# Improving Rule-based Reasoning in LLMs using Neurosymbolic Representations

Varun Dhanraj, Chris Eliasmith {vdhanraj, celiasmith}@uwaterloo.ca

Centre for Theoretical Neuroscience, University of Waterloo

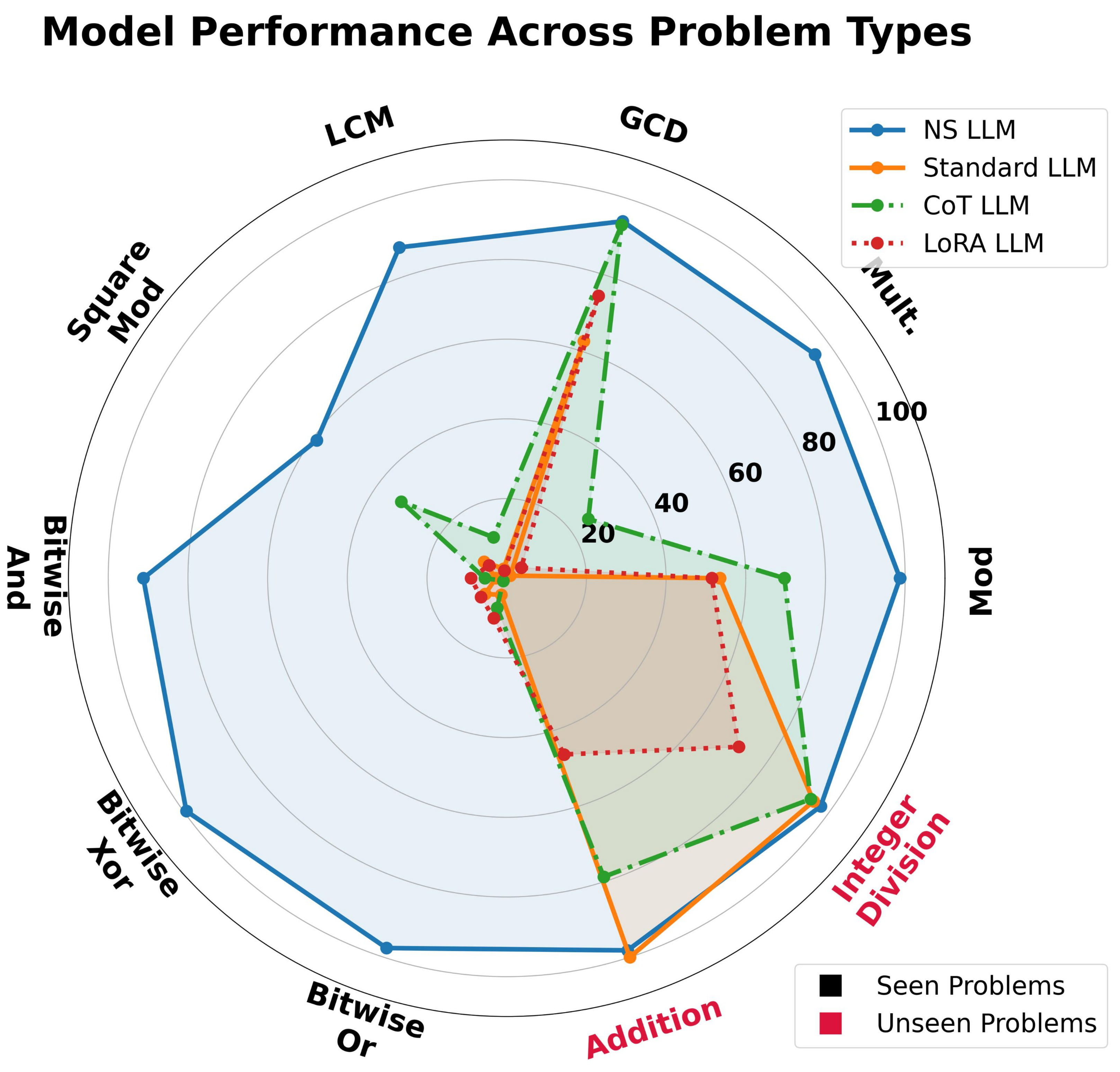
University of Waterloo

## INTRODUCTION

- Large Language Models (LLMs) perform well on pattern-based tasks, but often fail on problems that require strict rule-following, such as mathematical reasoning.
- These models can generate inconsistent or incorrect outputs due to unstructured internal representations.
- Symbolic reasoning systems provide precision and reliability but lack flexibility and scalability in real-world tasks.
- This work introduces a neurosymbolic method that combines the strengths of both paradigms.
- The proposed approach achieves:
  - **15.4× more problems solved** and **88.6% lower loss** on mathematical reasoning tasks compared to Chain-of-Thought and LoRA baselines.
  - No performance loss on general or out-of-distribution problems.

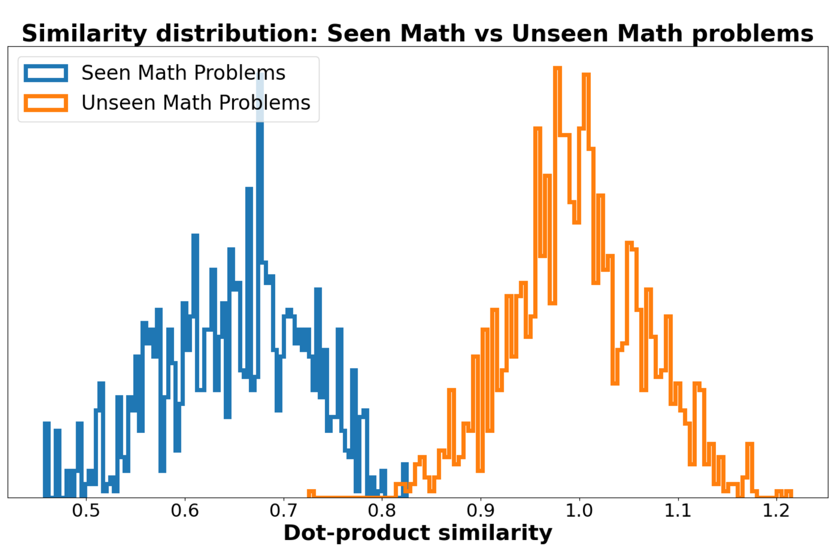
## MAIN RESULTS

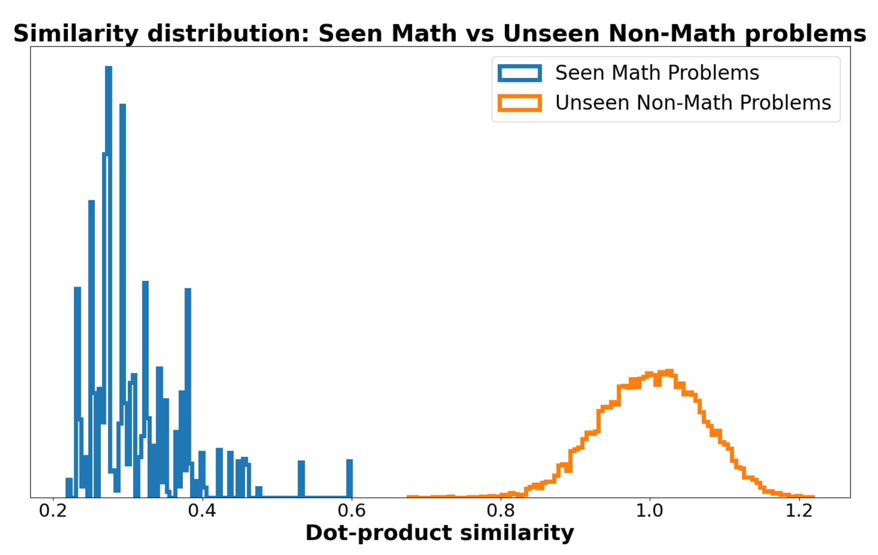
- On arithmetic tasks like Mod, Mult., LCM, and Bitwise f, the Neurosymbolic (NS) LLM outperforms baselines by **large margins**.
- On **unseen problems** (Addition, Integer Division), the NS LLM performs **on par with the standard LLM** since no intervention is triggered when the problem type is unseen during training.




## SELECTIVE ACTIVATION

- The neurosymbolic intervention only occurs when the model is confident the prompt is similar to the problems it has seen during training.
- Similarity is calculated by querying the encoded representation with the problem type. The average similarity of queries is:
  - ≈ 1 for math problems seen during training.
  - ≈ 0.65 for math problems not seen during training.
  - ≈ 0.35 for non-math problems not seen during training.




## CREATING NEUROSYMBOLIC REPRESENTATIONS

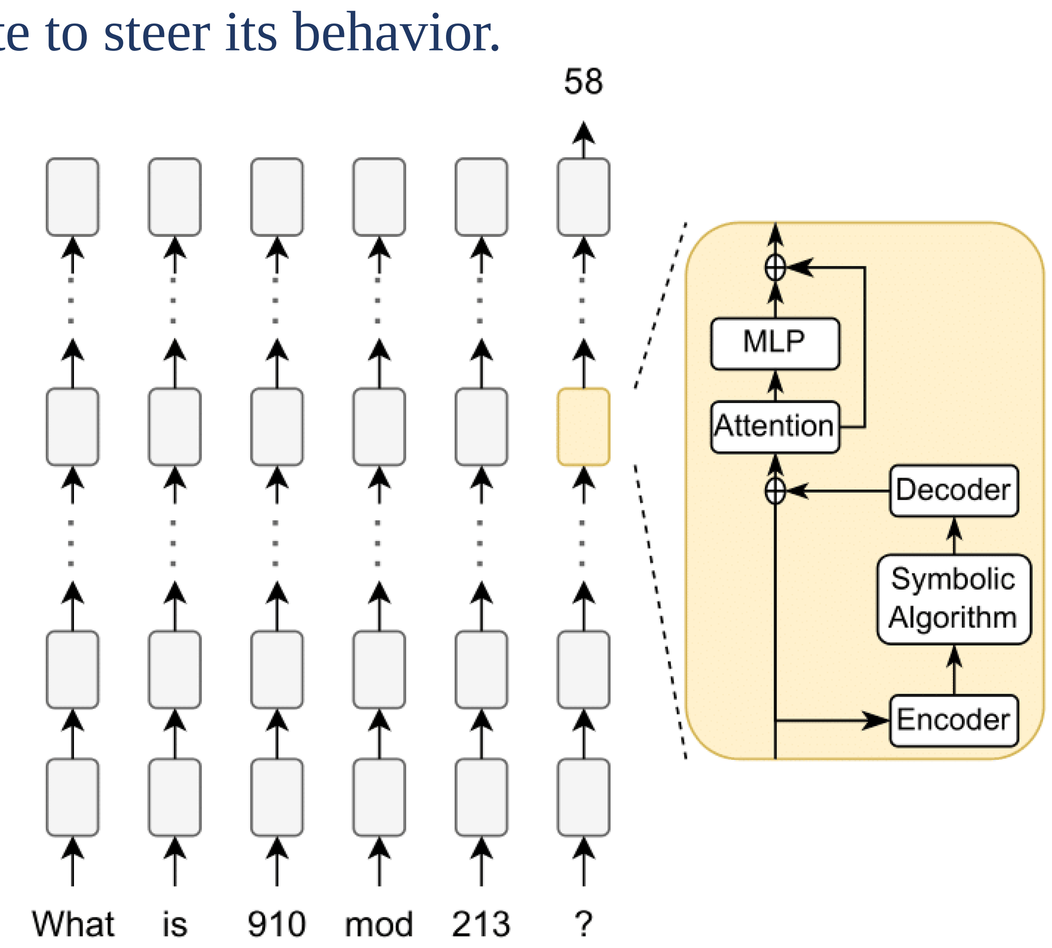
- LLM hidden states are converted into Vector Symbolic Algebra (VSA) representations, which allow for monosemantic concepts to be composed into one neurosymbolic vector, e.g.:

What is 910 mod 213?

$$n1 \circledast (hundreds \circledast 9 + tens \circledast 1 + ones \circledast 0) +$$
$$n2 \circledast (hundreds \circledast 2 + tens \circledast 1 + ones \circledast 3) +$$
$$problem\ type \circledast modulo$$

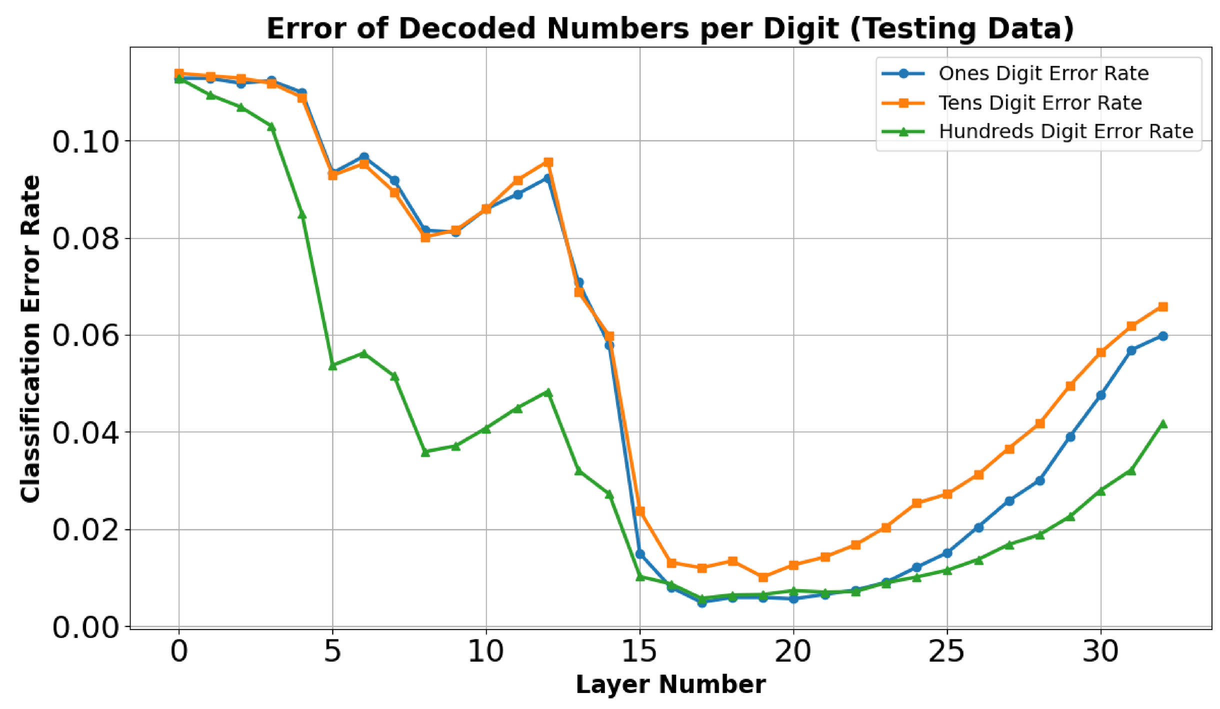
## METHODS

- Prompt the LLM with different math problems.
- Train encoder to map LLM hidden states into structured neurosymbolic vectors.
- Apply symbolic algorithms to compute the solution outside the LLM.
- Train and fine-tune decoder to map the result back into the LLM's hidden state to steer its behavior.




## ENCODING ACCURACY

- The encoder generates the most accurate encoding in the middle layers of the model, achieving under 2% classification error at layer 17.




## CONCLUSION

- Our **neurosymbolic method** combines the strengths of LLMs and symbolic reasoning to perform **structured, rule-based reasoning**.
- Our approach greatly increases the performance of LLMs on mathematical reasoning tasks (**15.4 more problems solved** and **88.6% lower loss**), while not affecting their behavior on other tasks.
- Unlike traditional black-box LLMs, our method offers interpretability: neurosymbolic vectors expose LLM intermediate structure.

GitHub Repository