Two Learning Operators for Clause Selection Guidance*

Martin Suda

Czech Technical University in Prague, Czech Republic

Clause selection is arguably the most important choice point in saturation-based automated theorem proving [8]. It amounts to deciding, in each iteration of the saturation loop, which next clause to *select* for *activation*, i.e., for the promotion from the *passive* set to the *active* set, which means enabling this selected clause's participation in generating inferences. In the commonly used Discount loop [1], a refutation can only be successfully completed when all clauses that the proof consists of have been selected.

Recently, several successful systems for machine-learned clause selection guidance have been developed, most notably ENIGMA in several incarnations [4, 5, 6, 3] and Deepire [9, 10]. Although the underlying machine learning (ML) models differ (e.g., boosted trees, neural networks, etc.), the systems all learn from previously observed prover derivations, training a *binary classifier* to recognize as positive those clauses that have appeared in the discovered proofs, against the background of all the recorded selected clauses. Although this is typically not stated explicitly, the learning setup in ENIGMA or Deepire can be seen to assume a working clause selection heuristic and seeks to improve upon it through the integration of the learned advice.

On the other hand, approaches inspired by the reinforcement learning (RL) paradigm [13, 14], strive to learn a standalone clause selection heuristic from scratch [2, 11]. Although the found proof is still used as the gold positive label for the clauses which appear in it, the background against which the model learns consists of the passive clauses (as opposed to the activated ones) and, moreover, each recorded derivation step can use the precise snapshot of the passive set content at that the given moment. This RL-inspired approach a priory leads to a *regression* model (as opposed to a classifier), although the technical details (notably the loss function) are surprisingly similar.

Despite the mentioned differences, both the ENIGMA-style and the RL-inspired learning operators ultimately aim to achieve the same pragmatic goal, namely to improve the prover performance by learning from experiences gathered while solving problems from some benchmark family or distribution of interest. However, to the best of my knowledge there is no direct experimental comparison of the two. I will center my talk around exactly that.

Using a version of Vampire [7] extended 1) to output information about successful prover runs (most notably recording the active and passive clause set traffic and, for every mentioned clause, a small set of easy-to-obtain features) and 2) to be guided by an ENIGMA-style clause classifier (discriminating based on the mentioned clause features) as well as, alternatively, by a model trained in the RL-inspired fashion to assign probabilities (via softmaxing; again as a function of the features) for selecting any of the current passive clause, I will report on an experimental comparison of these two basic learning operators on problems from the TPTP library [12]. In both cases, a small neural network (a MLP) will be trained. In the ENIGMA-style guidance, various ways of interfacing the model can be tried, where a layered approach with lazy evaluation is expected to perform the best [9]. On the RL-inspired side, it is natural to have the neural model determine an order for a single queue representation of the passive set. I will also discuss the challenges for efficiently *sampling* from such a queue (according to the predicted probabilities), which best matches the RL paradigm and encourages exploration, as opposed to just always taking the best clause (which seems less faithful, but is trivial to implement). Time permitting, I will also mention looping and iterative guidance improvement.

*This work was supported by the Czech Science Foundation project no. 24-12759S.

References

[1] J. Avenhaus, J. Denzinger, and M. Fuchs. DISCOUNT: A system for distributed equational deduction. In *Rewriting Techniques and Applications, 6th International Conference, RTA-95, Kaiserslautern, Germany, April 5-7, 1995, Proceedings*, vol. 914 of *Lecture Notes in Computer Science*, pp. 397–402. Springer, 1995.

[2] M. Crouse, I. Abdelaziz, B. Makni, S. Whitehead, C. Cornelio, P. Kapanipathi, K. Srinivas, V. Thost, M. Witbrock, and A. Fokoue. A deep reinforcement learning approach to first-order logic theorem proving. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 6279–6287. AAAI Press, 2021.

[3] J. Jakubuv, K. Chvalovský, M. Olšák, B. Piotrowski, M. Suda, and J. Urban. ENIGMA anonymous: Symbol-independent inference guiding machine (system description). In *Automated Reasoning - 10th International Joint Conference, IJCAR 2020, Paris, France, July 1-4, 2020, Proceedings, Part II*, vol. 12167 of *Lecture Notes in Computer Science*, pp. 448–463. Springer, 2020.

[4] J. Jakubuv and J. Urban. ENIGMA: efficient learning-based inference guiding machine. In *Intelligent Computer Mathematics - 10th International Conference, CICM 2017, Edinburgh, UK, July 17-21, 2017, Proceedings*, vol. 10383 of *Lecture Notes in Computer Science*, pp. 292–302. Springer, 2017.

[5] J. Jakubuv and J. Urban. Enhancing ENIGMA given clause guidance. In *Intelligent Computer Mathematics - 11th International Conference, CICM 2018, Hagenberg, Austria, August 13-17, 2018, Proceedings*, vol. 11006 of *Lecture Notes in Computer Science*, pp. 118–124. Springer, 2018.

[6] J. Jakubuv and J. Urban. Hammering mizar by learning clause guidance (short paper). In *10th International Conference on Interactive Theorem Proving, ITP 2019, September 9-12, 2019, Portland, OR, USA*, vol. 141 of *LIPIcs*, pp. 34:1–34:8. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[7] L. Kovács and A. Voronkov. First-order theorem proving and vampire. In *Computer Aided Verification - 25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings*, vol. 8044 of *Lecture Notes in Computer Science*, pp. 1–35. Springer, 2013.

[8] S. Schulz and M. Möhrmann. Performance of clause selection heuristics for saturation-based theorem proving. In *Automated Reasoning - 8th International Joint Conference, IJCAR 2016, Coimbra, Portugal, June 27 - July 2, 2016, Proceedings*, vol. 9706 of *Lecture Notes in Computer Science*, pp. 330–345. Springer, 2016.

[9] M. Suda. Improving enigma-style clause selection while learning from history. In *Automated Deduction - CADE 28 - 28th International Conference on Automated Deduction, Virtual Event, July 12-15, 2021, Proceedings*, vol. 12699 of *Lecture Notes in Computer Science*, pp. 543–561. Springer, 2021.

[10] M. Suda. Vampire with a brain is a good ITP hammer. In *Frontiers of Combining Systems - 13th International Symposium, FroCoS 2021, Birmingham, UK, September 8-10, 2021, Proceedings*, vol. 12941 of *Lecture Notes in Computer Science*, pp. 192–209. Springer, 2021.

[11] M. Suda. Elements of reinforcement learning in saturation-based theorem proving. In *7th Conference on Artificial Intelligence and Theorem Proving AITP 2022 – proceedings*, 2022. http://aitp-conference.org/2022/abstract/AITP_2022_paper_11.pdf.

[12] G. Sutcliffe. The TPTP Problem Library and Associated Infrastructure. From CNF to TH0, TPTP v6.4.0. *Journal of Automated Reasoning*, 59(4):483–502, 2017.

[13] R. S. Sutton and A. G. Barto. *Reinforcement learning - an introduction.* Adaptive computation and machine learning. MIT Press, 1998.

[14] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8:229–256, 1992.