Teaching Transformers to Solve Combinatorial Problems through Efficient Trial & Error

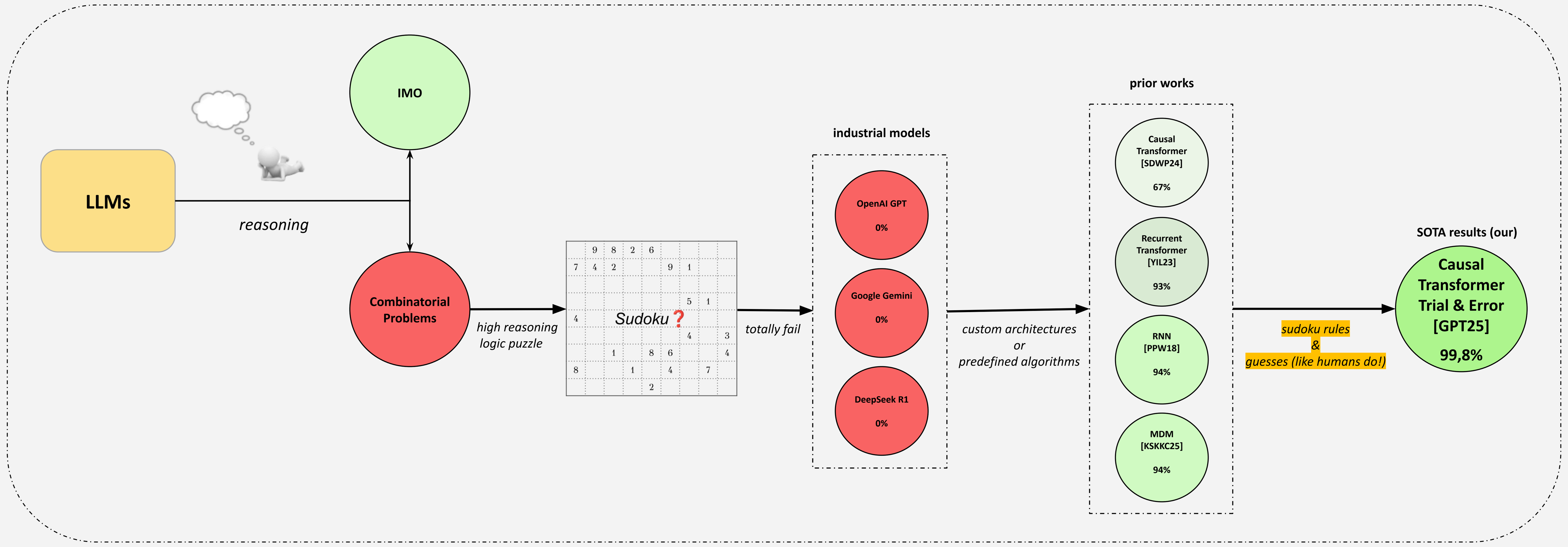
Panagiotis Giannoulis[1] *Yorgos Pantis*[2,3] *Christos Tzamos*[2,3]

[1]*National Technical University of Athens, Greece* [2]*National and Kapodistrian University of Athens, Greece* [3]*Archimedes, Athena Research Center, Greece*

Authors listed alphabetically; equal contribution

Presented at NeurIPS 2025, San Diego, USA — December 2025

Motivation



Key Contributions

- A generalized framework for NP-class problems
- No custom architecture: valina-decoder only GPT-2 with 42M parameters
- Training transcripts solely based on Sudoku rules (DFS method) and guesses
  - An approach that can solve Sudoku which we do not know yet deterministic strategies to solve them
  - Close related approach to human nature
- SOTA results; accuracy 99.8% on randomly generated Sudoku
- Minimization of guesses via a novel loss inspired by *Min-Sum Set Cover* problem
- SudokuPy: a Python library for generating random Sudoku puzzles

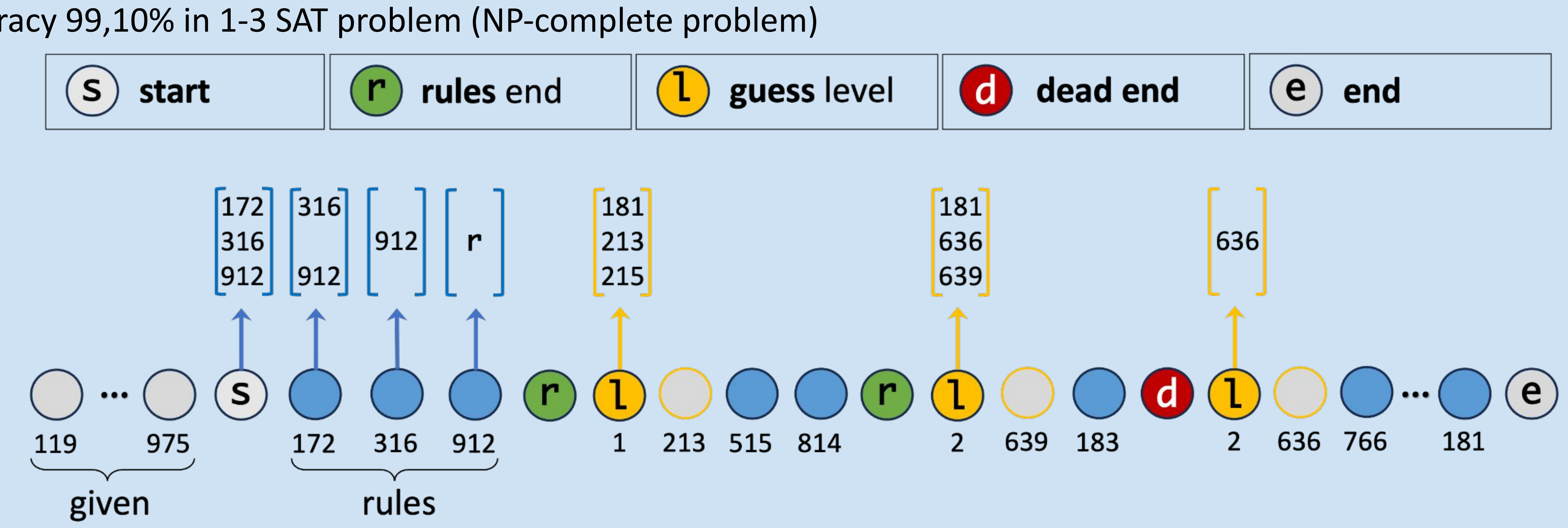
Imitation Learning

Sudoku rules + informed multiple guesses

- Encoding: Each move encoded as a 3-digit number rcv (111 → 999); row_column_value
- Multiple Targets: Combinatorial puzzles allow multiple valid next moves
  - Instead of a single deterministic label, we support multiple next-token predictions

    Cross Entropy loss: $-\log p_i$ → Multiple target loss: $-\sum \log p_i$ over all valid next tokens
- Results: Accuracy 98.9%; SOTA compared to other prior works

  Accuracy 99,10% in 1-3 SAT problem (NP-complete problem)



TL;DR

We show that Transformers can reason on NP-combinatorial problems, achieving 99.8% SOTA accuracy in Sudoku using only its rules and minimal guesses, like humans do!

QR Code

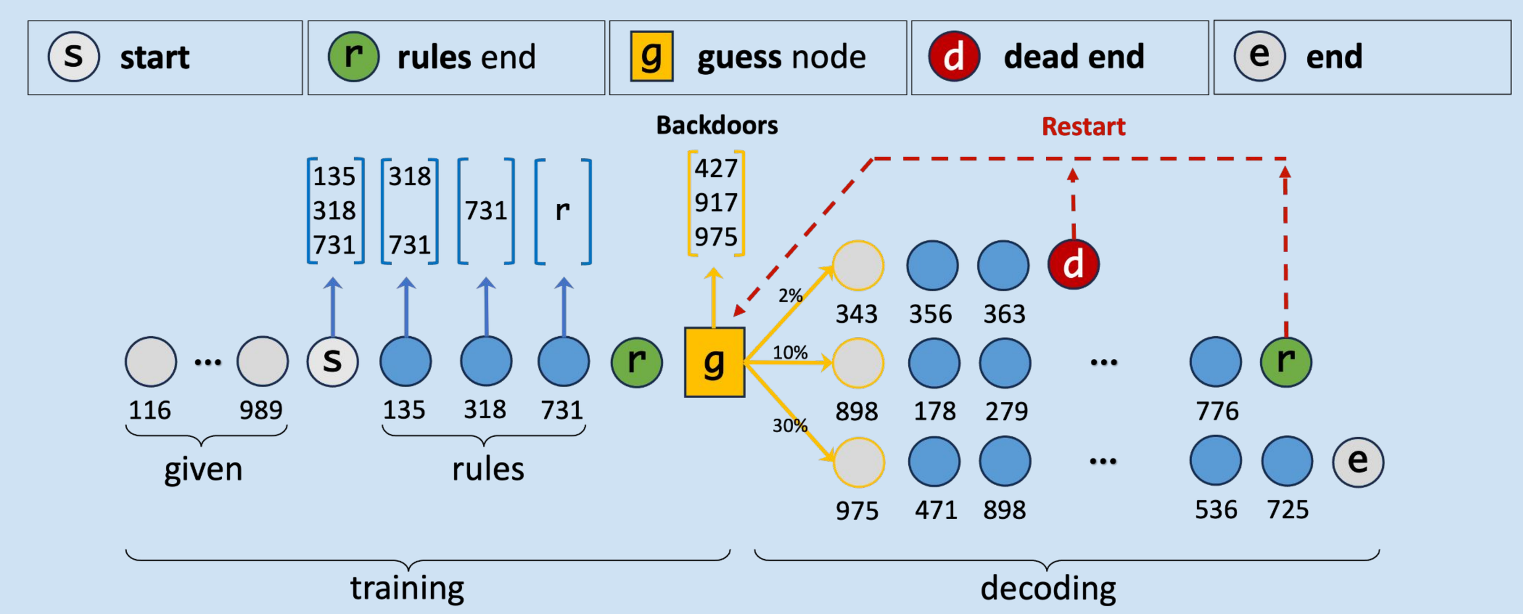
paper

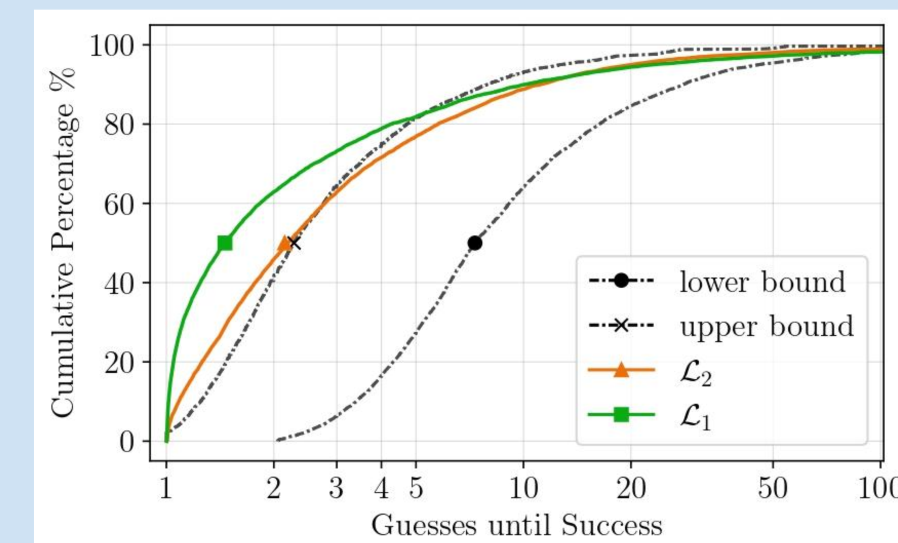
Beyond Imitation Learning

Sudoku rules + a single guess + guess loss inspired by Min-Sum Set Cover problem

- Encoding & Multiple Targets: Same as Imitation Learning
- Guesses: A single guess (backdoor; once identified, applying simple rules leads to the full solution)
- Loss Function: Instead of a single deterministic label, we support multiple next-token predictions

  Multiple target loss: $-\log p_i$ & Guesses loss: $(\sum p_i)^{-1}$ over all valid guesses (see mathematical insights) (1)
- Results: Accuracy 99.8%; new SOTA, outperforming even our previous best results





Insights: This approach shows empirically that 99.8% of Sudokus can be solved by using only one guess (backdoor guess)!

Mathematical Insights *(Min-Sum Set Cover problem)*

- Assumptions: depth-1 of search and non adaptive policy
- Challenge:
  - You face n possible choices, but only a hidden subset S is valid
  - Subset S is drawn from a known distribution D
  - Each test costs 1 time unit and once it is made you only learn if it is valid
- Goal: Find a policy π that minimizes the expected time to discover a valid choice

Theorem. *For any distribution $\mathcal{D}$ over sets $S \subseteq [n]$, it holds that for any permutation τ:*

$$\min_{\pi \in \Delta(n)} \mathbb{E}_{S\sim\mathcal{D}}\left[\frac{1}{\sum_{i\in S}\pi_i}\right] \le H_n \cdot \mathbb{E}_{S\sim\mathcal{D}}\left[\arg\min_{i=1}^{n}\{\tau_i \in S\}\right]$$

where $H_n = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} = \Theta(\log n)$ is the n-th harmonic number.

Remark: Loss function (1) yields solutions with a bounded approximation to the optimal policy, whereas treating the problem as a multi-class classification task (e.g., weighted Cross-Entropy Loss) leads to much worse approximations.

References

- R. Palm, U. Paquet, O. Winther. Recurrent relational networks. NeurIPS'18.
- Z. Yang, A. Ishay, J. Lee. Learning to solve constraint satisfaction problems with recurrent transformer. ICLR'23.
- K. Shah, N. Dikkala, X.Wang, R.Panigrahy. Causal language modeling can elicit search and reasoning capabilities on logic puzzles. NeurIPS'24.
- J.Kim, K. Shah, V. Kontonis, S. Kakade, S. Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. ICML'25.