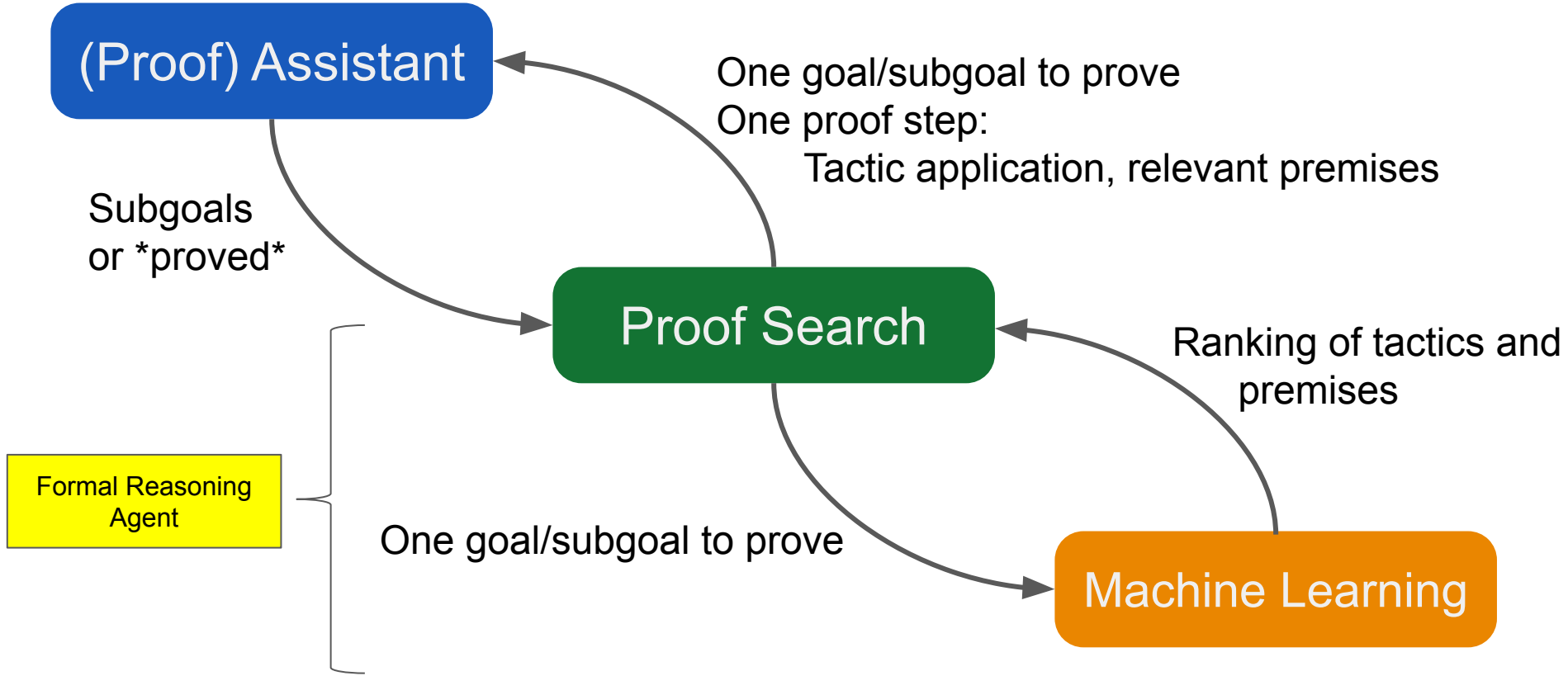


HOList

An Environment for Machine Learning of Higher-Order Theorem Proving



APIs for Theorem Prover Developers and ML Researchers



Proof Assistant Service

RegisterTheorem

Register a new theorem for use as premise in later proofs.

- Request:
 - Theorem
- Response: one of
 - TheoremFingerprint
 - Error

ApplyTactic

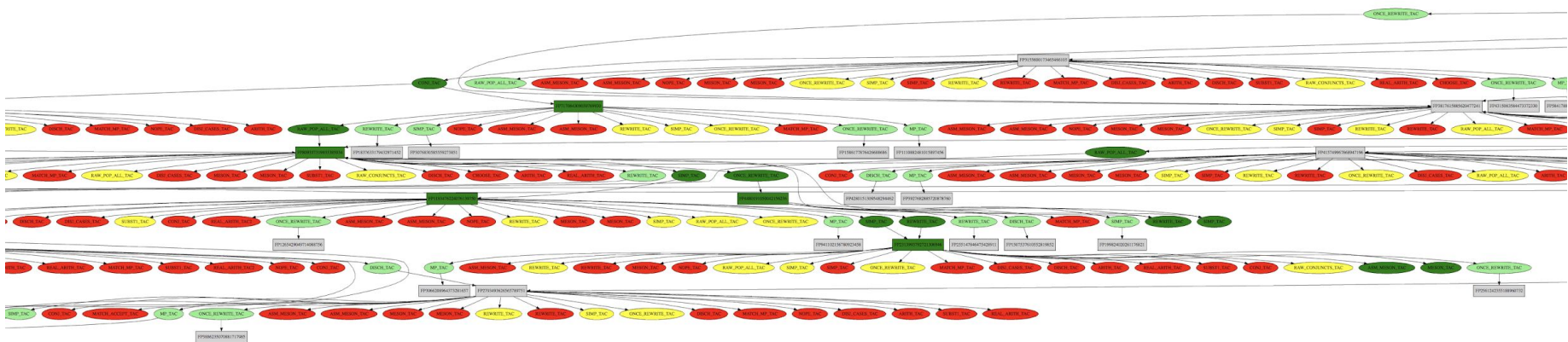
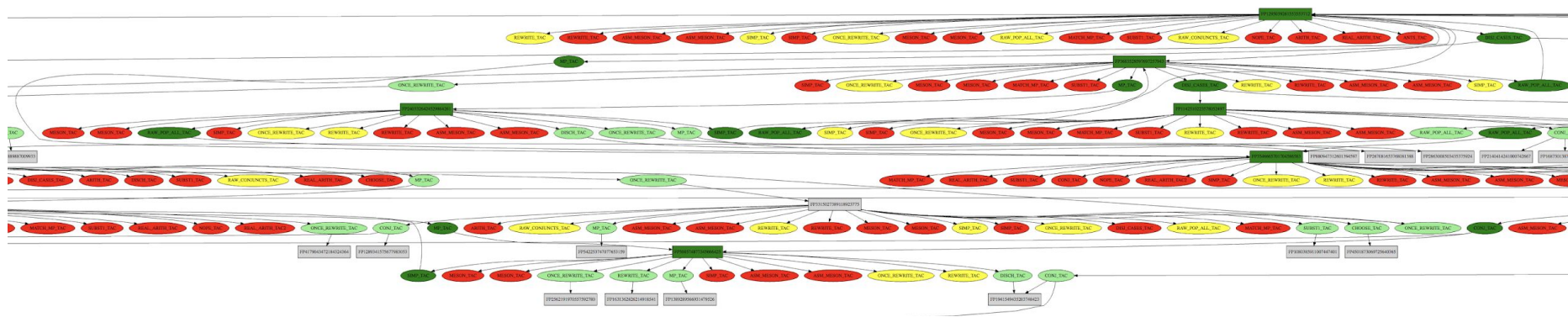
Apply a tactic to a goal, potentially generating new subgoals.

- Request:
 - Goal
 - Tactic
- Response: one of
 - Subgoals
 - Error

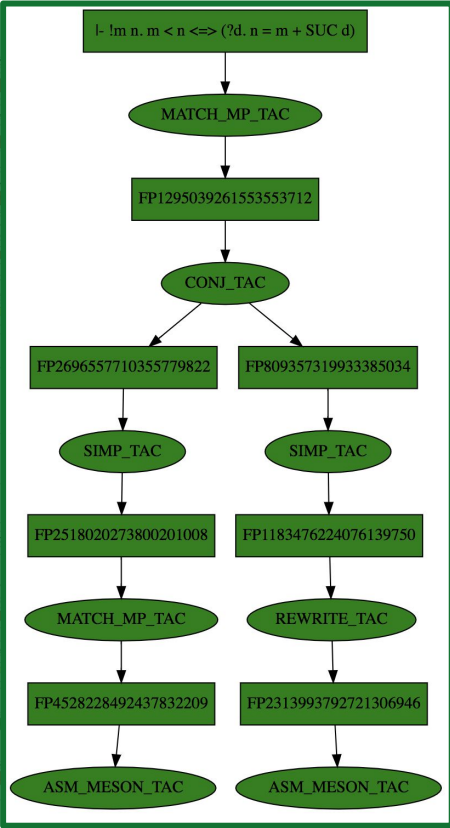
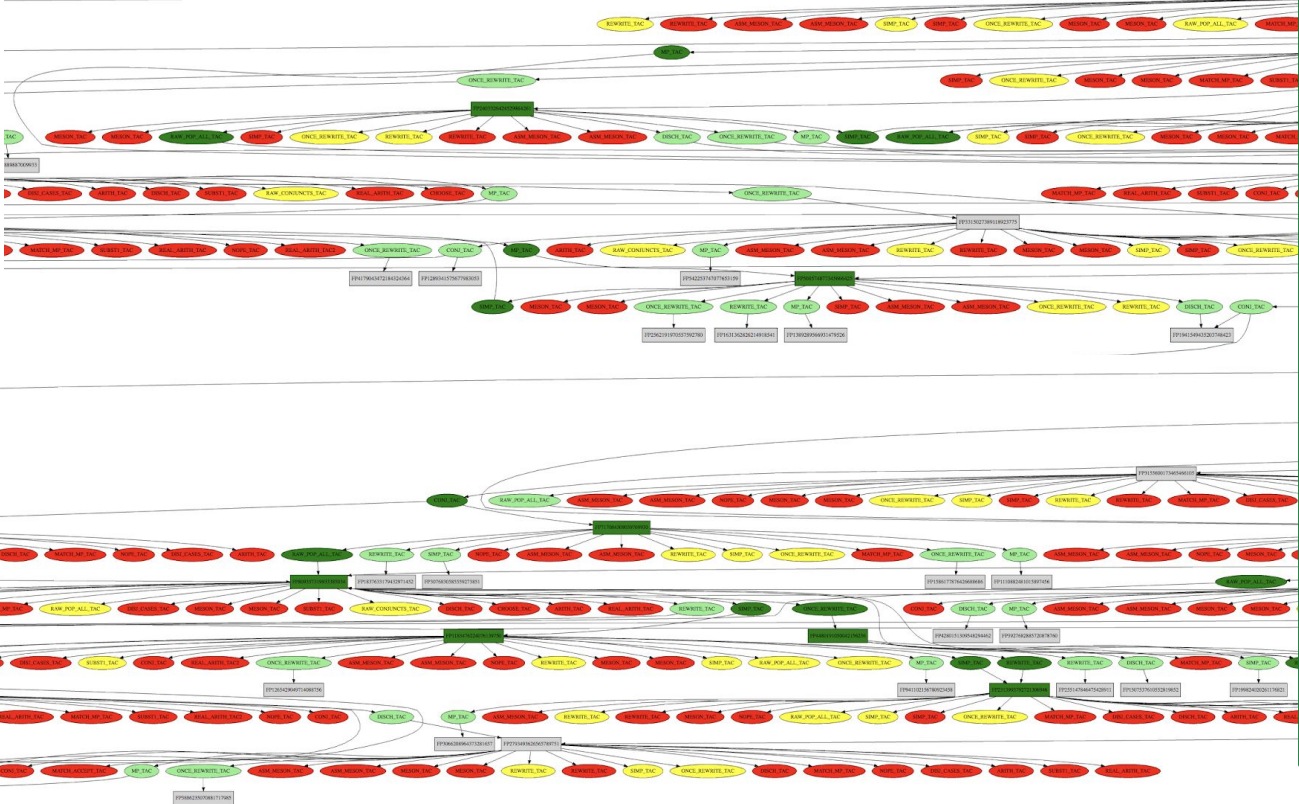
Proof Search Tree API

- Apply a tactic to any *goal* at any time.
- Controlled by any algorithm, e.g. neural algorithms.
- Automated merging of identical *goals*.
- On the fly tracking of:
 - Goals that are closed
 - Subgoals that can't help closing the main goal
- Collects statistics (e.g. running time, error codes).
- Serialized as ProofLog.

Proof Search Tree



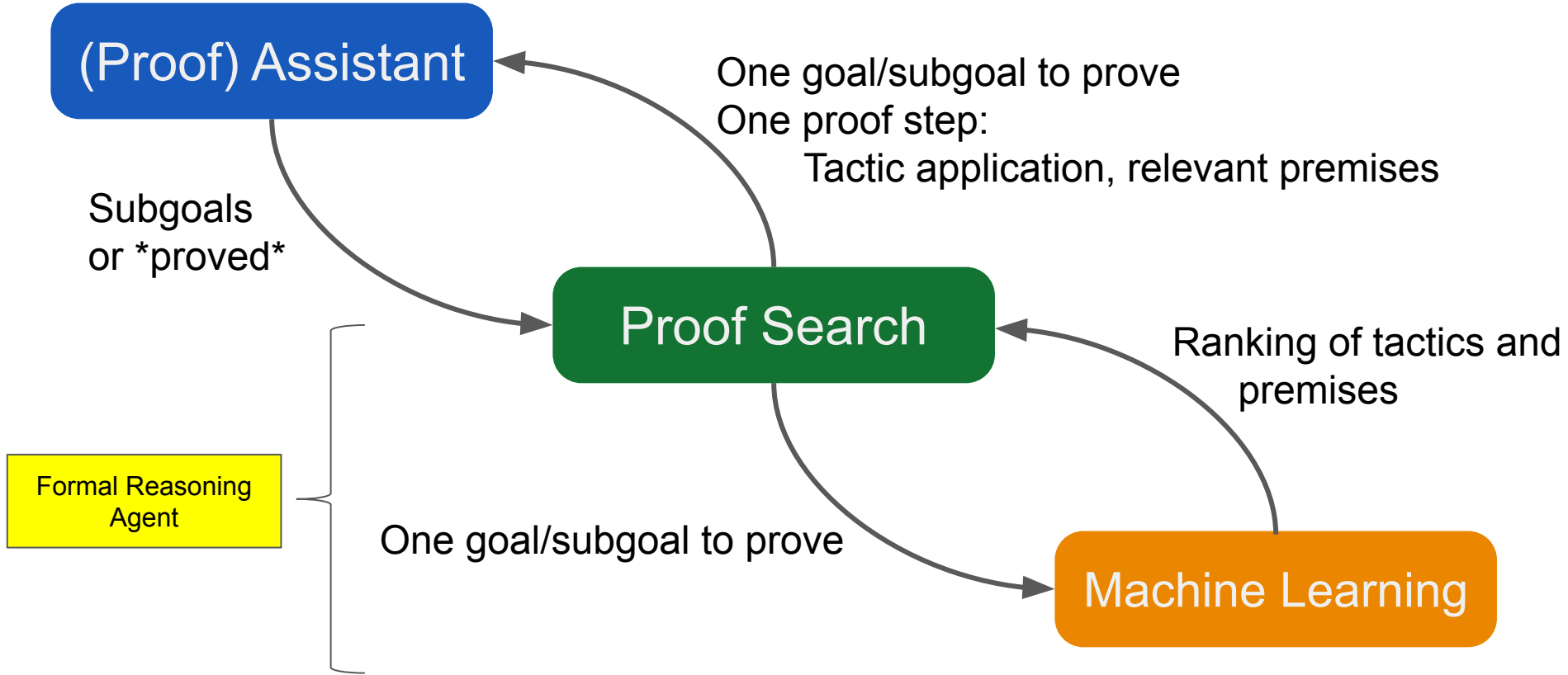
Proof Search Tree



Proof Search

- Our prover: simple BFS Prover built on this tree API, with limits on branching.
 - max_top_suggestions (default: 20)
 - max_successful_branches (default: 2)
 - max_explored_nodes (default: 100)
 - max_theorem_parameters (we used: 16)
- Built on Tree API, easy to extend for more interesting proof search.

APIs for Theorem Prover Developers and ML Researchers



Machine Learning

- Predictions API integrating with the **proof search**.
 - (Goal, Tactic ID) -> Score
 - (Goal, Premise) -> Score
- Our models, experiments: more in the next talk.

APIs for Theorem Prover Developers and ML Researchers

Assistant

RegisterTheorem
ApplyTactic

HOL-Light

Proof Search

- Manages the state of the proof search tree.
- Allows arbitrary nodes to be explored.

Machine Learning

Given:

- Current goal

Score:

- Tactic applied
- Premises used

Making available to researchers

Benchmark

Theorem Database

	Theorems	Definitions
--	----------	-------------

Core <i>required for creating in-built tactics</i>	2,320	240
---	-------	-----

Complex <i>separated into training, validation, testing</i>	16,623	396
---	--------	-----

FlySpeck <i>for evaluating generalization</i>	10,519	1,563
---	--------	-------

Making available to researchers

Data

- Proof Logs:
 - Synthetic proofs
 - Human proofs
- Proof Logs as TF Examples
 - Features:
 - Goal (string)
 - Labels:
 - Tactic applied (int)
 - Premises used (string)

Model

- Checkpoints of two-tower architecture from imitation learning and reinforcement learning.
- Sample training code.

Making available to researchers

Code

HOL Light (with our modifications)

[http://github.com/
brain-research/hol-light](http://github.com/brain-research/hol-light)

DeepHOL prover

[http://github.com/
tensorflow/deepmath](http://github.com/tensorflow/deepmath)

Docker images

HOL Light (server)

gcr.io/deepmath/hol-light

DeepHOL prover

gcr.io/deepmath/deephol

<http://deephol.org>

Code is on GitHub. Training data, checkpoints,
docker images also being made available.