LinkMirage: Enabling Privacy-preserving Analytics on Social Relationships

Changchang Liu, Prateek Mittal
Email: cl12@princeton.edu, pmittal@princeton.edu
Princeton University

February 23, 2016
Third party applications rely on users’ social relationships:

- E-commerce
- Spam detection
- Anonymous communication
- Sybil defenses
Social relationships are very sensitive!

Social relationships represent

- Trusted friendships
- Important interactions
- Even more, business relations, etc.
How to balance utility and privacy?

Protect privacy of sensitive social relationships
Preserve utility of obfuscated social relationships for real-world applications
Previous work of link privacy mechanisms

To protect link privacy, previous work
• obfuscate social relationships through link additions/deletions

\[ G \xrightarrow{p_{\text{add}} \quad p_{\text{del}}} G' \]
Limitations of previous link privacy mechanisms

To protect link privacy, previous work

- obfuscate social relationships through link additions/deletions

However, previous work

- only consider graph data where the links are static
However, social networks are dynamic

Temporal Facebook dataset (every three months) with 46,952 users and 876,993 edges
However, social networks are dynamic

An adversary can combine the previously perturbed graphs together
Our Objective

- Balance privacy and utility
- Handle both the static and dynamic social network topologies
- Provide rigorous privacy guarantees
- Useful in real-world applications
LinkMirage

LinkMirage Overview
Algorithm Description
Privacy Analysis
Utility Analysis
Social Relationship based Applications

Original Graph $G$

Untrusted Applications

Privacy-preserving graph analysis
Sybil defenses
Anonymous communication

OSN providers
Privacy-preserving Social Relationship based Applications

Original Graph $G$

Untrusted Applications

Privacy-preserving graph analysis
Sybil defenses
Anonymous communication

OSN providers
LinkMirage Architecture

LinkMirage System (Trusted)

Original Graph $G$

Obfuscation Algorithm

Obfuscated Graph $G'$

LinkMirage social link app

OSN providers

User1  User2  User3  ...

Untrusted Applications

Privacy-preserving graph analysis
Sybil defenses
Anonymous communication

$Q$

$Q(G')$
LinkMirage

LinkMirage Overview
Algorithm Description
Privacy Analysis
Utility Analysis
Key intuitions

- Naive method: independent perturbation
  - more information is leaked to others
- We need to
  - incorporate graph evolution
  - leverage the information already released in previous graphs
Algorithm Description

\[ G_{t-1} \]
Algorithm Description

1. clustering

$G_{t-1}$

$C_1$

$C_2$
Algorithm Description

1. clustering

2. perturbation
Algorithm Description

1. Clustering
   - $G_{t-1}$

2. Perturbation
   - $G'_{t-1}$

3. Evolution
   - $G_t$
Algorithm Description

1. Clustering

2. Perturbation

3. Evolution

4. Dynamic Clustering
Algorithm Description

1. clustering
2. perturbation
3. evolution
4. dynamic clustering
5. selective perturbation
Algorithm Description

Key step 1: dynamic clustering

Key step 2: selective perturbation

1. clustering

2. perturbation

3. evolution

$G_{t-1}$

$G_t$

$G'_{t-1}$

$G'_t$
Two Key Steps in Our Algorithm

Two key steps

- **Dynamic Clustering**
  - find communities by simultaneously considering consecutive graphs
  - backtrack based on clustering result of the previous graph

- **Selective Perturbation**
  - perturb the minimal amount of edges
  - use a very high privacy parameter while preserving structural properties (utility)
Facebook Temporal Dataset (46,952 users and 876,993 edges)
Utility Advantage

<table>
<thead>
<tr>
<th>t=3</th>
<th>t=4</th>
<th>t=5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original graphs</td>
<td>Mittal et al.</td>
<td>LinkMirage</td>
</tr>
</tbody>
</table>

Superior utility, due to dynamic clustering.
Utility advantage even exists in static scenario.
Utility Advantage

Superior utility, due to dynamic clustering
Utility advantage even exists in static scenario
Privacy Advantage

Original graphs

Overlapped edges (black) and Changed edges (yellow) between consecutive graphs
Privacy Advantage

Overlapped edges (black) and Changed edges (yellow) between consecutive graphs
Privacy Advantage

Overlapped edges (black) and Changed edges (yellow) between consecutive graphs

Superior privacy, due to selective perturbation
Anti-Inference Privacy

Assume the worst-case adversary knows

- the obfuscated graphs $\{ G'_i \}_{i=0}^t$
- all the other links except for one link $L_t$
- our obfuscation algorithm

The adversary computes the posterior probability

$$P(L_t | \{ G'_i \}_{i=0}^t, W) = \frac{P(\{ G'_i \}_{i=0}^t | L_t, W) \times P(L_t | W)}{P(\{ G'_i \}_{i=0}^t | W)} \quad (1)$$

and compare with the prior probability
Anti-Inference Privacy

Assume the worst-case adversary knows

- the obfuscated graphs \( \{ G'_i \}_{i=0}^t \)
- all the other links except for one link \( L_t \)
- our obfuscation algorithm

The adversary computes the posterior probability

\[
P(L_t | \{ G'_i \}_{i=0}^t, W) = \frac{P(\{ G'_i \}_{i=0}^t | L_t, W) \times P(L_t | W)}{P(\{ G'_i \}_{i=0}^t | W)} \tag{2}
\]

and compare with the prior probability

Higher similarity implies better anti-inference privacy
Anti-Inference Privacy

LinkMirage has higher anti-inference privacy!
Anti-Inference Privacy

LinkMirage achieves higher anti-inference privacy!
LinkMirage

LinkMirage Overview
Algorithm Description
Privacy Analysis
Utility Analysis
Privacy-preserving Graph Analytics

<table>
<thead>
<tr>
<th></th>
<th>Facebook</th>
<th>Original Graph</th>
<th>LinkMirage</th>
<th>Mittal et al.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modularity</td>
<td>0.488</td>
<td>0.487</td>
<td>0.415</td>
<td></td>
</tr>
</tbody>
</table>
Privacy-preserving Graph Analytics

<table>
<thead>
<tr>
<th>Facebook</th>
<th>Original Graph</th>
<th>LinkMirage</th>
<th>Mittal et al.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modularity</td>
<td>0.488</td>
<td>0.487</td>
<td>0.415</td>
</tr>
</tbody>
</table>

LinkMirage preserves graph analytics better!

Other graph analytics: pagerank, etc.

More applications:
- Sybil defenses
- Anonymous communication
Conclusion

Our LinkMirage system

- Both static and temporal graphs
- Provide rigorous privacy advantages
- Show utility advantages theoretically and using real-world applications
- Generalizable to communication networks and web graphs
Definition

The indistinguishability for a link \( L_t \) that the adversary can infer from the perturbed graph \( G'_t \) under the adversary’s prior information \( \{ \tilde{G}_i(L_t) \}_{i=0}^t \) is defined as

\[
\text{Privacy}_{id} = H(L_t|\{ G'_i \}_{i=0}^t, \{ \tilde{G}_i(L_t) \}_{i=0}^t)
\]  

(3)
Appendix 1: Indistinguishability

![Graph showing indistinguishability over time for different k values and Mittal et al. LinkMirage and Hay’s et al.](image-url)
Definition

The anti-aggregation privacy for a perturbed graph $G'_t$ with respect to the original graph $G_t$ and the perturbation parameter $k$ is

$$\text{Privacy}_{aa}(G_t, G'_t, k) = \| P^k_t - P'_t \|_{TV}$$

(4)
Appendix 2: Anti-aggregation Privacy

(b) Timestamp $t$

Anti-aggregation Privacy

$K=2$, Mittal et al.
$K=2$, LinkMirage

Anti-aggregation Privacy

(b) Timestamp $t$