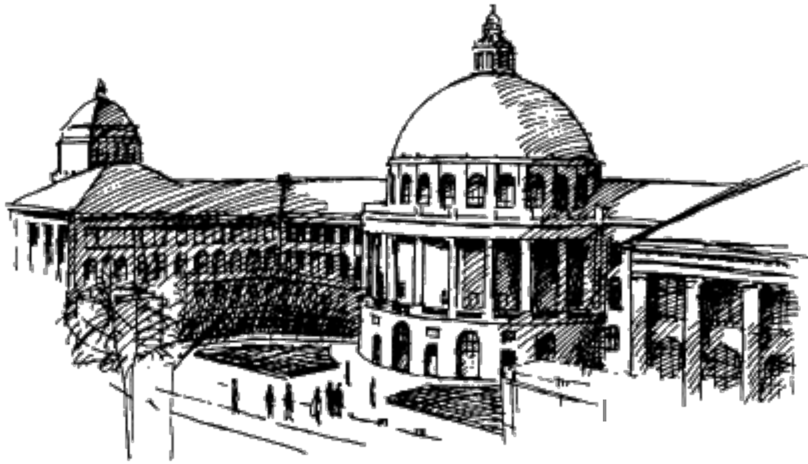
Safe and Robust Deep Learning

Gagandeep Singh
PhD Student
Department of Computer Science

ETH Zürich



SafeAI @ ETH Zurich (safeai.ethz.ch)

Joint work with

Martin Vechev

Markus Püschel

Timon Gehr

Matthew Mirman

Mislav Balunovic

Maximilian Baader

Petar Tsankov

Dana Drachsler

Publications:

- S&P'18: AI2: Safety and Robustness Certification of Neural Networks with Abstract Interpretation
- NeurIPS'18: Fast and Effective Robustness Certification
- POPL'19: An Abstract Domain for Certifying Neural Networks
- ICLR'19: Boosting Robustness Certification of Neural Networks
- ICML'18: Differentiable Abstract Interpretation for Provably Robust Neural Networks
- ICML'19: DL2: Training and Querying Neural Network with Logic

Systems:

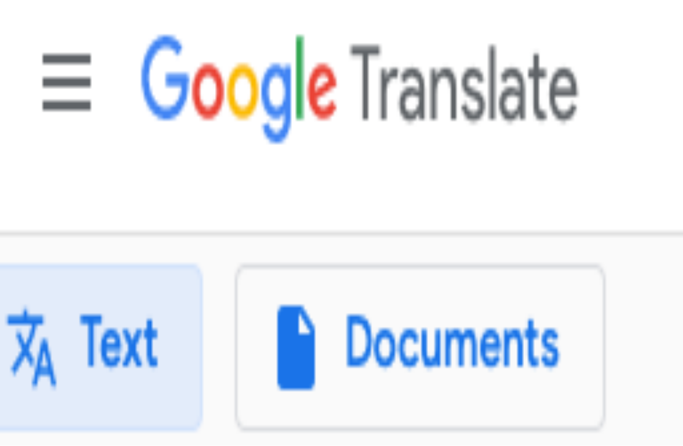
- ERAN: Generic neural network verifier
- DiffAI: System for training provably robust networks
- DL2: System for training and querying networks with logical constraints

Deep learning systems

Self driving cars

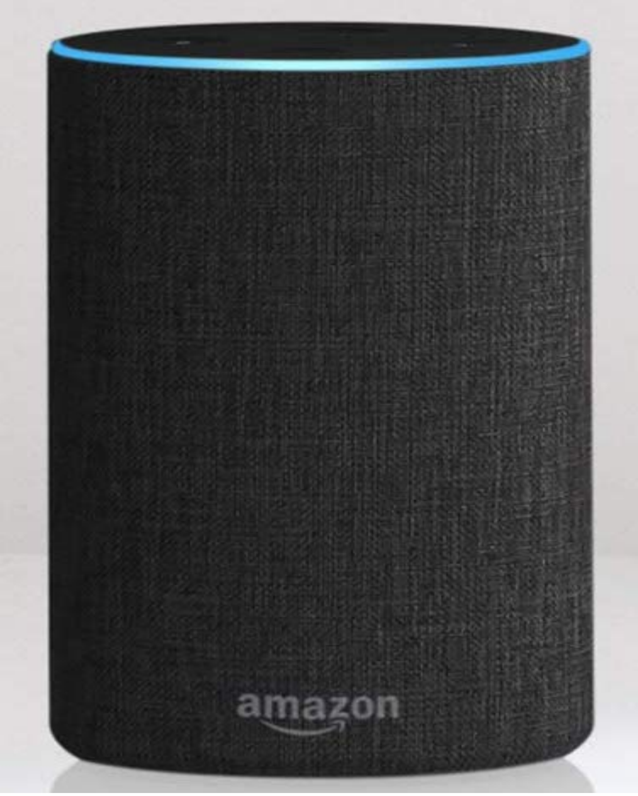
https://waymo.com/tech/

Translation



https://translate.google.com

Voice assistant



https://www.amazon.com/Amazon-Echo-And-Alexa-Devices

Attacks on deep learning

The self-driving car incorrectly decides to turn right on Input 2 and crashes into the guardrail

(a) Input 1 (b) Input 2 (darker version of 1)

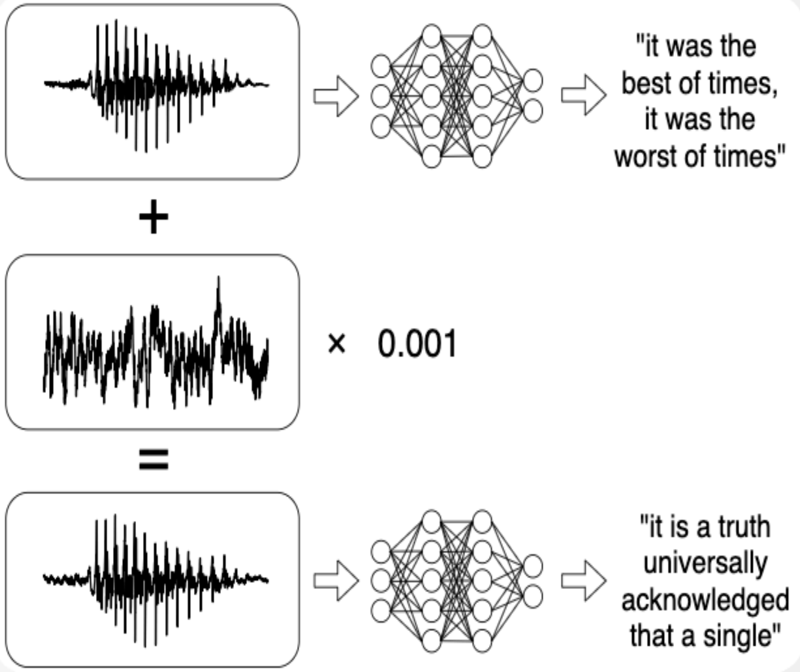
DeepXplore: Automated Whitebox Testing of Deep Learning Systems, SOSP'17

The Ensemble model is fooled by the addition of an adversarial distracting sentence in blue.

Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

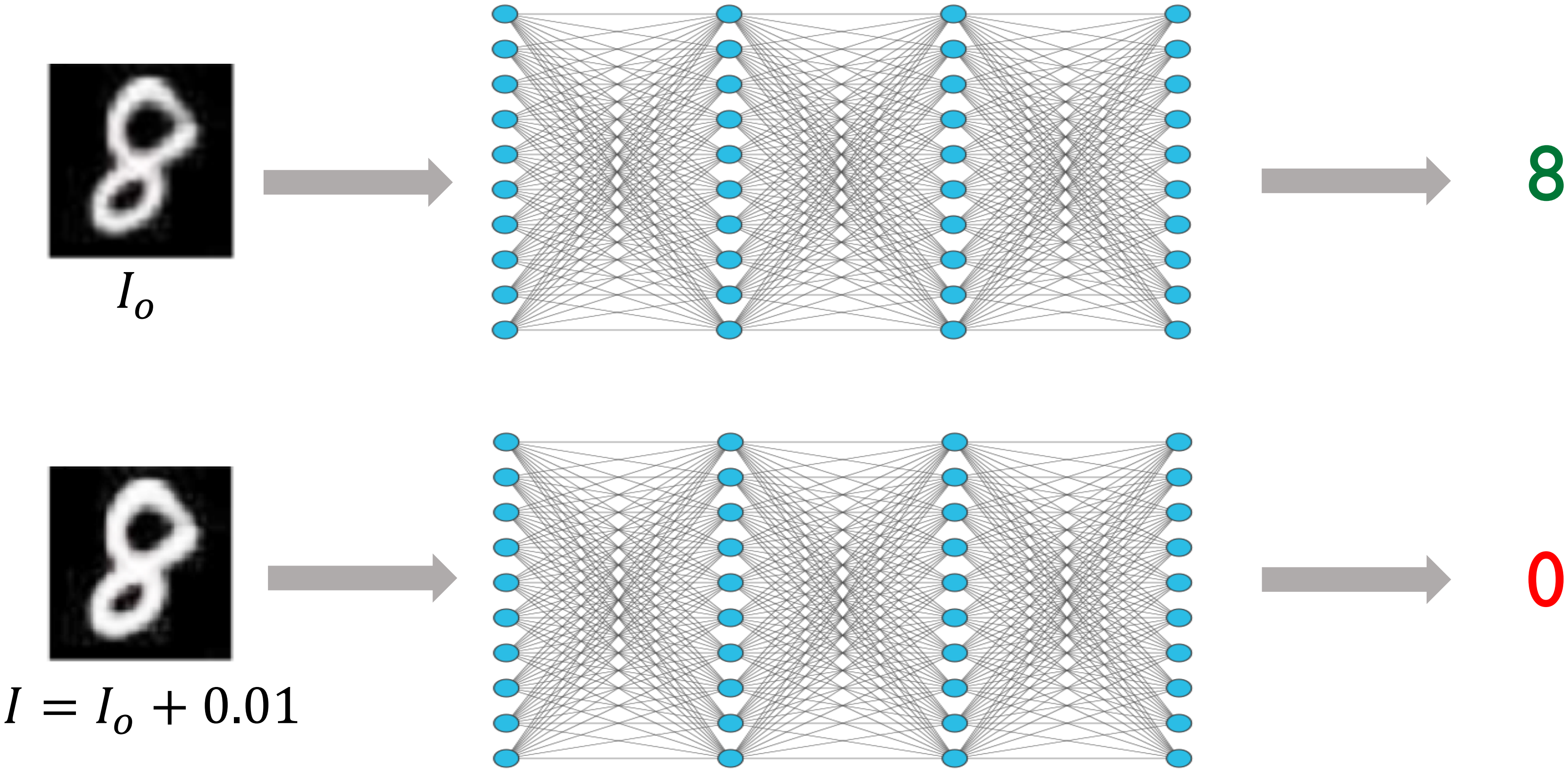
Adversarial Examples for Evaluating Reading Comprehension Systems, EMNLP'17

Adding small noise to the input audio makes the network transcribe any arbitrary phrase



Audio Adversarial Examples: Targeted Attacks on Speech-to-Text, ICML 2018

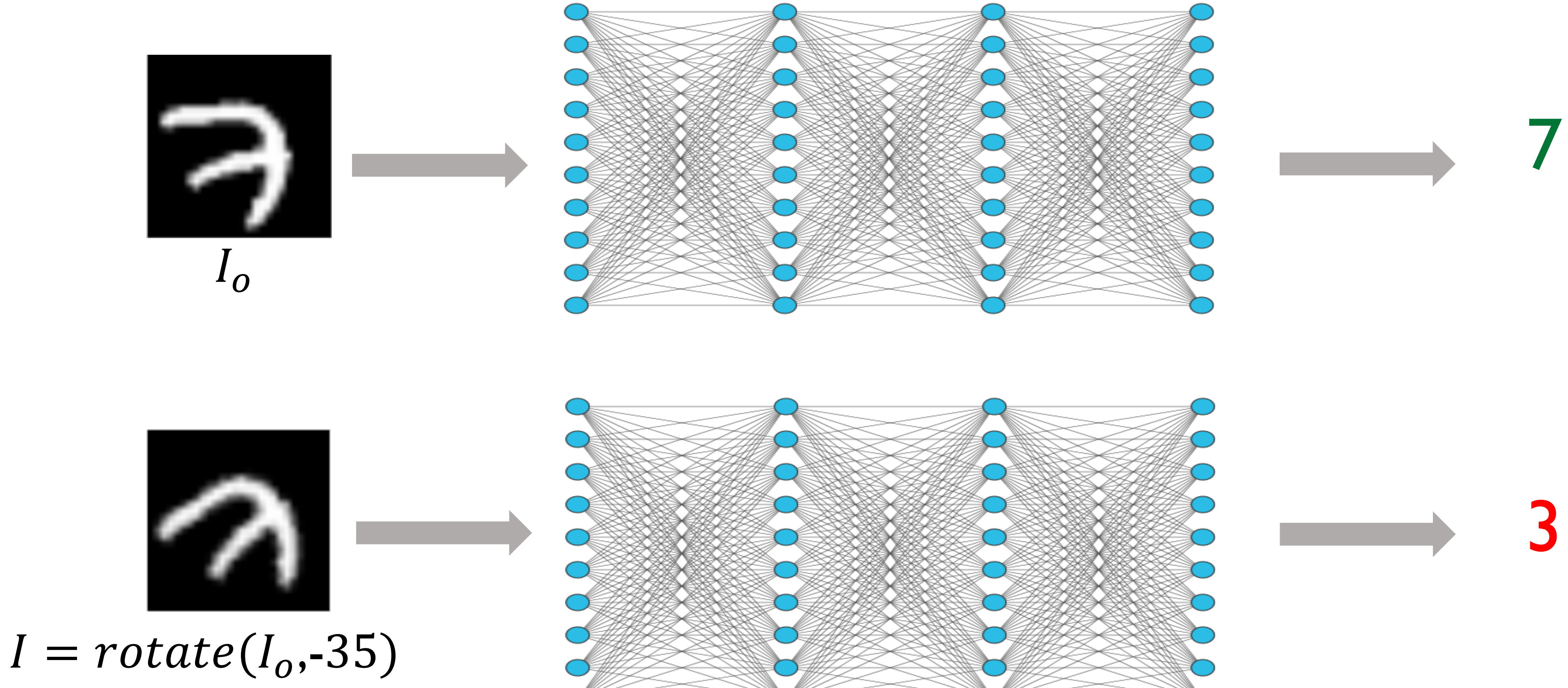
Attacks based on intensity changes in images



To verify absence of attack:

L_∞-norm: consider all images I in the ϵ-ball $\mathcal{B}_{(I_0,\infty)}(\epsilon)$ around I_0

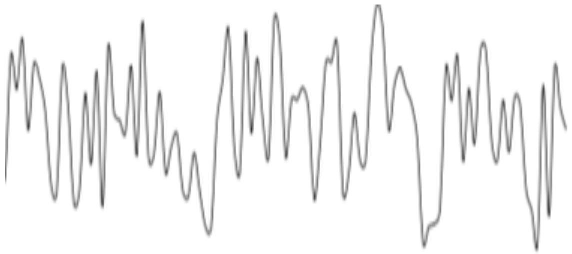
Attacks based on geometric transformations

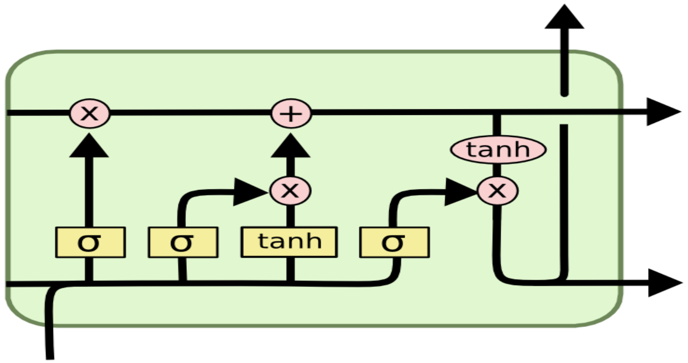


To verify absence of attack:

Consider all images I obtained by applying geometric transformations to I_0

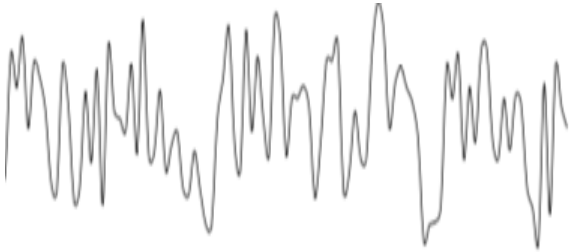
Attacks based on intensity changes to sound

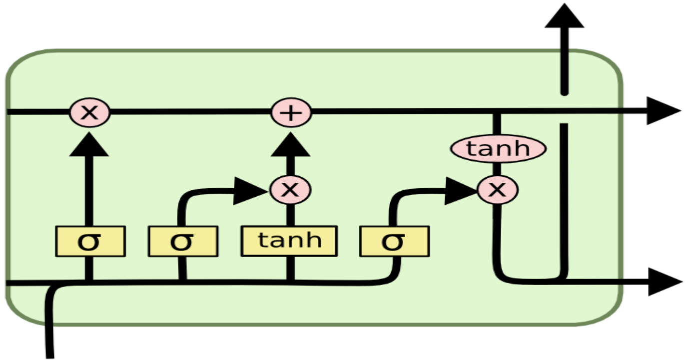




“Stop”

s_o





“Go”

$s = s_o - 110\ dB$

To verify absence of attack:

Consider all signals s in the ϵ-ball $\mathcal{B}_{(s_0,\infty)}(\epsilon)$ around s_0

Neural network verification: problem statement

Given: Neural Network f, Input Region $\mathcal{R}$, Safety Property ψ

Prove: $\forall I \in \mathcal{R}$, prove that $f(I)$ satisfies ψ

Example networks and regions:

Image classification network f
Region $\mathcal{R}$ based on changes to pixel intensity
Region $\mathcal{R}$ based on geometric: e.g., *rotation*

Speech recognition network f
Region $\mathcal{R}$ based on added noise to audio signal

Aircraft collision avoidance network f
Region $\mathcal{R}$ based on input sensor values

Input Region $\mathcal{R}$ can contain an infinite number of inputs, thus enumeration is infeasible

Experimental vs. certified robustness

Experimental robustness	Certified robustness
Tries to find violating inputs	**Prove** absence of violating inputs
Like testing, no full guarantees	Actual verification guarantees
E.g. Goodfellow 2014, Carlini & Wagner 2016, Madry et al. 2017	E.g.: Reluplex [2017], Wong et al. 2018, AI2 [2018]

In this talk we will focus on certified robustness

General approaches to network verification

Complete verifiers, but suffer from scalability issues:
SMT: Reluplex [CAV'17], MILP: MIPVerify [ICLR'19],
Splitting: Neurify [NeurIPS'18],…

Incomplete verifiers, trade-off precision for scalability:
Box/HBox [ICML'18], SDP [ICLR'18], Wong et.al. [ICML'18], FastLin [ICML'18], Crown [NeurIPS'18],…

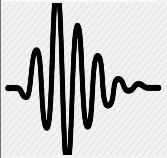
Key Challenge: scalable and precise automated verifier

Network verification with ERAN

Input region

Based on **Pixel Intensity** changes

Based on **Geometric** transformations: vector fields, rotations, etc.



Based on Audio processing

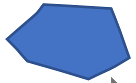
Aircraft sensors

Possible sensor values

Neural Network

Fully connected	ReLU
Convolutional	Sigmoid
Residual	Tanh
LSTM	Maxpool

Safety Property



ERAN verification framework
https://github.com/eth-sri/eran

- Box
- DeepZ [NeurIPS'18]
- DeepPoly [POPL'19]
- GPUPoly [submitted]
- RefineZono [ICLR'19]: MILP + DeepZ
- KPoly [submitted]: MILP + DeepPoly

Extensible to other verification tasks

State-of-the-art **complete** and **incomplete** verification

Sound w.r.t. floating point arithmetic

Yes

No

Complete and incomplete verification with ERAN

Faster Complete Verification

Aircraft collision avoidance system (ACAS)		
Reluplex	**Neurify**	**ERAN**
> 32 hours	921 sec	227 sec

Scalable Incomplete Verification

CIFAR10 ResNet-34		
ϵ	**%verified**	**Time (s)**
0.03	66%	79 sec

Geometric and audio verification with ERAN

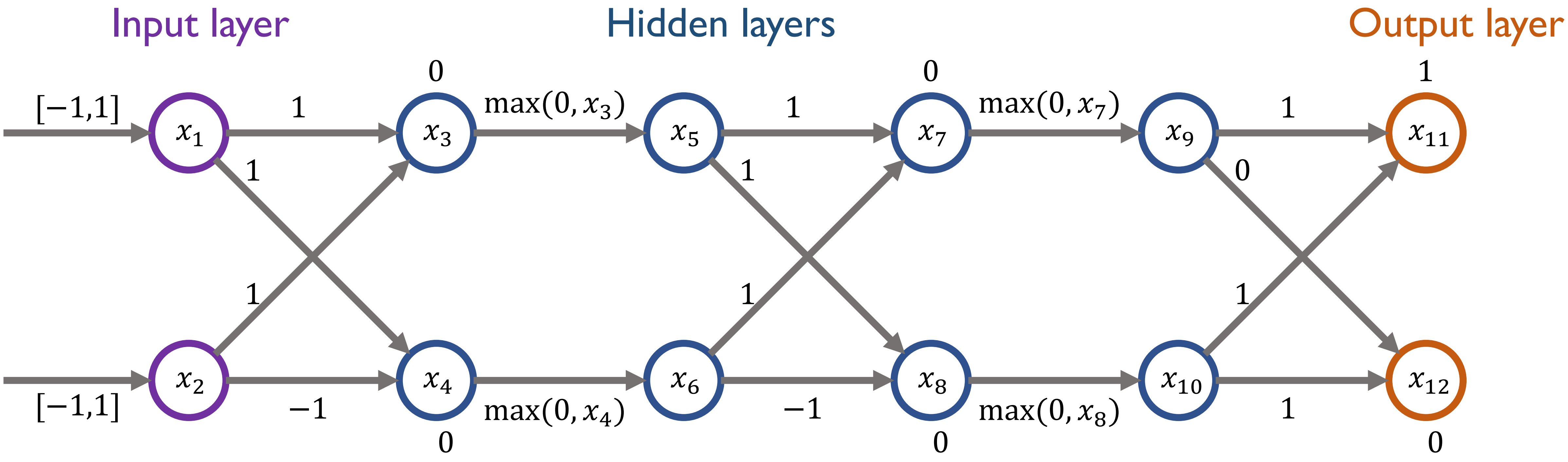
Geometric Verification

Rotation between -30° and 30° on MNIST CNN with 4,804 neurons		
ϵ	%verified	Time(s)
0.001	86	10 sec

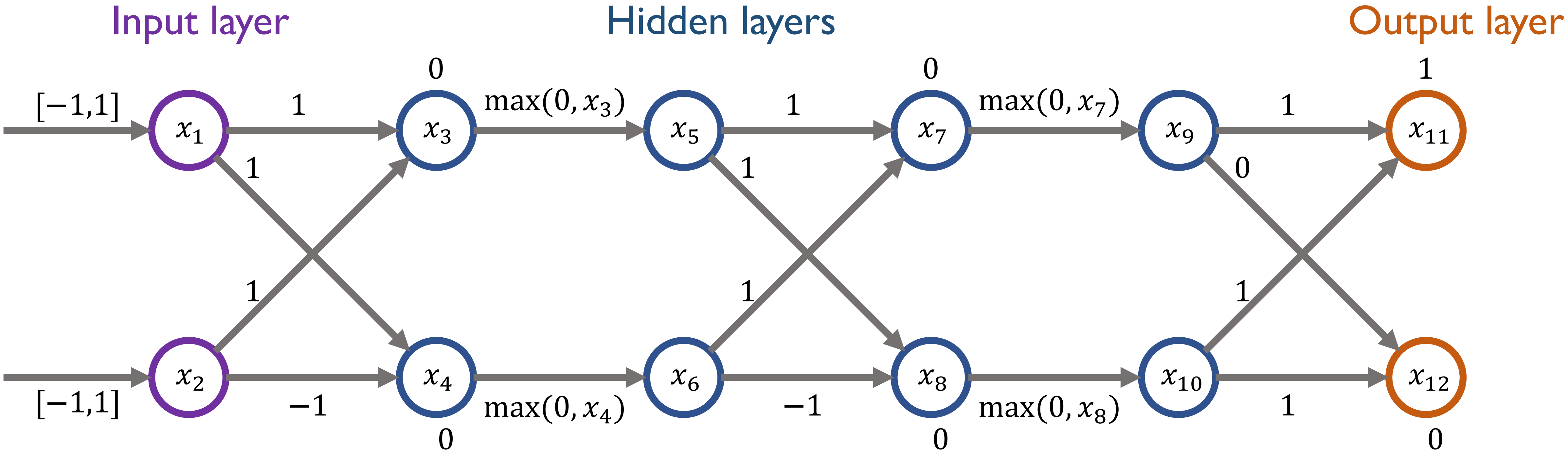
Audio Verification

LSTM with 64 hidden neurons		
ϵ	%verified	Time (s)
-110 dB	90%	9 sec

Example: analysis of a toy neural network



We want to prove that $x_{11} > x_{12}$ for all values of x_1, x_2 in the input set



$$\min\ x_{11} - x_{12}$$

$$s.t.: \ x_{11} = x_9 + x_{10} + 1,\ x_{12} = x_{10},$$
$$x_9 = \mathbf{max}(0, x_7),\ x_{10} = \mathbf{max}(0, x_8),$$
$$x_7 = x_5 + x_6,\ x_8 = x_5 - x_6,$$
$$x_5 = \mathbf{max}(0, x_3),\ x_6 = \mathbf{max}(0, x_4),$$
$$x_3 = x_1 + x_2,\ x_4 = x_1 - x_2,$$
$$-1 \leq x_1 \leq 1,\ -1 \leq x_2 \leq 1.$$

Each $x_j = \mathbf{max}(0, x_i)$ corresponds to

$(x_i \leq 0$ and $x_j = 0)$ or

$(x_i > 0$ and $x_j = x_i)$

Solver has to explore two paths per ReLU resulting in exponential number of paths

Complete verification with solvers often does not scale

Abstract interpretation


Patrick and Radhia Cousot
Inventors

An elegant framework for approximating concrete behaviors

Key Concept: Abstract Domain

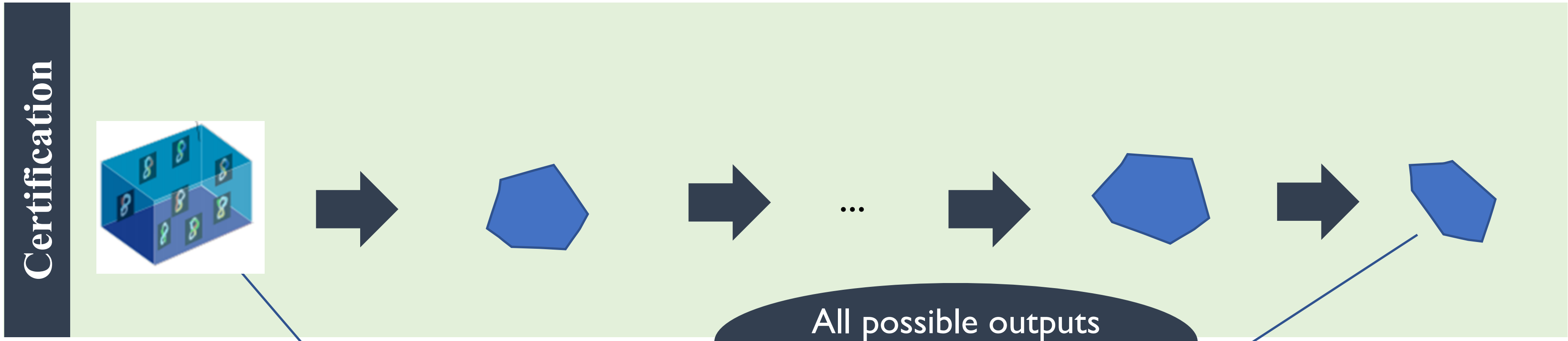
Abstract element: approximates set of concrete points

Concretization function γ: concretizes an abstract element to the set of points that it represents.

Abstract transformers: approximate the effect of applying concrete transformers e.g. affine, ReLU

Tradeoff between the precision and the scalability of an abstract domain

Network verification with ERAN: high level idea



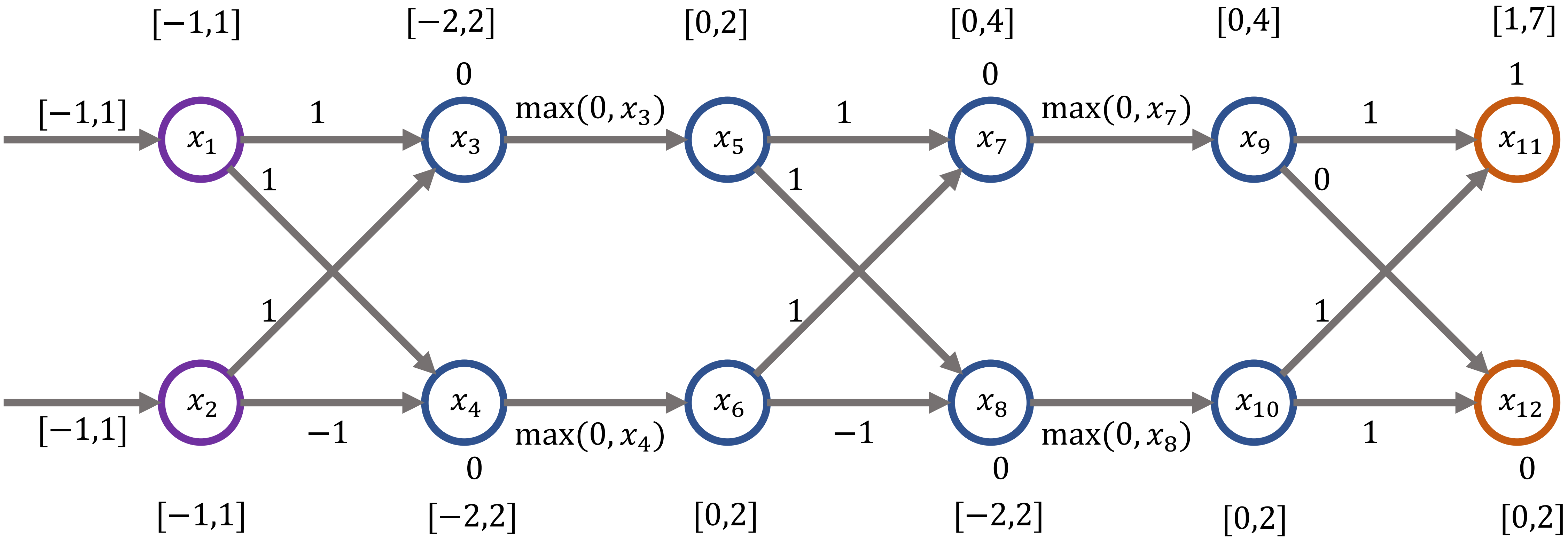
Attacker region L_∞ ball with $\epsilon = 0.1$:

$x_0 = [0.1,0.3]$
$x_1 = [0.4,0.6]$
$x_2 = [0.18,0.36]$
…
$x_{784} = [0.7,0.9]$

Output constraint φ_n

$x_0 = 0$
$x_1 = 2.60 + 0.015\eta_0 + 0.023\eta_1 + 5.181\eta_2 + \cdots$
$x_2 = 4.63 - 0.005\eta_0 - 0.006\eta_1 + 0.023\eta_2 + \cdots$
…
$x_9 = 0.12 - 0.125\eta_0 + 0.102\eta_1 + 3.012\eta_2 + \cdots$
$\forall i.\, \eta_i \in [0,1]$

Box approximation (scalable but imprecise)



Verification with the Box domain fails as it cannot capture relational information

DeepPoly approximation [POPL'19]

Shape: associate a lower polyhedral $a_i^{\leq}$ and an upper polyhedral $a_i^{\geq}$ constraint with each x_i

$$a_i^{\leq}, a_i^{\geq} \in \{x \mapsto v + \sum_{j\in[i-1]} w_j \cdot x_j \mid v \in \mathbb{R} \cup \{-\infty, +\infty\}, w \in \mathbb{R}^{i-1}\} \text{ for } i \in [n]$$

Concretization of abstract element a:

$$\gamma_n(a) = \{x \in \mathbb{R}^n \mid \forall i \in [n].\, a_i^{\leq}(x) \leq x_i \wedge a_i^{\geq}(x) \geq x_i\}$$

Domain invariant: store auxiliary concrete lower and upper bounds l_i, u_i for each x_i

$$\gamma_n(a) \subseteq \times_{i\in[n]} [l_i, u_i]$$

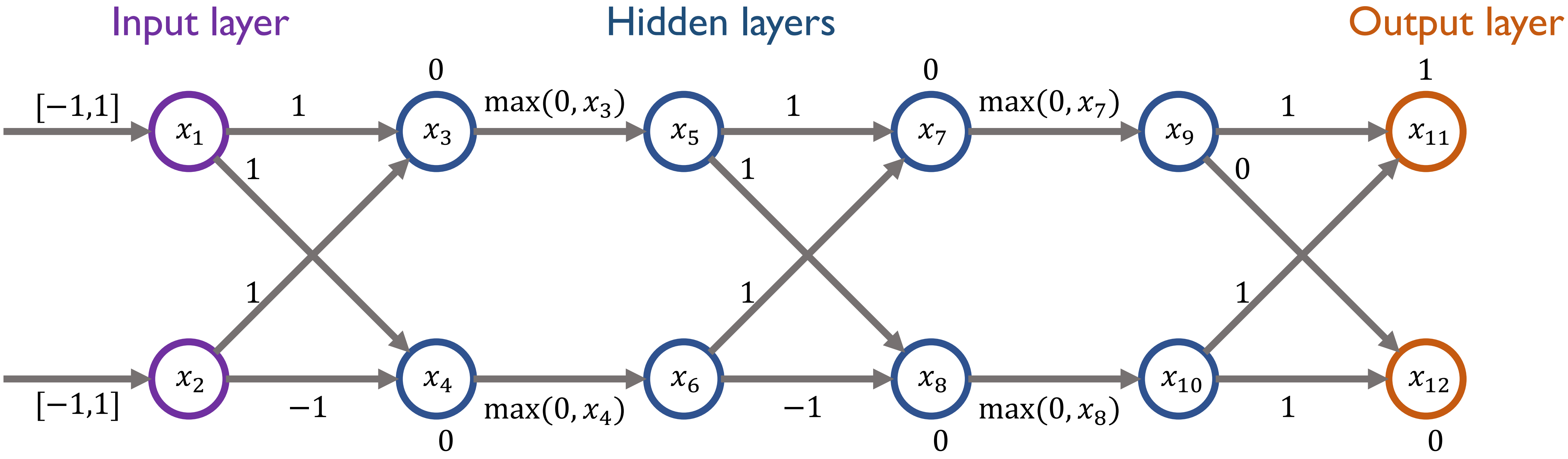
- less precise than Polyhedra, restriction needed to ensure scalability
- captures affine transformation precisely unlike Octagon, TVPI
- custom transformers for ReLU, sigmoid, tanh, and maxpool activations

n: #neurons, m: #constraints

w_{max}: max #neurons in a layer, L: # layers

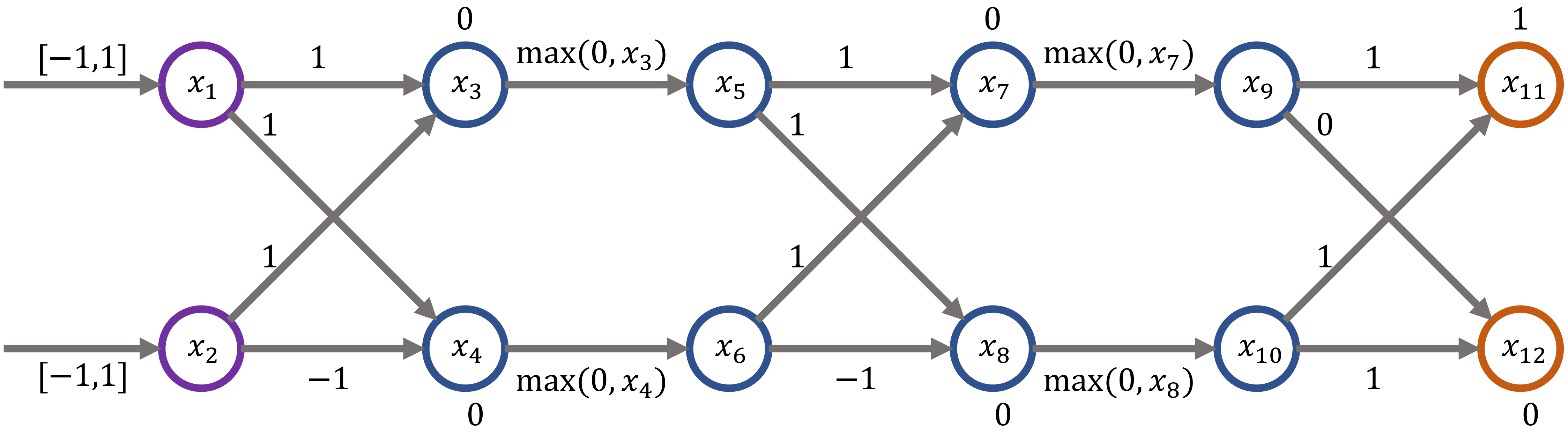
Transformer	Polyhedra	Our domain
Affine	$O(nm^2)$	$O(w_{max}^2 L)$
ReLU	$O(\exp(n, m))$	$O(1)$

Example: analysis of a toy neural network



1. 4 constraints per neuron
2. Pointwise transformers => parallelizable.
3. Backsubstitution => helps precision.
4. Non-linear activations => approximate and minimize the area

$$\begin{aligned} &\langle x_1 \geq -1, && \langle x_3 \geq x_1 + x_2, \\ &x_1 \leq 1, && x_3 \leq x_1 + x_2, \\ &l_1 = -1, && l_3 = -2, \\ &u_1 = 1\rangle && u_3 = 2\rangle \end{aligned}$$



$$\begin{aligned} &\langle x_2 \geq -1, && \langle x_4 \geq x_1 - x_2, \\ &x_2 \leq 1, && x_4 \leq x_1 - x_2, \\ &l_2 = -1, && l_4 = -2, \\ &u_2 = 1\rangle && u_4 = 2\rangle \end{aligned}$$

ReLU activation

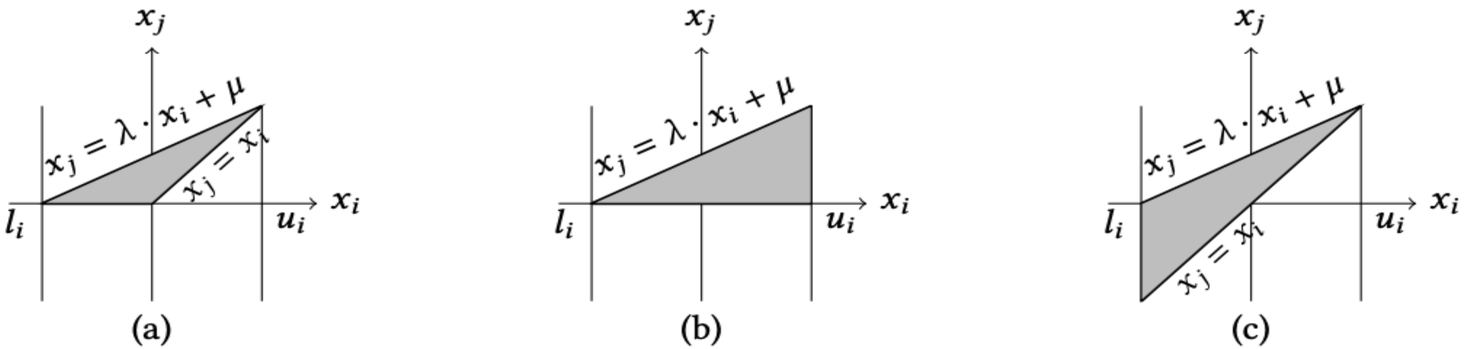
$$\langle x_5 \geq 0,$$
$$x_5 \leq 0.5 \cdot x_3 + 1,$$
$$l_5 = 0,$$
$$u_5 = 2\rangle$$

Pointwise transformer for $x_j := max(0, x_i)$ that uses l_i, u_i

$if\ u_i \leq 0, a_j^{\leq} = a_j^{\geq} = 0, l_j = u_j = 0,$

$if\ l_i \geq 0, a_j^{\leq} = a_j^{\geq} = x_i, l_j = l_i, u_j = u_i,$

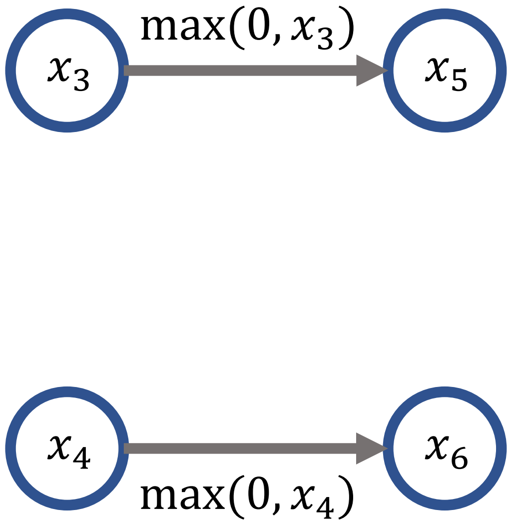
$if\ l_i < 0\ and\ u_i > 0$



(a) (b) (c)

(a)	(b)	(c)
$x_i \leq x_j, 0 \leq x_j,$	$0 \leq x_j,$	$x_i \leq x_j,$
$x_j \leq u_i(x_i - l_i)/(u_i - l_i).$	$x_j \leq u_i(x_i - l_i)/(u_i - l_i),$	$x_j \leq u_i(x_i - l_i)/(u_i - l_i),$
$l_j = 0, u_j = u_i$	$l_j = 0,\ u_j = u_i$	$l_j = l_i,\ u_j = u_i$

choose (b) or (c) depending on the area

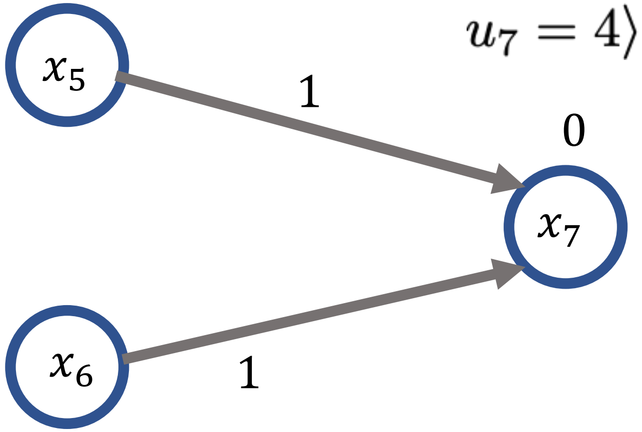


$$\langle x_6 \geq 0,$$
$$x_6 \leq 0.5 \cdot x_4 + 1,$$
$$l_6 = 0,$$
$$u_6 = 2\rangle$$

Constant runtime

Affine transformation after ReLU

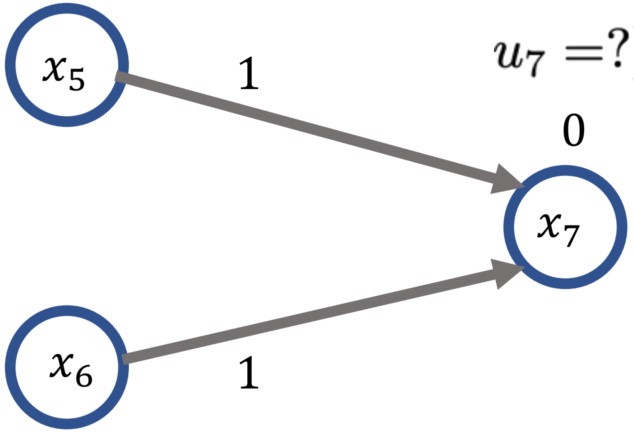
$$\langle x_7 \geq x_5 + x_6,$$
$$x_7 \leq x_5 + x_6,$$
$$l_7 = 0,$$
$$u_7 = 4\rangle$$



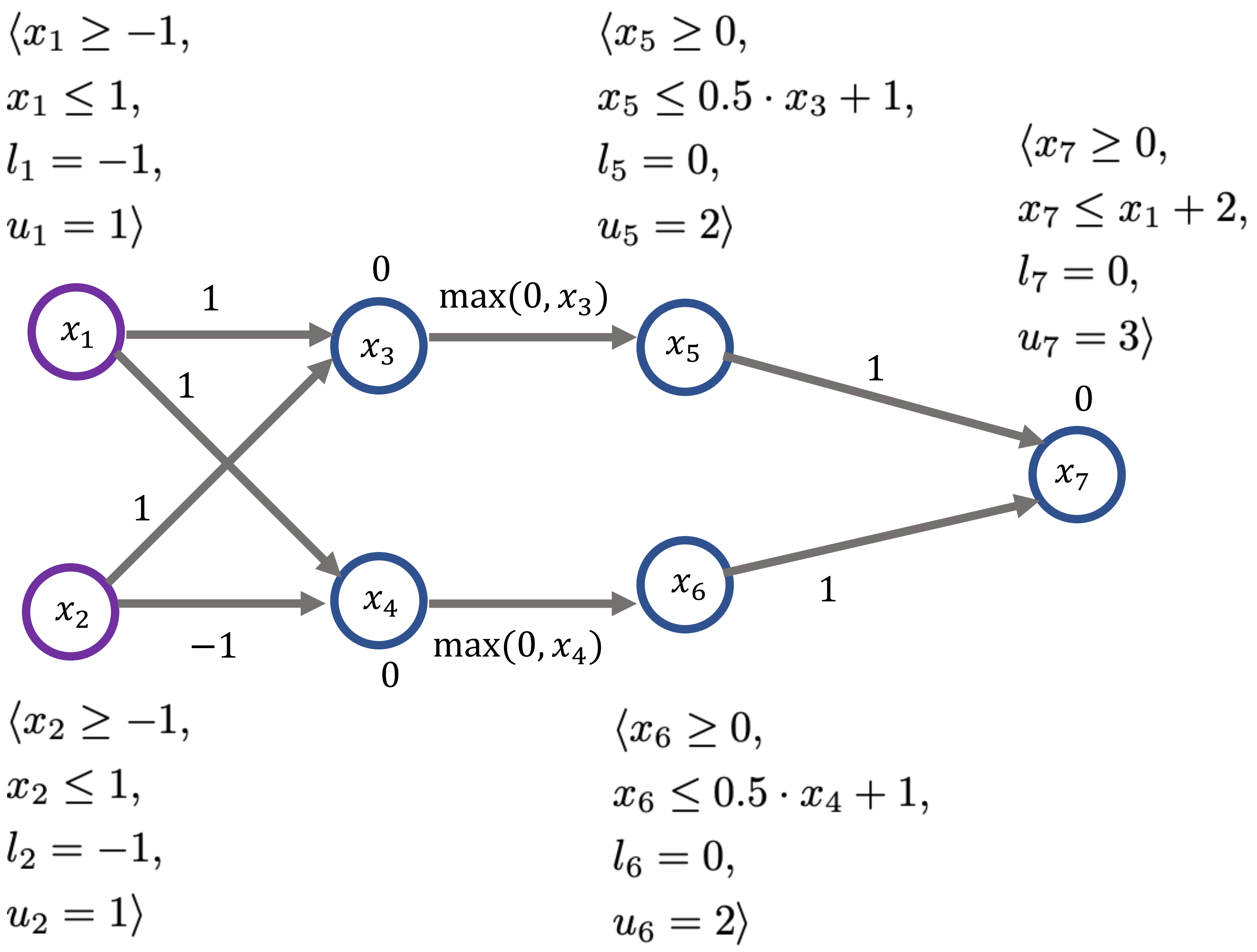
Imprecise upper bound u_7 by substituting u_5, u_6 for x_5 and x_6 in $a_7^{\geq}$

Backsubstitution

$$\langle x_7 \geq 0,$$
$$x_7 \leq 0.5 \cdot x_3 + 0.5 \cdot x_4 + 2,$$
$$l_7 = ?,$$
$$u_7 = ?\rangle$$



$$\langle x_6 \geq 0,$$
$$x_6 \leq 0.5 \cdot x_4 + 1,$$
$$l_6 = 0,$$
$$u_6 = 2\rangle$$



Affine transformation with backsubstitution is pointwise, complexity: $O(w_{max}^2 L)$

$$\langle x_1 \geq -1,\; x_1 \leq 1,\; l_1 = -1,\; u_1 = 1\rangle$$

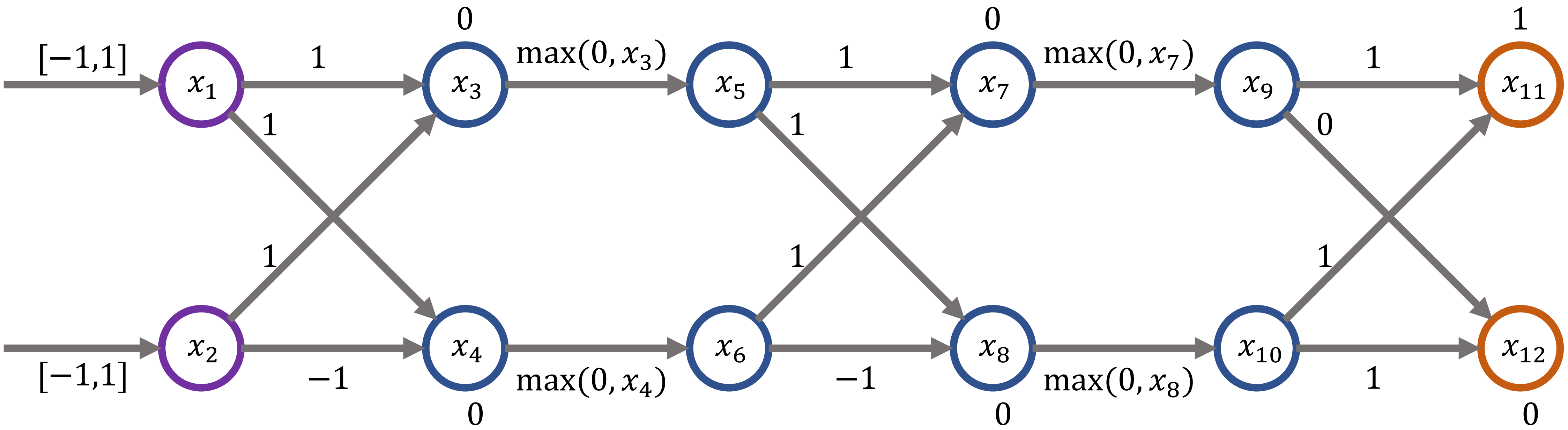
$$\langle x_3 \geq x_1 + x_2,\; x_3 \leq x_1 + x_2,\; l_3 = -2,\; u_3 = 2\rangle$$

$$\langle x_5 \geq 0,\; x_5 \leq 0.5 \cdot x_3 + 1,\; l_5 = 0,\; u_5 = 2\rangle$$

$$\langle x_7 \geq x_5 + x_6,\; x_7 \leq x_5 + x_6,\; l_7 = 0,\; u_7 = 3\rangle$$

$$\langle x_9 \geq x_7,\; x_9 \leq x_7,\; l_9 = 0,\; u_9 = 3\rangle$$

$$\langle x_{11} \geq x_9 + x_{10} + 1,\; x_{11} \leq x_9 + x_{10} + 1,\; l_{11} = 1,\; u_{11} = 5.5\rangle$$



$$\langle x_2 \geq -1,\; x_2 \leq 1,\; l_2 = -1,\; u_2 = 1\rangle$$

$$\langle x_4 \geq x_1 - x_2,\; x_4 \leq x_1 - x_2,\; l_4 = -2,\; u_4 = 2\rangle$$

$$\langle x_6 \geq 0,\; x_6 \leq 0.5 \cdot x_4 + 1,\; l_6 = 0,\; u_6 = 2\rangle$$

$$\langle x_8 \geq x_5 - x_6,\; x_8 \leq x_5 - x_6,\; l_8 = -2,\; u_8 = 2\rangle$$

$$\langle x_{10} \geq 0,\; x_{10} \leq 0.5 \cdot x_8 + 1,\; l_{10} = 0,\; u_{10} = 2\rangle$$

$$\langle x_{12} \geq x_{10},\; x_{11} \leq x_{10},\; l_{12} = 0,\; u_{12} = 2\rangle$$

Checking for robustness

Prove $x_{11} - x_{12} > 0$ for all inputs in $[-1,1] \times [-1,1]$

$$\langle x_{11} \geq x_9 + x_{10} + 1, \quad x_{11} \leq x_9 + x_{10} + 1, \quad l_{11} = 1, \quad u_{11} = 5.5 \rangle$$

$$\langle x_{12} \geq x_{10}, \quad x_{12} \leq x_{10}, \quad l_{12} = 0, \quad u_{12} = 0 \rangle$$

Computing lower bound for $x_{11} - x_{12}$ using l_{11}, u_{12} gives -1 which is an imprecise result

With backsubstitution, one gets 1 as the lower bound for $x_{11} - x_{12}$, proving robustness

Abstract interpretation + solvers

Key Idea: refine abstract interpretation results by calling the solver

- Refine neuron bounds before ReLU transformer is applied => less area

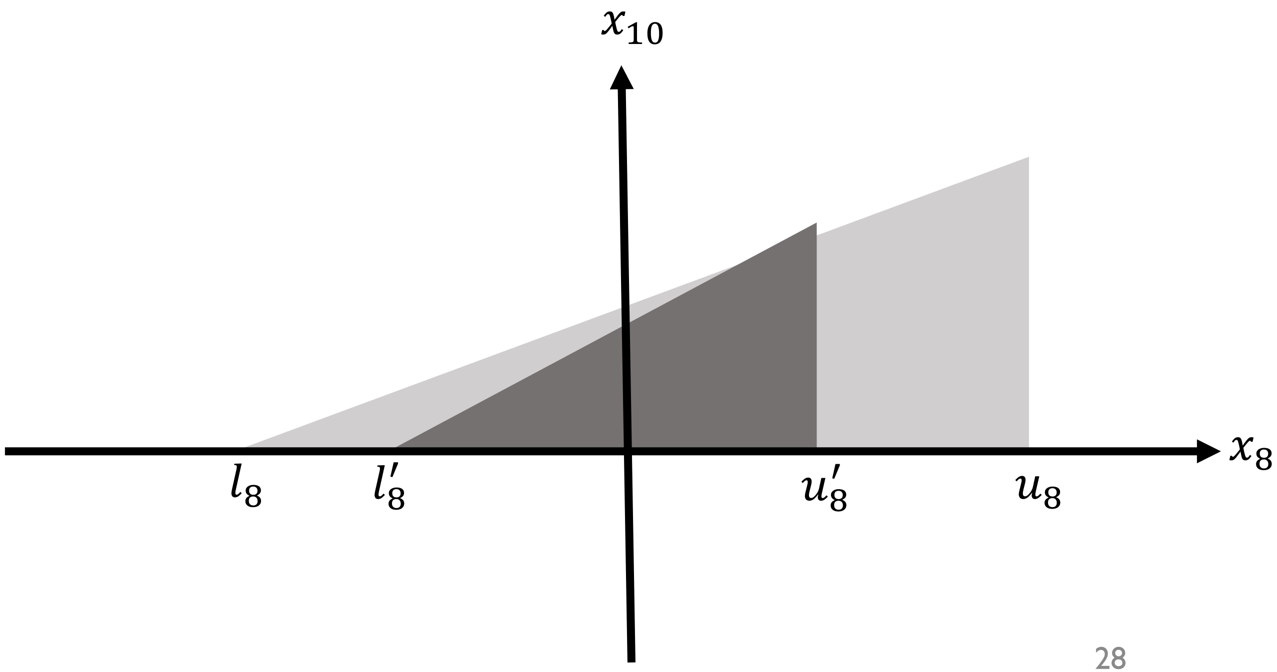
$$l'_8 := min\ x_8$$

$$s.t. : x_8 = x_5 - x_6,$$

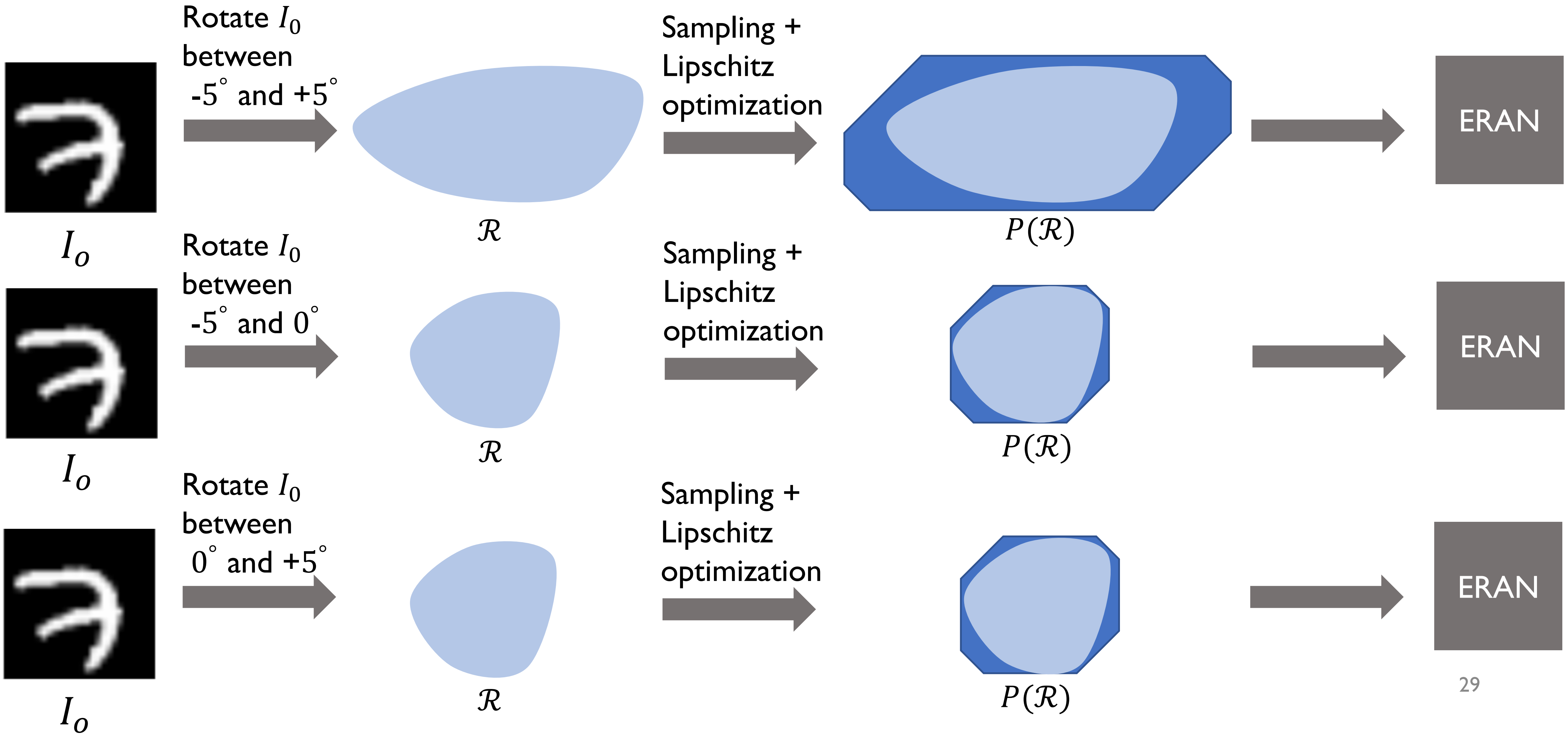
$$x_5 = \mathbf{max}(0, x_3),\ x_6 = \mathbf{max}(0, x_4),$$

$$x_3 = x_1 + x_2,\ x_4 = x_1 - x_2,$$

$$-1 \le x_1 \le 1,\ -1 \le x_2 \le 1.$$



Verification against geometric attacks



Medium sized benchmarks

Dataset	Model	Type	#Neurons	#Layers	Defense
MNIST	6×100	feedforward	610	6	None
	6×200	feedforward	1,210	6	None
	9×200	feedforward	1,810	9	None
	ConvSmall	convolutional	3,604	3	DiffAI
	ConvBig	convolutional	34,688	6	DiffAI
CIFAR10	ConvSmall	convolutional	4,852	3	Wong et al.
	ConvBig	convolutional	62,464	6	PGD

Results on medium benchmarks (100 test images)

Dataset	Model	#correct	ϵ	DeepPoly		kPoly	
				%✅	time(s)	%✅	time(s)
MNIST	6×100	99	0.026	21	0.3	44	151
	6×200	99	0.015	32	0.5	56	387
	9×200	97	0.015	29	0.9	54	1040
	ConvSmall	100	0.12	13	6.0	28	1018
	ConvBig	100	0.3	93	12.3	93	286
CIFAR10	ConvSmall	38	0.03	35	0.4	35	1.4
	ConvBig	65	0.008	39	49	40	2882

Large benchmarks

Dataset	Model	Type	#Neurons	#Layers	Defense
CIFAR10	ResNetTiny	residual	311K	12	PGD
	ResNet18	residual	558K	18	PGD
	ResNetTiny	residual	311K	12	DiffAI
	SkipNet18	residual	558K	18	DiffAI
	ResNet18	residual	558K	18	DiffAI
	ResNet34	residual	967K	34	DiffAI

Results on large benchmarks (500 test images)

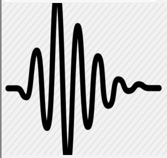
Model	Training	#correct	ϵ	Hbox[ICML'18]		GPUPoly	
				% ✅	time(s)	% ✅	time(s)
ResNetTiny	PGD	391	0.002	0	0.3	322	30
ResNet18	PGD	419	0.002	0	6.8	324	1400
ResNetTiny	DiffAI	184	0.03	118	0.3	127	7.6
SkipNet18	DiffAI	168	0.03	130	6.1	140	57
ResNet18	DiffAI	193	0.03	129	6.3	139	37
ResNet34	DiffAI	174	0.03	103	16	114	79

Network verification with ERAN

Input region

Based on **Pixel Intensity** changes

Based on **Geometric** transformations: vector fields, rotations, etc.



Based on Audio processing

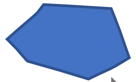
Aircraft sensors

Possible sensor values

Neural Network

Fully connected	ReLU
Convolutional	Sigmoid
Residual	Tanh
LSTM	Maxpool

Safety Property



ERAN verification framework
https://github.com/eth-sri/eran

- Box
- DeepZ [NeurIPS'18]
- DeepPoly [POPL'19]
- GPUPoly [submitted]
- RefineZono [ICLR'19]: MILP + DeepZ
- K-Poly [submitted]: MILP + DeepPoly

Extensible to other verification tasks

State-of-the-art **complete** and **incomplete** verification

Sound w.r.t. floating point arithmetic

Yes

No

In-progress work in verification/training (sample)

Verification Precision: More precise convex relaxations by considering multiple ReLUs

Verification Scalability: GPU-based custom abstract domains for handling large nets

Theory: Proof on Existence of Accurate and Provable Networks with Box

Provable Training: Procedure for training Provable and Accurate Networks

Applications: e.g., reinforcement learning, geometric, audio, sensors

Attacks on Deep Learning

The self-driving car incorrectly decides to turn right on Input 2 and crashes into the guardrail

(a) Input 1 (b) Input 2 (darker version of 1)

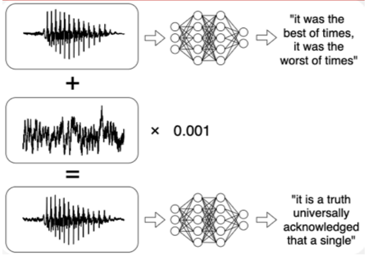
DeepXplore: Automated Whitebox Testing of Deep Learning Systems, SOSP'17

The Ensemble model is fooled by the addition of an adversarial distracting sentence in blue.

Article: Super Bowl 50
Paragraph: *"Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV."*
Question: *"What is the name of the quarterback who was 38 in Super Bowl XXXIII?"*
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

Adversarial Examples for Evaluating Reading Comprehension Systems, EMNLP'17

Adding small noise to the input audio makes the network transcribe any arbitrary phrase



Audio Adversarial Examples: Targeted Attacks on Speech-to-Text, ICML 2018

4

Neural Network Verification: Problem statement

Given: Neural Network f, Input Region $\mathcal{R}$, Safety Property ψ

Prove: $\forall I \in \mathcal{R}$, prove that $f(I)$ satisfies ψ

Example networks and regions:

- **Image classification network f**
  - Region $\mathcal{R}$ based on changes to pixel intensity
  - Region $\mathcal{R}$ based on geometric: e.g., *rotation*
- **Speech recognition network f**
  - Region $\mathcal{R}$ based on added noise to audio signal
- **Aircraft collision avoidance network f**
  - Region $\mathcal{R}$ based on input sensor values

Input Region $\mathcal{R}$ can contain an infinite number of inputs, thus enumeration is infeasible

Network Verification with ERAN

Input region

Based on **Pixel Intensity** changes

Based on **Geometric** transformations: vector fields, rotations, etc.

Based on Audio processing

Aircraft sensors

Possible sensor values

Neural Network

Fully connected	ReLU
Convolutional	Sigmoid
Residual	Tanh
LSTM	Maxpool

Safety Property

ERAN verification framework
https://github.com/eth-sri/eran

- Box
- DeepZ [NeurIPS'18]
- DeepPoly [POPL'19]
- GPUPoly [submitted]
- RefineZono [ICLR'19]: MILP + DeepZ
- K-Poly [submitted]: MILP + DeepPoly

Extensible to other verification tasks

State-of-the-art **complete** and **incomplete** verification

Sound w.r.t. floating point arithmetic

Yes

No

Complete and Incomplete Verification with ERAN

Faster Complete Verification

Aircraft collision avoidance system (ACAS)		
Reluplex	**Neurify**	**ERAN**
> 32 hours	921 sec	227 sec

Scalable Incomplete Verification

CIFAR10 ResNet-34		
ϵ	**%verified**	**Time (s)**
0.03	66%	79 sec