NeurIPS 2024

Constrained Adaptive Attack: Effective Adversarial Attack Against Deep Neural Networks for Tabular Data

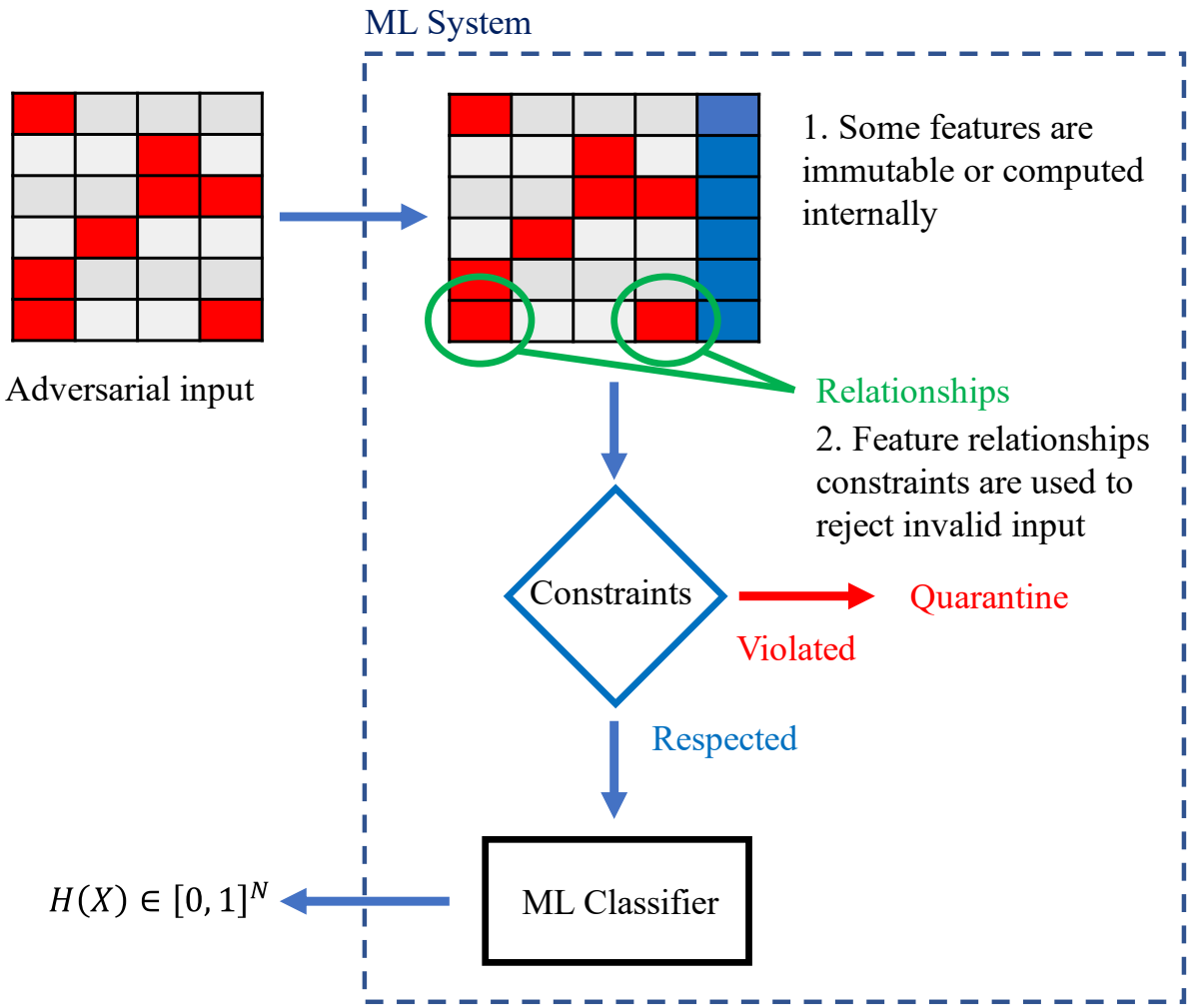
Thibault Simonetto[1], Salah Ghamizi[2, 3], Maxime Cordy[1]

[1]University of Luxembourg, Luxembourg

[2]Luxembourg Institute of Science and Technology, Luxembourg

[3]RIKEN Center for Advanced Intelligence Project, Tokyo, Japan

Adversarial examples in tabular data



Relation constraints on feature space

Finance:

$$avg\ transaction\ amount\ \leq max\ transaction\ amount$$

$$installment = loan\_amount \times \frac{int\_rate \times (1+int\_rate)^{term}}{(1+int\_rate)^{term}-1}$$

Problem formulation

Given a classification model H,

a maximum perturbation ϵ under a L_p distance

a set of **constraints Ω**

Objective of **constrained** adversarial attacks, for clean sample x find perturbation δ :

- ✓ With $H(x) \neq H(x+\delta)$
- ✓ With $L_p(x, x+\delta) < \epsilon$
- ✓ $x + \delta \vDash \Omega$

Feature relation constraints as a penalty function

Objective

$penalty(x, \omega)$ how far is x from satisfying ω $\quad$ $penalty(x, \omega) = 0 \ \ iff \ \omega(x) = \mathbf{True}$

Example

$\omega_1 \equiv avg\_transaction \leq max\_transaction$ $\quad$ $penalty(x, \omega_1) = max(0, avg\_transaction - max\_transaction)$

Mapping to penalty functions

Constraint	Penalty function
$\psi_1 = \psi_2$	$\mid \psi_1 - \psi_2 \mid$
$\psi_1 \leq \psi_2$	$max(0, \psi_1 - \psi_2)$
$\psi_1 < \psi_2$	$max(0, \psi_1 - \psi_2 + \tau)$
$\omega_1 \wedge \omega_2$	$\omega_1 + \omega_2$
$\omega_1 \vee \omega_2$	$\min(\omega_1, \omega_2)$

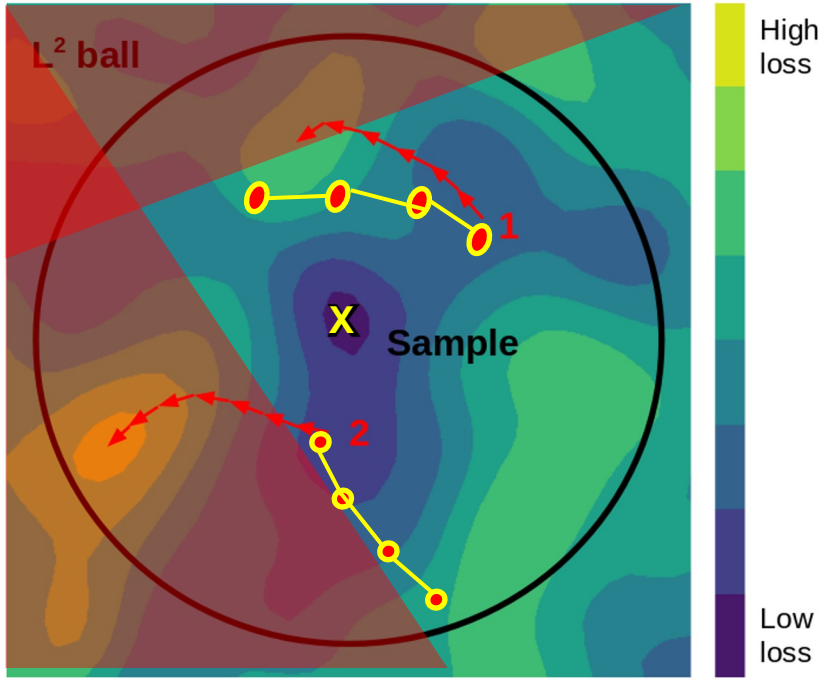
Constrained Adaptive PGD

Projected Gradient Descent (PGD)

$$x^{(k+1)} = P_{\mathcal{S}}(x^{(k)} + \eta \nabla l(h(x), y))$$

Gradient loss + Constraints regularization

$$\mathcal{L}'(x) = l(h(x), y) - \sum_{\omega_i \in \Omega} penalty(x, \omega_i)$$



Constrained Adaptive PGD

Gradient loss + Constraints regularization

$$\mathcal{L}'(x) = l(h(x), y) - \sum_{\omega_i \in \Omega} penalty(x, \omega_i)$$

Constrained Gradient Descent

$$z^{(k+1)} = P_{\mathcal{S}}(x^{(k)} + \eta^{(k)}(\nabla \mathcal{L}'(x^{(k)}))$$

Adaptive Step size η

$$\eta^{(0)} = 2\epsilon,\ \eta^{(k+1)} = \left\{ \begin{array}{ll} \eta^{(k)}/2, & \text{if } \mathcal{L}' \text{ does not decrease} \\ \eta^{(k)}, & \text{otherwise} \end{array} \right\}$$

Gradient step momentum α

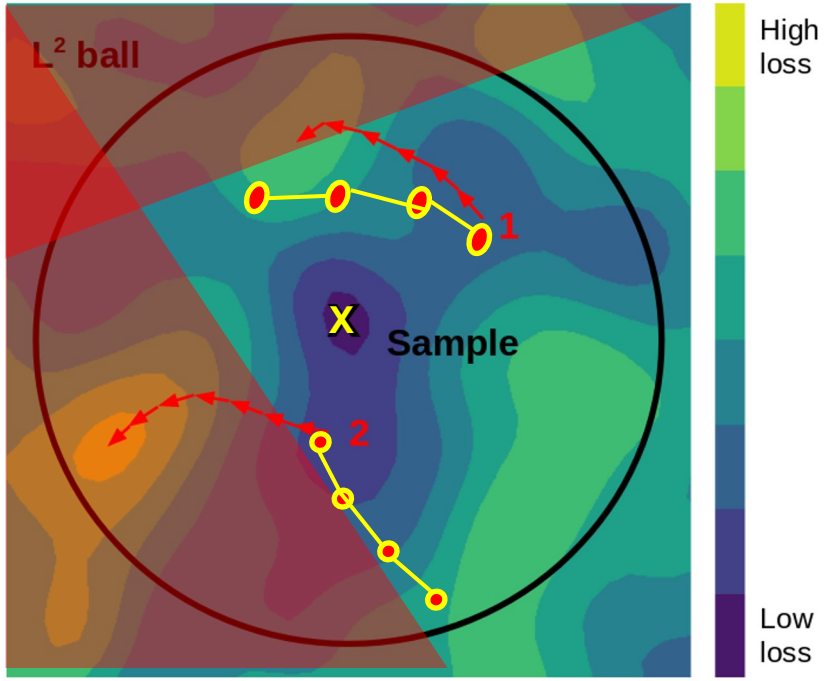
$$x^{(k+1)} = R_{\Omega}(P_{\mathcal{S}}(x^{(k)} + \alpha \cdot (z^{(k+1)} - x^{(k)}) + (1 - \alpha) \cdot (x^{(k)} - x^{(k-1)})))$$

Current gradient direction

Previous perturbation direction

Repair operator R_{Ω}

$$\omega_1 \equiv installment = loan\_amount \times \frac{int\_rate \times (1 + int\_rate)^{term}}{(1 + int\_rate)^{term} - 1}$$

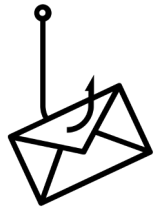


Experimental settings

Datasets:

Credit scoring
Lending Club Loan Data

Botnet detection
CTU



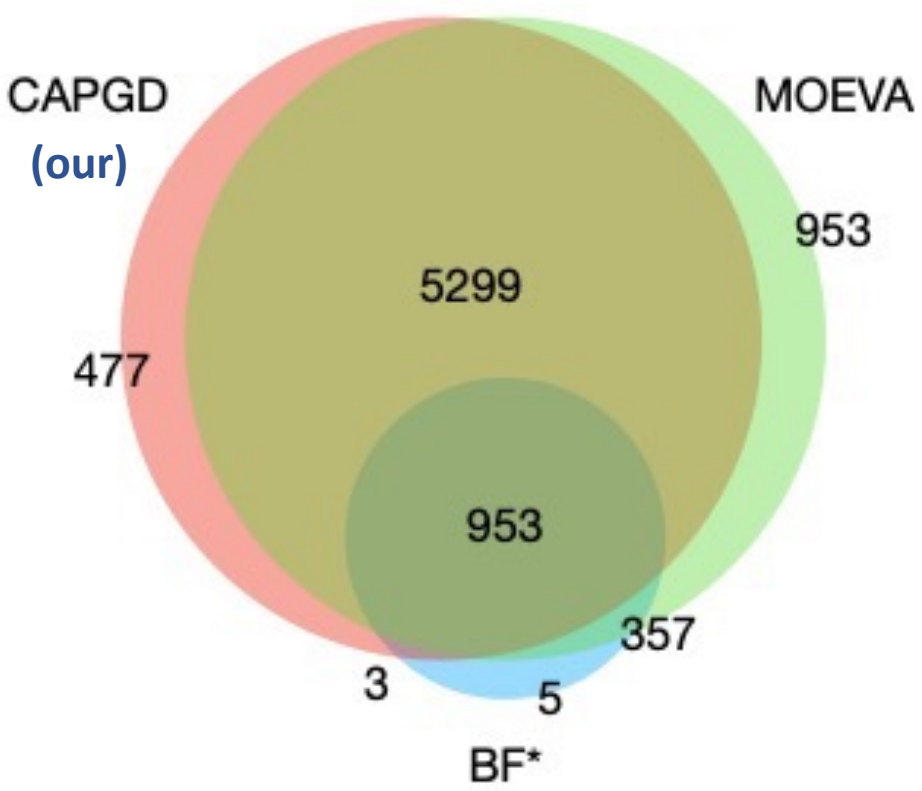
URL phishing
URL

ICU survival
WIDS

Models: 5 Neural network architectures

- 2 Regularizations
- 2 Transformers
- 1 Semi-supervised

Are these attacks complementary?



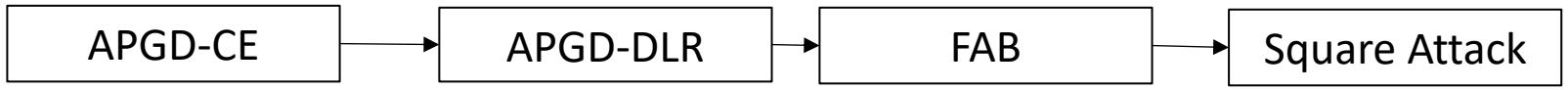
Set and number of successfull attacks
across all models and datasets

Insight:

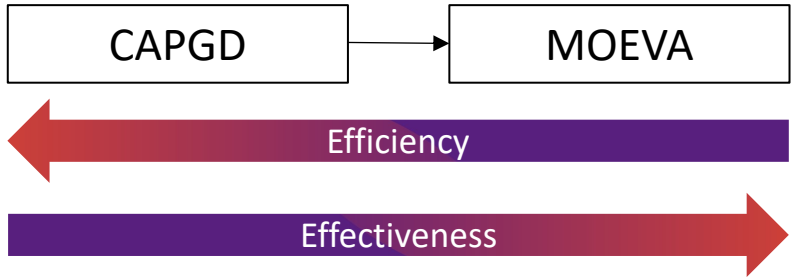
1. MOEVA and CAPGD are complementary
2. Together they subsume BF*

Constrained Adaptive Attack

Inspired from AutoAttack [1]



We propose CAA

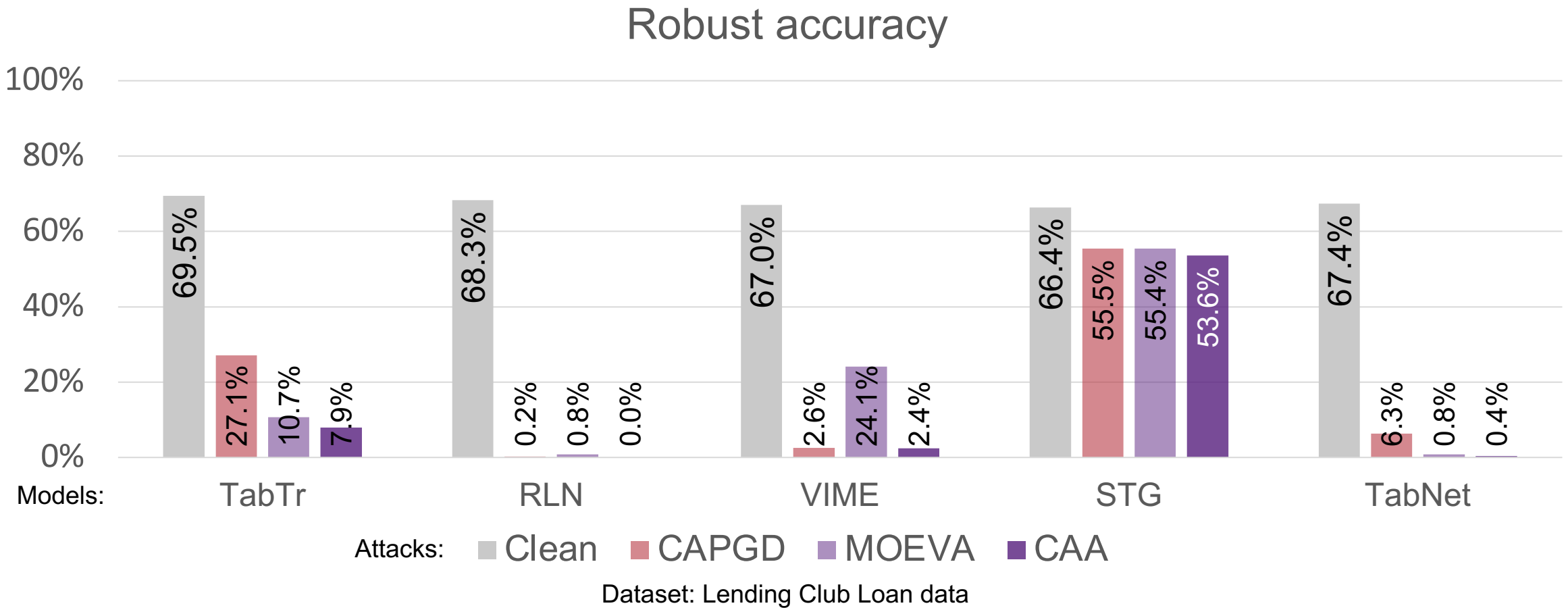


99.9%

Of examples can be generated by CAA

[1] Francesco Croce and Matthias Hein “Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks” ICML 2020

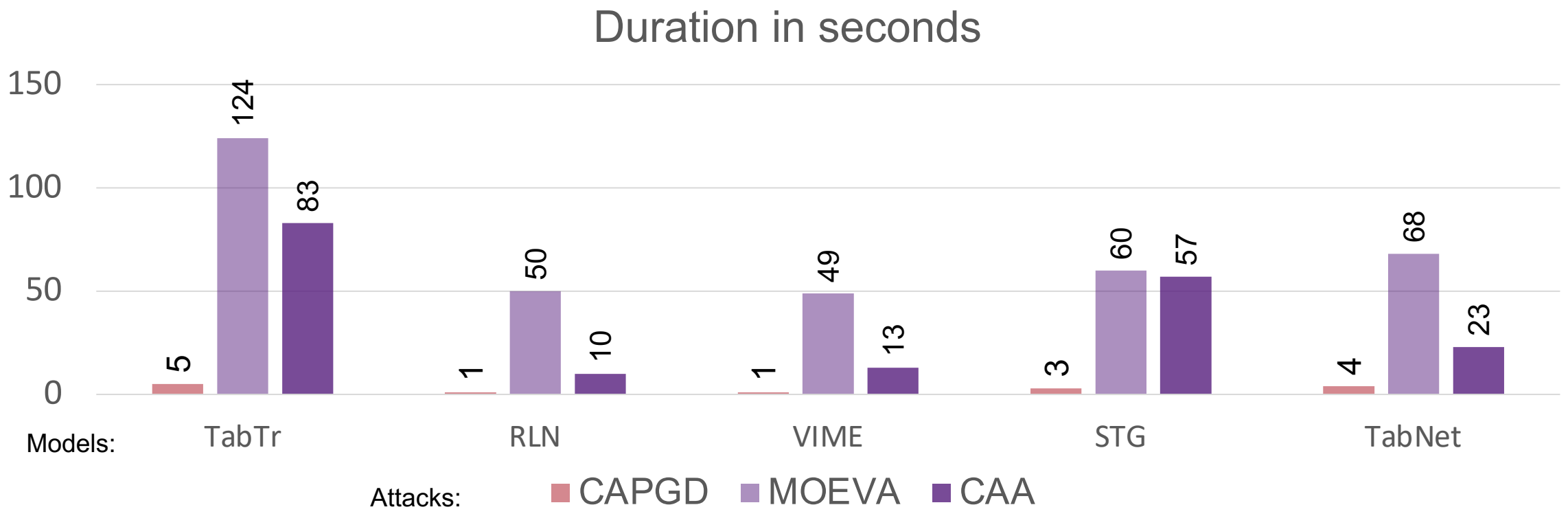
Impact of CAA



Dataset: Lending Club Loan data

Insight:
CAA effectively combines the benefits of CAPGD and MOEVA

Efficiency of CAA

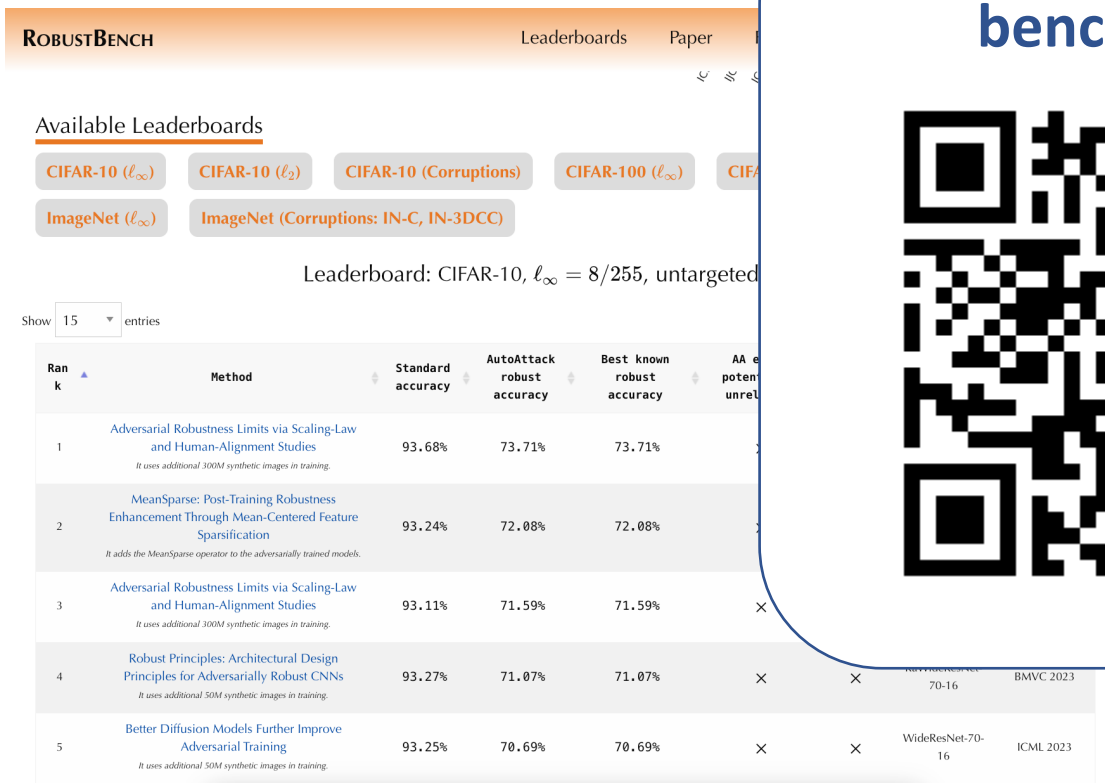


Dataset: Lending Club Loan data

Insight:
CAA reduces execution costs by up to 5 times compared to MOEVA

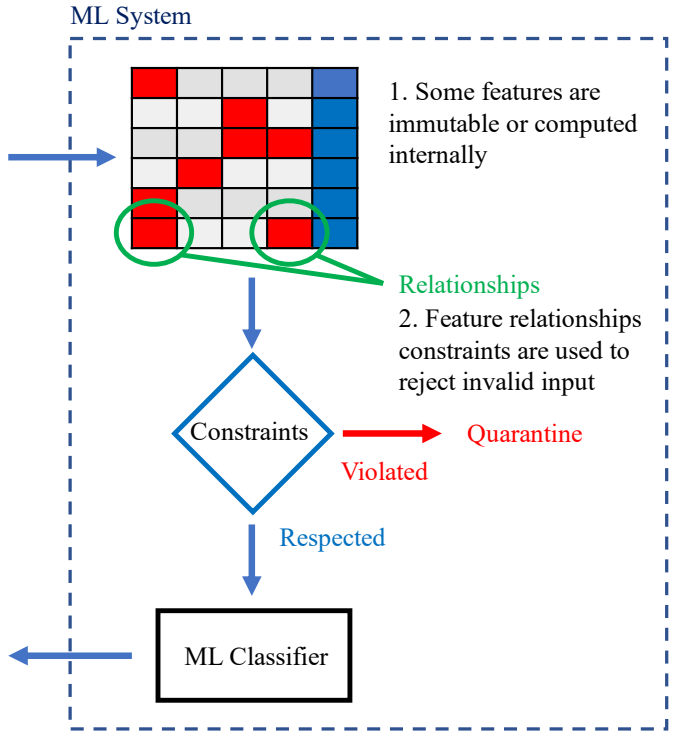
Conclusion

RobustBench



Checkout our benchmark

CAA: tabular attack under constraints



https://robustbench.github.io/

https://github.com/serval-uni-lu/tabularbench

This project is supported by the Luxembourg National Research Fund, grant BRIDGES/2022/IS/17437536.