Retrieval-Augmented Proof Step Synthesis

May 2021

Christian Szegedy, Markus Rabe, and Henryk Michalewski

Google Research, Mountain View, CA, USA
(szegedy,mrabe,henrykm)@google.com

Abstract

Automated theorem proving is relying increasingly on sophisticated language modelling approaches for synthesizing proof steps (tactic applications, rewrite rules). However, one of the most significant difficulties of proof search is finding the correct premises to be used. This raises the problem of combining premise selection with language modeling. There are two obvious avenues towards this goal: synthesizing the full theorem text to be utilized as a premise, or using a separate premise selection model that is used as an extra component to be used when referencing theorems. In this paper, we suggest a new solution based on language modelling that allows premise selection to become an organic component of the deep learning model and is not trained in separation. We compare this approach to theorem proving using a combination of pretrained premise selection and tactic synthesis on the HOList dataset.

1 Introduction

Premise selection [9] is a central problem of theorem proving in large theories. Realistic benchmarks of large-scale automated theorem proving are based on corpora of human-formalized mathematics and can include theories with hundreds of thousands of theorems to be utilized [10]. This suggests an evaluation setup in which the proof of each theorem is allowed to utilize only premises that were available for the original author of the corpus. Typically, the theorems are sorted in some topological order of dependence and only theorems preceding the current theorem are allowed during proving. In order to avoid information leaks, the machine learning models for premise selection have to be retrained incrementally for each theorem to be proved. This is a realistic scenario for machine learning models that can be updated very quickly, but has posed a challenge for deep-learning-based approaches. For this purpose, premise selection systems using deep learning have been evaluated by a two-phases methodology, in which the performance is measured on a held-out set of the theorems to be proved, but is trained on all possible premises, first. While this approach is compatible with most modern deep-learning-based setups, it has a potential for information leak as the premise selection for theorems might be based on information of theorems proved layer in the database. This does not model the real-life constraints in a conservative manner. Most current deep-learning-based theorem proving systems are evaluated based on the same questionable assumptions. While some results [2] on FlySpeck suggest that the effect of training on future theorems is not too critical, this methodology also reinforces the practice of developing systems that are not easily used in an incremental setup and does not measure this important aspect of the system faithfully.

2 Related Work

Theorem proving in large theories and premise selection was pioneered by [9, 10] and later in [5] for first order theorem proving. DeepMath [1] proposed a deep-learning-based approach for premise selection in a similar setup for the Mizar [6] corpus, however their methodology suffers from the same issue of training on the proof of future theorems. Later, TacTicToe [3] suggested premise selection for higher-order-logic theorem proving. HOList [7] was suggested to combine HOL-based theorem proving with graph-neural-network-based premise-selection. However, deep-learning-based language modeling has shown surprising effectiveness for this purpose [12] and recently, GPT-f [8] has demonstrated the usefulness of large language models for proof-automation for Lean.

3 Incremental Proof Step Synthesis

Here, we present an incremental proof step synthesis approach that relies heavily and integrates seamlessly with the state-of-the-art neural architectures: especially with transformer networks [11] designed for language modelling. While language modelling has been increasingly and successfully used for synthesizing proof steps, it is typically used in a setup in which the transformer model is given enough training steps to memorize the statements to be used. This way, the network can produce either the full theorem text or a reference by naming the theorem. However, as we will demonstrate, this approach results in theorem proving performance that lags behind systems that were trained for premise selection directly via contrastive training [1]. In this talk, we present an approach that augments transformers with a retrieval based model similarly to [4]. Our approach differs from pure language modeling in that our approach allows for looking up theorems immediately after they are proved: the embeddings of theorems are stored in a database that is consulted by the transformer model using a side-attention mechanism into this dynamically database and the keys of the embeddings are updated using the standard backpropagation mechanism of that attention later and the theorem names can be extracted from the value associated with those premises. The advantage of this approach is that it integrates directly with the transformer architecture and the lookup is trained, incrementally using standard attention layers which includes a large number of negative premises and therefore alleviates the need for hard negative mining. Still, the inference mechanism utilizes standard autoregressive decoding and the final result can consult any premises that are appropriate in the given context. This is different from previous approaches [7] in which the premises were preselected and the decoder did not have full control of the synthesized proof step. We present experiments with a system that integrates this memory lookup into the transformer architecture and trained in end-to-end manner and verify that it is competitive with those approaches that utilize a separate premise selection model trained explicitly for this purpose. This paves the way towards simpler systems that allow knowledge utilization conditioned on large knowledge bases in incremental fashion and requiring less training steps.

References

[1] Alexander A Alemi, François Chollet, Geoffrey Irving, Niklas Eén, Christian Szegedy, and Josef Urban. Deepmath-deep sequence models for premise selection. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, pages 2235–2243, 2016.

[2] Kshitij Bansal, Sarah M Loos, Markus N Rabe, Christian Szegedy, and Stewart Wilcox. HOList: An environment for machine learning of higher-order theorem proving. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 454–463. PMLR, 2019.

[3] Thibault Gauthier, Cezary Kaliszyk, and Josef Urban. TacticToe: Learning to reason with HOL4 tactics. In Thomas Eiter and David Sands, editors, *LPAR-21, 21st International Conference on Logic for Programming, Artificial Intelligence and Reasoning, Maun, Botswana, May 7-12, 2017*, volume 46 of *EPiC Series in Computing*, pages 125–143. EasyChair, 2017.

[4] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In Hal Daumé III and Aarti Singh, editors, *International Conference on Machine Learning*, pages 3929–3938. PMLR, 2020.

[5] Cezary Kaliszyk and Josef Urban. Mizar 40 for mizar 40. *Journal of Automated Reasoning*, 55(3):245–256, 2015.

[6] The Mizar Mathematical Library. Accessed: 2018/01/18.

[7] Aditya Paliwal, Sarah Loos, Markus Rabe, Kshitij Bansal, and Christian Szegedy. Graph representations for higher-order logic and theorem proving. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 2020.

[8] Stanislas Polu and Ilya Sutskever. Generative language modeling for automated theorem proving, 2020.

[9] Josef Urban. MPTP–motivation, implementation, first experiments. *Journal of Automated Reasoning*, 33(3-4):319–339, 2004.

[10] Josef Urban, Geoff Sutcliffe, Petr Pudlák, and Jiří Vyskočil. Malarea sg1-machine learner for automated reasoning with semantic guidance. In Baumgartner Peter Dowek Gilles Armando, Alessandro, editor, *International Joint Conference on Automated Reasoning*, pages 441–456. Springer, 2008.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008, 2017.

[12] Qingxiang Wang, Chad Brown, Cezary Kaliszyk, and Josef Urban. Exploration of neural machine translation in autoformalization of mathematics in Mizar. In Jasmin Blanchette and Cătălin Hriţcu, editors, *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs*, pages 85–98, 2020.