On Sybil Classification in Online Social Network Using Only OSN Structural Features

Dieudonne Mulamba, Indrajit Ray, Indrakshi Ray

Computer Science Department
Colorado State University
Online Social Network

- Preferred way to connect peoples
- Open platform: Anyone can join
- Some users can be fake or malicious – Sybils
Fake accounts (Sybils)

Sybils are for sale on the underground market
Fake accounts (Sybils)

Why are sybils so harmful?

Fake accounts can be used to:
- Send spam
- Do phishing
- Access personal user info
- ...

Fake accounts (Sybils)
Detecting Sybils is challenging

Detecting sybil accounts is difficult: These accounts may resemble real users

We want to be able to detect sybils automatically
Existing approaches

There are several approaches to detect sybils

• Content-based approaches
• Behavior-based approaches
• Graph-Structure based approaches
Existing Approaches

• Content-based approaches
  ▪ Collect user’s attributes (genre, age, mobility, power, …)
  ▪ Use machine-learning to classify users

• Behavior-based approaches
  ▪ Collect user’s activity data (like, posts, uploading image, …)
  ▪ Use machine-learning to classify users

• Graph-based approaches
  ▪ Leverage the relationship between nodes
Existing Approaches

• Content-based approaches

• Problems:
  • High false positive and negative rates
  • Some profiles are too easy to mimic
  • Information can be found online
Existing Approaches

• Content-based approaches

• The Fix: Hybrid approaches
  ▪ Add features from activity data (Behavior-based approach)
  ▪ Add features from the social graph (Graph-based approach)
  ▪ Use machine-learning to classify accounts.
Existing Approaches

- Hybrid approach: The workflow

- User profiles & activities
- Graph topology
- Features engineering
- Automated classification (Machine learning)
- Suspicious accounts
- Sybil detection
- Mitigation mechanisms
- Human verifier
Existing Approaches

**What is wrong?**

- Users do not always provide all the info requested in the profile
- Collecting user activities data raises the concern about user privacy

**New Direction:**

- Design features **ONLY** from network topology
- Use machine-learning to classify accounts.
Outline

1) Overview
2) Attack model
3) The Insights
4) Feature Engineering
   ➢ Existing features
   ➢ Proposed features
5) Feature selection
6) Dataset
7) Classification
8) Results
9) Conclusion
Our Work

• Avoid using features from user profiles, and user activity data
• Design features only from the topology of the social network
• Uses Machine-learning to detect Sybils
• Have evaluated results on many different types of synthetic datasets
  – Varies in size, and graph properties
• Have evaluated results on a real world OSN data (Twitter)
Our Work: Overview

• Convert the social network into an undirected graph
• Use graph theory to engineer features
• Select relevant features through features selection
• Build classification models
• Evaluate the results
Our Work: Overview

- Our approach: The workflow

- Graph topology
- Features engineering
- Automated classification (Machine learning)
- Suspicious accounts
- Sybil detection
- Human verifier
- Mitigation mechanisms
Our Work: Attack Model

• No assumption about attacker capabilities
• Attacker can create unlimited number of sybils
• Sybils may be connected to each other
• Attacker can befriend an unlimited number of benign nodes
• Attacker does not have control on the number of friend requests accepted
Our Work: The insight

- Features are engineered to capture the following patterns:
  - Sybils that form a dense friendship subgraph
  - Sybils that form a sparse friendship subgraph
  - Sybils tend to have friendship relationship with popular users
Our Work: The Features

• Features are designed using graph theory (centrality metrics)

• Existing features are:
  1. Average degree
  2. Average nearest neighbor degree
  3. Core number
  4. Average core number
  5. Clustering coefficient
  6. Average clustering coefficient
  7. Edge volume
  8. Weighted vertex volume
Our Work: The Features

• Features are designed using graph theory (centrality metrics)

• Proposed features are:
  1. Degree-intensity centrality
  2. Degree-coherence centrality
  3. Core-intensity centrality
  4. Core-coherence centrality
  5. Weighted degree-core centrality
  6. Weighted degree-clustering centrality
Our Work: Features Selection

• The feature selection model is: The Recursive Feature Elimination (RFE)

• Selected features are:
  1. Core number
  2. Average degree centrality
  3. Average clustering centrality
  4. Degree-coherence centrality
  5. Core-coherence centrality
  6. Edge volume centrality
  7. Weighted degree-core centrality
  8. Weighted degree-degree centrality
Our Work: Dataset

Facebook dataset

- Benign region: Facebook dataset
- Sybil region: network synthetically generated

<table>
<thead>
<tr>
<th>Region</th>
<th>Nodes</th>
<th>Edges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benign</td>
<td>4,039</td>
<td>88,234</td>
</tr>
<tr>
<td>Sybils</td>
<td>4,000</td>
<td>88,000</td>
</tr>
<tr>
<td>Attack edges</td>
<td>None</td>
<td>60,000</td>
</tr>
<tr>
<td>Total</td>
<td>8,039</td>
<td>236,234</td>
</tr>
</tbody>
</table>
Our Work: Dataset

Twitter dataset

• Real world dataset

<table>
<thead>
<tr>
<th>Region</th>
<th>Nodes</th>
<th>Edges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benign</td>
<td>372,251</td>
<td>906,102</td>
</tr>
<tr>
<td>Sybils</td>
<td>97,253</td>
<td>1,147,939</td>
</tr>
<tr>
<td>Attack edges</td>
<td>None</td>
<td>99,385</td>
</tr>
<tr>
<td>Total</td>
<td>469,504</td>
<td>2,153,426</td>
</tr>
</tbody>
</table>
Our Work: Classification

• Classifiers:
  - Adaboost (100 Estimators)
  - K-Nearest Neighbor (KNN)
  - Random Forest (100 trees)

• Evaluation metrics:
  - Precision
  - Recall
  - F-measure
  - Area Under the Curve (AUC)
Our Work: Results

- Classification on Facebook dataset

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaboost</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>KNN</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Random Forest</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>
Our Work: Results

- Classification on Twitter dataset

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaboost</td>
<td>0.95</td>
<td>0.94</td>
<td>0.94</td>
<td>0.94</td>
</tr>
<tr>
<td>KNN</td>
<td>0.99</td>
<td>0.99</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td>Random Forest</td>
<td>0.99</td>
<td>0.99</td>
<td>0.99</td>
<td>0.99</td>
</tr>
</tbody>
</table>
Our Work: Results

- Our method is very accurate
- We want to check for over-fitting
- We plot the learning curve to check for over-fitting
- There is not over-fitting
Conclusion

• We proposed a practical Sybil detection mechanism
• We classify users according to the topology of the graph
• We classify sybils with high accuracy (AUC=0.99)
• Topological features are hard to evade
• Future works: Use a dynamic graph
THANK YOU