# ModelGuard: Information-Theoretic Defense Against Model Extraction Attacks

Minxue Tang, Anna Dai, Louis DiValentin, Aolin Ding, Amin Hass,
Neil Zhenqiang Gong, Yiran Chen, Hai "Helen" Li

33RD USENIX SECURITY SYMPOSIUM

Duke 100 CENTENNIAL

accenture

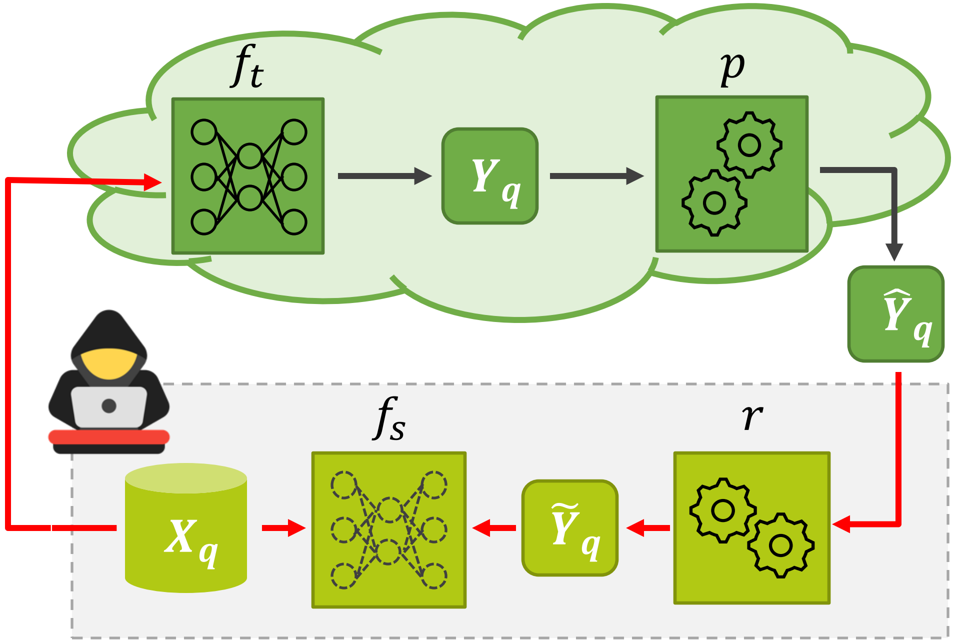
Duke

# Outline

- Background and Threat Model
- Defense Objective and Constraints
- Methodology
  - ModelGuard-W
  - ModelGuard-S
- Experimental Results
- Conclusions

# Background

- Model extraction of ML-as-a-Service (MLaaS) systems
  - Confidential Model: $f_t(\cdot; w_t)$
  - Substitute Model: $f_s(\cdot; w_s)$
  - Query Dataset: $X_q$
  - Clean Prediction: $Y_q = f_t(X_q; w_t)$
- Prediction perturbation defense
  - Prediction Perturbation Mechanism: $\hat{Y}_q = p(Y_q)$
- Adaptive model extraction attack
  - (Adaptive) Prediction Recovery: $\tilde{Y}_q = r\ (\hat{Y}_q)$




Duke

# Threat Model

- Parameter-stealing attack

$$\min_{w_s} \|w_s - w_t\|_2^2$$

- Functionality-stealing attack

$$\min_{w_s} L\left(f_s(X_q; w_s), f_t(X_q; w_t)\right) = L(\tilde{Y}_q, Y_q)$$

Duke

# Defense Objective

- Defense against parameter-stealing attack

$$\max_{\hat{Y}_q} \|w_s - w_t\|_2^2$$

  - Subject to $w_s = \textcolor{red}{\text{Train}}\left(X_q, r\left(p(\hat{Y}_q)\right)\right)$.

- Defense against functionality-stealing attack

$$\max_{\hat{Y}_q} L\left(\tilde{Y}_q, Y_q\right)$$

- Unified objective against both attacks (Lemma 1):

$$\|w_s - w_t\|_2^2 \geq \frac{2}{M}\left[L\left(\tilde{Y}_q, Y_q\right) - L(Y_q, Y_q)\right]$$

# Defense Constraints

- $\ell_1$ Distortion Constraint

$$\left\|\hat{y}_q - y_q\right\|_1 \leq \epsilon$$

- Top-1 Accuracy Preserving Constraint

$$\arg\max \hat{y}_q^{(k)} = \arg\max y_q^{(k)}$$

- Simplex Constraint

$$\sum_k \hat{y}_q^{(k)} = 1\,, \hat{y}_q \succcurlyeq 0$$

Duke

# Optimization Challenges

- Arbitraty recovery function $r$ used by the attacker.

$$\max_{\hat{Y}_q} L\left(r\left(\hat{Y}_q\right), Y_q\right)$$

- Two assumptions:
  - (ModelGuard-W) The attacker uses the perturbed prediction for training directly:

$$\tilde{Y}_q = r\left(\hat{Y}_q\right) = \hat{Y}_q$$

  - (ModelGuard-S) The attacker uses a strong adaptive attack that leads to the minimal recovery distance:

$$\min_{r} \mathbb{E}\left[\left\|r\left(\hat{Y}_q\right) - Y_q\right\|_2^2\right]$$

Duke

# ModelGuard-W

- Assumption 1:

$\tilde{y}_q = \hat{y}_q$,
CE Loss

$$\max_{\hat{y}_q} L_{CE}(\tilde{y}_q, y_q) = \min_{\hat{y}_q} \sum_k y_q^{(k)} \log \hat{y}_q^{(k)} = \min_{\hat{y}_q} \langle y_q, \log \hat{y}_q \rangle \xleftrightarrow{\text{Solution Similarity}} \min_{\hat{y}_q} \langle \log y_q , \hat{y}_q \rangle$$

Non-convex optimization

Linear Programming

Subject to
$\left\| \hat{y}_q - y_q \right\|_1 \leq \epsilon,$
$\arg\max \hat{y}_q^{(k)} = \arg\max y_q^{(k)},$
$\sum_k \hat{y}_q^{(k)} = 1 , \hat{y}_q \succcurlyeq 0.$

Duke

# ModelGuard-S

- Lower bound of the recovery distance (Lemma 2): $h(Y_q|\hat{Y}_q) = h(Y_q) - I(Y_q;\hat{Y}_q)$

$$\mathbb{E}\left[\left\|r(\hat{Y}_q) - Y_q\right\|_2^2\right] \geq \frac{NC}{2\pi e}\exp\left(\frac{2}{NC}h(Y_q|\hat{Y}_q)\right)$$

  - The lower bound is achieved by Bayes Attack $r(\hat{Y}_q) = r^*(\hat{Y}_q) = \mathbb{E}[Y_q|\hat{Y}_q]$.

- New optimization:

$$\boxed{\begin{array}{l}\min_{\hat{Y}_q} I(Y_q;\hat{Y}_q) \\ \text{Subject to } (\forall \hat{y}_q \in \hat{Y}_q) \\ \left\|\hat{y}_q - y_q\right\|_1 \leq \epsilon,\end{array}}$$

$$\arg\max \hat{y}_q^{(k)} = \arg\max y_q^{(k)},$$

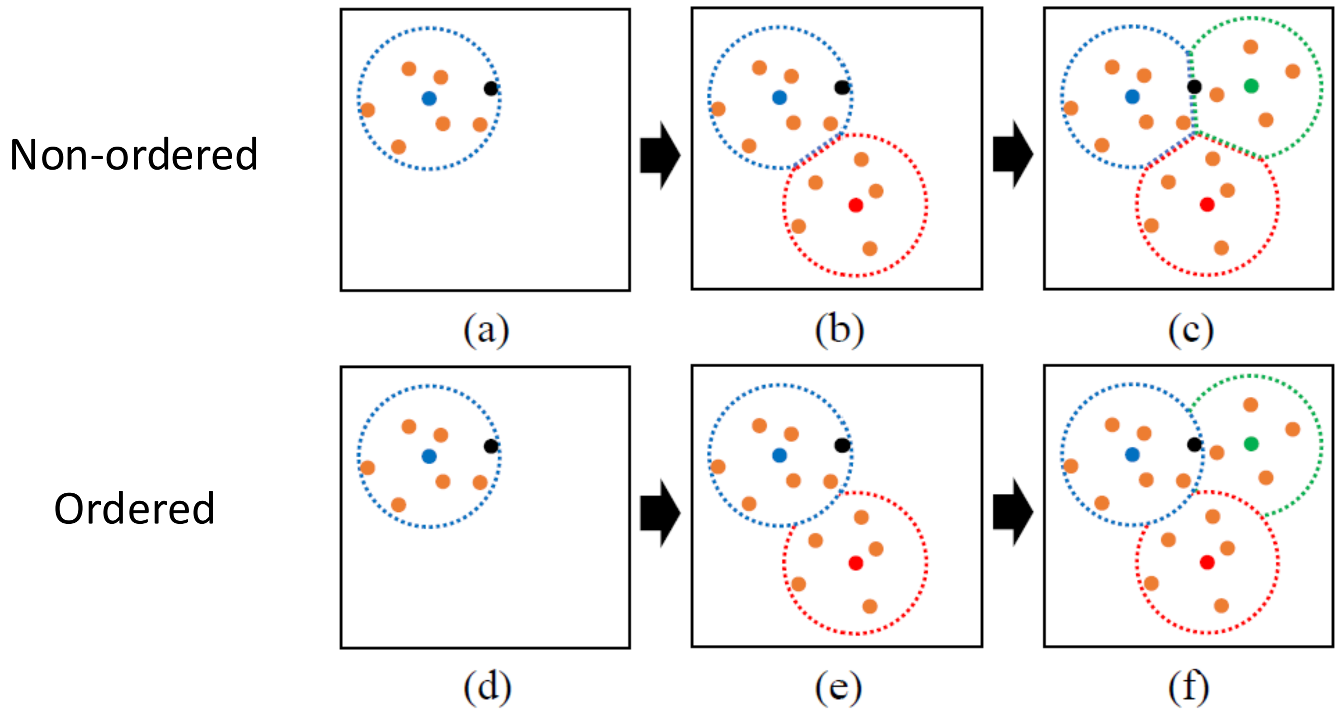
$$\textstyle\sum_k \hat{y}_q^{(k)} = 1, \hat{y}_q \succcurlyeq 0.$$

Rate-distortion Problem (Lossy Compression)

Duke

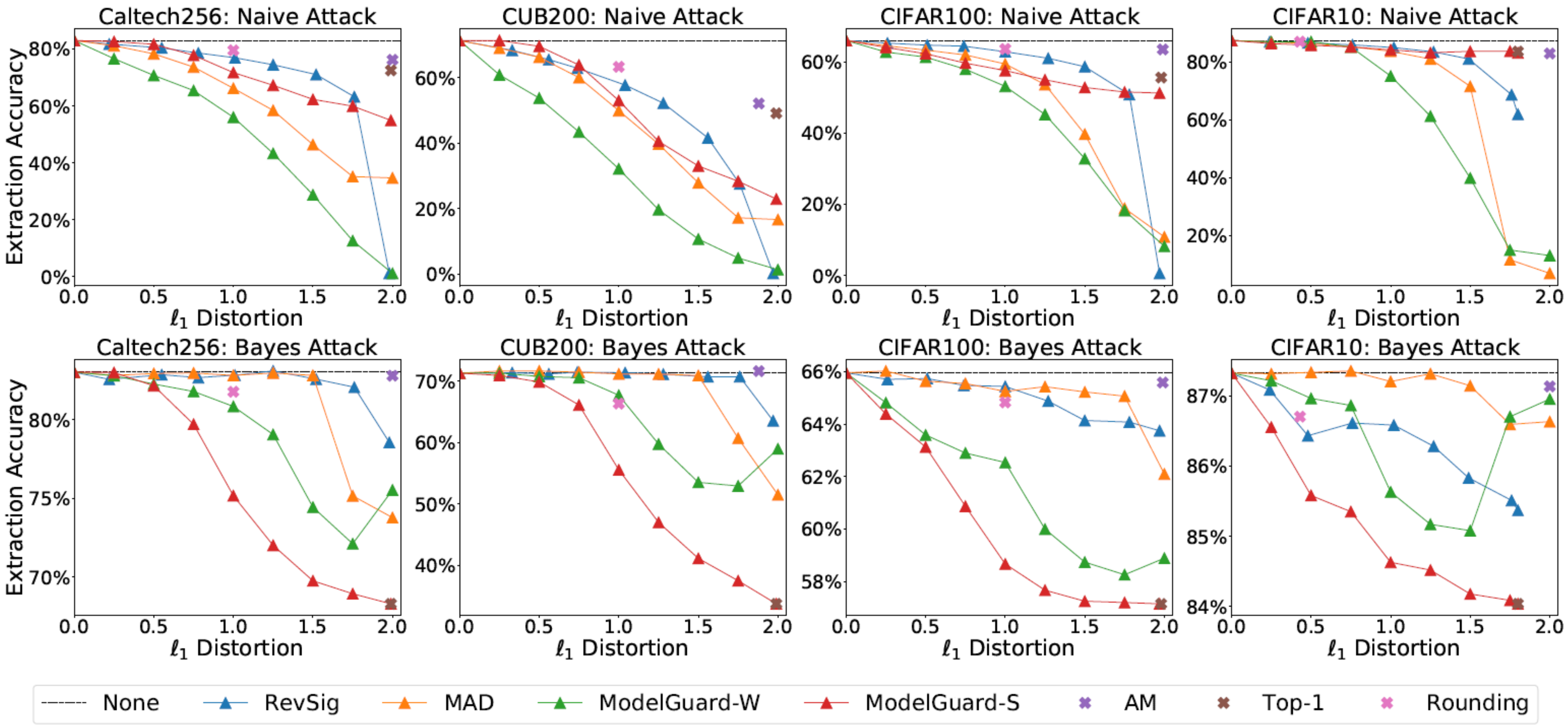
# ModelGuard-S

- Ordered Incremental Prediction Quantization:
  - Automatically adjust the number of clusters to meet the distortion constraint.
  - Avoid information leakage caused by change of the quantization boundary.



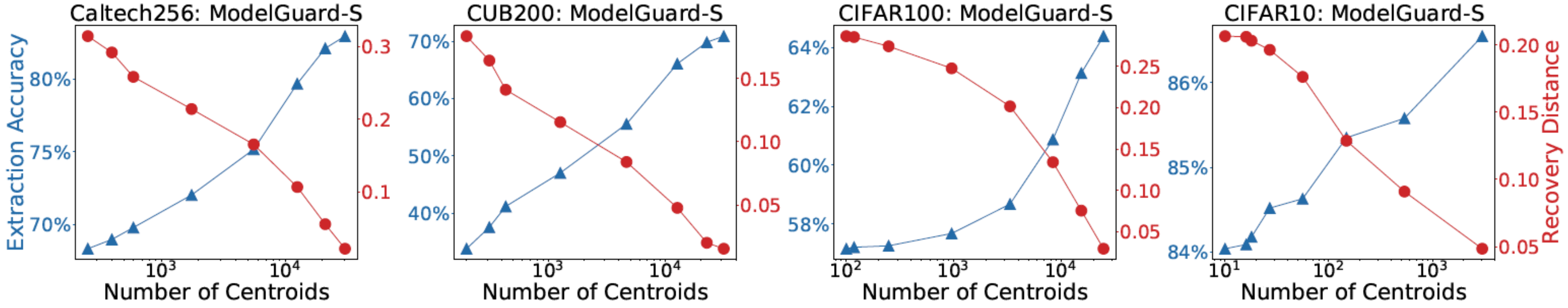
# Experimental Results

- ModelGuard achieves better defensive performance

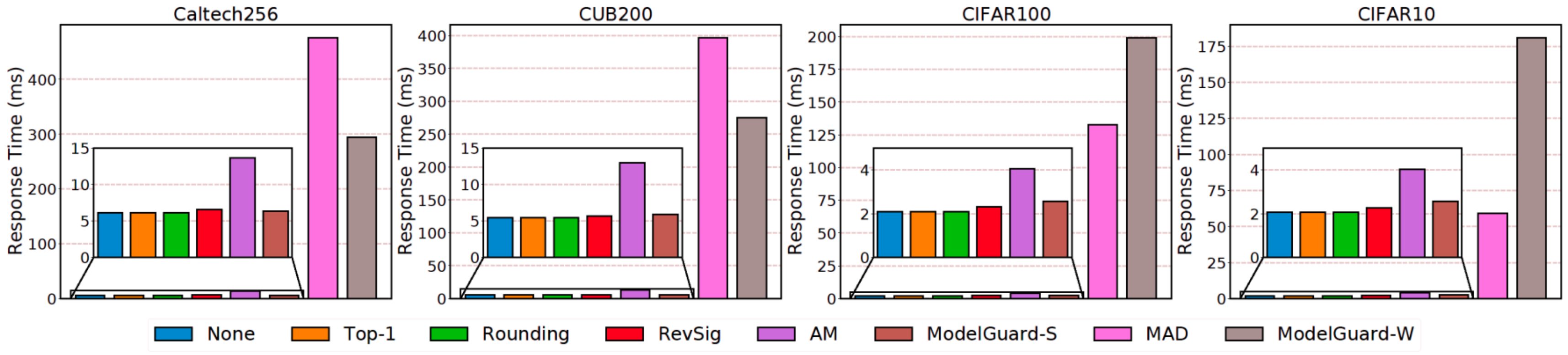


# Experimental Results

- How does mutual information influence the recovery and extraction?




- How efficient is ModelGuard?



# Conclusions

- We develop a general framework for model extraction attacks and defenses.
- We propose ModelGuard-W and ModelGuard-S.
- ModelGuard shows superiority compared with previous model extraction defense methods.

Duke

# Thank you!

- Q&A

- minxue.tang@duke.edu

Duke