Research in Neurosymbolic Reasoning

From LLM Capabilities to Real-World Legal Architectures

Otto Mättas
Tallinn University of Technology
13 April

Lecture Roadmap (90 mins)

1. The Boundary: When is AI actually 'Neurosymbolic'?
2. The Limits: Horn vs. Non-Horn SAT in LLMs
3. The Economics: Can 'Thinking' replace logic?
4. Applied WIP: The Legal Neurosymbolic Architecture
5. Open Challenges & Brainstorming

1. The Neurosymbolic Landscape

Goal: Integrate statistical learning with explicit knowledge & reasoning.

Four Core Themes (Mättas, Järv, & Tammet, 2025):

1. Performance (Efficiency & capability)
2. Understandability (Inspectable artifacts)
3. Reliability (Constraints & formal guarantees)
4. Ethics (Auditable alignment & governance)

1. The Neurosymbolic Landscape

Goal: Integrate statistical learning with explicit knowledge & reasoning.

Four Core Themes (Mättas, Järv, & Tammet, 2025):

1. Performance (Efficiency & capability)
2. Understandability (Inspectable artifacts)
3. Reliability (Constraints & formal guarantees)
4. Ethics (Auditable alignment & governance)

🗣️ **DISCUSSION: Is the AI industry's push for 'Performance' (massive scaling) fundamentally at odds with 'Understandability'?**

The Strict Boundary Rule

Tool Grounding (like RAG or API calls) ≠ True Neurosymbolic AI.

- RAG reduces error rates, but does NOT certify correctness.
- To be truly **Neurosymbolic**, the system must:
  a. Use explicit symbolic representations.
  b. Generate typed, executable artifacts.
  c. Enforce execution/checking by a solver.

The Strict Boundary Rule

Tool Grounding (like RAG or API calls) ≠ True Neurosymbolic AI.

- RAG reduces error rates, but does NOT certify correctness.
- To be truly **Neurosymbolic**, the system must:
  a. Use explicit symbolic representations.
  b. Generate typed, executable artifacts.
  c. Enforce execution/checking by a solver.

🗣️ **DISCUSSION: If RAG isn't reasoning, how does a purely neural RAG system handle retrieving two conflicting legal rules?**

2. Evaluating LLM Reasoning

Paper: Horn vs. Non-Horn SAT Decision in LLMs

We evaluated LLMs using Boolean Satisfiability (SAT) in Conjunctive Normal Form (CNF).

Why SAT?

- It strips away all linguistic ambiguity.
- Objective ground truth (Right vs. Wrong).
- We can perfectly control the computational hardness.

2. Evaluating LLM Reasoning

Paper: Horn vs. Non-Horn SAT Decision in LLMs

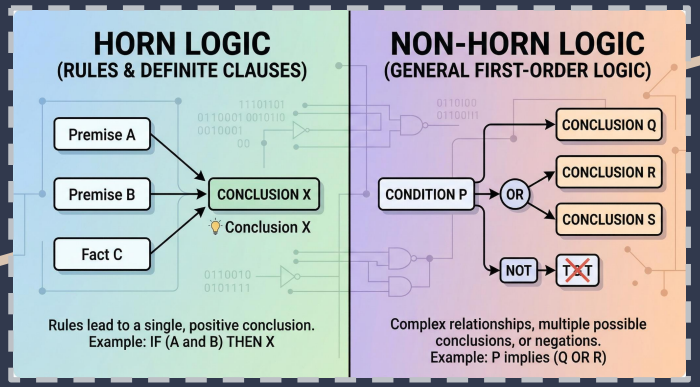
We evaluated LLMs using Boolean Satisfiability (SAT) in Conjunctive Normal Form (CNF).

Why SAT?

- It strips away all linguistic ambiguity.
- Objective ground truth (Right vs. Wrong).
- We can perfectly control the computational hardness.

🗣️ **DISCUSSION: Why test LLMs on abstract SAT formulas instead of natural language logic puzzles?**

The Structural Divide



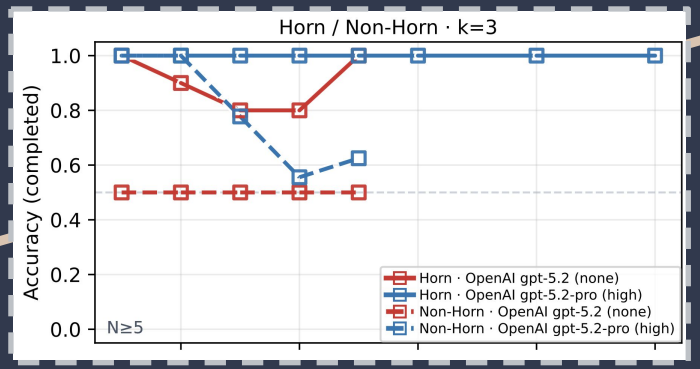
HORN CLAUSES

- Syntactically restricted (max 1 positive literal).
- Tractable: Linear-time satisfiability.
- Maps directly to 'If-Then' rules.

NON-HORN CLAUSES

- General CNF, no restrictions.
- Hard: NP-Complete.
- Requires complex backtracking (DPLL).

Result 1: The Tractability Divide



Horn can be '**solved**', but Non-Horn remains a massive bottleneck.

Horn Subset:

- gpt-5.2-pro (high) reaches 1.00 accuracy on Compact CNF.

Non-Horn Subset:

- Same model achieves only 0.74 accuracy.

Takeaway: Strong performance on restricted logic does NOT transfer to general combinatorial logic.

Result 2: The Satisfiable Asymmetry

The core failure mode of LLMs in formal logic:

LLMs have a massive bias towards guessing '**Satisfiable**'.

Example (Non-Horn, k=3, gpt-5.2 no-think):

- Satisfiable Accuracy: 1.00
- Unsatisfiable Accuracy: 0.00

Result 2: The Satisfiable Asymmetry

The core failure mode of LLMs in formal logic:

LLMs have a massive bias towards guessing '**Satisfiable**'.

Example (Non-Horn, k=3, gpt-5.2 no-think):

- Satisfiable Accuracy: 1.00
- Unsatisfiable Accuracy: 0.00

🗣️ **DISCUSSION: Based on the transformer architecture, why is proving 'Unsatisfiable' so much harder than guessing 'Satisfiable'?**

Result 3: Prompting Brittleness

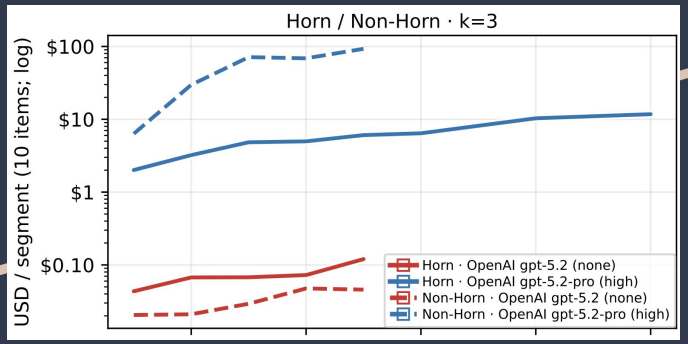
Interventions fix one segment, but break another.

Algorithmic Prompting (forcing line citations):

- Increased Unsatisfiable accuracy from 0.45 to 0.92.
- BUT crashed Satisfiable accuracy from 0.99 down to 0.21!

You cannot rely on prompt engineering for consistent logical performance.

Result 4: The Price of 'Thinking'

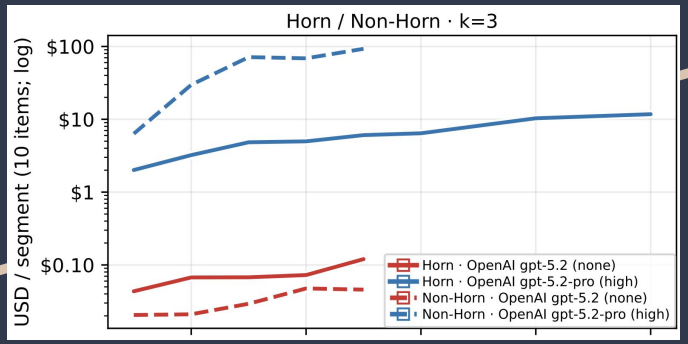


Test-time scaling yields gains, but at a steep cost increase.

Performance on Non-Horn Unsatisfiable (k=3):

- gpt-5.2 (none): $0.0036 per query (0% accuracy)
- gpt-5.2-pro (high): $7.52 per query (59% accuracy)

Result 4: The Price of 'Thinking'



Test-time scaling yields gains, but at a steep cost increase.

Performance on Non-Horn Unsatisfiable (k=3):

- gpt-5.2 (none): $0.0036 per query (0% accuracy)
- gpt-5.2-pro (high): $7.52 per query (59% accuracy)

🗣️ **DISCUSSION: At $7.52 per query for 59% accuracy, is inference scaling a dead end for strict logic?**

3. Applied Work: Formalising Law

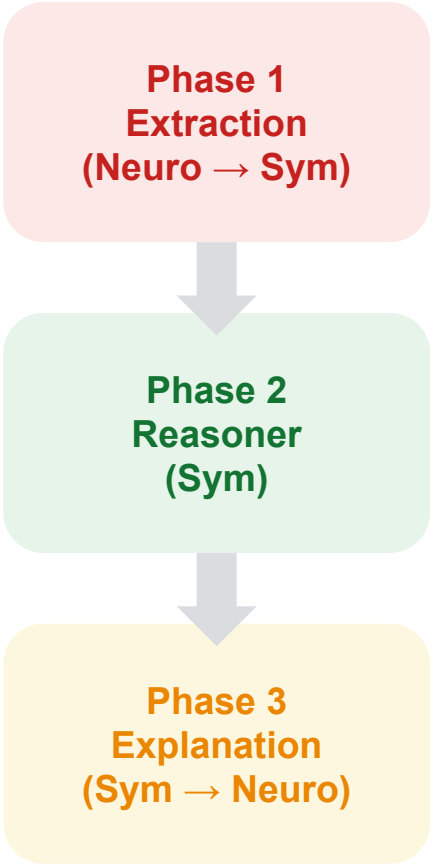
Transitioning from pure theory to my current, unpublished research.

The Challenge in Legal AI:

1. Laws are highly structured but linguistically complex.
2. User utterances are messy, incomplete, and contextual.

We need a system that handles messy language **BUT** provides deterministic, mathematically verifiable decisions.

The Legal Neurosymbolic Architecture



Phase 1: Meaning Extraction

NEURO → SYM

- Input: Messy user text + Legal text.
- Task: LLM acts strictly as a semantic translator.
- Output: Explicit, typed facts and rules.

We target representations that mimic Horn clauses where possible for reliability.

Phase 1: Example

The Law

"A resident who leaves the jurisdiction for over 6 months without prior authorization forfeits their status, unless traveling for a verified medical emergency."

The Case

John was abroad for 8 months for heart surgery without asking first.

Phase 1: Example

The Case

John was abroad for 8 months for heart surgery without asking first.

Rule

$$\text{ForfeitStatus}(x) \leftarrow \text{Resident}(x) \land \text{MonthsAbroad}(x, m) \land (m > 6) \land \neg\text{Authorized}(x) \land \neg\text{MedicalEmergency}(x)$$

Extracted Facts

- Resident(John)
- MonthsAbroad(John, 8)
- ¬Authorized(John)
- MedicalEmergency(John)

Phase 1: Meaning Extraction

NEURO → SYM

- Input: Messy user text + Legal text.
- Task: LLM acts strictly as a semantic translator.
- Output: Explicit, typed facts and rules.

We target representations that mimic Horn clauses where possible for reliability.

🗣️ **DISCUSSION: Law uses intentional vagueness (e.g., 'reasonable doubt'). How can we map human ambiguity to strict boolean logic?**

Phase 2: The Symbolic Reasoner

THE LOGIC ENGINE

- Input: Formal knowledge from Phase 1.
- Task: Deterministic execution.
- Output: A decision + A strict symbolic proof tree.

Advantages:

- Zero Hallucinations.
- Bypasses the 'Satisfiable Bias'.
- 100% accurate to the premises.

Phase 2: Example

Goal

?ForfeitStatus(John)

MonthsAbroad(John, 8) > 6 ➔ **TRUE**

¬Authorized(John) ➔ **TRUE**

MedicalEmergency(John) ➔ **TRUE** *(Breaks the negation requirement)*

Solver Output

FALSE (John does not forfeit his status).

Phase 2: The Symbolic Reasoner

THE LOGIC ENGINE

- Input: Formal knowledge from Phase 1.
- Task: Deterministic execution.
- Output: A decision + A strict symbolic proof tree.

Advantages:

- Zero Hallucinations.
- Bypasses the 'Satisfiable Bias'.
- 100% accurate to the premises.

🗣️ **DISCUSSION: Real-world laws contradict (e.g., new laws override old ones). Has anyone worked with non-monotonic logic?**

Phase 3:
Trace Translation

SYM → NEURO

If we hand a judge a mathematical proof tree, they will reject it.

- Task: LLM acts as an explainer.
- Constraint: It is strictly prompted to follow ONLY the nodes of the proof tree.

Phase 3: Example

Solver Output

FALSE (John does not forfeit his status).

The Explanation

John does not forfeit his status due to having an emergency (heart surgery), even though he was abroad for 8 months without asking first.

Phase 3: Trace Translation

SYM → NEURO

If we hand a judge a mathematical proof tree, they will reject it.

- Task: LLM acts as an explainer.
- Constraint: It is strictly prompted to follow ONLY the nodes of the proof tree.

🗣️ **DISCUSSION: Plausibility vs Faithfulness: How do we programmatically evaluate that the LLM didn't hallucinate extra steps in its explanation?**

Why this matters: Ethics & Accountability

This architecture enables true auditable governance:

1. Provenance: We know exactly which clause triggered the decision.
2. Reversibility: We can isolate errors (Was it bad LLM extraction? Or a missing rule?).
3. Redress: If parliament changes the law, we update one symbolic rule. We don't have to retrain a billion-parameter model.

Why this matters: Ethics & Accountability

This architecture enables true auditable governance:

1. Provenance: We know exactly which clause triggered the decision.
2. Reversibility: We can isolate errors (Was it bad LLM extraction? Or a missing rule?).
3. Redress: If parliament changes the law, we update one symbolic rule. We don't have to retrain a billion-parameter model.

🗣️ **DISCUSSION: If the LLM extracts a premise incorrectly, and the perfectly logical solver wrongly convicts someone, who is ethically liable?**

Conclusion & Future Trajectories

- LLMs are remarkable translators, but unreliable and expensive reasoners.
- Neurosymbolic architectures guarantee performance, reliability, and accountability.

Open Challenges for your theses/research:

1. Semantic Fidelity: Capturing legal nuance without losing strict logic.
2. Non-Monotonicity: Handling inherently contradictory legal texts.
3. Meta-Cognition: Letting the system know **when** to invoke a symbolic solver.

Q&A

Scan for contact

Let's continue the discussion.

Papers referenced:

- A survey of neurosymbolic artificial intelligence: foundations, advances, and future trajectories (Mättas, Järv, Tammet, 2026)
- Horn vs. Non-Horn SAT Decision in Large Language Models (Mättas, Järv, Tammet, 2026)
- Current Work: Legal Decisions & Explanations

Contact: otto.mattas@taltech.ee