Learning to Reason in Large Theories Without Imitation

Kshitij Bansal, Sarah M. Loos, Markus N. Rabe, Christian Szegedy

Slides by Jacob Nogas, MSc Computer Science

Outline of Talk

1. Background
    - ITP terminology
    - Proof search graph
    - RL
    - DeepHOL
2. New approach; imitation learning free
    - Premise Selection
    - Experimental results

ITP Terminology

- ITP: Interactive theorem prover; human (or ML system) interacts with proof assistant
- Goal: provable statement, ie. theorem
- Tactic:
  - Proof step
  - Represented as ID of preselected manipulation of goal that led to successful proof
  - Produces a list of subgoals
  - Success when tactic produces empty list of subgoals
  - Takes list of previously proven theorems (premise) as optional argument

Proof Search Graph

- Captures state of proof search
- Allows us to determine if proof for original goal is available
- Nodes: goals that have been seen
- Edges: tactic application (leads to new goals)
- Search for proof of goal by breadth first search

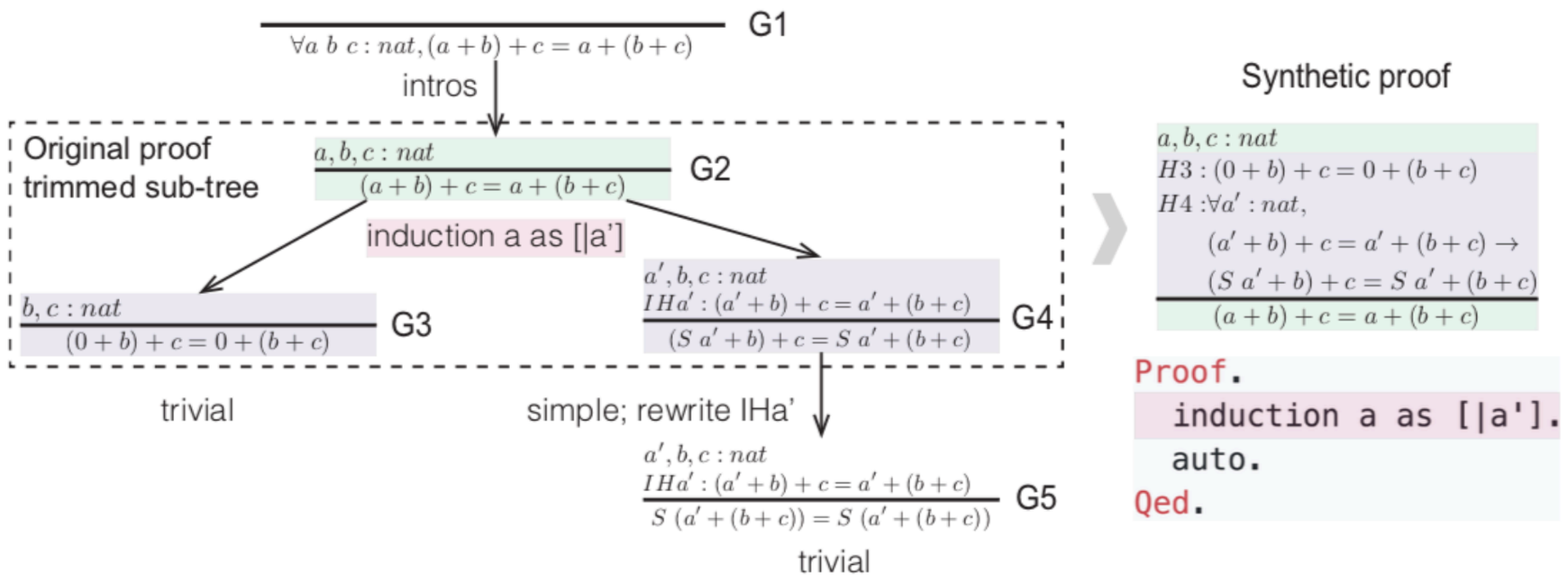

Figure F. Extracting a synthetic proof from the intermediate goal G2. Goals G3 and G4 are converted into premises in G2's local context. The synthetic proof corresponds to a trimmed sub-tree rooted at G2.

Reinforcement Learning - Framing

- Action: choose tactic, as well as premises
- State: Proof search graph
- State transition: New proof search graph populated with new sub-goals
- Reward: successful proof

Previous Work - DeepHOL

- Bansal et al. [2019] created the DeepHOL prover proves theorems in ITP setting with reinforcement learning

- Rely on imitation learning

- Key aspect of their reinforcement learning set up is the action generator network

DeepHOL - Action Generator

- During breadth first search, action generator neural network generates a ranked list of tactics and applies them in order

- Stops applying tactics when reach maximum number of unsuccessful tactic applications or minimum number of successful applications

- Search is stopped when a complete proof is found for the top level goal

Action Generator Details

- Ranks tactics in scoring vector $S(G(g))$, where S is linear layer producing logits of softmax classifier

- Ranks previously proven theorems in their usefulness as a tactic argument in transforming current goal towards closed proof

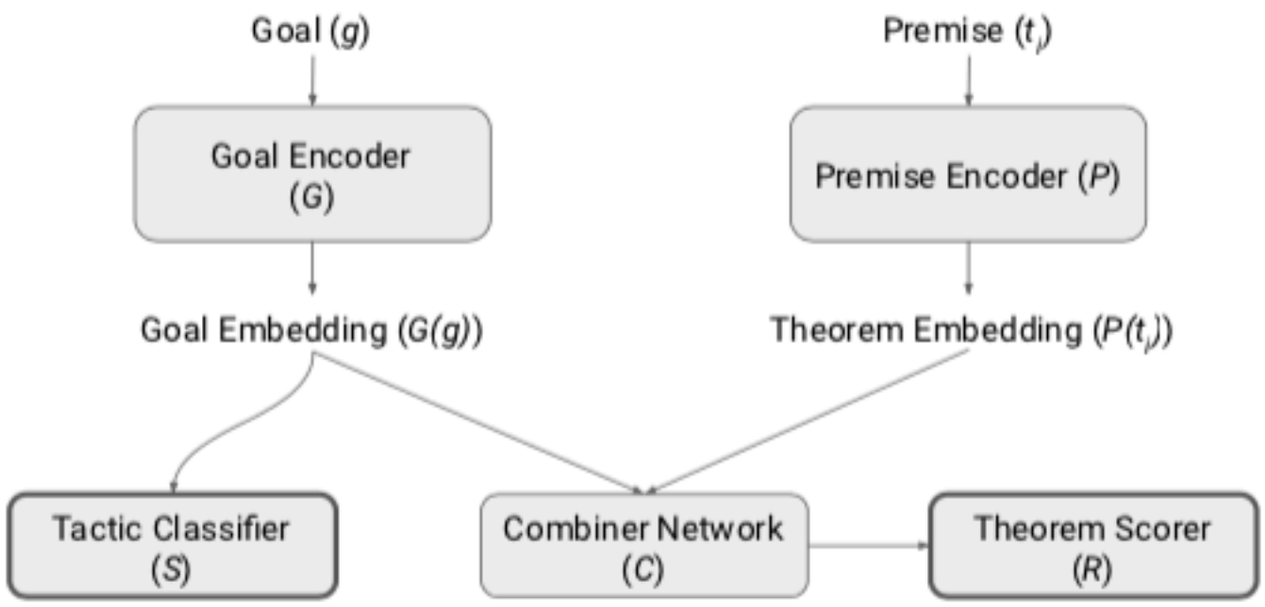

Figure 1: Two-tower neural architecture for ranking actions.

Why use Imitation?

- DeepHOL require the use of imitation learning as starting point in exploration

- Tactics can refer to definitions and theorems that have been proved, thus the action space is continuously expanding

- For example, the “rewrite” tactic performs a search in the current goal for a term to be rewritten by some of the equations provided for the tactic parameters (premises)

Exploring Premises

- Premise selection is crucial for good performance
- DeepHOL selects premises based on ranking network
- Without imitation, DeepHOL runs into issues:
  - Randomly initialized ranking model fails to learn useful similarity metric for comparing goals and premises
  - Fails to explore explore premises

Imitation Learning Drawbacks

- Learning without imitation learning addresses the key problem of exploration directly

- Theorem proving on new proof assistant platforms would require a new training data of existing proofs

- Existing proofs may not exist

- Performing better than humans requires going beyond imitating that which is achieved by existing human demonstrations

Proposed Solution

- This paper proposes a solution to exploring premises which does not use imitation learning

- Initialize network by training on a seed dataset for one round of proving with premise selection network that ranks premises by the cosine similarly between goal embedding and premise embedding (from two-tower neural net); P_1 are the top $k_1$scoring premises

- Add exploration by mixing in new elements to the proposed set of premises. Select premises from $P_1 \cup P_2$, P_2 is selected from one of the methods in the following slide

Selecting P_2

- **PET:** Cosine similarity as before, but then perturb with random noise, re-rank, and choose top k_2 as P_2

- **BoW1:** P_2 is selected as top k_2 scoring premises from cosine similarity between randomized bag-of-word (BoW) embeddings of goal and premises weighted by random noise

- **BoW2:** Same as BoW1, but with modification to random weighting (details in appendix)

Experimental Results - Training Set

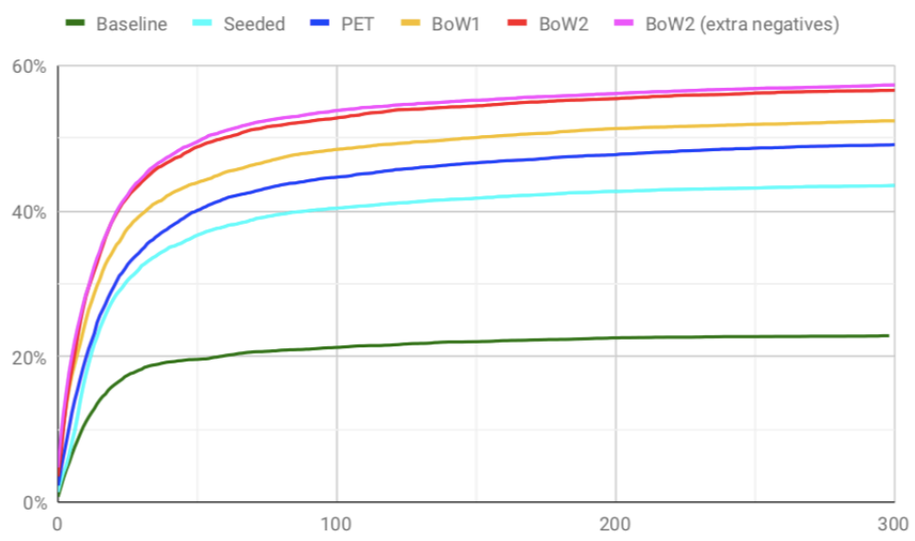

Figure 2: Theorems proved cumulatively on the training set.

Figure 3: Percentage of theorems proved in each round (out of 2000 randomly selected theorems to be proved in each round).

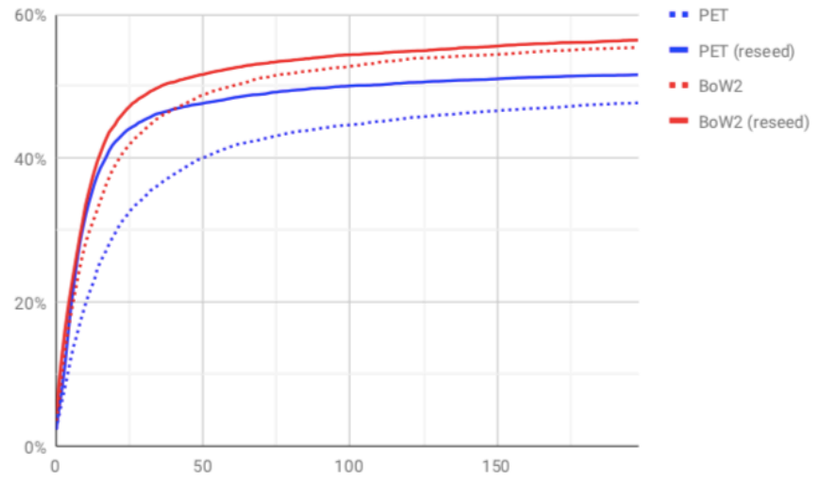

Figure 4: Theorems proved cumulatively on the training set by the first and second iteration of the reinforcement learning loop.

Experiment	Total till 70th (% of training)	Total till @round (% of training)	Proved @70th (on validation)	Proved @round (on validation)
Baseline	20.63%	22.85% @298	16.96%	18.32% @290
Seeded	38.79%	43.65% @336	31.10%	31.13% @60
PET	42.64%	49.76% @391	30.26%	32.18% @170
BoW1	46.32%	53.11% @391	32.00%	32.00% @70
BoW2	51.20%	57.61% @437	33.02%	33.92% @140
PET (reseed)	48.91%	52.89% @416	33.73%	34.26% @20
BoW2 (reseed)	53.16%	57.72% @357	33.33%	34.10% @90
BoW2 (extra -ves)	52.06%	57.42% @318	35.78%	36.62% @290
Union	57.70%	61.67%	N/A	N/A

Table 1: Total percentage of proofs on training set of 10200 theorem found by each loop till 70th round, and till when it was aborted. Percentage of theorems closed using various models on the validation set comprising of 3225 theorems at 70th round. We also report the best fraction proven by each loop where this evaluation was performed every 10th round.

model that used a combination of reinforcement learning and imitation learning. Even more encouraging that the union of all of our reinforcement loops together managed to collect more successful proofs for the training set (61.7%) than the union of all proofs given in the original DeepHOL paper (58.0%). Given that our approach does not utilize any human data, this is an encouraging sign, since

Experimental Results - Validation Set

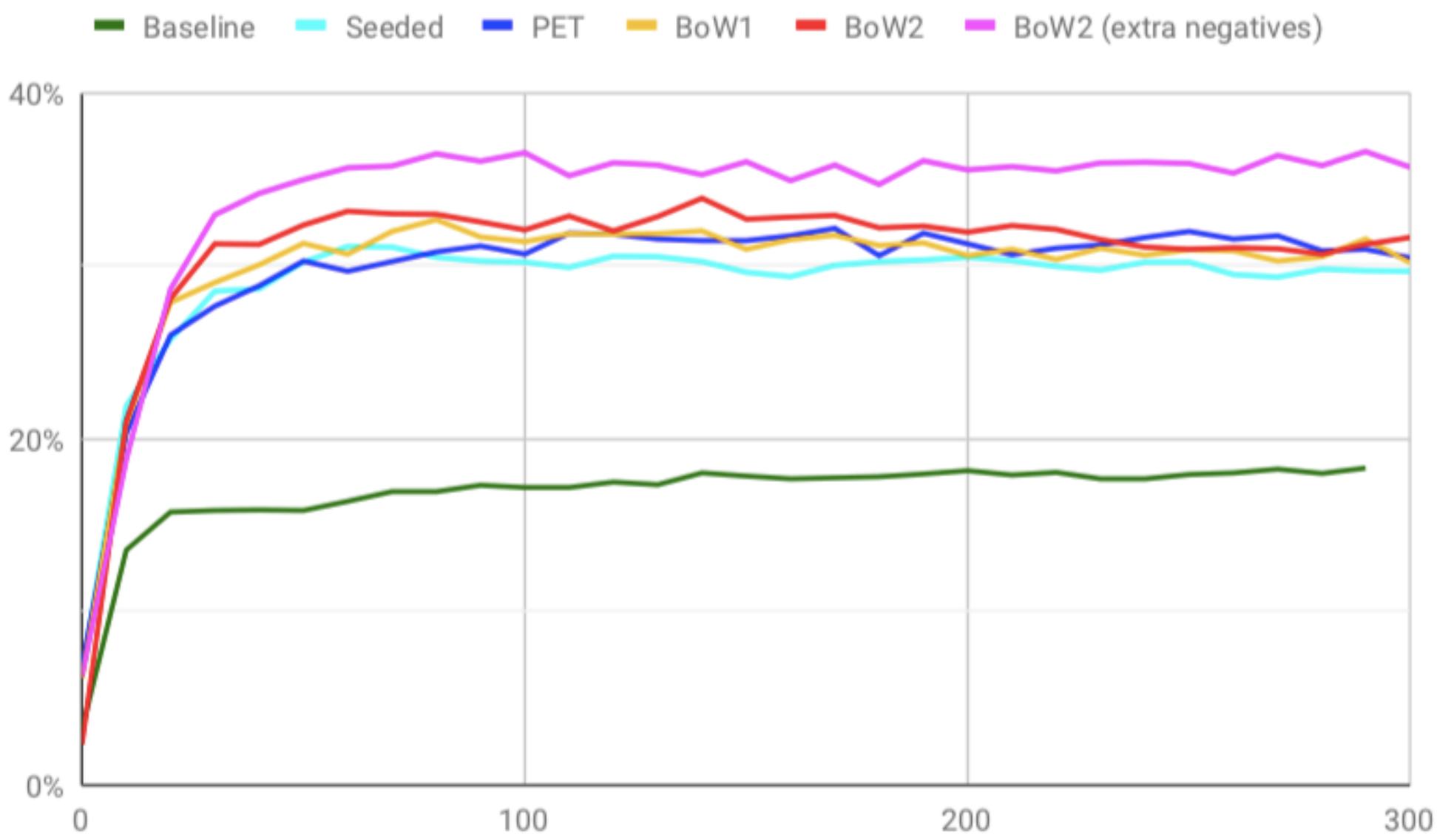

Figure 5: Percentage of theorems proved on the validation set at checkpoint every 10th round.

Appendix - Premise Selection

PET: Using the the premise-embedding tower (PET) P of the policy network, we compute the cosine similarity $s_i = s(P(g), P(p_i))$ between goal g and each possible (preceding) premise p_i in the theorem database. After selecting P_2', the top-k_2' in this ranking (k_2' was selected to be 100 in our experiment), we rerank P_2' by perturbing the similarity between the elements by adding ν_i to each cosine similarity s_i, where ν_i is sampled from a Gaussian noise with mean 0 and stddev 0.2 for each premise p_i independently. Then P_2 is selected to be the top k_2 highest scoring elements of P_2' with respect to $s_i + \nu_i$.

BoW1: P_2 is selected as a top-k_2 highest scoring elements from the the randomized bag-of-word (BoW) embeddings b of goal g and premise p_i. First we compute the weighted bag of word encoding of each sentence. First we assign a different one-hot vector $w(t)$ associated with each one the 884 tokens t occurring in our dictionary. Then we reweight each of the embeddings by $\tilde{w}(t_i) = \nu_i f_i w(t_i)$, where f_i is the inverse document frequency and ν_i is sampled from log normal distribution with the underlying normal distribution having zero mean and unit variance. For each goal g, we rank the premises p_i by the cosine similarity $s(\tilde{w}(g), \tilde{w}(p_i))$ between their embeddings and pick the top-k_2 highest scoring premises.

BoW2: Same as BoW1, but with $\tilde{w}(t_i) = \frac{|1+\nu_i|}{f_i} w(t_i)$, where f_i is the frequency of t_i in the whole dataset and ν_i is sampled from normal distribution with zero mean and unit variance.

- Fails when not all conditions are met, tactic cannot be applied

Reference Page

1. Kshitij Bansal, Sarah M Loos, Markus N Rabe, Christian Szegedy, and Stewart Wilcox. Holist: An en- vironment for machine learning of higher-order theorem proving. arXiv preprint arXiv:1904.03241, 2019.