Formal Reasoning About Confidence and Automated Verification of NNs

Mohammad Afzal, S. Akshay, Blaise Genest, Ashutosh Gupta
IIT Bombay India, TCS research Pune India, CNRS@CREATE & IPAL, Singapore

1. Motivation

Neural Networks in safety-critical applications:

- Autonomous vehicles [Bojarski M, et al, 2016]
- Medical diagnosis [Amato F, et al, 2013]
- Speech recognition [Hinton G, et al, 2012]

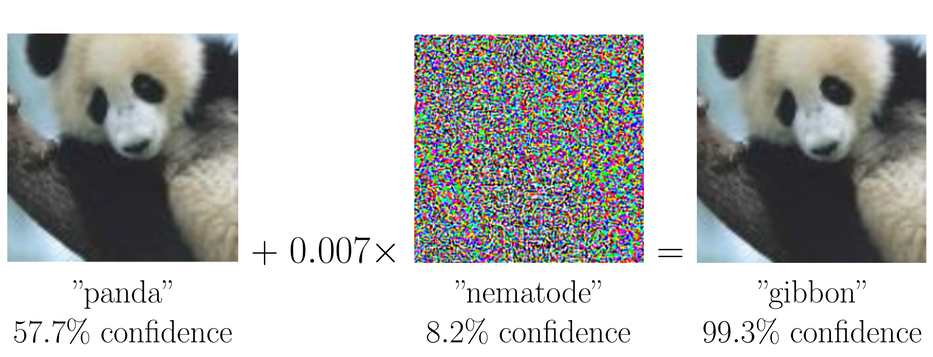


Figure 1: Image source: [Goodfellow I, et al, 2015]

2. Local robustness verification

- Neural network $N : \mathbb{R}^n \to \mathbb{C}$, $\mathbb{C}$ - classes
- Local robustness property, $\boldsymbol{x}$ given image, [Tjeng V, et. al, 2017]:
  $-\forall \boldsymbol{x'}.dist(\boldsymbol{x}, \boldsymbol{x'}) \leq \epsilon \implies N(\boldsymbol{x}) = N(\boldsymbol{x'})$

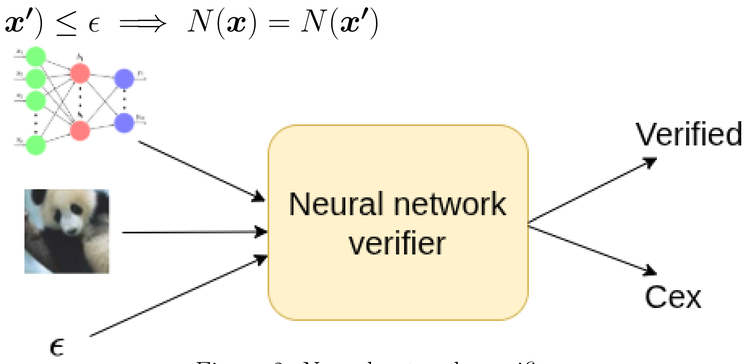
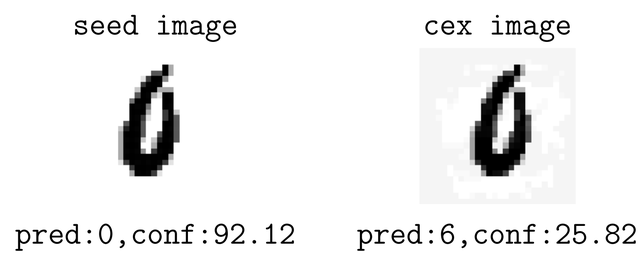


Figure 2: Neural networks verifier

- State-of-the-arts (SOTA) neural networks verifiers: $\alpha\beta$-CROWN, PyRAT, Marabou.
- SOTA ignore the classification confidence.

3. Confidence-based Robustness

$\textsc{Conf}(N(\boldsymbol{x}), c)$: Define the confidence on class c with input $\boldsymbol{x}$, using SOFTMAX function.



Relaxed Robustness: Avoid low confidence counterexamples.

- $\forall \boldsymbol{x'}.dist(\boldsymbol{x}, \boldsymbol{x'}) \leq \epsilon \implies N(\boldsymbol{x}) = N(\boldsymbol{x'}) \vee \textsc{Conf}(\boldsymbol{x'}, c) \leq th$

Other Variations:

Strong Robustness: High confidence decision should remain high.
Smoothness: Confidence should not change drastically.

Need of a unifying framework to capture all imagined confidence-based properties?

4. Unifying Framework for Property Variations

Grammar: Let $\boldsymbol{y} = y_1, y_2, \ldots, y_m = N(x)$

$$\begin{aligned}
\text{LE} &::= c_1 y_1 + c_2 y_2 + \cdots + c_m y_m + b, \forall i \in [m], c_i, b \in \mathbb{R} \\
\text{LC} &::= \text{LE} > 0 \mid \text{LE} \geq 0 \\
\text{CC} &::= \textsc{conf}(\boldsymbol{y}, t) \bowtie b, t \in [m], b \in \mathbb{R}^+, \bowtie \in \{\leq, <, \geq, >\} \\
\text{PC} &::= \text{PC} \wedge \text{PC} \mid \text{PC} \vee \text{PC} \mid \text{CC} \mid \text{LC} \mid \neg \text{LC}
\end{aligned}$$

- PC represents the set of post-conditions.
- Need to approximate the CONF.

Approx of $\textbf{conf}(\bar{y}, t) \leq b$

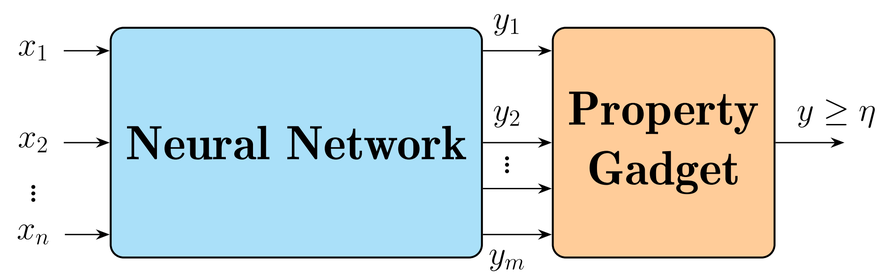
Let $y_{t'} = \max_{i=1, i \neq t}^{m}(y_i)$ and $\delta = -\ln\left(\frac{100}{b} - 1\right)$.

Lemma. If $y_t \leq y_{t'} + \delta$, then $\textsc{conf}(\bar{y}, t) \leq b$; otherwise, $\textsc{conf}(\bar{y}, t) \geq \frac{100}{1+(m-1)e^{-\delta}}$.

$$y_t \leq y_{t'} + \delta \iff \bigvee_{j=1, j \neq t}^{m} y_t \leq y_j + \delta$$

Each post condition can only be encoded manually to constraints-based solver.

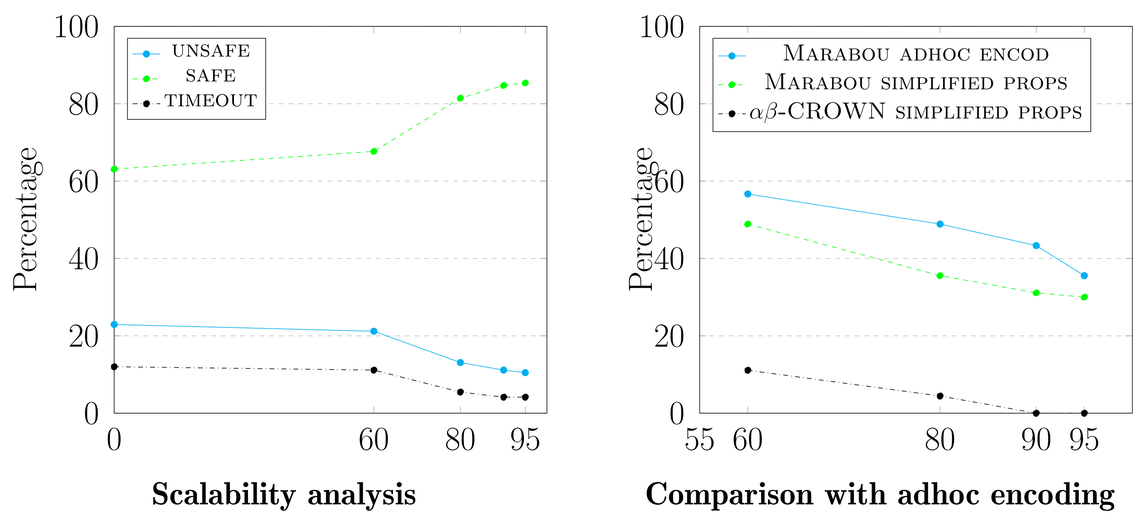
5. Layered Encoding



- Converts property to gadget and append to the network.
- Property gadget use only RELU activations.
- Property is simplified to $y \geq \eta$.
- Enables the use of SOTA, i.e. $\alpha\beta$-CROWN.

Property to gadget is a non-trivial conversion if $\bigwedge$ and $\bigvee$ both occurs in the property

6. Experiments



- Datasets: MNIST, CIFAR-10, GTSRB, and Image-1k. Sourced from VNNCOMP.
- Network size: 512 to 11.16M relus.
- Tool: $\alpha\beta$-CROWN.

Conclusion

- Existing works ignore the classification confidence.
- Introduced novel definition of robustness – Relaxed robustness.
- We introduced a grammar that captures a wide range of robustness properties.
- Provided a layered encoding that enabled the use of SOTA.
- Our approach scales on large networks.

References