Faster Repeated Evasion Attacks in Tree Ensembles

Lorenzo Cascioli, Laurens Devos, Ondrej Kuzelka and Jesse Davis

KU LEUVEN LEUVEN.AI

DTAI
DECLARATIVE LANGUAGES & ARTIFICIAL INTELLIGENCE

fwo

Tuples
TRUSTWORTHY AI

Accuracy is not all you need!

We often need models to guarantee additional properties.

ML model to accept/reject loans has 99% test accuracy.
Could a loan application ever be…

- accepted for a **man**, but rejected for a **woman**?
- rejected, but accepted with **imperceptible perturbation**?

Standard performance metrics cannot answer these questions!

We need **verification** techniques that **reason about model behavior** and **provide guarantees** over any possible unseen input.

Verification often involves repeatedly performing evasion attacks

Evasion Attack

Tiny input change
that fools the model.

$T($ $) = 4,$

$T($ $) = 3?$

e.g., one attack per each example in a large test set allows to…

- assess if a model is robust to small perturbations of the input
- improve a model’s robustness
- *more examples in the paper*

🔄 We often need to repeatedly solve **lots of similar queries.**

⚠ But current methods treat each query in **isolation**!

Research Question

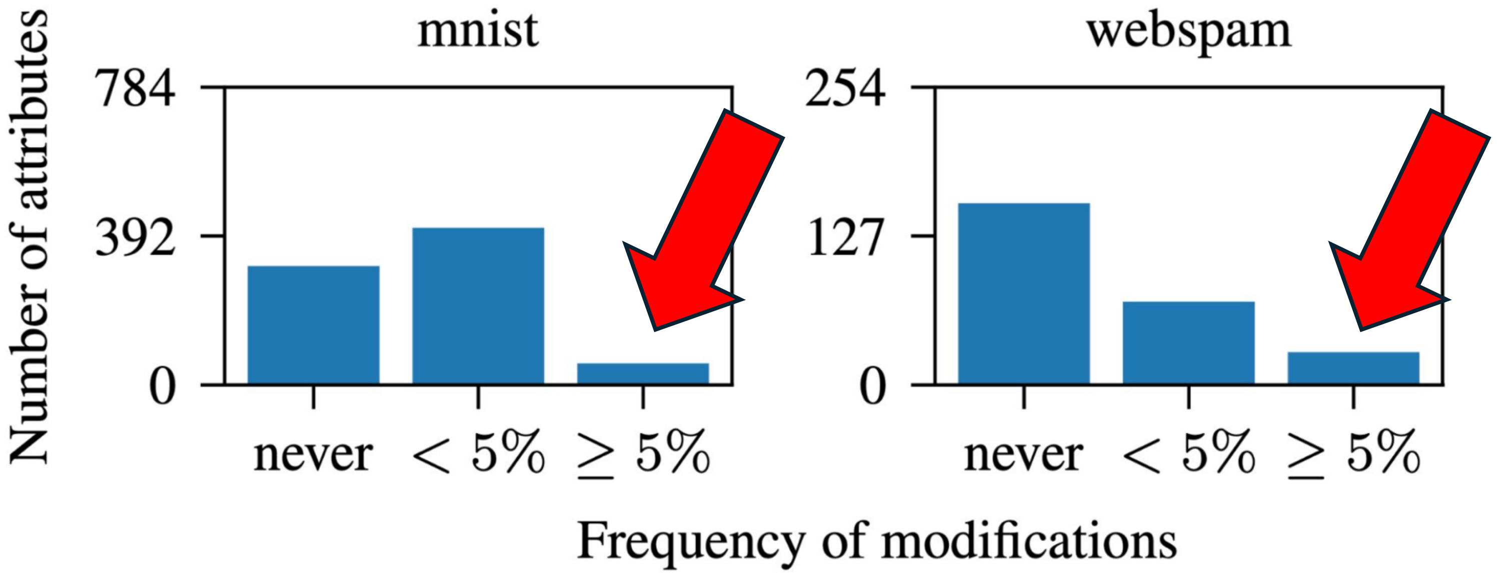
Can we exploit the fact that verification often consists of many similar problems?

Goal

Improve the applicability of verification tasks relying on repeated evasion attacks.

Contribution 1: Insight

Most attacks perturb the same small set of attributes.



Only 10-15% of the attributes is perturbed by more than 5% of the performed attacks.

Contribution 2: Method

A two-step approach for faster repeated evasion attacks.

1. **Quickly find the set of commonly perturbed attributes with a theoretically grounded procedure**
   - run a few attacks using all attributes
   - rank attributes based on how often they are perturbed, and select the top k%
   - a statistical test determines the value of k

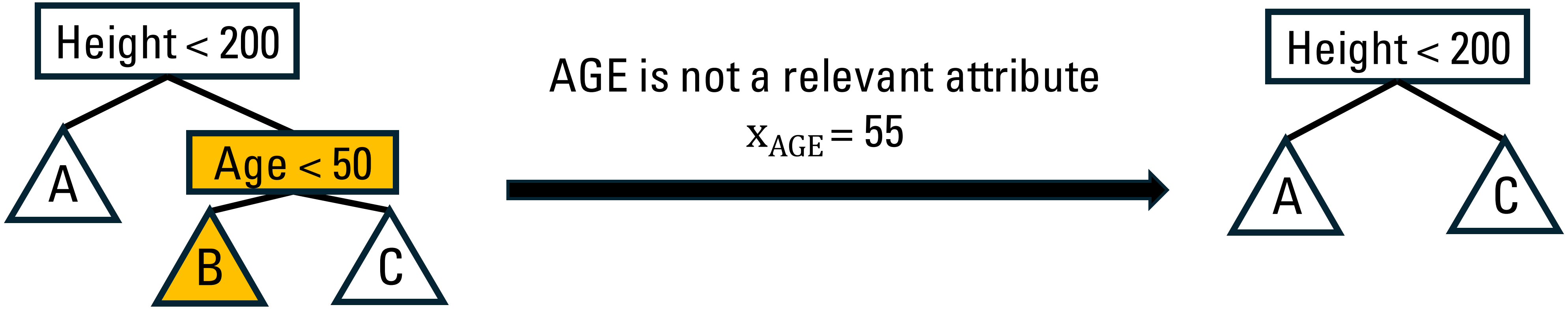
Contribution 2: Method

A two-step approach for faster repeated evasion attacks.

1. Quickly find the set of commonly perturbed attributes with a theoretically grounded procedure

2. **Modify attack methods to only perturb the identified attributes**

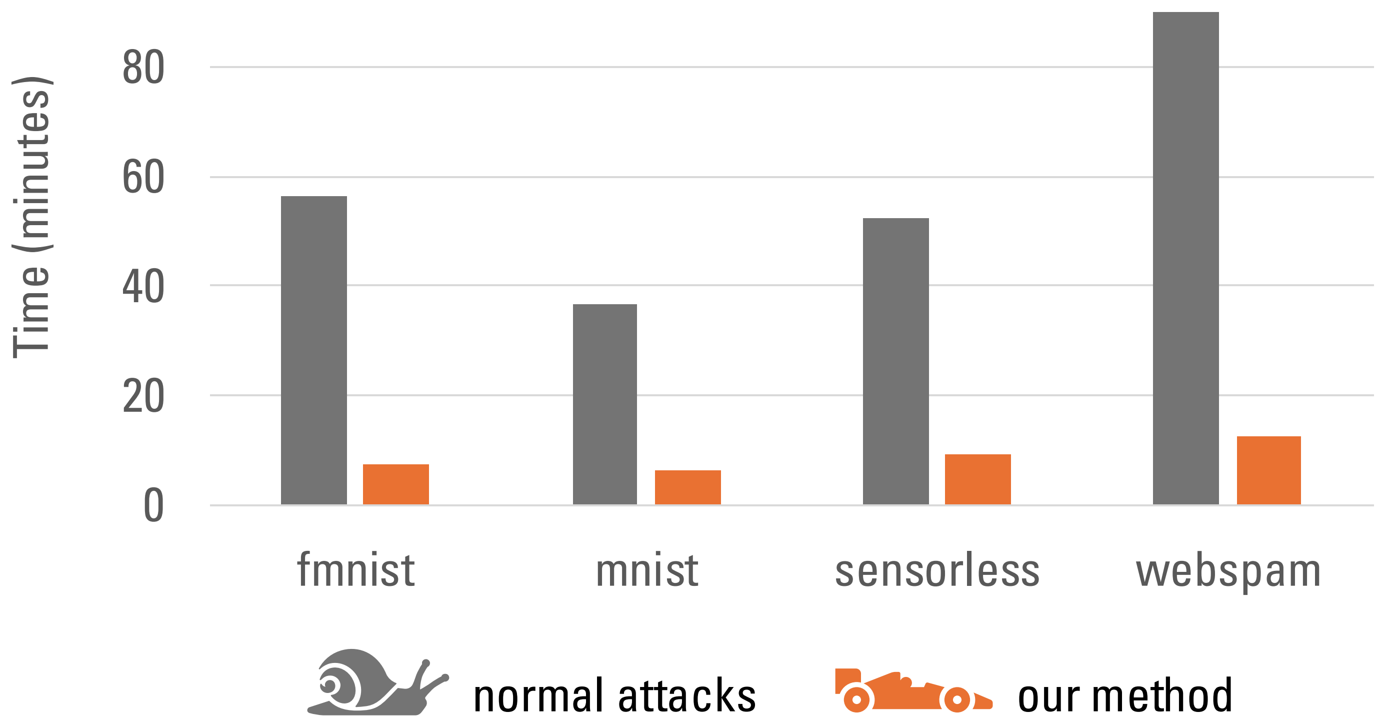
We use ensemble pruning to reduce model size:



Experimental Evaluation

Average speedup of **9x** to run 10 000 evasion attacks.

e.g., repeatedly attack a random forest ensemble using kantchelian attack:



Detailed results in the paper:

- more datasets
- more ensemble types
- more evasion attack methods

Faster Repeated Evasion Attacks in Tree Ensembles

Motivation Many verification tasks require to **perform lots of evasion attacks.**

Problem Current methods **repeatedly solve similar problems** from scratch.

Insight In practise, only **few attributes are often altered.**

Algorithm

1. **Identify** subset of **relevant attributes.**
2. Run evasion attacks **only perturbing relevant attributes.**

Lorenzo Cascioli
@l_cascioli

Laurens Devos
@laudevs

Ondrej Kuzelka
@Kuzelka0

Jesse Davis
@jessejdavis1

check out
paper and code!

KU LEUVEN LEUVEN.AI
DTAI Declarative Languages & Artificial Intelligence
fwo
Tuples TRUSTWORTHY AI