ReCon: Revealing and Controlling PII Leaks in Mobile Network Systems

Jingjing Ren, Martina Lindorfer, Ashwin Rao, Arnaud Legout, David Choffnes

(MobiSys '16)

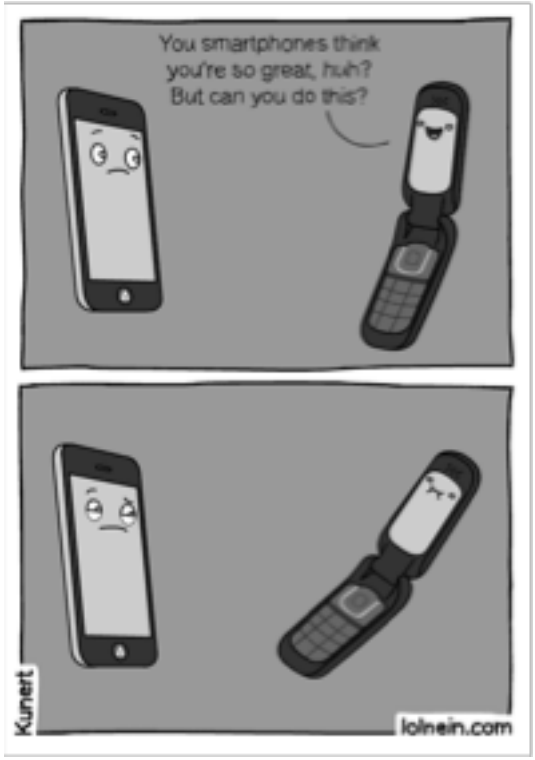
Presented by : Umar Farooq

CS 563

Fall 2018

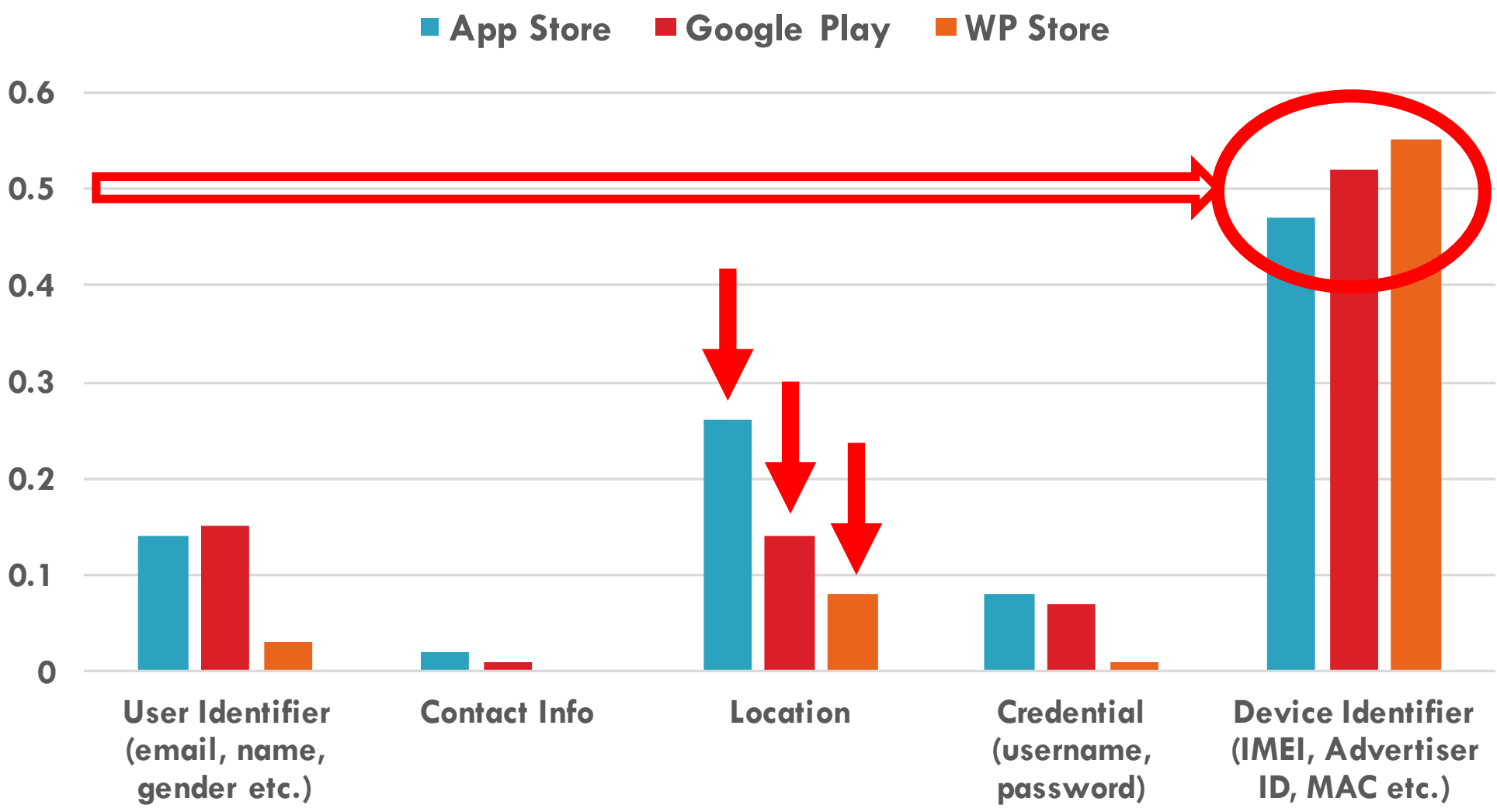
Mobile Phones today..

- ❑ Offer ubiquitous connectivity
- ❑ Equipped with a wide array of sensors
- ❑ Examples; GPS, camera, microphone etc.



Problems

- Personally identifiable info. (PII) leakage
  - Device Identifiers (IMEI, MAC address, etc.)
  - User Information (name, gender, contact info, etc.)
  - Location (GPS, zip code)
  - Credentials (?)
- Device Fingerprinting
- Cross Platform tracking

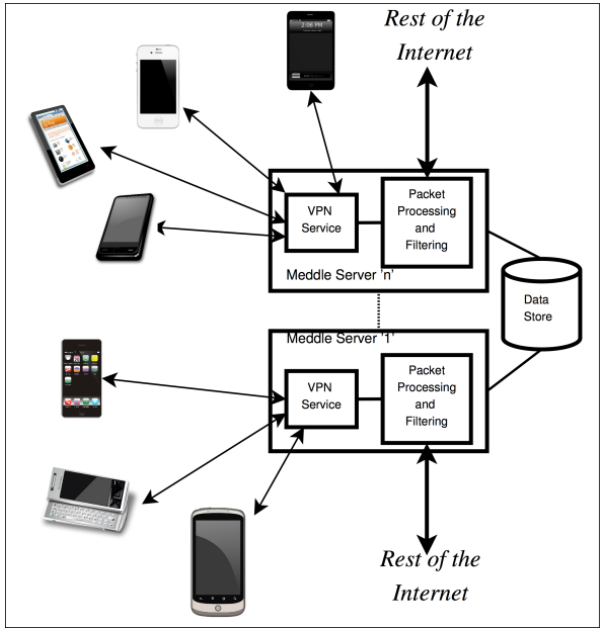

App Store
Google Play
WP Store
0.6
0.5
0.4
0.3
0.2
0.1
0
User Identifier (email, name, gender etc.)
Contact Info
Location
Credential (username, password)
Device Identifier (IMEI, Advertiser ID, MAC etc.)

Goals for this work

- ❑ Identify PII leakage without a priori information
- ❑ Provide users a platform to view potential PII leaks (i.e increase user visibility and transparency)

Approach..

- **Opportunity:** Almost all devices support **VPNs**
- Have a trusted third party system to audit network flows
  - Tunnel traffic to a controlled server (trusted server)
  - Measure, modify, shape or block - traffic with user opt in



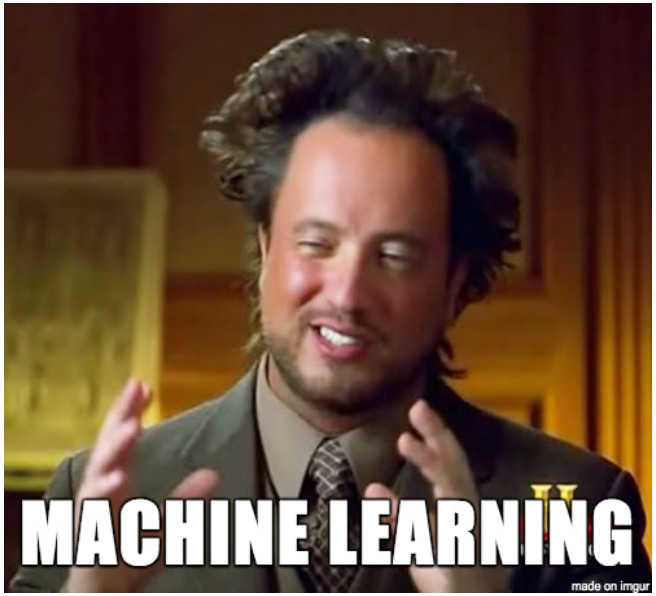
Why should this work?

So, what does a PII look like?

GET /index.html?id=12340;foo=bar;name=CS563@Illini;pass=jf3jNF#5h

How can we identify a PII leak?

Naïve approach: Pattern matching.



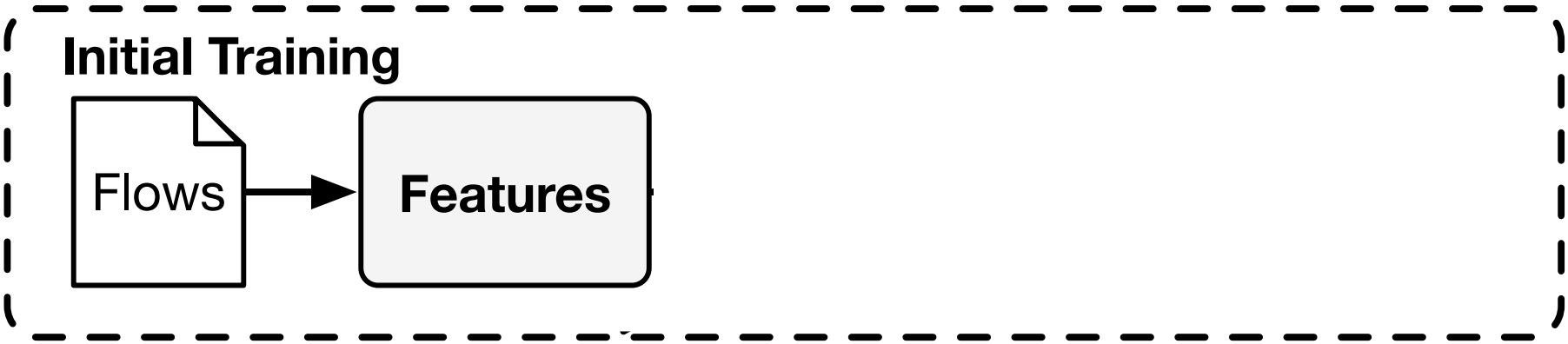
ReCon:

A system using supervised ML to accurately identify and control PII leaks from network traffic with crowdsource reinforcement.

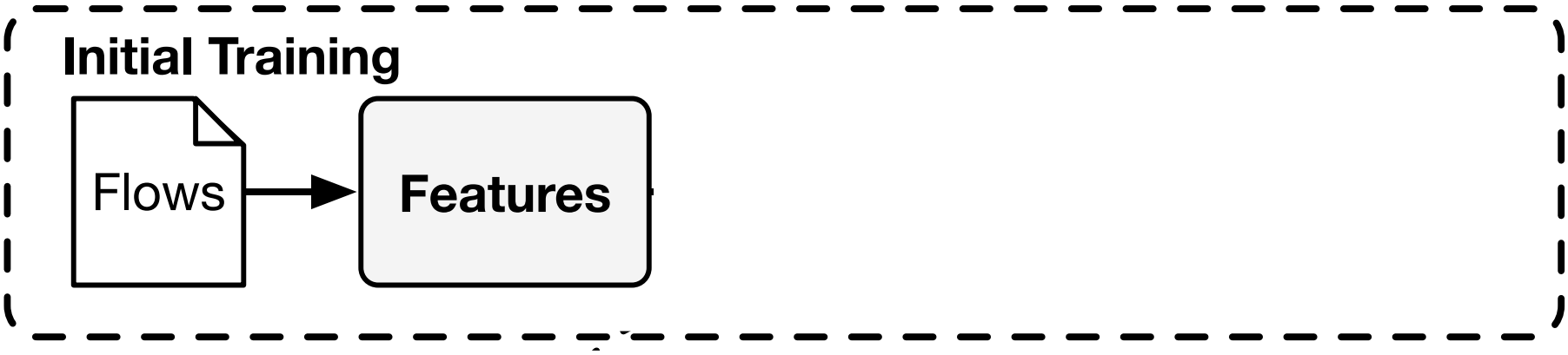
Automatically Identifying PII leaks

- Hypothesis: PII leaks have distinguishing characteristics
  - Is it just simple key/value pairs (e-g "user_id=563")
    - Nope, leads to high FPR (5.1%) and high FNR (18.8%).
- Need to **learn** structure of PII leaks.
- Approach: Build ML classifiers to reliably detect leaks.
  - Doesn't require knowing PII in advance
  - Resilient to changes in PII formats over time.

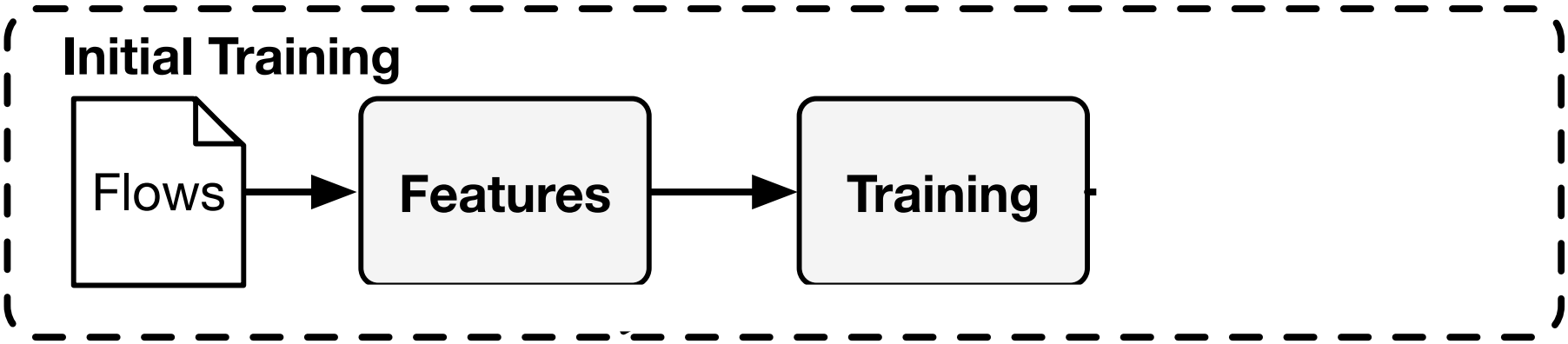
- Manual test: top 100 apps from each official store
- Automatic test: top 850 Android apps from a third party store



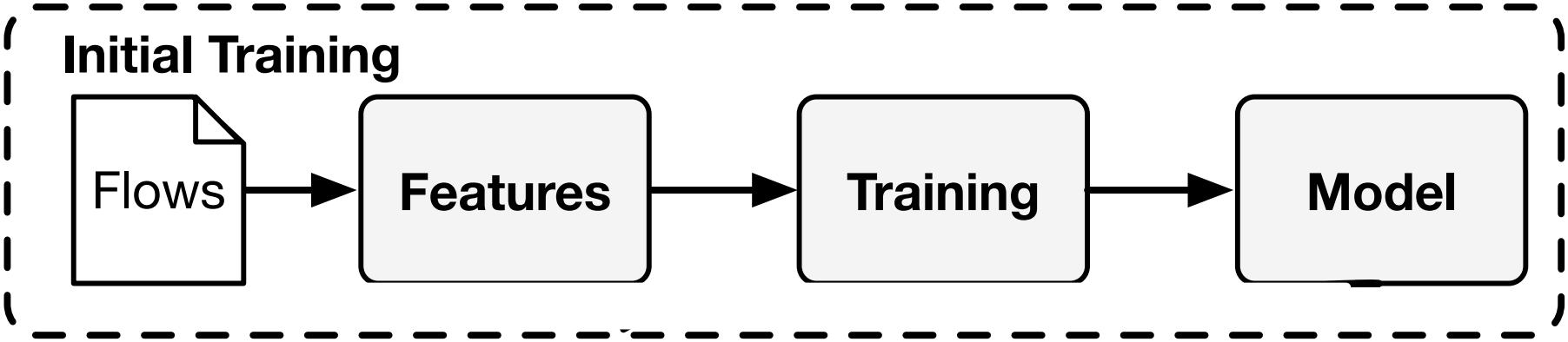
- Feature extraction: bag of words

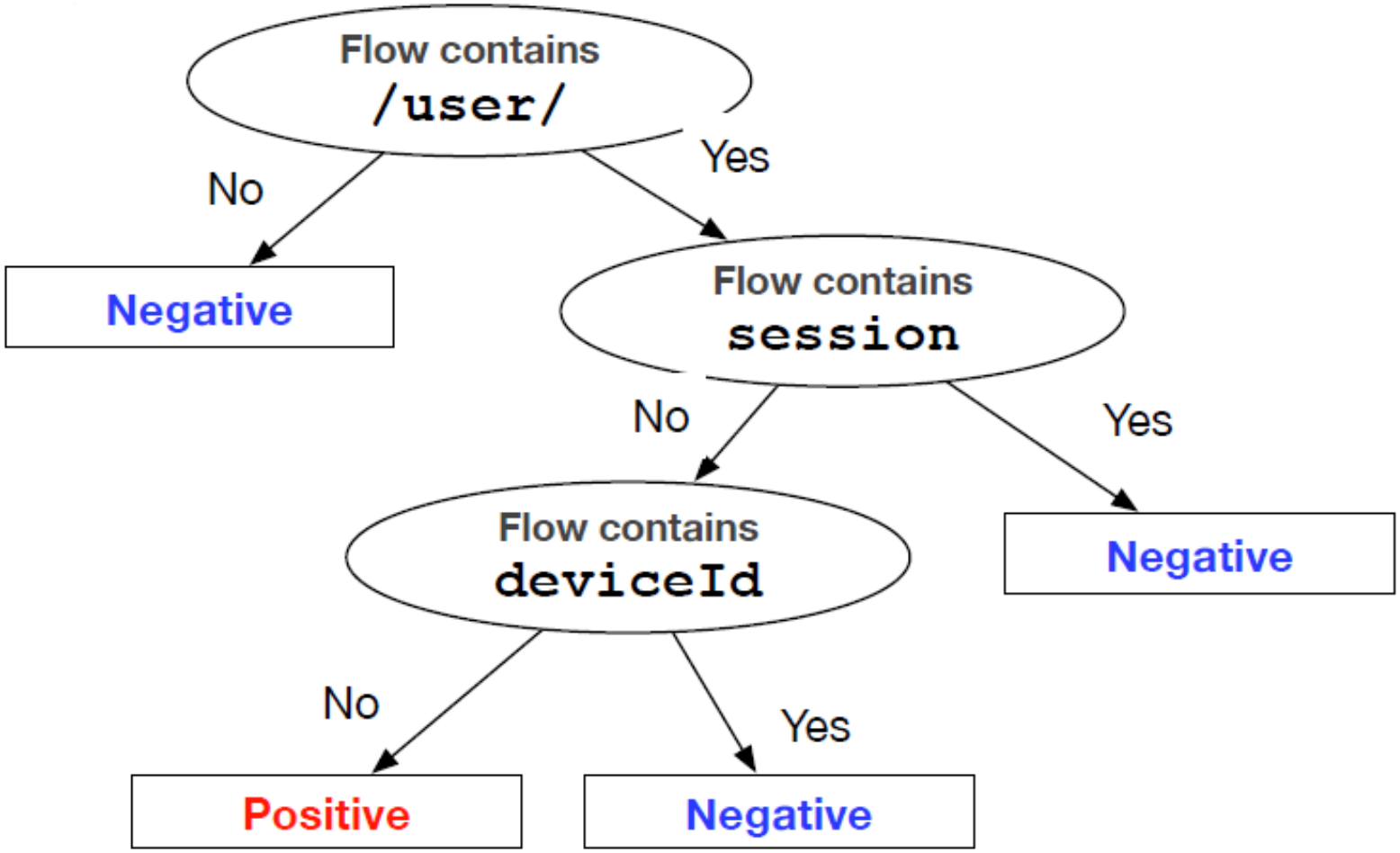


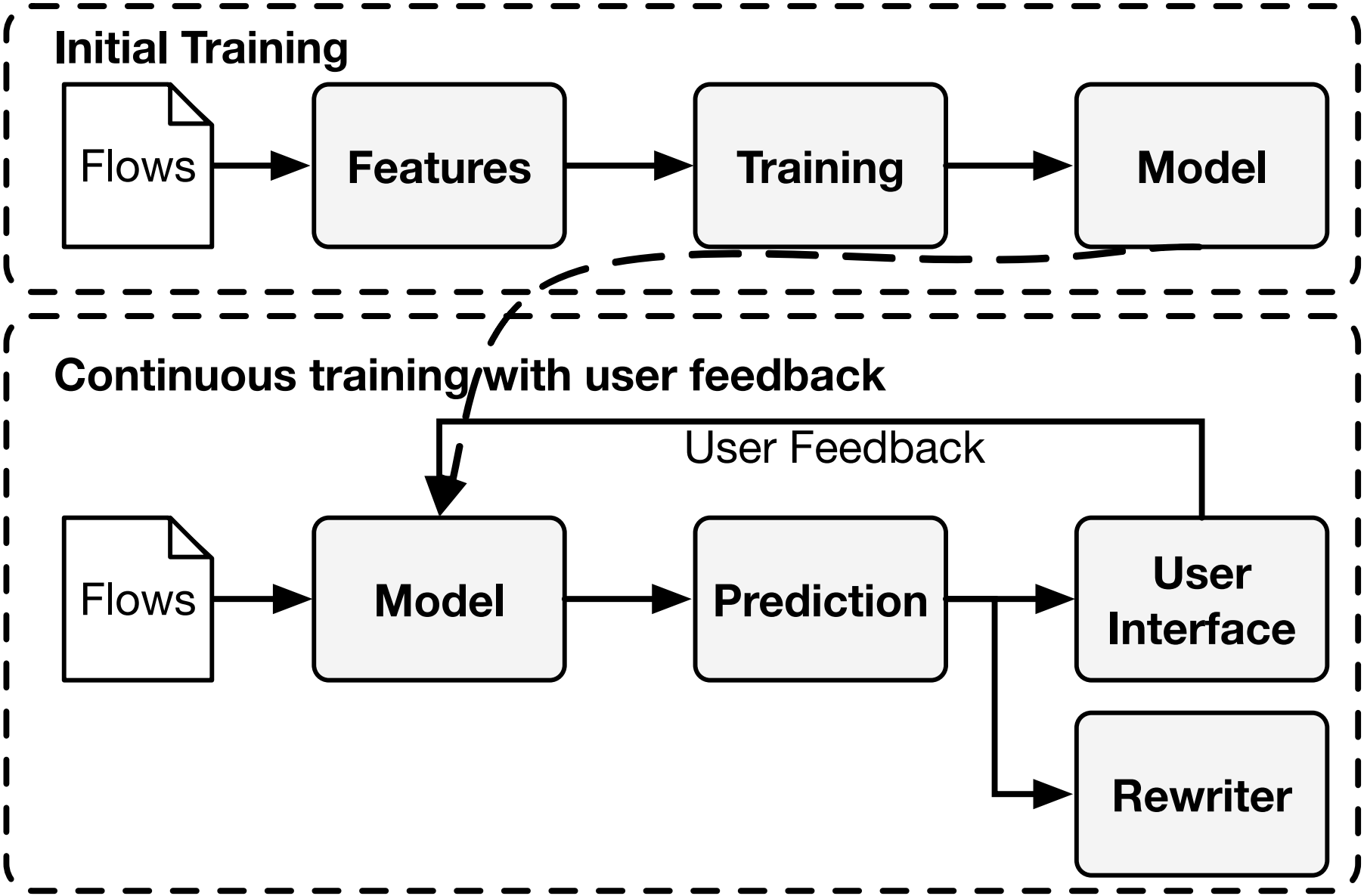
- Feature extraction: bag of words
- Use thresholds to remove infrequent or too frequent words



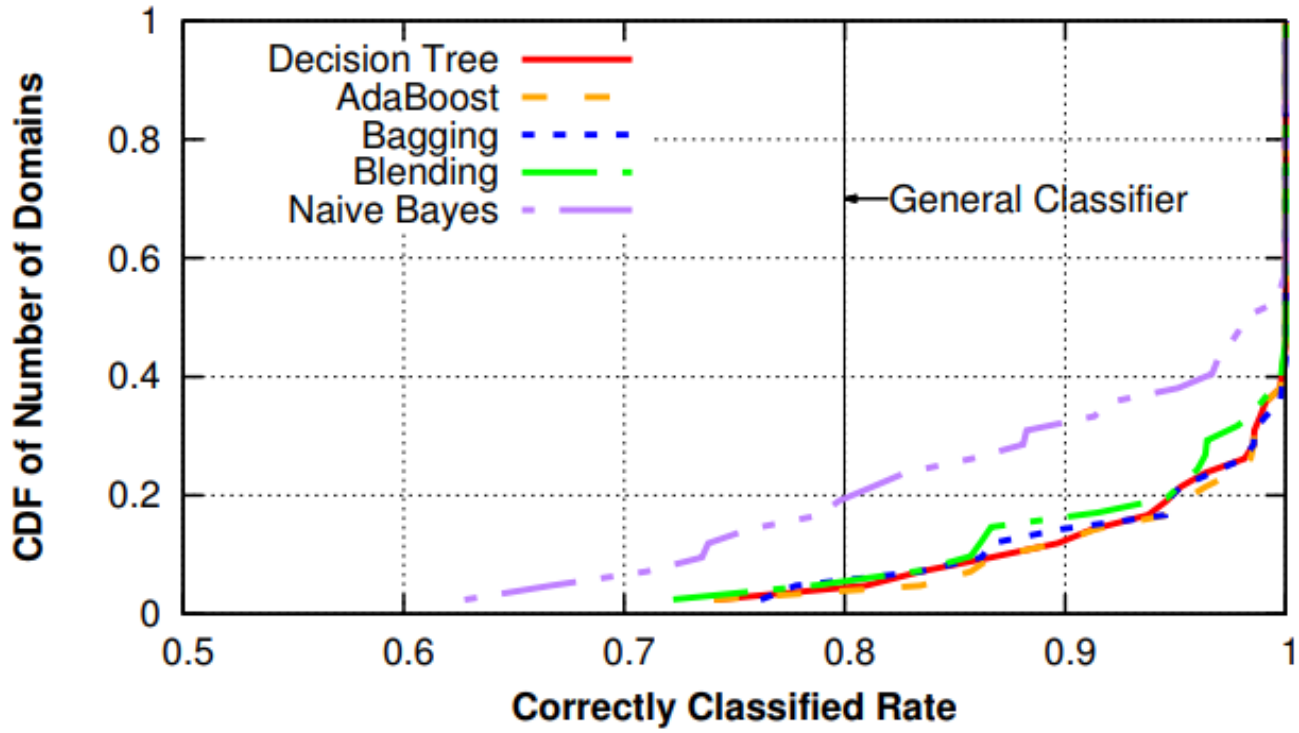
- Ground truth from the controlled experiments
- C4.5 decision tree
- Per-domain and per-OS classifier


Initial Training
Flows
Features
Training
Model


Flow contains
/user/
No
Yes
Negative
Flow contains
session
No
Yes
Flow contains
deviceId
Negative
No
Yes
Positive
Negative


Initial Training
Flows
Features
Training
Model
Continuous training with user feedback
User Feedback
Flows
Model
Prediction
User Interface
Rewriter

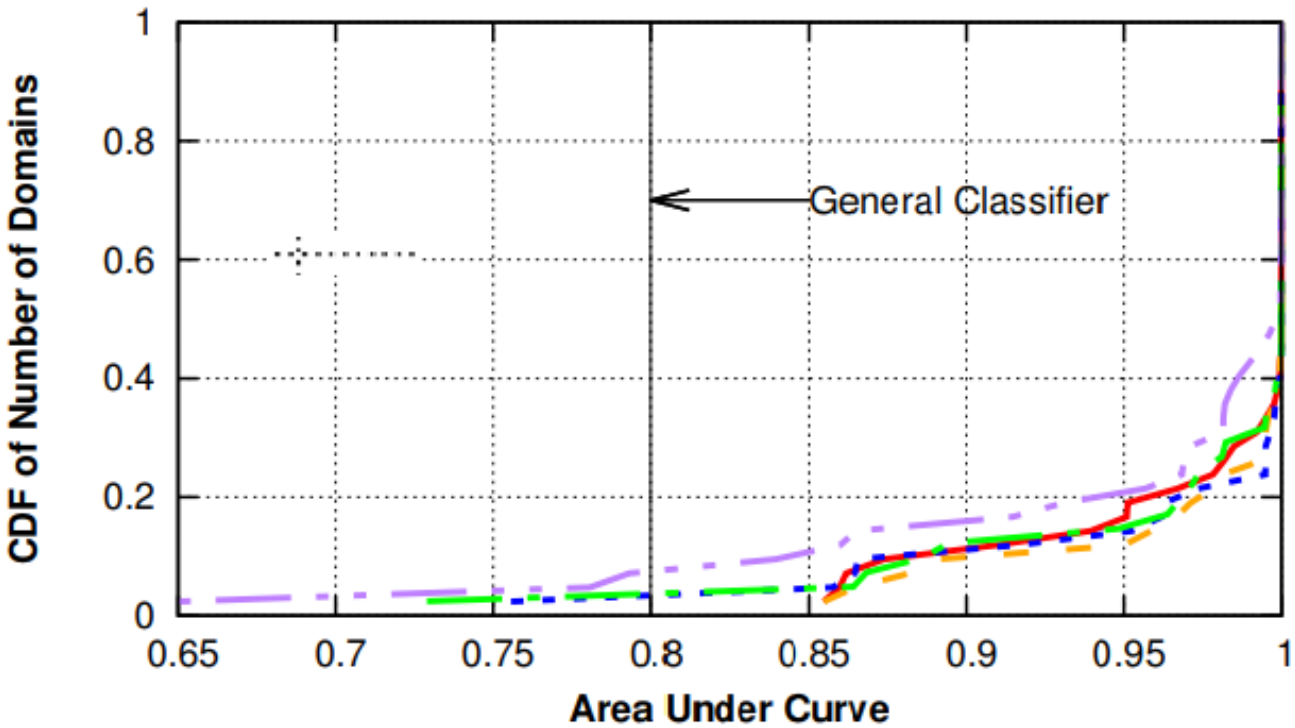
Evaluation – Accuracy (CCR)



CDF of per-domain-and-OS (PDAO) classifier accuracy

- DT outperforms Naïve Bayes
- Time: DT based ensembles take more time than a simple DT
- More than 95% accuracy per-domain-and per OS I
  - Greater than the General Classifier
  - 60% DTs zero error.

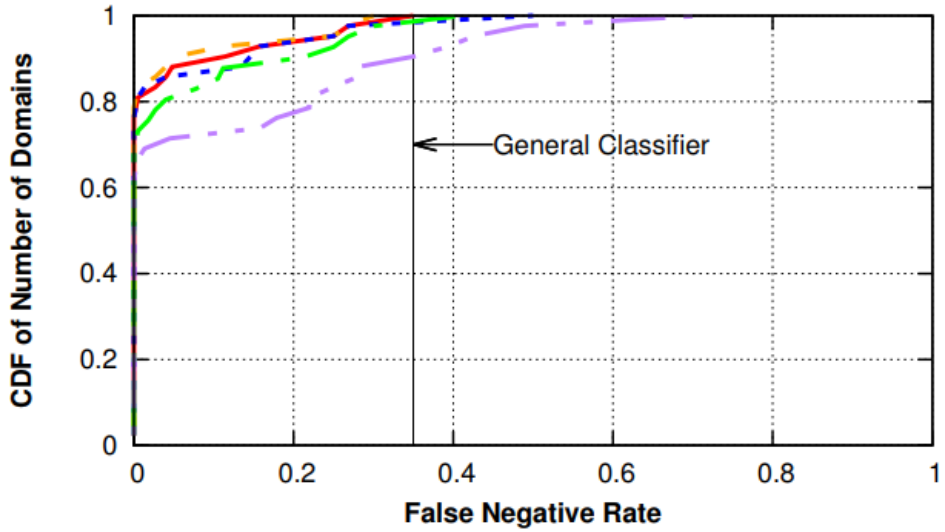
Evaluation – Accuracy (AUC)



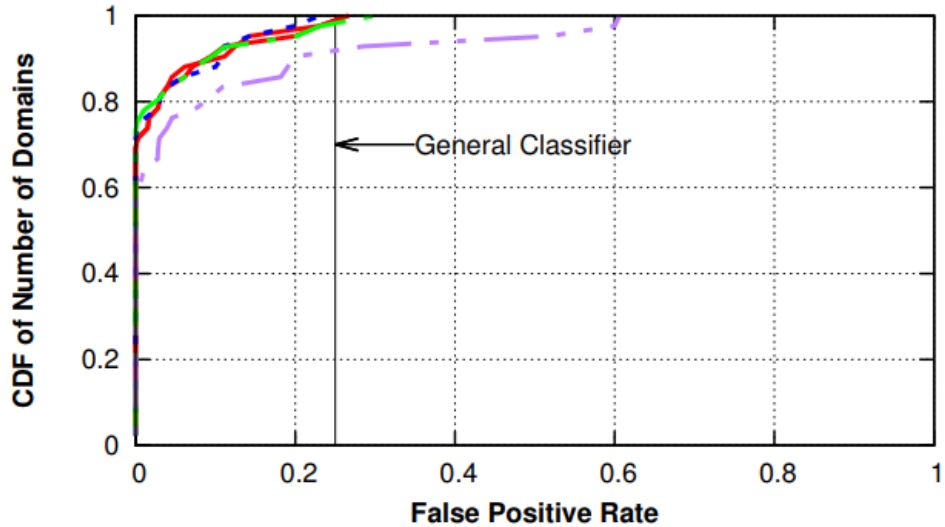
CDF of per-domain-and-OS (PDAO) classifier AUC

- Area under the curve (AUC) [0,1]
  - Demonstrates the predictive power of the classifier
- Most (67%) DT-based classifiers have AUC = 1

Evaluation – Accuracy (FNR and FPR)



(c) FNR



(d) FPR

CDF of per-domain-and-OS (PDAO) classifier accuracy

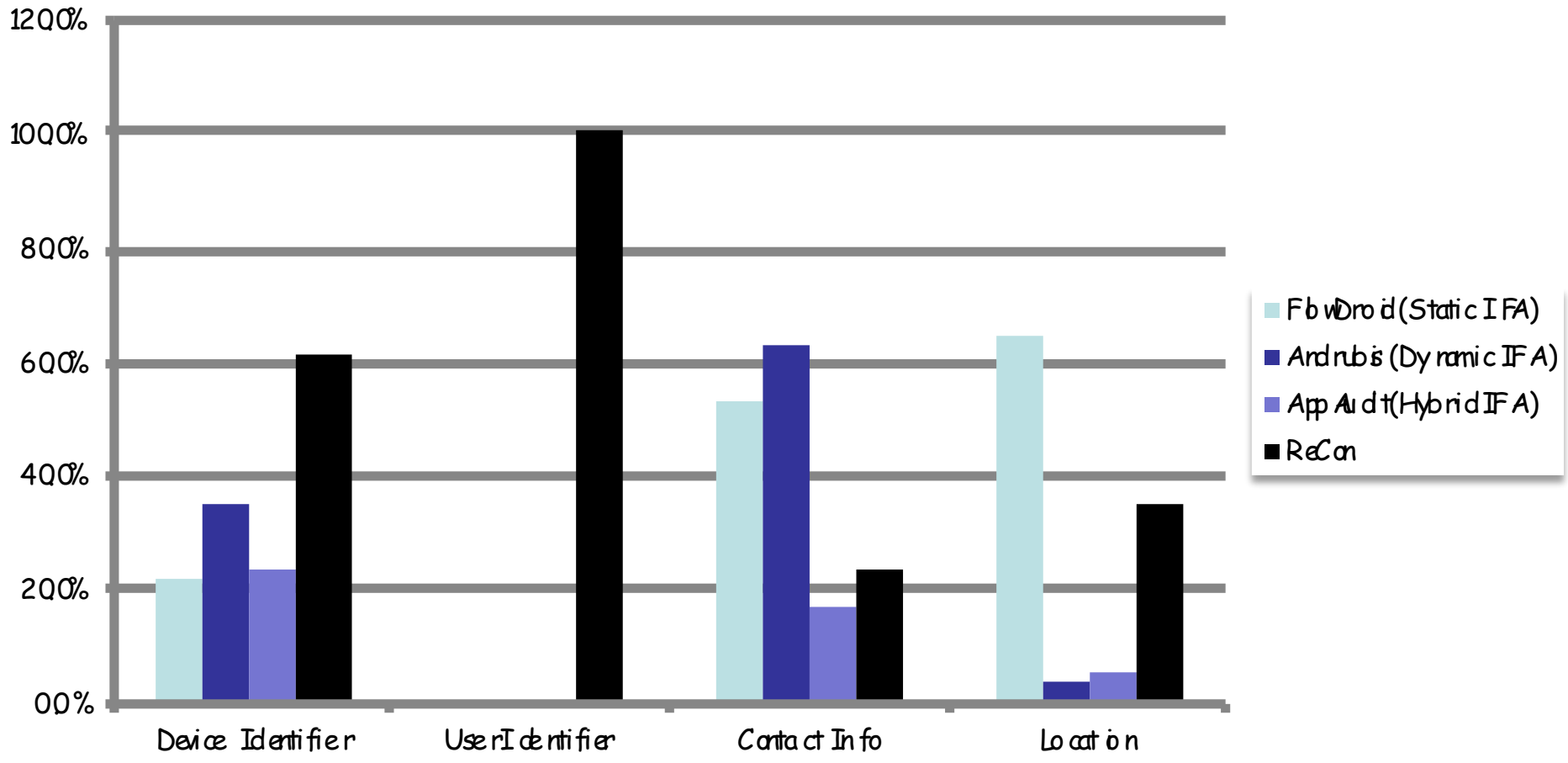
Most DT based classifiers have zero FPs (71.4%) and FNs (76.2%)

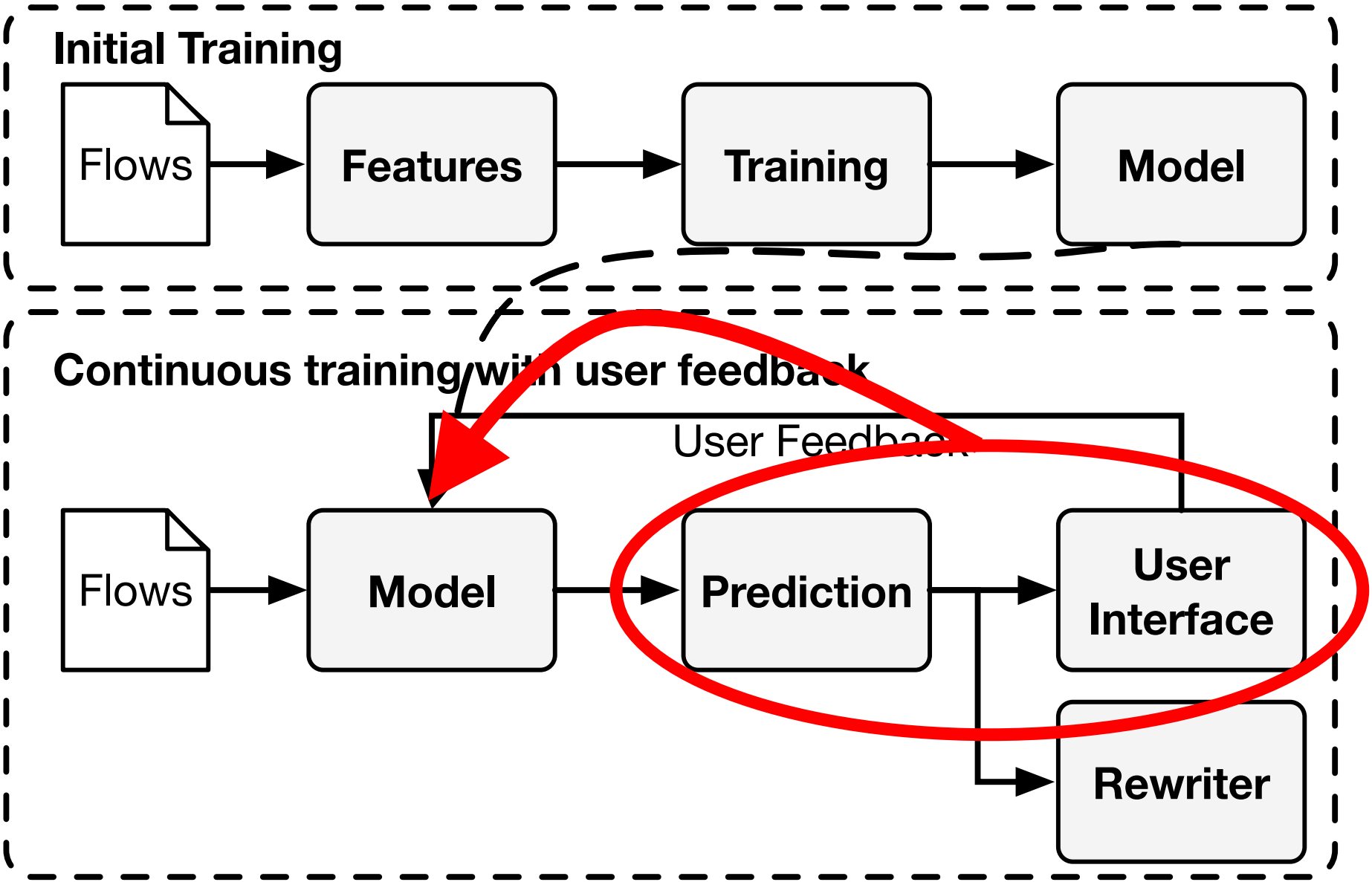
Evaluation – Comparison with IFA

- Information flow analysis (IFA)
  - Resilient to encrypted / obfuscated flow
    - Dynamic IFA: Andrubis
    - Static IFA: Flowdroid
    - Hybrid IFA: AppAudit

    Information flow analysis (IFA)
- Susceptible to **false positives,** but not false negatives

ReCon vs. static and dynamic analysis




Initial Training
Flows
Features
Training
Model
Continuous training with user feedback
User Feedback
Flows
Model
Prediction
User Interface
Rewriter

ReCon:

- The retraining phase is important
  - FP decreased by 92%
  - FN increased by 0.5%

ReCon in the wild

- 239 users in March 2016 (IRB approved)
- 137 iOS, 108 Android devices
- 14,101 PII found and 6,747 confirmed by users
- 21 apps exposing passwords in plaintext
  - Used by millions (Match, Epocrates)
  - Responsibly disclosed

Discussion

- Challenges
  - Encrypted Traffic (totally reliant on plaintext traffic)
  - 10-fold cross validation, does it help?
    - 2.2% FP and 3.5% FN, but what about overfitting?
    - Network flows too diverse, is the model generalizable?
  - Can miss out on PII leaks (FN) if model not trained for that class of PII. Standard program analysis susceptible to **false positives,** but not false negatives

Discussion - continued

- Can we use this approach for IoT devices?
  - Device Identification?
  - PII leakage?
  - Monitor if IoT devices “talk” to themselves?

Questions?