Teaching Vampire New Tricks: An Experimental Study of Neural Clause Selection*

Karel Chvalovský[1], Martin Suda[1], and Josef Urban[2]

[1] Czech Technical University in Prague, Czech Republic
{karel.chvalovsky,martin.suda}@cvut.cz
[2] AI4REASON and University of Gothenburg
josef.urban@gmail.com

Neural clause-selection guidance integrated into the Vampire theorem prover [1] was recently shown [8, 9] to substantially improve the success rate of the prover's default strategy on the TPTP benchmark [10], using an efficient neural clause-scoring architecture and a RL-inspired approach for iterative model training and prover improvement. The corresponding Deepire II system was introduced also at AITP 2025 [7].

In this work, we ask several follow-up questions: How well does the neural guidance transfer to other benchmark domains, particularly those derived from interactive theorem prover (ITP) hammers? How does it interact with proving strategies and large-scale portfolio construction? We report results from a comprehensive experimental study across four benchmark families.

Per-Dataset Boosting. We train a guiding model on each of four datasets (more precisely, on random fractions of reasonable size): TPTP (15 500 problems), Mizar40 [6] (57 880), Isabelle/Sledgehammer [3] export [5] (276 363), and CoqHammer [4] (82 366 problems). Using Vampire's default strategy as the basis, we obtain large per-dataset improvements: +26.5% on TPTP, +113% on Mizar40 (+101% on a held-out test set), +36.6% on Isabelle (+24.7% test), and +248% on CoqHammer (+93.5% test). The hardest dataset (CoqHammer) benefits the most, though not as much of the gain persists to the test set, likely due to the stricter train/test-split hygiene (disjoint Coq packages) used there.

Cross-Dataset Transfer. We evaluate each trained model on all four datasets. The result is a clear negative: as a model specializes to its training dataset and improves on it, it consistently gets *worse* on any other dataset, to the point of underperforming the plain default strategy. Even models trained on the diverse TPTP library fail to transfer once Mizar-origin problems are excluded. This suggests the network capitalizes on encoding-specific patterns (e.g., type guards vs. type tags in different ITP encodings) rather than learning general proof-search principles.

Strategy Interaction. We construct five complementary strategies via a greedy hill-climbing algorithm and train a guiding model for each. Each strategy is individually boosted by ∼24% on average. However, complementarity shrinks with boosting: the 5-strategy portfolio gains only 13.9% from neural guidance (the neurally guided prover delegates clause selection entirely to the model, thus erasing some of the heuristic differences between strategies). We can also train a single model on all five strategies and it performs almost as well as five separate models. Following Goertzel [1], we also test an unconstrained calculus strategy (plain resolution and paramodulation, no ordering constraints), which only solves 40.4% of TPTP on its own but reaches 61.3% after boosting—surpassing the unguided default (55.3%).

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

Portfolio Construction. Using spider-style strategy search [2], we generate both plain and neurally-boosted strategies over 20 days. A combined (plain+neural) greedy schedule covers 85.5% of TPTP, compared to 82.1% for the plain-only schedule (+4.1%). At the practically relevant time mark of ∼12 s per problem, the combined schedule solves 76.1% versus 67.8% (+12.2%), and reaches its final coverage faster (in 0.5 h single-core vs. 0.65 h). Neural guidance thus offers proportionally stronger gains under tight time budgets.

Outlook. Neural clause-selection guidance of Deepire II consistently and substantially helps per domain but does not transfer across benchmark families. Portfolio gains are real but exhibit diminishing returns as strategic complementarity already captures much of the search space. Future directions include training on larger subsets of the available quarter-million Isabelle problems, analyzing the learned clause and symbol representations, and exploring architectures better suited for cross-domain transfer.

References

[1] F. Bártek, A. Bhayat, R. Coutelier, M. Hajdú, M. Hetzenberger, P. Hozzová, L. Kovács, J. Rath, M. Rawson, G. Reger, M. Suda, J. Schoisswohl, and A. Voronkov. The Vampire diary. In *Computer Aided Verification - 37th International Conference, CAV 2025, Zagreb, Croatia, July 23-25, 2025, Proceedings, Part III*, vol. 15933 of *LNCS*, pp. 57–71. Springer, 2025.

[2] F. Bártek, K. Chvalovský, and M. Suda. Regularization in spider-style strategy discovery and schedule construction. In *IJCAR 2024*, vol. 14739 of *LNCS*, pp. 194–213. Springer, 2024.

[3] S. Böhme and T. Nipkow. Sledgehammer: Judgement day. In *IJCAR 2010*, vol. 6173 of *LNCS*, pp. 107–121. Springer, 2010.

[4] L. Czajka and C. Kaliszyk. Hammer for Coq: Automation for dependent type theory. *J. Autom. Reason.*, 61(1-4):423–453, 2018.

[5] Z. A. Goertzel, J. Jakubuv, C. Kaliszyk, M. Olšák, J. Piepenbrock, and J. Urban. The Isabelle ENIGMA. In *13th International Conference on Interactive Theorem Proving, ITP 2022, Haifa, Israel, August 7-10, 2022*, vol. 237 of *LIPIcs*, pp. 16:1–16:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

[6] C. Kaliszyk and J. Urban. Mizar 40 for mizar 40. *J. Autom. Reason.*, 55(3):245–256, 2015.

[7] M. Suda. Deepire II = RL(GNN+2RvNN). In *10th Conference on Artificial Intelligence and Theorem Proving AITP 2024 – proceedings*, 2025. `https://aitp-conference.org/2025/abstract/AITP_2025_paper_19.pdf`.

[8] M. Suda. Efficient neural clause-selection reinforcement. In *Automated Deduction - CADE 30 - 30th International Conference on Automated Deduction, Stuttgart, Germany, July 28-31, 2025, Proceedings*, vol. 15943 of *LNCS*, pp. 403–422. Springer, 2025.

[9] M. Suda. Efficient neural clause-selection reinforcement. *CoRR*, abs/2503.07792, 2025.

[10] G. Sutcliffe. Stepping Stones in the TPTP World. In *IJCAR 2024*, number 14739 in LNAI, pp. 30–50, 2024.