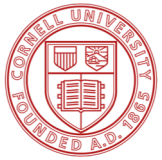
Adversarial Sequential Decision Making

Goran Radanović, Adish Singla, Wen Sun, Xiaojin Zhu

International Joint Conference on AI (IJCAI) 2022

MAX PLANCK INSTITUTE
FOR SOFTWARE SYSTEMS



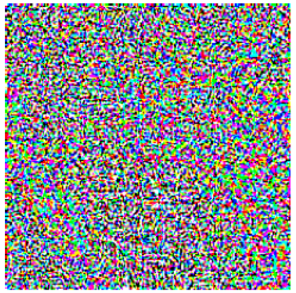
25th July, 2022

Adversarial Attacks on AI

$\boldsymbol{x}$

"panda"
57.7% confidence

$+ .007 \times$



$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"nematode"
8.2% confidence

$=$

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"gibbon"
99.3 % confidence

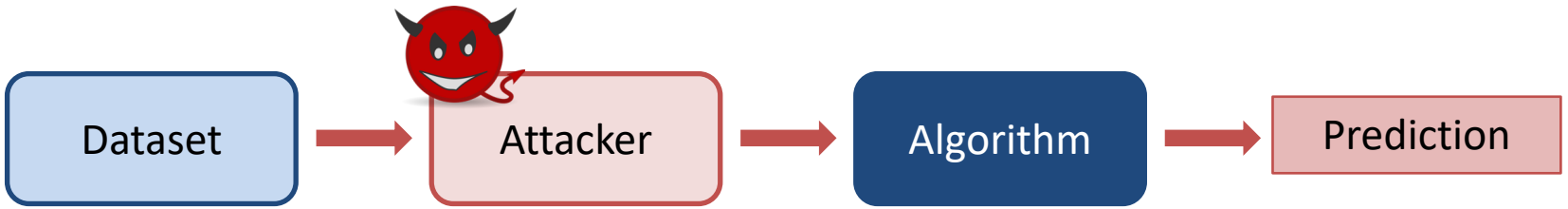
[Goodfellow et al., 2015]

[Sharif et al., 2016]

Adversarial Attacks on ML

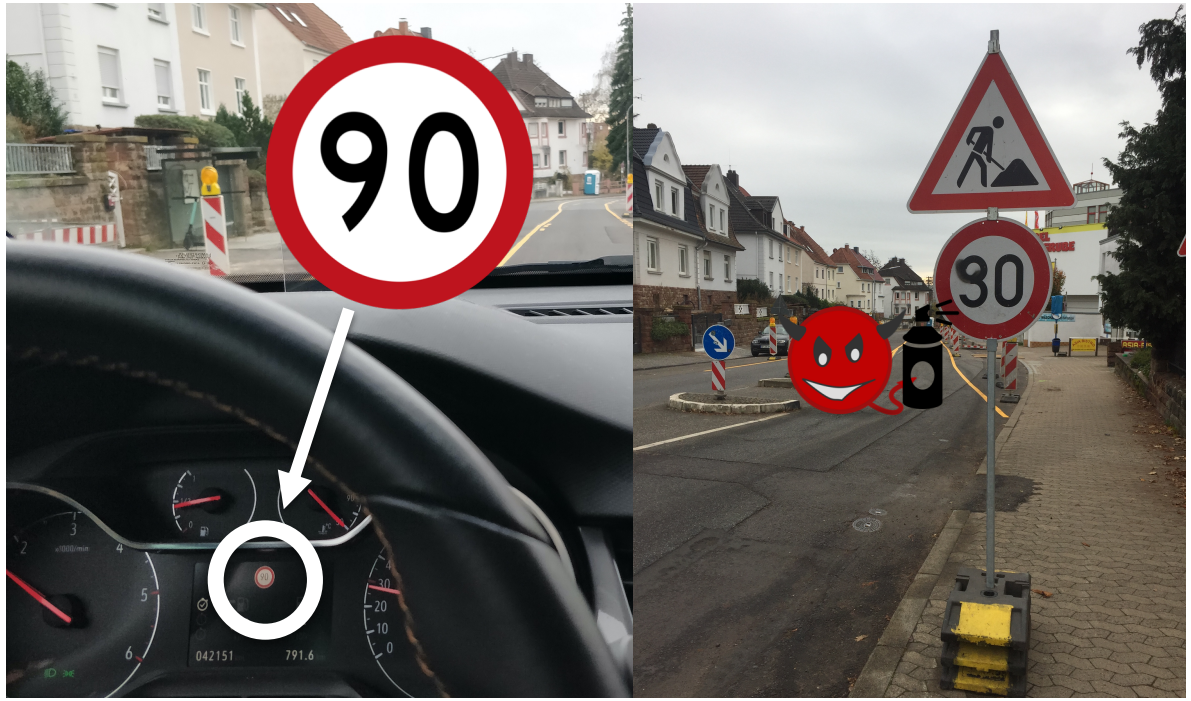
Learning to predict

Influencing prediction



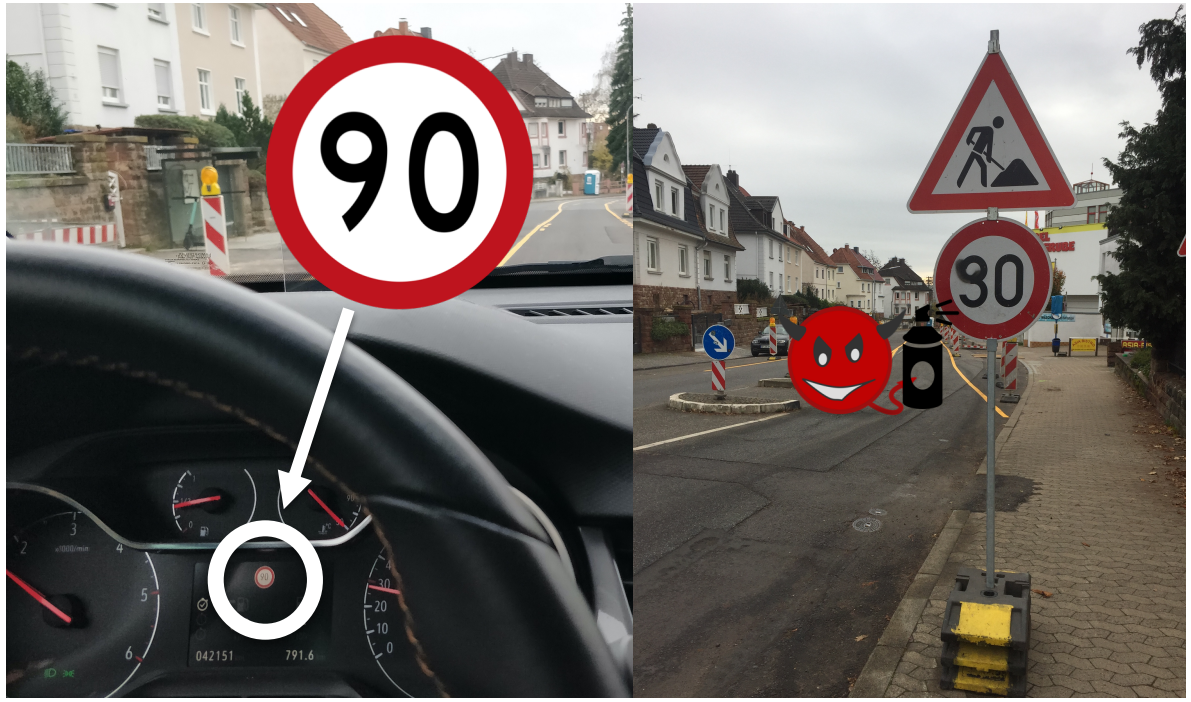
Accounting for Decisions

Attacks on Driving Systems

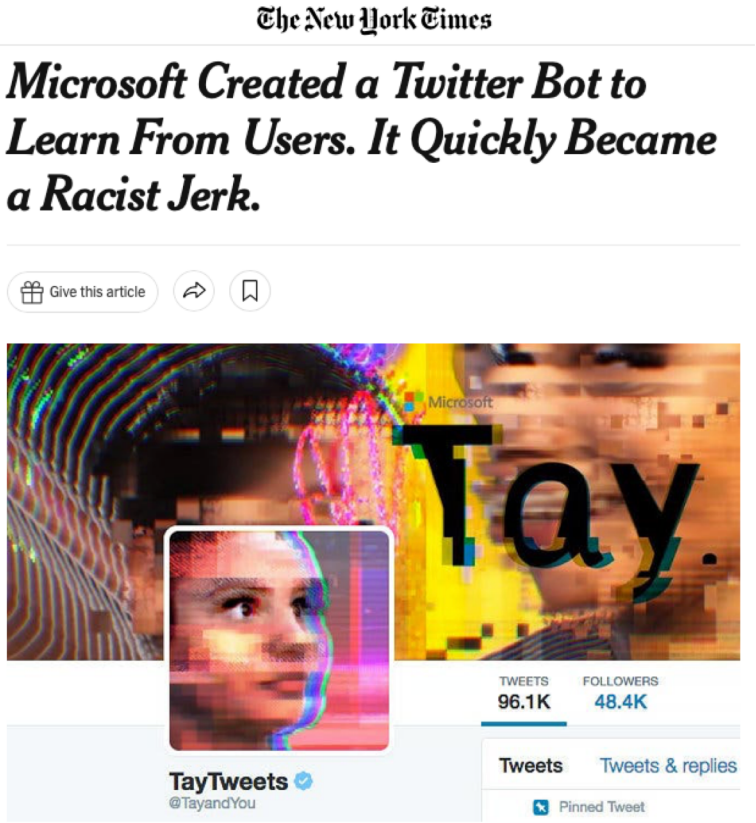


Accounting for Decisions

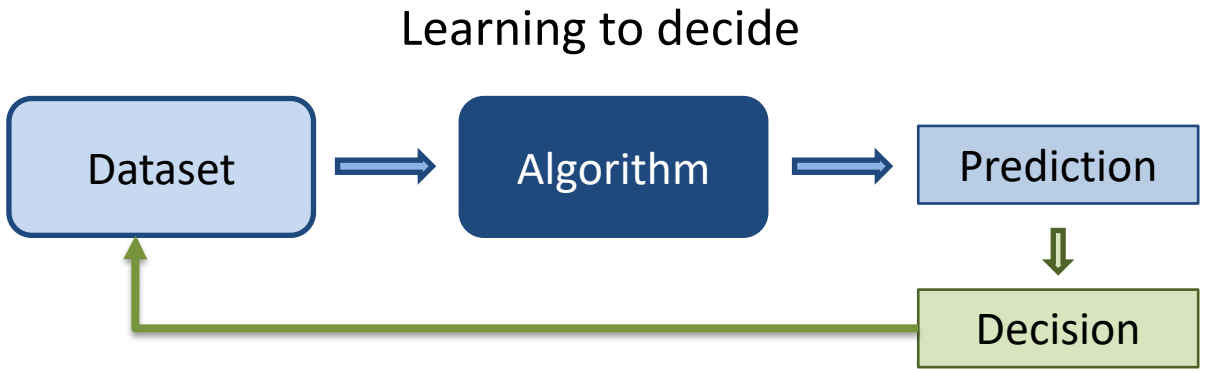
Attacks on Driving Systems

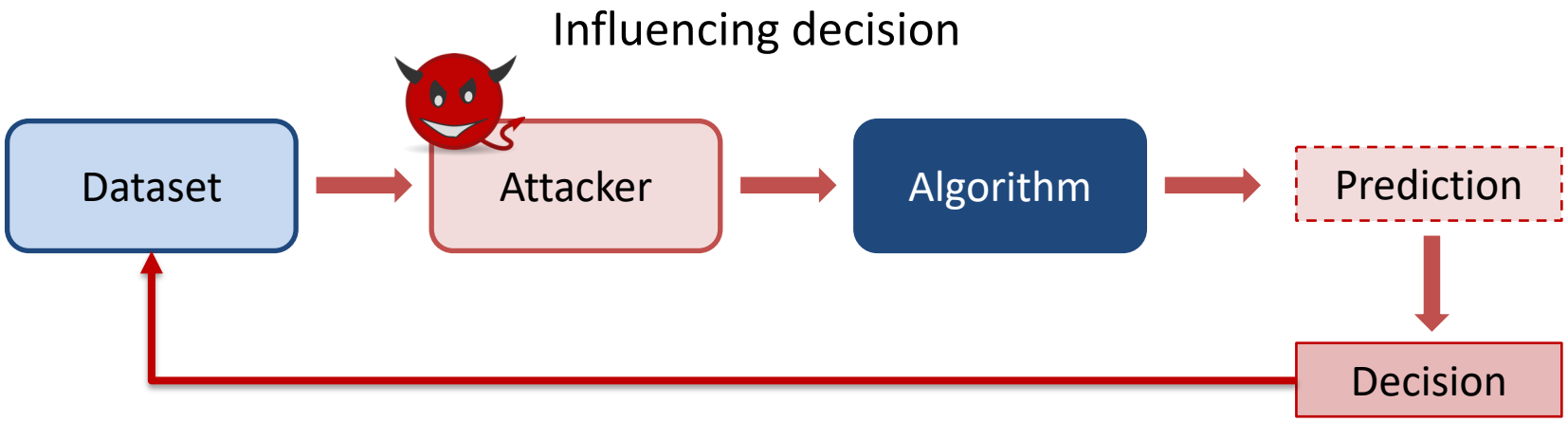


Attacks on Conversational AI



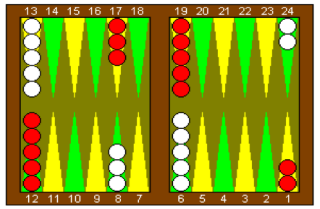
From Prediction to Decisions





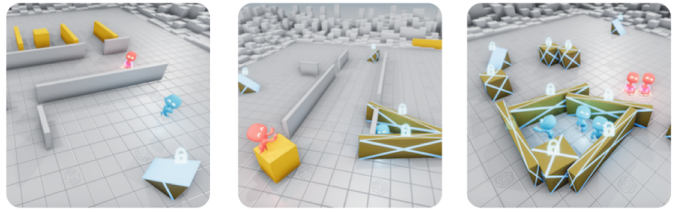
Credit to Dylan Foster for ML vs. Decision Making distinction. See link.

Trustworthy Decision Making

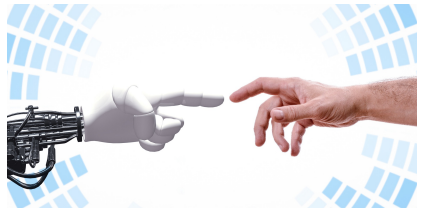


[Tesauro, 1995]

[Vynials et al., 2019]



[Baker et al., 2020]



Reinforcement Learning

Early works

Algorithmic foundations

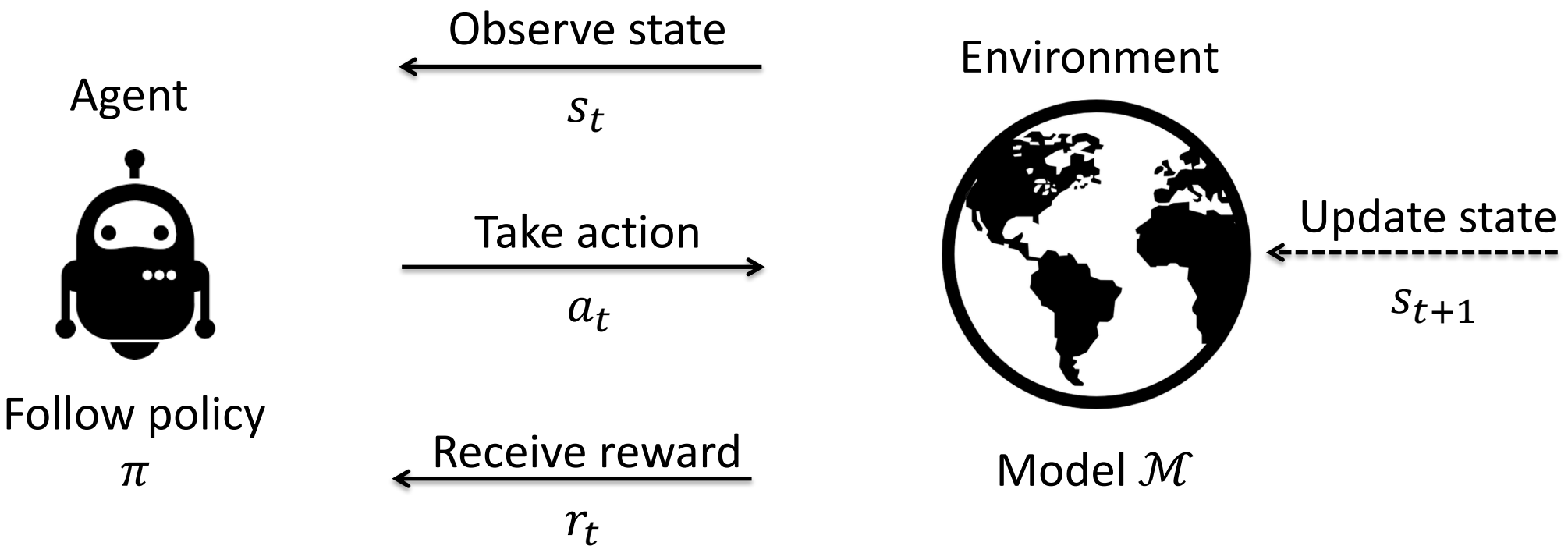
2010s-2020s

Scalable algorithms

Next grand challenge!

Trustworthy decision making

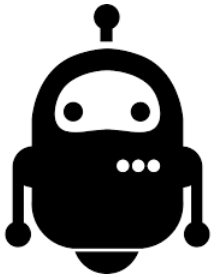
Sequential Decision Making



Maximize performance

Adversarial Sequential Decision Making

Agent



Observe state

Take action

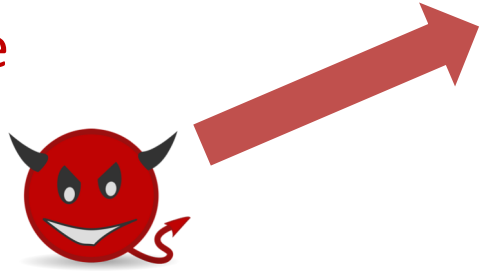
Receive reward

Environment

Update state

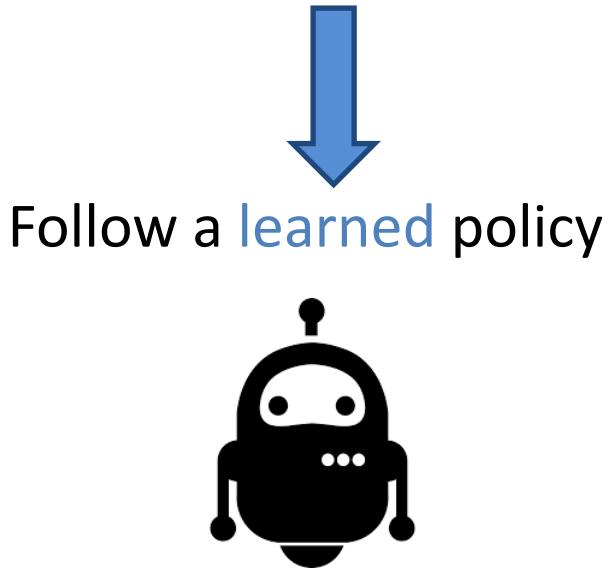
Model $\mathcal{M}$

Manipulate Interaction



Maximize performance

Attack Modalities: Test-Time Attacks



Follow a learned policy

Observe state

Take action

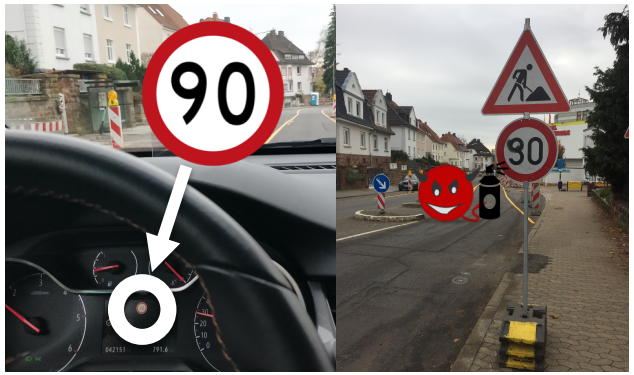
Receive reward

Environment

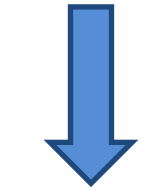
Update state

E.g. manipulate observations to minimize performance

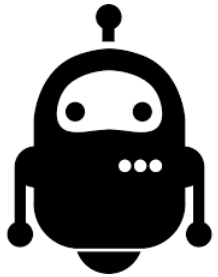
Example



Attack Modalities: Training-Time Attacks



Learn a policy



Observe state

Take action

Receive reward

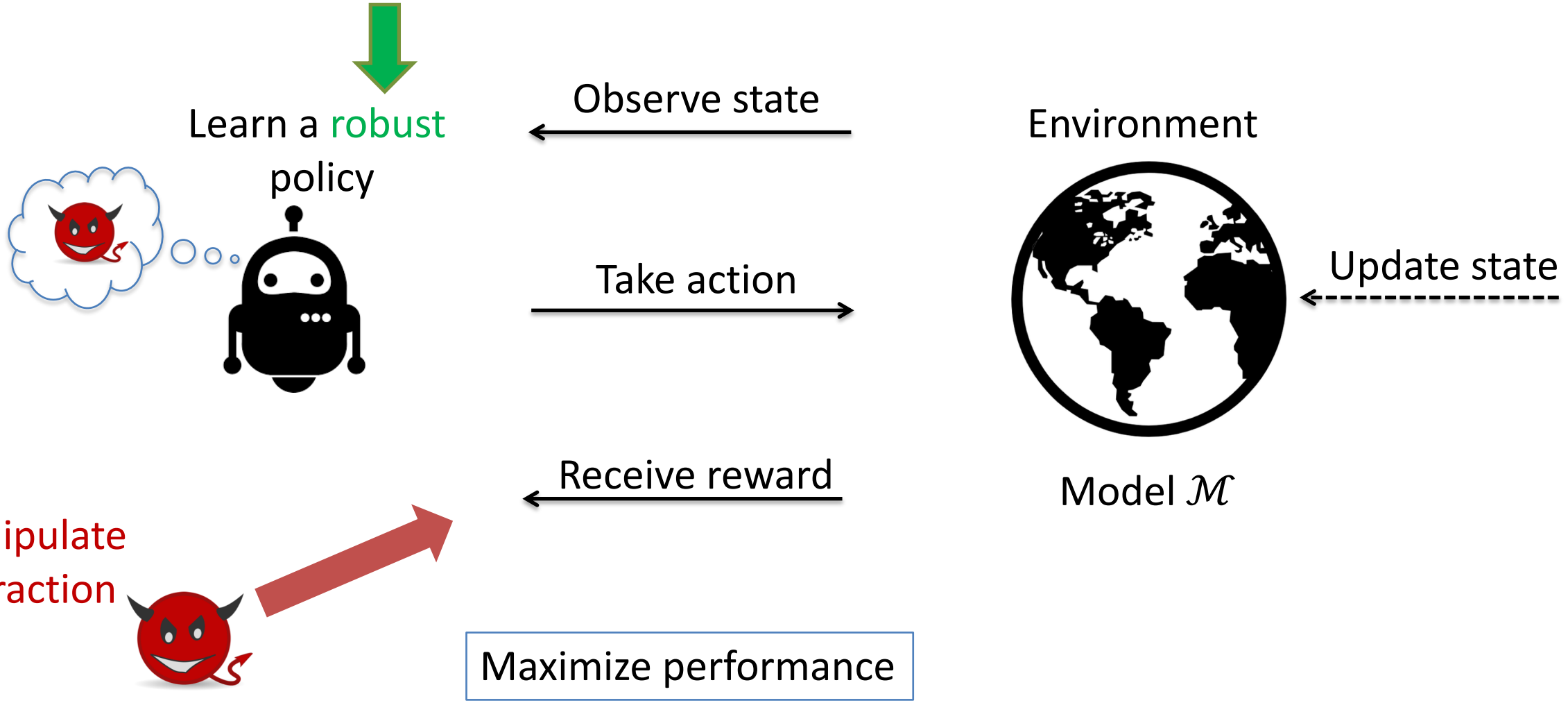
Environment

Update state

E.g., manipulate rewards
& transitions to force
a target policy

Example

Defenses Against Adversarial Attacks



Outline

- Preliminaries
- Test-time Attacks and Defenses in RL
- Training-time Attacks in RL
- Training-time Defenses in RL
- Adversarial Attacks in Multi-agent RL
- Concluding Remarks

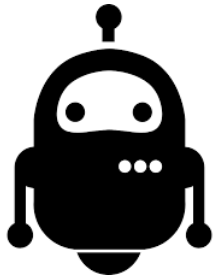
Outline

- **Preliminaries**
- Test-time Attacks and Defenses in RL
- Training-time Attacks in RL
- Training-time Defenses in RL
- Adversarial Attacks in Multi-agent RL
- Concluding Remarks

Markov Decision Processes

MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, \gamma, \mu)$

- $P: \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$
- $R: \mathcal{S} \times \mathcal{A} \to \mathbb{R}$
- $\gamma \in [0,1)$
- $\mu \in \Delta(\mathcal{S})$



Observe s_t

$a_t \sim \pi(\cdot\,|s_t)$

Receive $R(s_t, a_t)$

Update state

$s_{t+1} \sim P(\cdot\,|s_t, a_t)$

$s_0 \sim \mu(\cdot)$

- Stochastic stationary policy $\pi: \mathcal{S} \to \Delta(\mathcal{A})$

$$\max_\pi \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \,|\, s_0 \sim \mu, a_t \sim \pi(\cdot\,|s_t), s_{t+1} \sim P(\cdot\,|s_t, a_t)\right]$$

Value Functions

Value function $V^{\pi} : \mathcal{S} \rightarrow \mathbb{R}$

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) | \, s_0 = s, \pi\right]$$

Value Functions

Value function $V^{\pi} : \mathcal{S} \rightarrow \mathbb{R}$

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) | \, s_0 = s, \pi\right]$$

Note that the optimization problem is …

$$\max_{\pi} V^{\pi}(\mu) = \sum_{s} \mu(s) \cdot V^{\pi}(s)$$

Value Functions

Value function $V^{\pi} : \mathcal{S} \rightarrow \mathbb{R}$

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) | \, s_0 = s, \pi\right]$$

How do we find V^{π}?

$$V^{\pi}(s) = R\,(s, \pi) + \gamma \cdot \sum_{s'} P(s'|s, \pi) \cdot V^{\pi}\,(s')$$

Bellman equation

Value Functions

Value function $V^{\pi} : \mathcal{S} \rightarrow \mathbb{R}$

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) | \, s_0 = s, \pi\right]$$

State-action value function $Q^{\pi} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$

$$Q^{\pi}(s, a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) | \, s_0 = s, a_0 = a, \pi\right]$$

Value Functions

Value function $V^{\pi} : \mathcal{S} \rightarrow \mathbb{R}$

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) | \, s_0 = s, \pi\right]$$

Advantage function:

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s)$$

State-action value function $Q^{\pi} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$

$$Q^{\pi}(s,a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) | \, s_0 = s, a_0 = a, \pi\right]$$

Value Functions

Bellman optimality operator

$$(\mathcal{T}Q)(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) \max_{a'} Q(s',a')$$

If Q^* satisfies $\mathcal{T}Q^* = Q^*$, then

$$\pi^*(a|s) = 1.0 \text{ s.t. } a \in \text{argmax}_{a'}\, Q^*(s,a')$$

is optimal.

Finding Optimal Policy

- Planning in MDPs: P and R are given
  - Policy iteration: policy evaluation + policy improvement
  - Q-value iteration: calculate Q^*

- Reinforcement learning
  - **Policy gradient**
  - Q-learning: learn Q^*

Policy Gradient

- Parametric policy $\pi_\theta(a|s)$

- Gradient update rule: $\theta_{k+1} = \theta_k + \eta \cdot \nabla_\theta V^{\pi_\theta}(\mu)|_{\theta=\theta_k}$

Policy Gradient

- Parametric policy $\pi_\theta(a|s)$
- Gradient update rule: $\theta_{k+1} = \theta_k + \eta \cdot \nabla_\theta V^{\pi_\theta}(\mu)|_{\theta=\theta_k}$
- Policy gradient theorem:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma} \cdot \mathbb{E}_{s,a\sim d_\mu^{\pi_\theta}}[A^{\pi_\theta}(s,a) \cdot \nabla_\theta \log \pi_\theta(a|s)]$$

$$\downarrow$$

$$d_\mu^{\pi_\theta}(s,a) = (1-\gamma) \cdot \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t-1} \cdot \mathbb{I}[s_t = s, a_t = a] | \mu, \pi\right]$$

Outline

- **Preliminaries**
- Test-time Attacks and Defenses in RL
- Training-time Attacks in RL
- Training-time Defenses in RL
- Adversarial Attacks in Multi-agent RL
- Concluding Remarks

References

- Goodfellow et al., Explaining and Harnessing Adversarial Examples, 2015.
- Sharif et al., Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition, 2016.
- Sharif et al., A General Framework for Adversarial Examples with Objectives, 2019.