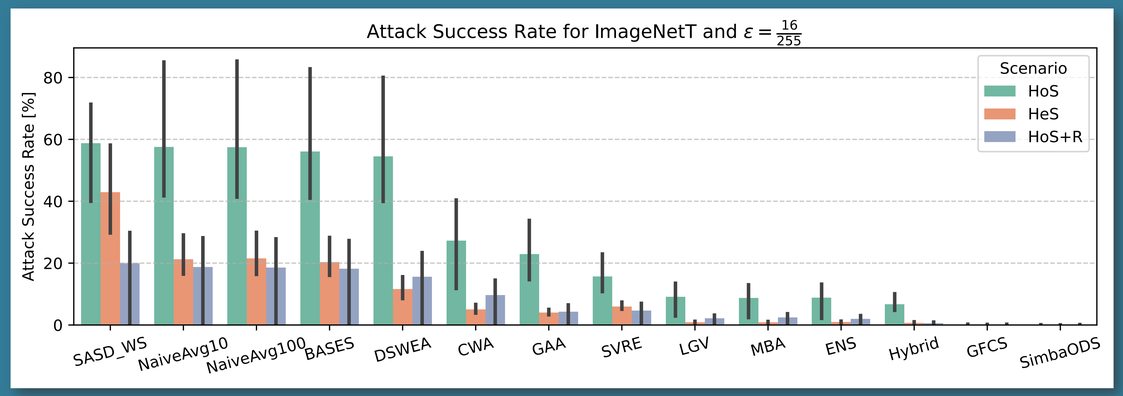
NEURAL INFORMATION PROCESSING SYSTEMS

Stop Trusting Flawed Evaluations



TransferBench defines a new standard for adversarial black-box benchmarking

sAIfer Lab
Joint lab on Safety and Security of AI

TRANSFERBENCH

Benchmarking Ensemble-based Black-box Transfer Attacks

Fabio Brau[a], Maura Pintor[a], Antonio Emanuele Cinà[b], Raffaele Mura[a], Luca Scionis[ac], Luca Oneto[b], Fabio Roli[b], Battista Biggio[a]

[a]University of Cagliari, Italy [b]University of Genoa, Italy [c]Sapienza University of Rome, Italy

TransferBench is a modular benchmark for evaluating ensemble-based black-box transfer attacks under realistic conditions, revealing how surrogate choice, target robustness, and query feedback affect attack transferability.

Formulation

Ensemble-based black-box minimum problem with m surrogate models:

$$x^* \in \arg\min_{x \in \mathcal{X}} \mathcal{L}_{\text{ens}}(x, t, \mathbf{f}; g(x)) \quad \text{s.t.} \quad \|x - x_0\|_p < \varepsilon$$

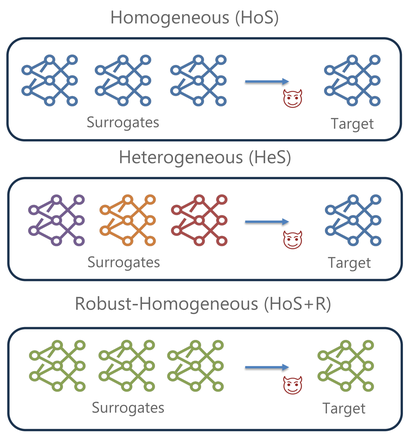
The computation can be decoupled into two sub-problems:

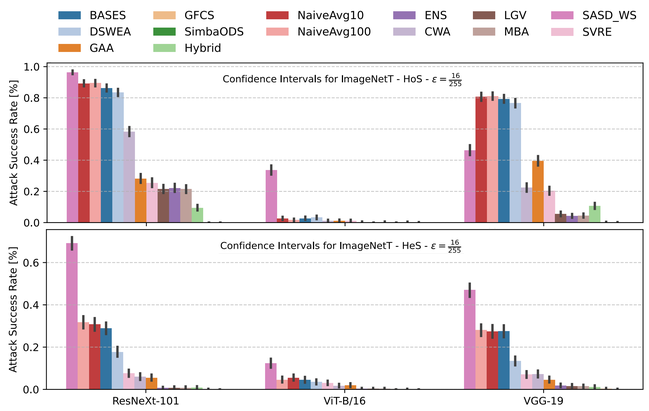
Surrogate-based Attack: $x^*(w) \in \arg\min_{x \in \mathcal{X}} \mathcal{L}_{\text{loc}}(x, t, \mathbf{f}; w)$

Query-based Refinement: $w^* \in \arg\min_{w \in \mathcal{W}} \mathcal{L}(g(x^*(w)), t)$

where $w \in \mathcal{W}$ represents ensemble parameters, e.g., ensemble weights.

Scenarios





Surrogate choice matters!

Attack success depends mostly on the surrogate pool.

Homogeneous surrogates yield high transferability, while cross-family ensembles sharply reduce ASR.

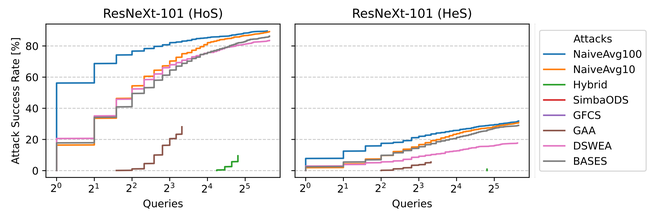
This indicates that surrogate similarity, not algorithmic design, drives most of the observed gains.

When evaluated on diverse or mismatched surrogates, the performance of the methods is worse than baseline

Queries or Multiple Trials?

Query refinement never improves over the naïve-average.

Simply querying the model to check success, without updating the weights, outperforms refinement strategies.



Usage

```python
from transferbench import AttackEval

# The user can define a custom method
def myattack(target, surrogates, *data, p, eps, Q) -> Tensor:
...
# Initializing the evaluation
evaluator = AttackEval(myattack)

# Selecting scenario (download datasets and models)
evaluator.set_scenarios("omeo-imagenet-inf")
results = evaluator.run()
```


Acknowledgment

This work has been partly supported by the EU-funded ELSA (GA no. 101070617), Sec4AI4Sec (GA no.101120393), and CoEvolution (GA no. 101168560); by the projects SERICS (PE00000014) and FAIR (PE00000013); by the EU—NGEU National Sustainable Mobility Center (CN00000023).