CS 499/579: Trustworthy ML

05.11: (Targeted) Poisoning

Tu/Th 10:00 – 11:50 AM

Sanghyun Hong

sanghyun.hong@oregonstate.edu

Oregon State University

SAIL

Secure AI Systems Lab

Heads-up!

- Note
  - 3 over 7 critiques left are optional now
  - You received reviews from your peer
- Due dates
  - 5/16: Group project checkpoint II
  - 5/18: Written paper critique
  - 5/18: Homework 3 due
- Recommendation
  - Discuss slides with SH for in-class paper presentation
    - 5/18
    - 5/30 and 6/1
    - 6/6

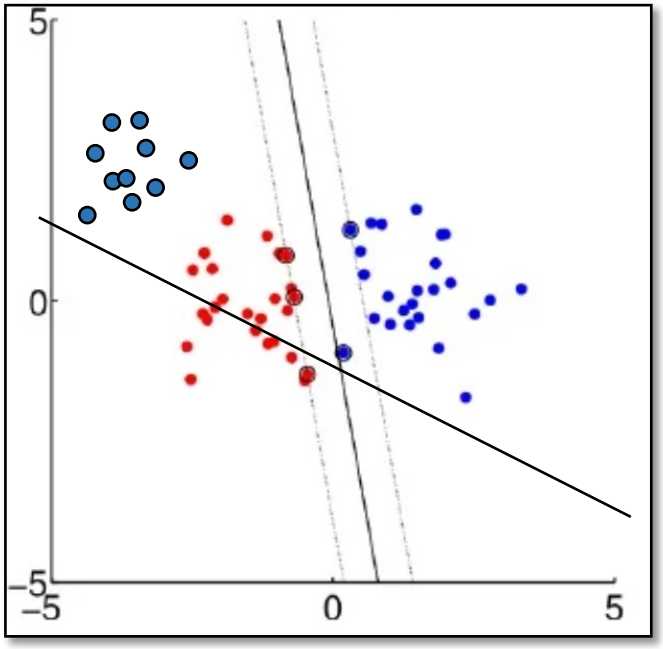
Heads-up!

- Note
  - Checkpoint Presentation II (on the 16th)
    - *15-min* presentation + *3-5 min* Q&A
    - Presentation *MUST* cover:
      - 1 slide on your research topic
      - 1-2 slides on your motivation and goal(s)
      - 1-2 slides on your *ideas* (how do you plan to achieve your goals)
      - 1-2 slides on your *experimental design* (in detail)
      - 1-2 slides on your *hypotheses* and *preliminary results* [*very important*]
      - 1 slide on your *next steps* until the final presentation

Topics for today

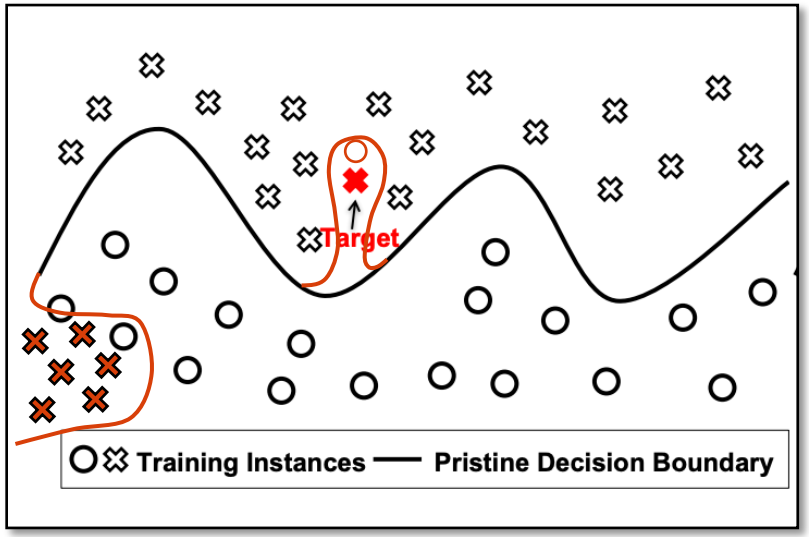
- (Targeted) Data Poisoning
  - Motivation
  - Threat Model
  - Prior attacks on
    - Clean-label poisoning attacks
    - (Advanced) Clean-label poisoning attacks
  - Conclusion (and implications)

Recap: conceptual illustration of the vulnerability to poisoning



← Linear model (SVM)

Neural Network →



Motivation

- Practical Constraints
  - You don't have control over your inputs
  - You don't want to mess-up the entire system

https://globalnews.ca/news/4567183/facial-recognition-technology-u-s-airports/
https://techcrunch.com/2016/08/11/friendblock/

Motivation – cont'd

- Practicality
  - Many ways to insert your malicious data
  - Human-inspection is not feasible (think about t

Data collection method

The Weibo data is obtained by a python crawler. The crawler automatically uses Weibo's advanced search function for keyword indexing.

The keywords we used included:

- COVID-19
- novel coronavirus(新型冠状病毒)
- corona(新冠)
- epidemics(疫情)
- novel pneumonia(新型肺炎)
- pneumonia in Wuhan(武汉+肺炎)

The crawler program automatically entered one of the keywords into the query box and set the query time range to be a specific hour. As illustrated in Figure 2, the crawler sets the time range from January 18, 2020, 00:00:00 to January 18, 2020, 01:00:00.

For each query, the time range increased by one hour, and each query searched all the new posts within an hour. Each query returned a maximum of 50 pages, each contained around 20 posts. If the number posts exceed the page limits, we cannot fully collect the information due to the limitations of the search function.

https://yiling-chen.github.io/flickr-cropping-dataset/
https://github.com/yleng/COVID-Weibo

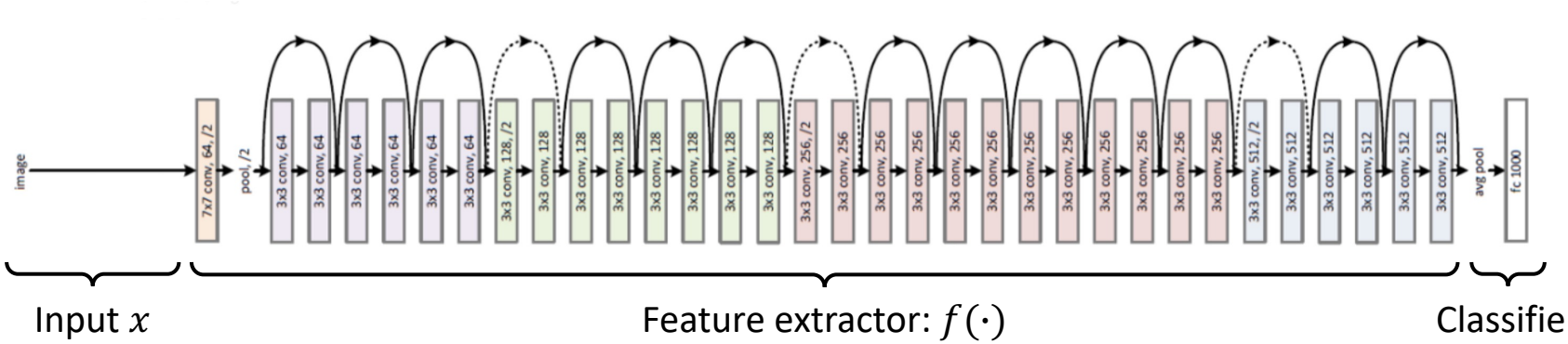
Poisoning threat model

- Goal
  - **Targeted** attack
  - Model causes a misclassification of (x_t, y_t), while preserving acc. on D_{val}
- Capability
  - Know a target (x_t, y_t)
  - Pick p candidates from test data (x_{c1}, y_{c1}), $(x_{c2}$… and craft poisons (x_{p1}, y_{p1}), $(x_{p2}$…
  - Inject them into the training data
- Knowledge
  - D_{tr} : training data
  - D_{test}: test-set data (validation data)
  - f: a model and its parameters θ
  - A: training algorithm (*e.g.*, mini-batch SGD)

Poisoning threat model

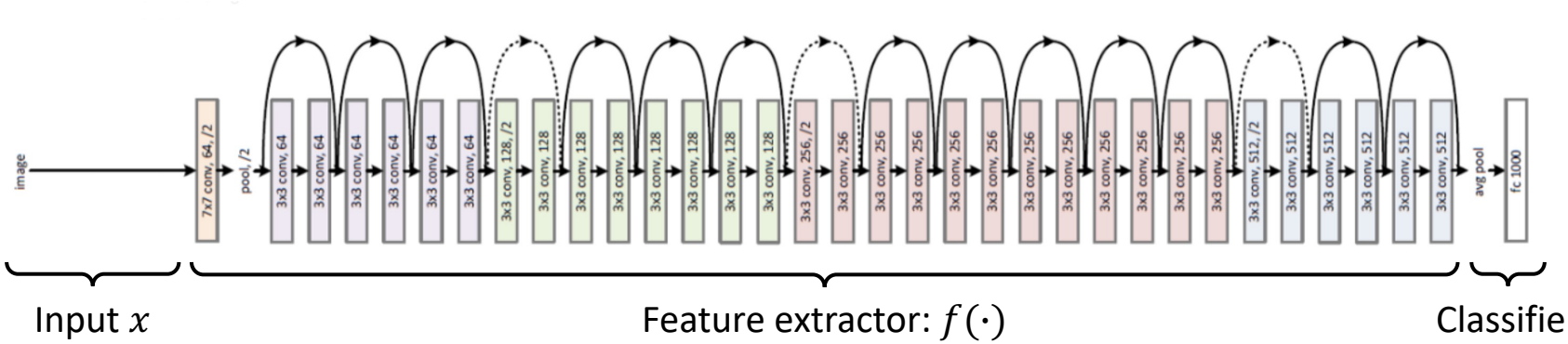
- Goal
  - Targeted **clean-label** ($y_{c1} = y_{p1}$) attack
  - Model causes a misclassification of (x_t, y_t), while preserving acc. on D_{val}
- Capability
  - Know a target (x_t, y_t)
  - Pick p candidates from test data (x_{c1}, y_{c1}), $(x_{c2}$… and craft poisons (x_{p1}, y_{p1}), $(x_{p2}$…
  - Inject them into the training data
- Knowledge
  - D_{tr} : training data
  - D_{test}: test-set data (validation data)
  - f: a model and its parameters θ
  - A: training algorithm (*e.g.*, mini-batch SGD)

Background: convolutional neural networks



- A conventional view:
  - Convolutions: extract features, embeddings, latent representations, …
  - Last layer: uses the output for a classification task

Background: convolutional neural networks



- Input-space ≠ Feature-space:
  - Two samples similar in the input-space can be far from each other in the feature-space
  - Two samples very different in the input-space can be close to each other in f

Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks

Shafahi *et al.* (Presented by Anirudh)

MetaPoison: Practical General-purpose Clean-label Data Poisoning

Huang *et al.*

Revisit: poisoning threat model

- Goal
  - Targeted **clean-label** ($y_{c1} = y_{p1}$) attack
  - Model causes a misclassification of (x_t, y_t), while preserving acc. on D_{val}
- Capability
  - Know a target (x_t, y_t)
  - Pick p candidates from test data (x_{c1}, y_{c1}), $(x_{c2}$… and craft poisons (x_{p1}, y_{p1}), $(x_{p2}$…
  - Inject them into the training data
- Knowledge
  - D_{tr} : training data
  - D_{test}: test-set data (validation data)
  - f: a model and its parameters θ
  - A: training algorithm (*e.g.*, mini-batch SGD)

Revisit: the key idea – feature collision

- Goal
  - Your poisons should work against any f and θ
  - Objective:

$$\mathbf{p} = \underset{\mathbf{x}}{\operatorname{argmin}} \; \|f(\mathbf{x}) - f(\mathbf{t})\|_2^2 + \beta \|\mathbf{x} - \mathbf{b}\|_2^2$$

Now you don't know the f, how can you estimate this?

- Revisit the previous idea
  - Bi-level optimization

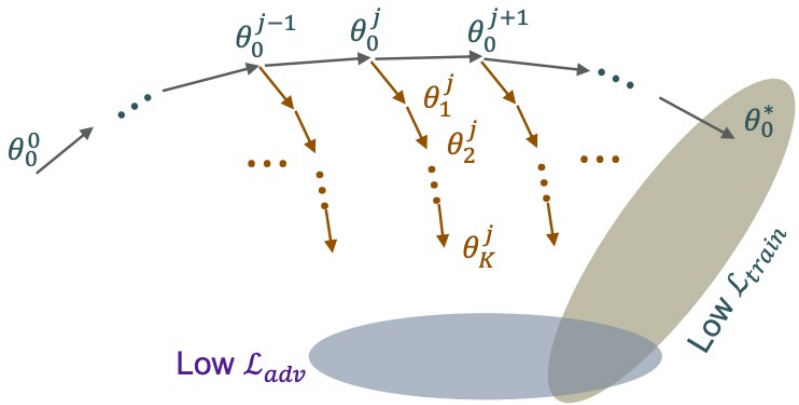
$$\begin{aligned} \arg\max_{\mathcal{D}_p} \quad & \mathcal{W}(\mathcal{D}', \boldsymbol{\theta}_p^\star), \\ \text{s.t.} \quad & \boldsymbol{\theta}_p^\star \in \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\mathcal{D}_{\text{tr}} \cup \mathcal{D}_p, \boldsymbol{\theta}) \end{aligned}$$

$$\begin{aligned} X_p^* &= \underset{X_p}{\operatorname{argmin}} \; \mathcal{L}_{\text{adv}}(x_t, y_{\text{adv}}; \theta^*(X_p)) \\ \theta^*(X_p) &= \underset{\theta}{\operatorname{argmin}} \; \mathcal{L}_{\text{train}}(X_c \cup X_p, Y; \theta) \end{aligned}$$

Problem: no control over θ

THE KEY IDEA: UNROLLING

- Goal
  - You *simulate all* the training procedures with *possible* $f, \theta s$ while crafting your poisons



Algorithm 1 Craft poison examples via MetaPoison

Input Training set of images and labels (X, Y) of size N, target image x_t, adversarial class y_{adv}, ϵ and ϵ_c thresholds, $n \ll N$ subset of images to be poisoned, T range of training epochs, M randomly initialized models.
Begin
Stagger the M models, training the mth model weights θ_m up to $\lfloor mT/M \rfloor$ epochs
Select n images from the training set to be poisoned, denoted by X_p. Remaining clean images denoted X_c
For $i = 1, \ldots, C$ crafting steps:
  For $m = 1, \ldots, M$ models:
    Copy $\tilde{\theta} = \theta_m$
    For $k = 1, \ldots, K$ unroll steps[a]:
      $\tilde{\theta} = \tilde{\theta} - \alpha \nabla_{\tilde{\theta}} \mathcal{L}_{\text{train}}(X_c \cup X_p, Y; \tilde{\theta})$
    Store adversarial loss $\mathcal{L}_m = \mathcal{L}_{\text{adv}}(x_t, y_{\text{adv}}; \tilde{\theta})$
    Advance epoch $\theta_m = \theta_m - \alpha \nabla_{\theta_m} \mathcal{L}_{\text{train}}(X, Y; \theta_m)$
    If θ_m is at epoch $T + 1$:
      Reset θ_m to epoch 0 and reinitialize
  Average adversarial losses $\mathcal{L}_{\text{adv}} = \sum_{m=1}^{M} \mathcal{L}_m / M$
  Compute $\nabla_{X_p} \mathcal{L}_{\text{adv}}$
  Update X_p using Adam and project onto ϵ, ϵ_c ball
Return X_p

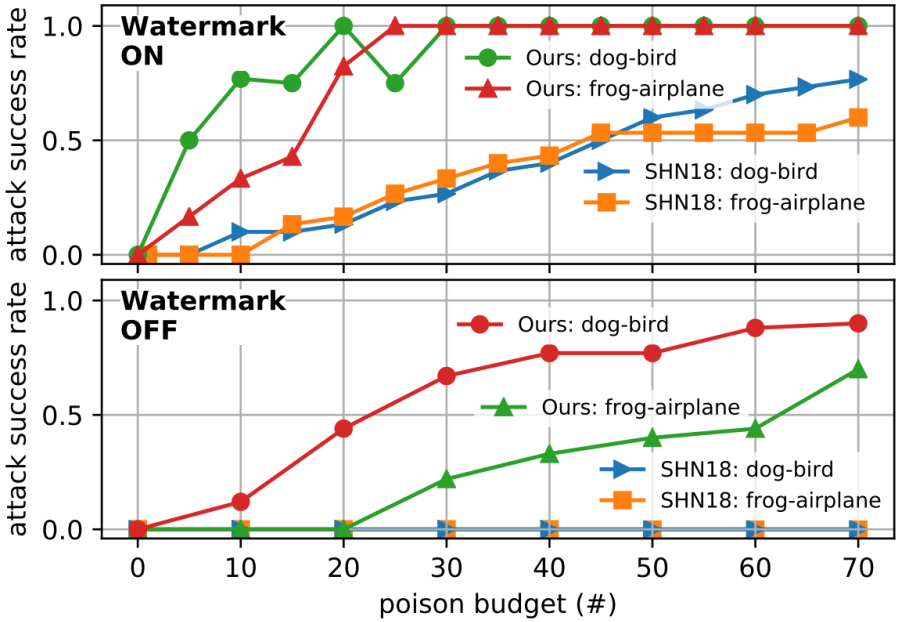
Evaluation

- Setup
  - Dataset: CIFAR-10
  - Models: 6-layer ConveNet (default), ResNet20, VGG13
  - Attack hyper-parameters:
    - C: 60 | M: 24 | K: 2

- Attacks
  - 30 randomly chosen targets
  - Increase the # poisons from 1 – 10% of the training data n
  - Baseline:
    - Poison Frogs!

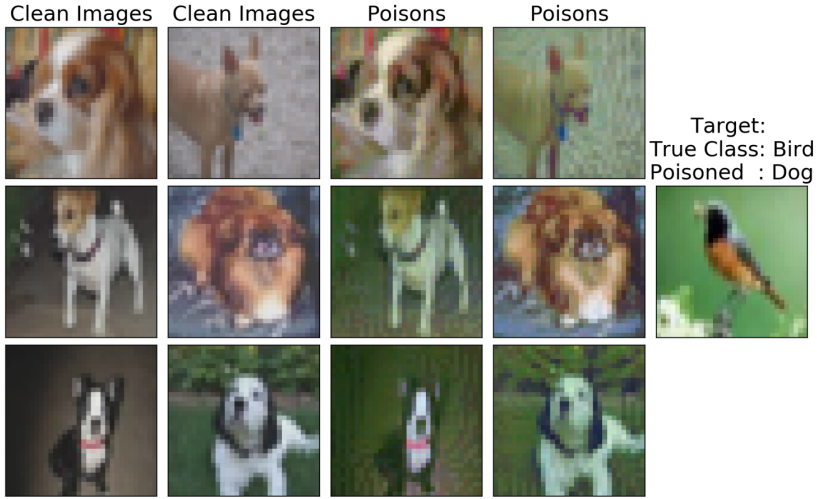
EVALUATION: TRANSFER LEARNING SCENARIO

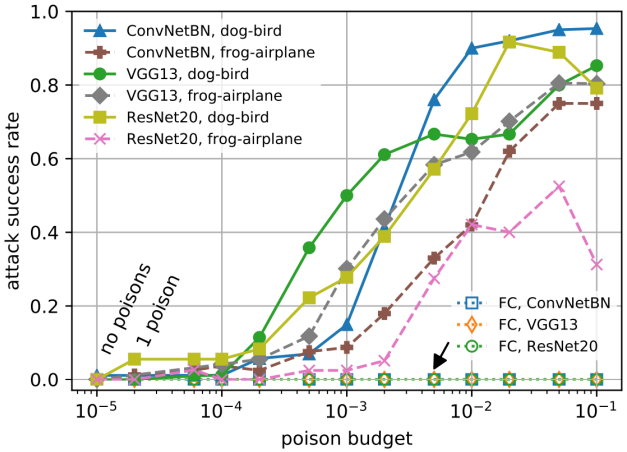
- MetaPoison vs. Poison Frogs

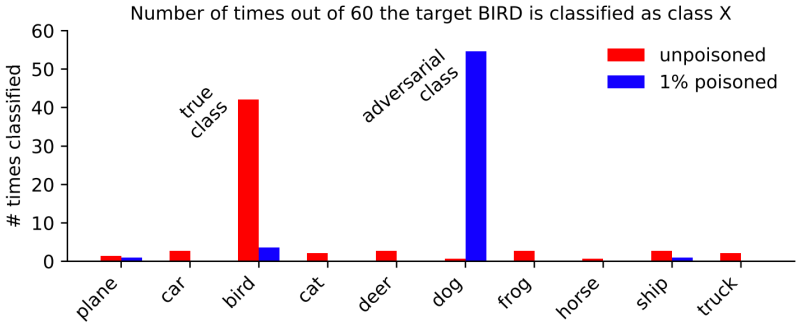


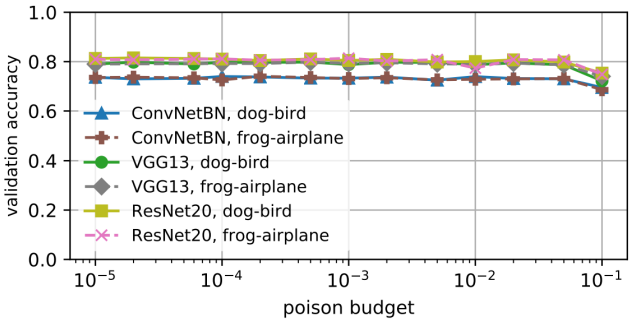
Evaluation: end-to-end scenario

- MetaPo



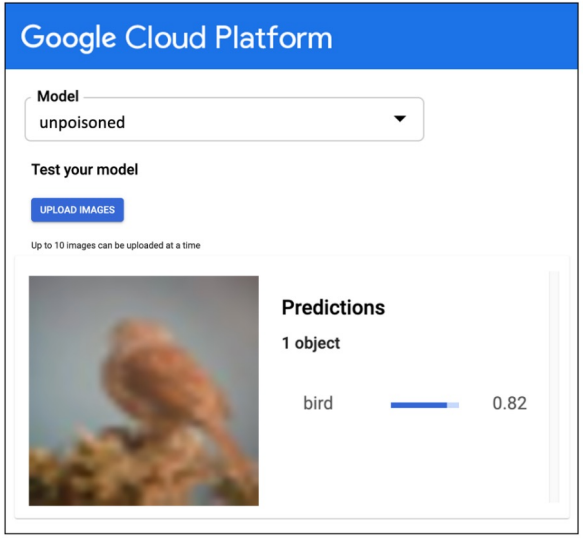


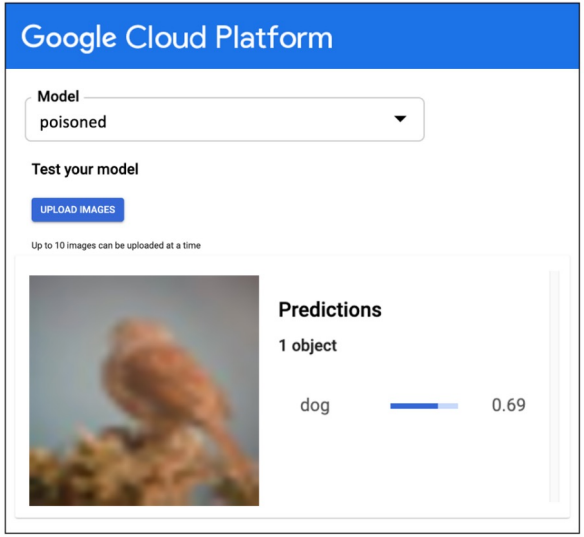


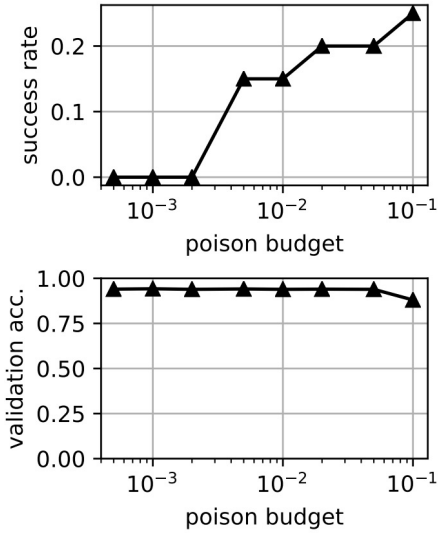


EVALUATION: EXPLOITATION IN REAL-WORLD

- Results







Topics for today

- (Targeted) Data Poisoning
  - Motivation
  - Threat Model
  - Prior attacks on
    - Clean-label poisoning attacks
    - (Advanced) Clean-label poisoning attacks
  - Conclusion (and implications)

Thank You!

Tu/Th 10:00 – 11:50 AM

Sanghyun Hong

https://secure-ai.systems/courses/MLSec/W22

Oregon State University

SAIL

Secure AI Systems Lab