Self-Supervised Transformers as Iterative Solution Improvers for Constraint Satisfaction

Yudong Will Xu , Wenhao Li, Elias B. Khalil, Scott Sanner

UNIVERSITY OF TORONTO

ICML International Conference On Machine Learning

Motivation

Can Transformers solve Constraint Reasoning Problems?

Existing approaches:

- Supervised Learning: labels are often *difficult to obtain* and sometimes *ambiguous*.
- Reinforcement Learning: uses black box reward function which are often complex and *difficult to train*.

Our approach: Self-Supervised Learning.

Background

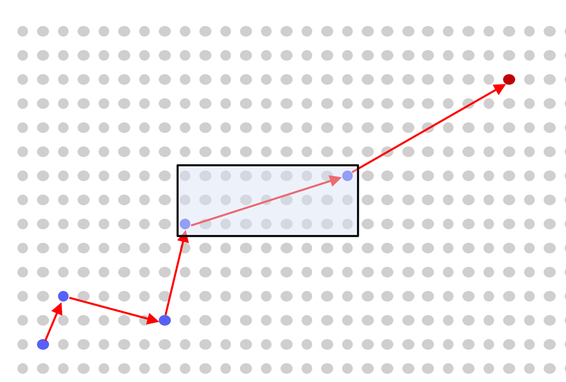
Constraint Satisfaction Problem

Variables $X = \{X_{1,1}, X_{1,2}, \ldots, X_{9,9}\}$

Domains $D = \{D_1, D_2, \ldots, D_{81}\}$
$D_i = \{1,2,3,4,5,6,7,8,9\}$

Constraints $AllDifferent(X_{1,1}, X_{1,2}, \ldots, X_{1,9})$
$AllDifferent(X_{1,1}, X_{2,1}, \ldots, X_{9,1})$
$AllDifferent(X_{1,1}, X_{1,2}, \ldots, X_{3,3})$
...

Stochastic Local Search



ConsFormer

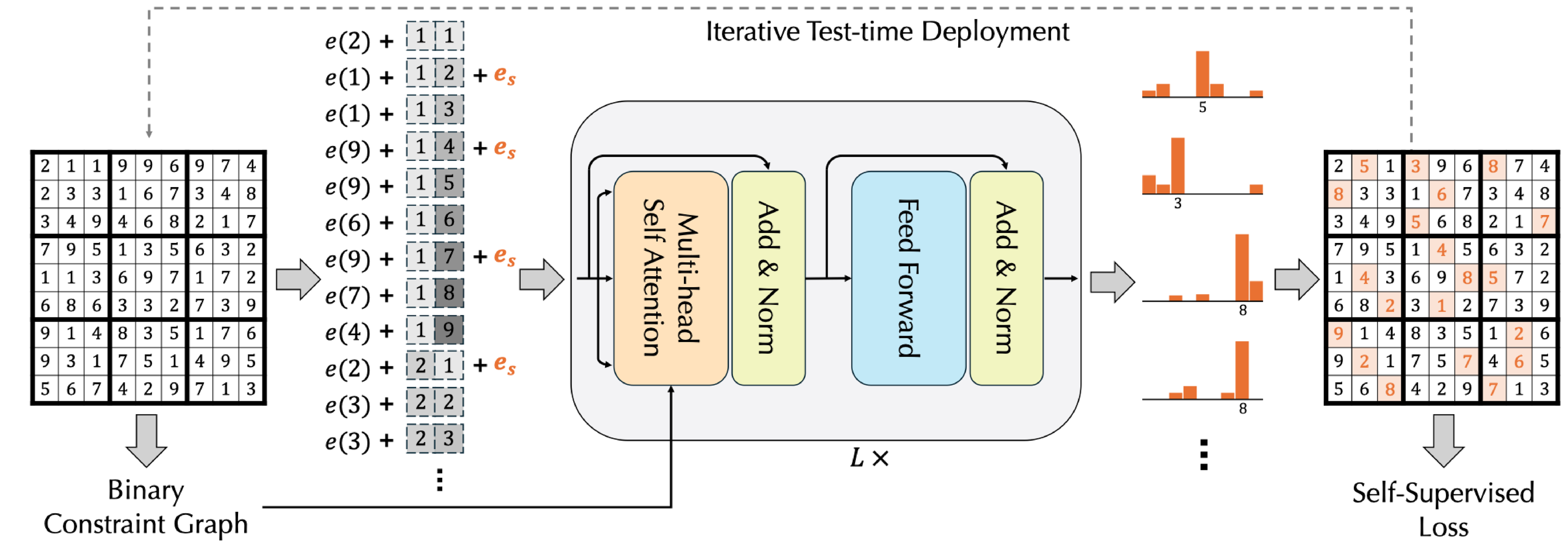
Representation

- Variable indices as absolute positional encodings (APEs)

$$\mathrm{APE}(x_{1,2}) = \mathrm{Concat}(\mathrm{PE}(1), \mathrm{PE}(2))$$

- Binary constraint graph as relative positional encodings (RPEs)

$$\mathrm{RPE}(i,j) = c \cdot \mathbb{I}[(i,j) \notin E]$$



Architecture: Single-step Transformer

- Our model mimics local search by randomly selecting a subset of the variables to update, which are then marked by a special subset embedding e_s.
- Variable assignment with Gumbel Softmax.

$$\hat{\mathbf{y}}_i = \mathrm{GumbelSoftmax}\left(\mathbf{W}_{\mathrm{out}}\mathbf{h}_i^{(L)} + \mathbf{b}_{\mathrm{out}}\right)$$

$$v_i' = \arg\max \hat{\mathbf{y}}_i, \quad \forall i \in S.$$

Training: Self-Supervised Loss

- Define continuous penalty functions p_i for constraint c_i such that

$$p_i(X_i) = 0 \iff c_i(X_i) = \texttt{True},$$

- Loss is then defined by

$$\mathcal{L} = \sum_i \lambda_i f(p_i(X_i)) \quad \forall p_i \in P,$$

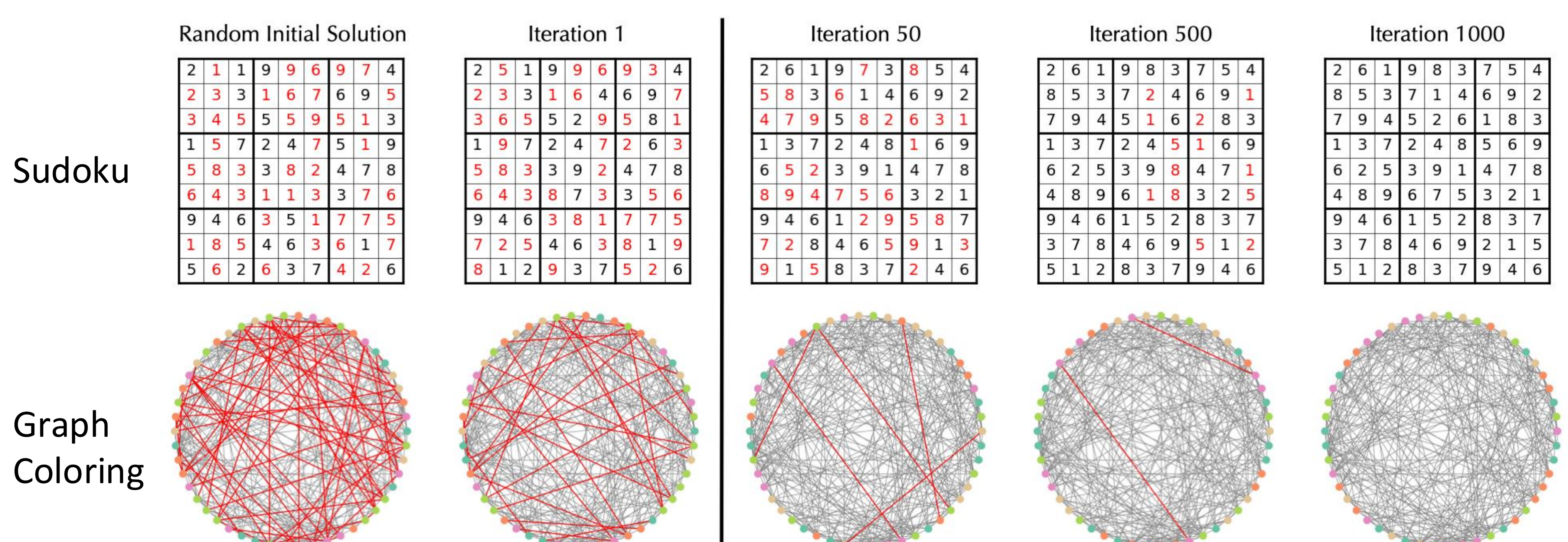
$\textsc{AllDifferent}_{m=n}(x_1, \ldots, x_n)$

$$\implies p = \sum_j \left| 1 - \sum_i x_i^{(j)} \right|.$$

Valid Assignment:
$x_1 = (1,0,0),\ x_2 = (0,1,0),\ x_3 = (0,0,1)$
$\sum_i x_i^{(1)} = 1,\ \sum_i x_i^{(2)} = 1,\ \sum_i x_i^{(3)} = 1$
$p = |1-1| + |1-1| + |1-1| = 0.$

Invalid Assignment:
$x_1 = (0.9,0.1,0),\ x_2 = (0.9,0.1,0),\ x_3 = (0,0,1)$
$\sum_i x_i^{(1)} = 1.8,\ \sum_i x_i^{(2)} = 0.2,\ \sum_i x_i^{(3)} = 1$
$p = |1-1.8| + |1-0.2| + |1-1| = 0.8 + 0.8 + 0 = 1.6.$

Experiments



Training: Single-step

Testing: Iterative Deployment

Method	Test Instances	Harder OOD Instances
Wang et al. (2019)	98.3	3.2
Palm et al. (2018)	99.8	28.6
Yang et al. (2023)	**100**	32.9
Yang et al. (2023) (2k Iters)	97.7	14.0
Du et al. (2024)	99.4	62.1
ConsFormer (2k Iters)	**100**	65.88
ConsFormer (10k Iters)	**100**	**77.74**

Method	Test Instances	Harder OOD Instances
Graph-Coloring-5 ($n = 50 \to n = 100$)		
OR-Tools (10s)	**83.08**	**57.16**
ANYCSP (10s)	79.17	34.83
ConsFormer (10s)	81.00	47.33
Graph-Coloring-10 ($n = 100 \to n = 200$)		
OR-Tools (10s)	52.41	10.25
ANYCSP (10s)	0.00	0.00
ConsFormer (10s)	**52.60**	**11.92**

Key Takeaways

- Iterative single-step reasoning enables out of distribution generalization.

  Harder problem → Run more iterations.

- Self-supervised Learning enables constraints reasoning *without labeled examples.*

Paper

Code

References

Tönshoff, Jan, et al. "One model, any CSP: graph neural networks as fast global search heuristics for constraint satisfaction." *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, 2023.
Yang, Zhun, Adam Ishay, and Joohyung Lee."Learning to Solve Constraint Satisfaction Problems with Recurrent Transformer." *The Eleventh International Conference on Learning Representations*, 2023.
Wang, Po-Wei, et al. "Satnet: Bridging deep learning and logical reasoning using a differentiable satisfiability solver." *International Conference on Machine Learning*. PMLR, 2019.
Palm, Rasmus, Ulrich Paquet, and Ole Winther. "Recurrent relational networks." *Advances in neural information processing systems* 31, 2018.
Du, Yilun, Jiayuan Mao, and Joshua B. Tenenbaum. "Learning Iterative Reasoning through Energy Diffusion." *Forty-first International Conference on Machine Learning,* 2024.