Man vs. Machine: Adversarial Detection of Malicious Crowdsourcing Workers

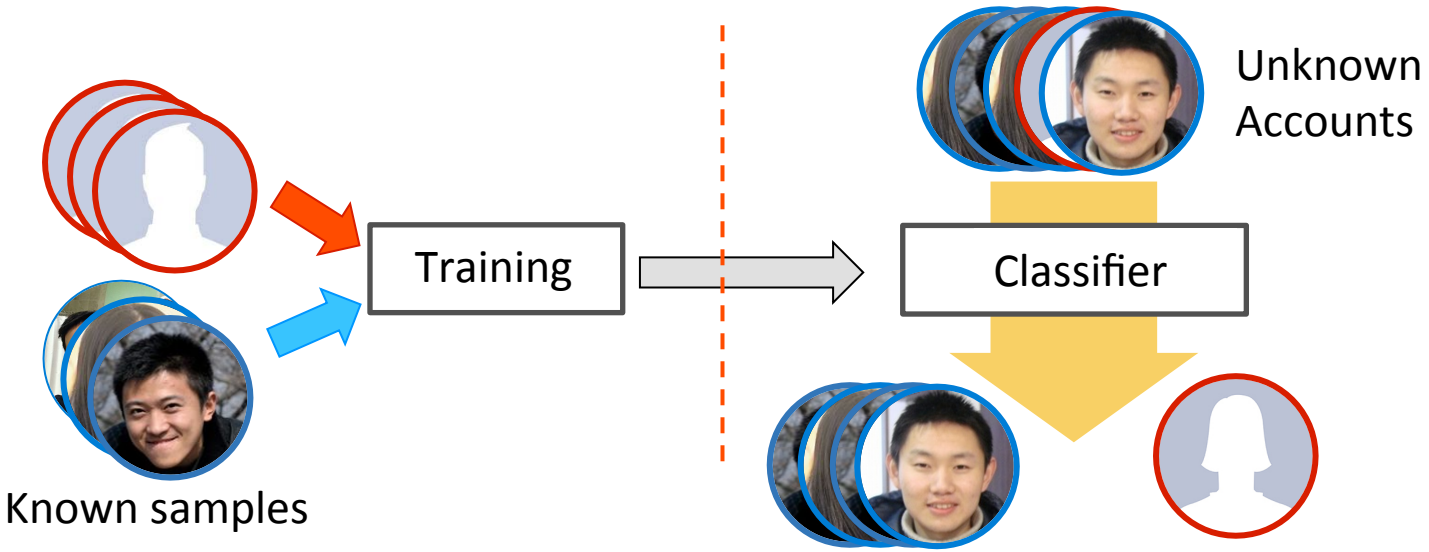
Gang Wang, Tianyi Wang, Haitao Zheng, Ben Y. Zhao

UC Santa Barbara

gangw@cs.ucsb.edu

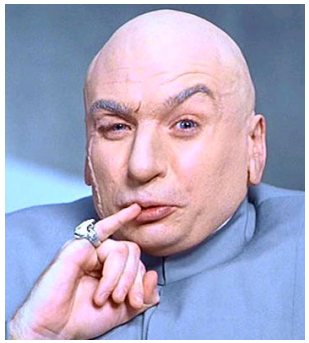
Machine Learning for Security

- Machine learning (ML) to solve security problems
  - Email spam detection
  - Intrusion/malware detection
  - Authentication
  - Identifying fraudulent accounts (Sybils) and content
- Example: ML for Sybil detection in social networks



Adversarial Machine Learning

- Key vulnerabilities of machine learning systems
  - ML models derived from fixed datasets
  - Assuming similar distribution of training and real-world data

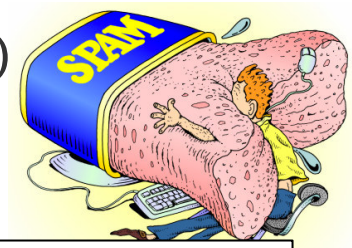


- Strong adversaries in ML systems
  - Aware of usage, reverse engineering ML systems
  - Adaptive evasion, temper with the trained model

- Practical adversarial attacks
  - What are the practical constrains for adversaries?
  - With constrains, how effective are adversarial attacks?

Context: Malicious Crowdsourcing

- New threat: malicious crowdsourcing = crowdturfing
  - Hiring a large army of real users for malicious attacks
  - Fake customer reviews, rumors, targeted spam
  - Most existing defenses fail against real users (CAPTCHA)



Samsung Fined For Paying People to Criticize HTC's P

Oct. 24, 2013

FAST COMPANY

MECHANICAL TURK'S UNSAVORY SIDE EFFECT: MASSIVE

Shady

Monday, May

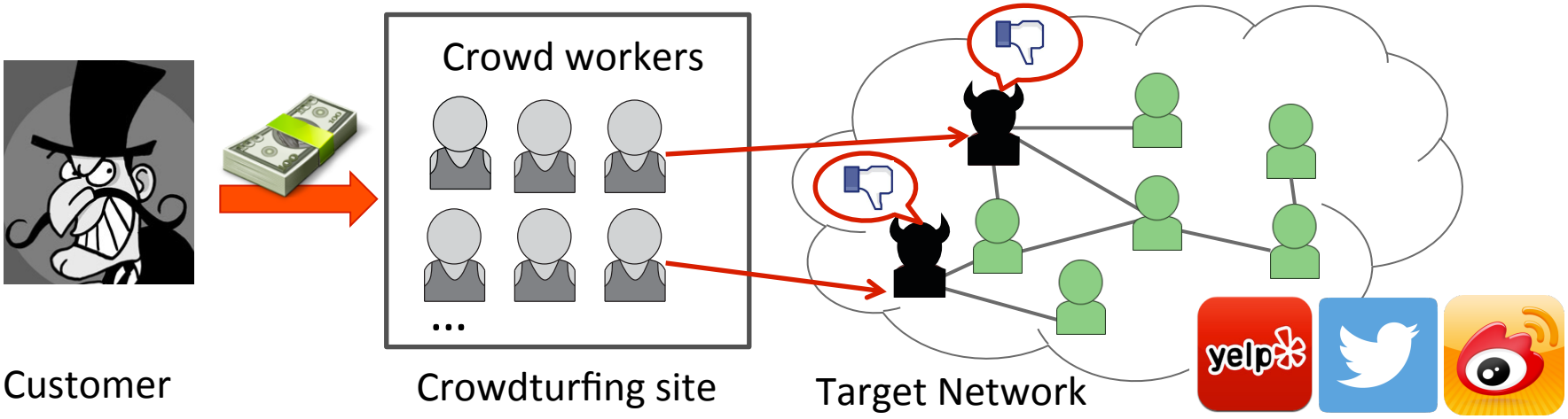
Vietnam admits deploying bloggers to support government

Republicans Use Crowdsourcing to Attack Obama Campaign

By MICHAEL D. SHEAR MAY 4, 2012 8:11 AM 35 Comments

Online Crowdturfing Systems

- Online crowdturfing systems (services)
  - Connect customers with online users willing to spam for money
  - Sites located across the glob, e.g. China, US, India



- Crowdturfing in China
  - Largest crowdturfing sites: ZhuBaJie (ZBJ) and SanDaHa (SDH)
  - Million-dollar industry, tens of millions of tasks finished

Machine Learning vs. Crowdturfing

- Machine learning to detect crowdturfing workers
  - Simple methods usually fail (e.g. CAPTCHA, rate limit)
  - Machine learning: more sophisticated modeling on user behaviors
    - "You are how you click" [USENIX'13]

- Perfect context to study adversarial machine learning
  1. Highly adaptive workers seeking evasion
  2. Crowdturfing site admins tamper with training data by changing all worker behaviors

Goals and Questions

- Our goals
  - Develop defense against crowdturfing on Weibo (Chinese Twitter)
  - Understand the impact of adversarial countermeasures and the robustness of machine learning classifiers

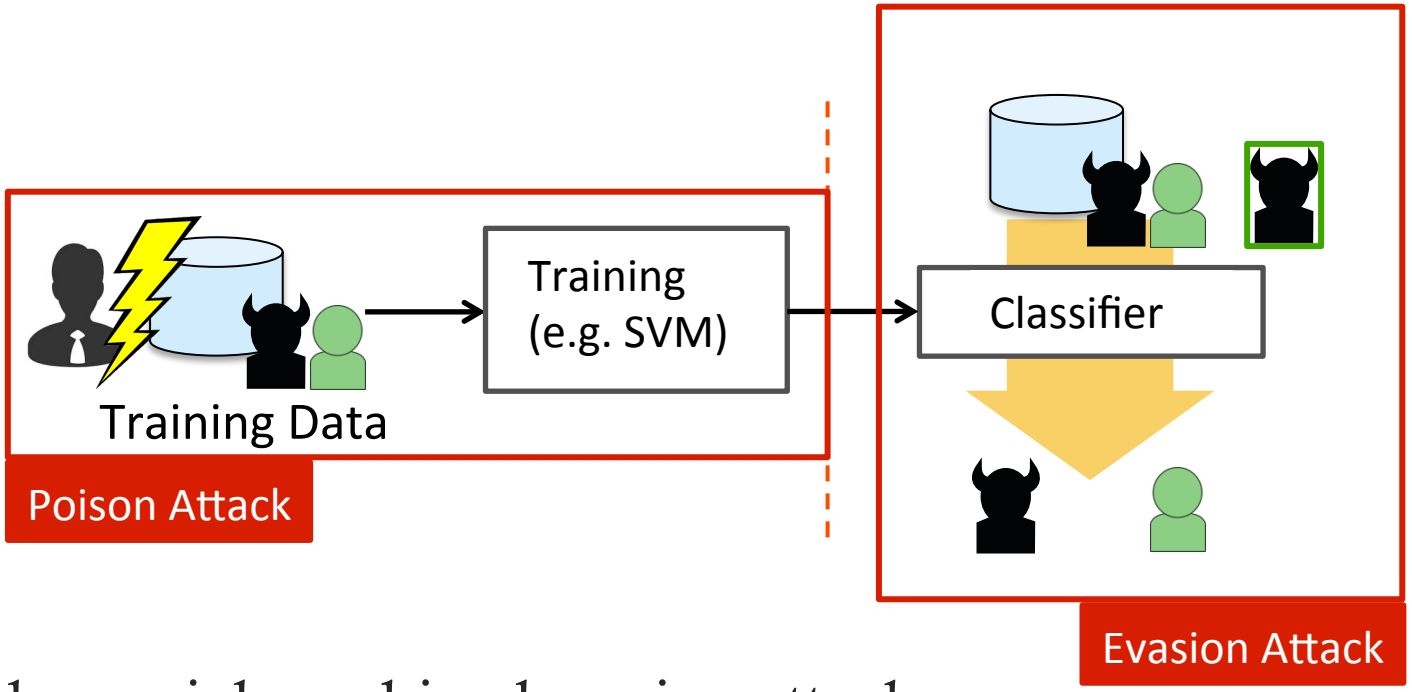
- Key questions
  - What ML algorithms can accurately detect crowdturfing workers?
  - What are possible ways for adversaries to evade classifiers?
  - Can adversaries attack ML models by tampering with training data?

Outline

- ~~Motivation~~
- Detection of Crowdturfing
- Adversarial Machine Learning Attacks
- Conclusion

Methodology

- Detect crowdturf workers on Weibo



- Adversarial machine learning attacks
  - Evasion Attack: workers evade classifiers
  - Poisoning Attack: crowdturfing admins tamper with training data

Ground-truth Dataset

- Crowdturfing campaigns targeting Weibo
  - Two largest crowdturfing sites ZBJ and SDH
  - Complete historical transaction records for 3 years (2009-2013)
  - 20,416 Weibo campaigns: > 1M tasks, 28,947 Weibo accounts

- Collect Weibo profiles and their latest tweets
  - **Workers:** 28K Weibo accounts used by ZBJ and SDH workers
  - **Baseline users:** snowball sampled 371K baseline users

Features to Detect Crowd-workers

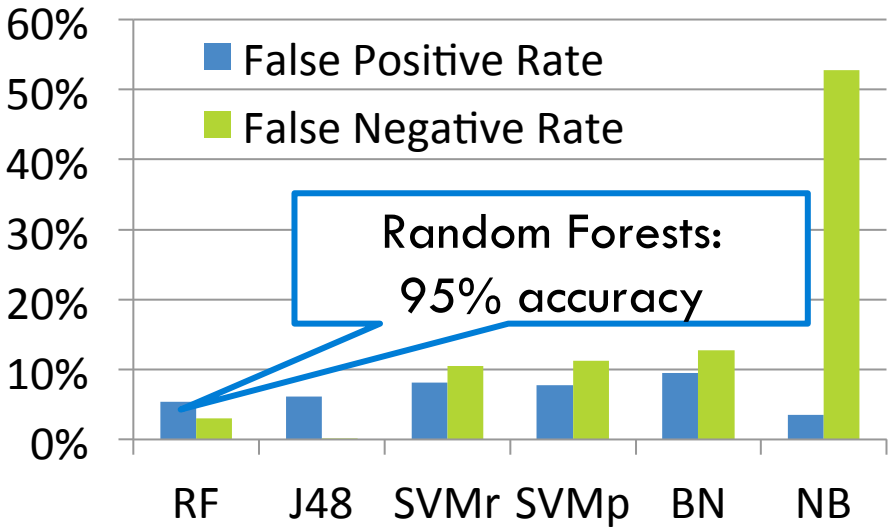
- Search for behavioral features to detect workers
- Observations
  - Aged, well established accounts
  - Balanced follower-followee ratio
  - Using cover traffic
- Final set of useful features: 35
  - Baseline profile fields (9)
  - User interaction (comment, retweet) (8)
  - Tweeting device and client (5)
  - Burstiness of tweeting (12)
  - Periodical patterns (1)

Active at posting but have less bidirectional interactions

Task-driven nature

Performance of Classifiers

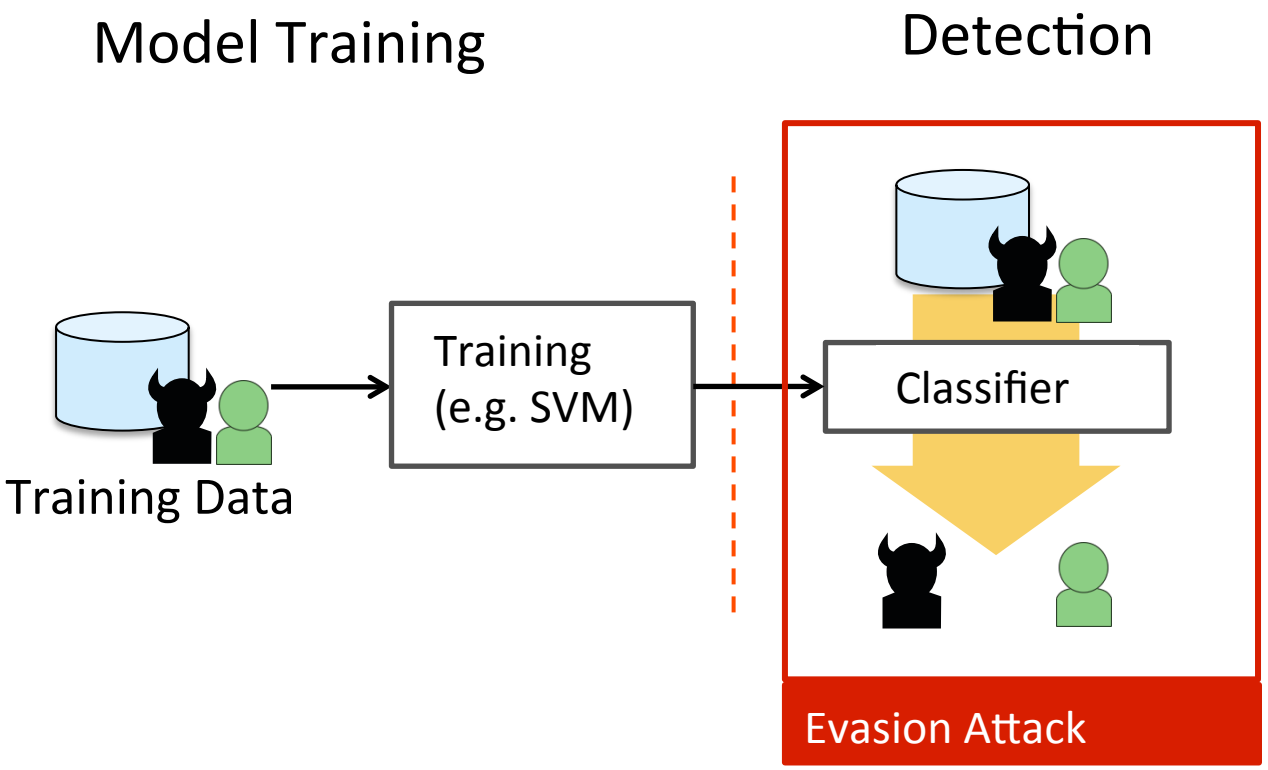
- Building classifiers on ground-truth data
  - Random Forests (RF)
  - Decision Tree (J48)
  - SVM radius kernel (SVMr)
  - SVM polynomial (SVMp)
  - Naïve Bayes (NB)
  - Bayes Network (BN)



- Classifiers dedicated to detect "professional" workers
  - Workers who performed > 100 tasks
  - Responsible for 90% of total spam
  - More accurate to detect the professionals ➔ 99% accuracy

Outline

- ~~Motivation~~
- ~~Detection of Crowdturfing~~
- Adversarial Machine Learning Attacks
  - Evasion attack
  - Poisoning attack
- Conclusion


Model Training
Detection
Training Data
Training
(e.g. SVM)
Classifier
Evasion Attack

Attack #1: Adversarial Evasion

- Individual workers as adversaries
  - Workers seek to evade a classifier by mimicking normal users
  - Identify the key set of features to modify for evasion

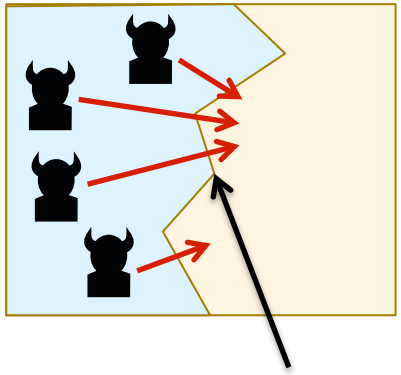
- Attack strategy depends on worker's knowledge on classifier
  - Learning algorithm, feature space, training data

- What knowledge is practically available? How does different knowledge level impact workers' evasion?

A Set of Evasion Models

- Optimal evasion scenarios
  - **Per-worker optimal:** Each worker has perfect knowledge about the classifier
  - **Global optimal:** knows the direction of the boundary
  - **Feature-aware evasion:** knows feature ranking

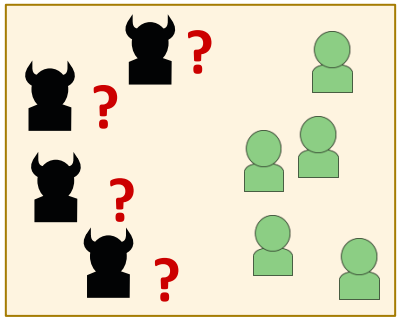
Optimal



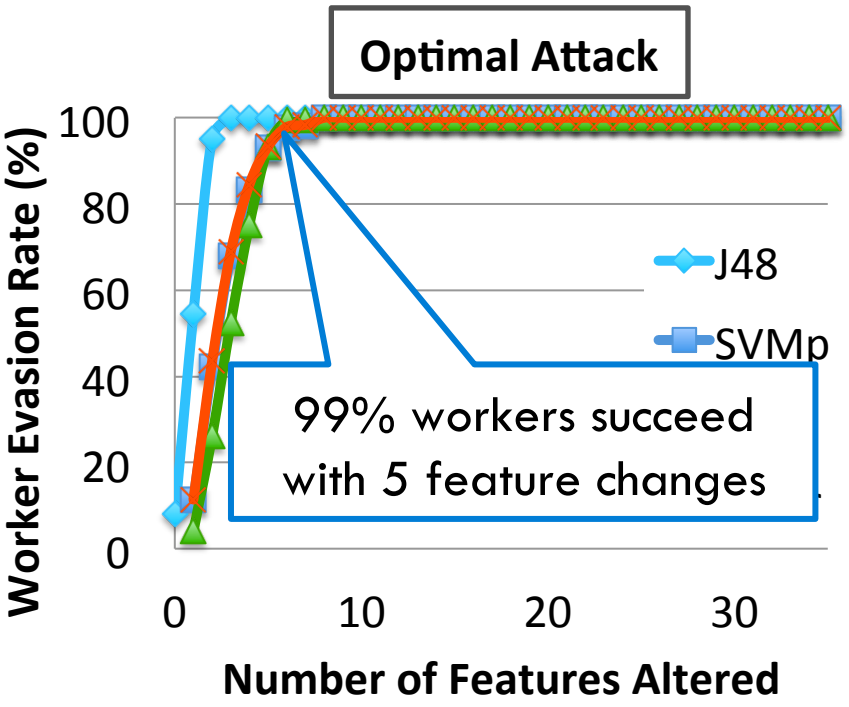
Classification boundary

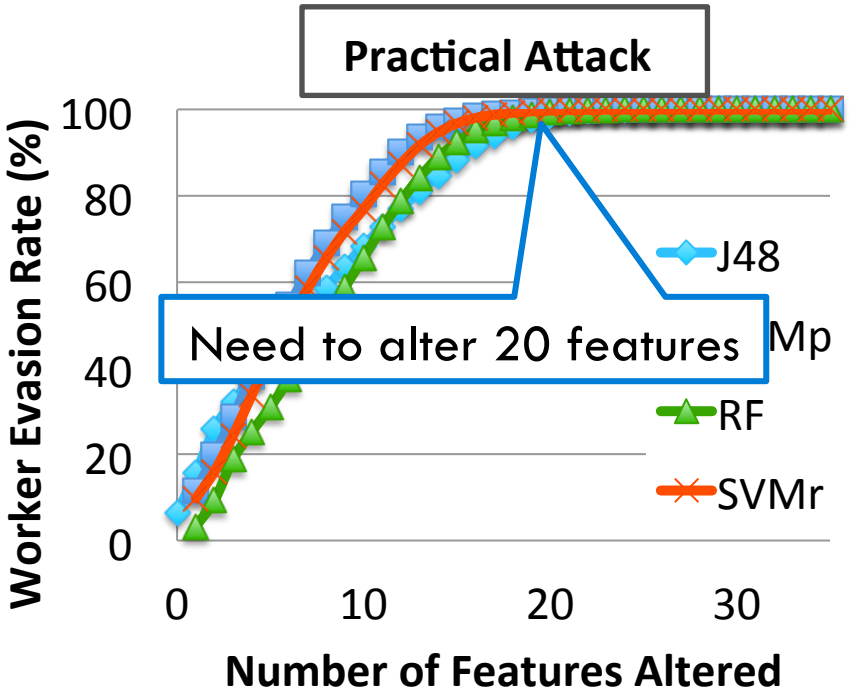
- Practical evasion scenario
  - Only knows normal users statistics
  - Estimate which of their features are most "abnormal"

Practical

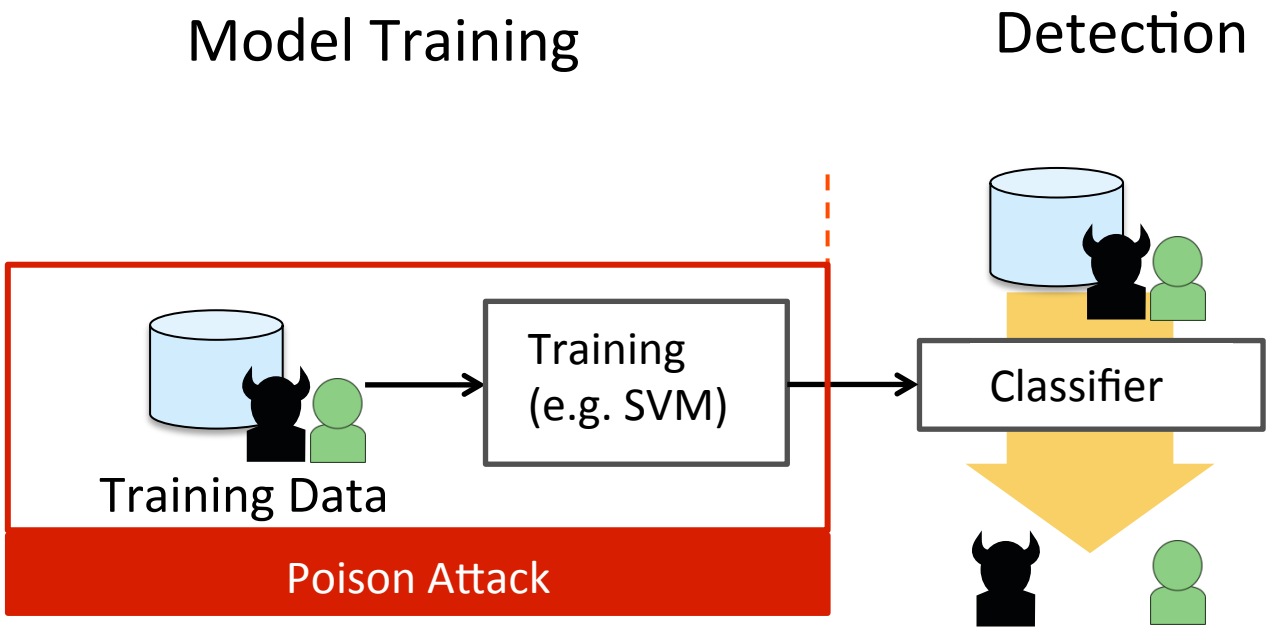


Evasion Attack Results



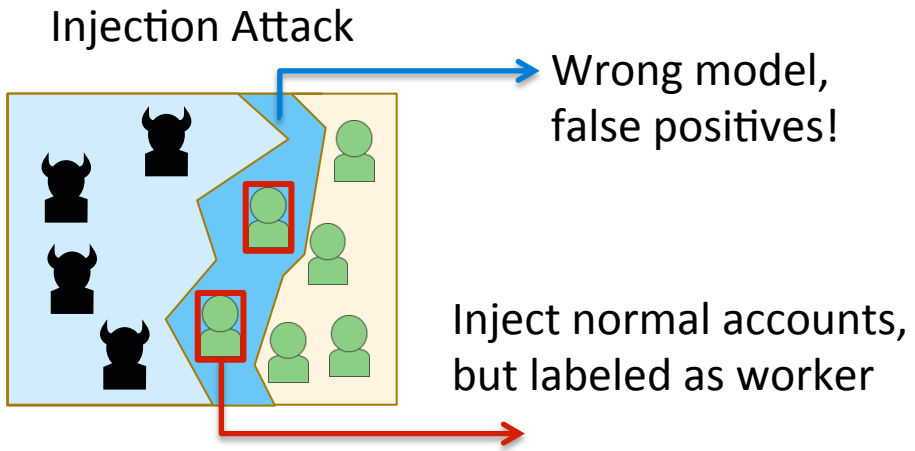


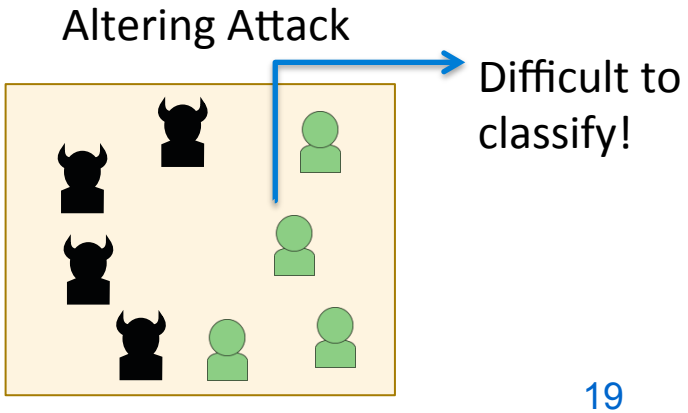
- Evasion is highly effective with perfect knowledge, but less effective in practice
- No single classifier is robust against evasion.
  The key is to limit adversaries' knowledge


Model Training
Detection
Training Data
Training
(e.g. SVM)
Classifier
Poison Attack

Attack #2: Poisoning Attack

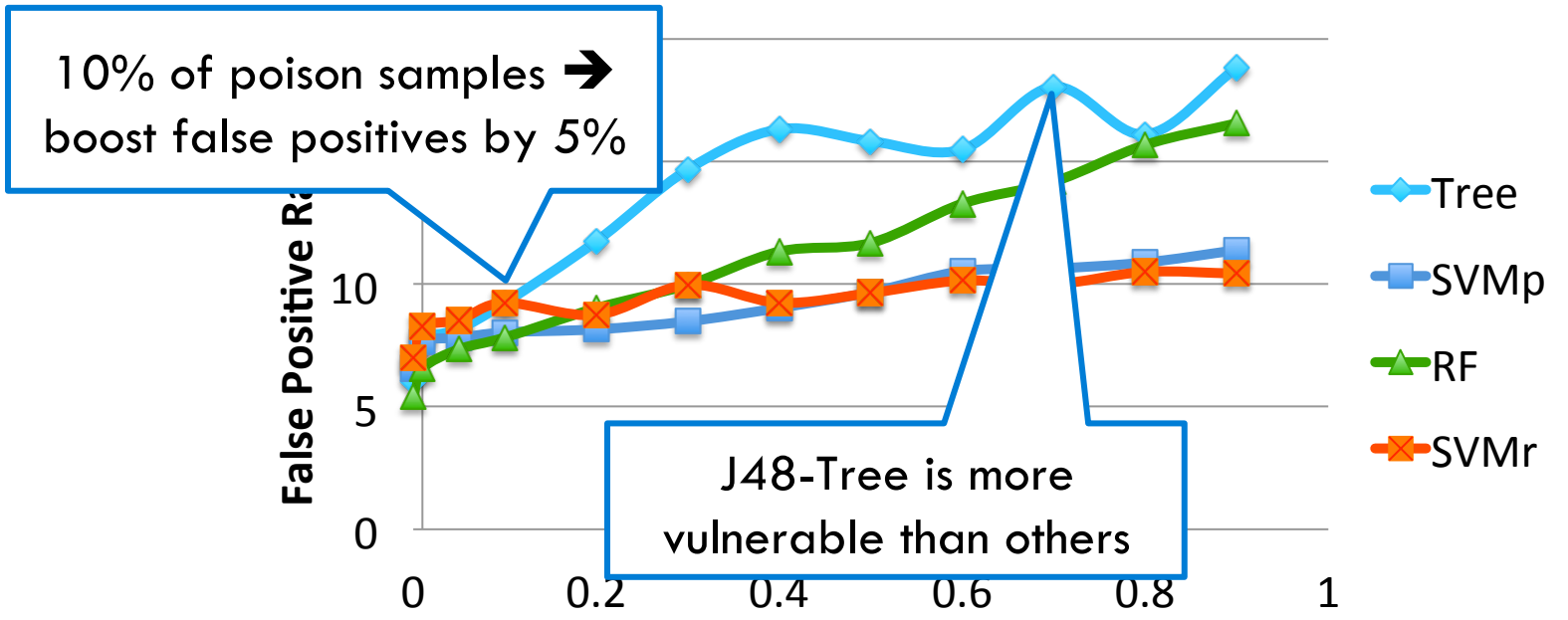
- Crowdturfing site admins as adversaries
  - Highly motivated to protect their workers, centrally control workers
  - Tamper with the training data to manipulate model training
- Two practical poisoning methods
  - Inject mislabeled samples to training data ➔ wrong classifier
  - Alter worker behaviors uniformly by enforcing system policies ➔ harder to train accurate classifiers





Injecting Poison Samples

- Injecting benign accounts as "workers" into training data
  - Aim to trigger false positives during detection



Poisoning attack is highly effective
More accurate classifier can be more vulnerable

Discussion

- Key observations
  - **Accurate** machine learning classifiers can be highly vulnerable
  - No single classifier excels in all attack scenarios, **Random Forests** and **SVM** are more robust than **Decision Tree**.
  - Adversarial attack impact highly depends on adversaries' knowledge

- Moving forward: improve robustness of ML classifiers
  - Multiple classifier in one detector (ensemble learning)
  - Adversarial analysis in unsupervised learning

Thank You!
Questions?