Multi-Network Relational Verification and Certifiable Training

Debangshu Banerjee
Shaurya Gomber
Calvin Xu

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

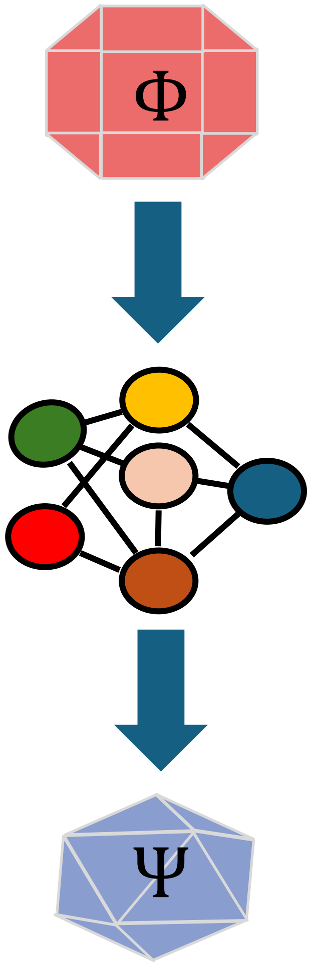
Motivation

Current Gap: Existing verifiers for individual networks fail for multi-network systems

- ❑ Ensemble models for enhanced reliability and performance
- ❑ Conformal prediction systems with a classification network
- ❑ Equivalence checks between a pair of models

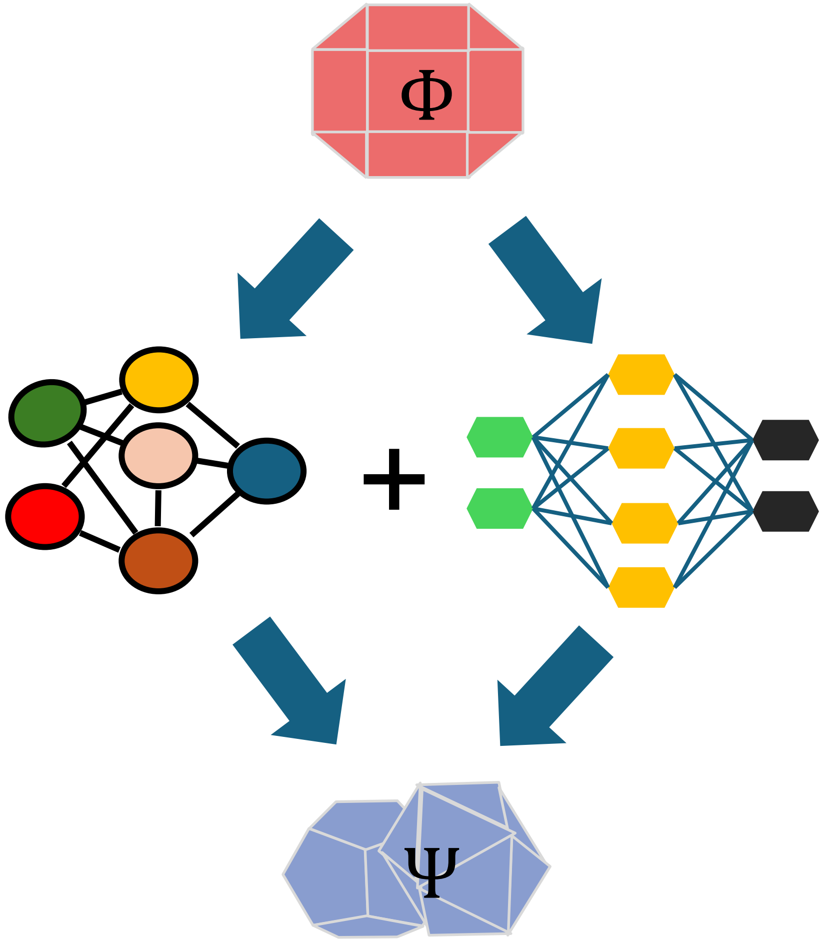
How to design scalable verifiers for multi-network systems ?

Formulation: Classical DNN Property



- ❑Single Network
- ❑Ψ only includes **conjunctions**
- ❑ Local Robustness

Formulation: DNN Hyper-properties



- ❑Multiple Networks
- ❑Ψ can include **disjunctions** (e.g. one network in an ensemble can fail)

Formulation: DNN Hyper-properties

Neural Networks $NN_1, NN_2, \cdots NN_K$ $\quad where\ NN_i: \mathbb{R}^{n_o^i} \rightarrow \mathbb{R}^{n_l^i}$

Input Specification

$$\Phi(x_1, \cdots x_k): \bigwedge_{i=1}^{K} x_i \in X_i \ \wedge\ \Phi_{cross}\,(x_1, \cdots, x_K)$$

Individual Network Constraints

$$X_i = \{\, x \in \mathbb{R}^{n_o^i}\ \ s.t.\ \ ||\, x - x_{i_0}\, ||_\infty \leq \epsilon \,\}$$

Input Relational Constraints

Constraints over inputs to capture their relationships

Output Specification

$$\Psi(NN_1(x_1), \cdots NN_k(x_k)): \bigvee_{i=1}^{M} \psi_i \quad where\ \ \psi_i = \bigwedge_{j=1}^{N} (C_{i,j}^T\, NN_a\,(x_a) \geq b_{i,j})$$

Verification Problem $\qquad \forall x_1 \cdots \forall x_k\,.\ \ \Phi(x_1, \cdots x_k) \Longrightarrow \Psi(NN_1(x_1), \cdots NN_k(x_k))$

Formulation: Examples

Ensemble Local Robustness: If $f\left(NN_1(x), NN_2(x) \cdots, NN_K(x)\right) = \ell$

$$\forall\, x' \cdot ||\, x' - x\, ||_\infty \leq \epsilon \Longrightarrow \left(f\left(NN_1(x'), NN_2(x') \cdots, NN_K(x')\right) = \ell\right)$$

Conformal Prediction Robustness: Classification network NN_{class} and threshold network NN_t

$$\forall\, x' \cdot ||\, x' - x\, ||_\infty \leq \epsilon \Longrightarrow score\left(y, NN_{class}(x')\right) > NN_t(x')$$

Network Equivalence Verification: Networks NN_1 and NN_2 around input x

$$\forall\, x' \cdot ||\, x' - x\, ||_\infty \leq \epsilon \Longrightarrow |\, C^T NN_1(x') - C^T NN_1(x')| \leq \epsilon$$

Methodology: Verification

1. Handle multiple networks from different architecture

- ❑ Convert parametric linear approximations
- ❑ Learn the parameters jointly over multiple networks

2. Handle disjunctive output specification

- ❑ $\forall\, x. \Psi_1(x) \wedge \Psi_2(x) \equiv \forall\, x. \Psi_1(x) \wedge \forall\, x. \Psi_2(x)$

- ❑ $\forall\, x. \Psi_1(x) \vee \Psi_2(x)$ Do not distribute

Methodology: Verification

2. Handle disjunctive output specification

- ❑ $\forall\, x.\, \Psi_1(x) \wedge \Psi_2(x) \equiv \forall\, x.\, \Psi_1(x) \wedge \forall\, x.\, \Psi_2(x)$

- ❑ $\forall\, x.\, \Psi_1(x) \vee \Psi_2(x)$ Do not distribute

- ❑ Use parametric linear approximation $(C_i)s$ to formulate an equivalent Linear Program

$$\min t \;\; s.t.\; C_1^T x \leq t,\; C_2^T x \leq t,\; \Psi_i(x) = C_i^T x \geq 0$$

- ❑ Write the max-min dual formulation

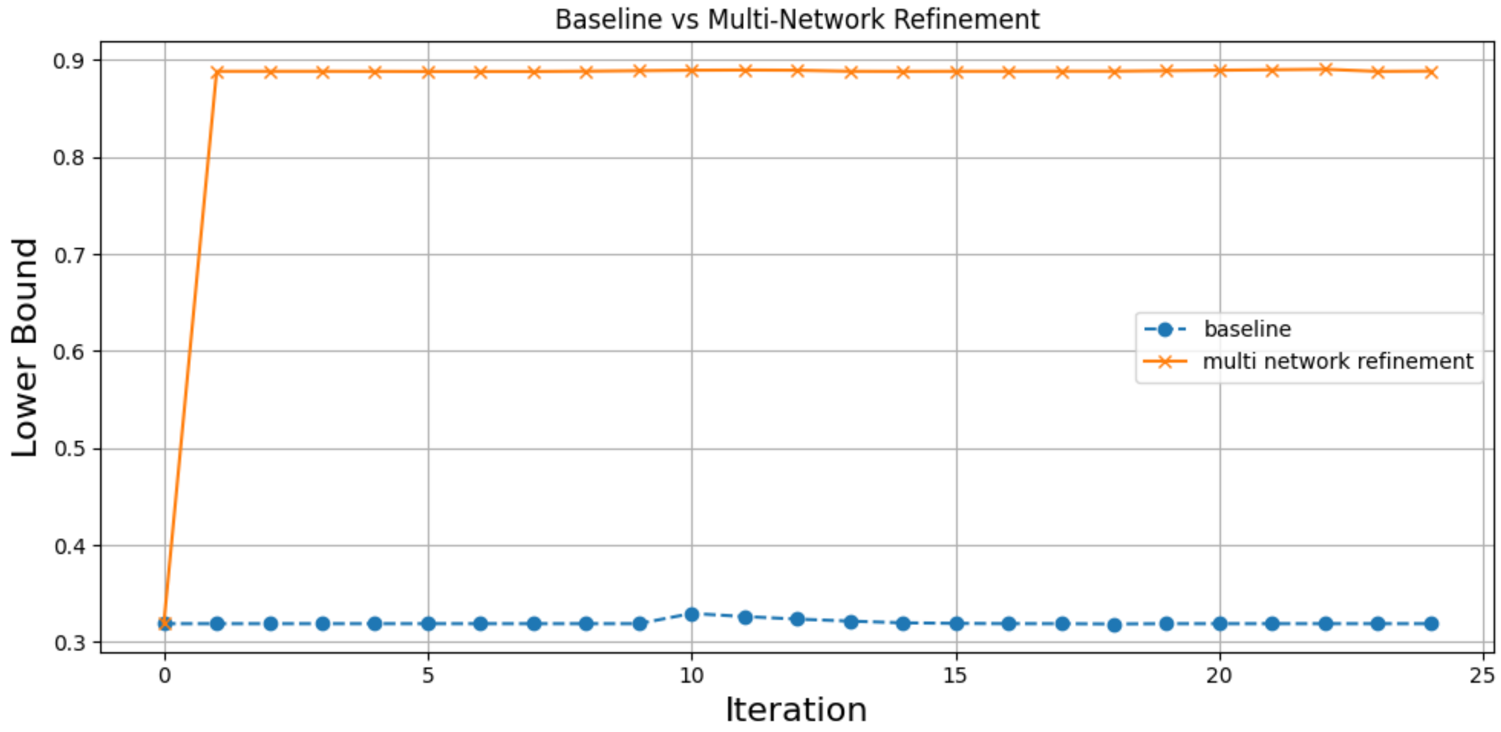
- ❑ **Find the closed form of the inner minimization problem**

Methodology: Training

Output Spec: $\Psi = \psi_1 \vee \psi_2$

- ❑ Train ensemble with **non-overlapping adversarial examples** from ψ_1 and ψ_2
- ❑ Ensures overall correctness even when some properties are not satisfied

Results For Disjunctive Ψ



For disjunctive output specification. The proposed bound refinement produces a tighter bound.

Experiments from an ensemble of COLT, SABR, CITRUS networks with $\epsilon = 3/255$

Results For Ensembles

ϵ	Net 1 Training	Net 1 Accuracy	Net 2 Training	Net 2 Accuracy	Net 3 Training	Net 3 Accuracy	Ensemble
2/255	COLT	50	CITRUS	54	SABR	58	56
3/255	COLT	43	CITRUS	43	SABR	44	43
4/255	COLT	41	CITRUS	32	SABR	29	30
0.25/255	Standard	54	DiffAI	51	PGD	68	61
0.5/255	Standard	48	DiffAI	50	PGD	65	59
2.0/255	Standard	1.0	DiffAI	45	PGD	36	24

Questions?