Fake Co-visititation Injection Attacks to Recommender Systems

Guolei Yang, Neil Zhenqiang Gong, Ying Cai
Co-visitation Recommender System is Popular

We show co-visitation recommender systems can be spoofed to recommend items as an attacker desires.
Brief Intro to Co-visititation Recommender System

- Key idea: *Items that are frequently visited together in the past are likely to be visited together in the future*
Key Data Structure: Co-visititation Graph

Each vertex represents an item
Key Data Structure: Co-visitation Graph

Number of views of Item 1 (Popularity)

Item 1 -> Item 2: 8
Item 1 -> Item 3: 4
Item 1 -> Item 4: 6
Item 2 -> Item 3: 12
Item 2 -> Item 4: 5
Item 3 -> Item 4: 9
Item 3 -> Item 5: 7
Item 4 -> Item 5: 9

Item 1

22

Item 3

11

Item 4

35

Item 2

27

Item 5

13
Key Data Structure: Co-visititation Graph

Number of co-visitations between Item 1 and 2

Item 1 — Item 2

Item 1

Item 2

Item 3

Item 4

Item 5
Two Recommendation Tasks

**item-to-item recommendation**

1. Co-visitation graph
2. Other information

**user-to-item recommendation**

1. Profile of user $u$
2. Co-visitation graph
3. Other information

Recommendation engine -> Top-$N$ recommended items
Item-to-Item Recommendation

Compute item-item similarity

\[ s_{ij} = \frac{w_{ij}}{f(w_i, w_j)} \]

e.g., On YouTube

\[ f(w_i, w_j) = w_i \times w_j \]

Rank items by similarity

For **Item 1**:
1. **Item 2**
2. **Item 4**
3. **Item 3**

Generate recommendation list

Include items
1) with high similarity
2) satisfy **popularity threshold**

For **Item 1**:

View

Recommend items

**Item 2**
Related Work

- Xing et al. (USENIX Security’13) proposed *pollution attacks* to the user-to-item recommendation
  - *Relies on Cross-Site Request Forgery (CSRF)*
  - *Not applicable to item-to-item recommendation*

- **Profile injection (Shilling) attacks** to recommender systems via user-item rating matrices
  - Not applicable to co-visitation recommender systems which do not rely on user-item rating matrix.

- Relationship to adversarial machine learning
  - Our attack is data poisoning attack to recommender systems
Roadmap

- Threat model
- Proposed attacks
- Evaluations on synthetic data
- Evaluations on real-world recommender systems
- Countermeasures
Threat Model

- Attacker’s background knowledge
  - High knowledge
    - Co-visitation Graph
    - Popularity Threshold
  - Medium knowledge
    - Recommendation Lists
    - Item Popularity
  - Low knowledge
    - Recommendation Lists

- Scenario
  - Insider
  - YouTube ...
  - Amazon, eBay...

- Attacker’s goal
  - User Impression (UI): The probability that a random visitor will see the item
  - Increase UI of a target item
  - Decrease UI of a target item
Proposed Attacks

- Promotion attack
  - Goal: Increase UI of a Target Item
  - Make the target Item appear in the recommendation lists of as many items as possible
Proposed Attacks

- Promotion attack
  - Goal: Increase UI of a **Target Item**
  - Make the target Item appear in the recommendation lists of as many items as possible
Proposed Attacks

- Promotion attack
  - Goal: Increase UI of a Target Item
  - Make the target Item appear in the recommendation lists of as many items as possible
Proposed Attacks

- Demotion attack
  - Goal: Decrease UI of a **Target Item**
  - Remove the target Item from the recommendation lists of as many items as possible

![Diagram showing demotion attack with anchor items and recommendation items](attachment:diagram.png)
Key Challenge

- Given a target item and a limited number fake co-visitations
  - *How to select the anchor item(s) to attack?*
  - *How many fake co-visitations to insert for each anchor item?*
Key Challenge

- Given a target item
  - *How to select the anchor item(s) to attack?*
  - *How many fake co-visitations to insert for each anchor item?*

- Solution: Formulate the attack as an optimization problem
  - *Select the best anchor items to attack*
  - *Determine how many fake co-visitation is needed to attack each anchor*
Promotion Attack – High Knowledge Attacker

Attacker’s Goal: Promote *Item 3*

Select anchor items

Original Co-visitation graph

Recommend items

*Item 2*
Promotion Attack – High Knowledge Attacker

Attacker’s Goal: Promote *Item 3*

*Insert 10 fake co-visitations of Item 1 & 3*

**Attacked Co-visititation graph**
Promotion Attack – High Knowledge Attacker

Attacker’s Goal: Promote Item 3

Insert 10 fake co-visitations of Item 3 & 4

Attacked Co-visitation graph
Promotion Attack – High Knowledge Attacker

Attacker’s Goal: Promote *Item 3*

**Attacked Co-visititation graph**

- **Item 1**: 32
  - 14
  - 6
- **Item 2**: 27
  - 8
  - 12
  - 5
- **Item 3**: 31
  - 17
- **Item 4**: 45
  - 9
- **Item 5**: 13

**Recommend items**
- *Item 3*
Promotion Attack – High Knowledge Attacker

Attacker’s Goal: Promote *Item 3*

- Medium knowledge attacker can be converted into high knowledge attacker by *estimating edge weight*
- Low knowledge attacker can be converted into medium knowledge attacker by *estimating vertex weight*

*Attacked Co-visitation graph*
Attacker’s Goal: Demote Item 4

Original Co-visititation graph

Recommend items
Item 4

Recommend items
Item 4

Recommend items
Item 4
Demotion Attack – High Knowledge Attacker

Attacker’s Goal: Demote Item 4

Attacked Co-visititation graph

Recommend items

Item 1

Item 2

Item 5

Item 3

Item 4

Item 2

Recommend items

Item 5

Item 1
Evaluation on Synthetic Data

- Question we aim to answer
  - How does attacker’s background knowledge impact our attacks
  - How does the co-visititation graph structure impact our attacks?
  - How does the number of inserted fake co-visititations impact our attacks?
Impact of Attacker’s Background Knowledge

![Bar chart showing the impact of promotion and demotion attacks with different levels of background knowledge.](chart.png)
Impact of Co-visititation Graph Structure

![Graph showing the impact of promotion and demotion attacks on different graph structures. The x-axis represents different graph structures: Regular Co-visititation, ER, PowerLaw. The y-axis represents the average IUI / DUI (%).]
Impact of Number of Fake Co-visitations

Average IUI (%) vs. Number of injected co-visitations

- High knowledge
- Medium knowledge
- Low knowledge
Evaluation on Real-World Recommender Systems

- Initialization
- Select anchor items
- Insert fake co-visitations
- Exam results

Repeated for approx. 21 days

48 ~ 72 hours
Results on YouTube

Number of anchors

Day

Anchors selected
Successful attacks
Results on YouTube

Promotion attacks
Demotion attacks

Sum of popularity

Day

8x10^5
6x10^5
4x10^5
2x10^5
0x10^0
3 6 9 12 15 18 21
Countermeasures

- Limiting background knowledge
  - The website can discretize item popularities

- Funny Video
  - 3827 Views
  - Shows exact popularity

- Funny Video
  - 3500+ Views
  - Discretize Granularity = 500

- Funny Video
  - 2000+ Views
  - Discretize Granularity = 2000
Countermeasures

- Limiting background knowledge
  - The website can *discretize item popularities*
Conclusion

- Recommender systems are vulnerable to *Fake Co-visitation Injection Attacks*.

- An attacker can use our attacks to spoof a recommender system to make recommendations as the attacker desires.
Parameter Estimation

- Convert medium/low knowledge attackers into high knowledge attacker
- The missing knowledge is estimated based on publically available information

*Insert a fake item as probe*

*Insert co-visitations until it appears in the recommendation list of an item*
Convert *medium/low knowledge attackers* into *high knowledge attacker*

- The missing knowledge is estimated based on publically available information

*Parameter Estimation*

- Insert a fake item as probe
- Insert co-visitations until it appears in the recommendation list of an item
Proposed Attack Algorithm

- General steps

**Initialization**
- Knowledge acquire
- Parameters estimation

**Select items to attack**
- Construct & solve the optimization problem

**Insert fake co-visitations**
- Repeatedly view selected items in the same browser session

**Exam results**

- Goal achieved?
  - Yes: Terminate
  - No: Iterate

---

**Goal achieved?**

- Yes: Terminate
- No: Iterate
Experiments on Real-world Recommender Systems

- Results on *YouTube*

(a) Promotion attacks

(b) Demotion attacks

(c) Popularity of successfully attacked anchors

(d) Cost vs. anchor popularity
Experiments on Real-world Recommender Systems

- Results on eBay

(a) Promotion attack

(b) Demotion attacks

(c) Purchases of successfully attacked anchors
Experiments on Real-world Recommender Systems

- Results on Amazon

- Results on Yelp
Experiments on Real-world Recommender Systems

- Results on LinkedIn

![Graphs showing promotion and demotion attacks over time](image)
Countermeasures

- Limiting fake co-visitations
  - Use CAPTCHA

- Fake co-visititation detection

- Using co-visitations from registered users only