Recommendation/Reputation

Ennan Zhai
ennan.zhai@yale.edu
Lecture Outline

• Background

• Reputation System: EigenTrust & Credence

• Sybil-Resistance: DSybil
Lecture Outline

• Background

• Reputation System: EigenTrust & Credence

• Sybil-Resistance: DSybil
Motivating Example
Motivating Example
Motivating Example
Motivating Example

Search Result

- Green circles
- Red circles with horns
- Yellow arrows
- Computer
- User

Diagram showing relationships and search results.
Motivating Example

It is highly possible for the consumer to select a bad object.
Motivating Example

But reputation systems can help you!
Reputation/Recommendation

- What is the reputation system?
Reputation/Recommendation

• What is the reputation system?

• What is the recommendation system?
Reputation/Recommendation

- What is the reputation system?
- What is the recommendation system?
- Difference?
Categories of Reputation Systems

Three different types:

• Peer-based reputation systems, e.g., EigenTrust;
Categories of Reputation Systems

Three different types:

• Peer-based reputation systems, e.g., EigenTrust;

• Object-based reputation systems, e.g., Credence;
Categories of Reputation Systems

Three different types:

• Peer-based reputation systems, e.g., EigenTrust;

• Object-based reputation systems, e.g., Credence;

• Hybrid reputation systems, e.g., Scrubber.
Lecture Outline

• Background

• Reputation System: EigenTrust & Credence

• Sybil-Resistance: DSysbil
Lecture Outline

• Background

• Reputation System: EigenTrust & Credence

• Sybil-Resistance: DSybil
EigenTrust

The first peer-based reputation system:

• Similar to PageRank;

• Not a pure decentralized system.
Credence
Decentralized file-sharing system

Request

Alice
<table>
<thead>
<tr>
<th>Name</th>
<th>Sources</th>
<th>Voters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{10}$</td>
<td>$P_2\ P_6$</td>
<td>$P_2\ P_4\ P_6$</td>
</tr>
<tr>
<td>$F_{22}$</td>
<td>$P_2\ P_6\ P_8$</td>
<td>$P_2\ P_7$</td>
</tr>
<tr>
<td>$F_4$</td>
<td>$P_2\ P_4$</td>
<td>$P_2\ P_4\ P_7$</td>
</tr>
<tr>
<td>$F_6$</td>
<td>$P_{11}\ P_{13}\ P_{14}$</td>
<td>$P_{11}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Computing each File’s Reputation

<table>
<thead>
<tr>
<th>Name</th>
<th>Sources</th>
<th>Voters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{10}$</td>
<td>$P_2$ $P_6$</td>
<td>$P_2$ $P_4$ $P_6$</td>
</tr>
<tr>
<td>$F_{22}$</td>
<td>$P_2$ $P_6$ $P_8$</td>
<td>$P_2$ $P_7$</td>
</tr>
<tr>
<td>$F_4$</td>
<td>$P_2$ $P_4$</td>
<td>$P_2$ $P_4$ $P_7$</td>
</tr>
<tr>
<td>$F_6$</td>
<td>$P_{11}$ $P_{13}$ $P_{14}$</td>
<td>$P_{11}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
A File’s Reputation

Credence uses weighted averaging to compute the reputation of a file.
A File’s Reputation

\[
\text{Rep}(F) = \frac{\sum_{i=1}^{n} V_i \cdot \theta(\text{Alice}, \text{Voter}_i)}{\sum_{i=1}^{n} |\theta(\text{Alice}, \text{Voter}_i)|} \quad [\text{-1, 1}]
\]

Credence uses weighted averaging to compute the reputation of a file.
A File’s Reputation

\[ \text{Rep}(F) = \frac{\sum_{i=1}^{n} V_i \cdot \theta(Alice, \text{Voter}_i)}{\sum_{i=1}^{n} \left| \theta(Alice, \text{Voter}_i) \right|} \leq 1 \]

The vote cast by voter \( i \) on F (+1 or -1)
A File’s Reputation

\[
\text{Rep}(F) = \frac{\sum_{i=1}^{n} V_i \cdot \theta(Alice, \text{Voter}_i)}{\left| \sum_{i=1}^{n} \theta(Alice, \text{Voter}_i) \right|} \in [-1, 1]
\]

The vote cast by voter\(_i\) on F (+1 or -1)

The similarity between Alice and voter\(_i\)

The range is [-1, +1].
Similarity between Nodes
Similarity between Nodes

\[ \theta = \frac{(p-ab)}{\sqrt{a(1-a)b(1-b)}} \]
Similarity between Nodes

\[ \theta = \frac{(p-ab)}{\sqrt{a(1-a)b(1-b)}} \]

For the overlapping voting set (e.g., S) between Alice and P_i:
Simularity between Nodes

\[ \theta = \frac{p-ab}{\sqrt{a(1-a)b(1-b)}} \]

For the overlapping voting set (e.g., S) between Alice and P_i :

\[
a = \frac{\text{# of positive votes cast by Alice on the files in S}}{\text{# of all the votes cast by Alice on the files in S}}
\]
$\theta = \frac{p-ab}{\sqrt{a(1-a)b(1-b)}}$

For the overlapping voting set (e.g., S) between Alice and $P_i$:

$a = \frac{\text{# of positive votes cast by Alice on the files in S}}{\text{# of all the votes cast by Alice on the files in S}}$

$b = \frac{\text{# of positive votes cast by } C_i \text{ on the files in S}}{\text{# of all the votes cast by } C_i \text{ on the files in S}}$
\[
\theta = \frac{(p-ab)}{\sqrt{a \cdot (1-a) \cdot b \cdot (1-b)}}
\]

For the overlapping voting set (e.g., S) between Alice and \( P_i \):

- \( a = \frac{\text{# of positive votes cast by Alice on the files in S}}{\text{# of all the votes cast by Alice on the files in S}} \)
- \( b = \frac{\text{# of positive votes cast by } C_i \text{ on the files in S}}{\text{# of all the votes cast by } C_i \text{ on the files in S}} \)
- \( p = \frac{\text{# of positive votes cast by both Alice and } C_i \text{ on the files in S}}{\text{# of all the votes agreed by both Alice and } C_i \text{ on the files in S}} \)
Example

\[ \theta = \frac{p-\alpha \beta}{\sqrt{\alpha (1-\alpha) \beta (1-\beta)}} \]
Practical Issues

There are a few practical issues in Credence:

• Cold start;

• Overlapping voting history.
Flow-based Reputation:

\[ \theta_{ac} = \theta_{ab} \times \theta_{bc} \]
Possible Solution

Flow-based Reputation:

\[ \theta_{ac} = \theta_{ab} \times \theta_{bc} \]
Recall Reputation Computation

\[
\text{Rep}(F) = \frac{\sum_{i=1}^{n} V_i \cdot \theta(\text{Alice}, \text{Voter}_i)}{\left| \sum_{i=1}^{n} \theta(\text{Alice}, \text{Voter}_i) \right|} [ -1, 1 ]
\]

Why it works?
<table>
<thead>
<tr>
<th>Name</th>
<th>Sources</th>
<th>Voters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{10}$</td>
<td>$P_2 \ P_6$</td>
<td>$P_2 \ P_4 \ P_6$</td>
</tr>
<tr>
<td>$F_{22}$</td>
<td>$P_2 \ P_6 \ P_8$</td>
<td>$P_2 \ P_7$</td>
</tr>
<tr>
<td>$F_4$</td>
<td>$P_2 \ P_4$</td>
<td>$P_2 \ P_4 \ P_7$</td>
</tr>
<tr>
<td>$F_6$</td>
<td>$P_{11} \ P_{13} \ P_{14}$</td>
<td>$P_{11}$</td>
</tr>
</tbody>
</table>

... ... ... ...
## Reputation

<table>
<thead>
<tr>
<th>Name</th>
<th>Sources</th>
<th>Voters</th>
</tr>
</thead>
<tbody>
<tr>
<td>F_{10} = 0.8</td>
<td>P_2 P_6</td>
<td>P_2 P_4 P_6</td>
</tr>
<tr>
<td>F_{22}</td>
<td>P_2 P_6 P_8</td>
<td>P_2 P_7</td>
</tr>
<tr>
<td>F_4</td>
<td>P_2 P_4</td>
<td>P_2 P_4 P_7</td>
</tr>
<tr>
<td>F_6</td>
<td>P_{11} P_{13} P_{14}</td>
<td>P_{11}</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
# Reputation

<table>
<thead>
<tr>
<th>Name</th>
<th>Sources</th>
<th>Voters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{10} = 0.8$</td>
<td>$P_2 \ P_6$</td>
<td>$P_2 \ P_4 \ P_6$</td>
</tr>
<tr>
<td>$F_{22} = 0.5$</td>
<td>$P_2 \ P_6 \ P_8$</td>
<td>$P_2 \ P_7$</td>
</tr>
<tr>
<td>$F_4$</td>
<td>$P_2 \ P_4$</td>
<td>$P_2 \ P_4 \ P_7$</td>
</tr>
<tr>
<td>$F_6$</td>
<td>$P_{11} \ P_{13} \ P_{14}$</td>
<td>$P_{11}$</td>
</tr>
</tbody>
</table>

Alice
**Reputation**

<table>
<thead>
<tr>
<th>Name</th>
<th>Sources</th>
<th>Voters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{10} = 0.8$</td>
<td>$P_2 \ P_6$</td>
<td>$P_2 \ P_4 \ P_6$</td>
</tr>
<tr>
<td>$F_{22} = 0.5$</td>
<td>$P_2 \ P_6 \ P_8$</td>
<td>$P_2 \ P_7$</td>
</tr>
<tr>
<td>$F_4 = 0.9$</td>
<td>$P_2 \ P_4$</td>
<td>$P_2 \ P_4 \ P_7$</td>
</tr>
<tr>
<td>$F_6$</td>
<td>$P_{11} \ P_{13} \ P_{14}$</td>
<td>$P_{11}$</td>
</tr>
<tr>
<td>Name</td>
<td>Sources</td>
<td>Voters</td>
</tr>
<tr>
<td>-------</td>
<td>---------</td>
<td>--------</td>
</tr>
<tr>
<td>$F_{10} = 0.8$</td>
<td>$P_2 \ P_6$</td>
<td>$P_2 \ P_4 \ P_6$</td>
</tr>
<tr>
<td>$F_{22} = 0.5$</td>
<td>$P_2 \ P_6 \ P_8$</td>
<td>$P_2 \ P_7$</td>
</tr>
<tr>
<td>$F_4 = 0.9$</td>
<td>$P_2 \ P_4$</td>
<td>$P_2 \ P_4 \ P_7$</td>
</tr>
<tr>
<td>$F_6 = 0.6$</td>
<td>$P_{11} \ P_{13} \ P_{14}$</td>
<td>$P_{11}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Alice**
### Ranking

<table>
<thead>
<tr>
<th>Name</th>
<th>Sources</th>
<th>Voters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_4 = 0.9$</td>
<td>$P_2 \ P_4$</td>
<td>$P_2 \ P_4 \ P_7$</td>
</tr>
<tr>
<td>$F_{10} = 0.8$</td>
<td>$P_2 \ P_6$</td>
<td>$P_2 \ P_4 \ P_6$</td>
</tr>
<tr>
<td>$F_6 = 0.6$</td>
<td>$P_{11} \ P_{13} \ P_{14}$</td>
<td>$P_{11}$</td>
</tr>
<tr>
<td>$F_{22} = 0.5$</td>
<td>$P_2 \ P_6 \ P_8$</td>
<td>$P_2 \ P_7$</td>
</tr>
</tbody>
</table>

... ... 

... ... 

... ...
Discussions

Credence is a practical system but has many problems:

• Cold start;
• Overlapping voting history;
• Sybil attack.
Sybil Attack
Sybil Attack

Sybil attack is the killer of reputation systems:

• What is sybil attack?
Sybil Attack

Sybil attack is the killer of reputation systems:

• What is sybil attack?

• How does sybil attack compromise Credence?
Sybil Attack

Sybil attack is the killer of reputation systems:

• What is sybil attack?

• How does sybil attack compromise Credence?

• Can we defend against sybil attack?
Lecture Outline

• Background

• Reputation System: EigenTrust & Credence

• Sybil-Resitance: DSybil
Lecture Outline

• Background

• Reputation System: EigenTrust & Credence

• Sybil-Resistance: DSybil
Sybil Defense

• A very hot but very hard topic
• Many good efforts
DSybil

- Based on feedback and reputation
- Loss (# of bad recommendations) is provable $O(D \log M)$ even under worst-case attack
  
  D: Dimension of the objects (less than 10 in Digg)
  M: Max # of sybil identities voting on each obj
- The authors prove Dsybil’s loss is optimal
Challenges

It is very challenging to defend against sybil:

1. How to identify “correct” but “non-helpful” votes?
Challenges

It is very challenging to defend against sybil:

1. How to identify “correct” but “non-helpful” votes?
2. How to assign initial trust to new identities?
Challenges

It is very challenging to defend against sybil:

1. How to identify “correct” but “non-helpful” votes?
2. How to assign initial trust to new identities?
3. How exactly to grow trust?
Challenges

It is very challenging to defend against sybil:

1. How to identify “correct” but “non-helpful” votes?
2. How to assign initial trust to new identities?
3. How exactly to grow trust?
4. How to rank the results?
Key #1: Heavy-Tail Distribution

Leveraging typical voting behavior of honest users:
• Exist very active users who cast many votes
Key #1: Heavy-Tail Distribution

Leveraging typical voting behavior of honest users:

• Exist very active users who cast many votes
Key #1: Heavy-Tail Distribution

Leveraging typical voting behavior of honest users:

- Exist very active users who cast many votes
Key #2: If user is already getting “enough help”, then do not give out more trust:

This insight enables us to avoid giving trust to some sybil identities;

The insight can make us strike an optimal balance.
System Model

- Objects to be recommended are either good or bad;
System Model

• Objects to be recommended are either good or bad;

• Votes are positive. Namely, DSybil only has positive votes.
System Model

• Objects to be recommended are either good or bad;

• Votes are positive. Namely, DSybil only has positive votes.

• DSybil is personalized:
After search, we enter one round

2 good objs 2 bad objs

DSybil does not know which are good
After search, we enter one round

DSybil does not know which are good

Each round has a pool of objects:

• DSybil recommends one object for Alice to consume;
After search, we enter one round

2 good objs  2 bad objs

DSybil does not know which are good

Each round has a pool of objects:

• DSybil recommends one object for Alice to consume;
• Alice provides feedback after consumption;
After search, we enter one round

DSybil does not know which are good

Each round has a pool of objects:
• DSybil recommends one object for Alice to consume;
• Alice provides feedback after consumption;
• DSybil adjusts reputations based on the feedback.
Each identity starts with initial trust 0.2

An object is overwhelming if total trust $\geq C$ ($C = 1.0$)
Initial Round

1. Recommend uniformly random object
Initial Round

1. Recommend uniformly random object
2. Adjust reputation after feedback
   - if obj is bad, multiply trust of voters by $\beta = 0.5$
   - if obj is good, multiply trust of voters by $\alpha = 2$
Rounds with Overwhelming Obj

E: 1.0
H: 0.2
Total: 1.2

G: 0.2
H: 0.2
Total: 0.4

F: 1.0
Total: 1.0
Rounds with Overwhelming Obj

1. Recommend arbitrary overwhelming object
Rounds with Overwhelming Obj

1. Recommend arbitrary overwhelming object
I. Recommend arbitrary overwhelming object
   - Will confiscate sufficient trust if obj is bad
Rounds with Overwhelming Obj

1. Recommend arbitrary overwhelming object
   - Will confiscate sufficient trust if obj is bad
2. Adjust trust after feedback
   - if obj is bad, multiply trust of voters by $\beta = 0.5$
   - if obj is good, no additional trust given out (#2 key)
Why it works?
Guide: A set of honest users in the system are very active. Usually, the votes casted by guides can cover 60% objects.
Guide: A set of honest users in the system are very active. Usually, the votes casted by guides can cover 60% objects.
The minimal guide has 2 elements:
{X,Y} or {X,W} or {X,A}
Guide

The minimal guide has 2 elements:
\{X,Y\} or \{X,W\} or \{X,A\}

Note that DSybil does not know who are the guide
Minimal guide = \{X\}
<table>
<thead>
<tr>
<th>X</th>
<th>X</th>
<th>A</th>
<th>Y</th>
<th>W</th>
<th>A</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>B</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>A</td>
<td>A</td>
<td>B</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
</tr>
</tbody>
</table>

Guide
Guide

{X,Y} or {X,W}
Guide

<table>
<thead>
<tr>
<th>X</th>
<th>X</th>
<th>Y</th>
<th>A</th>
<th>A</th>
<th>W</th>
<th>A</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Guide = \{A\}
Leveraging Key Insight #1

The elements in the minimal guide is typically small in practice.
Leveraging Key Insight #1

The elements in the minimal guide is typically small in practice

Small number of elements -> will encounter guides frequently when picking random objs:

* Reputation to guides quickly grow to C
Leveraging Key Insight #1

The elements in the minimal guide is typically small in practice.

Small number of elements -> will encounter guides frequently when picking random objs:
  * Reputation to guides quickly grow to C
  * This will result in overwhelming objs
Leveraging Key Insight #2

Consuming good overwhelming obj = Alice already has sufficient help
Leveraging Key Insight #2

Consuming good overwhelming obj = Alice already has sufficient help

Thus do not give out additional reputation:
* Prevent sybil identities from getting reputation “for free”
* May hurt honest identities (but remember this is optimal)
Proof for $O(D \log M)$ loss even under worst-case attack.
Sybil Attack Summary

- Again, sybil-defense is very interesting but very hard
- In the future lecture, we will learn social network based solution against sybil attack.