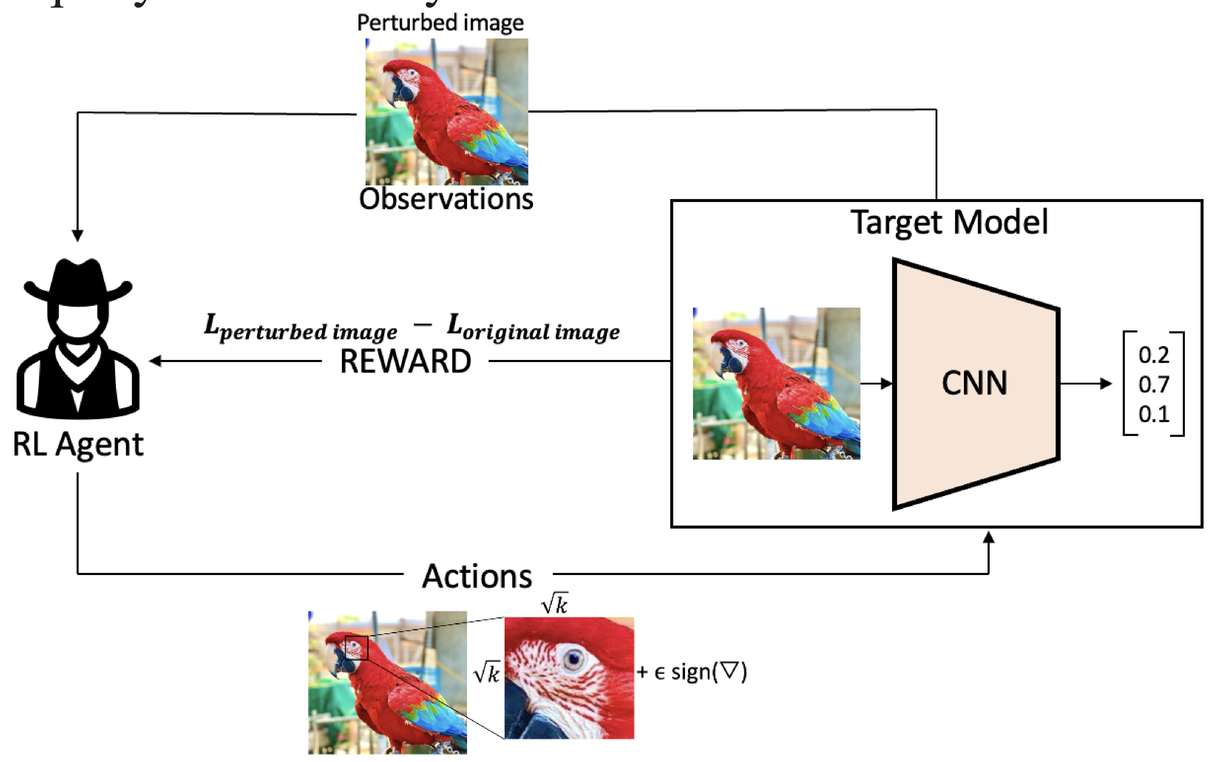
Deep Query Attacks: A Reinforcement Learning Approach

Mohammad Alomrani, Farhan Rahman, Rahavi Selvarajan
University of Toronto

Problem Definition and Contribution

Goal: Attacking a Black-box Convolutional Neural Network in the L-∞ setting using reinforcement learning in a query efficient way.



Motivations:

- Recent black-box query-based attacks overcome the need to training surrogate models as they require **no access to a representative dataset** or any **knowledge of the target model architecture**.
- Existing query attacks either use derivative free methods [1] or finite difference approximations (differential querying) [2] to generate adversarial examples.
- A trade-off exists between the success rate and the number of queries.
- Query attacks do not *learn* or utilize previous historical knowledge to decrease the number of future queries.

Key Contributions:

- We propose a **reinforcement learning framework** which combines **differential querying** and **structured grouping** (tiling) of pixels.
- Our proposed method utilizes historical knowledge from previous queries in the training stage to learn more query-efficient attack strategies.
- The RL agent only requires an attack dataset for training which need not be representative of the training dataset of the target model.

Attack Model:

- **Query-level access** to a black-box target model to obtain logits given an input.
- No rate limiting.
- No knowledge of the model's training dataset, architecture, or algorithms.

Methodology

Main idea:
Instead of perturbing one pixel at a time, we utilize the **tiling** approach [1] where **pixels are grouped into disjoint tiles** and pixels in the same tile are perturbed with the same gradient signal.

To train RL algorithms, the problem of generating adversarial examples must be formulated as a Markov Decision Process (MDP):

- **State:** The state S at timestep t is the current perturbed image $\hat{X}$, the original image X, and the current loss L.
- **Action:** At each timestep t, the agent will pick a tile of size $\sqrt{k} \times \sqrt{k}$ to be perturbed with the estimated gradient sign, obtained via *finite difference* approximation.
- **Reward Function:** The reward is defined as the change in the loss on the original input X and the current input $\hat{X}$. Consequently, the cumulative reward R at the terminal state is the total change in the loss after all perturbations.

$$R = L(\hat{X}) - L(X) \tag{1}$$

Algorithm 1 RL Agent in Action

Input: $X, k, \epsilon, L(\cdot), \pi$;
Output: Perturbed image $\hat{X}$;
$\hat{X} = X$
Split $\hat{X}$ into disjoint tiles T_i of size $\sqrt{k} \times \sqrt{k}$
for $j \leftarrow \lceil \frac{d}{k} \rceil$ **do**
  The RL agent π selects a tile of pixels T_i
  Initialize v such that $v_l = 1$ iff $l \in T_i$
  $\forall l \in T_i$, let $\nabla_{T_i} L(\hat{X}) = \frac{L(\hat{X}+\delta v) - L(\hat{X}-\delta v)}{2\delta}$
  $\hat{X} = \hat{X} + \nabla_{T_i} L(\hat{X})$

In the end, all tiles are perturbed. The agent is only concerned about the order in which the tiles are perturbed such that the image becomes adversarial early on.

Experiments & Results

Experiments:

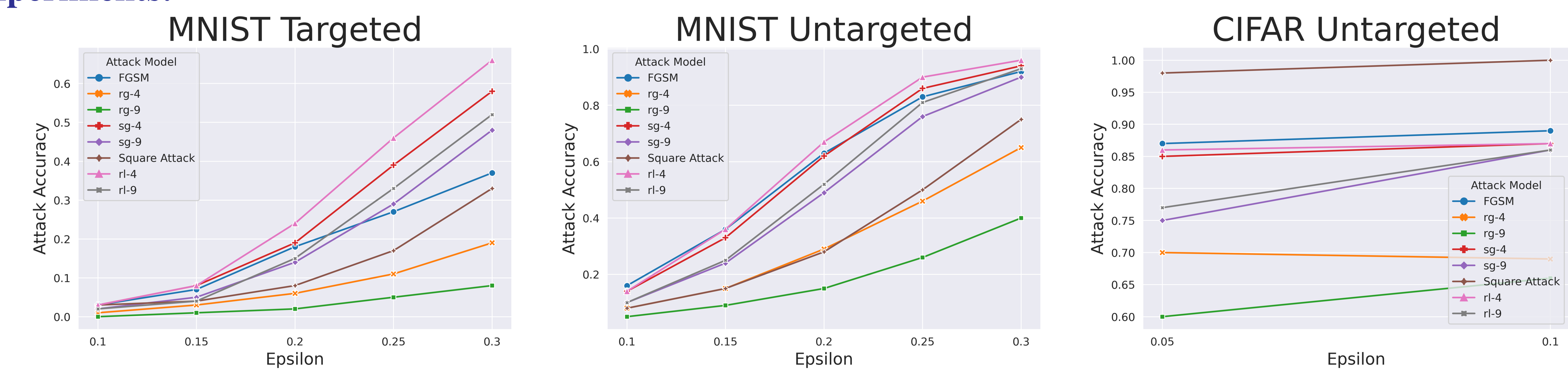


Figure 1: Attack accuracy for different values of ϵ for different methods on MNIST and CIFAR10. Here FGSM - Fast Gradient Sign Attack; rg-k - random grouping of k pixels; sg-k - structured-grouping i.e iteratively perturb all tiles of size k from top to bottom; rl-k - RL picks tiles of size k to be perturbed

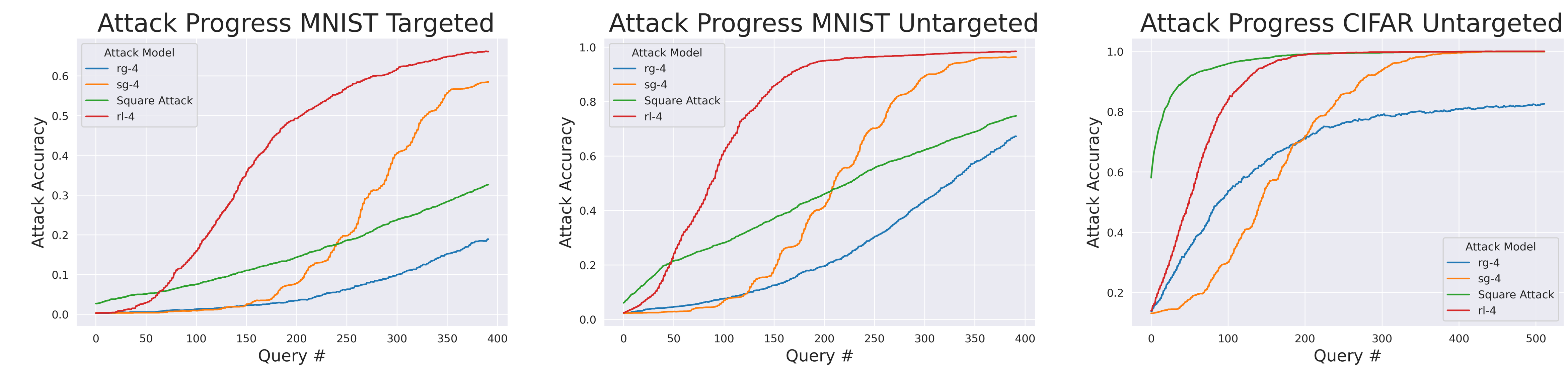


Figure 2: Attack progress over queries for different methods on MNIST and CIFAR10

ϵ	Untargeted Attack													
	rl-4		rl-9		Square-Attack		sg-9		sg-4		rg-9		rg-4	
	A	Q	A	Q	A	Q	A	Q	A	Q	A	Q	A	Q
0.1	**14.00**	139.80	10.00	**62.11**	7.95	169.18	10.09	111.56	13.69	228.97	4.95	100.42	7.84	233.10
0.15	**35.64**	148.73	25.49	**70.78**	14.55	165.15	23.90	118.19	32.99	238.84	8.65	99.498	15.25	241.55
0.2	**67.39**	140.35	51.99	**67.63**	27.65	182.99	49.14	123.15	61.65	229.19	14.79	108.55	29.19	256.40
0.25	**90.34**	116.85	81.30	**61.84**	49.55	187.82	75.95	119.33	85.99	229.19	26.15	113.54	46.34	261.70
0.3	**96.19**	95.07	92.90	**52.70**	74.75	165.59	90.49	111.42	94.09	210.57	39.70	117.71	64.99	256.34

Table 1: Performance of various methods for different values of ϵ on MNIST dataset

ϵ	Untargeted Attack					
	rl-4		rl-9		Square-Attack	
	A	Q	A	Q	A	Q
0.05	85.74	104.09	76.84	**70.40**	**97.5**	84.93
0.1	86.84	65.62	86.30	45.01	**99.95**	**19.04**

Table 2: Performance of various methods for different values of ϵ on CIFAR10 dataset

Observations:

- The RL agent achieves the best trade-off between number of queries (Q) and attack accuracy (A) for MNIST.
- RL agent is competitive with the white-box FGSM attack.
- RL agent is competitive with Square Attack on CIFAR-10 even though we use the same RL architecture for both datasets.
- Can we extend the framework to iterative FGSM attacks?

References
[1] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. *Square Attack: a query-efficient black-box adversarial attack via random search* arXiv:1912.00049v3, 2020.

[2] Arjun Nitin Bhagoji1, Warren He2, Bo Li3,Dawn Song2. *Practical Black-box Attacks on Deep Neural Networks using Efficient Query Mechanisms* ECCV, 2018.