The Latent Behavior Space
Sequential Behavior Information Encoded in a Vector Space

Jan Reubold, Stephan Escher, Thorsten Strufe
TU Dresden

15th August 2017
Social Bots in OSNs

- Social Bots = automation software created to control an OSN account and tries to pose as a human

**Intentions**

**Distributing Information**
- prestige influencing (hype/denounce products, companies, people, ...)
- political influencing (framing, hype denounce political topics, ...)
- traditional malicious content (spam, phishing, malware)
- ‘legit’ distribution (news, weather, ...)
- ...

**Harvesting Information**
- ‘Crawler’
- Identity Theft
- Personalized information distribution (spear phishing, ...)
- ...

Motivation /
Social Bots in OSNs
Detection Mechanisms

- Countermeasures: Prevention, Detection, Awareness, ...
  - **Graph Based** - detect Sybils with social / interaction graph analysis
  - **Community Based** - user based bot detection + experts
  - **Machine Learning Based**
    - Behavior Based
    - Crowd Based - Coordinated activity of multiple ‘users’
Behavior-Based Detection

Approaches I

- Sybil detection using clickstreams
- Two approaches (supervised, unsupervised)
  - SVM using 12 features:
    - ∅ clicks per session
    - ∅ session length
    - ∅ time between clicks
    - ∅ # sessions per day
    - Clickstreams → 8 Categories → BoW
  - Clustering:
    - Pure clickstreams + clickstreams enriched with timing (e.g. \([c_1, t_1, c_2, t_2, \ldots]\))
    - Clickstreams + Timings → Distance Function → Graph

Behavior-Based Detection

Approaches II

• Model for normal user behavior

• Behavior that does not fit ⇒ anomalous behavior (sybil + cyborg detection)

• Unsupervised Method:
  Principal Component Analysis (Feature space $F$)
  
  # likes per day
  
  # likes in specific categories (e.g. sports, politics, education)

  Evolution of spatial distribution of observed like categories

• PCA ⇒ latent subspace $S$
Time-Series Data
Pipelines

**Task:** Prediction, Classification, ...
e.g. Social-Bot Detection

**Input:** Time-Series
e.g. Click-Traces

- Labels
- Additional Information

**Time-Series Features:**
- Categorical-valued observations
e.g. login, send msg, share, ...
- Temporal order
- Timings
**Time-Series Data**

**Pipelines**

**Task:** Prediction, Classification, ...
  e.g. Social-Bot Detection

**Input:** Time-Series
  e.g. Click-Traces
  🗂️ Labels
  📄 Additional Information

**Time-Series Features:**
  - Categorical-valued observations
    e.g. login, send msg, share, ...
  - Temporal order
  - Timings

---

- Time-Series Data
  - Specialized Methods
    - Transformation
      - [1, 0, 1, 2, 1, 1, 1, 0]
      - [3, 3, 0, 0, 0, 0, 1, 1]
      - [1, 1, 0, 1, 0, 0, 1, 0]
  - Vector Space Methods
  - Solve Task
Standardize scheme

- Efficient use of vector space methods for time-series data (SVM, kNN, PCA, ...)

Idea

- Abstract & split time-series (in parallel)

Latent Behavior Space

- Span vector space by found patterns
- Express users by shown behavior patterns
Our Approach

Setup and Concept

Input:
- Set of time-series
e.g. user sessions on Facebook
- Time-series of categorical-valued observations
e.g. ‘send message’, ‘newsfeed’, ‘like’

Concept
- Abstraction of time-series data
- Represent by behavior patterns
called super states

\[ X = \{x_1, x_2, \ldots, x_M\} \]
\[ x_i = \begin{bmatrix} x_{i1} \ldots x_{il_i} \end{bmatrix} \]
\[ x_{ij} \in Y, \quad z_{ij} \in C \]

\[ C = \{\text{send message}, \text{newsfeed}, \text{like}\} \]

Time-Series Data

\begin{align*}
X &= \{x_1, x_2, \ldots, x_M\} & \text{Data} \\
x_i &= \begin{bmatrix} x_{i1} \ldots x_{il_i} \end{bmatrix} & \text{Time-Series} \\
x_{ij} &\in Y, \quad z_{ij} \in C & \text{Activity, Super State}
\end{align*}
Our Approach

Super State Graph

- Simple transition graph
- Transition probabilities governed by importance of super states
- Each node represents super state e.g. on click traces of an OSN: manifestation of an intention

Super State Graph

State Importance:
\[ \beta | \gamma \sim \text{Dir}(\gamma) \]

Transition Probability:
\[ \pi_c | \alpha, \beta \sim \text{Dir}(\alpha \beta + (1-\tau) \rho I) \]
Our Approach
Super States

- **Initial-State Distribution**
  \[ \theta_c^I | \lambda \sim \text{Dir} (\lambda) \]

- **Transition Distribution (+End-State)**
  \[ G_c | \psi_c \sim \text{Dir} (\psi_c) \quad \theta_{cs}^T | \lambda, G_c \sim \text{Dir}(\lambda G_c) \]

- **State-Duration Model**
  \[ \mu_{cs} \sim t_v \left( \mu_{cs}, \tilde{\mu}_{cs}, \tilde{\sigma}_{cs}^2 / \tilde{\kappa}_{cs} \right) \]
  \[ \sigma_{cs}^2 \sim \chi^{-2} \left( \sigma_{cs}^2 | \tilde{\nu}, \tilde{\sigma}_{cs}^2 \right) \]
Our Approach
The Latent Behavior Space

- Segmentation of user behavior into known patterns
- Represent user by her behavior / exhibited patterns
- Use vector space methods

Latent Behavior Space

Transformation

$$\nu_u = \phi(\mathbf{X}_u)$$

- Count-based

$$\phi^O_u(\mathbf{X}_u) \triangleq \sum_{(i, j) \in \tilde{Z}_u} \frac{1_{z_{ij}}}{|\tilde{Z}_u|}$$

- Time-based

$$\phi^D_u(\mathbf{X}_u) \triangleq \frac{\sum_{1 \leq i \leq |\mathbf{X}_u|} \sum_{1 \leq j \leq t_{ij}} 1_{z_{ij}} t_{ij}}{\sum_{1 \leq i \leq |\mathbf{X}_u|} \sum_{1 \leq j \leq t_{ij}} t_{ij}}$$
Our Approach

The Latent Behavior Space

- Segmentation of user behavior into known patterns
- Represent user by her behavior / exhibited patterns
- Use vector space methods

Latent Behavior Space

Transformation

\[ \nu_u = \phi(X_u) \]

- Count-based

\[ \phi^O(X_u) \triangleq \sum_{(i,j) \in \tilde{Z}_u} \frac{1_{z_{ij}}}{|\tilde{Z}_u|} \]

- Time-based

\[ \phi^D(X_u) \triangleq \frac{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq t_{ij}} 1_{z_{ij}} t_{ij}}{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq t_{ij}} t_{ij}} \]
Our Approach
The Latent Behavior Space

- Segmentation of user behavior into known patterns
- Represent user by her behavior exhibited patterns
- Use vector space methods

Latent Behavior Space

Transformation
\[ \nu_u = \phi \left( X_u \right) \]

- Count-based
\[ \phi^O_u \left( X_u \right) \triangleq \frac{1}{|\tilde{z}_u|} \sum_{(i,j) \in \tilde{z}_u} I_{ij} \]

- Time-based
\[ \phi^D_u \left( X_u \right) \triangleq \frac{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq t_{ij}} I_{ij} t_{ij}}{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq t_{ij}} t_{ij}} \]
Our Approach
The Latent Behavior Space

- Segmentation of user behavior into known patterns
- Represent user by her behavior / exhibited patterns
- Use vector space methods

Latent Behavior Space

Transformation
\[
\nu_u = \phi (X_u)
\]

- Count-based
\[
\phi^O_z (X_u) = \frac{\sum_{(i,j) \in \tilde{z}_u} 1^{z_{ij}}}{|\tilde{z}_u|}
\]

- Time-based
\[
\phi^D_z (X_u) = \frac{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq l_i} 1^{z_{ij}} t_{ij}}{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq l_i} t_{ij}}
\]
Our Approach

The Latent Behavior Space

- Segmentation of user behavior into known patterns
- Represent user by her behavior / exhibited patterns
- Use vector space methods

Latent Behavior Space

Transformation

\[ \nu_u = \phi(X_u) \]

- Count-based

\[ \phi^O_k(X_u) \triangleq \frac{\sum_{(i,j) \in \tilde{Z}_u} z_{ij}}{|\tilde{Z}_u|} \]

- Time-based

\[ \phi^D_k(X_u) \triangleq \frac{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq t_i} 1_{z_{ij}} t_{ij}}{\sum_{1 \leq i \leq |X_u|} \sum_{1 \leq j \leq t_i} t_{ij}} \]
Evidence on controlled settings

- Impact of sequential information for process recovery
  - 3 scenarios (sets of super states)
  - Scenario I → III: Increasing state space overlap
  - Recovery of processes (FNR)
Evidence on controlled settings

- Impact of sequential information for process recovery
  - 3 scenarios (sets of super states)
  - Scenario I → III: Increasing state space overlap
  - Recovery of processes (FNR)
Evaluations /

Social Bot Detection
Work in Progress

• Develop attacker models based on characteristics
  • Profile Characteristic (Sybils, Cyborgs, Zombies, ...)
  • Social Bot-Networks (Union of social bots)
  • Massattacks vs Targeted Attacks
  • OSN Structure
  • Complexity (send every hour, user based behavior)

• Enrich real-world data set with behavior traces of theoretical attackers
**Our Approach**

**Summary**

**Feature Design**

by leveraging behavior patterns

**Goal:** Vector space + time-series

**Problem:** Loss of information

**Approaches:**

*Direct:*
Integrate sequential information into time-series

*Abstraction:*
Learn patterns from time-series
→ represent time-series by patterns
Our Approach

Summary

Feature Design by leveraging behavior patterns

**Goal:** Vector space + time-series

**Problem:** Loss of information

**Approaches:**

*Direct:* Integrate sequential information into time-series

*Abstraction:* Learn patterns from time-series → represent time-series by patterns
Future Work

Current Work
• Evaluate on real-world data (replace individual transformations with LBS)
• Evaluate impact of loss of information (segmentation → LBS)
• Build theoretical attacker models

General
• Further work on social bot detection (behavior graph)
• Investigate influence streams (e.g. by means of frames) and echo chambers
Future Work

Current Work
- Evaluate on real-world data (replace individual transformations with LBS)
- Evaluate impact of loss of information (segmentation $\rightarrow$ LBS)
- Build theoretical attacker models

General
- Further work on social bot detection (behavior graph)
- Investigate influence streams (e.g. by means of frames) and echo chambers

Thank you! Questions?