Data Science and Engineering Lab

Adversarial Attacks for Recommendations

Wenqi Fan

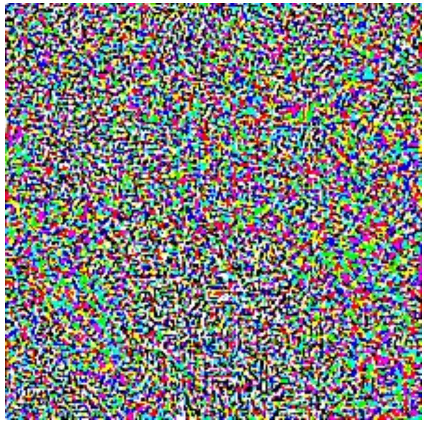
The Hong Kong Polytechnic University

https://wenqifan03.github.io, wenqifan@polyu.edu.hk

Tutorial website: https://advanced-recommender-systems.github.io/ijcai2021-tutorial/

Adversarial Attacks on Deep Learning

+



=

Classified as panda

x

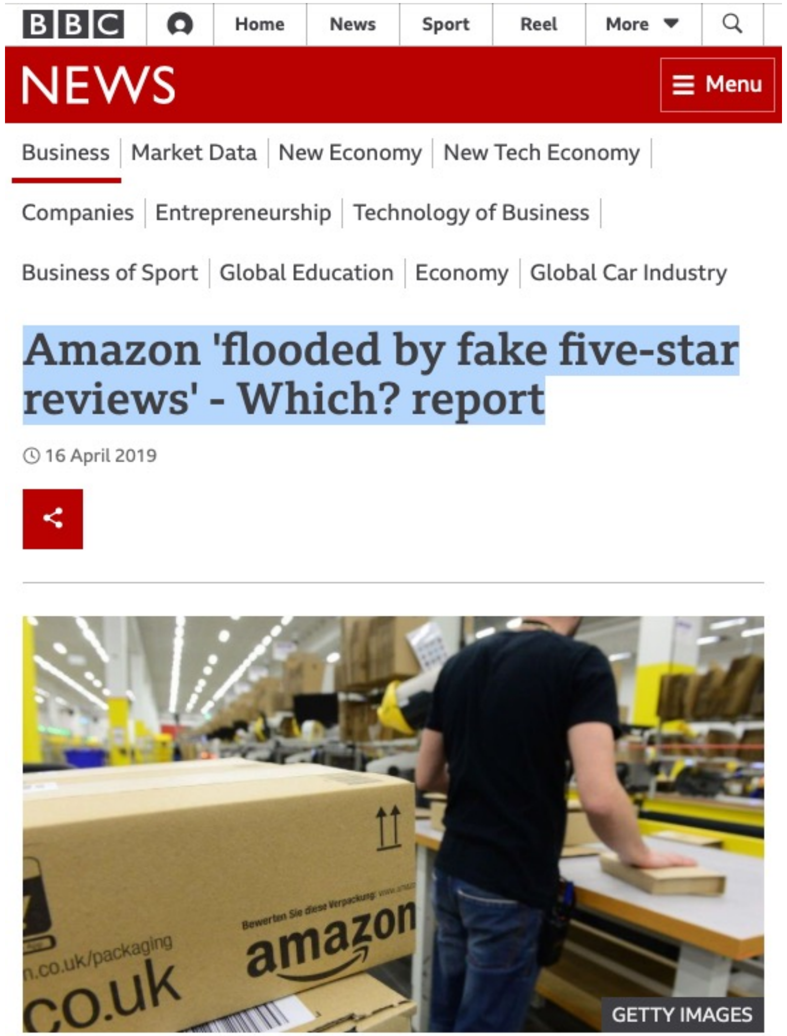
Small adversarial noise

ϵ

Classified as gibbon

x'

Attacks can happen in Recommender Systems


BBC | Home | News | Sport | Reel | More

NEWS ≡ Menu

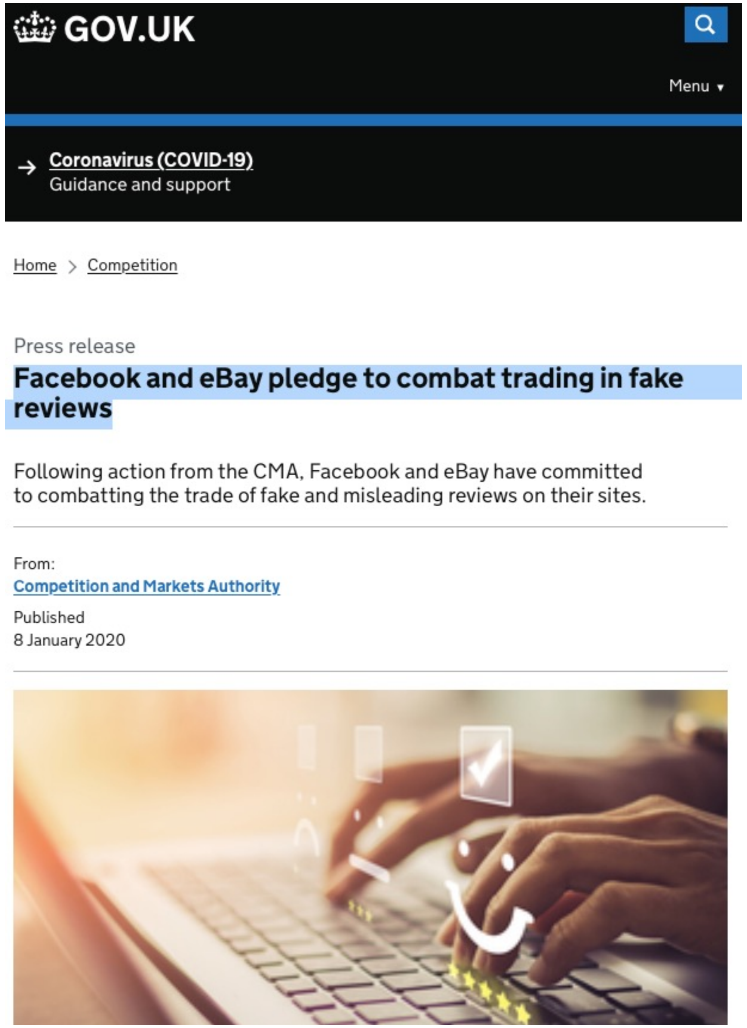
Business | Market Data | New Economy | New Tech Economy

Companies | Entrepreneurship | Technology of Business

Business of Sport | Global Education | Economy | Global Car Industry

Amazon 'flooded by fake five-star reviews' - Which? report

16 April 2019


GOV.UK

Menu

Coronavirus (COVID-19)
Guidance and support

Home > Competition

Press release

Facebook and eBay pledge to combat trading in fake reviews

Following action from the CMA, Facebook and eBay have committed to combatting the trade of fake and misleading reviews on their sites.

From:
Competition and Markets Authority

Published
8 January 2020

"More than three-quarters of people are influenced by reviews when they shop online."

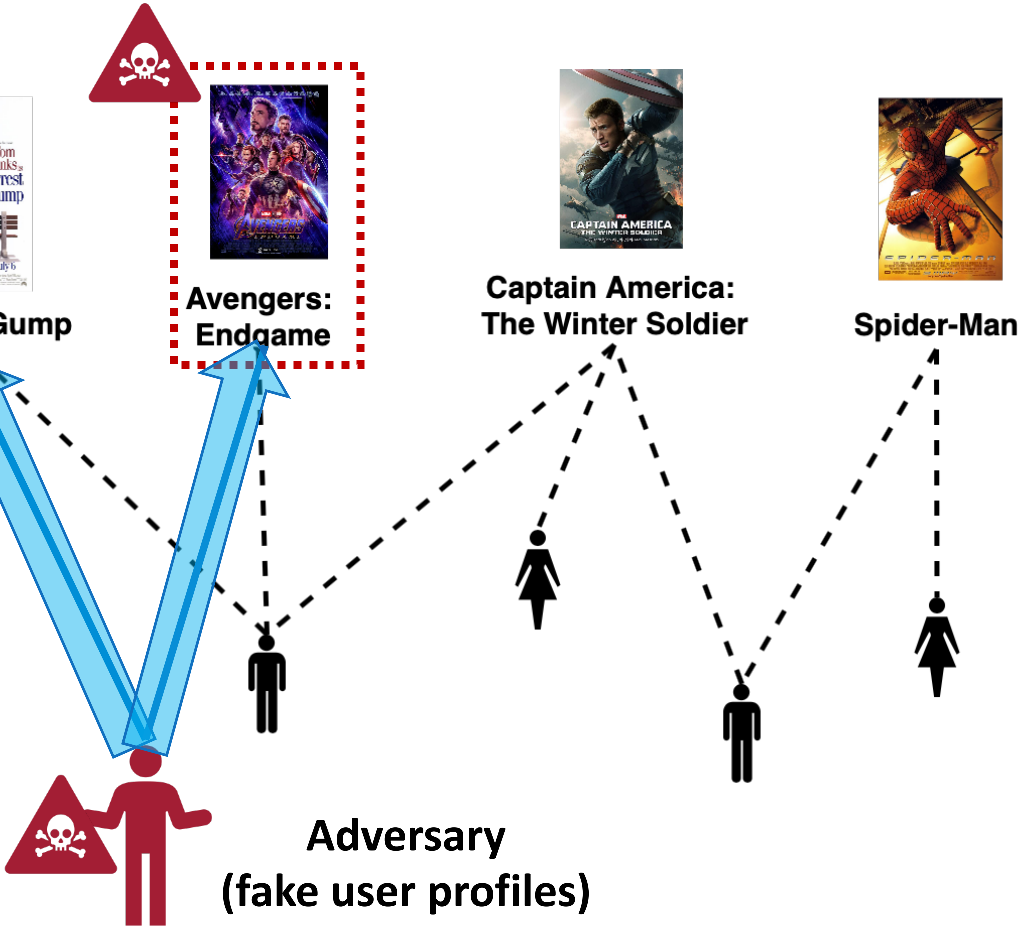
Understand how attacks can be performed

Defend against potential adversarial attacks

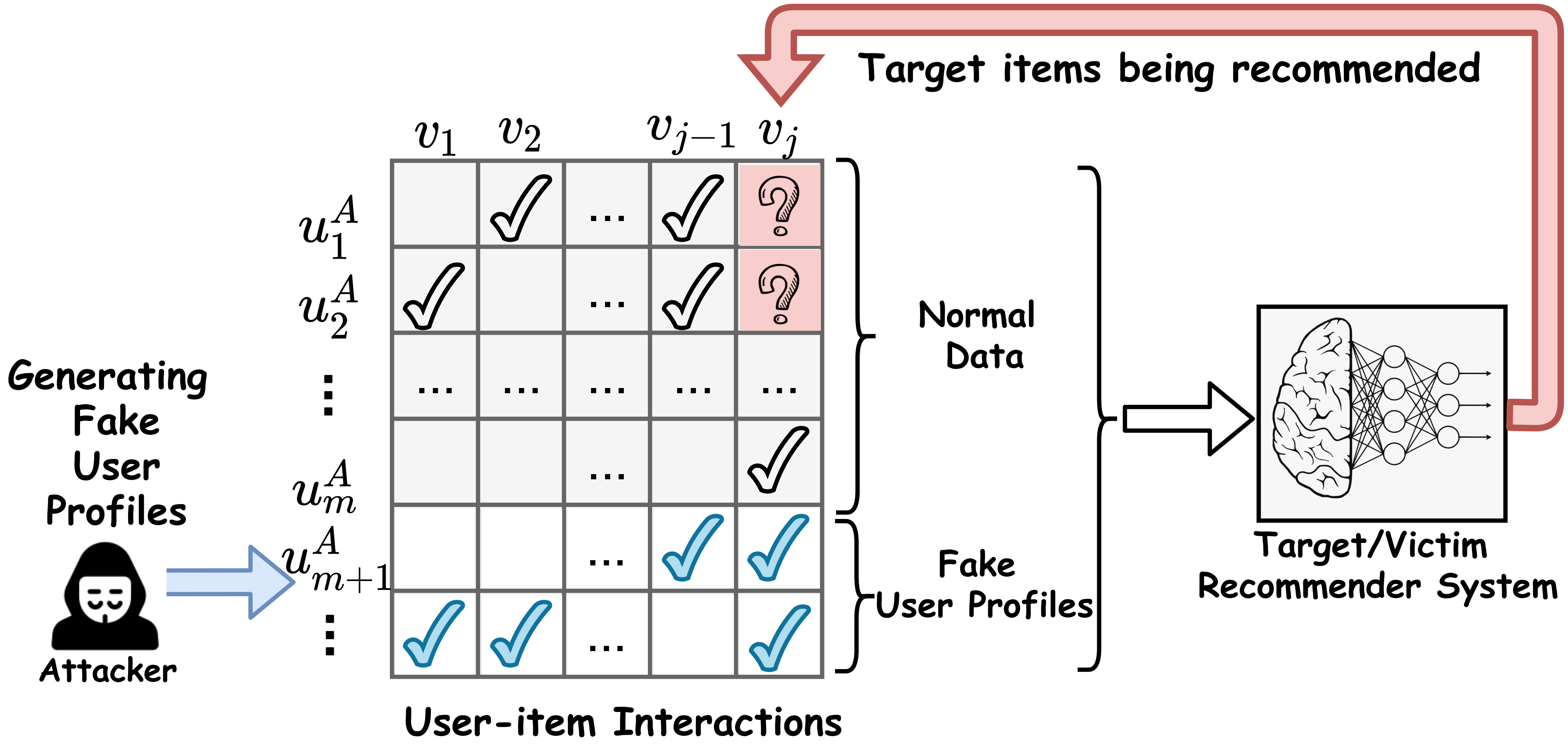
https://www.bbc.com/news/business-47941181
https://www.gov.uk/government/news/facebook-and-ebay-pledge-to-combat-trading-in-fake-reviews

Attacks can happen in Recommender Systems

- Security (Attacking) in Recommender Systems
  - **Data poisoning/shilling attacks**: promote/demote a set of items



A General Attacking Framework



Attack settings

❑White/grey-box attacks vs. Black-box attacks.
- have **full/partial knowledge** of the victim model/have **no knowledge**.

❑Targeted Attacks vs. Non-Targeted Attacks.
- attack **specific target** items / hurt the overall recommendation performance.

Adversarial Attacks

- White-box Attacks
  - Data Poisoning Attacks on Factorization-Based Collaborative Filtering (NIPS’16)

- Grey-box Attacks
  - Revisiting Adversarially Learned Injection Attacks Against Recommender Systems (RecSys’20)
  - Adversarial Attacks on an Oblivious Recommender (RecSys’19)

- Black-box Attacks
  - CopyAttack: Attacking Black-box Recommendations via Copying Cross-domain User Profiles (ICDE’21)
  - PoisonRec: An Adaptive Data Poisoning Framework for Attacking Black-box Recommender Systems (ICDE’20)

Adversarial Attacks

- White-box Attacks
  - Data Poisoning Attacks on Factorization-Based Collaborative Filtering (NIPS’16)
- Grey-box Attacks
  - Revisiting Adversarially Learned Injection Attacks Against Recommender Systems (RecSys’20)
  - Adversarial Attacks on an Oblivious Recommender (RecSys’19)
- Black-box Attacks
  - CopyAttack: Attacking Black-box Recommendations via Copying Cross-domain User Profiles (ICDE’21)
  - PoisonRec: An Adaptive Data Poisoning Framework for Attacking Black-box Recommender Systems (ICDE’20)

Preliminaries

- Collaborative Filtering:
  - Given data. $\mathbf{M} \in \mathbb{R}^{m \times n}$, $\Omega = \{(i,j) : \mathbf{M}_{ij} \text{ is observed}\}$
  - Goal: matrix completion

$$\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \|\mathcal{R}_\Omega(\mathbf{M} - \mathbf{X})\|_F^2, \quad s.t. \;\; \mathrm{rank}(\mathbf{X}) \leq k.$$

- Alternating minimization:

$$\min_{\mathbf{U} \in \mathbb{R}^{m \times k}, \mathbf{V} \in \mathbb{R}^{n \times k}} \{\|\mathcal{R}_\Omega(\mathbf{M} - \mathbf{U}\mathbf{V}^\top)\|_F^2 + 2\lambda_U \|\mathbf{U}\|_F^2 + 2\lambda_V \|\mathbf{V}\|_F^2\}$$

Data Poisoning Attacks on Factorization-Based Collaborative Filtering, NIPS, 2016

Attacking Formulation

- Inject malicious users $\widetilde{\mathbf{M}} \in \mathbb{R}^{m' \times n}$
- The CF formulations will be:

$$\mathbf{\Theta}_\lambda(\widetilde{\mathbf{M}};\mathbf{M}) = \arg\min_{\mathbf{U},\widetilde{\mathbf{U}},\mathbf{V}} \|\mathcal{R}_\Omega(\mathbf{M}-\mathbf{U}\mathbf{V}^\top)\|_F^2 + \boxed{\|\mathcal{R}_{\tilde{\Omega}}(\widetilde{\mathbf{M}}-\widetilde{\mathbf{U}}\mathbf{V}^\top)\|_F^2} + 2\lambda_U(\|\mathbf{U}\|_F^2+\|\widetilde{\mathbf{U}}\|_F^2)+2\lambda_V\|\mathbf{V}\|_F^2$$

- Goal： $\widetilde{\mathbf{M}}^* \in \mathrm{argmax}_{\widetilde{\mathbf{M}}\in\mathbb{M}} R(\widehat{\mathbf{M}}(\mathbf{\Theta}_\lambda(\widetilde{\mathbf{M}};\mathbf{M})),\mathbf{M})$

- Solution: Projected gradient ascent (PGA)

$$\widetilde{\mathbf{M}}^{(t+1)} = \mathrm{Proj}_{\mathbb{M}}\left(\widetilde{\mathbf{M}}^{(t)} + s_t \cdot \nabla_{\widetilde{\mathbf{M}}} R(\widehat{\mathbf{M}},\mathbf{M})\right)$$

Data Poisoning Attacks on Factorization-Based Collaborative Filtering, NIPS, 2016

Adversarial Attacks

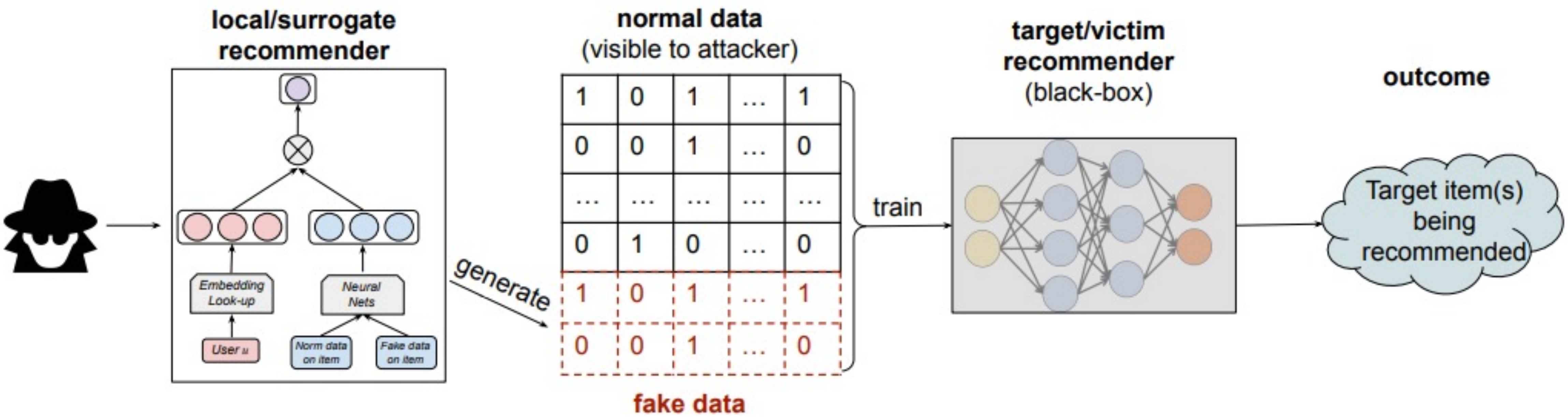
- White-box Attacks
  - Data Poisoning Attacks on Factorization-Based Collaborative Filtering (NIPS’16)

- Grey-box Attacks
  - Revisiting Adversarially Learned Injection Attacks Against Recommender Systems (RecSys’20)
  - Adversarial Attacks on an Oblivious Recommender (RecSys’19)

- Black-box Attacks
  - CopyAttack: Attacking Black-box Recommendations via Copying Cross-domain User Profiles (ICDE’21)
  - PoisonRec: An Adaptive Data Poisoning Framework for Attacking Black-box Recommender Systems (ICDE’20)

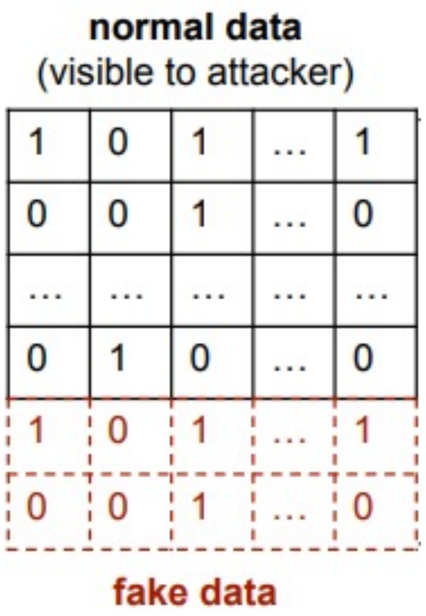
Threat Model

Attacker's Goal: promote certain items availability of being recommended
Attacker's knowledge: fully (partial) observable dataset



How to attack a RecSys: A bi-level optimization problem

- Step 1: Train surrogate model



Training Recommender System

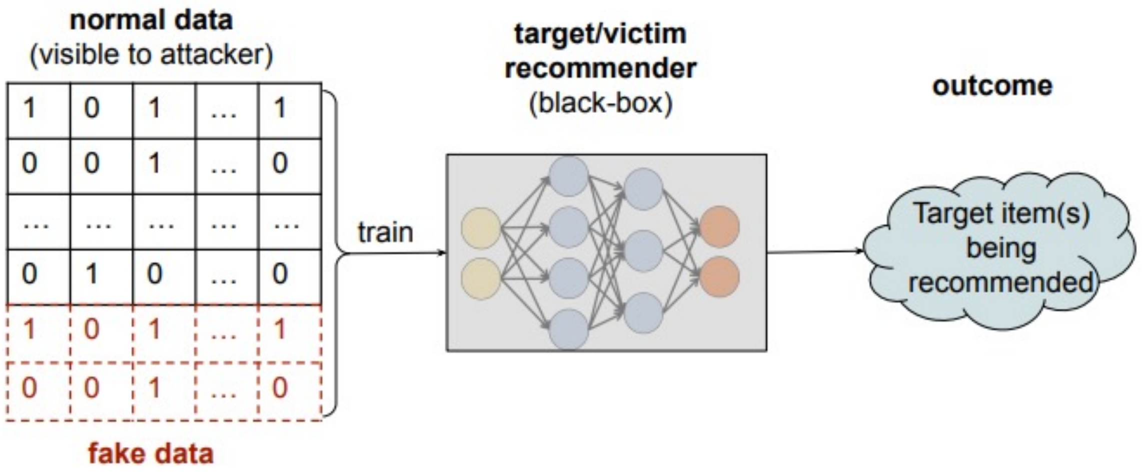
$$\theta^* = \arg\min_{\theta} \left(\mathcal{L}_{\text{train}}(\underbrace{\boldsymbol{X}}_{\text{Normal Data}}, \boldsymbol{R}_\theta) + \mathcal{L}_{\text{train}}(\underbrace{\widehat{\boldsymbol{X}}}_{\text{Injected fake data}}, \widehat{\boldsymbol{R}}_\theta) \right)$$

where $\hat{X}$ is the fake rating matrix, θ^* is the parameters of the surrogate model

Revisiting Adversarially Learned Injection Attacks Against Recommender Systems. RecSys 2020.

How to attack a RecSys: A bi-level optimization problem

- Step 2: Evaluate the malicious goal after fake data are consumed



Adversarial objective
(defined on prediction on normal data)

$$\min_{\widehat{\boldsymbol{X}}} \mathcal{L}_{\text{adv}}(\boldsymbol{R}_{\theta^*}) = -\sum_{u \in \mathcal{U}} \log \left(\frac{\exp(r_{uk})}{\sum_{i \in \mathcal{I}} \exp(r_{ui})} \right).$$

$$\text{subject to} \quad \boxed{\theta^*} = \arg\min_{\theta} \left(\mathcal{L}_{\text{train}}(\boldsymbol{X}, \boldsymbol{R}_\theta) + \mathcal{L}_{\text{train}}(\widehat{\boldsymbol{X}}, \widehat{\boldsymbol{R}}_\theta) \right)$$

Well-trained surrogate model parameters

Revisiting Adversarially Learned Injection Attacks Against Recommender Systems. RecSys 2020.

Solving the bi-level optimization: gradient-based

Algorithm 1 Learning fake user data with Gradient Descent

```
Input: Normal user data X; learning rate for inner and outer objective: α and η; max iteration for inner and outer objective: L and T.
Output: Learned fake user data for malicious goal.
Initialize fake data X̂^(0) and surrogate model parameters θ^(0)
for t = 1 to T do
    for l = 1 to L do
        Optimize inner objective with SGD: θ^(l) ← θ^(l−1) − α · ∇_θ(L_train(X, R_θ^(l−1)) + L_train(X̂^(t), R̂^(t)_θ^(l−1)))
    end for
    Evaluate L_adv(R_θ^(L)) and compute gradients ∇_X̂ L_adv
    Update fake data: X̂^(t) = Proj_Λ(X̂^(t−1) − η · ∇_X̂ L_adv)
end for
Return: X̂^(T)
```

Train surrogate model based on new fake data

Obtain gradient and update fake data

Repeat until converge

Revisiting Adversarially Learned Injection Attacks Against Recommender Systems. RecSys 2020.

Limitations

How to obtain the desired gradients $\nabla_{\widehat{X}} \mathcal{L}_{\text{adv}}$

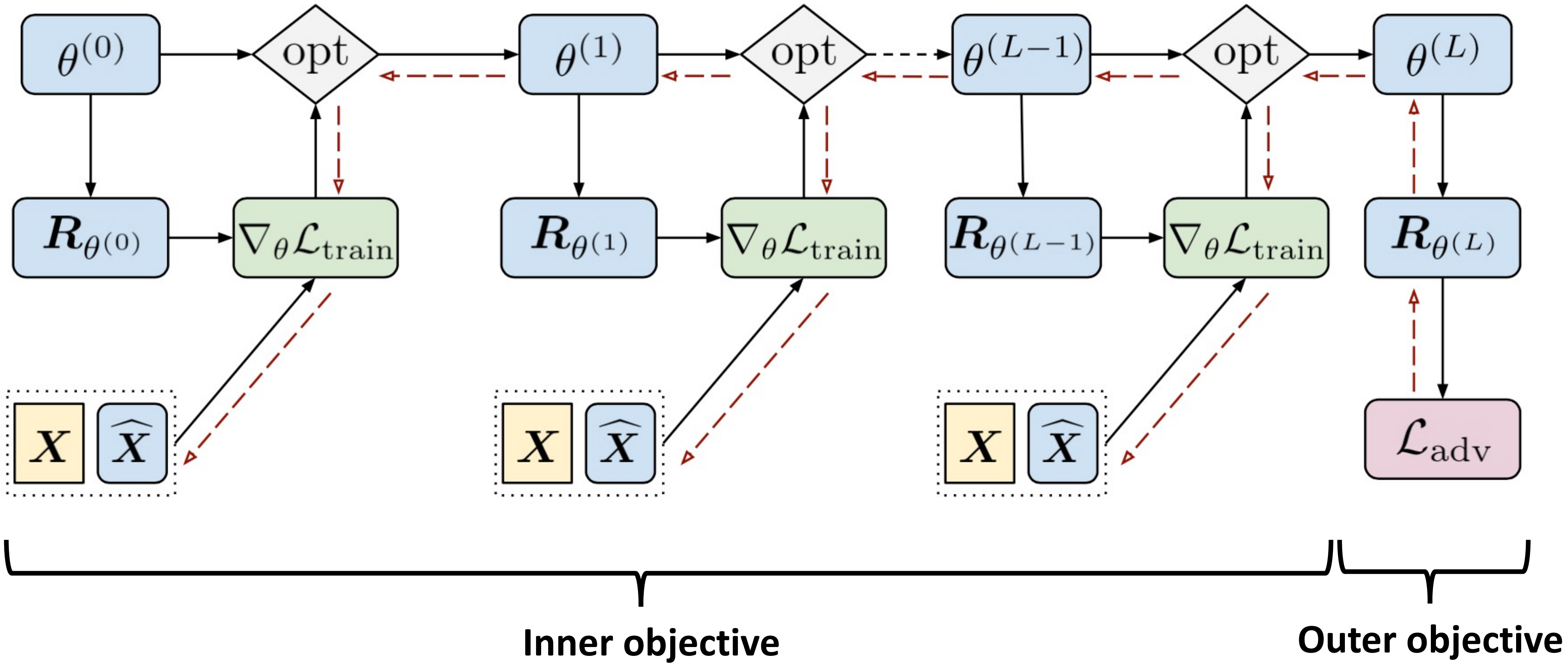
Lacking exactness in gradient computation

$$\nabla_{\widehat{X}} \mathcal{L}_{\text{adv}} = \frac{\partial \mathcal{L}_{\text{adv}}}{\partial \widehat{X}} + \underbrace{\frac{\partial \mathcal{L}_{\text{adv}}}{\partial \theta^*} \cdot \frac{\partial \theta^*}{\partial \widehat{X}}}_{\text{ignored}}.$$

Revisiting Adversarially Learned Injection Attacks Against Recommender Systems. RecSys 2020.

Computational graph

- **Exact Solution**



Forward computation flow: solid black arrow
Gradient backpropagation flow: dashed red arrow

Revisiting Adversarially Learned Injection Attacks Against Recommender Systems. RecSys 2020.

Adversarial Attacks

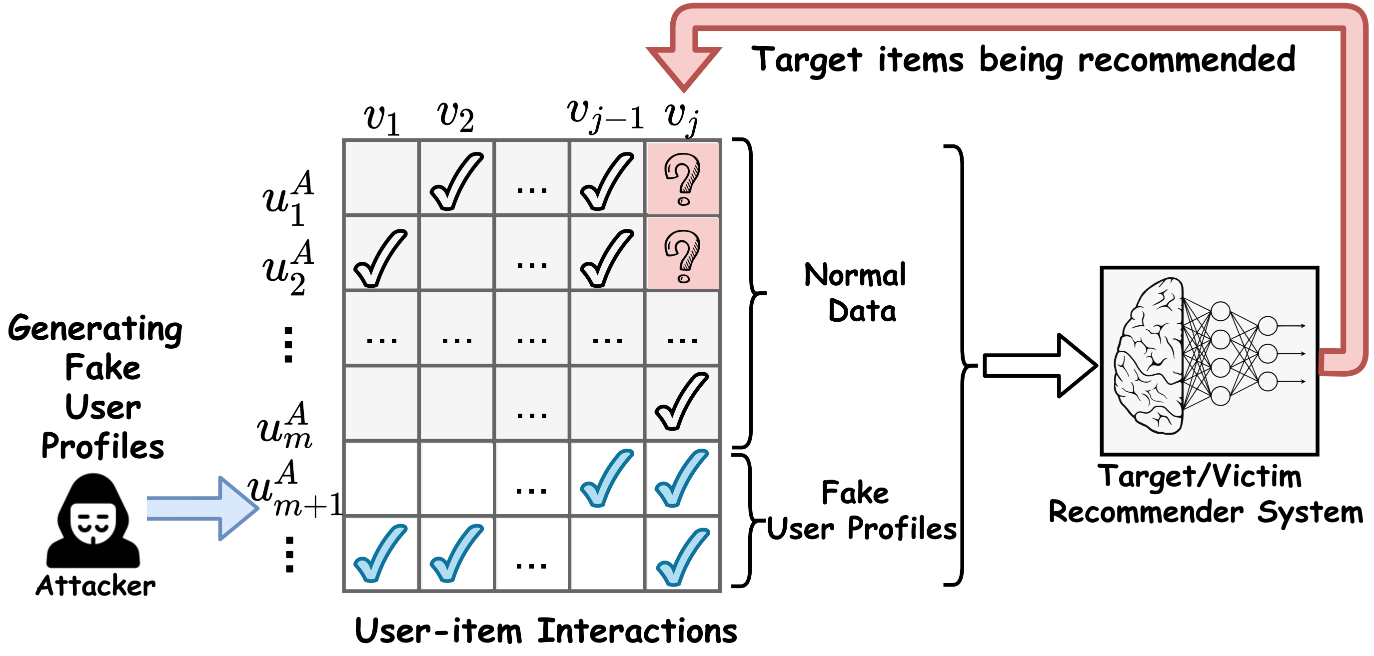
- White-box Attacks
  - Data Poisoning Attacks on Factorization-Based Collaborative Filtering (NIPS’16)

- Grey-box Attacks
  - Revisiting Adversarially Learned Injection Attacks Against Recommender Systems (RecSys’20)
  - Adversarial Attacks on an Oblivious Recommender (RecSys’19)

- Black-box Attacks
  - CopyAttack: Attacking Black-box Recommendations via Copying Cross-domain User Profiles (ICDE’21)
  - PoisonRec: An Adaptive Data Poisoning Framework for Attacking Black-box Recommender Systems (ICDE’20)

Challenges

- **Challenges in** existing attacking methods:
  - **Less "realistic" user profiles (easily detected)**



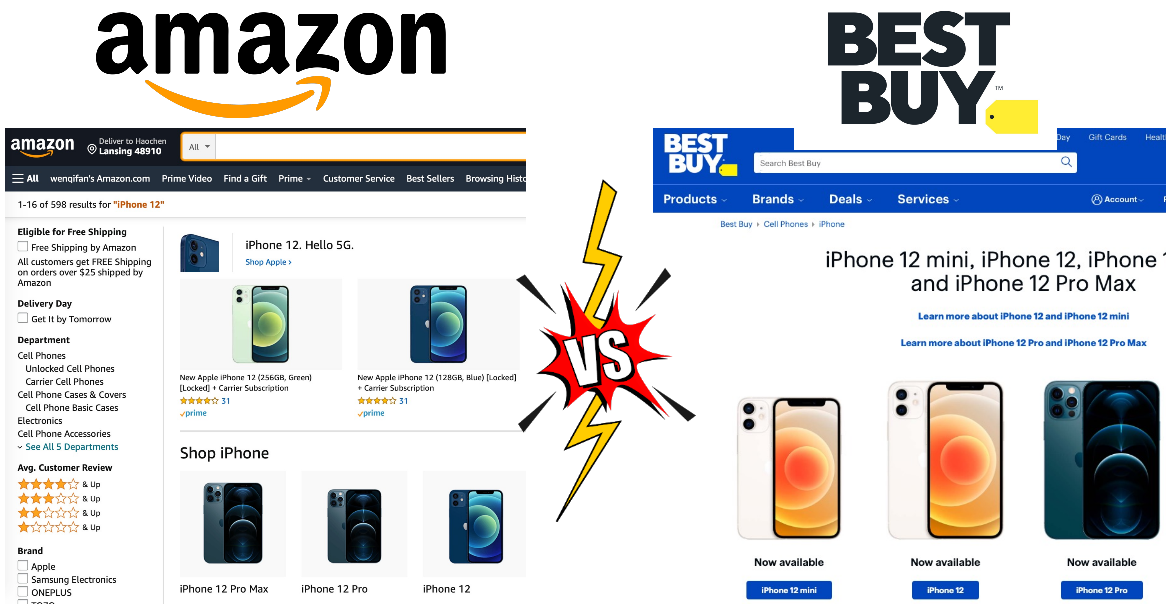
Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.

Solution

- **Cross-domain Information**
  - Share a lot of items
  - Users from these platforms with **similar functionalities** also share similar behavior patterns/preferences.

Taobao

JD.com

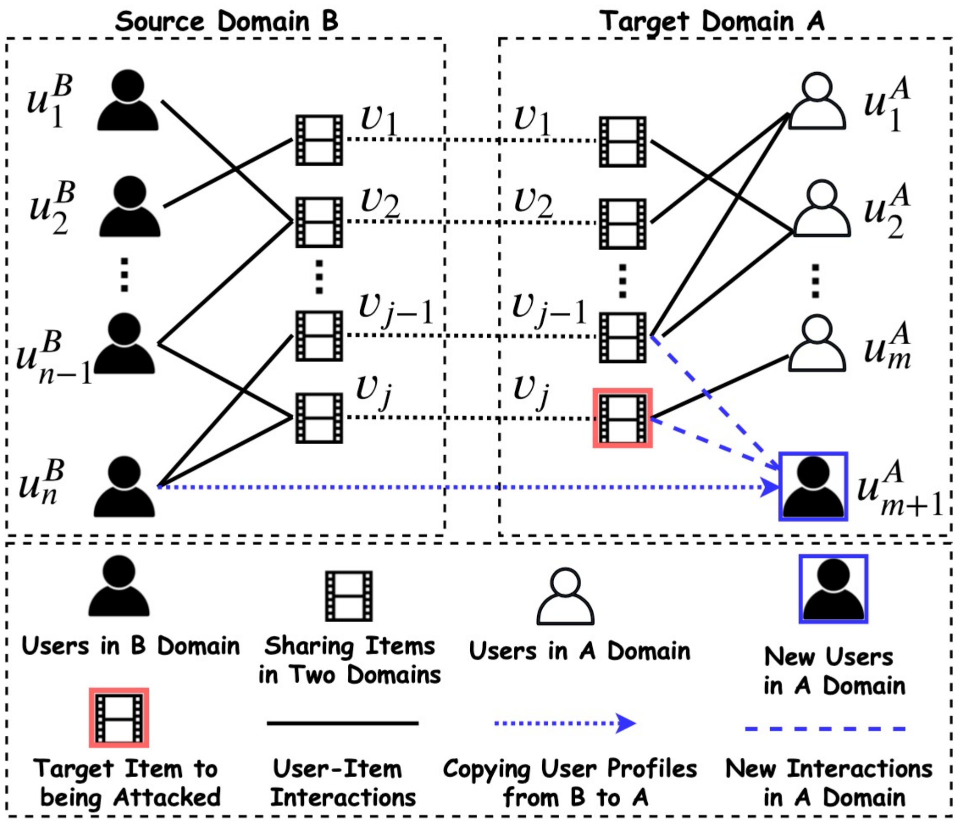


Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.

Solution

- **Challenges in** existing attacking methods:
  - **Less "realistic" user profiles (easily detected)**

Copy cross-domain users with real profiles from other domains



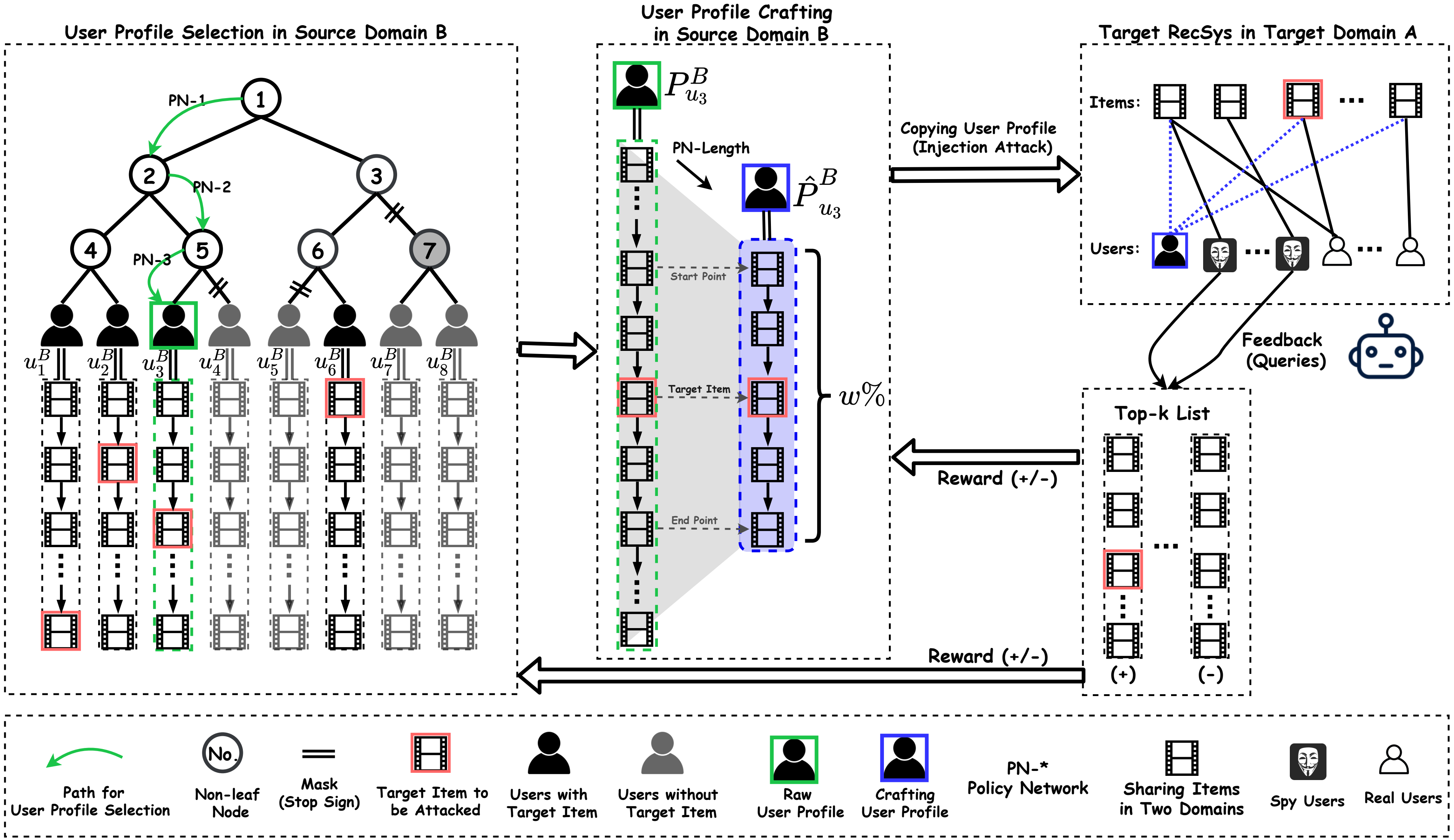
Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.

Challenges

- **Challenges in existing attacking methods:**
  - **Less "realistic" user profiles (easily detected)**
    - Cross-domain Information
  - **White/Grey-box setting** (i.e., model architecture and parameters, and datasets)
    → impossible and unrealistic (**privacy and security**)
  - **Black-box setting**
    - ➢ Reinforcement Learning (RL) -- Query Feedback (Reward)

Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.

CopyAttack



Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.

User Profile Selection

- User Profile Selection
  - Construct hierarchical clustering tree
  - Masking Mechanism - specific target items
  - Hierarchical-structure Policy Gradient

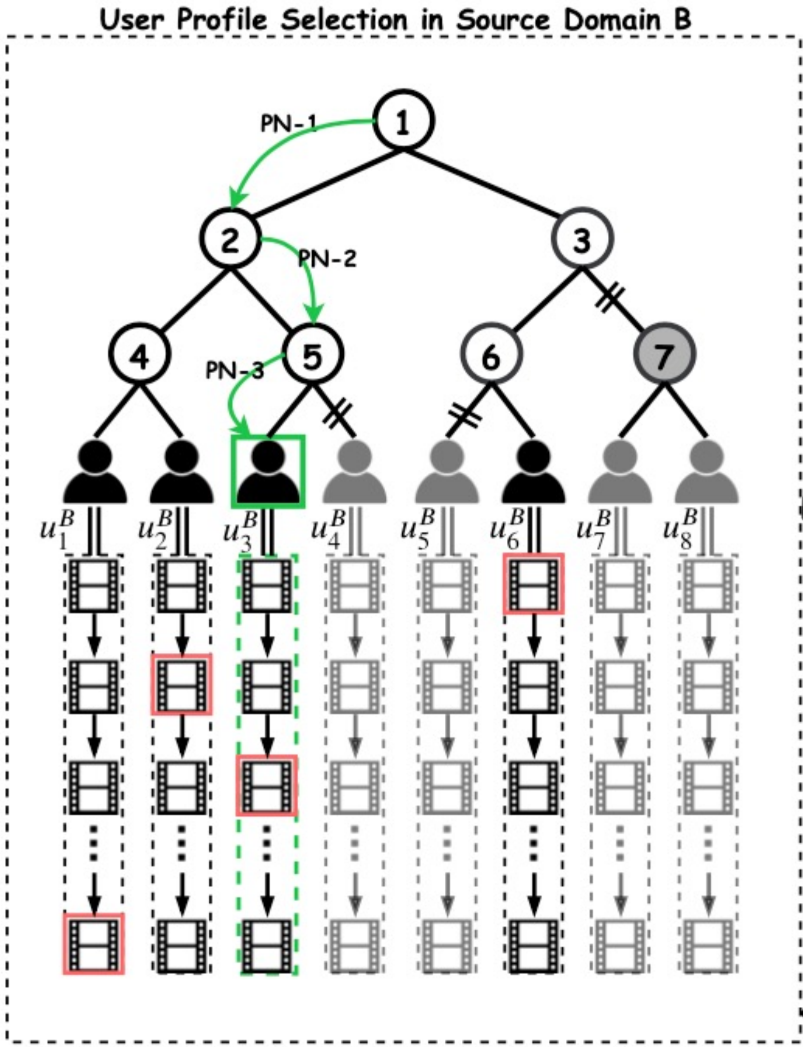
$$a_t^u = \left\{a_{[t,1]}^u, a_{[t,2]}^u, \ldots, a_{[t,d]}^u\right\}$$

$$\begin{aligned} p^u(a_t^u|s_t^u) &= \prod^{d} p_d^u(a_t^u|\cdot, s_t^u) \\ &= p_d^u(a_{[t,d]}^u|s_t^u) \cdot p_{d-1}^u(a_{[t,d-1]}^u|s_t^u) \cdots p_1^u(a_{[t,1]}^u|s_t^u) \end{aligned}$$

$$\mathbf{x}_{v_*} = RNN(\mathcal{U}_t^{B\rightarrow A}),$$

$$p_i^u(\cdot|s_t^u) = softmax(MLP([\mathbf{q}_{v_*}^B \oplus \mathbf{x}_{v_*}]|\theta_i^u))$$

Time Complexity： $\mathcal{O}(|\mathcal{U}^B|) \longrightarrow \mathcal{O}(d \times |\mathcal{U}^B|^{1/d})$



Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.

User Profile Crafting

- User Profile Crafting
  - Clipping operation to craft the raw user profiles

$$W = \{10\%, 20\%, 30\%, 40\%, 50\%, 60\%, 70\%, 80\%, 90\%, 100\%\}$$

  - Sequential patterns **(forward/backward)**

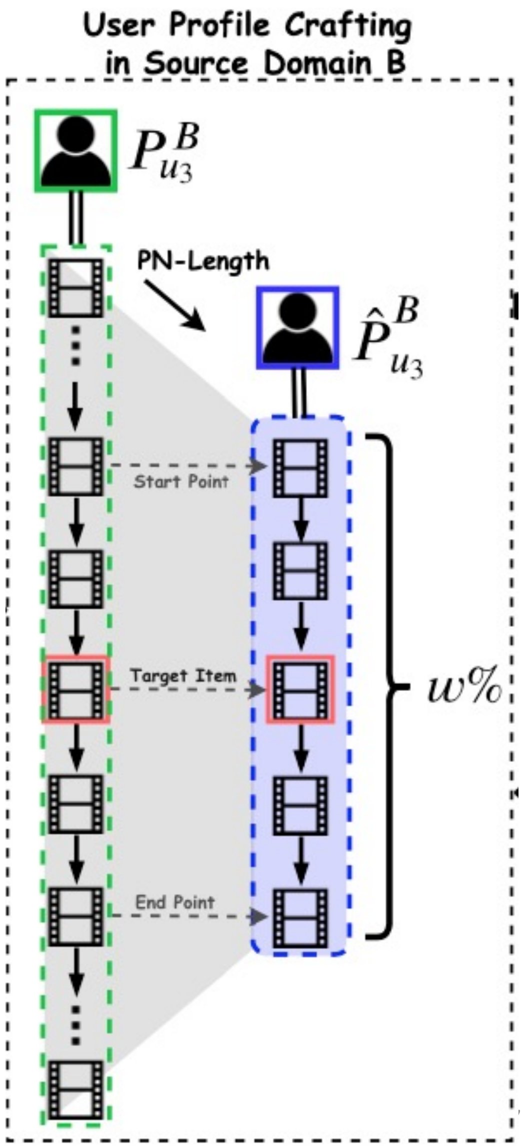
Example:

w = 50%

$$P^B_{u_i} = \{v_1 \rightarrow v_2 \rightarrow \boxed{v_3 \rightarrow v_4 \rightarrow v_{5*} \rightarrow v_6 \rightarrow v_7} \rightarrow v_8 \rightarrow v_9 \rightarrow v_{10}\}$$

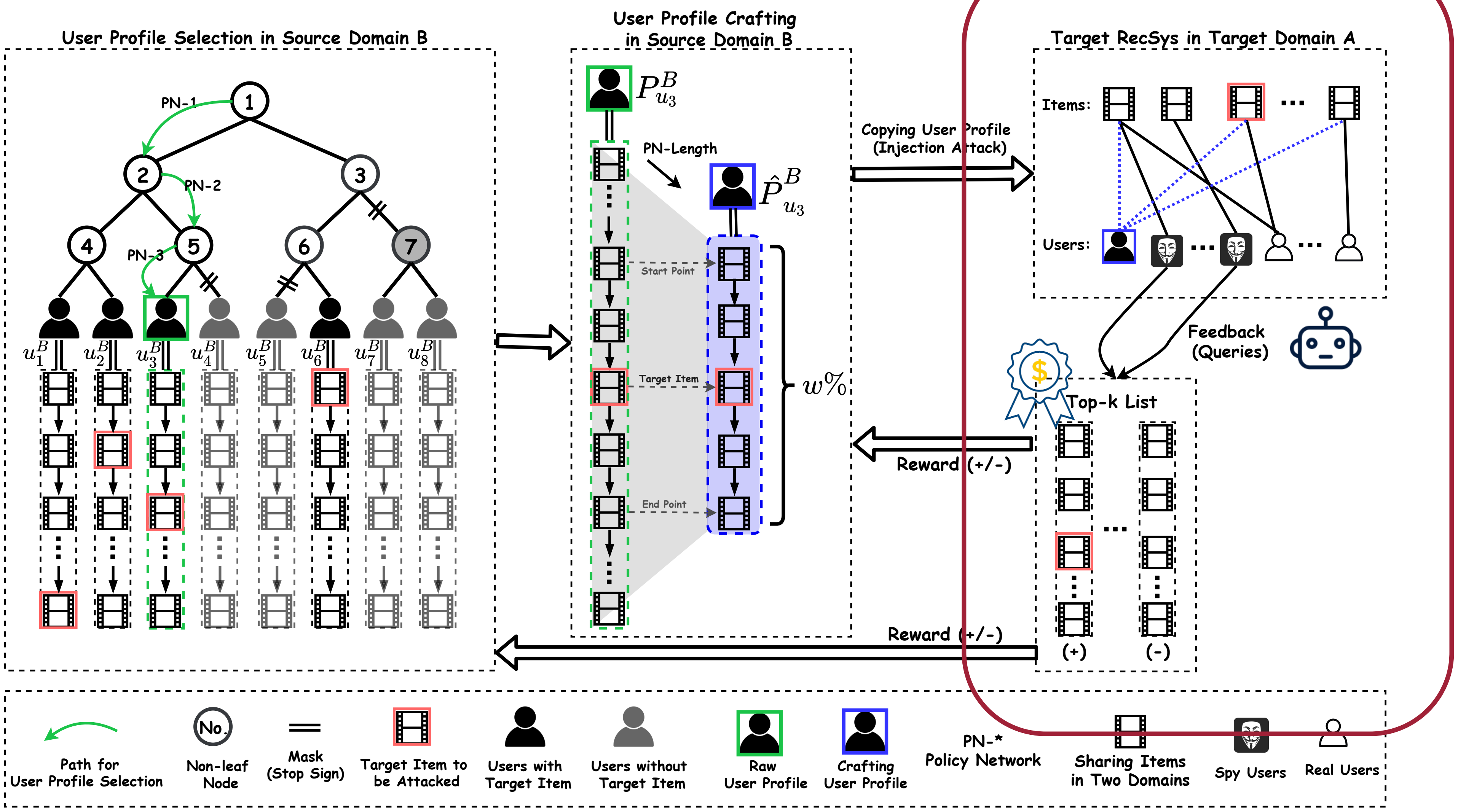
$$\hat{P}^B_{u_i} = \{v_3 \rightarrow v_4 \rightarrow v_{5*} \rightarrow v_6 \rightarrow v_7\}$$

$$p^l(\cdot|s^l_t) = softmax(MLP([\mathbf{p}^B_i \oplus \mathbf{q}^B_{v_*}]|\theta^l))$$



Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.

CopyAttack



Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE, 2021.