Internal Field Report: Horn Logic Embeddings (Zhang et al., NeSy 2025)

Report 3 for Ramifications Can Transformers Reason Logically? (Pan et al.) and Actionable Extensions

Chris Esposo
Georgia Institute of Technology

October 15, 2025

Abstract

Zhang et al. propose a metric-learning recipe for embeddings that respect *unification* in Horn logic and show that such representations substantially shrink the search explored by a guided backward-chaining reasoner. We (i) deconstruct their system into the representation/learning/control stack and restate the core loss geometry; (ii) map those elements to our DPLL-style SAT setting in Pan et al. [2025]; (iii) derive concrete training and inference designs that let us import "unification geometry" as a *literal/decision* prior under a strict verifier.

1 What Zhang et al. built (and what I heard in the room)

System decomposition. The talk cleanly separates (1) an **embedding model** $e(\cdot)$ mapping atoms to $\mathbb{R}^d$, (2) a **scoring model** that evaluates (goal, rule) pairs, and (3) a **guided reasoner** that uses the learned scores to steer backward chaining. *Representation matters*: they focus on learning $e(\cdot)$ so that *unifying* atoms are close and non-unifying atoms are far; the scoring/reasoner then benefits downstream. This is also how the speaker framed it live ("we segment [the] reasoning system into three pieces: embedding, scoring, search").

Loss geometry and data. They train with *triplet loss* on procedurally generated $(\text{anchor}, \text{pos}, \text{neg})$ triples where *pos* unifies with the anchor and *neg* does not. Contributions: (i) oversample anchors with repeated terms (to encode variable-equality constraints); (ii) balance easy/medium/hard positives/negatives; (iii) periodic *hard mining* cycles that refit on the highest-loss subset.[1] The talk explicitly emphasized triplet-loss framing ("Google's Triplet Loss for face recognition, anchor/positive/negative"), which matches the paper.

Downstream effect. On synthetic KBs (250/375/500 statements), the *new embeddings* cut explored nodes by large factors vs. their prior embeddings, at similar failure rates (Table 1). For example at 250, mean nodes drop from $\sim$982 to $\sim$76.2. Representation alone gave most of the realized efficiency gains in their guided prover.

[1] Hard-mining protocol described around Sec. 4.3; triplets synthesized at scale and reweighted toward semi-hard/hard examples.

Positioning. This sits in a broader line of neural guidance for logic: differentiable unification/backward chaining [Rocktäschel and Riedel, 2017], learning-guided ATP (ENIGMA and successors), and automaton-augmented retrieval (RetoMaton). Restricting to Horn is defensible—Horn-SAT admits linear-time closure [Dowling and Gallier, 1984].

2 Our setting: how it differs and rhymes

Recall of our result. We prove an existence construction: a decoder-only Transformer with $O(p^2)$ parameters that implements a DPLL-like CoT to *decide* 3-$\mathsf{SAT}_{p,c}$; compiled models achieve 100% exactness for $p \leq 20$ near the phase transition and trained models track traces well in-range, with length-generalization cliffs beyond.

Contrasts. (i) **Horn vs. general CNF.** Zhang et al. reason over Horn rules with unification; we run DPLL + unit-propagation over arbitrary CNF. (ii) **What is learned.** They learn representations and a scoring head to guide symbolic backward chaining; we construct a verifier-aligned algorithm and optionally learn a *policy* to emit the next literal plus an explanation trace. (iii) **Where embeddings help.** Their gains come from *representation geometry* that respects unification; our bottlenecks are *decision quality* and *length generalization*. The bridge: import unification-style geometry as a *decision prior* while preserving our exact checks.

3 From unification to unit-propagation geometry

3.1 Two primitives to align

Unification kernel. In Horn, unification is a discrete relation $\text{Unify}(\alpha, \beta) \in \{0, 1\}$ computed by most-general substitutions. Zhang et al. seek $e(\cdot)$ with

$$\|e(\alpha) - e(\beta)\| \text{ small} \iff \text{Unify}(\alpha, \beta) = 1,$$

and maximize a triplet margin against non-unifying β^-.

Unit-propagation kernel. In CNF, a clause C under partial assignment A becomes *unit* iff exactly one literal in C is not

falsified by A; the surviving literal is forced. Write

$$K_{\mathrm{UP}}(A, C) := \mathbb{1}\big[C \text{ is unit under } A\big],$$
$$\ell^\star(A, C) := \text{the forced literal if } K_{\mathrm{UP}}{=}1.$$

Pan et al. show this can be computed by vector tests (clause vs. not-false encodings) inside our compiled model.[2]

3.2 Design goal

Learn embeddings $\phi(A)$ and $\psi(C)$ such that a bilinear score $s(A, C) := \langle \phi(A), \psi(C) \rangle$ ranks clauses by how close they are to unit under A and further places $\psi(C)$ near $\psi(C')$ when C' is a *single-flip* variant that flips $\ell^\star$. Intuition: this is the DPLL analogue of placing unifying atoms together.

3.3 Triplets for CNF (difficulty-aware)

We can mirror Zhang et al.'s triplet protocol with CNF semantics:

- **Anchor:** A (current partial assignment) or $(A, \hat{C})$ where $\hat{C}$ is the smallest active clause.
- **Positive:** C^+ such that $K_{\mathrm{UP}}(A, C^+) = 1$ and/or the literal $\ell^\star(A, C^+)$ matches the golden trace decision in our compiled proofs.
- **Negative:** C^- sampled with controlled hardness:
  1. *Easy*: C^- already satisfied or contradicted under A.
  2. *Medium*: C^- needs 2 flips to become unit (Hamming-2 to a unit-clause event).
  3. *Hard*: C^- becomes unit after a single non-forced flip; or it would force a *different* literal than the trace's next move.

This is a faithful transposition of their easy/medium/hard synthesis to CNF; we also import periodic *hard mining* (validate every n epochs, refit on top-loss half of triplets).

Why this helps. In their setting, unification-respecting geometry allows the scorer to focus rollouts on plausible rule applications. In ours, K_{UP}-respecting geometry should pull *the unit frontier* toward the current A, smoothing the decision landscape before the exact verifier fires. Representation first; the verifier remains the arbiter.

4 Potential integration points

4.1 (I) CNF embeddings as a *gated prior* for decisions

Attach a 2-layer ReLU MLP over $\phi(A)$ and $\psi(C)$ to produce a gating scalar $g(A, C) \in [0, 1]$. During decoding, for each candidate literal ℓ we compute

$$\mathrm{logit}'(\ell) \;=\; \mathrm{logit}_{\mathrm{LM}}(\ell) + \lambda \cdot \max_{C \ni \ell} g(A, C),$$

with a small λ and strict legality checks (unit consistency, no duplicate assignments). This is our analogue of Zhang et al.'s scorer-guided control, but we *mix* with the Transformer policy and enforce the SAT verifier at every step (no correctness risk). Their own results indicate representation alone can yield substantial node reductions; we aim for fewer backtracks/shorter traces at fixed exactness.

4.2 (II) CNF→Horn pretraining and renamable Horn

A warm start: generate Horn or *renamable* Horn instances from our CNF pool, compute the least model via linear-time closure, and pretrain $e(\cdot)$ using true unification kernels before switching to K_{UP}. This copies Zhang et al.'s core bias into our encoder with provable structure on day one. (Horn closure and renamable Horn are standard.)

4.3 (III) Speculative - RetoMaton + CNF embeddings

Our earlier note proposes Local RetoMaton over SAT traces to reduce gradient steps and improve provenance. [Alon et al., 2022] showed that automaton memory reduces perplexity or nearest-neighbor cost; and at NeSy 2025 they further demonstrate accuracy gains on GSM8K/MMLU/TriviaQA by moving from global→domain→local retomata and gives formal analysis of clustering effects. Combine that with ϕ, ψ: we cluster states in the automaton using $(\phi(A), \max_C g(A, C))$ so that retrieval is *logic-aware*. This could reduce retrieval noise and align automaton transitions with SAT semantics.

5 A compact formal lens

5.1 A margin view of decision quality

Assume a distribution $\mathcal{D}$ over (F, A) and a finite candidate set $\mathcal{C}(F)$ of active clauses. Let the gold decision literal be $\ell^\star(F, A)$. Define the binary label

$$y(A, C) := \mathbb{1}\big[\exists \text{ unit event with forced literal } \ell^\star(F, A) \text{ in } C\big].$$

Consider a triplet loss over (A, C^+, C^-):

$$\mathcal{L}_{\text{trip}} = \big[\alpha + s(A, C^-) - s(A, C^+)\big]_+,$$

with $s(A, C) = \langle \phi(A), \psi(C) \rangle$. If the embedding family achieves margin γ on $\mathcal{D}$ under our difficulty curriculum (i.e., $\mathbb{P}[s(A, C^+) - s(A, C^-) \geq \gamma] \geq 1 - \delta$), then a simple softmax gate $\pi(C \mid A) \propto e^{s(A,C)}$ has top-1 regret bounded by $\tilde{O}(\delta/\gamma)$ relative to the oracle that picks clauses containing $\ell^\star$. This is a standard margin-regret argument; the point is that *representational* progress directly caps decision regret before the verifier. (Zhang et al. implicitly exploit the same margin geometry for unification.)

[2] See the dot-product tests for satisfaction and conflicts (Eqs. (1)–(2) in my prior note).

5.2 Resolution adjacency as a geometric prior

For any active C, let $\mathcal{R}(C)$ be clauses reachable by a single resolution step on variables appearing in C. Encourage $\psi(C)$ to be close to $\psi(C')$ for $C' \in \mathcal{R}(C)$ *only when* $\ell^\star$ survives; otherwise push apart. This connects CNF geometry to proof-theoretic neighborhoods (a SAT analogue of "unifiers must co-locate") and should stabilize decisions under small clause perturbations.

6 Proposed Training plan

TP0: infrastructure. From compiled/learned traces, dump (F, A_t) at each step, the active clause set, the unit frontier, and the gold $\ell_t^\star$. (We already log this.) Build CNF triplets per Sec. 3.3 with $2\times$ multithreading.

TP1: representation pretraining.

1. **Horn warm-start.** Pretrain $e(\cdot)$ on Horn/renamable Horn with pure unification triplets; then switch to K_{UP}-triplets.
2. **Difficulty curriculum.** Start with easy/medium; introduce hard at epoch 5 and enable 50% hard-mining refresh every 10 epochs (Zhang et al.'s schedule). :contentReference[oaicite:16]index=16
3. **Adaptive margins.** Small margins for easy, larger for hard triplets (mirrors their conclusion section).

TP2: on-policy fine-tune with a gate. Freeze the verifier and LM. Train the gating MLP on (A, C) with targets $y(A, C)$ and a ranking loss toward clauses/literals used in traces. Anneal $\lambda : 0.3 \to 0.1$ to avoid over-reliance.

7 Empirical readouts (accept/reject gates)

1. **Trace length at fixed exactness.** Mean CoT steps and backtracks on $p \in \{10, 12, 14, 16, 18, 20\}$ at $\alpha \in [4.1, 4.4]$. Accept if $\geq$10–15% reduction vs. baseline at same 100% exactness.
2. **Steps-to-stability.** SGD steps to 99% *trace validity* on $p \leq 12$; accept if $\geq$30% fewer steps.
3. **Low-α stress.** Evaluate at $\alpha \in \{3.2, 3.5, 3.8\}$; the geometry should particularly help under-constrained regimes (harder to learn from examples).
4. **Ablations.** (a) remove hard-mining; (b) remove resolution adjacency loss; (c) Horn warm-start off; (d) gate off (representations only).

8 Theoretical ramifications

8.1 From Horn closure to SAT guidance

Horn closure computes a least Herbrand model in linear time; unification-consistent embeddings approximate the discrete closure operator by a smooth metric. *Transferring this idea*, K_{UP}-consistent embeddings approximate the unit-closure front in CNF. This does not change worst-case complexity (DPLL remains exponential) but can provably shrink expected search trees under margin conditions (Sec. 5.1) and stabilize learning dynamics by making the "right next literal" linearly separable near the decision boundary.

8.2 A note on renamable Horn

A measurable fraction of industrial instances have large renamable-Horn backbones. Even when a formula is not renamable-Horn globally, blocks often are. A Horn-initialized $e(\cdot)$ is therefore a *structural prior* that we can validate cheaply: run the renamable-Horn test per clause family as a gate for pretraining exposure. (Background on Horn/renamable Horn is standard.)

9 Bottom line (my read)

Zhang et al. convincingly demonstrate that *unification-aware embeddings* pay downstream dividends in logic search, even before fancy control policies enter. Their three practical tricks (duplicate-term anchors, difficulty-aware triplets, hard mining) map naturally to *unit-propagation-aware geometry* in CNF and give us a low-lift, verifier-safe way to (a) shorten traces, (b) reduce gradient steps, and (c) make our decisions more robust in under-constrained regimes. I recommend we implement TP1–TP3 as outlined; cost is modest, and the signal should show up quickly on our standard dashboards.

Most notably, in the room, the speaker framed the work as injecting neural representations into a symbolic pipeline (embedding→scoring→search) with triplet loss and hard-example cycles.

Possible Action items.

1. Implement CNF triplet synthesis with easy/medium/hard and hard mining; run Horn warm-start.
2. Wire a small gating head on top of ϕ, ψ; enforce verifier; log per-step *why* a literal was boosted.
3. Run the ablations and the low-α stress test; adopt if we meet gates.

Acknowledgments

Written in my capacity as coauthor on Pan et al. [2025]; any errors are mine.

References

L. Pan, V. Ganesh, J. Abernethy, C. Esposo, and W. Lee. Can Transformers Reason Logically? A Study in SAT Solving. *arXiv:2410.07432*, 2025. *Local file used for exact statements.*

Y. Zhang, Y. White, D. Clark, J. Sanchez, J. Lipsey, A. Hirst, and J. Heflin. High Quality Embeddings for Horn Logic Reasoning. In *Proc. NeSy 2025*, PMLR 284:1–14, 2025.

T. Rocktäschel and S. Riedel. End-to-end Differentiable Proving. In *NeurIPS 2017*. *Background on differentiable unification*.

U. Alon, F. Xu, J. He, S. Sengupta, D. Roth, and G. Neubig. Neuro-Symbolic Language Modeling with Automaton-augmented Retrieval. In *ICML 2022*, PMLR 162:468–485. *RetoMaton*.

W. F. Dowling and J. H. Gallier. Linear-Time Algorithms for Testing the Satisfiability of Propositional Horn Formulae. *Journal of Logic Programming*, 1(4):267–284, 1984. *Horn closure background.*

J. Jakubùv and J. Urban. ENIGMA: Efficient Learning-based Inference Guiding Machine. In *Intelligent Computer Mathematics*, 2017. *ATP guidance line*.