Social Networks of Spammers

Alfred O. Hero, III

Department of Electrical Engineering and Computer Science
University of Michigan

September 22, 2008

Outline

Outline

Acknowledgements

- My co-authors on “Revealing social networks of spammers through spectral clustering,” ICC09 (submitted)
  - Kevin Xu, Yilun Chen, Peter Woolf - University of Michigan
  - Mark Kliger - Medasense Biometrics, Inc
- Other collaborators
  - John Bell, Nitin Nayar - University of Michigan
  - Matthew Prince, Eric Langheinrich, Lee Holloway - Unspam Technologies
  - Matt Roughan, Olaf Maennel - University of Adelaide
- Sponsors
  - National Science Foundation CCR-0325571
  - Office of Naval Research N00014-08-1065
  - NSERC graduate fellowship program

Acknowledgements

- My co-authors on “Revealing social networks of spammers through spectral clustering,” ICC09 (submitted)
  - Kevin Xu, Yilun Chen, Peter Woolf - University of Michigan
  - Mark Kliger - Medasense Biometrics, Inc
- Other collaborators
  - John Bell, Nitin Nayar - University of Michigan
  - Matthew Prince, Eric Langheinrich, Lee Holloway - Unspam Technologies
  - Matt Roughan, Olaf Maennel - University of Adelaide
- Sponsors
  - National Science Foundation CCR-0325571
  - Office of Naval Research N00014-08-1065
  - NSERC graduate fellowship program

Acknowledgements

- My co-authors on “Revealing social networks of spammers through spectral clustering,” ICC09 (submitted)
  - Kevin Xu, Yilun Chen, Peter Woolf - University of Michigan
  - Mark Kliger - Medasense Biometrics, Inc
- Other collaborators
  - John Bell, Nitin Nayar - University of Michigan
  - Matthew Prince, Eric Langheinrich, Lee Holloway - Unspam Technologies
  - Matt Roughan, Olaf Maennel - University of Adelaide
- Sponsors
  - National Science Foundation CCR-0325571
  - Office of Naval Research N00014-08-1065
  - NSERC graduate fellowship program

Objectives

- Objectives of this study
  - To reveal social networks of spammers
    - Identifying communities of spammers
    - Finding characteristics or “signatures” of communities
  - To understand temporal dynamics of spammers’ behavior
    - Detecting changes in social structure
- Motivation
  - Current anti-spam methods
    - Content filtering
    - IP address blacklisting
  - Allows us to fight spam from another perspective by using spammers’ social structure

Objectives

- Objectives of this study
  - To reveal social networks of spammers
    - Identifying communities of spammers
    - Finding characteristics or “signatures” of communities
  - To understand temporal dynamics of spammers’ behavior
    - Detecting changes in social structure
- Motivation
  - Current anti-spam methods
    - Content filtering
    - IP address blacklisting
  - Allows us to fight spam from another perspective by using spammers’ social structure

Background

Much of past research on spam has focussed on scalar analysis

- Spam/phishing content structural analysis [Chandrasekaran, Narayanan, Uphadhyaya CSC06]
- Server lifetime and reachability analysis [Duan,Gopalan, Yuan, ICC07]
- Spam botnet behavior patterns [Ramachandran, Feamster, SIGCOMM06]
- Honeypot summary statistics [Prince, Holloway, Langheinrich, Dahl, Keller EAS05]

- We perform analysis of spammer interactions over entire spam cycle

Background

Much of past research on spam has focussed on scalar analysis

- Spam/phishing content structural analysis [Chandrasekaran, Narayanan, Uphadhyaya CSC06]
- Server lifetime and reachability analysis [Duan,Gopalan, Yuan, ICC07]
- Spam botnet behavior patterns [Ramachandran, Feamster, SIGCOMM06]
- Honeypot summary statistics [Prince, Holloway, Langheinrich, Dahl, Keller EAS05]

- We perform analysis of spammer interactions over entire spam cycle

The Spam Cycle

- Two phases of the spam cycle
  - Harvesting: collecting email addresses from web sites using spam bots
  - Spamming: sending large amounts of emails to collected addresses using spam servers
- Spammers conceal their identity (IP address) in spamming phase by using public SMTP servers, open proxies, botnets, etc.
- Key assumption: spammer IP address in harvesting phase is closely related to actual location
  - Previous study found harvester IP address more closely related to actual spammer than spam server IP address (Prince et. al, 2005)
  - We treat the harvester as the spam source

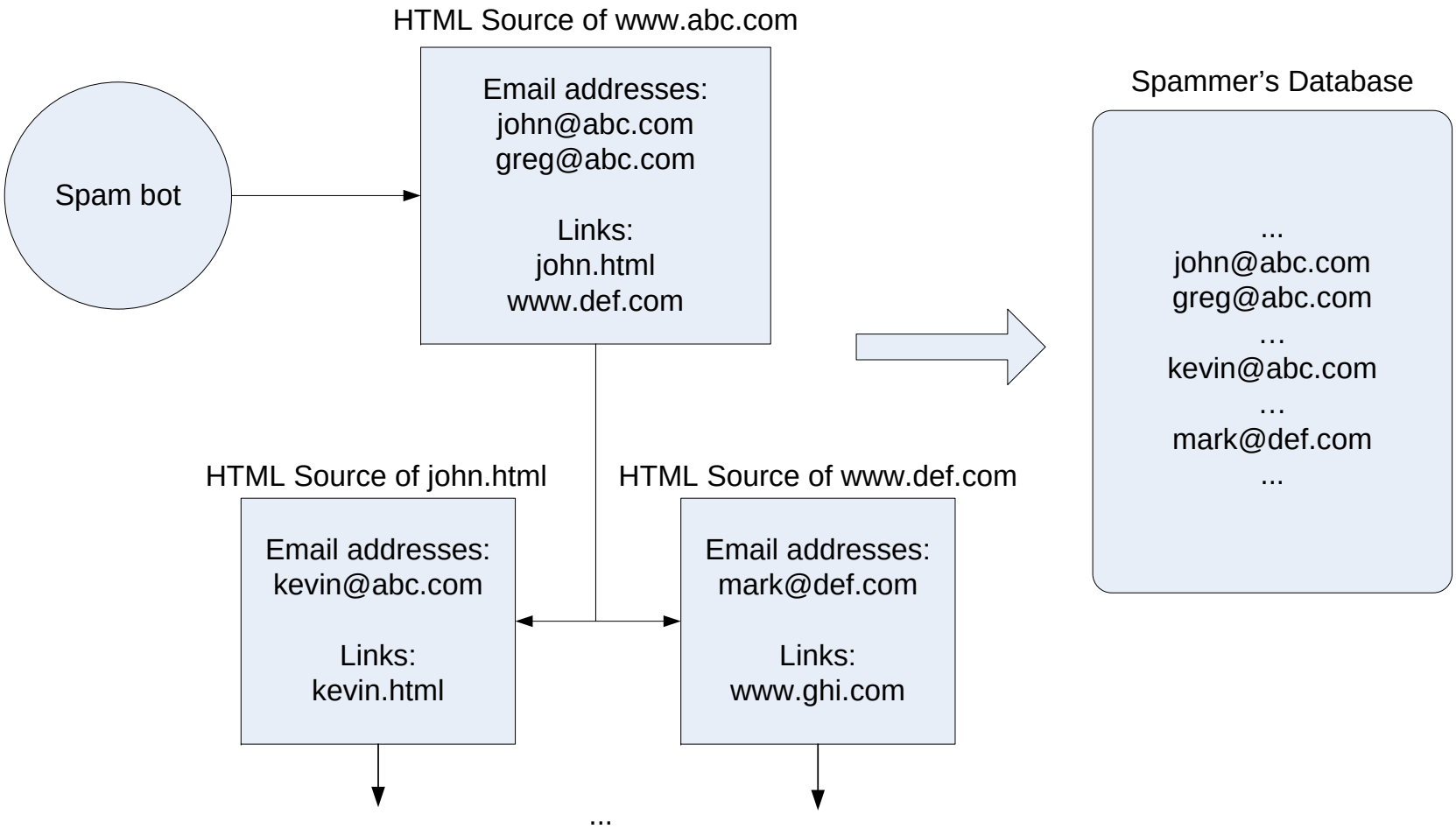
The Spam Cycle

- Two phases of the spam cycle
  - Harvesting: collecting email addresses from web sites using spam bots
  - Spamming: sending large amounts of emails to collected addresses using spam servers
- Spammers conceal their identity (IP address) in spamming phase by using public SMTP servers, open proxies, botnets, etc.
- Key assumption: spammer IP address in harvesting phase is closely related to actual location
  - Previous study found harvester IP address more closely related to actual spammer than spam server IP address (Prince et. al, 2005)
  - We treat the harvester as the spam source

The Spam Cycle

- Two phases of the spam cycle
  - Harvesting: collecting email addresses from web sites using spam bots
  - Spamming: sending large amounts of emails to collected addresses using spam servers
- Spammers conceal their identity (IP address) in spamming phase by using public SMTP servers, open proxies, botnets, etc.
- Key assumption: spammer IP address in harvesting phase is closely related to actual location
  - Previous study found harvester IP address more closely related to actual spammer than spam server IP address (Prince et. al, 2005)
  - We treat the harvester as the spam source

Harvestor Email Address Collection



How harvesters acquire email addresses using spam bots

The Path of Spam

The path of spam: from an email address on a web page to your inbox

Project Honey Pot

- Network of decoy web pages (“honey pots”) with trap email addresses
- All email received is spam
- Unique email address generated at each visit
- Visitor (harvester) IP address is tracked
- When spam is received, we know the harvester IP address in addition to the spam server IP address

Project Honey Pot

- Network of decoy web pages (“honey pots”) with trap email addresses
- All email received is spam
- Unique email address generated at each visit
- Visitor (harvester) IP address is tracked
- When spam is received, we know the harvester IP address in addition to the spam server IP address

Project Honey Pot

- Network of decoy web pages (“honey pots”) with trap email addresses
- All email received is spam
- Unique email address generated at each visit
- Visitor (harvester) IP address is tracked
- When spam is received, we know the harvester IP address in addition to the spam server IP address

Project Honey Pot

- Network of decoy web pages (“honey pots”) with trap email addresses
- All email received is spam
- Unique email address generated at each visit
- Visitor (harvester) IP address is tracked
- When spam is received, we know the harvester IP address in addition to the spam server IP address

Project Honey Pot

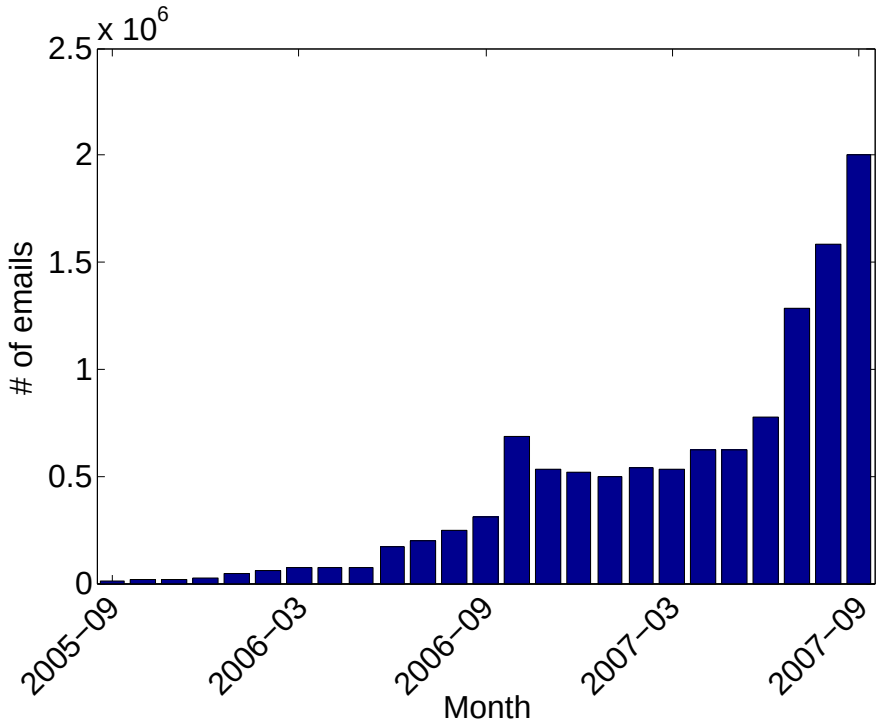
- Network of decoy web pages (“honey pots”) with trap email addresses
- All email received is spam
- Unique email address generated at each visit
- Visitor (harvester) IP address is tracked
- When spam is received, we know the harvester IP address in addition to the spam server IP address

Project Honey Pot Statistics (as of Sept. 16, 2008)

- Spam Trap Addresses Monitored: 29,765,172
- Spam Trap Monitoring Capability: 272,870,000,000
- Spam Servers Identified: 29,712,922
- Harvesters Identified: 52,069

www.projecthoneypot.org

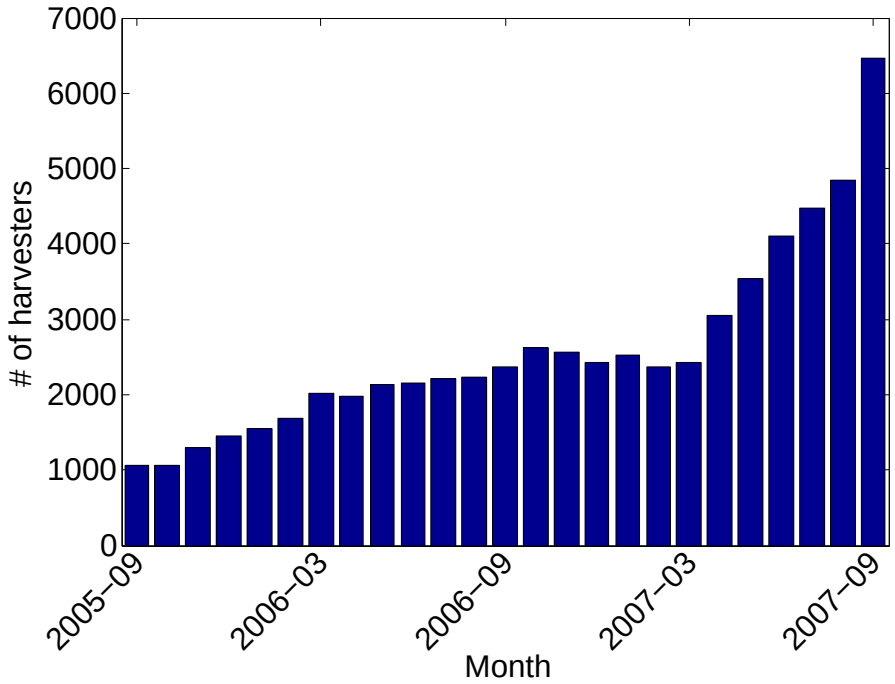
Total Emails By Month



Total emails received at Project Honey Pot trap addresses by month

- Outbreak of spam observed in October 2006 consistent with media reports

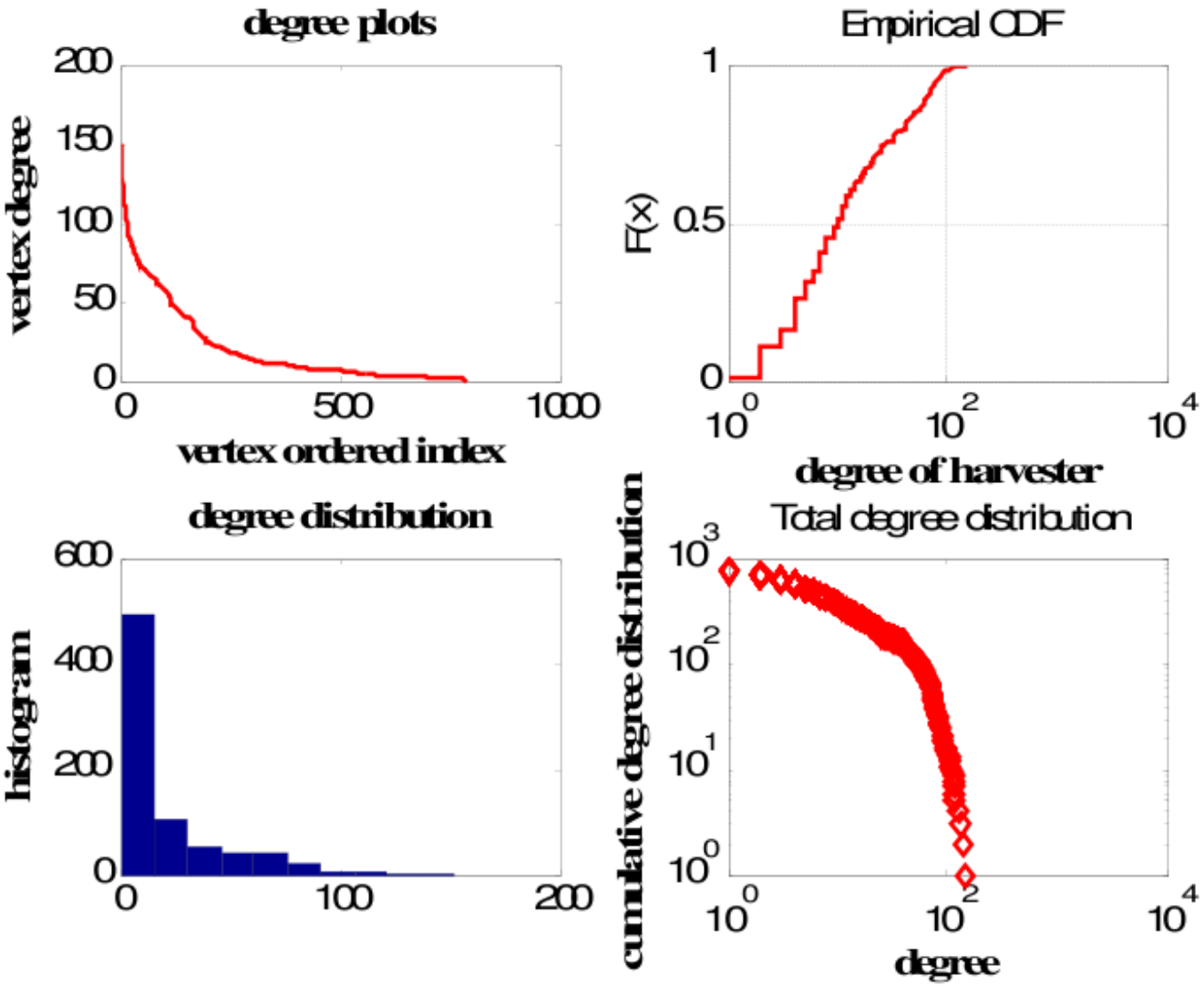
Total Active Harvesters By Month



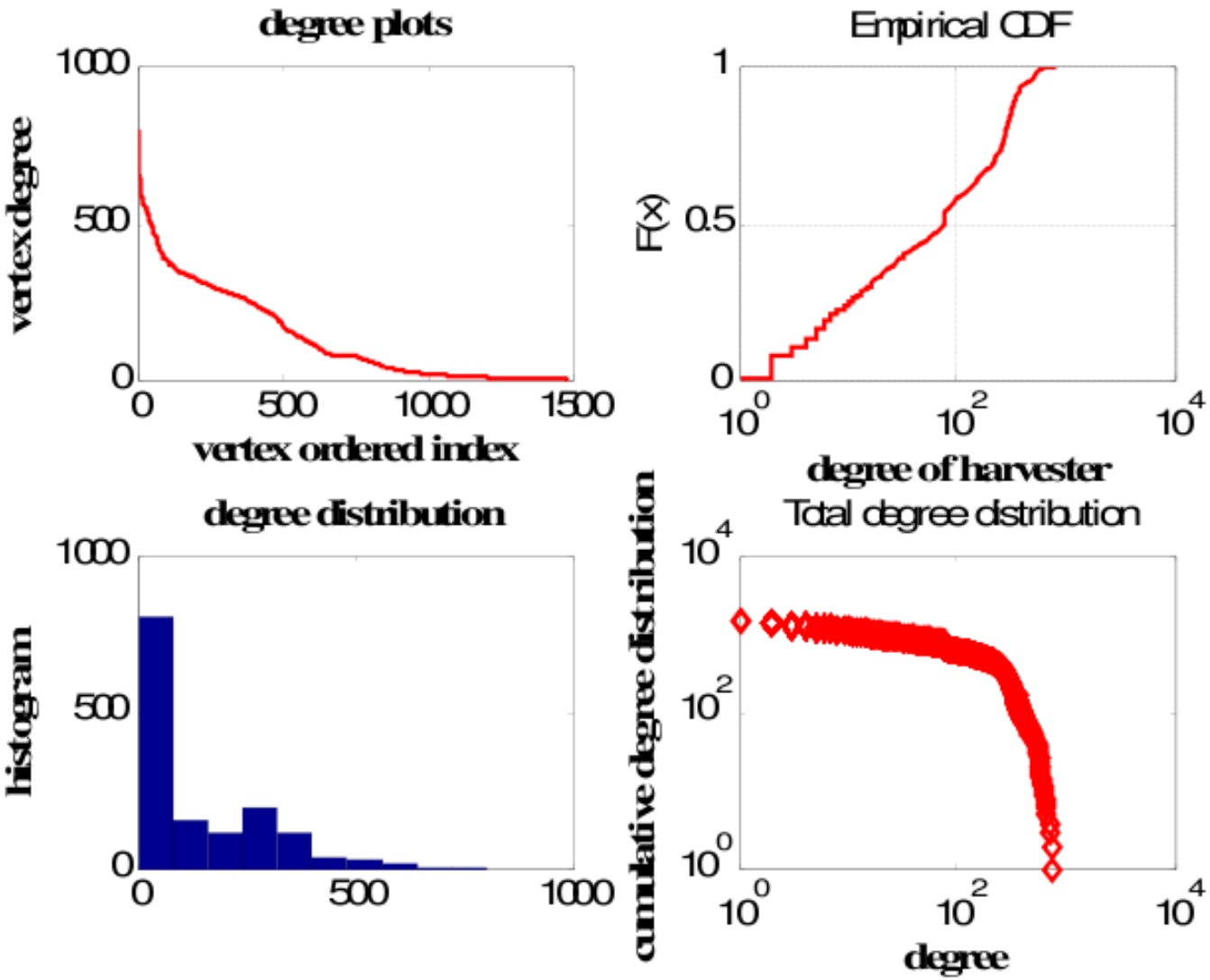
Total active harvesters tracked by Project Honey Pot by month

- Increase in harvesters in October 2006 not as significant as increase in number of spam emails

Harvestor-to-server degree distribution: May 2006



Harvestor-to-server degree distribution: Oct 2006



Phishing

- Phishing is an attempt to fraudulently acquire sensitive information by appearing to represent a trustworthy entity
- Project Honey Pot is an excellent data source for studying phishing emails
  - Trap email address cannot, for example, sign up for a PayPal account
  - All emails supposedly received from financial institutions can be classified as phishing
- We classify an email as a phishing email if its subject contains common phishing words

Phishing

- Phishing is an attempt to fraudulently acquire sensitive information by appearing to represent a trustworthy entity
- Project Honey Pot is an excellent data source for studying phishing emails
  - Trap email address cannot, for example, sign up for a PayPal account
  - All emails supposedly received from financial institutions can be classified as phishing
- We classify an email as a phishing email if its subject contains common phishing words

Phishing

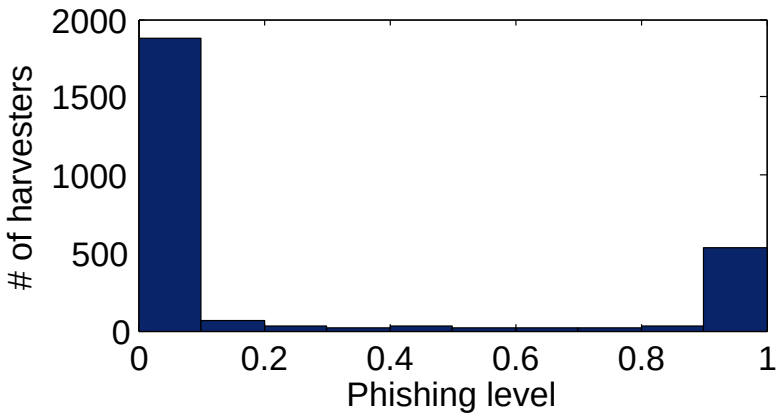
- Phishing is an attempt to fraudulently acquire sensitive information by appearing to represent a trustworthy entity
- Project Honey Pot is an excellent data source for studying phishing emails
  - Trap email address cannot, for example, sign up for a PayPal account
  - All emails supposedly received from financial institutions can be classified as phishing
- We classify an email as a phishing email if its subject contains common phishing words

Phishing Statistics

- Define a *phishing level* for each harvester as

$$\text{Phishing level} = \frac{\text{\# of phishing emails sent}}{\text{total \# of emails sent}}$$

- Label harvesters with phishing level > 0.5 as phishers
- October 2006 statistics
  - 4.5% of emails were phishing emails
  - 23% of harvesters were phishers

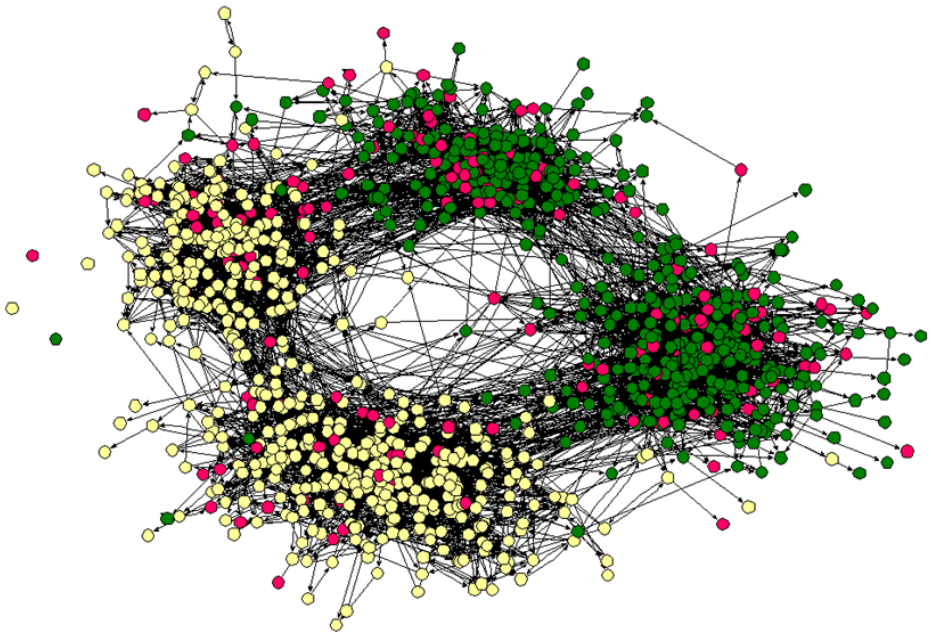


Histogram of harvesters' phishing levels from October 2006

Social Networks

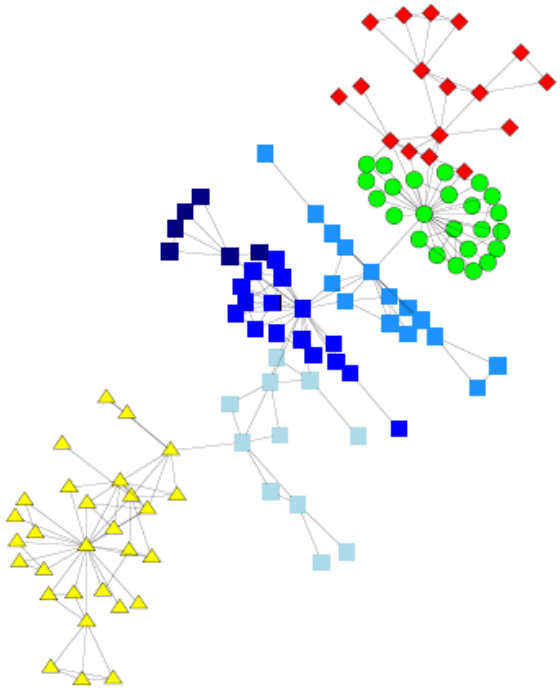
- Social network: social structure consisting of actors and ties
  - Actors represent individuals
  - Ties represent relationships between individuals

School friendships



Moody, 2001

Scientific collaborations



Girvan and Newman, 2002

Outline

Graph of Harvester Interactions

- Represent network of harvesters by undirected weighted graph $G = (V, E, W)$
  - V: set of vertices (harvesters)
  - E: set of edges between harvesters
  - W: matrix of edge weights (adjacency matrix of graph)
- Edge weights represent strength of connection between two harvesters
- Total weights of edges between two sets of harvesters $A, B \subset V$ is defined by

$$\text{links}(A, B) = \sum_{i \in A} \sum_{j \in B} w_{ij}$$

- Degree of a set A is defined by

$$\deg(A) = \text{links}(A, V)$$

Graph of Harvester Interactions

- Represent network of harvesters by undirected weighted graph $G = (V, E, W)$
  - V: set of vertices (harvesters)
  - E: set of edges between harvesters
  - W: matrix of edge weights (adjacency matrix of graph)
- Edge weights represent strength of connection between two harvesters
- Total weights of edges between two sets of harvesters $A, B \subset V$ is defined by

$$\text{links}(A, B) = \sum_{i \in A} \sum_{j \in B} w_{ij}$$

- Degree of a set A is defined by

$$\deg(A) = \text{links}(A, V)$$

Graph of Harvester Interactions

- Represent network of harvesters by undirected weighted graph $G = (V, E, W)$
  - V: set of vertices (harvesters)
  - E: set of edges between harvesters
  - W: matrix of edge weights (adjacency matrix of graph)
- Edge weights represent strength of connection between two harvesters
- Total weights of edges between two sets of harvesters $A, B \subset V$ is defined by

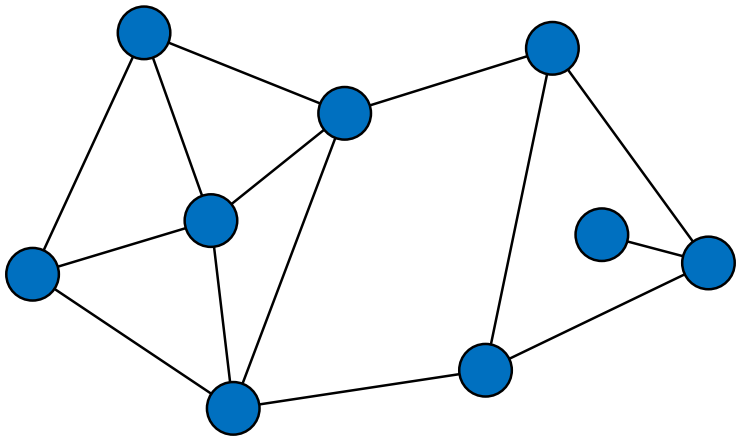
$$\text{links}(A, B) = \sum_{i \in A} \sum_{j \in B} w_{ij}$$

- Degree of a set A is defined by

$$\deg(A) = \text{links}(A, V)$$

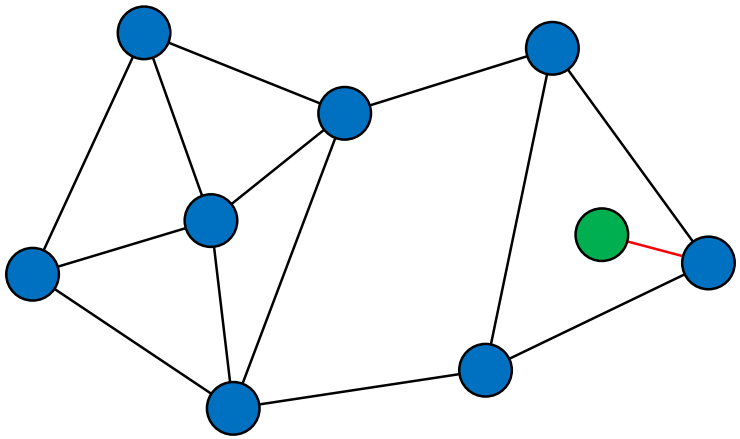
Community Detection

- Characteristics of a community
  - High similarity between actors within community
  - Low similarity between actors in different communities
- Formulate community detection as a graph partitioning problem
  - Divide the graph into clusters
  - Maximize edge weights within clusters (association)
  - Minimize edge weights between clusters (cut)
- Using edge weights normalized by group sizes results in better groups



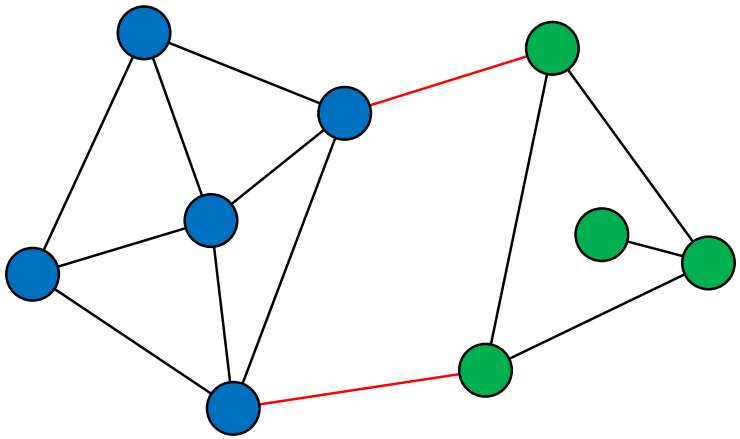
Community Detection

- Characteristics of a community
  - High similarity between actors within community
  - Low similarity between actors in different communities
- Formulate community detection as a graph partitioning problem
  - Divide the graph into clusters
  - Maximize edge weights within clusters (association)
  - Minimize edge weights between clusters (cut)
- Using edge weights normalized by group sizes results in better groups



Community Detection

- Characteristics of a community
  - High similarity between actors within community
  - Low similarity between actors in different communities
- Formulate community detection as a graph partitioning problem
  - Divide the graph into clusters
  - Maximize edge weights within clusters (association)
  - Minimize edge weights between clusters (cut)
- Using edge weights normalized by group sizes results in better groups



Normalized Cut and Association

- Normalized cut of a graph partition Γ_V^K is defined as

$$\text{KNcut}(\Gamma_V^K) = \frac{1}{K}\sum_{i=1}^{K}\frac{\text{links}(V_i, V\backslash V_i)}{\deg(V_i)}$$

- Normalized association of Γ_V^K is defined as

$$\text{KNassoc}(\Gamma_V^K) = \frac{1}{K}\sum_{i=1}^{K}\frac{\text{links}(V_i, V_i)}{\deg(V_i)}$$

- $\text{KNcut}(\Gamma_V^K) + \text{KNassoc}(\Gamma_V^K) = 1$ so minimizing normalized cut simultaneously maximizes normalized association
- We try to maximize normalized association

Normalized Cut and Association

- Normalized cut of a graph partition Γ_V^K is defined as

$$\text{KNcut}(\Gamma_V^K) = \frac{1}{K}\sum_{i=1}^{K}\frac{\text{links}(V_i, V\backslash V_i)}{\deg(V_i)}$$

- Normalized association of Γ_V^K is defined as

$$\text{KNassoc}(\Gamma_V^K) = \frac{1}{K}\sum_{i=1}^{K}\frac{\text{links}(V_i, V_i)}{\deg(V_i)}$$

- $\text{KNcut}(\Gamma_V^K) + \text{KNassoc}(\Gamma_V^K) = 1$ so minimizing normalized cut simultaneously maximizes normalized association
- We try to maximize normalized association

Normalized Cut and Association

- Normalized cut of a graph partition Γ_V^K is defined as

$$\text{KNcut}(\Gamma_V^K) = \frac{1}{K}\sum_{i=1}^{K}\frac{\text{links}(V_i, V\backslash V_i)}{\deg(V_i)}$$

- Normalized association of Γ_V^K is defined as

$$\text{KNassoc}(\Gamma_V^K) = \frac{1}{K}\sum_{i=1}^{K}\frac{\text{links}(V_i, V_i)}{\deg(V_i)}$$

- $\text{KNcut}(\Gamma_V^K) + \text{KNassoc}(\Gamma_V^K) = 1$ so minimizing normalized cut simultaneously maximizes normalized association
- We try to maximize normalized association

The Discrete Optimization Problem

- Represent graph partition Γ_V^K by matrix $X = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_K}]$
- $\mathbf{x_i}$: column indicator vector with ones in the rows corresponding to harvesters in cluster i
- Degree matrix $D = \text{diag}(W 1_M)$
- Rewrite links and deg as

$$\text{links}(V_i, V_i) = \mathbf{x_i}^T W \mathbf{x_i}$$
$$\text{deg}(V_i) = \mathbf{x_i}^T D \mathbf{x_i}$$

- KNassoc maximization problem becomes

$$\text{maximize} \quad \text{KNassoc}(X) = \frac{1}{K} \sum_{i=1}^{K} \frac{\mathbf{x_i}^T W \mathbf{x_i}}{\mathbf{x_i}^T D \mathbf{x_i}}$$

$$\text{subject to} \quad X \in \{0,1\}^{M \times K}$$
$$X 1_K = 1_M$$

The Discrete Optimization Problem

- Represent graph partition Γ_V^K by matrix $X = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_K}]$
- $\mathbf{x_i}$: column indicator vector with ones in the rows corresponding to harvesters in cluster i
- Degree matrix $D = \text{diag}(W1_M)$
- Rewrite links and deg as

$$\text{links}(V_i, V_i) = \mathbf{x_i}^T W \mathbf{x_i}$$
$$\text{deg}(V_i) = \mathbf{x_i}^T D \mathbf{x_i}$$

- KNassoc maximization problem becomes

$$\text{maximize} \quad \text{KNassoc}(X) = \frac{1}{K} \sum_{i=1}^{K} \frac{\mathbf{x_i}^T W \mathbf{x_i}}{\mathbf{x_i}^T D \mathbf{x_i}}$$

$$\text{subject to} \quad X \in \{0,1\}^{M \times K}$$
$$X1_K = 1_M$$

The Discrete Optimization Problem

- Represent graph partition Γ_V^K by matrix $X = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_K}]$
- $\mathbf{x_i}$: column indicator vector with ones in the rows corresponding to harvesters in cluster i
- Degree matrix $D = \text{diag}(W1_M)$
- Rewrite links and deg as

$$\text{links}(V_i, V_i) = \mathbf{x_i}^T W \mathbf{x_i}$$
$$\text{deg}(V_i) = \mathbf{x_i}^T D \mathbf{x_i}$$

- KNassoc maximization problem becomes

$$\text{maximize} \quad \text{KNassoc}(X) = \frac{1}{K} \sum_{i=1}^{K} \frac{\mathbf{x_i}^T W \mathbf{x_i}}{\mathbf{x_i}^T D \mathbf{x_i}}$$

$$\text{subject to} \quad X \in \{0,1\}^{M \times K}$$
$$X1_K = 1_M$$

The Discrete Optimization Problem

- Represent graph partition Γ_V^K by matrix $X = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_K}]$
- $\mathbf{x_i}$: column indicator vector with ones in the rows corresponding to harvesters in cluster i
- Degree matrix $D = \text{diag}(W 1_M)$
- Rewrite links and deg as

$$\text{links}(V_i, V_i) = \mathbf{x_i}^T W \mathbf{x_i}$$
$$\deg(V_i) = \mathbf{x_i}^T D \mathbf{x_i}$$

- KNassoc maximization problem becomes

$$\begin{aligned} \text{maximize} \quad & \text{KNassoc}(X) = \frac{1}{K} \sum_{i=1}^{K} \frac{\mathbf{x_i}^T W \mathbf{x_i}}{\mathbf{x_i}^T D \mathbf{x_i}} \\ \text{subject to} \quad & X \in \{0,1\}^{M \times K} \\ & X 1_K = 1_M \end{aligned}$$

Spectral Clustering

- KNassoc maximization problem has exponential complexity even for $K = 2$
- Define $Z = X(X^T DX)^{-1/2}$
- Reformulate problem

$$\begin{aligned} \text{maximize} \quad & \text{KNassoc}(Z) = \frac{1}{K}\text{tr}(Z^T WZ) \\ \text{subject to} \quad & Z^T DZ = I_K \end{aligned}$$

- Relax Z into continuous domain
- Solve generalized eigenvalue problem

$$W\bar{\mathbf{z}}_\mathbf{i} = \lambda D\bar{\mathbf{z}}_\mathbf{i}$$

- Form optimal continuous partition matrix $\overline{Z} = [\bar{\mathbf{z}}_\mathbf{1}, \bar{\mathbf{z}}_\mathbf{2}, \ldots, \bar{\mathbf{z}}_\mathbf{K}]$ and discretize to get near global-optimal solution (Yu and Shi, 2003)

Spectral Clustering

- KNassoc maximization problem has exponential complexity even for $K = 2$
- Define $Z = X(X^T DX)^{-1/2}$
- Reformulate problem

$$\text{maximize} \quad \text{KNassoc}(Z) = \frac{1}{K}\text{tr}(Z^T WZ)$$

$$\text{subject to} \quad Z^T DZ = I_K$$

- Relax Z into continuous domain
- Solve generalized eigenvalue problem

$$W\bar{\mathbf{z}}_\mathbf{i} = \lambda D\bar{\mathbf{z}}_\mathbf{i}$$

- Form optimal continuous partition matrix $\overline{Z} = [\bar{\mathbf{z}}_\mathbf{1}, \bar{\mathbf{z}}_\mathbf{2}, \ldots, \bar{\mathbf{z}}_\mathbf{K}]$ and discretize to get near global-optimal solution (Yu and Shi, 2003)

Spectral Clustering

- KNassoc maximization problem has exponential complexity even for $K = 2$
- Define $Z = X(X^T DX)^{-1/2}$
- Reformulate problem

$$\begin{aligned} \text{maximize} \quad & \text{KNassoc}(Z) = \frac{1}{K}\text{tr}(Z^T WZ) \\ \text{subject to} \quad & Z^T DZ = I_K \end{aligned}$$

- Relax Z into continuous domain
- Solve generalized eigenvalue problem

$$W\bar{\mathbf{z}}_\mathbf{i} = \lambda D\bar{\mathbf{z}}_\mathbf{i}$$

- Form optimal continuous partition matrix $\overline{Z} = [\bar{\mathbf{z}}_\mathbf{1}, \bar{\mathbf{z}}_\mathbf{2}, \ldots, \bar{\mathbf{z}}_\mathbf{K}]$ and discretize to get near global-optimal solution (Yu and Shi, 2003)

Spectral Clustering

- KNassoc maximization problem has exponential complexity even for $K = 2$
- Define $Z = X(X^T DX)^{-1/2}$
- Reformulate problem

$$\begin{aligned} \text{maximize} \quad & \text{KNassoc}(Z) = \frac{1}{K}\text{tr}(Z^T WZ) \\ \text{subject to} \quad & Z^T DZ = I_K \end{aligned}$$

- Relax Z into continuous domain
- Solve generalized eigenvalue problem

$$W\bar{\mathbf{z}}_\mathbf{i} = \lambda D\bar{\mathbf{z}}_\mathbf{i}$$

- Form optimal continuous partition matrix $\overline{Z} = [\bar{\mathbf{z}}_\mathbf{1}, \bar{\mathbf{z}}_\mathbf{2}, \ldots, \bar{\mathbf{z}}_\mathbf{K}]$ and discretize to get near global-optimal solution (Yu and Shi, 2003)

Spectral Clustering

- KNassoc maximization problem has exponential complexity even for $K = 2$
- Define $Z = X(X^T DX)^{-1/2}$
- Reformulate problem

$$\begin{aligned} \text{maximize} \quad & \text{KNassoc}(Z) = \frac{1}{K}\text{tr}(Z^T WZ) \\ \text{subject to} \quad & Z^T DZ = I_K \end{aligned}$$

- Relax Z into continuous domain
- Solve generalized eigenvalue problem

$$W\bar{\mathbf{z}}_\mathbf{i} = \lambda D\bar{\mathbf{z}}_\mathbf{i}$$

- Form optimal continuous partition matrix $\overline{Z} = [\bar{\mathbf{z}}_\mathbf{1}, \bar{\mathbf{z}}_\mathbf{2}, \ldots, \bar{\mathbf{z}}_\mathbf{K}]$ and discretize to get near global-optimal solution (Yu and Shi, 2003)

Spectral Clustering

- KNassoc maximization problem has exponential complexity even for $K = 2$
- Define $Z = X(X^T DX)^{-1/2}$
- Reformulate problem

$$\begin{aligned} \text{maximize} \quad & \text{KNassoc}(Z) = \frac{1}{K}\text{tr}(Z^T WZ) \\ \text{subject to} \quad & Z^T DZ = I_K \end{aligned}$$

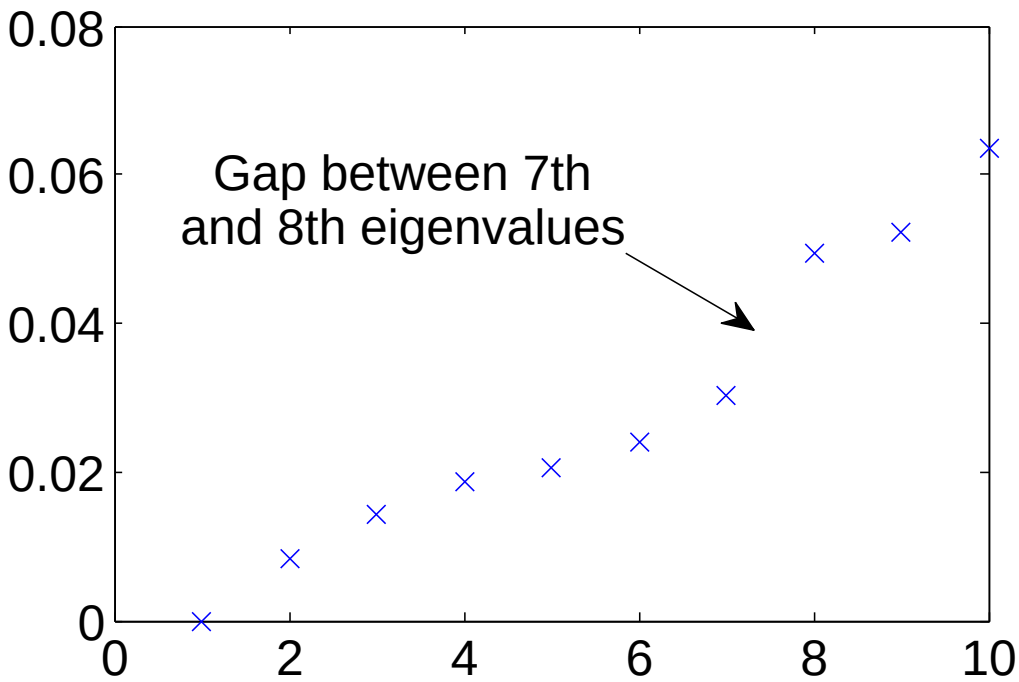
- Relax Z into continuous domain
- Solve generalized eigenvalue problem

$$W\bar{\mathbf{z}}_\mathbf{i} = \lambda D\bar{\mathbf{z}}_\mathbf{i}$$

- Form optimal continuous partition matrix $\overline{Z} = [\bar{\mathbf{z}}_\mathbf{1}, \bar{\mathbf{z}}_\mathbf{2}, \ldots, \bar{\mathbf{z}}_\mathbf{K}]$ and discretize to get near global-optimal solution (Yu and Shi, 2003)

Choosing the Number of Clusters

- How do we choose the number of clusters?
- Heuristic for spectral clustering: look at the gap between eigenvalues of the Laplacian matrix of the graph (von Luxburg, 2007)



Ten smallest eigenvalues of Laplacian matrix

Choosing Edge Weights

- Edge weights w_{ij} represent strength of connection between harvesters i and j
- We cannot observe direct relationships between harvesters
- Use indirect relationships to determine edge weights
  - Similarity in spam server usage
  - Similarity in temporal spamming
  - Similarity in temporal harvesting
- Choice of similarity measure determines topology of the graph
- Poor choice could lead to detecting no community structure
- Create coincidence matrix H as intermediate step to creating adjacency matrix W

Choosing Edge Weights

- Edge weights w_{ij} represent strength of connection between harvesters i and j
- We cannot observe direct relationships between harvesters
- Use indirect relationships to determine edge weights
  - Similarity in spam server usage
  - Similarity in temporal spamming
  - Similarity in temporal harvesting
- Choice of similarity measure determines topology of the graph
- Poor choice could lead to detecting no community structure
- Create coincidence matrix H as intermediate step to creating adjacency matrix W

Choosing Edge Weights

- Edge weights w_{ij} represent strength of connection between harvesters i and j
- We cannot observe direct relationships between harvesters
- Use indirect relationships to determine edge weights
  - Similarity in spam server usage
  - Similarity in temporal spamming
  - Similarity in temporal harvesting
- Choice of similarity measure determines topology of the graph
- Poor choice could lead to detecting no community structure
- Create coincidence matrix H as intermediate step to creating adjacency matrix W

Choosing Edge Weights

- Edge weights w_{ij} represent strength of connection between harvesters i and j
- We cannot observe direct relationships between harvesters
- Use indirect relationships to determine edge weights
  - Similarity in spam server usage
  - Similarity in temporal spamming
  - Similarity in temporal harvesting
- Choice of similarity measure determines topology of the graph
- Poor choice could lead to detecting no community structure
- Create coincidence matrix H as intermediate step to creating adjacency matrix W

Choosing Edge Weights

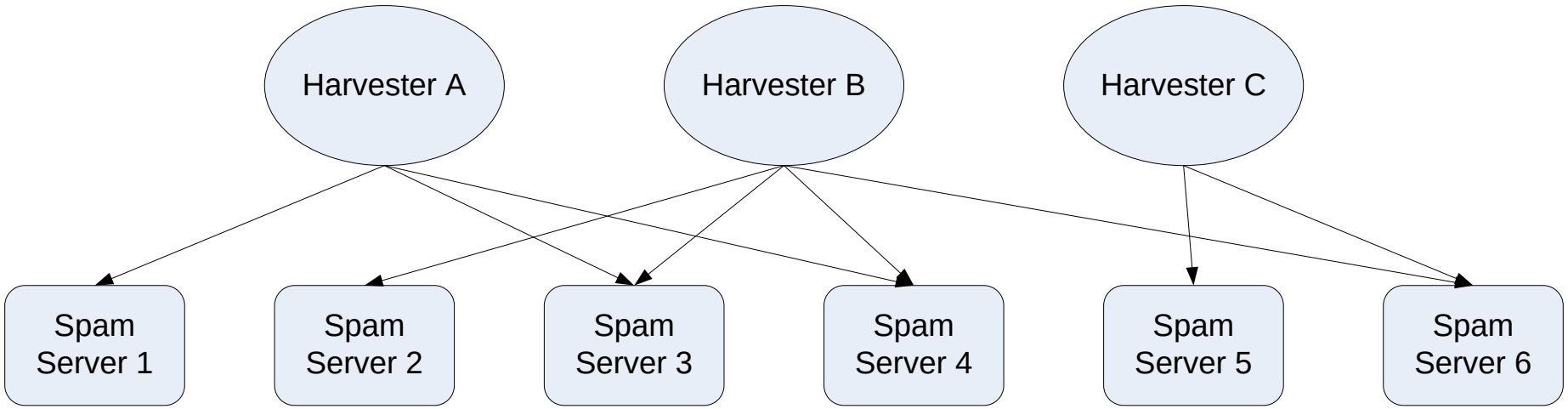
- Edge weights w_{ij} represent strength of connection between harvesters i and j
- We cannot observe direct relationships between harvesters
- Use indirect relationships to determine edge weights
  - Similarity in spam server usage
  - Similarity in temporal spamming
  - Similarity in temporal harvesting
- Choice of similarity measure determines topology of the graph
- Poor choice could lead to detecting no community structure
- Create coincidence matrix H as intermediate step to creating adjacency matrix W

Choosing Edge Weights

- Edge weights w_{ij} represent strength of connection between harvesters i and j
- We cannot observe direct relationships between harvesters
- Use indirect relationships to determine edge weights
  - Similarity in spam server usage
  - Similarity in temporal spamming
  - Similarity in temporal harvesting
- Choice of similarity measure determines topology of the graph
- Poor choice could lead to detecting no community structure
- Create coincidence matrix H as intermediate step to creating adjacency matrix W

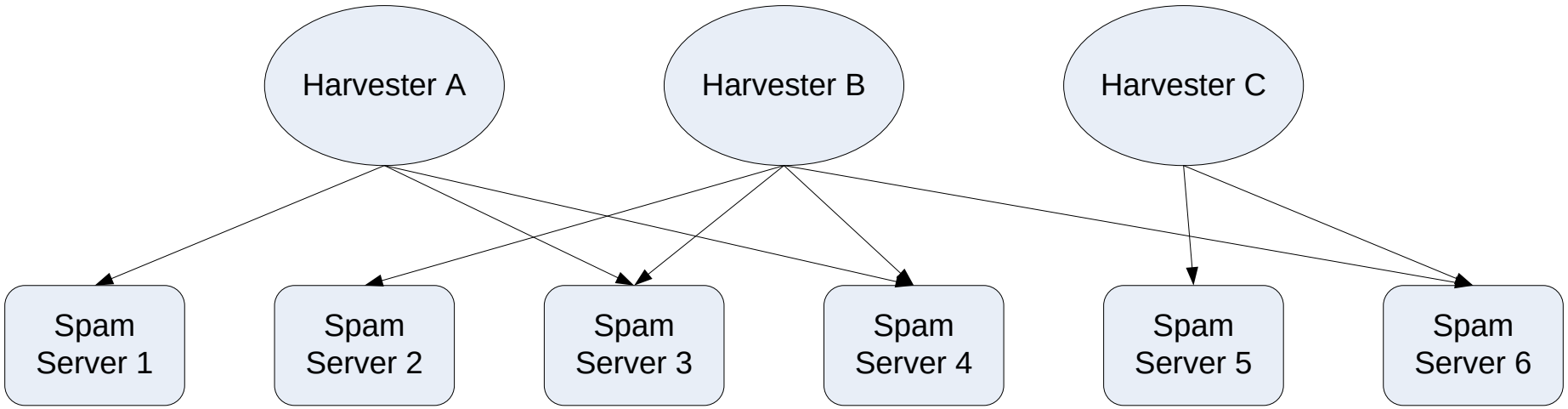
Similarity in Spam Server Usage

- Spammers need spam servers to send emails
- Common usage of spam servers between harvesters may indicate social connection
- Create bipartite graph of harvesters and spam servers
- Choose edge weights based on correlation in spam server usage



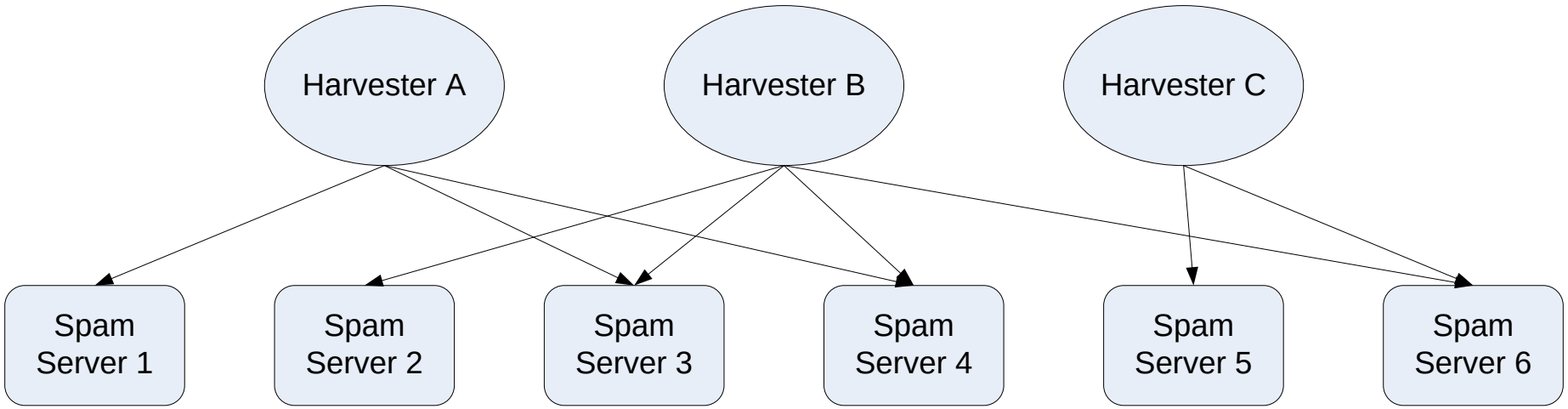
Similarity in Spam Server Usage

- Spammers need spam servers to send emails
- Common usage of spam servers between harvesters may indicate social connection
- Create bipartite graph of harvesters and spam servers
- Choose edge weights based on correlation in spam server usage



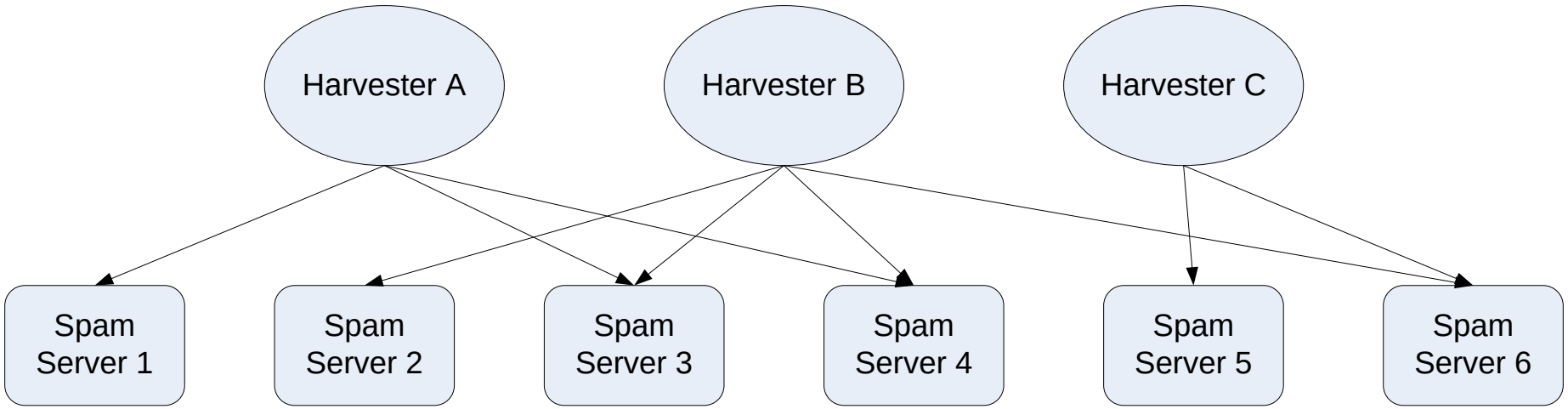
Similarity in Spam Server Usage

- Spammers need spam servers to send emails
- Common usage of spam servers between harvesters may indicate social connection
- Create bipartite graph of harvesters and spam servers
- Choose edge weights based on correlation in spam server usage



Similarity in Spam Server Usage

- Spammers need spam servers to send emails
- Common usage of spam servers between harvesters may indicate social connection
- Create bipartite graph of harvesters and spam servers
- Choose edge weights based on correlation in spam server usage



Similarity in Spam Server Usage Coincidence Matrix

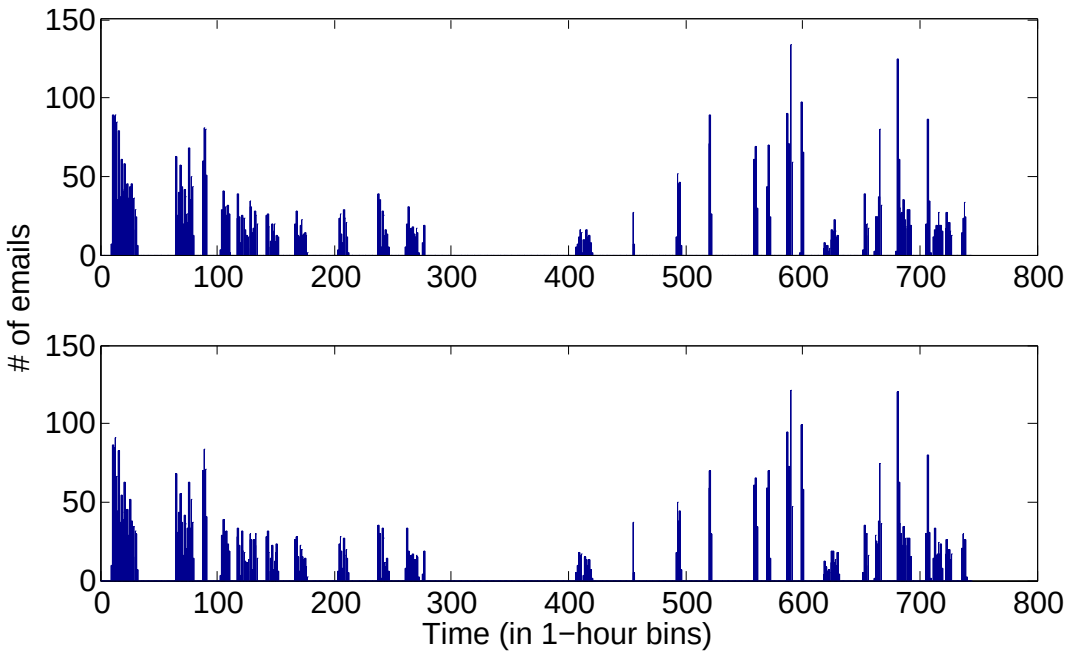
- Create coincidence matrix H between harvesters and spam servers

$$H = \left[\frac{p_{ij}}{d_j e_i} \right]_{i,j=1}^{M,N}$$

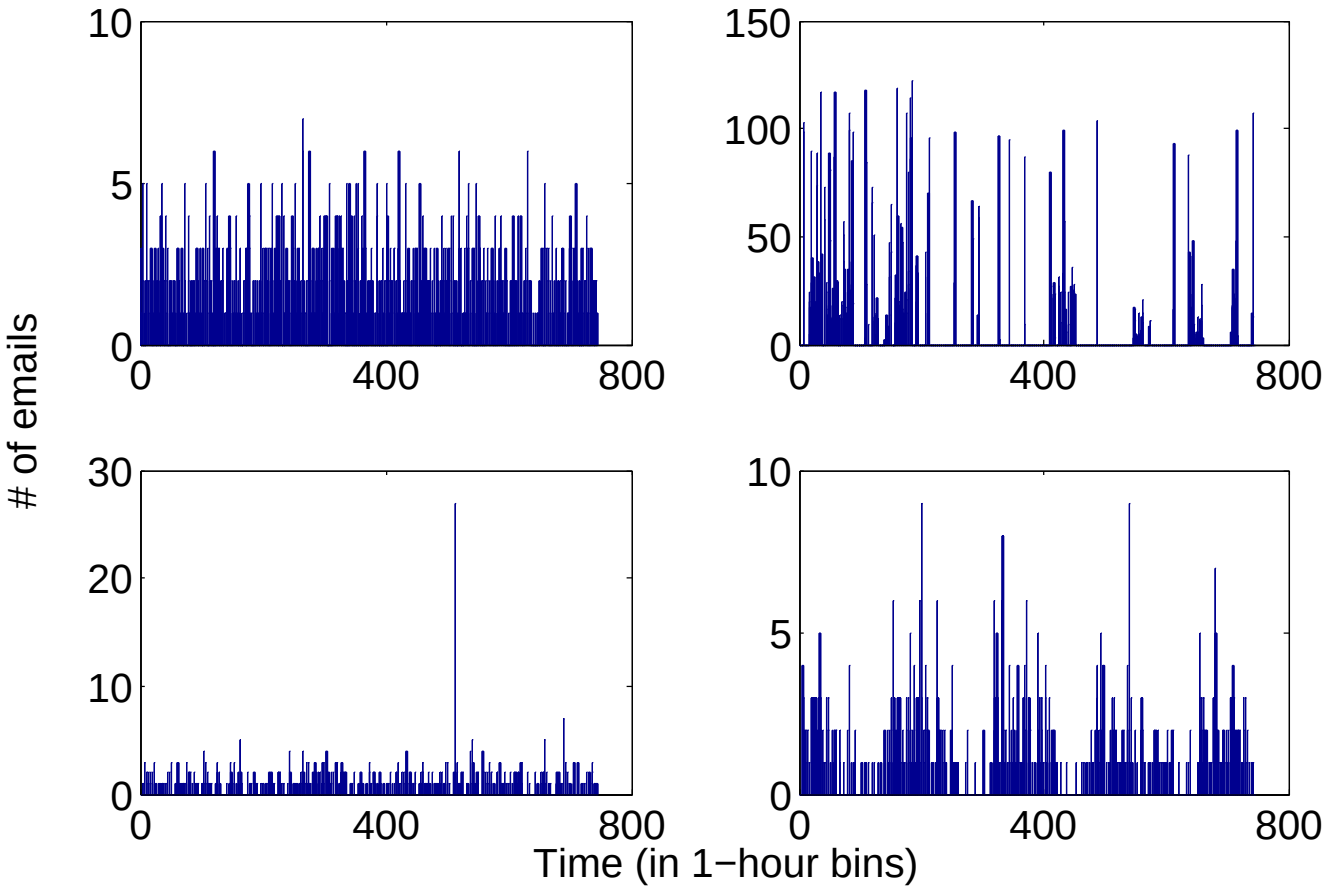
- p_{ij}: the number of emails sent using spam server j to email addresses collected by harvester i
- d_j: the total number of emails sent by spam server j
- e_i: the total number of email addresses collected by harvester i
- Entries of incidence matrix represent harvester i's percentage of usage of spam server j per address he has acquired

Temporal Similarity

- Common temporal patterns of activity may also indicate social connection
- Look at number of emails sent or email addresses collected as function of time
- Discretize time into 1-hour intervals



Sample Temporal Histograms



Sample temporal spamming histograms representing four types of distributions

Temporal Similarity Coincidence Matrices

- Choose edge weights based on correlation in number of emails sent during each time interval
- Similarity in temporal spamming
  - Create coincidence matrix H between harvesters and discretized time intervals

$$H = \left[\frac{s_{ij}}{e_i} \right]_{i,j=1}^{M,N}$$

  - s_{ij}: number of emails sent by harvester i during jth time interval
  - e_i: the total number of email addresses collected by harvester i
- Similarity in temporal harvesting
  - Create coincidence matrix H between harvesters and discretized time intervals

$$H = [a_{ij}]_{i,j=1}^{M,N}$$

  - a_{ij}: number of addresses collected by harvester i during jth time interval

Temporal Similarity Coincidence Matrices

- Choose edge weights based on correlation in number of emails sent during each time interval
- Similarity in temporal spamming
  - Create coincidence matrix H between harvesters and discretized time intervals

$$H = \left[\frac{s_{ij}}{e_i} \right]_{i,j=1}^{M,N}$$

  - s_{ij}: number of emails sent by harvester i during jth time interval
  - e_i: the total number of email addresses collected by harvester i
- Similarity in temporal harvesting
  - Create coincidence matrix H between harvesters and discretized time intervals

$$H = [a_{ij}]_{i,j=1}^{M,N}$$

  - a_{ij}: number of addresses collected by harvester i during jth time interval

Temporal Similarity Coincidence Matrices

- Choose edge weights based on correlation in number of emails sent during each time interval
- Similarity in temporal spamming
  - Create coincidence matrix H between harvesters and discretized time intervals

$$H = \left[\frac{s_{ij}}{e_i} \right]_{i,j=1}^{M,N}$$

  - s_{ij}: number of emails sent by harvester i during jth time interval
  - e_i: the total number of email addresses collected by harvester i
- Similarity in temporal harvesting
  - Create coincidence matrix H between harvesters and discretized time intervals

$$H = [a_{ij}]_{i,j=1}^{M,N}$$

  - a_{ij}: number of addresses collected by harvester i during jth time interval

Creating the Adjacency Matrix

- From coincidence matrix H we can obtain a matrix of unnormalized pairwise similarities $S = HH^T$
- Normalize S to obtain matrix of normalized pairwise similarities $N = D^{-1/2} S D^{-1/2}$
  - $D = \text{diag}(S)$
  - Scales similarities so each harvester's self-similarity is 1
  - Ensures each harvester is equally important
- Connect harvesters to their k nearest neighbors according to similarities in N to form adjacency matrix W
  - Results in sparser adjacency matrix
  - How to choose k?
  - Heuristic: Choose $k = \log n$ to start and increase as necessary to avoid artificially disconnecting components (von Luxburg, 2007)

Creating the Adjacency Matrix

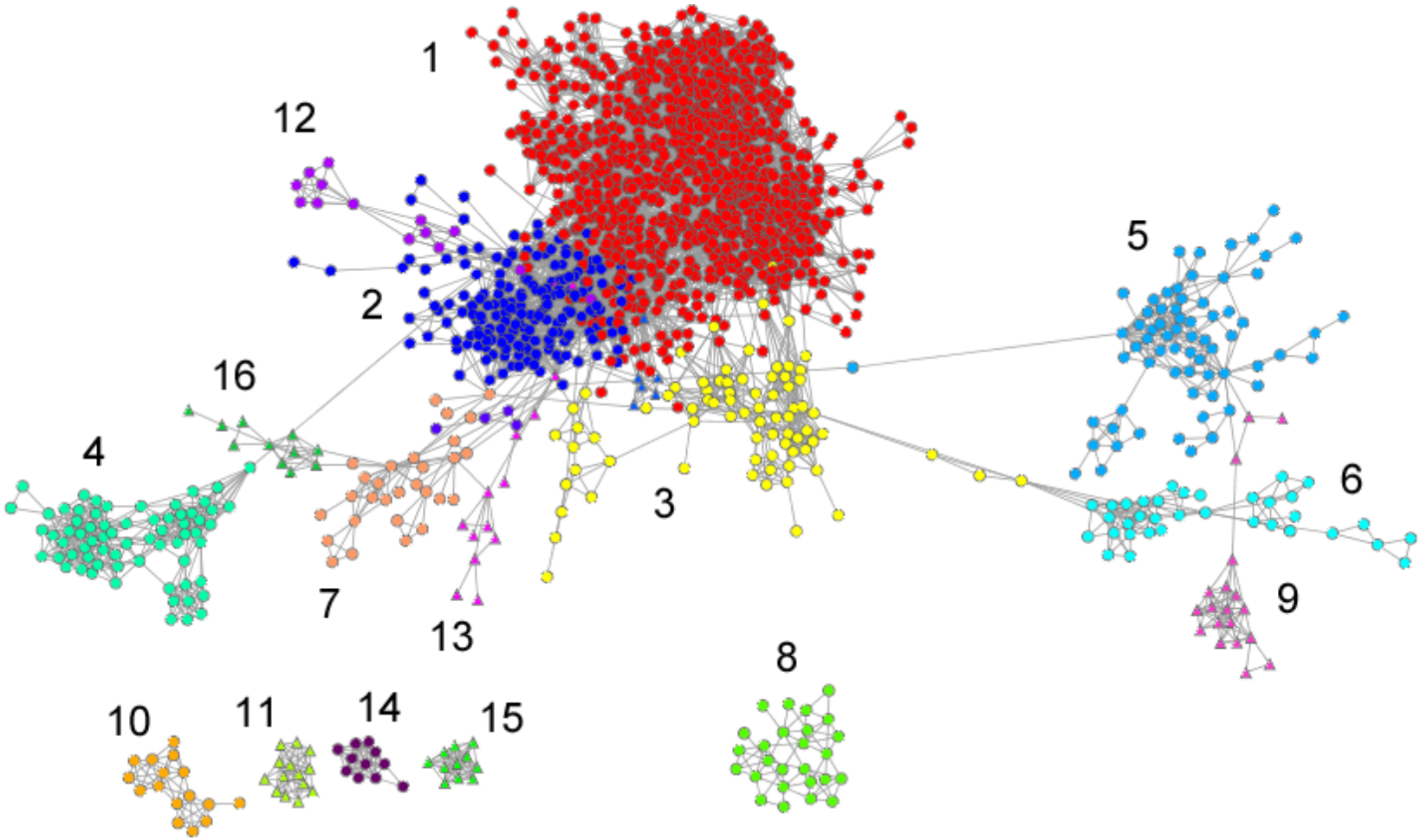
- From coincidence matrix H we can obtain a matrix of unnormalized pairwise similarities $S = HH^T$
- Normalize S to obtain matrix of normalized pairwise similarities $N = D^{-1/2} S D^{-1/2}$
  - $D = \text{diag}(S)$
  - Scales similarities so each harvester's self-similarity is 1
  - Ensures each harvester is equally important
- Connect harvesters to their k nearest neighbors according to similarities in N to form adjacency matrix W
  - Results in sparser adjacency matrix
  - How to choose k?
  - Heuristic: Choose $k = \log n$ to start and increase as necessary to avoid artificially disconnecting components (von Luxburg, 2007)

Creating the Adjacency Matrix

- From coincidence matrix H we can obtain a matrix of unnormalized pairwise similarities $S = HH^T$
- Normalize S to obtain matrix of normalized pairwise similarities $N = D^{-1/2} S D^{-1/2}$
  - $D = \text{diag}(S)$
  - Scales similarities so each harvester's self-similarity is 1
  - Ensures each harvester is equally important
- Connect harvesters to their k nearest neighbors according to similarities in N to form adjacency matrix W
  - Results in sparser adjacency matrix
  - How to choose k?
  - Heuristic: Choose $k = \log n$ to start and increase as necessary to avoid artificially disconnecting components (von Luxburg, 2007)

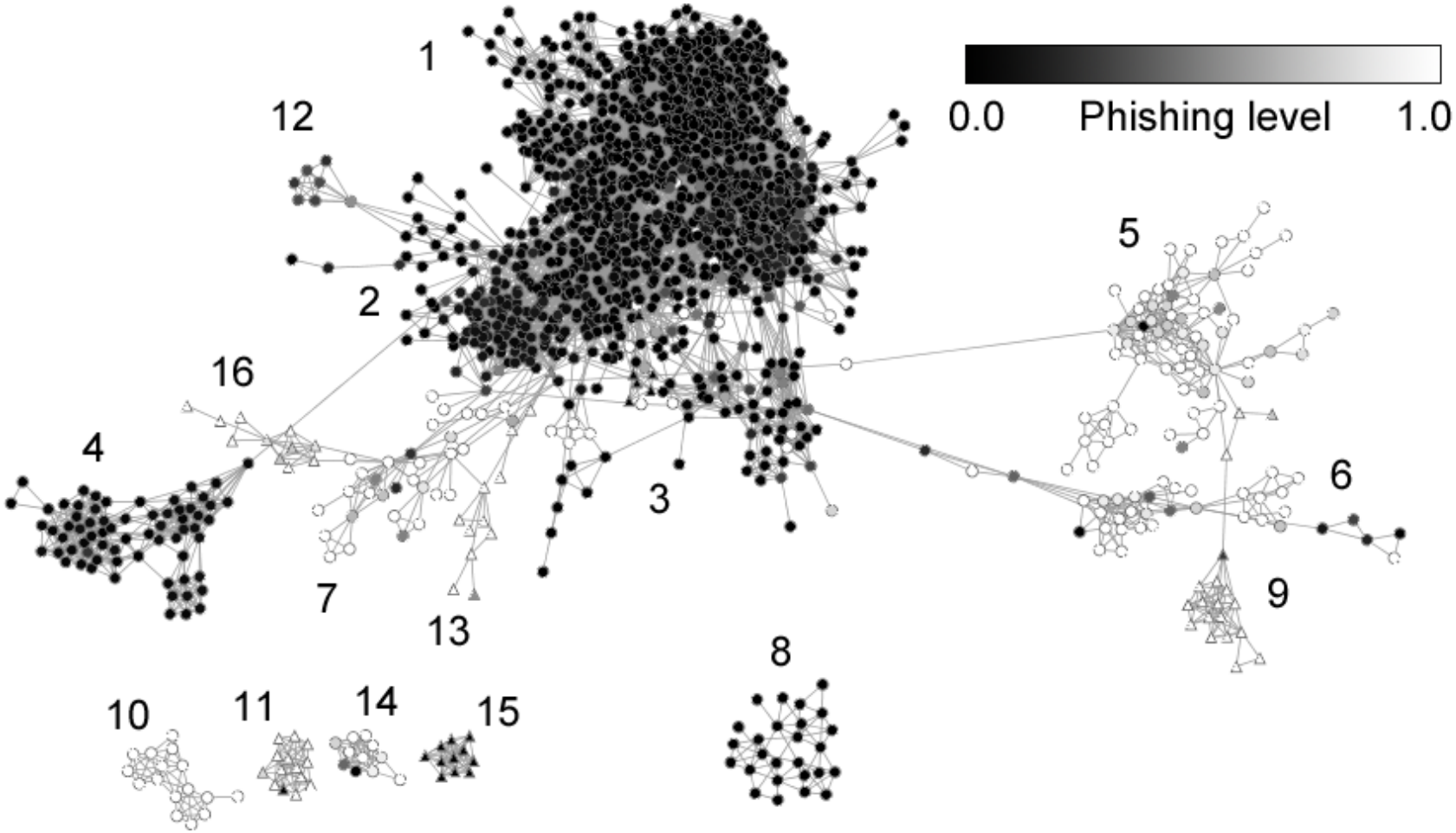
Outline

Similarity in Spam Server Usage



Results from October 2006 using similarity in spam server usage (visualization created using Cytoscape)

Alternate View Colored By Phishing Level



Results from October 2006 colored by phishing level

Distribution of Phishers in Clusters

Distribution of phishers in clusters from October 2006 results

Label	1	2	3	4	5	6	7	8
Cluster size	1040	188	77	68	68	35	29	26
# of phishers	17	0	10	0	65	28	24	0
% of phishers	1.63	0	13.0	0	95.6	80	82.8	0

Label	9	10	11	12	13	14	15	16
Cluster size	19	16	14	14	14	11	11	11
# of phishers	18	16	13	1	12	9	0	11
% of phishers	94.7	100	92.9	7.14	85.7	81.8	0	100

- Very few phishers in large, loosely-connected cluster
- Many small, tightly-connected clusters have high concentration of phishers

Cluster Validation Indices

- Rand index: measure of agreement between clustering results and labels

$$\text{Rand index} = \frac{a+d}{a+b+c+d}$$

  - a: number of pairs of nodes with same label and in same cluster
  - b: number of pairs with same label but in different clusters
  - c: number of pairs with different labels but in the same cluster
  - d: number of pairs with different labels and in different clusters
- Adjusted Rand index: Rand index corrected for chance (Hubert and Arabie, 1985)
- Expected adjusted Rand index for random clustering result is 0
- Label clusters as phishing clusters if ratio of phishers to harvesters > 0.5
- Use phisher or non-phisher as label for each harvester
- Look for agreement between harvester labels and cluster labels

Cluster Validation Indices

- Rand index: measure of agreement between clustering results and labels

$$\text{Rand index} = \frac{a+d}{a+b+c+d}$$

  - a: number of pairs of nodes with same label and in same cluster
  - b: number of pairs with same label but in different clusters
  - c: number of pairs with different labels but in the same cluster
  - d: number of pairs with different labels and in different clusters
- Adjusted Rand index: Rand index corrected for chance (Hubert and Arabie, 1985)
- Expected adjusted Rand index for random clustering result is 0
- Label clusters as phishing clusters if ratio of phishers to harvesters > 0.5
- Use phisher or non-phisher as label for each harvester
- Look for agreement between harvester labels and cluster labels

Cluster Validation Indices

- Rand index: measure of agreement between clustering results and labels

$$\text{Rand index} = \frac{a+d}{a+b+c+d}$$

  - a: number of pairs of nodes with same label and in same cluster
  - b: number of pairs with same label but in different clusters
  - c: number of pairs with different labels but in the same cluster
  - d: number of pairs with different labels and in different clusters
- Adjusted Rand index: Rand index corrected for chance (Hubert and Arabie, 1985)
- Expected adjusted Rand index for random clustering result is 0
- Label clusters as phishing clusters if ratio of phishers to harvesters > 0.5
- Use phisher or non-phisher as label for each harvester
- Look for agreement between harvester labels and cluster labels

Cluster Validation Indices

- Rand index: measure of agreement between clustering results and labels

$$\text{Rand index} = \frac{a+d}{a+b+c+d}$$

  - a: number of pairs of nodes with same label and in same cluster
  - b: number of pairs with same label but in different clusters
  - c: number of pairs with different labels but in the same cluster
  - d: number of pairs with different labels and in different clusters
- Adjusted Rand index: Rand index corrected for chance (Hubert and Arabie, 1985)
- Expected adjusted Rand index for random clustering result is 0
- Label clusters as phishing clusters if ratio of phishers to harvesters > 0.5
- Use phisher or non-phisher as label for each harvester
- Look for agreement between harvester labels and cluster labels

Cluster Validation Indices

- Rand index: measure of agreement between clustering results and labels

$$\text{Rand index} = \frac{a+d}{a+b+c+d}$$

  - a: number of pairs of nodes with same label and in same cluster
  - b: number of pairs with same label but in different clusters
  - c: number of pairs with different labels but in the same cluster
  - d: number of pairs with different labels and in different clusters
- Adjusted Rand index: Rand index corrected for chance (Hubert and Arabie, 1985)
- Expected adjusted Rand index for random clustering result is 0
- Label clusters as phishing clusters if ratio of phishers to harvesters > 0.5
- Use phisher or non-phisher as label for each harvester
- Look for agreement between harvester labels and cluster labels

Cluster Validation Indices

- Rand index: measure of agreement between clustering results and labels

$$\text{Rand index} = \frac{a+d}{a+b+c+d}$$

  - a: number of pairs of nodes with same label and in same cluster
  - b: number of pairs with same label but in different clusters
  - c: number of pairs with different labels but in the same cluster
  - d: number of pairs with different labels and in different clusters
- Adjusted Rand index: Rand index corrected for chance (Hubert and Arabie, 1985)
- Expected adjusted Rand index for random clustering result is 0
- Label clusters as phishing clusters if ratio of phishers to harvesters > 0.5
- Use phisher or non-phisher as label for each harvester
- Look for agreement between harvester labels and cluster labels

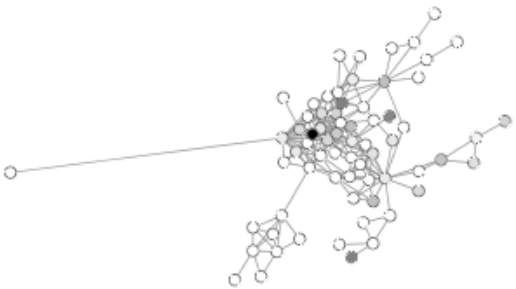
Validation Indices For Similarity in Spam Server Usage

Validation indices for similarity in spam server usage results

Year	2006		2007		
Month	July	October	January	April	July
Rand index	0.884	0.936	0.923	0.937	0.880
Adj. Rand index	0.759	0.847	0.803	0.802	0.618

- Very high Rand and adjusted Rand indices indicates good agreement between labels and clustering results
- Results highly unlikely to be caused by chance

Top Subject Lines in Phishing Clusters



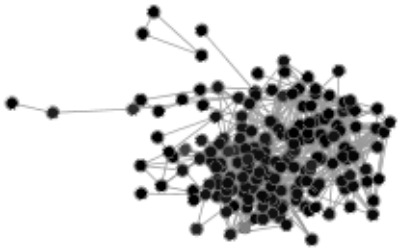
Cluster 5

Subject line	Hits
Password Change Required	126
Question from eBay Member	69
Credit Union OnlineÂ® $50 Reward Survey	47
PayPal Account	42
PayPal Account - Suspicious Activity	40

Cluster 9

Subject line	Hits
Notification from Billing Department	49
IMPORTANT: Notification of limited accounts	25
PayPal Account Review Department	22
Notification of Limited Account Access	13
A secondary e-mail address has been added to your	11

Top Subject Lines in Non-Phishing Clusters



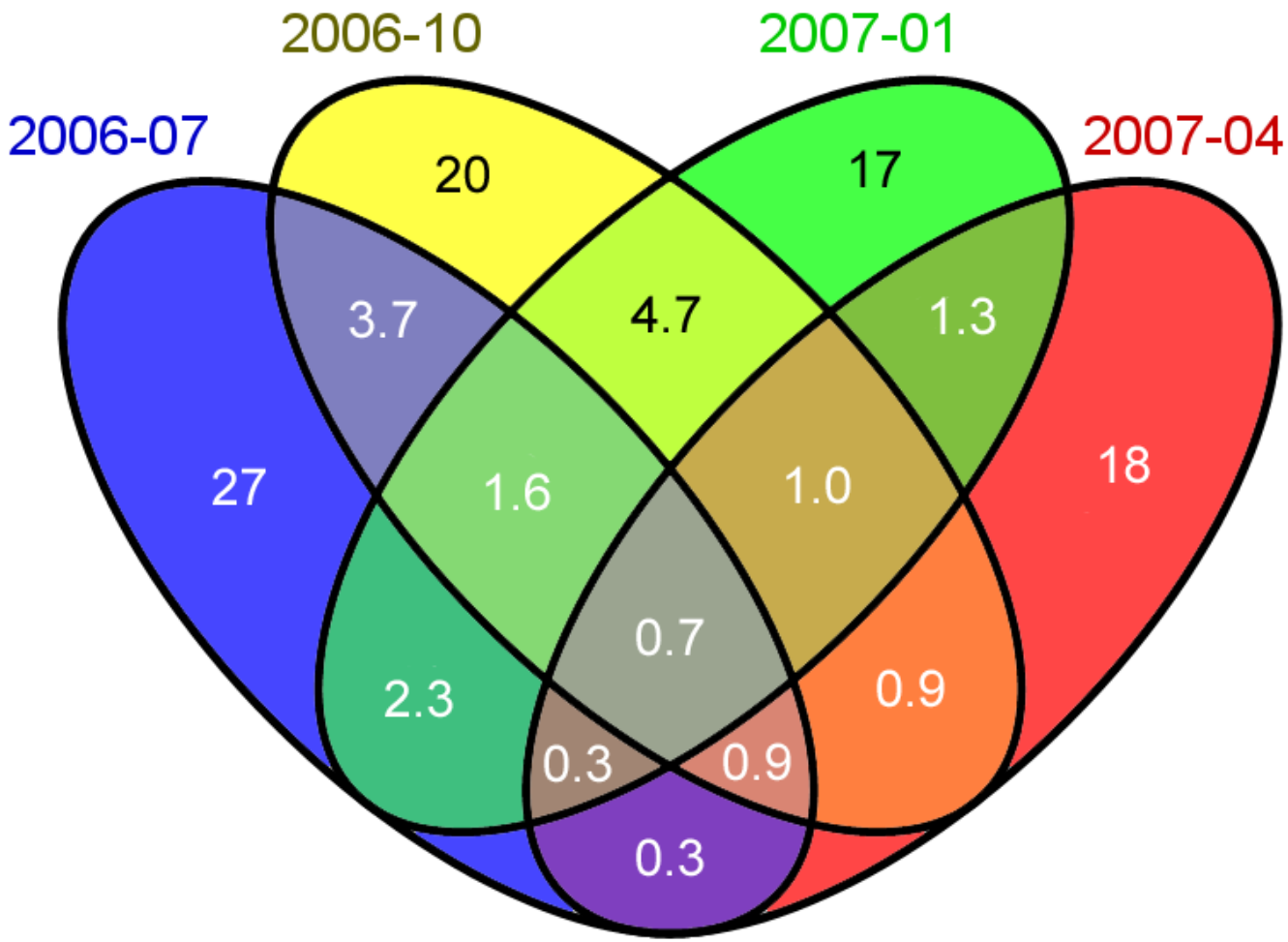
Cluster 2

Subject line	Hits
tthemee	6893
St ock 6	6729
Notification	4516
Access granted to send emails to	4495
Thanks for joining	4405

Cluster 4

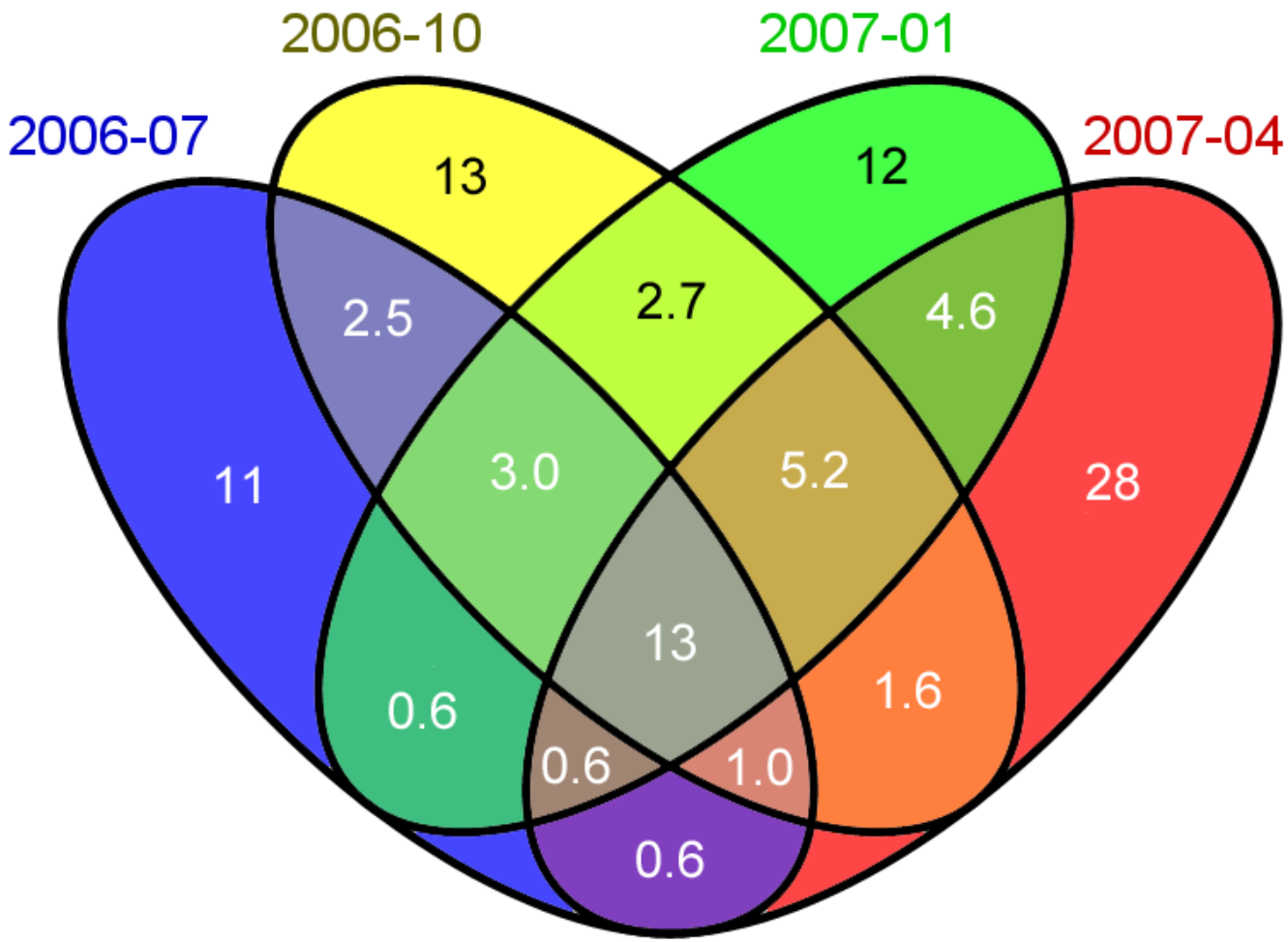
Subject line	Hits
Make Money by Sharing Your Life with Friends and F	1027
Premiere Professional & Executive Registries Invit	750
Texas Land/Golf is the Buzz	459
Keys to Stock Market Success	408
An Entire Case of Fine Wine plus Exclusive Gift fo	367

Venn Diagram of Phishers' Life Times



Venn diagram of phishers' life times as percentage of total (1805 total phishers)

Venn Diagram of Non-Phishers' Life Times



Venn diagram of non-phishers' life times as percentage of total (4801 total non-phishers)

Findings from Similarity in Spam Server Usage

- Clustering divides spammers into communities of mostly phishers and mostly non-phishers
- Empirical evidence that phishers tend to form small groups and share resources
- Phishers have shorter life times than non-phishers
- Discovered community structure is highly unlikely by chance

Findings from Similarity in Spam Server Usage

- Clustering divides spammers into communities of mostly phishers and mostly non-phishers
- Empirical evidence that phishers tend to form small groups and share resources
- Phishers have shorter life times than non-phishers
- Discovered community structure is highly unlikely by chance

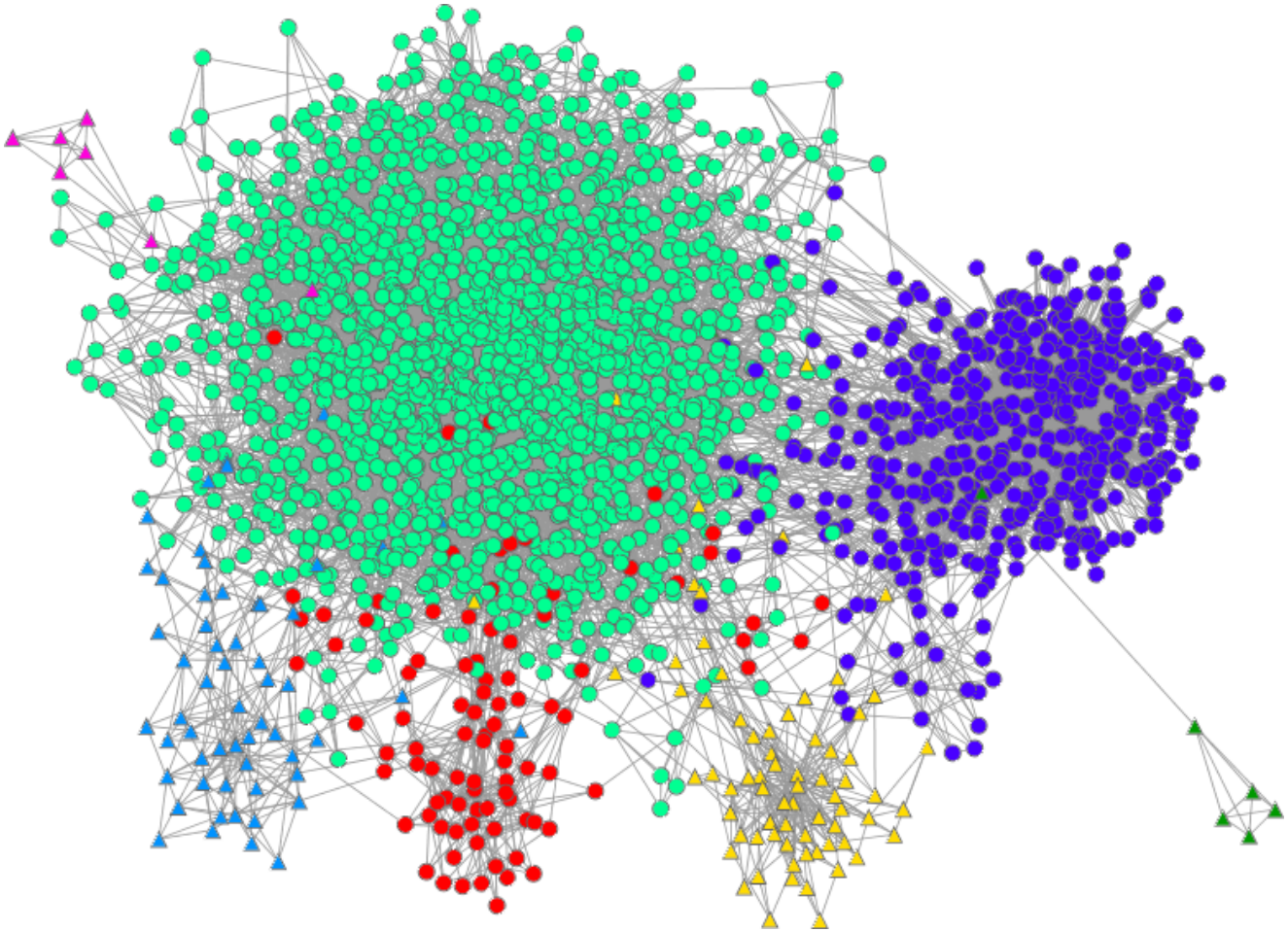
Findings from Similarity in Spam Server Usage

- Clustering divides spammers into communities of mostly phishers and mostly non-phishers
- Empirical evidence that phishers tend to form small groups and share resources
- Phishers have shorter life times than non-phishers
- Discovered community structure is highly unlikely by chance

Findings from Similarity in Spam Server Usage

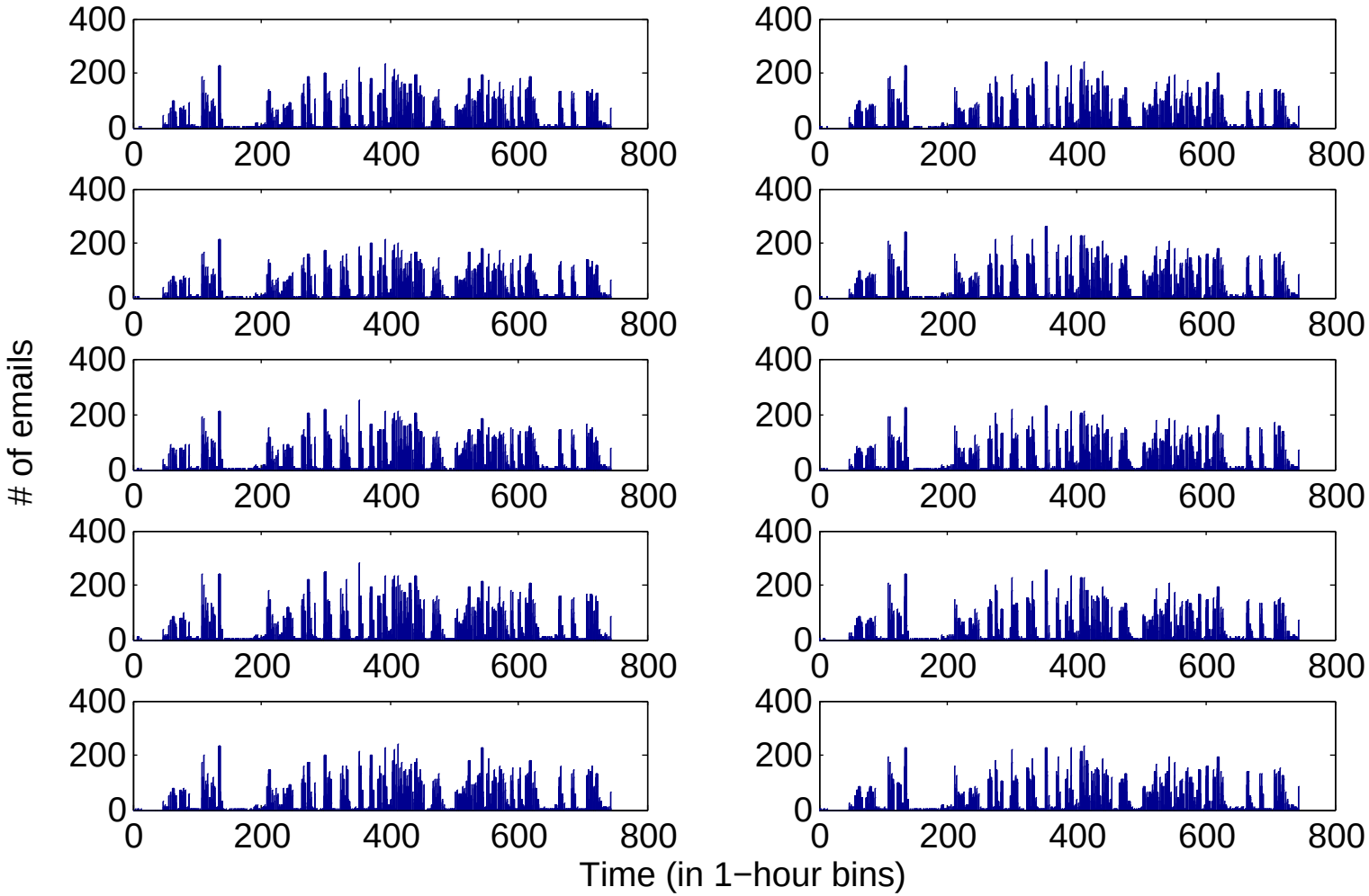
- Clustering divides spammers into communities of mostly phishers and mostly non-phishers
- Empirical evidence that phishers tend to form small groups and share resources
- Phishers have shorter life times than non-phishers
- Discovered community structure is highly unlikely by chance

Similarity in Temporal Spamming



Results from October 2006 using similarity in temporal spamming

Temporal Spamming Histograms



Temporal spamming histograms of ten harvesters from same cluster

Statistics from Similarity in Temporal Spamming

Average temporal spamming correlation coefficients between two harvesters in the aforementioned group

Year	2006		2007		
Month	July	October	January	April	July
ρ_{avg}	0.979	0.988	0.933	0.950	0.937

- IP addresses of all harvesters in this group have 208.66.195/24 prefix
- These harvesters are among the heaviest spammers in each month
- We discovered several other groups with coherent temporal behavior and similar IP addresses

Statistics from Similarity in Temporal Spamming

Average temporal spamming correlation coefficients between two harvesters in the aforementioned group

Year	2006		2007		
Month	July	October	January	April	July
ρ_{avg}	0.979	0.988	0.933	0.950	0.937

- IP addresses of all harvesters in this group have 208.66.195/24 prefix
- These harvesters are among the heaviest spammers in each month
- We discovered several other groups with coherent temporal behavior and similar IP addresses

Statistics from Similarity in Temporal Spamming

Average temporal spamming correlation coefficients between two harvesters in the aforementioned group

Year	2006		2007		
Month	July	October	January	April	July
ρ_{avg}	0.979	0.988	0.933	0.950	0.937

- IP addresses of all harvesters in this group have 208.66.195/24 prefix
- These harvesters are among the heaviest spammers in each month
- We discovered several other groups with coherent temporal behavior and similar IP addresses

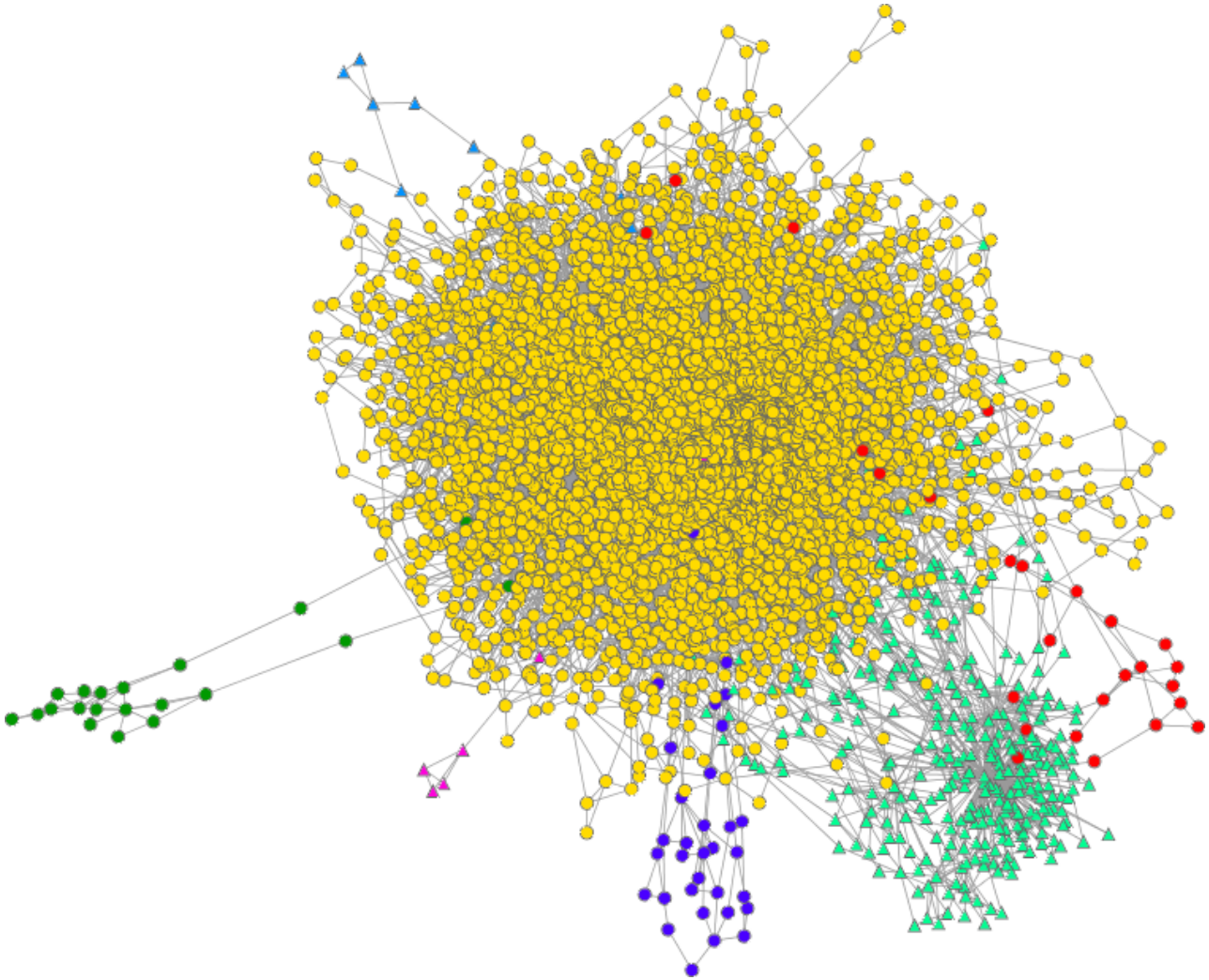
Statistics from Similarity in Temporal Spamming

Average temporal spamming correlation coefficients between two harvesters in the aforementioned group

Year	2006		2007		
Month	July	October	January	April	July
ρ_{avg}	0.979	0.988	0.933	0.950	0.937

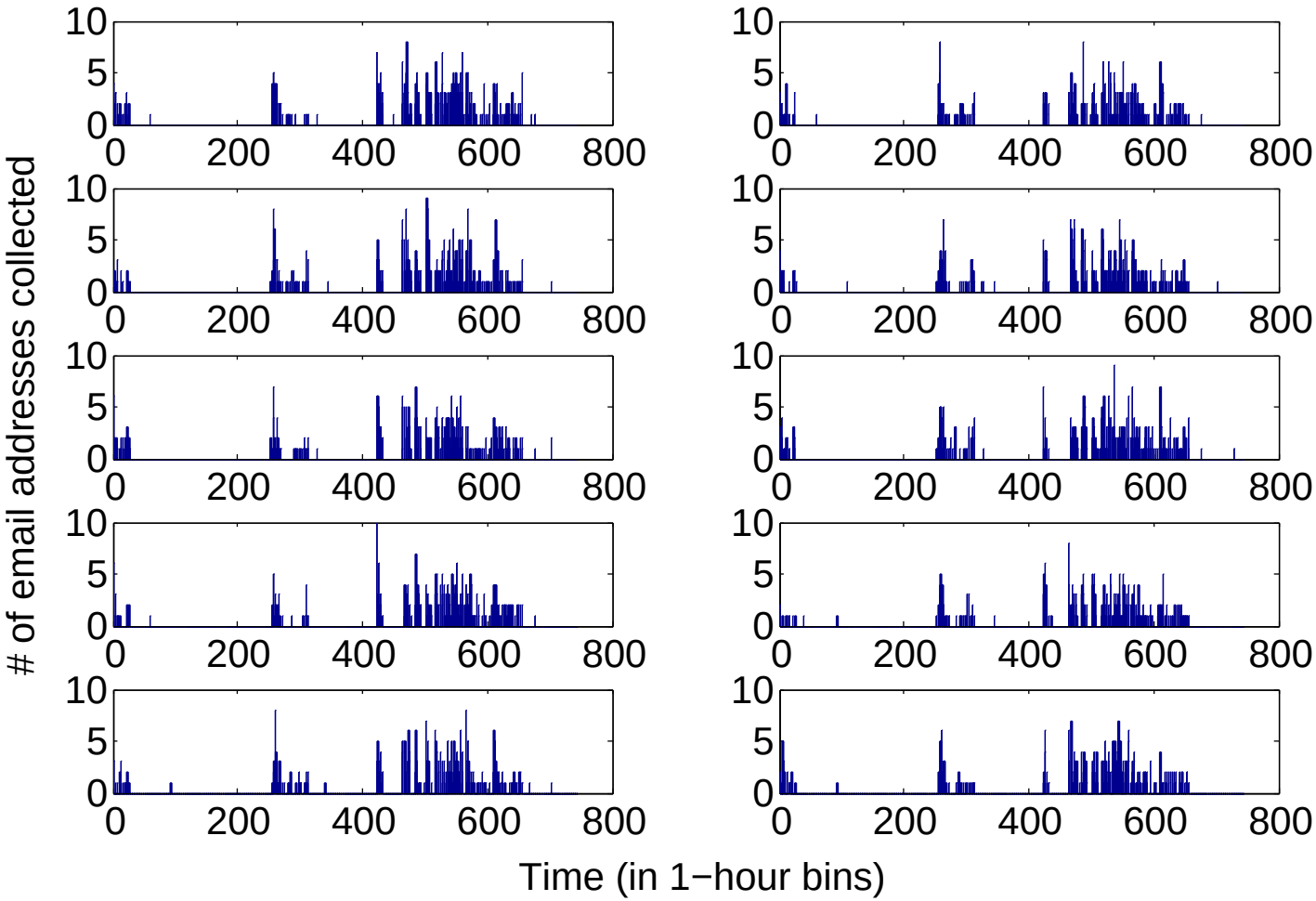
- IP addresses of all harvesters in this group have 208.66.195/24 prefix
- These harvesters are among the heaviest spammers in each month
- We discovered several other groups with coherent temporal behavior and similar IP addresses

Similarity in Temporal Harvesting



Results from July 2006 using similarity in temporal harvesting

Temporal Harvesting Histograms



Temporal spamming histograms of 208.66.195/24 group of harvesters

Statistics from Similarity in Temporal Harvesting

Average temporal harvesting correlation coefficients between two harvesters in the 208.66.195/24 group

Year	2006				
Month	May	June	July	August	September
ρ_{avg}	0.579	0.645	0.661	0.533	0.635

- Correlation is not as high as with temporal spamming
- Lower correlation is expected due to randomness of address acquisition times
- Results still indicate high behavioral correlation
- All harvesting was done between May and September 2006

Statistics from Similarity in Temporal Harvesting

Average temporal harvesting correlation coefficients between two harvesters in the 208.66.195/24 group

Year	2006				
Month	May	June	July	August	September
ρ_{avg}	0.579	0.645	0.661	0.533	0.635

- Correlation is not as high as with temporal spamming
- Lower correlation is expected due to randomness of address acquisition times
- Results still indicate high behavioral correlation
- All harvesting was done between May and September 2006

Statistics from Similarity in Temporal Harvesting

Average temporal harvesting correlation coefficients between two harvesters in the 208.66.195/24 group

Year	2006				
Month	May	June	July	August	September
ρ_{avg}	0.579	0.645	0.661	0.533	0.635

- Correlation is not as high as with temporal spamming
- Lower correlation is expected due to randomness of address acquisition times
- Results still indicate high behavioral correlation
- All harvesting was done between May and September 2006

Statistics from Similarity in Temporal Harvesting

Average temporal harvesting correlation coefficients between two harvesters in the 208.66.195/24 group

Year	2006				
Month	May	June	July	August	September
ρ_{avg}	0.579	0.645	0.661	0.533	0.635

- Correlation is not as high as with temporal spamming
- Lower correlation is expected due to randomness of address acquisition times
- Results still indicate high behavioral correlation
- All harvesting was done between May and September 2006

Statistics from Similarity in Temporal Harvesting

Average temporal harvesting correlation coefficients between two harvesters in the 208.66.195/24 group

Year	2006				
Month	May	June	July	August	September
ρ_{avg}	0.579	0.645	0.661	0.533	0.635

- Correlation is not as high as with temporal spamming
- Lower correlation is expected due to randomness of address acquisition times
- Results still indicate high behavioral correlation
- All harvesting was done between May and September 2006

Findings from Temporal Similarity

- We discover several groups with coherent temporal behavior and similar IP addresses
- In particular, a group of ten heavy spammers with 208.66.195/24 IP address prefix
  - Indicates that these computers are very close geographically
  - Either the same spammer or a group of spammers in same physical location
- Highly likely that these groups are coordinated

Findings from Temporal Similarity

- We discover several groups with coherent temporal behavior and similar IP addresses
- In particular, a group of ten heavy spammers with 208.66.195/24 IP address prefix
  - Indicates that these computers are very close geographically
  - Either the same spammer or a group of spammers in same physical location
- Highly likely that these groups are coordinated

Findings from Temporal Similarity

- We discover several groups with coherent temporal behavior and similar IP addresses
- In particular, a group of ten heavy spammers with 208.66.195/24 IP address prefix
  - Indicates that these computers are very close geographically
  - Either the same spammer or a group of spammers in same physical location
- Highly likely that these groups are coordinated

Findings from Temporal Similarity

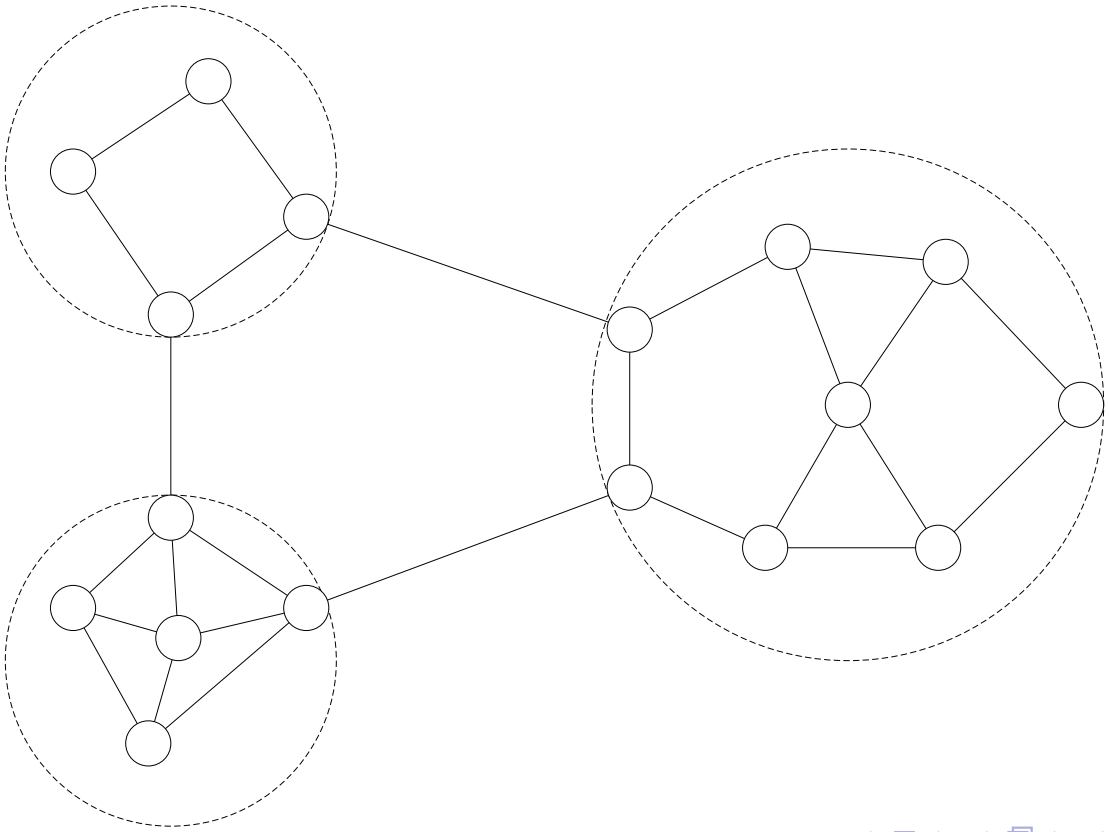
- We discover several groups with coherent temporal behavior and similar IP addresses
- In particular, a group of ten heavy spammers with 208.66.195/24 IP address prefix
  - Indicates that these computers are very close geographically
  - Either the same spammer or a group of spammers in same physical location
- Highly likely that these groups are coordinated

Findings from Temporal Similarity

- We discover several groups with coherent temporal behavior and similar IP addresses
- In particular, a group of ten heavy spammers with 208.66.195/24 IP address prefix
  - Indicates that these computers are very close geographically
  - Either the same spammer or a group of spammers in same physical location
- Highly likely that these groups are coordinated

Border Gateway Protocol

- Border Gateway Protocol (BGP) is the core routing protocol at the highest level in the Internet
- Routers on edge of autonomous systems (ASes) send updates between themselves about connectivity within their AS



BGP Life Span

- BGP life span of a spam server is roughly the amount of time it is connected to the rest of the Internet
- It has been observed that some spam servers have short BGP life spans, perhaps to remain untraceable (Ramachandran and Feamster, 2006)
- Are harvesters which use short-lived spam servers tightly connected?
  - Few spam servers have short BGP life spans
  - No significant correlation found between BGP life span and phishing level

BGP Life Span

- BGP life span of a spam server is roughly the amount of time it is connected to the rest of the Internet
- It has been observed that some spam servers have short BGP life spans, perhaps to remain untraceable (Ramachandran and Feamster, 2006)
- Are harvesters which use short-lived spam servers tightly connected?
  - Few spam servers have short BGP life spans
  - No significant correlation found between BGP life span and phishing level

BGP Life Span

- BGP life span of a spam server is roughly the amount of time it is connected to the rest of the Internet
- It has been observed that some spam servers have short BGP life spans, perhaps to remain untraceable (Ramachandran and Feamster, 2006)
- Are harvesters which use short-lived spam servers tightly connected?
  - Few spam servers have short BGP life spans
  - No significant correlation found between BGP life span and phishing level

BGP Life Span

- BGP life span of a spam server is roughly the amount of time it is connected to the rest of the Internet
- It has been observed that some spam servers have short BGP life spans, perhaps to remain untraceable (Ramachandran and Feamster, 2006)
- Are harvesters which use short-lived spam servers tightly connected?
  - Few spam servers have short BGP life spans
  - No significant correlation found between BGP life span and phishing level

BGP Life Span

- BGP life span of a spam server is roughly the amount of time it is connected to the rest of the Internet
- It has been observed that some spam servers have short BGP life spans, perhaps to remain untraceable (Ramachandran and Feamster, 2006)
- Are harvesters which use short-lived spam servers tightly connected?
  - Few spam servers have short BGP life spans
  - No significant correlation found between BGP life span and phishing level

Summary

- Clustering using similarity in spam server usage reveals communities of phishers and non-phishers
  - Phishing is a phenotype: most harvestors are either phishers or non-phishers
  - Phishers form gangs: they share resources in isolated closely knit communities.
- Clustering using temporal similarity reveals coordinated groups of harvesters
- Spammer social network patterns might be used for detection and interdiction
- Future work
  - Statistical latent variable models for spammer community discovery
  - Clustering based on combinations of similarity measures
  - Evolutionary models for community behavior
- The dual problem: discovery of spam server communities.

Summary

- Clustering using similarity in spam server usage reveals communities of phishers and non-phishers
  - Phishing is a phenotype: most harvestors are either phishers or non-phishers
  - Phishers form gangs: they share resources in isolated closely knit communities.
- Clustering using temporal similarity reveals coordinated groups of harvesters
- Spammer social network patterns might be used for detection and interdiction
- Future work
  - Statistical latent variable models for spammer community discovery
  - Clustering based on combinations of similarity measures
  - Evolutionary models for community behavior
- The dual problem: discovery of spam server communities.

Summary

- Clustering using similarity in spam server usage reveals communities of phishers and non-phishers
  - Phishing is a phenotype: most harvestors are either phishers or non-phishers
  - Phishers form gangs: they share resources in isolated closely knit communities.
- Clustering using temporal similarity reveals coordinated groups of harvesters
- Spammer social network patterns might be used for detection and interdiction
- Future work
  - Statistical latent variable models for spammer community discovery
  - Clustering based on combinations of similarity measures
  - Evolutionary models for community behavior
- The dual problem: discovery of spam server communities.

Summary

- Clustering using similarity in spam server usage reveals communities of phishers and non-phishers
  - Phishing is a phenotype: most harvestors are either phishers or non-phishers
  - Phishers form gangs: they share resources in isolated closely knit communities.
- Clustering using temporal similarity reveals coordinated groups of harvesters
- Spammer social network patterns might be used for detection and interdiction
- Future work
  - Statistical latent variable models for spammer community discovery
  - Clustering based on combinations of similarity measures
  - Evolutionary models for community behavior
- The dual problem: discovery of spam server communities.

Summary

- Clustering using similarity in spam server usage reveals communities of phishers and non-phishers
  - Phishing is a phenotype: most harvestors are either phishers or non-phishers
  - Phishers form gangs: they share resources in isolated closely knit communities.
- Clustering using temporal similarity reveals coordinated groups of harvesters
- Spammer social network patterns might be used for detection and interdiction
- Future work
  - Statistical latent variable models for spammer community discovery
  - Clustering based on combinations of similarity measures
  - Evolutionary models for community behavior
- The dual problem: discovery of spam server communities.

Summary

- Clustering using similarity in spam server usage reveals communities of phishers and non-phishers
  - Phishing is a phenotype: most harvestors are either phishers or non-phishers
  - Phishers form gangs: they share resources in isolated closely knit communities.
- Clustering using temporal similarity reveals coordinated groups of harvesters
- Spammer social network patterns might be used for detection and interdiction
- Future work
  - Statistical latent variable models for spammer community discovery
  - Clustering based on combinations of similarity measures
  - Evolutionary models for community behavior
- The dual problem: discovery of spam server communities.

Summary

- Clustering using similarity in spam server usage reveals communities of phishers and non-phishers
  - Phishing is a phenotype: most harvestors are either phishers or non-phishers
  - Phishers form gangs: they share resources in isolated closely knit communities.
- Clustering using temporal similarity reveals coordinated groups of harvesters
- Spammer social network patterns might be used for detection and interdiction
- Future work
  - Statistical latent variable models for spammer community discovery
  - Clustering based on combinations of similarity measures
  - Evolutionary models for community behavior
- The dual problem: discovery of spam server communities.

References

1. M. Girvan and M. E. J. Newman, “Community Structure in Social and Biological Networks,” *National Academy of Sciences* (2002).
2. L. Hubert and P. Arabie, “Comparing Partitions,” *Journal of Classification* (1985).
3. J. Moody, “Race, School Integration, and Friendship Segmentation in America,” *American Journal of Sociology* (2001).
4. M. Prince et al., “Understanding How Spammers Steal Your E-Mail Address: An Analysis of the First Six Months of Data from Project Honey Pot,” *2nd Conference on Email and Anti-Spam* (2005).
5. U. von Luxburg, “A Tutorial on Spectral Clustering,” *Statistics and Computing*, (2007).
6. S. Yu and J. Shi, “Multiclass Spectral Clustering,” *9th IEEE International Conference on Computer Vision* (2003).