# Towards Preserving Server-Side Privacy of On-Device Models

**Akanksha Atrey[1], Ritwik Sinha[2], Somdeb Sarkhel[2], Saayan Mitra[2], David Arbour[2], Akash V. Maharaj[3], Prashant Shenoy[1]**

[1]University of Massachusetts Amherst, [2]Adobe Research, [3]Adobe Inc.

## Introduction

On-device machine learning models are becoming ubiquitous in distributed service applications due to their advantages.

**Low latency**
No round trip to the centralized server

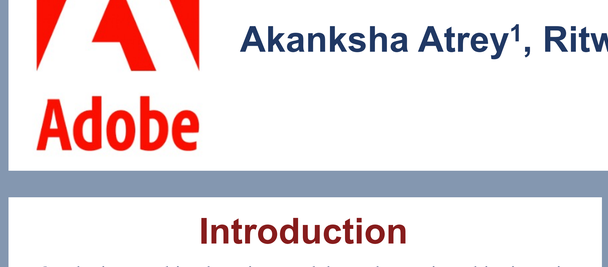
**User data privacy**
Processing happens on the device

**Works offline**
No network connectivity required

**Reduces cost**
No cost of servers or cloud compute cycles

**However they can leak sensitive information about the service provider!**

**Example:** Bank Application




**Most applications in privacy preserving ML have focused on user privacy. Our work deals with evaluating the extent to which a service provider's intellectual property can be exploited and misused in on-device models.**

## Contributions

In this work, we establish the importance of server-side privacy in on-device service models with the following contributions:

**C1. Developed** a taxonomy of on-device ML models focusing on distributed services.

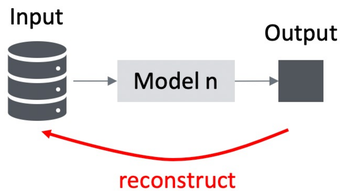
**C2. Proposed** multiple privacy attacks on on-device models and **evaluated** their efficacy on a real-world dataset.

**C3. Designed** preliminary ideas on how to protect the service provider's proprietary information embedded in on-device models.

## Privacy Attacks on On-Device Models

**We focus on server-side privacy attacks on on-device models which aim to recover the representations learned by model *M*.**
In this work, we focus on one-vs-all multi-classification models.

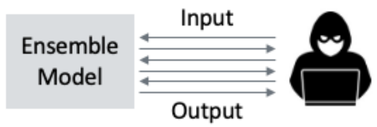
**Model Inversion Attack (white-box)**




**Known:** model, model input, model output

**Goal:** Exploit each binary classification model and reconstruct the input for each class using backpropagation

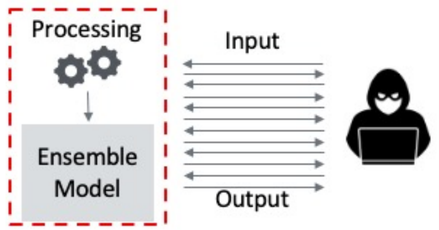
**Random Querying Attack (white-box)**




**Known:** ensemble model, model input, number of output classes

**Goal:** Iteratively query the features used by the model to recover the classes

**Large-Scale Querying Attack (black-box)**




**Known:** ensemble model, all features collected by the service provider

**Goal:** Iteratively query the whole feature set at scale to recover the classes

## Taxonomy of On-Device Models

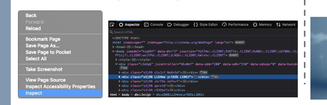
**We focus on two classes of on-device models, white-box and black-box. Since on-device models reside on a user's device can execute offline, certain aspects such as model input and output are automatically accessible.**

Descriptions of white-box (WB) and black-box (BB) on-device models with the components of the ML model accessible by a user.

| Model Type | Feature Space | | Output Space | Internals | |
|---|---|---|---|---|---|
| | All | Model Input | Model Output | Weights | Rep |
| WB | ✓ | ✓ | ✓ | ✓ | - |
| BB | ✓ | - | - | - | - |

**White-box examples:**

- Webpage inspect element
- Mobile application with memory/binary hacked



**Black-box examples:**

- Mobile application with ML model embedded into binary

## Evaluation of Server-Side Privacy Leakage

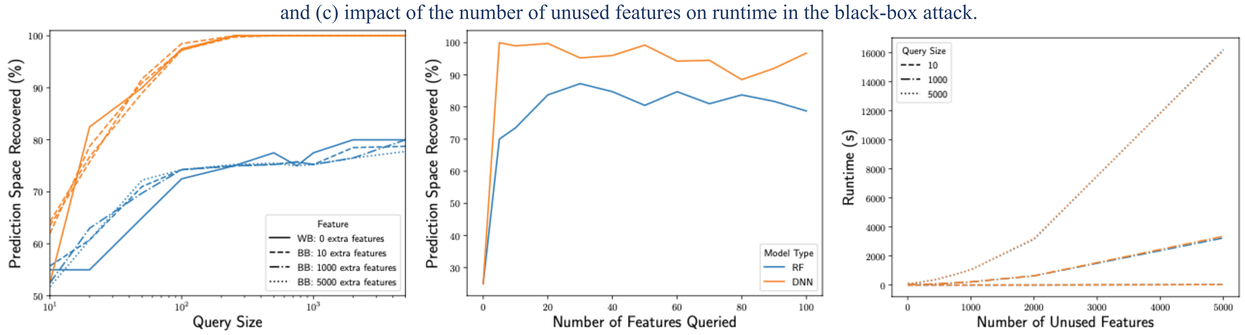
Difference between traditional serialized models and ONNX models (on-device) for random forests (RF) and deep neural networks (DNN).

| | | Size (KB) | Runtime (s) | Accuracy (%) |
|---|---|---|---|---|
| RF | Cloud | 11300 | 0.2642 | 99.24 |
| | ONNX | 5895 | 0.4176 | 99.24 |
| DNN | Cloud | 564 | 0.2578 | 98.84 |
| | ONNX | 560 | 0.0989 | 98.84 |

Runtimes of privacy attacks to recover full set of input to output mappings.

| Model Type | Attack | Runtime (s) |
|---|---|---|
| White-box | Model Inversion | 0.0531 |
| | Random Querying | 1.5323 |
| Black-box | Large Scale Querying | 55.4894 |

Results of evaluating the white-box (WB) and black-box (BB) querying attacks on RFs and DNNs:
(a) impact of varying query sizes on the attack efficacy; (b) impact of varying the number of features being queried in the white-box attack; and (c) impact of the number of unused features on runtime in the black-box attack.




(a) (b) (c)

**Takeaway**

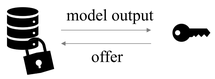
**Model inversion and limited querying attacks can recover the service provider's private information efficiently in white-box models. Although results demonstrate that black-box models are theoretically effective, they are not efficient due to higher runtimes. The simplicity of the attacks heighten the risk of server-side privacy leakage.**

## Preserving Server-Side Privacy

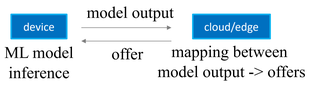
On-device models are more vulnerable to server-side privacy attacks as they have the ability of running offline.

### Countermeasures for White-Box Attacks

**Inference on Encrypted Models.** Conduct inference on encrypted data to avoid giving the adversary access to too much information.

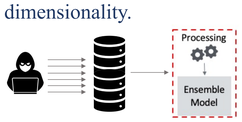



**Distributing Service.** Conduct inference on the device but keep the mapping of model output to tangible service on some central node. Each unique query will require connection to a central node.




### Countermeasures for Black-Box Attacks

**Increasing Dimensionality.** Collect more data. Since the subset of features used by the model are unknown, the adversary will have to search over a higher dimensionality.




**Query-based Model Degradation.** Degrade the weights of the model as more queries are conducted. Eventually the model will output random noise.