Collective Spammer Detection in Evolving Multi-Relational Social Networks

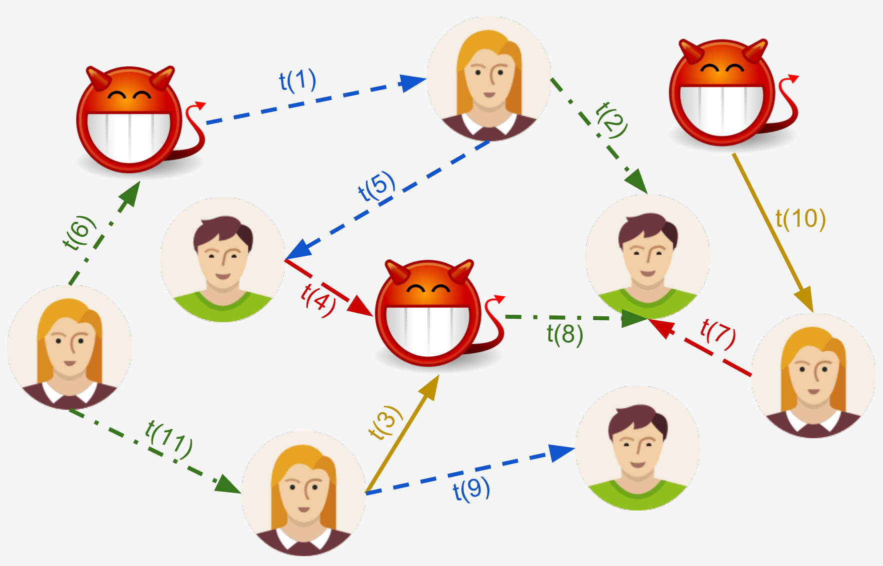
Shobeir Fakhraei[1,2], James Foulds[2], Madhusudana Shashanka[3], and Lise Getoor[2]

[1]University of Maryland, College Park, MD, USA
[2]University of California, Santa Cruz, CA, USA
[3]if(we) Inc. (Currently Niara Inc., CA, USA)

Motivation

- Spam is pervasive in social networks.
- Traditional approaches don't work well:
  - Spammers can manipulate content-based approaches. E.g., change patterns, split malicious content across messages.
  - Content may not be available due to privacy reasons.
- Spammers have more ways to interact with users in social networks compared to email and the web.

Problem Statement



- We have a time-stamped multi-relational social network with legitimate users and spammers.
- links = actions at time **t** (e.g. profile view, message, or poke).
- **Task:**
  Snapshot of the social network +
  Labels of already identified spammers
  Find other spammers in the network.

Contribution and Proposed Solution

- Use only the multi-relational meta-data for spammer detection:
  - Graph Structure.
  - Action Sequences.
- Collectively refine user generated abuse reports.

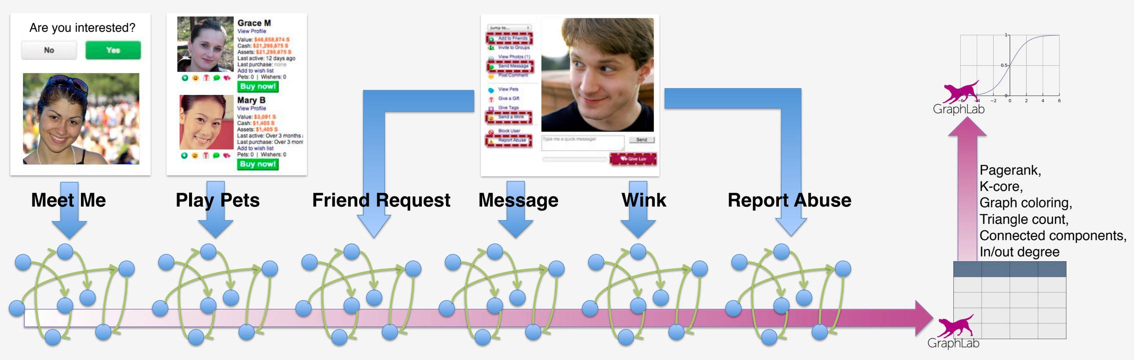
Data

- A data sample from Tagged.com, including all active users and their activities in a specific timeframe.
- Tagged is a social network for meeting new people with multiple methods for users to interact.
- It was founded in 2004 and has over 300 million registered members.

Entity	Count
$\lvert\mathcal{V}\rvert$ (total users)	5,607,454
$\lvert\mathcal{E}\rvert$ (total actions)	912,280,409
$\max(\lvert\mathcal{E}_r\rvert)$ (number of actions that are most frequent action type)	350,724,903
$\min(\lvert\mathcal{E}_r\rvert)$ (number of actions that are least frequent action type)	137,550
total users labeled as spammers	(%3.9) 221,305

Data Sample Statistics.

Graph Structure Features

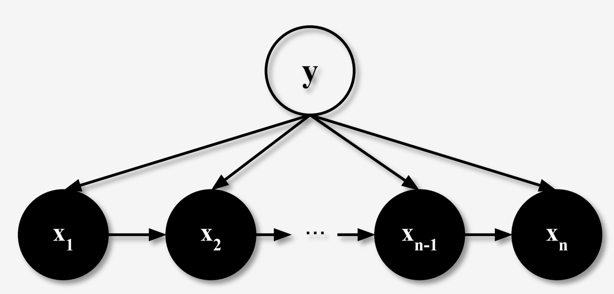


In each relation graph we compute:

- **PageRank:** Score for each node based on number and quality of links to it.
- **Degree:** Total degree, in-degree, and out-degree of each node.
- **k-Core:** Centrality measure via recursive pruning of the least connected vertices.
- **Graph Coloring:** Assignment of colors to vertices, where no two adjacent vertices share the same color.
- **Connected Components:** Group of vertices with a path between each.
- **Triangle Count:** Number of triangles the vertex participates in.

Sequence-Based Features

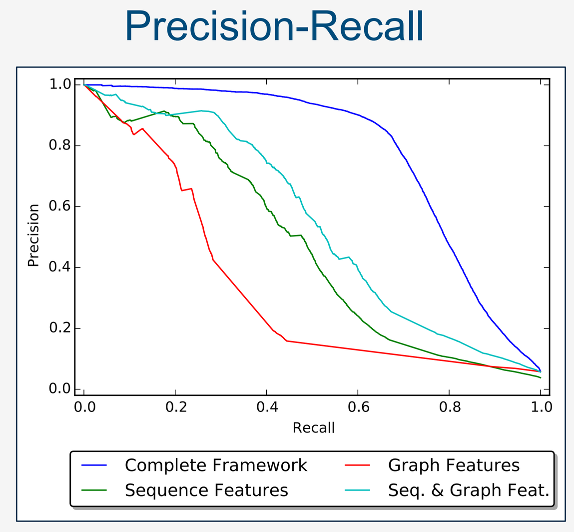
- **Sequential k-gram Features:** Short sequence segment of k consecutive actions, to capture the order of events.
- **Mixture of Markov Models:** Also called chain-augmented or tree-augmented naive Bayes model to capture longer sequences.



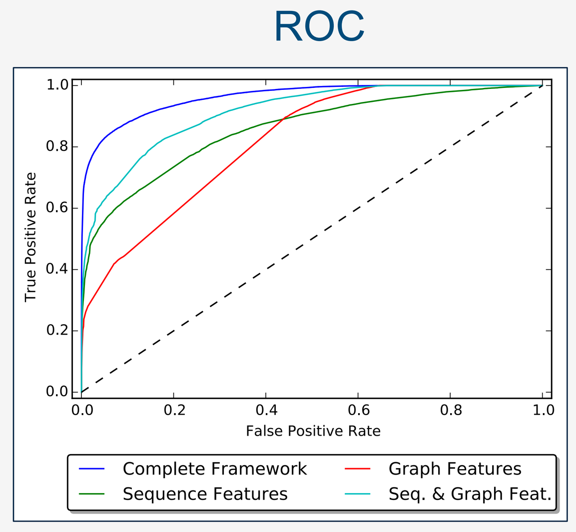
$$P(y,x) = P(y)P(x_1|y)\prod_{i=2}^{n} P(x_i|x_{i-1},y) \ ,$$

Graph Structure and Sequence-Based Results

Precision-Recall



ROC



- Complete framework includes graph structure and sequence features, and three demographic features (i.e., age, gender, and time since registration).
- We used Graphlab Create for feature extraction and classification with Gradient-Boosted Decision Trees.

HL-MRFs and Probabilistic Soft Logic

- Hinge-loss Markov random fields (HL-MRFs) are a general class of conditional, continuous probabilistic models.
- Probabilistic soft logic (PSL) uses a first-order logical syntax as a templating language for HL-MRFs.
- General rules:

$$\omega : P(A,B) \land Q(B,C) \rightarrow R(A,C)$$

- Predicates have soft truth values between [0,1]
- Rule satisfaction: $r_{body} \rightarrow r_{head}$

$$I(r_{body}) \leq I(r_{head})$$

- Distance from satisfaction:

$$\delta_r = \max\{0, I(r_{body}) - I(r_{head})\}$$

- Most probable explanation (MPE) by optimizing:

$$f(\mathcal{I}) = \frac{1}{\mathcal{Z}} \exp\left[-\sum_{r\in\mathcal{R}} \omega_r \delta_r(\mathcal{I})\right]$$

Collective Classification with Reports

Users can report abusive behavior, but the reports contain a lot of noise.

- Model using only reports:

$$\begin{aligned} \mathit{REPORTED}(v_1,v_2) &\rightarrow \mathit{SPAMMER}(v_2) \\ &\neg\mathit{SPAMMER}(v) \end{aligned}$$

- Model using reports and credibility of the reporter:

$$\begin{aligned} \mathit{CREDIBLE}(v_1) \land \mathit{REPORTED}(v_1,v_2) &\rightarrow \mathit{SPAMMER}(v_2) \\ \mathit{PRIOR\text{-}CREDIBLE}(v) &\rightarrow \mathit{CREDIBLE}(v) \\ \neg\mathit{PRIOR\text{-}CREDIBLE}(v) &\rightarrow \neg\mathit{CREDIBLE}(v) \\ &\neg\mathit{SPAMMER}(v) \end{aligned}$$

- Model using reports, credibility of the reporter, and collective reasoning:

$$\begin{aligned} \mathit{CREDIBLE}(v_1) \land \mathit{REPORTED}(v_1,v_2) &\rightarrow \mathit{SPAMMER}(v_2) \\ \mathit{SPAMMER}(v_2) \land \mathit{REPORTED}(v_1,v_2) &\rightarrow \mathit{CREDIBLE}(v_1) \\ \neg\mathit{SPAMMER}(v_2) \land \mathit{REPORTED}(v_1,v_2) &\rightarrow \neg\mathit{CREDIBLE}(v_1) \\ \mathit{PRIOR\text{-}CREDIBLE}(v) &\rightarrow \mathit{CREDIBLE}(v) \\ \neg\mathit{PRIOR\text{-}CREDIBLE}(v) &\rightarrow \neg\mathit{CREDIBLE}(v) \\ &\neg\mathit{SPAMMER}(v) \end{aligned}$$

Results of Classification Using Reports

Experiment	AUPR	AUROC
Reports Only	0.674±0.008	0.611±0.007
Reports & Credibility	0.869±0.006	0.862±0.004
Reports & Credibility & Collective Reasoning	0.884±0.005	0.873±0.004