POPQORN: Quantifying Robustness of Recurrent Neural Networks

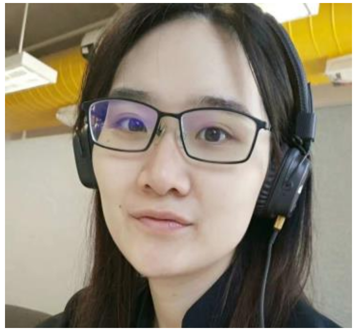
Ching-Yun Ko *^, Zhaoyang Lyu *, Tsui-Wei Weng, Luca Daniel, Ngai Wong, Dahua Lin

*** Equal Contribution** **^ Presenter**

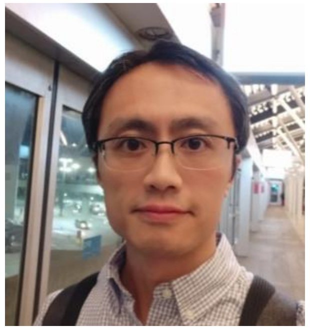
★ **arXiv:** https://arxiv.org/abs/1905.07387

★ **github:** https://github.com/ZhaoyangLyu/POPQORN

A joint research by








Should technology be banned?

Facebook translates 'good morning' into 'attack them', leading to arrest.

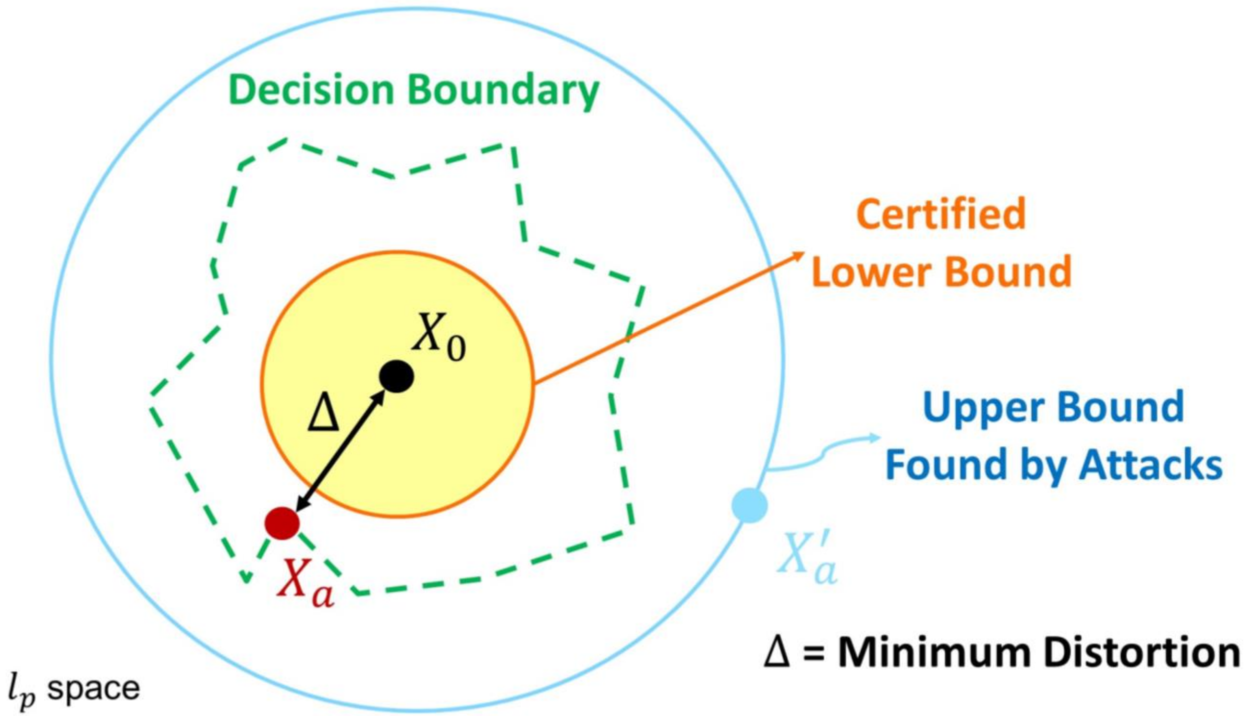
Google Translate got a Mexican native arrested and redeemed.

San Francisco banned facial-recognition technology.

Concerns are rooted not just in a long national history of racially-biased state surveillance, but in the potential inaccuracy of facial recognition technology.

To justify the use of neural networks, the first step is to realize **neural networks are fragile**.

Our goal is to certify bounds around an input such that the top-1 classification result is consistent within the balls.



I.e. we want to provide a certified lower bound of the **minimum adversarial distortion**

Evaluating RNN robustness

Method	Application	Architecture	Certificate
FGSM (Papernot et al., 2016)	NLP	LSTM	✗
(Gong & Poellabauer, 2017)	Speech	WaveRNN (RNN/ LSTM)	✗
Houdini (Ciss´e et al., 2017)	Speech	DeepSpeech-2 (LSTM)	✗
(Jia & Liang, 2017)	NLP	LSTM	✗
(Zhao et al., 2018)	NLP	LSTM	✗
(Ebrahimi et al., 2018)	NLP	LSTM	✗
C&W (Carlini & Wagner, 2018)	Speech	DeepSpeech (LSTM)	✗
Seq2Sick (Cheng et al., 2018)	NLP	Seq2seq(LSTM)	✗
CLEVER (Weng et al., 2018b)	CV/ NLP/ Speech	RNN/LSTM/GRU	✗
POPQORN (This work)	**CV/ NLP/ Speech**	**RNN/LSTM/GRU**	✓

POPQORN provides safeguarded lower bounds!

Safeguarded lower bounds

Network architectures	**Certification algorithms**
MLP + ReLU activation	Fast-Lin[1], DeepZ[2], Neurify[3]
MLP + general activation	CROWN [4], DeepPoly[5]
CNN (pooling, resnet)	CNN-Cert [6]
RNN, LSTM, GRU	**POPQORN (This work)**

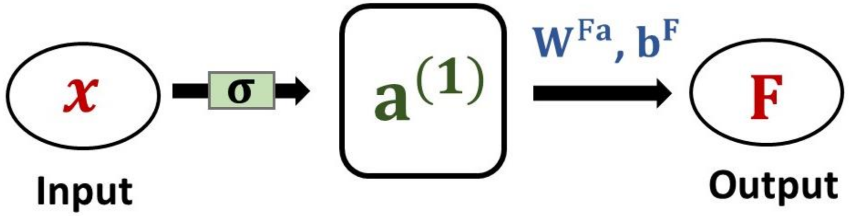
Applications: Video streams, Texts, Audio…

[1] Weng etal, “Toward Fast Computation of Certified Robustness for ReLU Networks”, ICML’18
[2] Singh etal, “Fast and Effective Robustness Certification”, NeurIPS’18
[3] Wang etal, “Efficient Formal Safety Analysis of Neural Networks”, NeurIPS’18
[4] Zhang etal, “Efficient Neural Network Robustness Certification with General Activation Functions”, NeurIPS’18
[5] Singh etal, “Fast and effective robustness certification”, NeurIPS'18
[6] Boopathy etal, “CNN-Cert: An Efficient Framework for Certifying Robustness of Convolutional Neural Networks”, AAAI’19

From MLP/ CNN to LSTM/ GRU

General activations: ReLU, tanh, sigmoid, etc

$$a^{(k)} = \sigma(W^{(k)}a^{(k-1)} + b^{k})$$



Coupled nonlinearity:
cross-nonlinearity

Input gate: $\mathbf{i}^{(k)} = \sigma(\mathbf{W}^{ix}\mathbf{x}^{(k)} + \mathbf{W}^{ia}\mathbf{a}^{(k-1)} + \mathbf{b}^{i});$

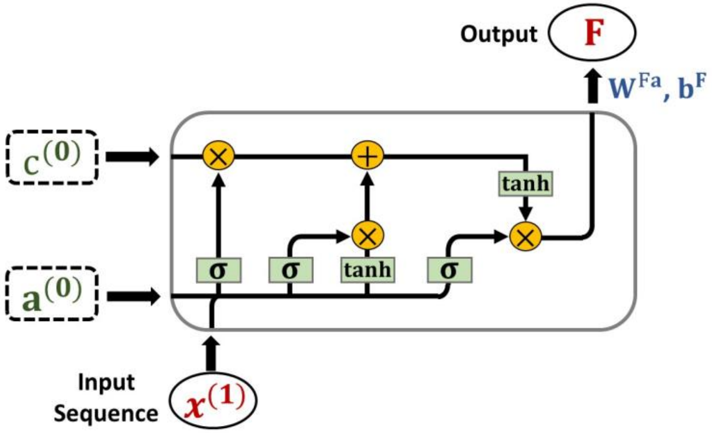
Forget gate: $\mathbf{f}^{(k)} = \sigma(\mathbf{W}^{fx}\mathbf{x}^{(k)} + \mathbf{W}^{fa}\mathbf{a}^{(k-1)} + \mathbf{b}^{f});$

Cell gate: $\mathbf{g}^{(k)} = \tanh(\mathbf{W}^{gx}\mathbf{x}^{(k)} + \mathbf{W}^{ga}\mathbf{a}^{(k-1)} + \mathbf{b}^{g});$

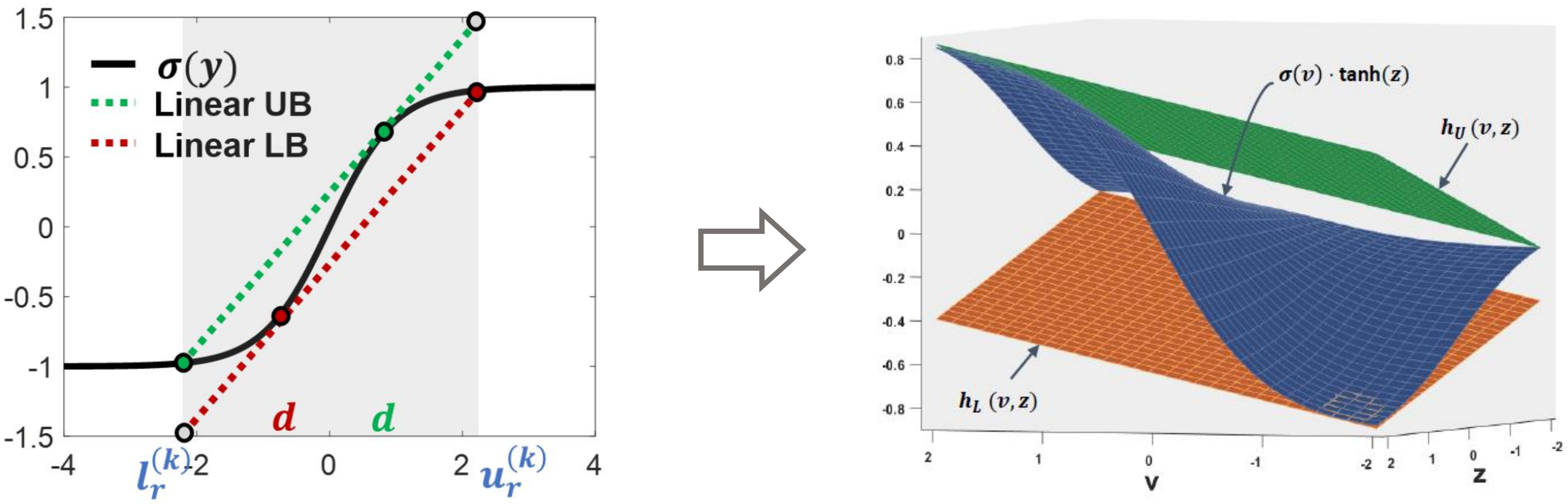
Output gate: $\mathbf{o}^{(k)} = \sigma(\mathbf{W}^{ox}\mathbf{x}^{(k)} + \mathbf{W}^{oa}\mathbf{a}^{(k-1)} + \mathbf{b}^{o});$

Cell state: $\mathbf{c}^{(k)} = \mathbf{f}^{(k)} \odot \mathbf{c}^{(k-1)} + \mathbf{i}^{(k)} \odot \mathbf{g}^{(k)};$

Hidden state: $\mathbf{a}^{(k)} = \mathbf{o}^{(k)} \odot \tanh(\mathbf{c}^{(k)}).$



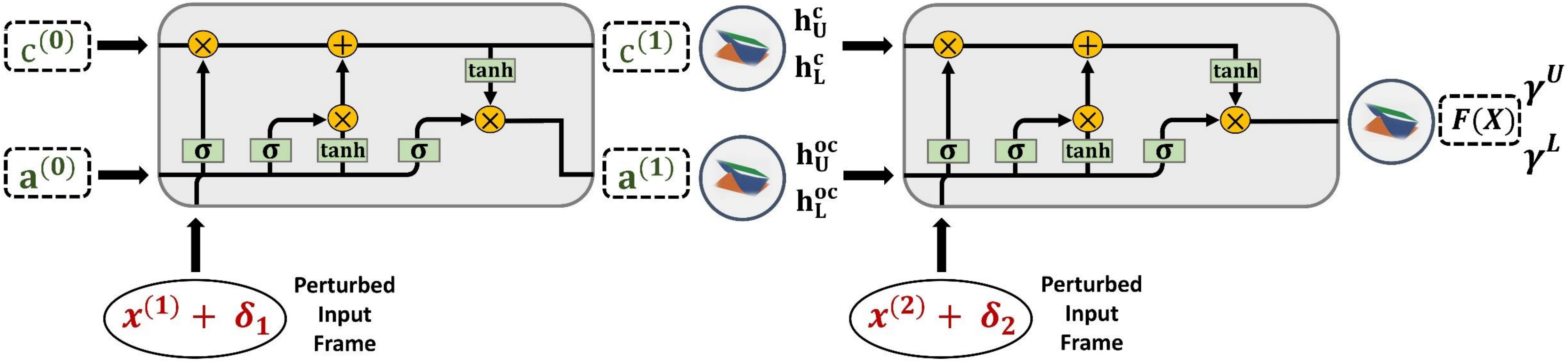
Tackling the "cross-nonlinearity"



Use 2D planes to bound the "cross-nonlinearity" specifically in LSTMs/ GRUs.

Basic ideas

1. Compute the lower and upper bounds of the output units given a perturbed input sequence $X + \delta$, where $||\delta||_p \leq \epsilon$.
2. If the lower bound of the true label output unit γ_i^L is larger than the upper bounds of all other output units $\gamma_j^U (j \neq i)$, we can certify that the classification result won't change within this l_p ball.



Theoretical Results

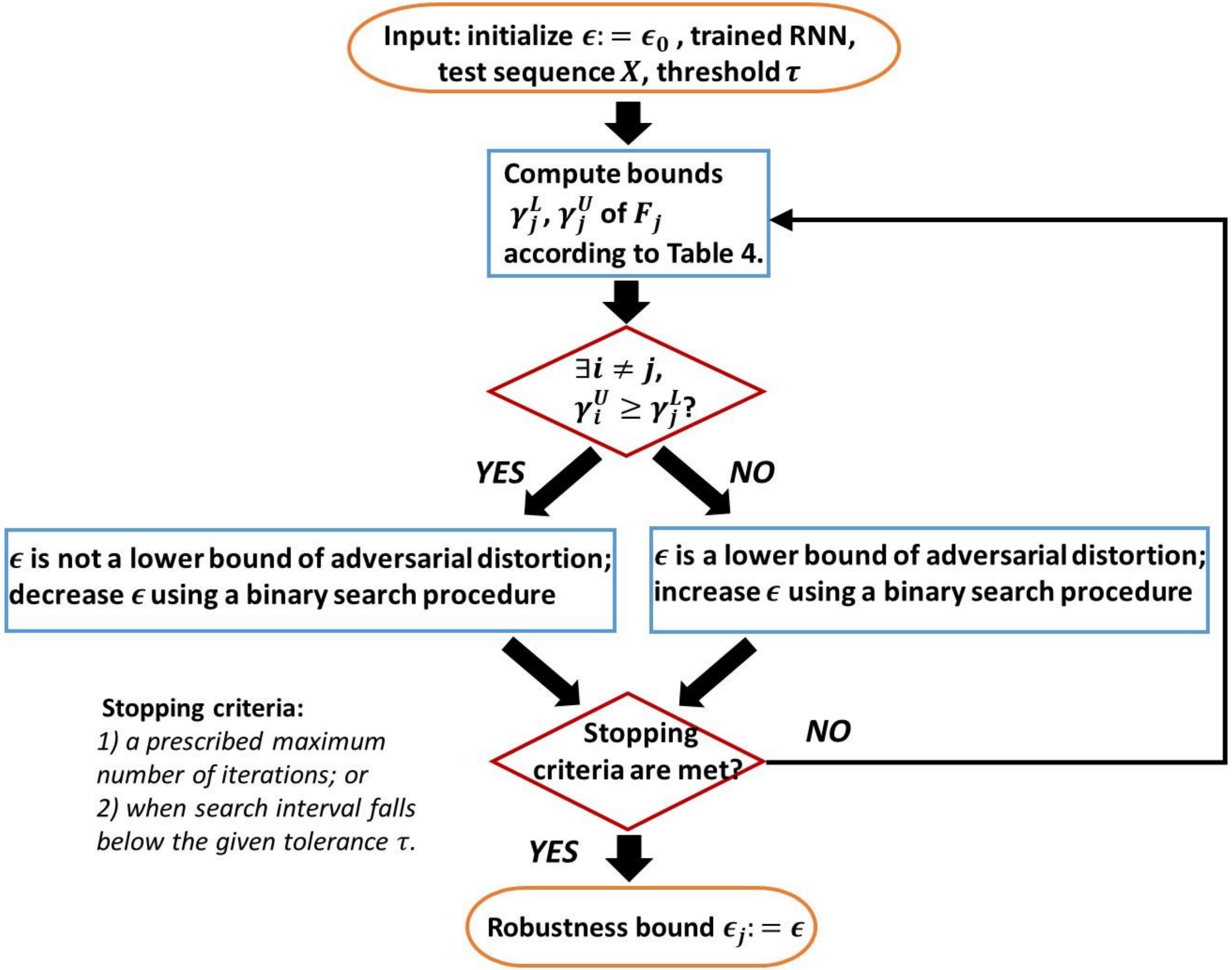
We can write out the lower and upper bounds of output units as functions of radius ϵ.

($X + \delta$, where $||\delta||_p \leq \epsilon$)

Certified robustness bounds for various RNNs

Networks	$\gamma_j^L \leq F_j \leq \gamma_j^U$	Closed-form formulas
Vanilla RNN	Upper bounds γ_j^U	$\mathbf{\Lambda}_{j,:}^{(0)}\mathbf{a}^{(0)} + \sum_{k=1}^m \epsilon \|\mathbf{\Lambda}_{j,:}^{(k)}\mathbf{W}^{ax}\|_q + \sum_{k=1}^m \mathbf{\Lambda}_{j,:}^{(k)}\mathbf{W}^{ax}\mathbf{x}_0^{(k)} + \sum_{k=1}^m \mathbf{\Lambda}_{j,:}^{(k)}(\mathbf{b}^a + \mathbf{\Delta}_{:,j}^{(k)}) + \mathbf{b}_j^F$
	Lower bound γ_j^L	$\mathbf{\Omega}_{j,:}^{(0)}\mathbf{a}^{(0)} - \sum_{k=1}^m \epsilon \|\mathbf{\Omega}_{j,:}^{(k)}\mathbf{W}^{ax}\|_q + \sum_{k=1}^m \mathbf{\Omega}_{j,:}^{(k)}\mathbf{W}^{ax}\mathbf{x}_0^{(k)} + \sum_{k=1}^m \mathbf{\Omega}_{j,:}^{(k)}(\mathbf{b}^a + \mathbf{\Theta}_{:,j}^{(k)}) + \mathbf{b}_j^F$
LSTM	Upper bounds γ_j^U	$\tilde{\mathbf{W}}_{U,j,:}^{a(1)}\mathbf{a}^{(0)} + \mathbf{\Lambda}_{\Delta,j,:}^{fc(1)}\mathbf{c}^{(0)} + \sum_{k=1}^m \epsilon \|\tilde{\mathbf{W}}_{U,j,:}^{x(k)}\|_q + \sum_{k=1}^m \tilde{\mathbf{W}}_{U,j,:}^{x(k)}\mathbf{x}_0^{(k)} + \sum_{k=1}^m \tilde{\mathbf{b}}_{U,j}^{(k)} + \mathbf{b}_j^F$
	Lower bound γ_j^L	$\tilde{\mathbf{W}}_{L,j,:}^{a(1)}\mathbf{a}^{(0)} + \mathbf{\Omega}_{\Theta,j,:}^{fc(1)}\mathbf{c}^{(0)} - \sum_{k=1}^m \epsilon \|\tilde{\mathbf{W}}_{L,j,:}^{x(k)}\|_q + \sum_{k=1}^m \tilde{\mathbf{W}}_{L,j,:}^{x(k)}\mathbf{x}_0^{(k)} + \sum_{k=1}^m \tilde{\mathbf{b}}_{L,j}^{(k)} + \mathbf{b}_j^F$
GRU	Upper bounds γ_j^U	$\tilde{\mathbf{W}}_{U,j,:}^{a(1)}\mathbf{a}^{(0)} + \sum_{k=1}^m \epsilon \|\tilde{\mathbf{W}}_{U,j,:}^{x(k)}\|_q + \sum_{k=1}^m \tilde{\mathbf{W}}_{U,j,:}^{x(k)}\mathbf{x}_0^{(k)} + \sum_{k=1}^m \tilde{\mathbf{b}}_{U,j}^{(k)} + \mathbf{b}_j^F$
	Lower bound γ_j^L	$\tilde{\mathbf{W}}_{L,j,:}^{a(1)}\mathbf{a}^{(0)} - \sum_{k=1}^m \epsilon \|\tilde{\mathbf{W}}_{L,j,:}^{x(k)}\|_q + \sum_{k=1}^m \tilde{\mathbf{W}}_{L,j,:}^{x(k)}\mathbf{x}_0^{(k)} + \sum_{k=1}^m \tilde{\mathbf{b}}_{L,j}^{(k)} + \mathbf{b}_j^F$

POPQORN: Robustness Quantification Algorithm

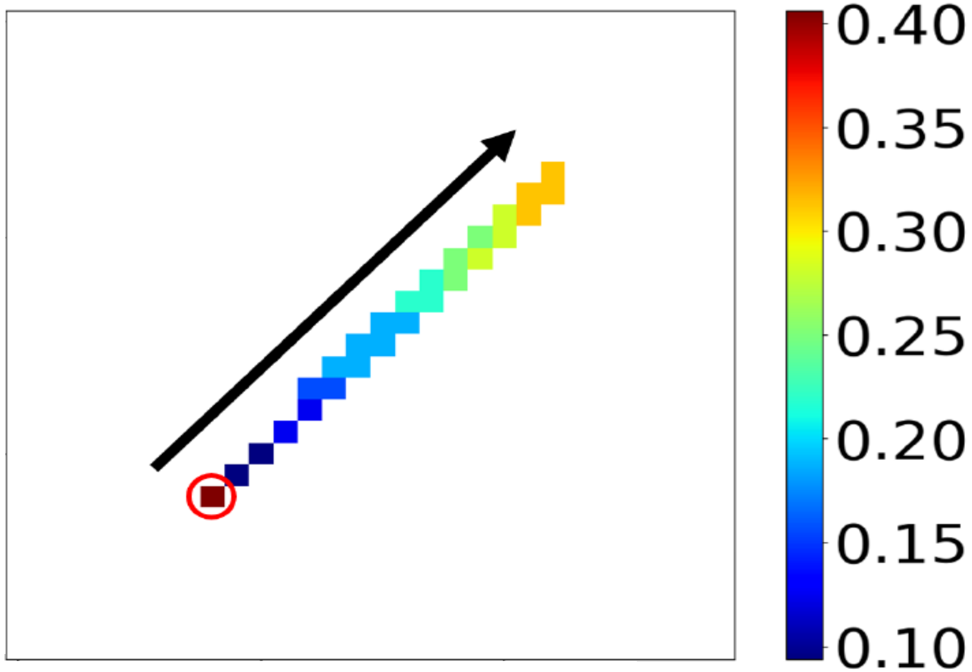


Steps in computing bounds for recurrent neural networks.

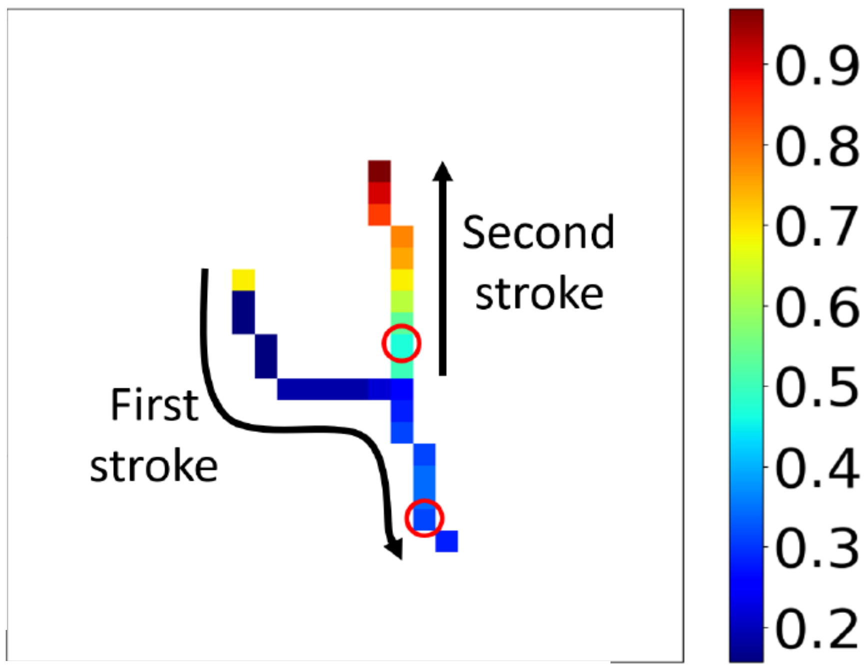
Experiment 1: Sequence MNIST

We compute the untargeted POPQORN bound on each time step, and the stroke with minimal bounds are the most **sensitive** ones.

- The starting point of one's stroke is **not** important
- Points in the back can tolerate larger perturbations



digit "1"



digit "4"

Experiment 2: Question Classification

We compute the untargeted POPQORN bound on one single input frame, and call the words with minimal bounds *sensitive words*

``ENTY" (entity), ``LOC" (location)

Example	**What**	**is**	**the**	**name**	**of**	***Roy***	***Roger***	***'s***	***dog***	***?***
ENTY	0.34	0.50	0.53	**0.27**	0.39	**0.19**	**0.32**	1.02	0.67	0.93

Example	**What**	**is**	**the**	**fourth**	**highest**	**mountain**	***in***	***the***	***world***	***?***
LOC	**0.47**	0.75	0.95	0.67	**0.48**	**0.55**	1.19	1.11	0.85	0.91

Experiment 3: News Title Classification

Example	**Samsung**	**to**	**launch**	**galaxy**	**s**	**sequel**	**In**	**south**	**korea**	**in**	**late**	**april**
Sci&Tech	**0.42**	0.73	0.55	**0.46**	**0.52**	0.57	0.80	0.66	0.67	0.85	0.72	0.81

Example	**3**	**journalists**	**kidnapped**	**in**	**afghanistan**	**are**	**set**	**free**
World	0.45	**0.43**	**0.42**	0.73	**0.39**	0.65	0.60	0.55

Conclusions

POPQORN has three important advantages:

1) ***Novel*** - it is a general and the first work to provide a robustness evaluation for RNNs with robustness guarantees.

2) ***Effective*** - it can handle complicated LSTMs and GRUs with challenging coupled nonlinearities.

3) ***Versatile*** - it can be widely applied in computer vision, natural language processing, and speech recognition.

POPQORN: Quantifying Robustness of Recurrent Neural Networks

- ★ **poster:** Tue Jun 11 @ Pacific Ballroom #67
- ★ **arXiv:** https://arxiv.org/abs/1905.07387
- ★ **github:** https://github.com/ZhaoyangLyu/POPQORN

Follow our project!