Knock It Off: Profiling the Online Storefronts of Counterfeit Merchandise

Matt Der
Lawrence Saul
Stefan Savage
Geoff Voelker

KDD 2014
Problem in a nutshell

Behind the many online storefronts for counterfeit goods lurk a small handful of sophisticated criminal operations.

How can automated, data-driven methods help to identify and target them?
Counterfeit online storefronts
Counterfeit online storefronts
Counterfeit online storefronts
Who is running the store?

“affiliate programs”
What is an affiliate program?

• Illegitimate business that sells counterfeit goods: 
  millions of $$ of revenue per month
• Manage Web sites that serve as online storefronts
• Enlist spammers to advertise their storefronts via bulk email
• Contract out payment & fulfillment services
Key insight

100s of thousands of storefronts
Key insight

100s of thousands of storefronts

dozens of affiliate programs
• **Goal**: classify storefronts by affiliate program; disrupt their operation to undermine spam business model

• **Approach**: HTML bag-of-words, nearest neighbor classification (automated system)

• **Takeaway**: highly accurate — even with simple classifier & limited labeled examples
Challenges
Challenges

1. Web pages that render very differently are often linked to the same affiliate program
Challenges

2. Difficulty in acquiring training data
2. Difficulty in acquiring training data
2. Difficulty in acquiring training data
Challenges

2. Difficulty in acquiring training data
Challenges

2. Difficulty in acquiring training data
Challenges

2. Difficulty in acquiring training data
2. Difficulty in acquiring training data

expert labeling is slow & tedious!
A role for machine learning

- Security experts labeled **178k storefronts**
  - Estimated **~200 man-hours**
  - Painstaking manual process
    - Inspect HTML source for signals

- **NOT** once-and-for-all effort
  - Storefronts change over time

- Ripe opportunity for machine learning — a more automated approach to aid security practitioners
Feature extraction

HTML src

<html>
...
</html>

screenshot

DNS records
Feature extraction

- Affiliate programs use in-house software engines to generate storefront templates
- HTML contains distinctive signatures
- Bag-of-words on HTML – automated!
Data set

- classes: 44
- labeled exs: 178k
- largest class: 58k
- smallest class: 2

Data is high-dimensional & sparse
Visualization of EvaPharmacy
Proof-of-concept experiment

• **Question**: are these HTML features enough to distinguish affiliate programs’ storefronts?
• Favorable setting: plenty of labeled data
• Unlabeled Web pages → “other” class
  – First: discovered & labeled ~4k more storefronts!
• 45-way 1-nearest neighbor classification
• 10 random 70/30 train/test splits
Proof-of-concept experiment

- **Question:** are these HTML features enough to distinguish affiliate programs’ storefronts?
- Favorable setting: plenty of labeled data
- Unlabeled Web pages → “other” class
  - First: discovered & labeled ~4k more storefronts!
- 45-way 1-nearest neighbor classification
- 10 random 70/30 train/test splits

Avg accuracy = 99.95%
Proof-of-concept experiment

- **Question**: are these HTML features enough to distinguish affiliate programs’ storefronts?
- Favorable setting: plenty of labeled data
- Unlabeled Web pages → “other” class
  - First: discovered & labeled ~4k more storefronts!
- 45-way 1-nearest neighbor classification
- 10 random 70/30 train/test splits

Avg accuracy = 99.95%

How so good?!
HTML distances are highly predictive

Distances from every point to nearest neighbor in EvaPharmacy

![Graph showing distances to nearest neighbor in EvaPharmacy](image-url)

- **Eva**
- **Other affiliates**
- **Unlabeled**
Mimicking an operational deployment

- Experts must label **some** storefronts, but how many?
- Learning from scratch: only small initial seed of labeled storefronts
Mimicking an operational deployment

• Experts must label **some** storefronts, but how many?
• Learning from scratch: only small initial seed of labeled storefronts
Mimicking an operational deployment

• Experts must label **some** storefronts, but how many?
• Learning from scratch: only small initial seed of labeled storefronts
Classification in operational setting

<table>
<thead>
<tr>
<th>Avg accuracy</th>
<th>75%</th>
<th>85%</th>
<th>93%</th>
<th>97%</th>
<th>98%</th>
<th>99.95%</th>
</tr>
</thead>
<tbody>
<tr>
<td># of training examples per class</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>ALL (200 hours)</td>
</tr>
</tbody>
</table>
One-shot learning

Singly labeled storefront

Correctly classified storefronts

33drugs

RX-Promotions
Conclusion

• Automated system for identifying affiliate programs behind illegal online storefronts

• Simple model is highly accurate
  – Templatized storefronts, many near-duplicates
  – Affiliate programs’ efforts to operate at scale make automated defense possible

• Big win for security practitioners
  – Modest labeling effort is enough to bootstrap the system
Thank you!

Questions?