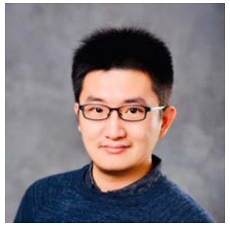
Trustworthy Recommender Systems: Foundations and Frontiers




Wenqi Fan[1], Xiangyu Zhao[2], Lin Wang[1], Xiao Chen[1], Jingtong Gao[2], Qidong Liu[2], Shijie Wang[1]

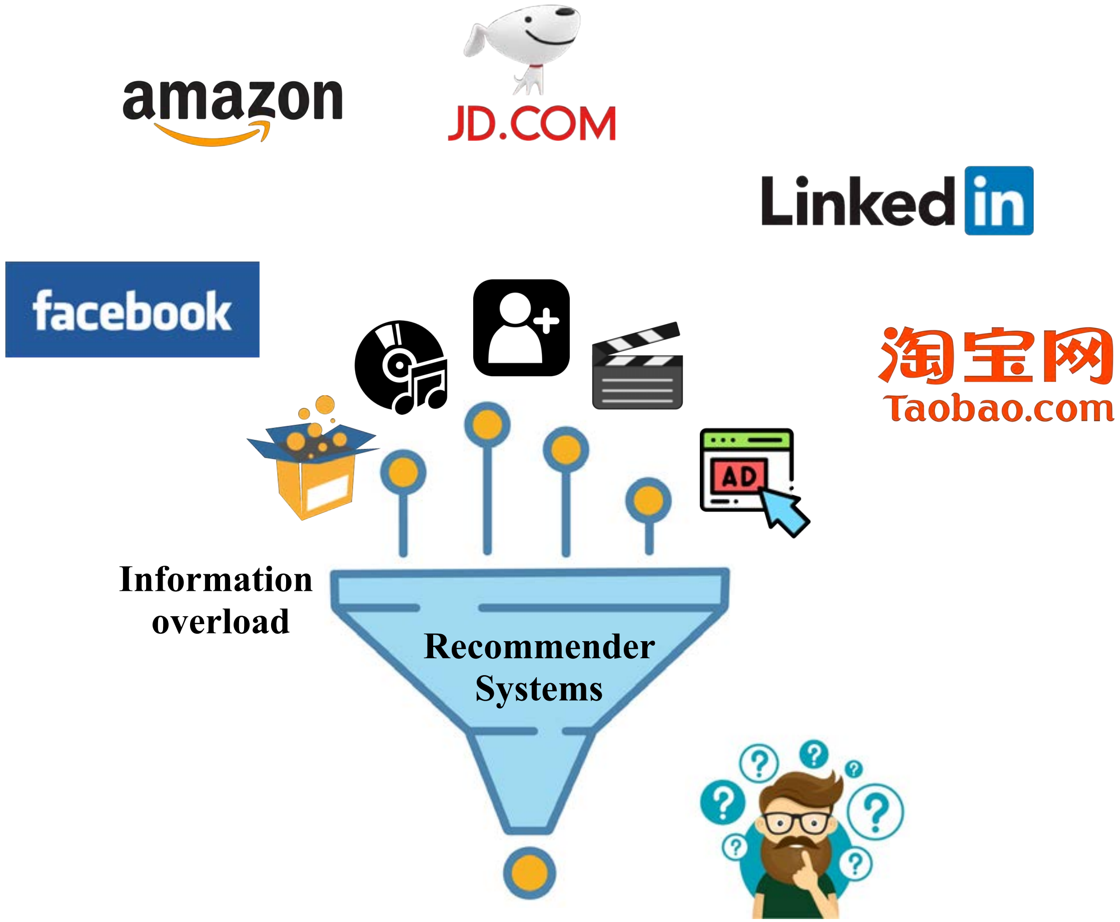
[1]The Hong Kong Polytechnic University, [2]City University of Hong Kong

Website (Slides): https://advanced-recommender-systems.github.io/trustworthy-recommendations/

Survey: A Comprehensive Survey on Trustworthy Recommender Systems, arXiv:2209.10117, 2022.

Recommender Systems

Age of Information Explosion



Recommend item X to user

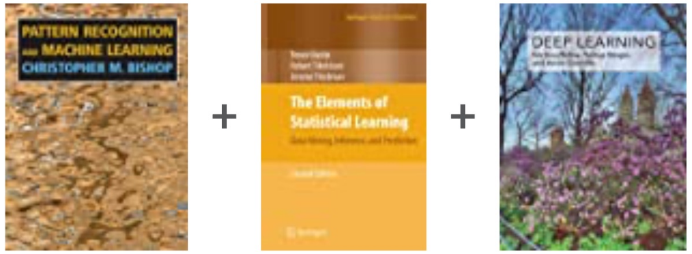
Items can be: Products, Friends, News, Movies, Videos, etc.

Recommender Systems

Recommendation has been widely applied in online services:
- **E-commerce**, Content Sharing, Social Networking ...

Product Recommendation

Frequently bought together



A + B + C

Total price: $208.9

Add all three to Cart

Add all three to List

Amazon's recommendation algorithm drives **35%** of its sales [from McKinsey, 2012]

Recommender Systems

Recommendation has been widely applied in online services:

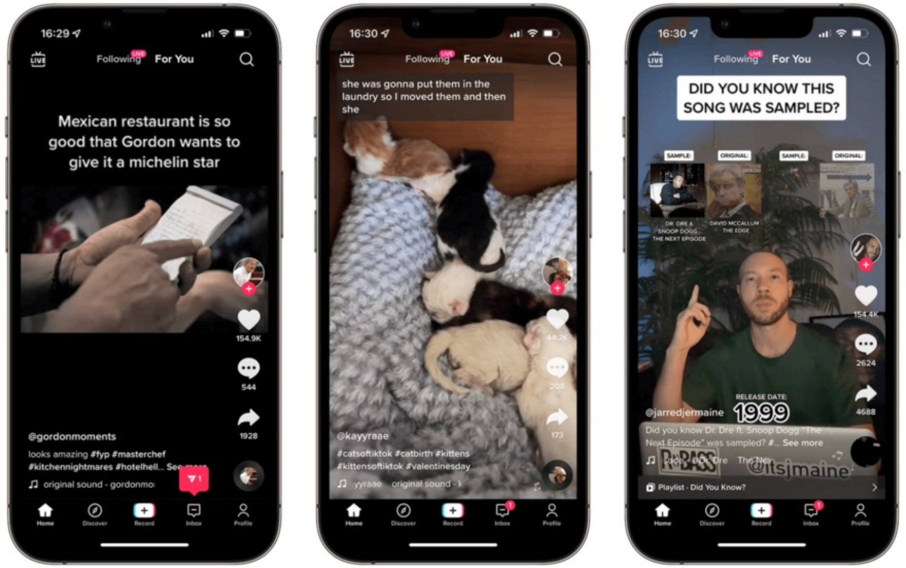
- E-commerce, **Content Sharing**, Social Networking ...

News/Video/Image Recommendation

TikTok's recommendation algorithm

Top 10 Global Breakthrough Technologies in 2021

MIT Technology Review



Recommender Systems

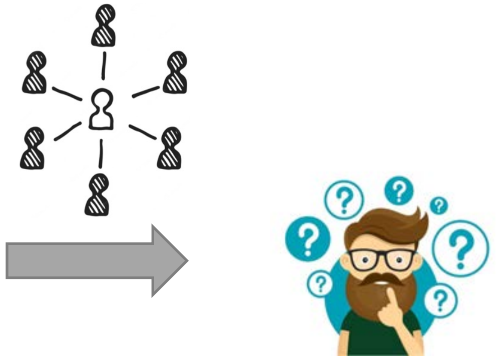
Recommendation has been widely applied in online services:

- E-commerce, Content Sharing, **Social Networking** ...

facebook

新浪微博 weibo.com

LinkedIn



Social Recommendations

WeChat

Like what you read?
Share with your friends!

Subscriptions (訂閱號信息)

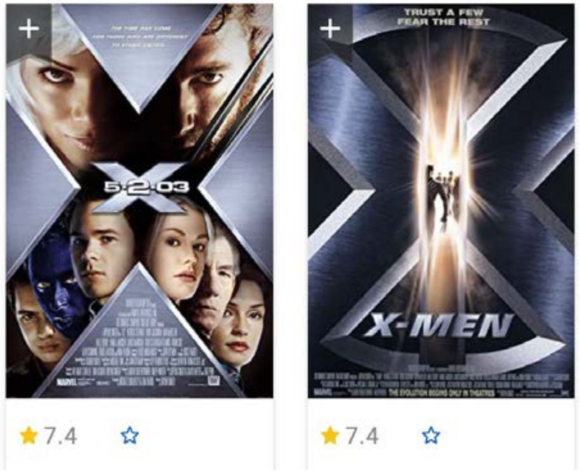
Read by 9 friends

Top Stories（看一看）
Wow (朋友在看)

Recommender System is Everywhere

Business

Healthcare



Entertainment

Education

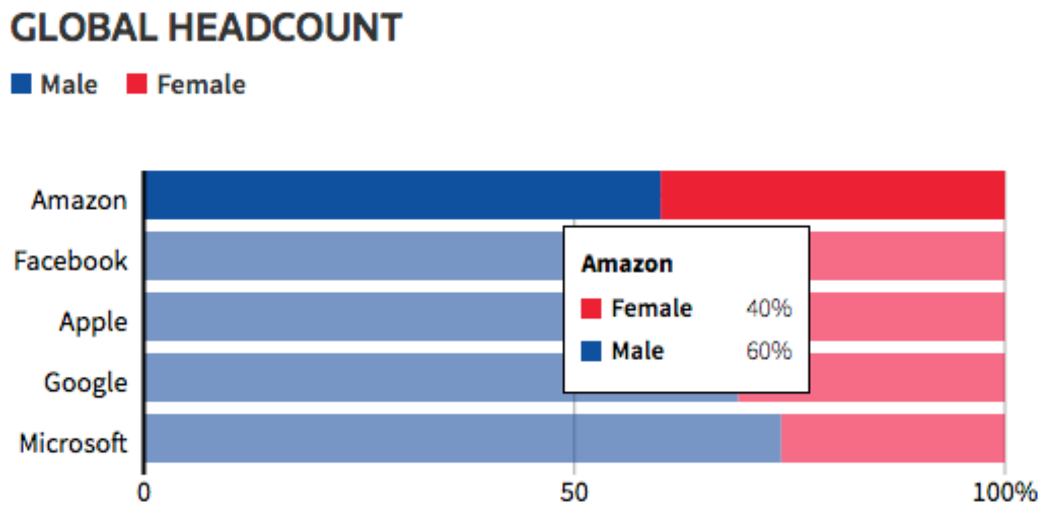
The Good and The Bad

The Good

<u>The Bad</u>

Discrimination & Fairness Issue

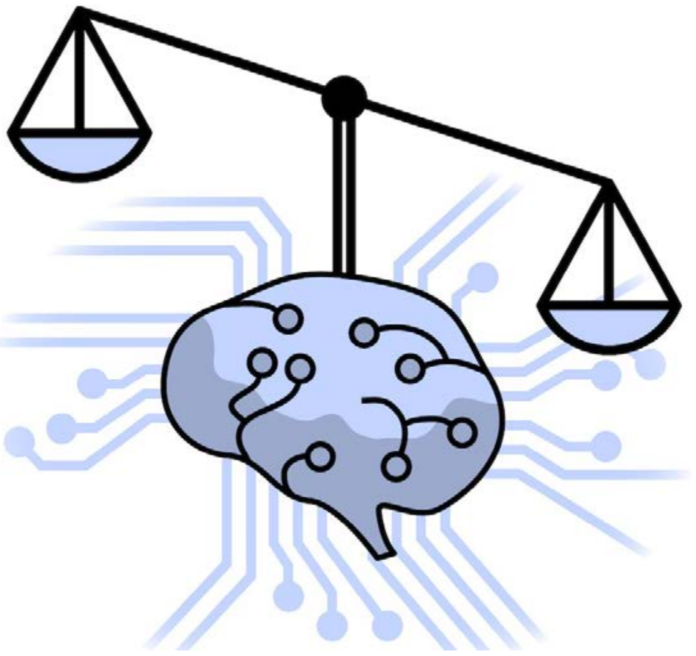
Job recommendation
(Lambrecht et al., 2019)



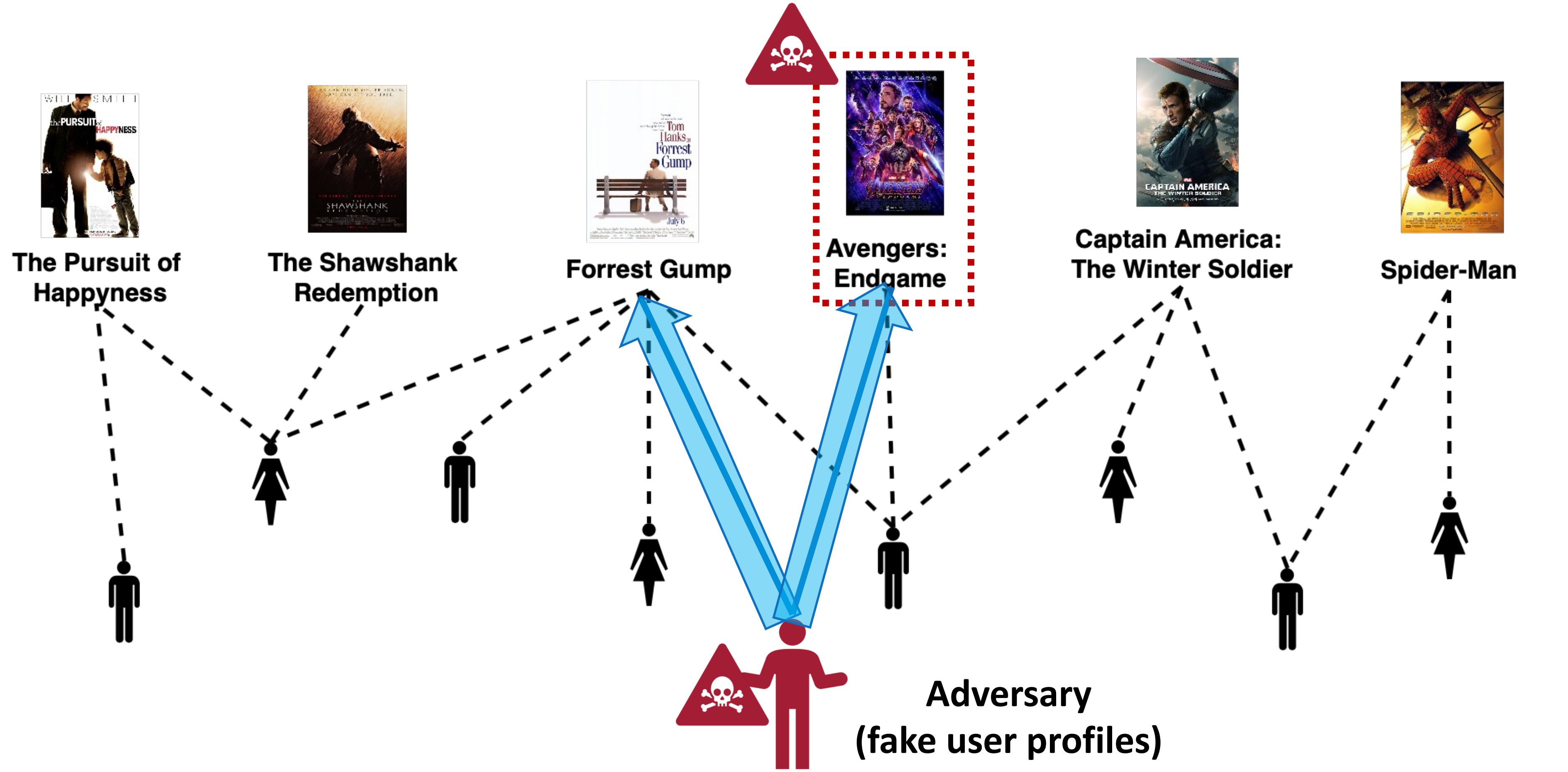
Lambrecht, et al. "Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of STEM career ads." *2019*.
Bias and Debias in Recommender System: A Survey and Future Directions, 2021.

Non-discrimination & Fairness

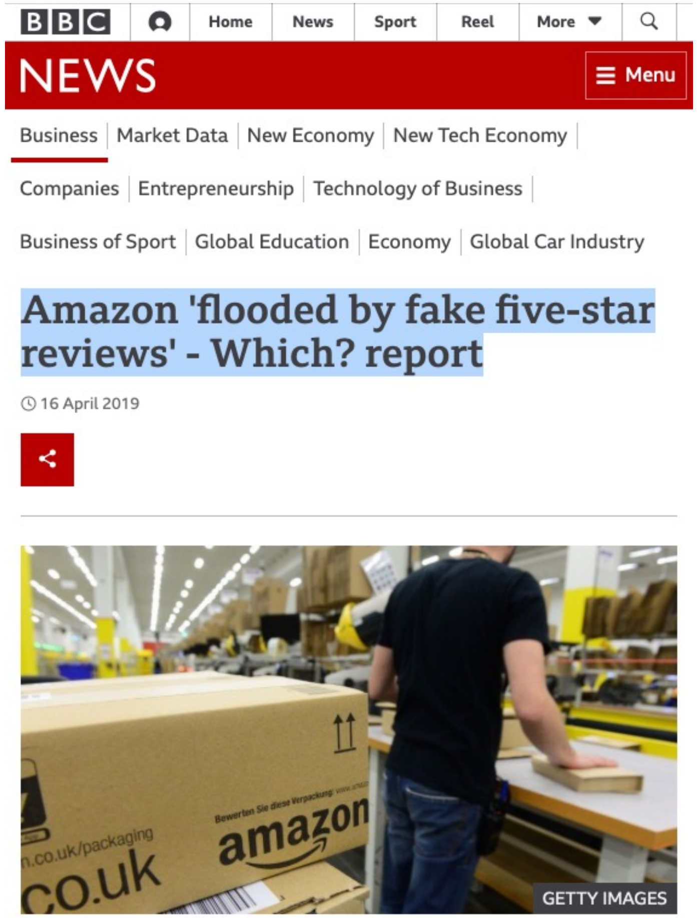
- A recommender system should avoid discriminatory behaviors in human-machine interaction.
- A recommender system should ensure fairness in decision-making.

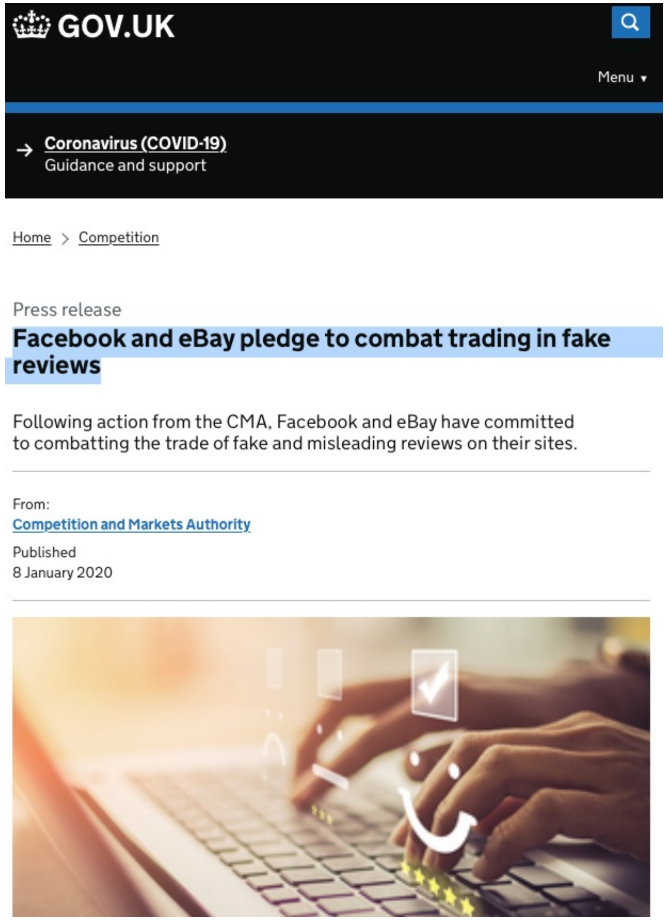


Safety & Robustness Issue



Attacks can happen in Recommender Systems





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

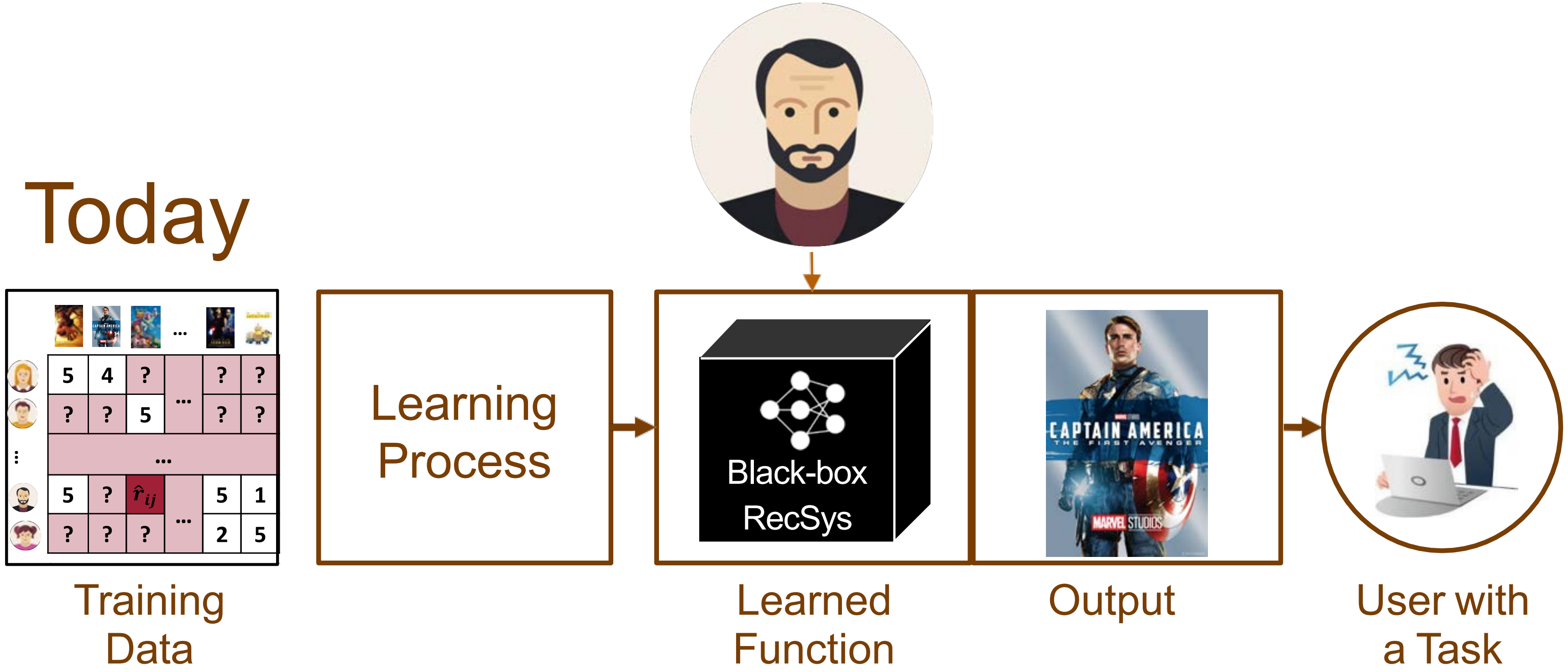
Understand system's vulnerability and how attacks can be performed

Defend against potential adversarial attacks

"The Impact of Fake Reviews on Online Visibility: A Vulnerability Assessment of the Hotel Industry", Information Systems Research, 2016
https://www.bbc.com/news/business-47941181
https://www.gov.uk/government/news/facebook-and-ebay-pledge-to-combat-trading-in-fake-reviews

Black-box Issue

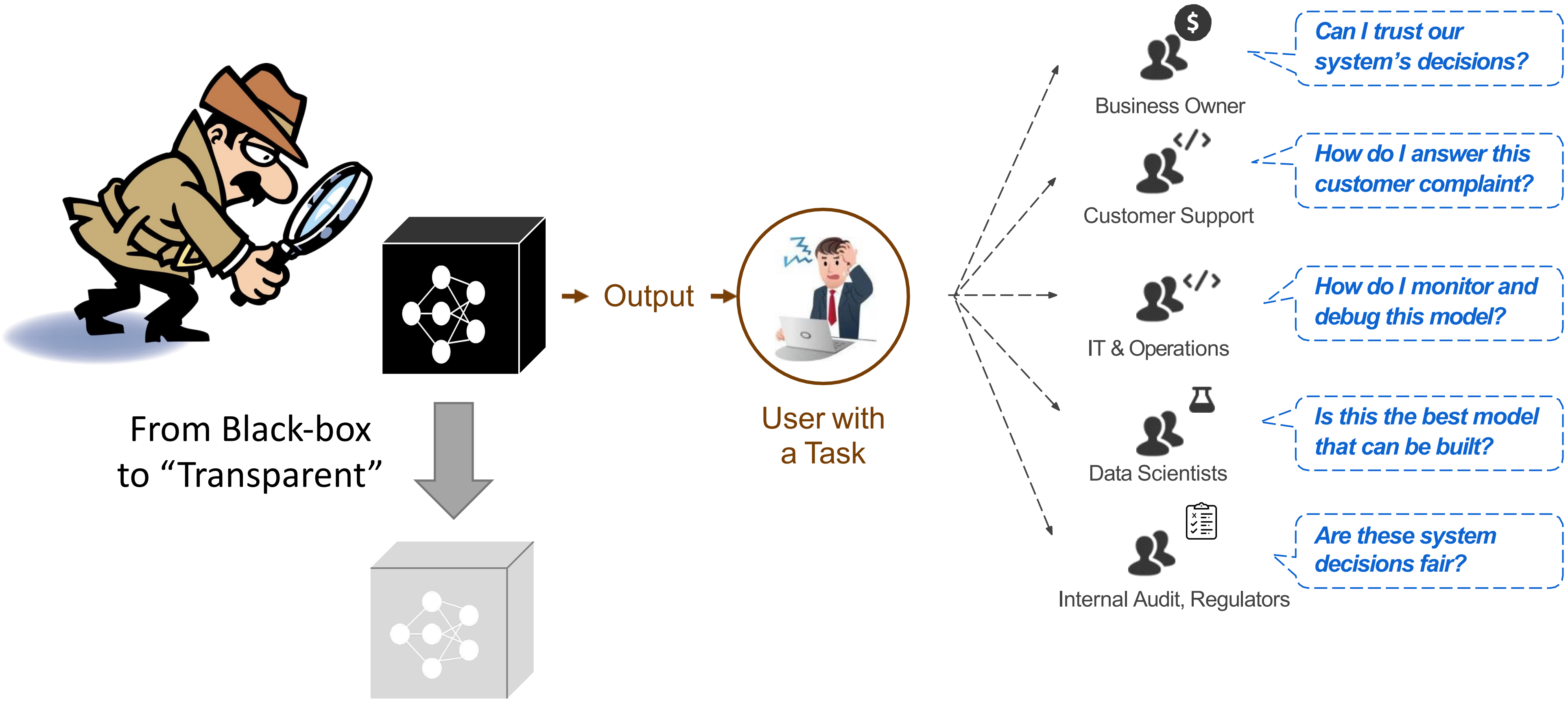
How recommender systems work?



- Why did you do that?
- Why not something else?
- When do you succeed?
- When do you fail?
- When can I trust you?
- How do I correct an error?

Explainability

Black-box system creates confusion and doubt



The Need for Explainable Recommendation

Yongfeng Zhang, et.al, Explainable Recommendation: A Survey and New Perspectives, 2020.

Privacy Issue

- The success of recommender systems heavily relies on data that might contain private and sensitive information.
- Can we still take the advantages of data while effectively protecting the privacy?

Environmental Issue

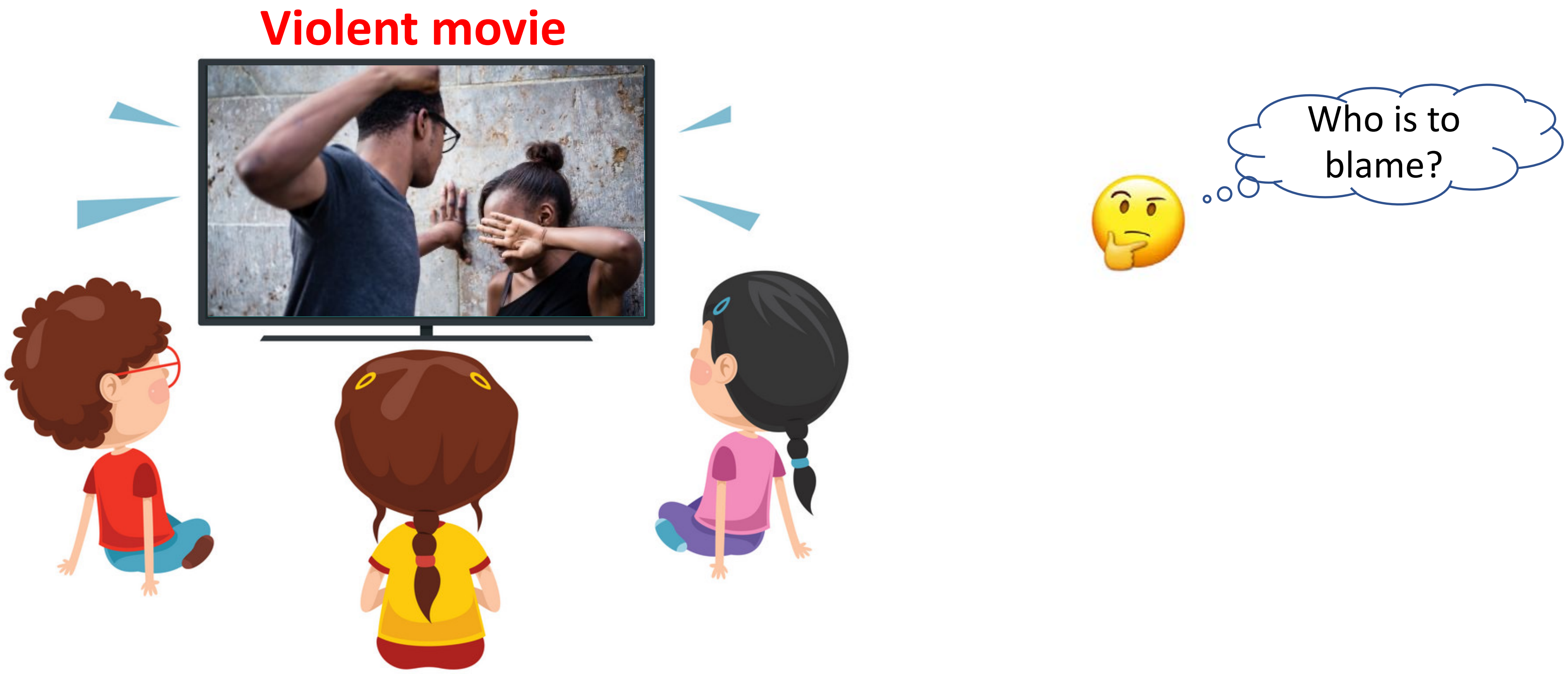
GPU Power Consumption Comparison

Dataset	XDL	DLRM	FAE
Criteo Kaggle	61.83W	58.91W	55.81W
Alibaba	56.39W	60.21W	56.62W
Criteo Terabyte	59.71W	62.47W	57.03W
Avazu	60.2W	58.03W	56.4W

Estimated carbon emissions from training common recommendation models

Accelerating recommendation system training by leveraging popular choices, VLDB, 2021.

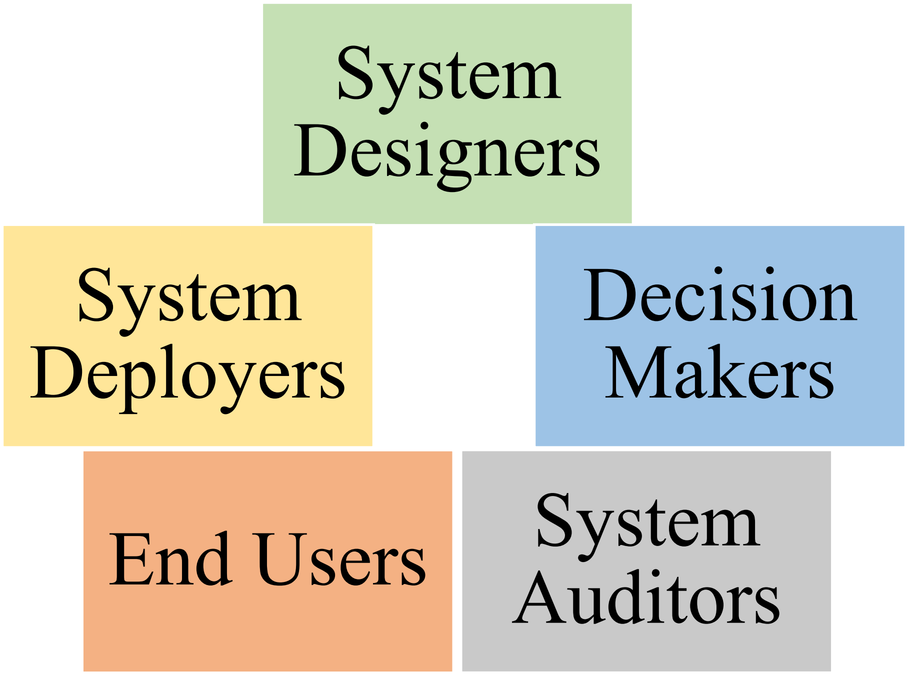
Auditability & Accountability



A clear responsibility distribution, which focuses on who should take the responsibility for what impact of recommender systems.

Auditability & Accountability

- Five roles in Recommender Systems



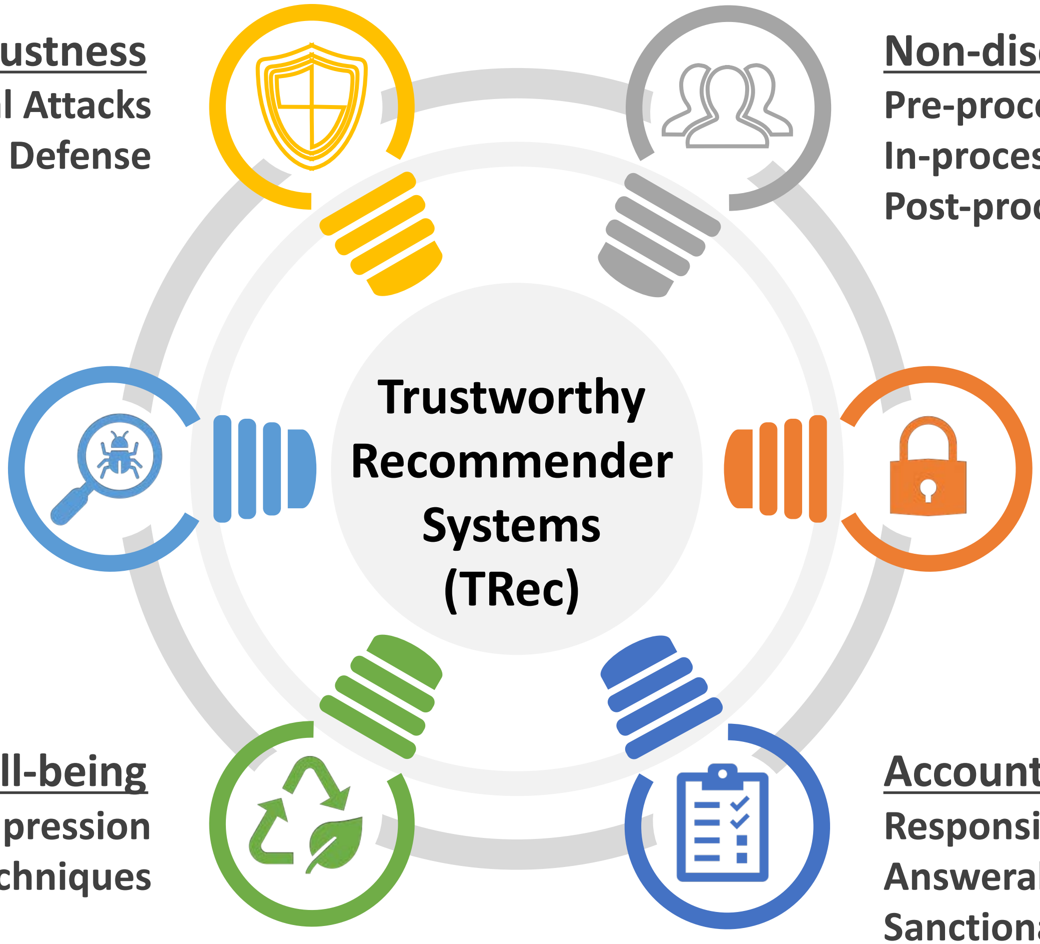
It is necessary to determine the roles and the corresponding responsibility of different parties in the function of a recommender system.

Interactions Among Different Dimensions

How do these SIX dimensions influence each other?

There exist both **accordance** and the **conflicts** among the six dimensions.

Trustworthy Recommender Systems



Safety & Robustness
Adversarial Attacks
Defense

Non-discrimination & Fairness
Pre-processing
In-processing
Post-processing

Explainability
Model-intrinsic & Post-hoc
(Un-)structured Explanations

Privacy
Privacy Attacks
Privacy-preserving

Environmental Well-being
Model Compression
Acceleration Techniques

Accountability & Auditability
Responsibility
Answerability
Sanctionability

"A Comprehensive Survey on Trustworthy Recommender Systems", arXiv:2209.10117, 2022.

A Survey on The Computational Perspective

A Comprehensive Survey on Trustworthy Recommender Systems

WENQI FAN, The Hong Kong Polytechnic University, Hong Kong
XIANGYU ZHAO*, City University of Hong Kong, Hong Kong
XIAO CHEN, The Hong Kong Polytechnic University, Hong Kong
JINGRAN SU, The Hong Kong Polytechnic University, Hong Kong
JINGTONG GAO, City University of Hong Kong, Hong Kong
LIN WANG, The Hong Kong Polytechnic University, Hong Kong
QIDONG LIU, City University of Hong Kong, Hong Kong
YIQI WANG, Michigan State University, USA
HAN XU, Michigan State University, USA
LEI CHEN, The Hong Kong University of Science and Technology, Hong Kong
QING LI, The Hong Kong Polytechnic University, Hong Kong

https://arxiv.org/abs/2209.10117

KDD'2023 Tutorial Website (Slides)

https://advanced-recommender-systems.github.io/trustworthy-recommendations/

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

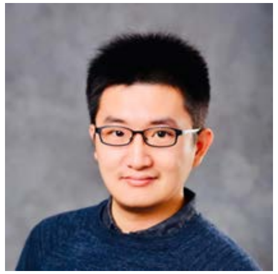
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

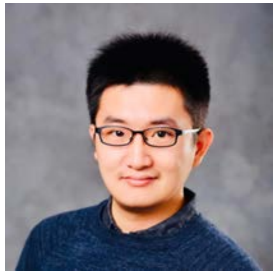
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Contents

CONCEPTS AND TAXONOMY

METHODOLOGY

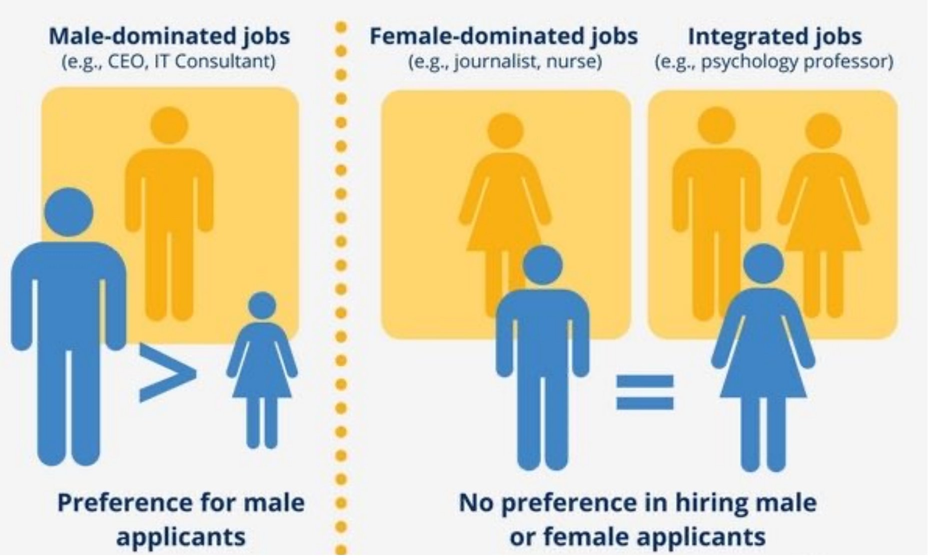
APPLICATIONS

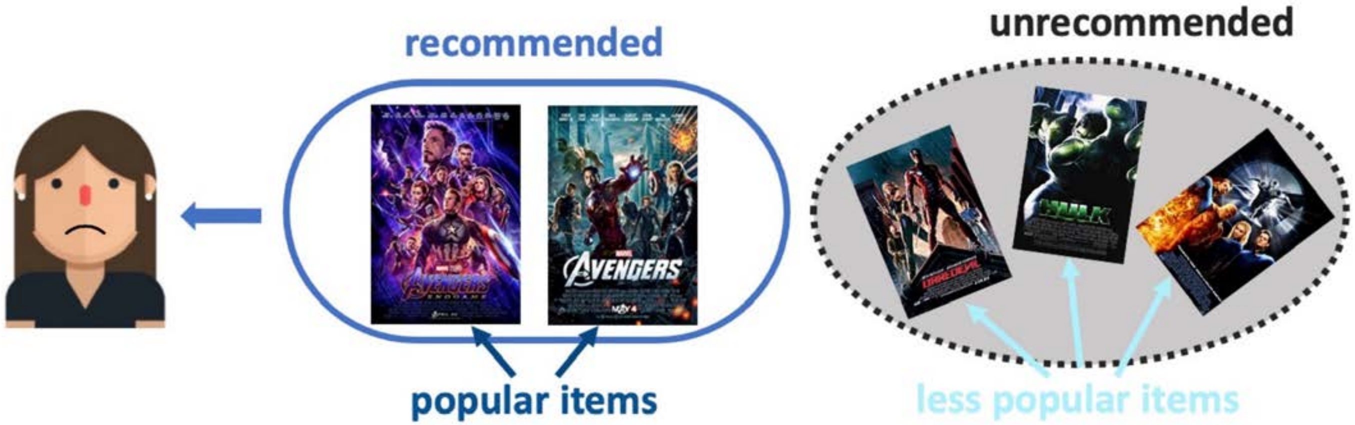
SURVEYS AND TOOLS

FUTURE DIRECTIONS

Potential discrimination and bias in RecSys

- Recommender Systems make unfair decisions for specific user/item groups





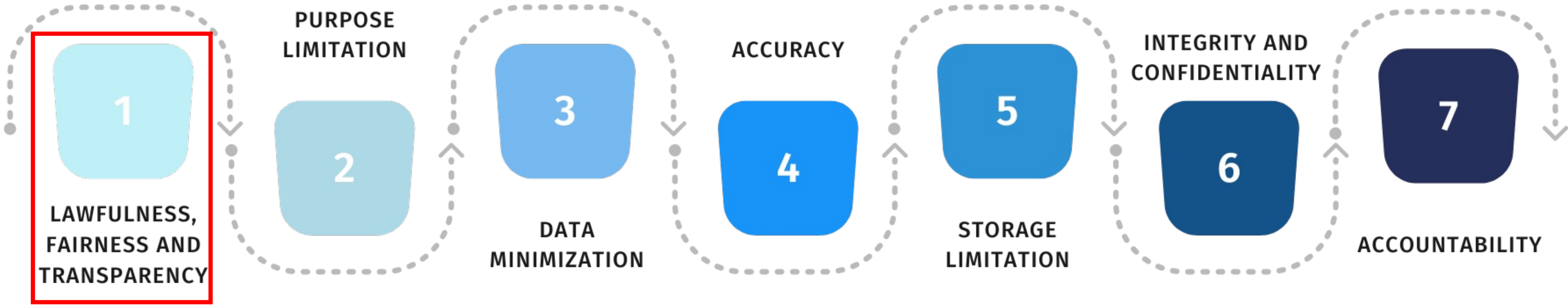
Gender Discriminatory Bias [1]

Popularity Bias [2]

[1] Lambrecht, et al. "Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of STEM career ads." 2019.
[2] Abdollahpouri, et al. "Popularity bias in ranking and recommendation." 2019.

Why Need Fairness in RecSys: From the Ethics Perspective

- 7 principles of EU GDPR regulation



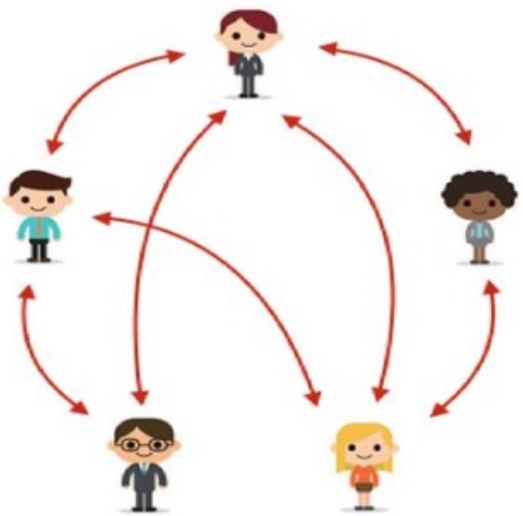
Fairness often couples with other responsible AI perspectives (e.g., explainability).

https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/principles/lawfulness-fairness-and-transparency

Why Need Fairness in RecSys: From the Utility Perspective

- Fair exposure opportunity guarantees the sustainable development of the RecSys platform

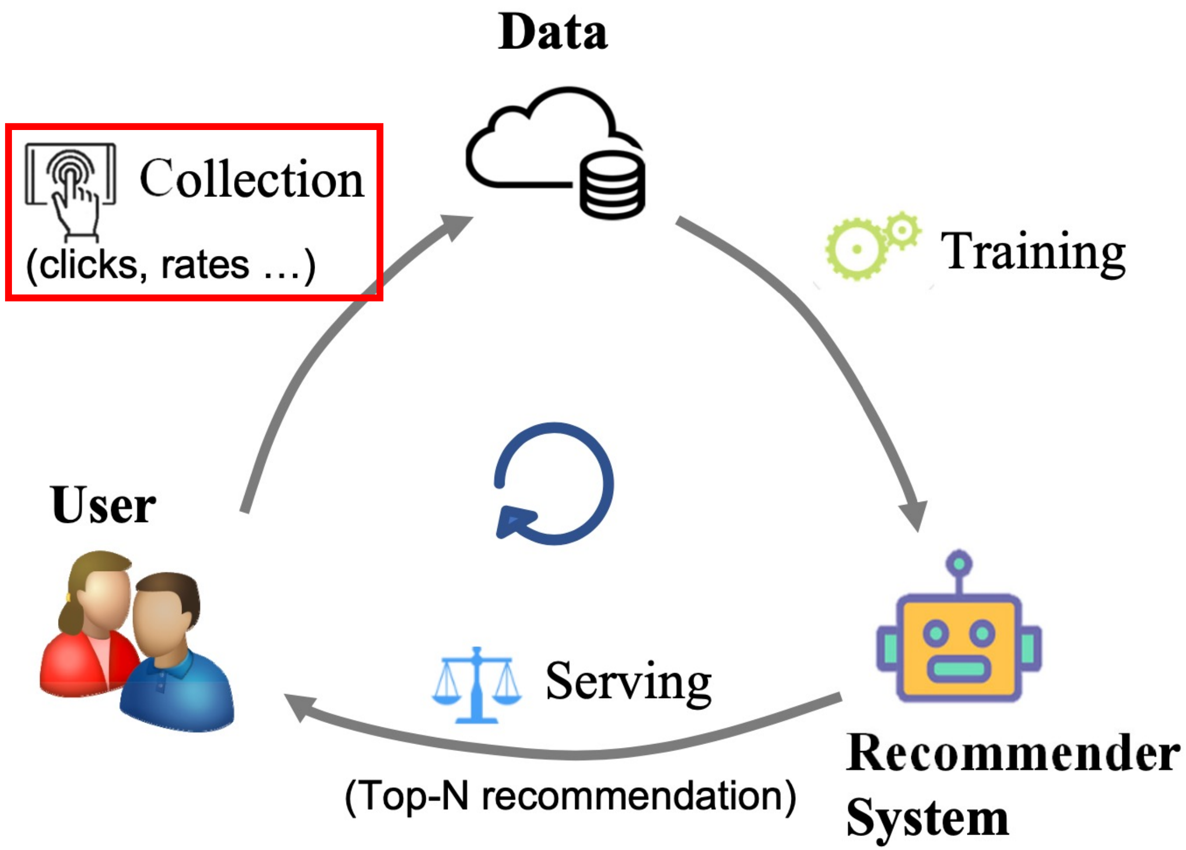
Big retailors vs. Small retailors
in the e-commerce system



Star accounts vs. Grassroot accounts
in the social recommendation system

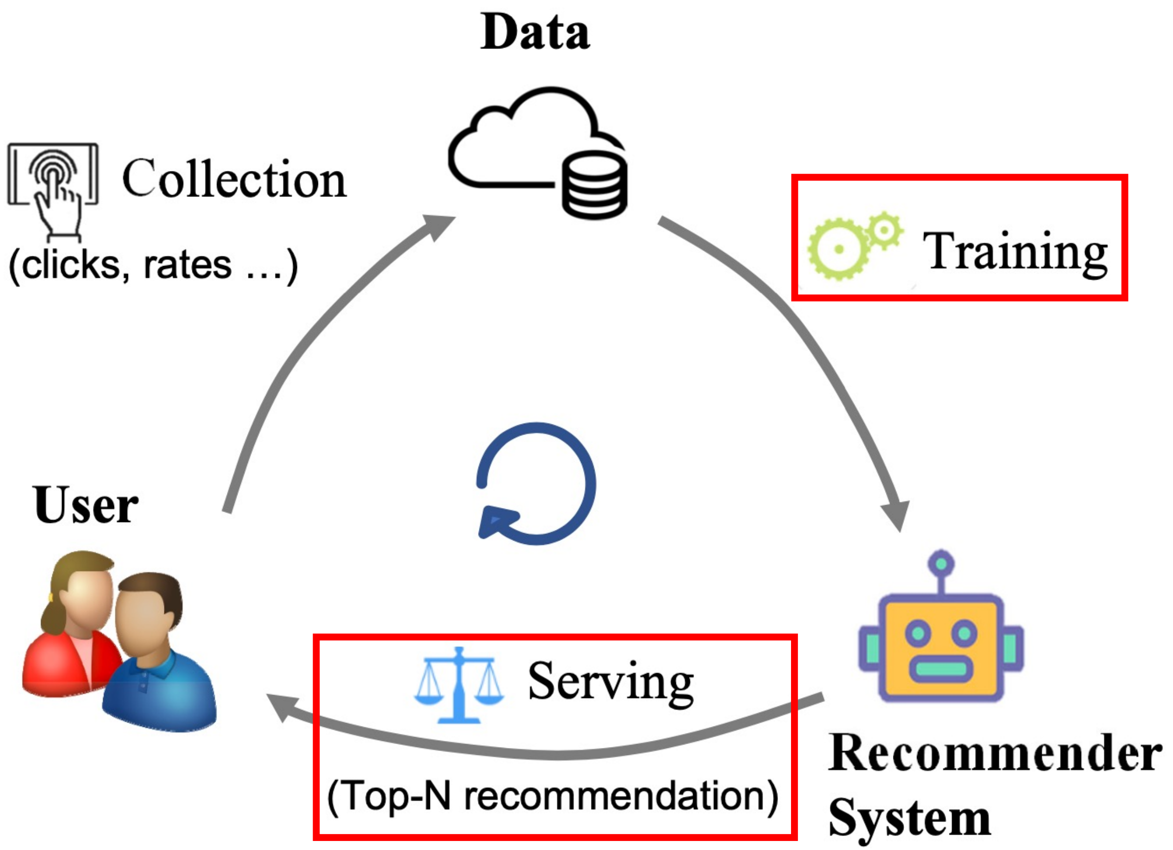
Sources of Bias

- **Data bias**
  - **Selection Bias:**
    selecting rating behavior of users
  - **Exposure Bias:**
    unobserved interactions may not fully represent the disliked items of users
  - **Conformity Bias:**
    users behave similarly to other group members
  - **Position Bias:**
    the higher positions on a recommendation list tends to receive more interaction



Sources of Bias

- Data bias
  - Selection Bias
  - Exposure Bias
  - Conformity Bias
  - Position Bias
- **Model and result bias**
  - **Popularity Bias:**
    popular items are over-recommended compared to what their popularity warrant



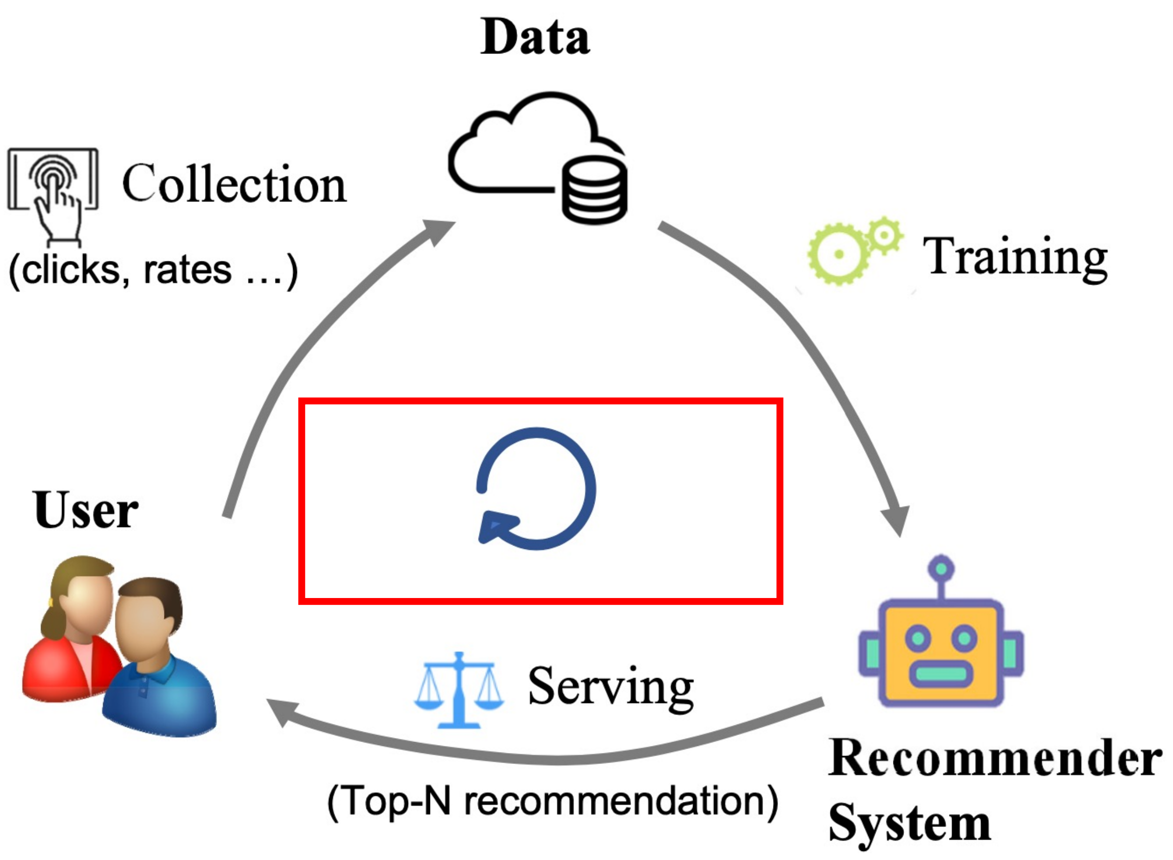
Chen, et al. "Bias and debias in recommender system: A survey and future directions." *TOIS* 2023.

Sources of Bias

- Data bias
  - Selection Bias
  - Exposure Bias
  - Conformity Bias
  - Position Bias
- Model and result bias
  - Popularity Bias
- **Feedback loop bias**
  - **Reinforced RS Feedback Loop Bias:**

    Unfair recommendations would influence users' behaviors in the online serving process

    Biased user behavior data enlarges model discrimination



Fairness Definition

- **Procedural Fairness:** procedural justice in decision-making processes
- **Outcome Fairness:** fair outcome performance

User Fairness vs. Item Fairness

Group Fairness vs. Individual Fairness

Causal Fairness vs. Associative Fairness

Static Fairness vs. Dynamic Fairness

Fairness Evaluation Metrics

- **Absolute Difference (AD):** group-wise utility difference

$$AD = |u(G_0) - u(G_1)|$$

- **Variance:** performance dispersion at the group/individual-level

$$\text{Variance} = \frac{1}{|\mathcal{V}|^2} \sum_{v_i \neq v_j} \left(u(v_i) - u(v_j)\right)^2$$

- **Min-Max Difference:** the difference between the maximum and the minimum score value of all allocated utilities

- **Entropy**
- **KL-Divergence …**

Contents

CONCEPTS AND TAXONOMY

METHODOLOGY

APPLICATIONS

SURVEYS AND TOOLS

FUTURE DIRECTIONS

Method category

Pre-processing	In-processing	Post-processing
Transform the data to remove the data bias before training	Modify the learning algorithms to remove discrimination during the model training process	Perform post-processing by evaluating a holdout set that was not involved during model training

Pre-processing methods

- **Resampling**

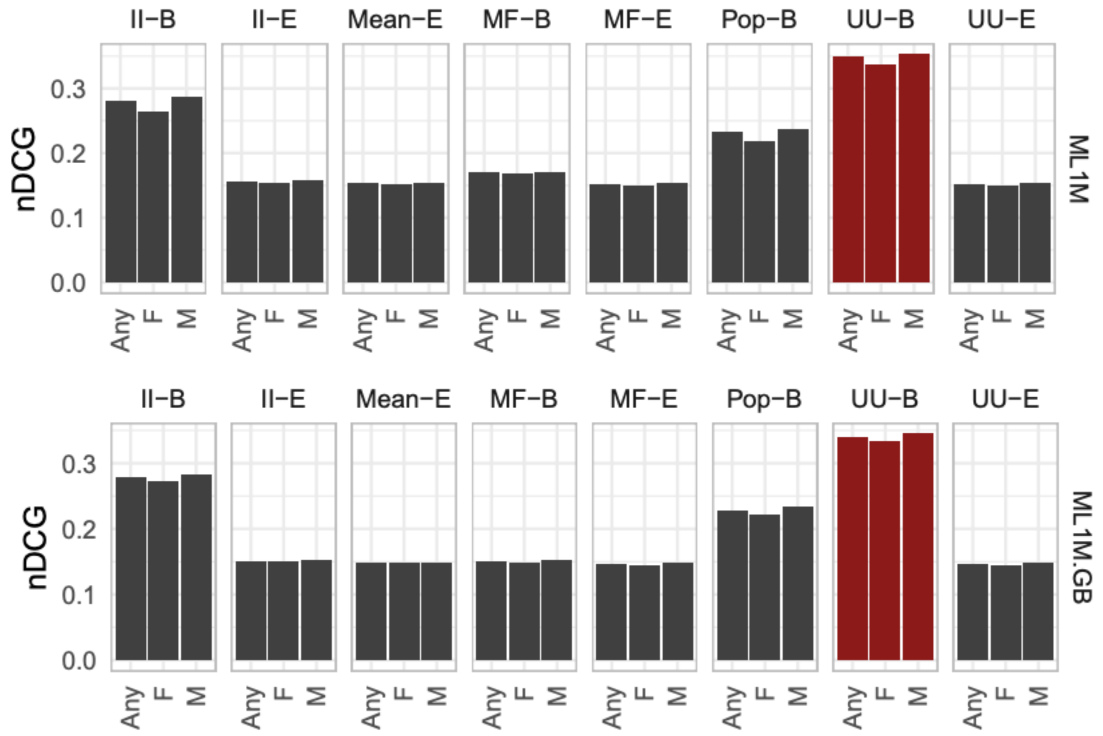
  Rebalance the dataset distribution w.r.t the sensitive attribute

- **Data Augmentation**

  Generating additional data for promoting the fairness of recommender systems

Pre-processing method (Resampling)

Idea: Different demographic groups obtain different utilities due to imbalanced data distribution. Balance the ratio of various user groups via a re-sampling strategy.



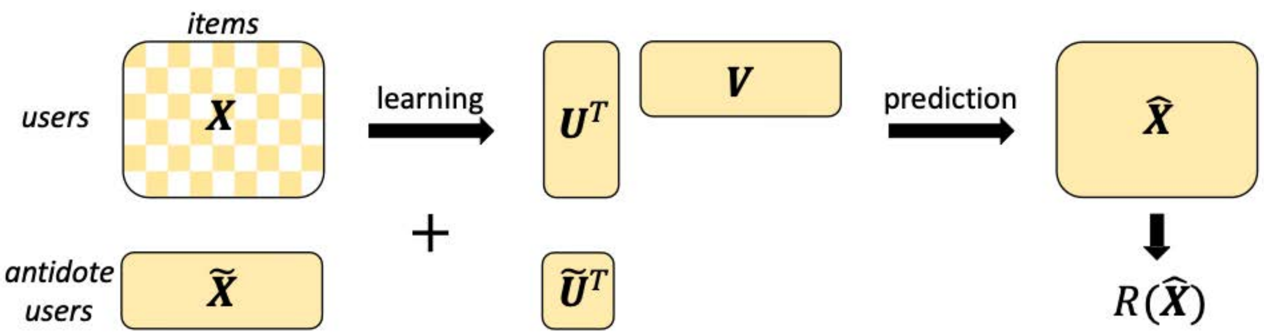
statistically-significant differences between gender groups

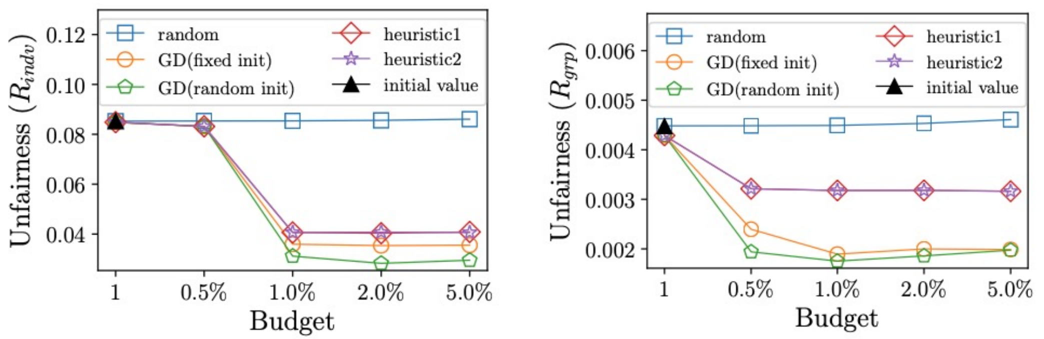
results on gender-balanced dataset

All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. ICFAT 2018.

Pre-processing method (Adding Antidote Data)

Idea: Improving the social desirability of recommender system outputs by adding more "antidote" data to the input.





(a) Individual fairness (b) Group fairness

Matrix Factorization: $$\arg\min_{\mathbf{U},\mathbf{V}} \ ||P_\Omega(\mathbf{X} - \mathbf{U}^\intercal\mathbf{V})||_F^2 + \lambda(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2)$$

Objectives: $$\arg\min_{\tilde{\mathbf{X}}\in\mathbb{M}} \ R(\hat{\mathbf{X}}(\Theta(\mathbf{X};\tilde{\mathbf{X}})))$$

fairness objective *antidote data*

Fighting Fire with Fire: Using Antidote Data to Improve Polarization and Fairness of Recommender Systems. WSDM 19

Summary of Pre-processing methods

Flexibility, decoupled with the recommender systems

Performance gains might be degraded by the following steps

In-processing method

- **Regularization and constrained optimization**
- **Adversary Learning**
- **Causal graph**
- **Reinforcement Learning**
- **Others**

In-processing method (Regularization)

Idea: propose four new metrics that address different forms of unfairness. These metrics can be optimized by adding fairness terms to the learning objective [1].

$$U_{abs} = \frac{1}{n}\sum_{i=1}^{n}\left| |E_{adv}[y]_i - E_{adv}[r]_i| - |E_{\neg adv}[y]_i - E_{\neg adv}[r]_i| \right|,$$

$$\min_{\boldsymbol{P},\boldsymbol{Q},\boldsymbol{u},\boldsymbol{v}} J(\boldsymbol{P},\boldsymbol{Q},\boldsymbol{u},\boldsymbol{v}) + U \,.$$

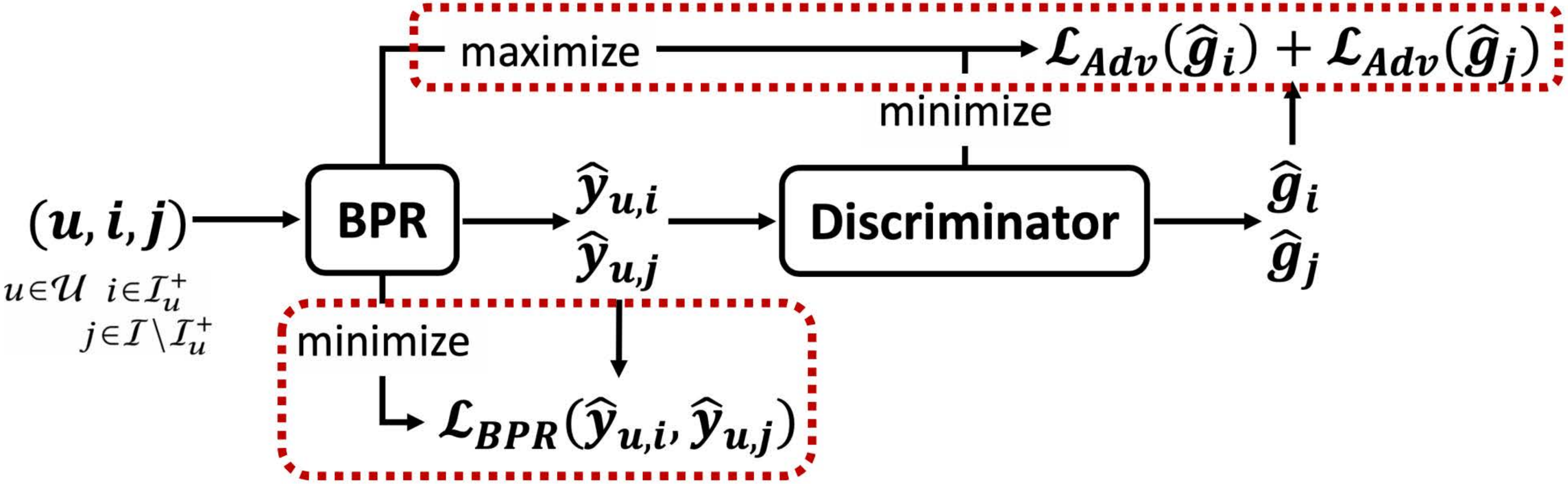
Idea: a novel pairwise regularizer for pairwise ranking fairness [2].

$$\min_{\theta}\left(\sum_{(\mathbf{q},j,y,z)\in\mathcal{D}} \mathcal{L}_{rec}\left(f_\theta\left(\mathbf{q},\mathbf{v}_j\right),(y,z)\right)\right) + |\mathrm{Corr}_{\mathcal{P}}(A,B)|,$$

[1] Beyond Parity: Fairness Objectives for Collaborative Filtering. NeurIPS17
[2] Fairness in recommendation ranking through pairwise comparisons. KDD19

In-processing method (Adversary Learning)

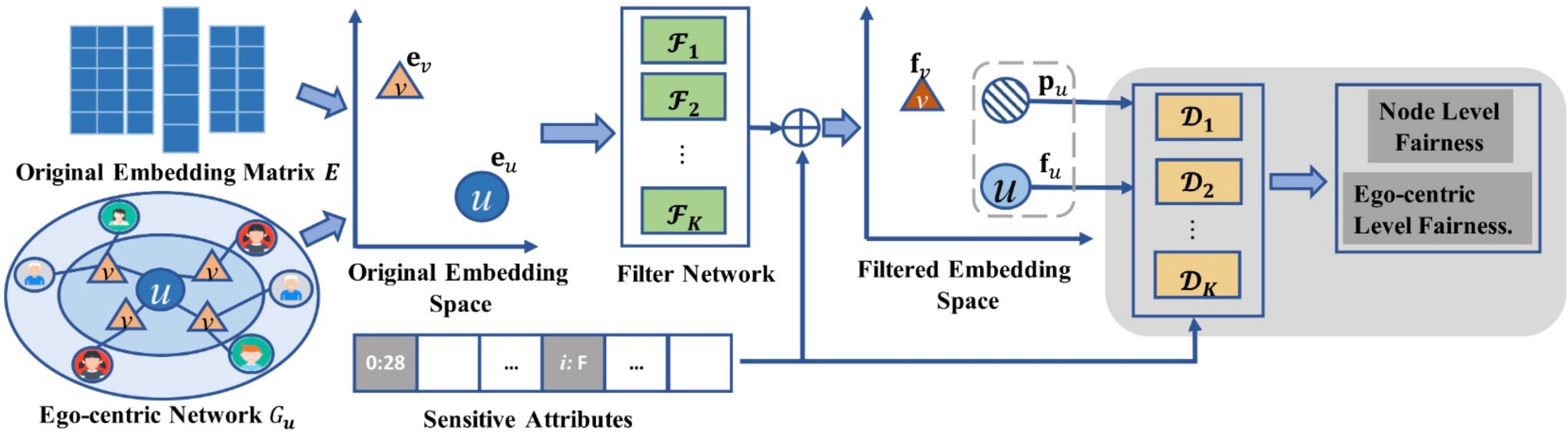
Idea: normalize the score distribution for each user to align predicted score with ranking position.
decouple the predicted score with the group attribute.



Measuring and Mitigating Item Under-Recommendation Bias in Personalized Ranking Systems. SIGIR20

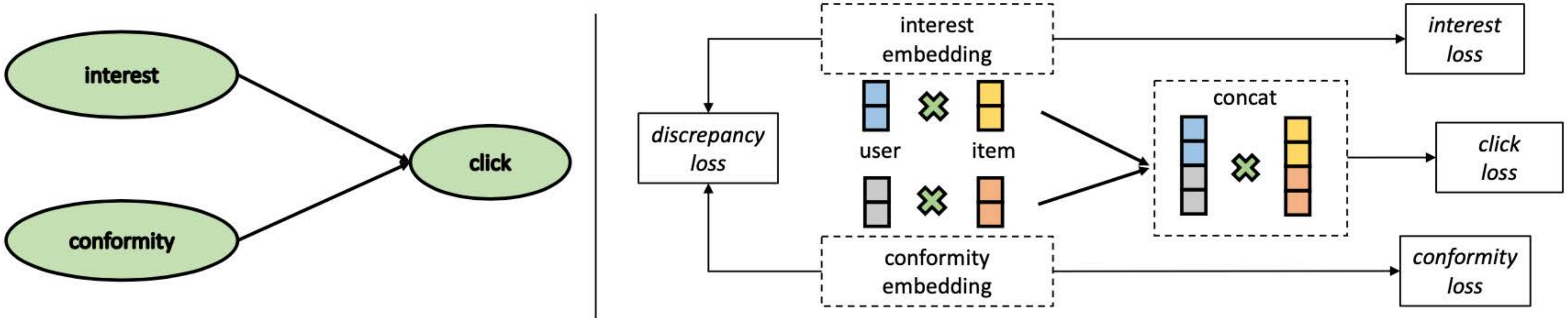
In-processing method (Adversary Learning)

Idea: propose a graph-based perspective for fairness-aware representation learning of any recommendation models. Adversarial learning of a user-centric graph.



In-processing method (Causal Graph)

Idea: Disentangling Interest and Conformity with Causal Embedding (DICE).
Separate embeddings are adopted to capture the two causes, and are trained with cause-specific data.

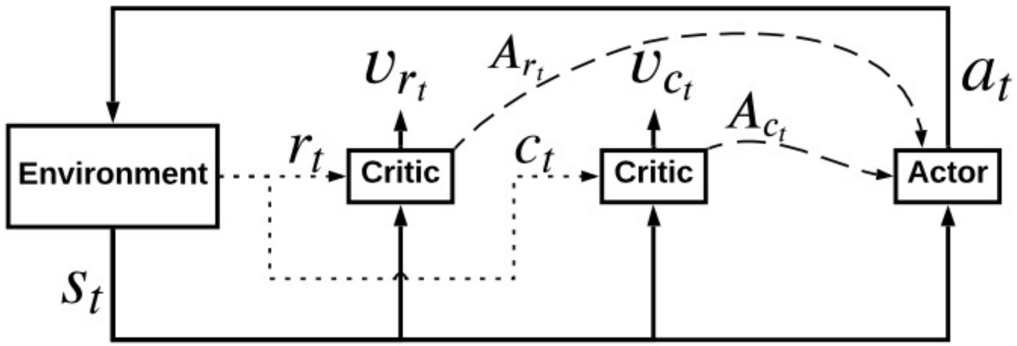


(a) Causal Graph (b) Causal Embedding

Disentangling user interest and conformity for recommendation with causal embedding. WWW21.

In-processing method (Reinforcement Learning)

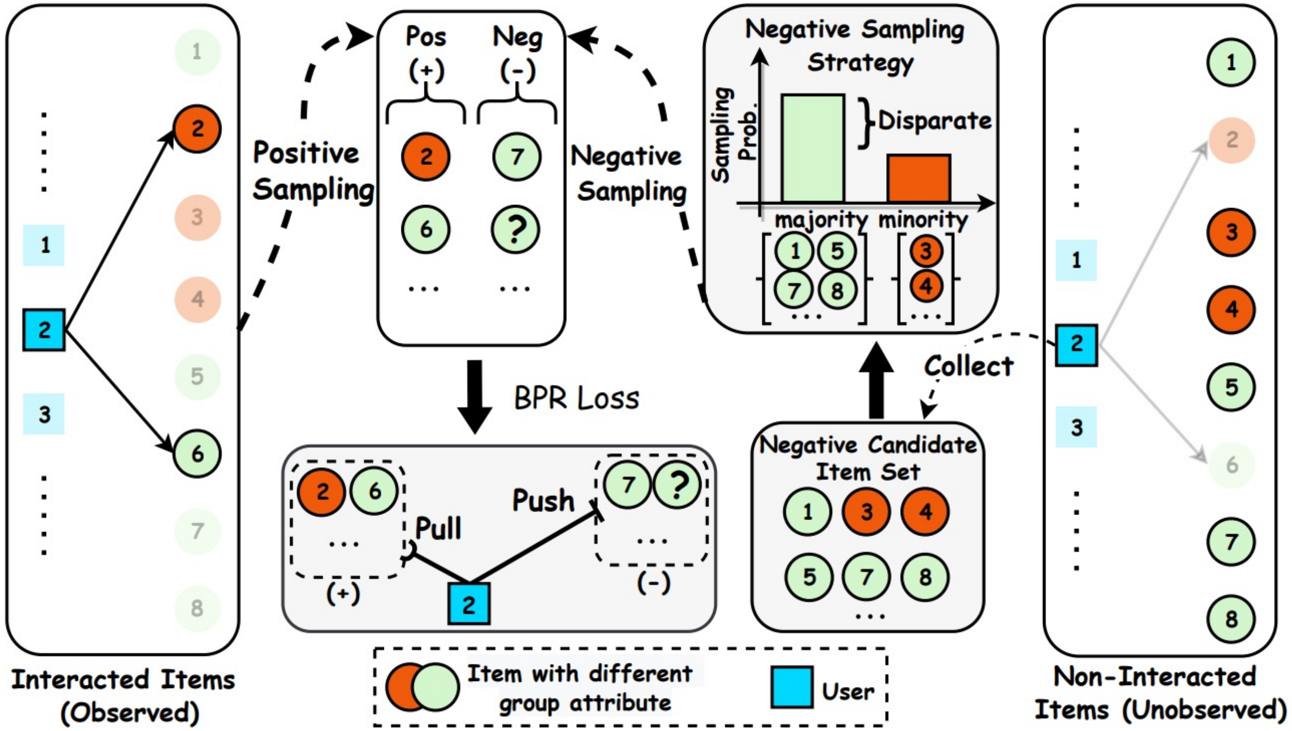
Idea: propose a fairness-constrained reinforcement learning algorithm, which models the recommendation problem as a Constrained Markov Decision Process (CMDP). Dynamically adjust the recommendation policy for the fairness requirement.



Towards Long-term Fairness in Recommendation. WSDM21.

In-processing method (Negative Sampling)

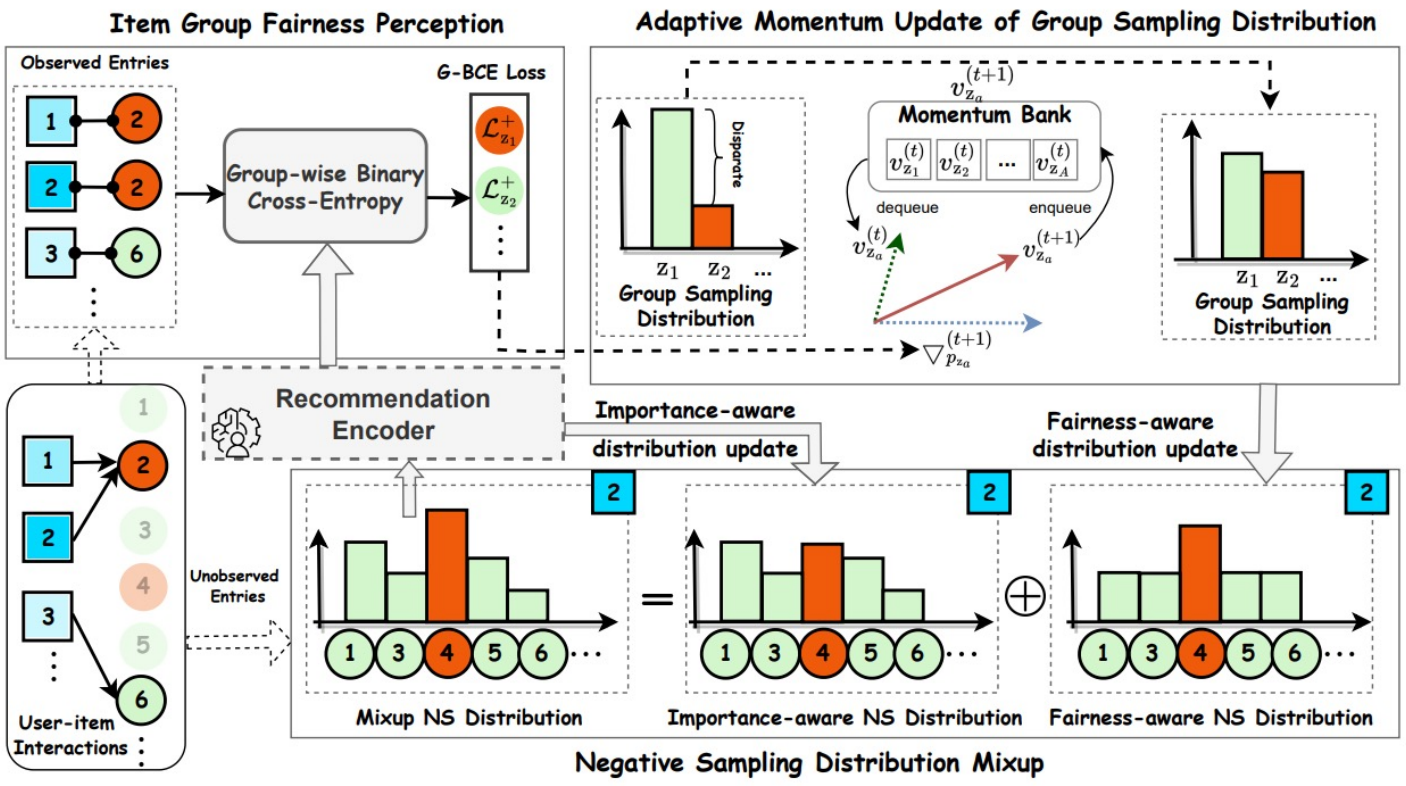
- **Observation:** the majority item group obtains low (biased) prediction scores via the BPR loss (group-wise performance disparity)



Fairly Adaptive Negative Sampling for recommendations. WWW 23

In-processing method (Negative Sampling)

- **Idea:** adjust the negative sampling distribution (group-wise) adaptively in the training process for meeting the item group fairness objective



Fairly Adaptive Negative Sampling for recommendations. WWW 23

In-processing method (Negative Sampling)

- Bi-level Optimization of FairNeg

The optimization of the group-wise negative sampling distribution is nested within the recommendation model parameters optimization

$$\boldsymbol{p}^* = \arg\min_{\boldsymbol{p}} \mathcal{L}_{\text{Recall-Disp}}(\Theta_{\boldsymbol{p}}) := \sum_{\mathrm{z_a} \in Z} \left| \mathcal{L}^+_{\mathrm{z}_a} - \frac{1}{|A|} \Sigma_{\mathrm{z} \in Z} \mathcal{L}^+_{\mathrm{z}} \right|,$$

$$\Theta^*_{\boldsymbol{p}} = \arg\min_{\Theta} \mathcal{L}_{\text{utility}}(\Theta, \boldsymbol{p}) := -\sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{V}^+_u, j \in \mathcal{V}^-_u} \mathcal{L}_{\text{BPR}}(\boldsymbol{u}, \boldsymbol{i}, \boldsymbol{j}; \Theta, \boldsymbol{p}),$$

- Updating Group Sampling Distribution

(1) Group-wise gradient calculation

$$\nabla^{(t)}_{p_{\mathrm{z}_a}} := \mathcal{L}^{+(t)}_{\mathrm{z}_a} - \frac{1}{|A|} \Sigma_{\mathrm{z} \in Z} \mathcal{L}^{+(t)}_{\mathrm{z}},$$

(2) Adaptive momentum update

$$v^{(t+1)}_{\mathrm{z}_a} = \gamma v^{(t)}_{\mathrm{z}_a} + \alpha \cdot \nabla^{(t+1)}_{p_{\mathrm{z}_a}},$$

$$p^{(t+1)}_{\mathrm{z}_a} = p^{(t)}_{\mathrm{z}_a} - v^{(t+1)}_{\mathrm{z}_a},$$

Fairly Adaptive Negative Sampling for recommendations. WWW 23

Summary of In-processing methods

Substantial fairness improvements

Fairness and utility trade-off

Resource-intensive

Post-processing method

- **Slot-wise reranking**

- **Global-wise reranking**

- **User-wise reranking**

Slot-wise Re-ranking

Idea: propose a personalized re-ranking algorithm to achieve a fair microlending RS.

A combination of personalization score and a fairness term.

$$\max_{v \in R(u)} \underbrace{(1-\lambda)P(v \mid u)}_{\text{personalization}} + \underbrace{\lambda \sum_{c} P(\mathcal{V}_c) \mathbb{1}_{\{v \in \mathcal{V}_c\}} \prod_{i \in S(u)} \mathbb{1}_{\{i \notin \mathcal{V}_c\}}}_{\text{fairness}},$$

Personalized Fairness-aware Re-ranking for Microlending. RecSys 19

User-wise Re-ranking

Idea: formulate fairness constraints on rankings in terms of exposure allocation. Find rankings that maximize the utility for the user while provably satisfying a specific notion of fairness.

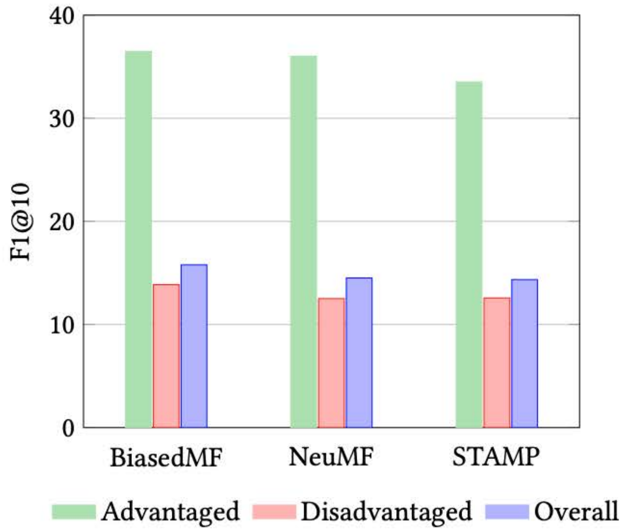
$$
\begin{aligned}
\mathbf{P} = \operatorname{argmax}_{\mathbf{P}} \quad & \mathbf{u}^T \mathbf{P} \mathbf{v} && \text{(expected utility)} \\
\text{s.t.} \quad & \mathbb{1}^T \mathbf{P} = \mathbb{1}^T && \text{(sum of probabilities for each position)} \\
& \mathbf{P}\mathbb{1} = \mathbb{1} && \text{(sum of probabilities for each document)} \\
& 0 \leq \mathbf{P}_{i,j} \leq 1 && \text{(valid probability)} \\
& \mathbf{P} \text{ is fair} && \text{(fairness constraints)}
\end{aligned}
$$

$$
\begin{aligned}
& \text{Exposure}(G_0|\mathbf{P}) = \text{Exposure}(G_1|\mathbf{P}) && (4) \\
\Leftrightarrow \; & \frac{1}{|G_0|} \sum_{d_i \in G_0} \sum_{j=1}^{N} \mathbf{P}_{i,j} \mathbf{v}_j = \frac{1}{|G_1|} \sum_{d_i \in G_1} \sum_{j=1}^{N} \mathbf{P}_{i,j} \mathbf{v}_j && (5) \\
\Leftrightarrow \; & \sum_{d_i \in \mathcal{D}} \sum_{j=1}^{N} \left(\frac{\mathbb{1}_{d_i \in G_0}}{|G_0|} - \frac{\mathbb{1}_{d_i \in G_1}}{|G_1|} \right) \mathbf{P}_{i,j} \mathbf{v}_j = 0 && (6) \\
\Leftrightarrow \; & \mathbf{f}^T P \mathbf{v} = 0 && \left(\text{with } \mathbf{f}_i = \frac{\mathbb{1}_{d_i \in G_0}}{|G_0|} - \frac{\mathbb{1}_{d_i \in G_1}}{|G_1|}\right)
\end{aligned}
$$

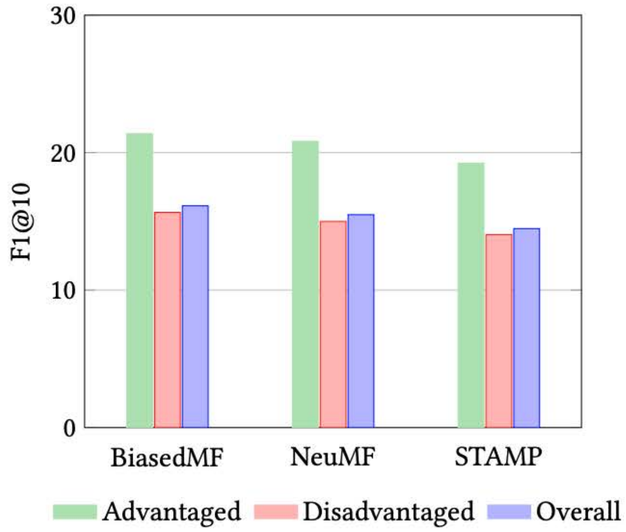
Fairness of Exposure in Rankings. KDD 18

Global-wise Re-ranking

Idea: a re-ranking approach to mitigate this unfairness problem by adding constraints over evaluation metrics.



(a) Original



(b) Fair Method

$$\max_{\mathbf{W}_{ij}} \sum_{i=1}^{n} \sum_{j=1}^{N} \mathbf{W}_{ij} \mathrm{S}_{i,j}$$

$$\text{s.t.} \quad \mathrm{UGF}\left(Z_1, Z_2, \mathbf{W}\right) < \varepsilon$$

$$\sum_{j=1}^{N} \mathbf{W}_{ij} = K, \mathbf{W}_{ij} \in \{0, 1\}$$

Summary of Post-processing methods

Can be applied to any recommendation systems

Constrained to unfair recommendation model outputs

• Summary of existing methods

Taxonomy	**Method type**	Related research
Pre-processing	Data Re-sampling	[95]
	Adding Antidote Data	[289]
In-processing	Regularization & Constrained Optimization	[26, 351, 393, 409, 461]
	Adversarial Learning	[33, 207, 215, 221, 285, 379, 380]
	Reinforcement Learning	[120, 122, 244]
	Causal Graph	[121, 162, 387, 452]
	Others	[31, 110, 167, 224]
Post-processing	Slot-wise Re-ranking	[124, 185, 189, 243, 262, 300, 305] [306, 323, 328, 405, 419]
	User-wise Re-ranking	[28, 253, 304, 318]
	Global-wise Re-ranking	[87, 114, 219, 250, 279, 335, 384, 462]

A Comprehensive Survey on Trustworthy Recommender Systems. Arxiv 22

Contents

CONCEPTS AND TAXONOMY

METHODOLOGY

APPLICATIONS

SURVEYS AND TOOLS

FUTURE DIRECTIONS

Applications

- **Ecommerce (Amazon, Etsy)**
- **Social Media (Twitter, LinkedIn)**
- **Content Streaming (Spotify, Youtube)**
- **Ride-hailing (Uber, Lyft)**



Contents

CONCEPTS AND TAXONOMY

METHODOLOGY

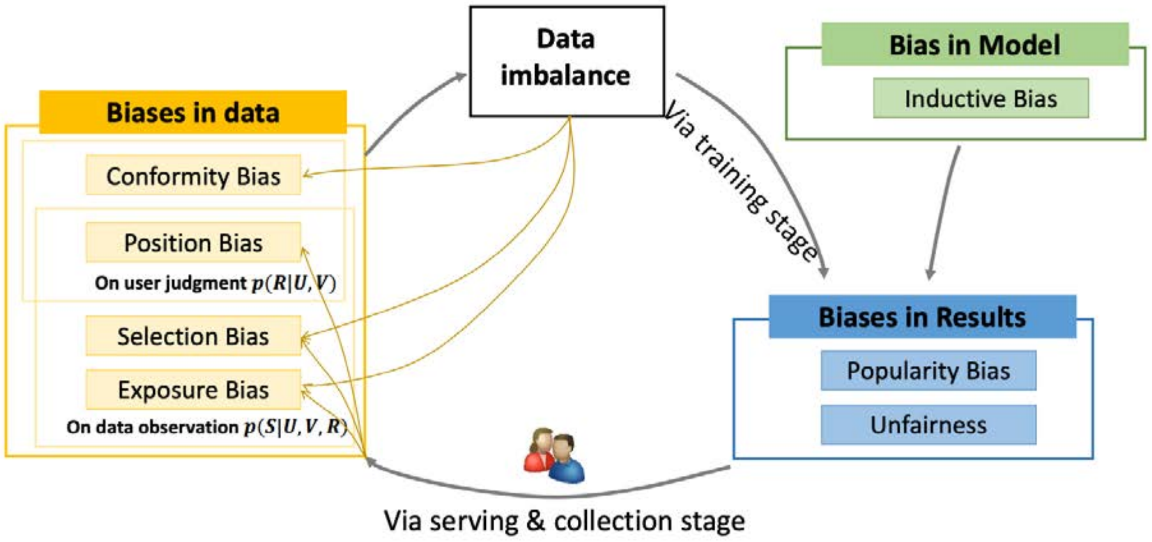
APPLICATIONS

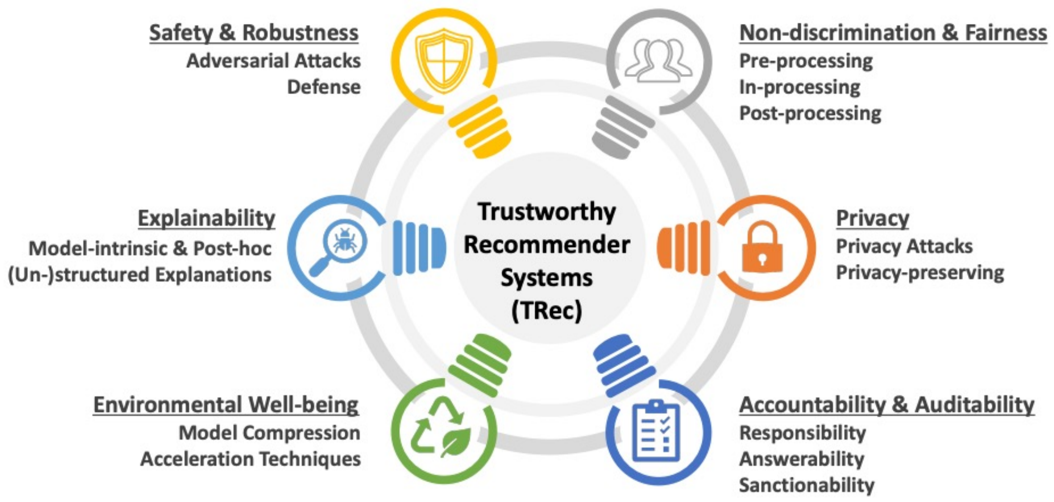
SURVEYS AND TOOLS

FUTURE DIRECTIONS

Surveys

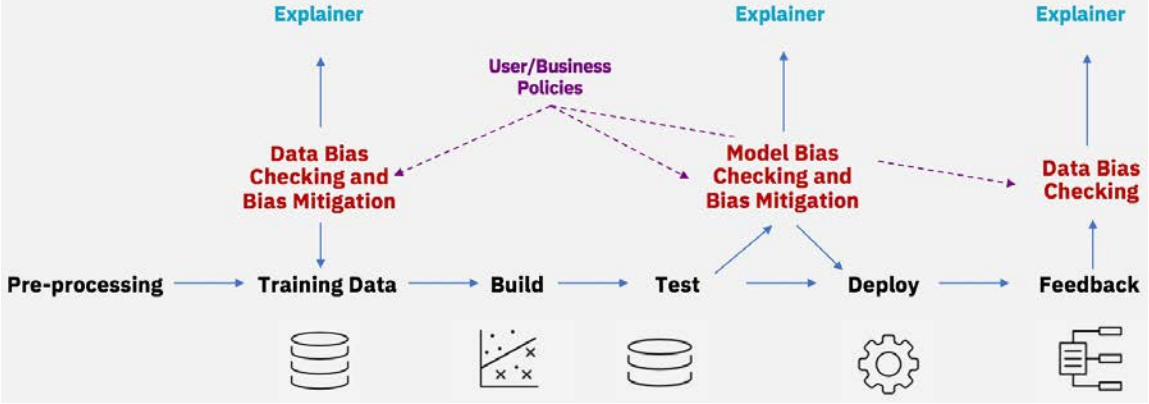
- TOIS 23’ Bias and Debias in Recommender System: A Survey and Future Directions
- TOIS 23 ’ Fairness in Recommendation: Foundations, Methods and Applications
- Arxiv 22’ A Comprehensive Survey on Trustworthy Recommender Systems



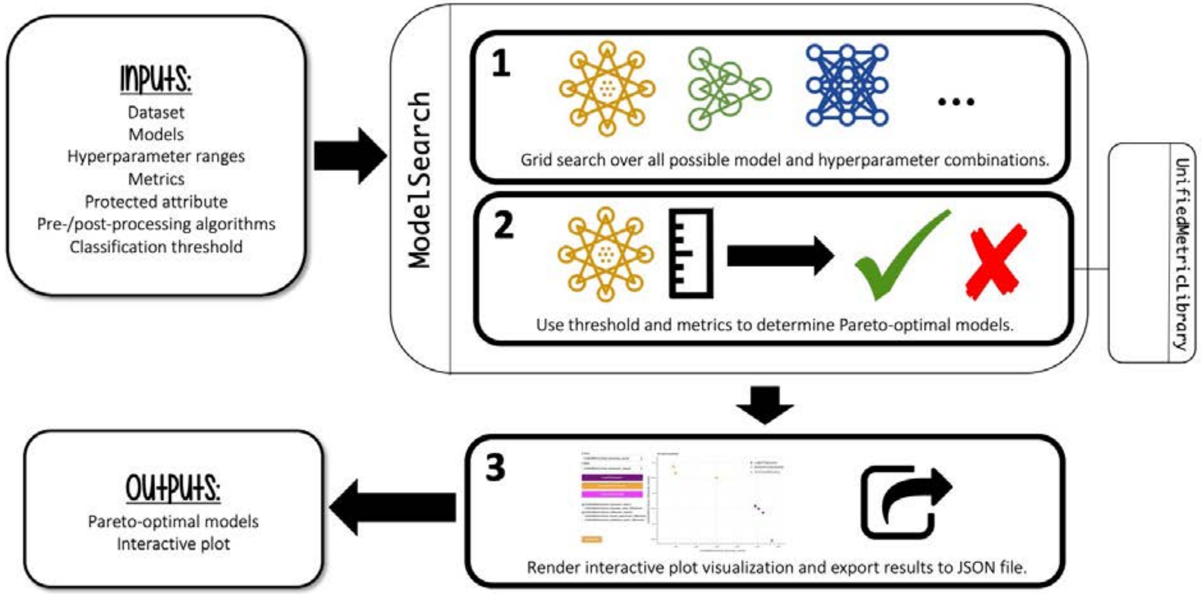


Tools

- **IBM Fairness 360**



- **Fairkit-learn**



Contents

CONCEPTS AND TAXONOMY

METHODOLOGY

APPLICATIONS

SURVEYS AND TOOLS

FUTURE DIRECTIONS

Future Directions

- **Consensus on Fairness Definition**
- **Fairness-Utility tradeoff**
- **Fairness-aware algorithm design**
- **Better evaluation metrics**

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

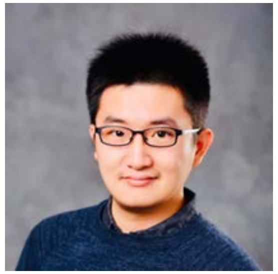
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Real World Attacks in Recommender Systems

DIGITAL LIVING | JULY 26, 2022

Amazon's War on Fake Reviews

By Matt Stieb, Intelligencer staff writer

Photo-Illustration: Intelligencer; Photos: Getty Images/Amazon

BUSINESS

How merchants use Facebook to flood Amazon with fake reviews

By Elizabeth Dwoskin and Craig Timberg
April 23, 2018 at 1:26 p.m. EDT

An Amazon distribution center in Madrid, shown in November. (Emilion Naranjo/EPA-EFE/Shutterstock)

https://nymag.com/intelligencer/2022/07/amazon-fake-reviews-can-they-be-stopped.html
https://www.washingtonpost.com/business/economy/how-merchants-secretly-use-facebook-to-flood-amazon-with-fake-reviews/2018/04/23/5dad1e30-4392-11e8-8569-26fda6b404c7_story.html

Safety and Robustness

"A decision aid, no matter how sophisticated or 'intelligent' it may be, may be rejected by a decision maker who does not trust it, and so its potential benefits to system performance will be lost."

—Bonnie M. Muir, psychologist at University of Toronto

Safety and Robustness

By examining Adversarial Robustness,
we expect the recommender system to:

- Be reliable, secure and stable

Outline

Concepts and Taxonomy → Adversarial Attack → Adversarial Defense → Application → Adversarial Learning Surveys and Tools → Future directions

Taxonomy

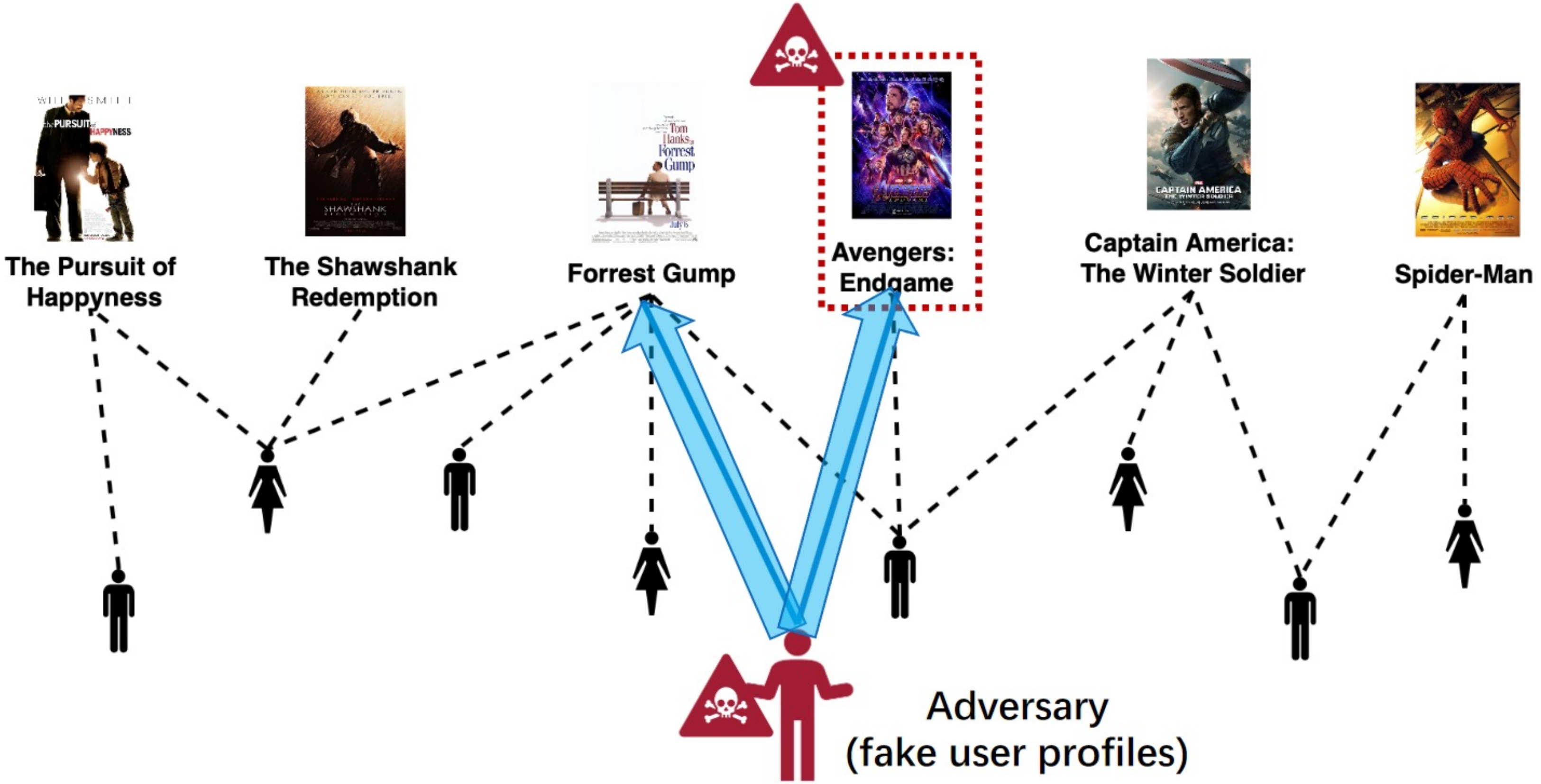
- Adversarial Recommender System
  - Adversarial Attack
    - Different Types
    - Different perturbation
    - Different scenarios
  - Adversarial Defense

Adversarial Attack

- Poisoning Attacks vs. Evasion Attacks
  - They happen in training phase/ happen in test/inference phase
- White-box attacks vs. Grey-box attacks vs. Black-box attacks
  - They have all knowledge of the recommender system / have partial knowledge/ have no knowledge or limit knowledge
- Targeted Attacks vs. Untargeted Attacks
  - They aim to promote/demote a set of target items/ aim to degrade a recommendation system's overall performance

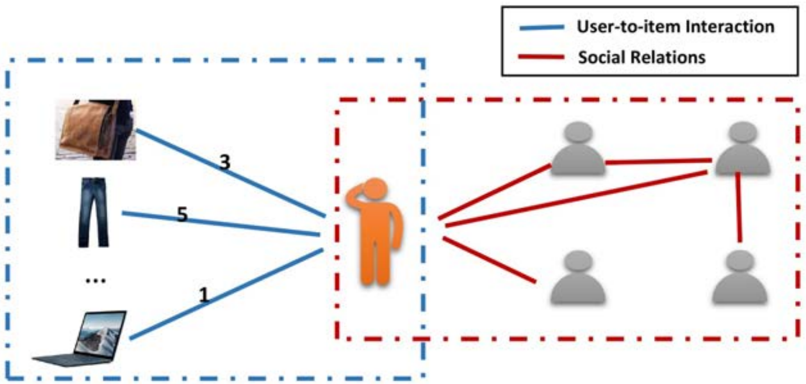
Adversarial in Different Perturbation

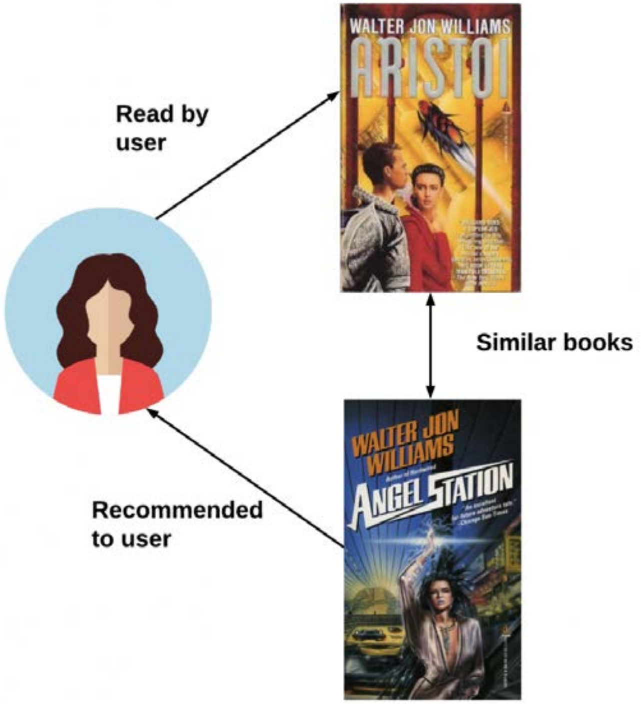
- Adding fake user profiles into user-item interactions, modifying user attributes information, adding social relations, etc

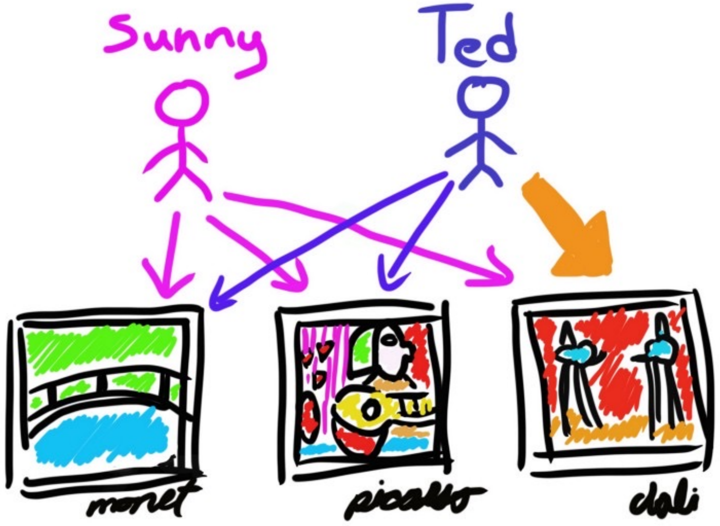


Adversarial in Different Scenarios

- Collaborative Filtering Recommender System
- Social Recommender System
- Content-based Recommender System
- . . .







Graph neural networks for social recommendation, Fan et al 2019.
https://thingsolver.com/introduction-to-recommender-systems/

Adversarial Defenses

- Perturbations Detection vs. Adversarial Training

  - It is to identify perturbations data and remove them/ enhances the robustness of recommender systems

Outline

Concepts and Taxonomy

Adversarial Attack

Adversarial Defense

Application

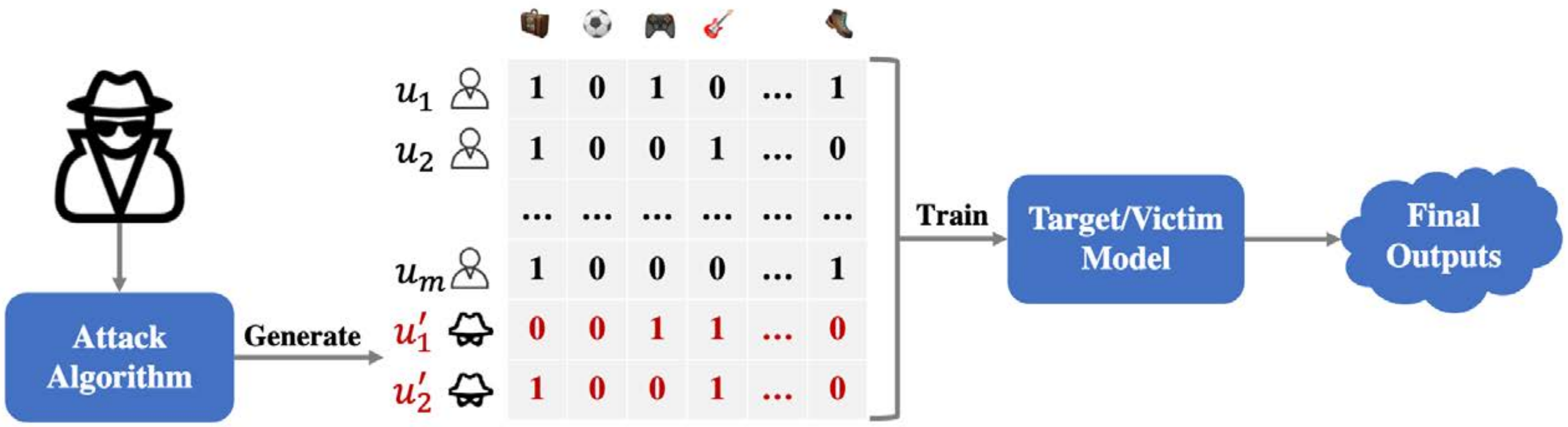
Adversarial Learning Surveys and Tools

Future directions

Adversarial Attack for Recommender System

- A Unified Formulation of Poisoning Attack

$$\min_{\widehat{U}} \mathcal{L}_{adv}(\theta^*), \quad \text{s.t.} \quad \theta^* = \arg\min_{\theta}(\mathcal{L}_{rec}(\boldsymbol{R}, \boldsymbol{O}_\theta) + \mathcal{L}_{rec}(\widehat{\boldsymbol{R}}, \boldsymbol{O}_\theta))$$



Heuristic Attack

- Heuristic Attack Method
  - It assigns high scores to target items
  - Give a low score to random others
  - It interacts with some popular items
  - Include random attack, average attack, bandwagon attack, and segment attack
  - . . .

Heuristic Attack

Heuristic Attack

- Random Attack
  - Attacker's Goal: promote certain items availability of being recommended

high scores to target item

	Item1	Item2	Item3	Item4	Item5	Item6
User1	4	3	4	-	3	4
User2	5	5	1	4	1	3
User3	1	5	2	5	4	2
User4	5	1	5	3	-	5
User5	3	5	4	4	1	0
User6	-	5	5	4	-	2
Attacker1	1	-	1	1	5	-
Attacker2	-	1	1	1	5	-

low score to random others

Shilling Recommender Systems for Fun and Profit, WWW 2004.

Heuristic Attack

- Average Attack

high scores to target item

	Item1	Item2	Item3	Item4	Item5	Item6
User1	4	3	4	-	3	4
User2	5	5	1	4	1	3
User3	1	5	2	5	4	2
User4	5	1	5	3	-	5
User5	3	5	4	4	1	0
User6	-	5	5	4	-	2
Attacker1	3	4	3	4	5	-
Attacker2	3	4	3	4	5	-

average score to random others

Shilling Recommender Systems for Fun and Profit, WWW 2004.

Heuristic Attack

- Bandwagon attack

popular item

target item

	Item1	Item2	Item3	Item4	Item5	Item6
User1	-	4	4	-	3	-
User2	-	5	1	-	1	3
User3	1	4	2	1	4	-
User4	-	4	5	-	-	-
User5	-	5	4	-	1	-
User6	-	5	5	-	-	-
Attacker1	-	4	4	-	5	-
Attacker2	-	4	4	-	5	-

Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness, TOIT 2007.

Heuristic Attack

- Segment attack

similar item

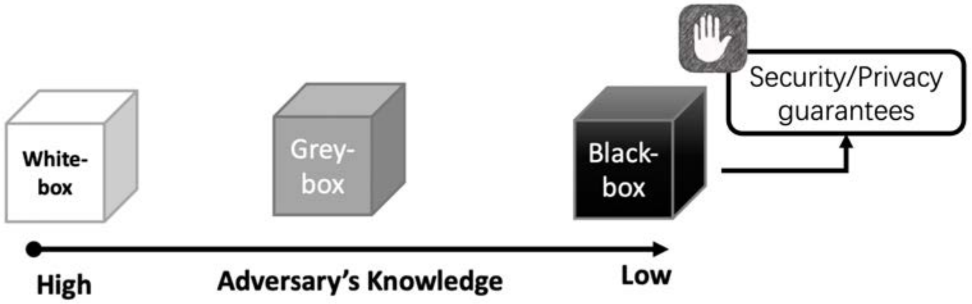
target item

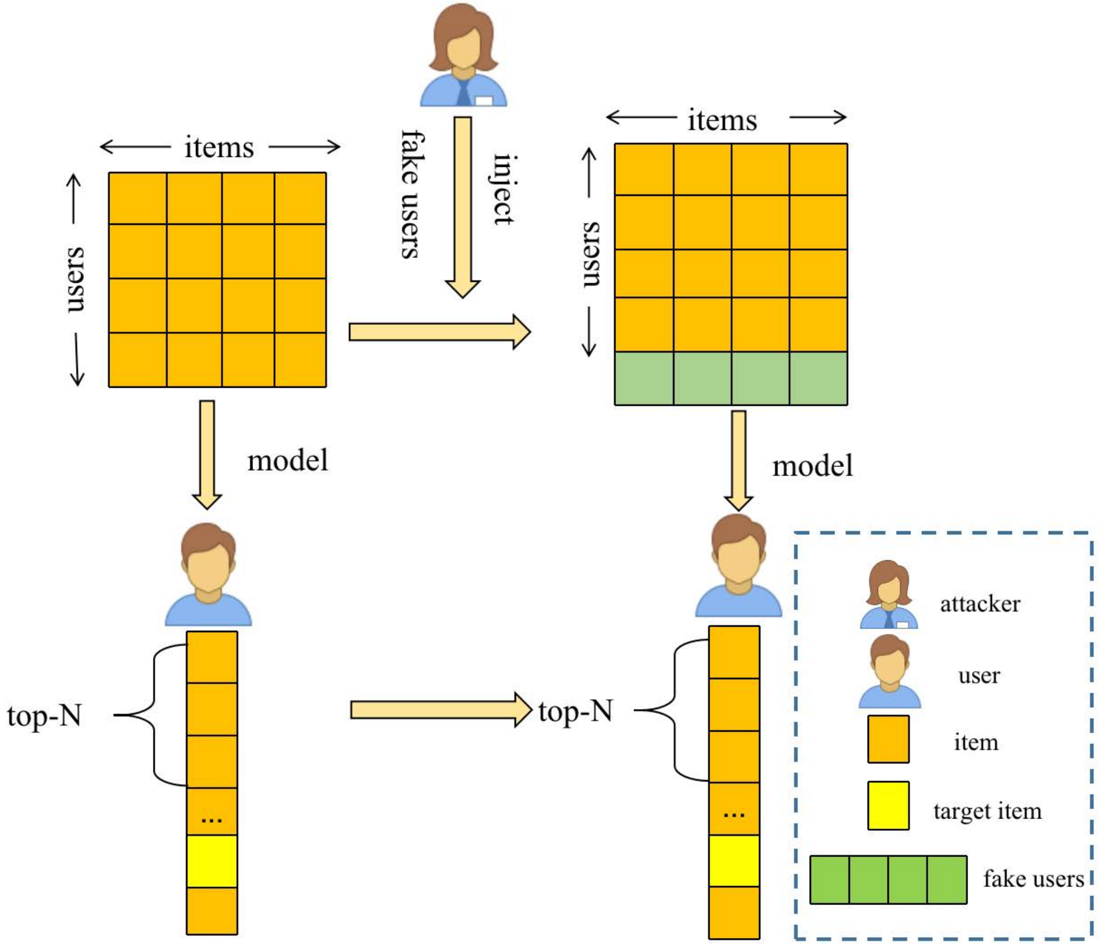
	Item1	Item2	Item3	Item4	Item5	Item6
User1	4	3	4	-	3	4
User2	5	5	1	4	1	3
User3	1	5	2	5	4	2
User4	5	1	5	3	-	5
User5	3	5	4	4	1	0
User6	-	5	5	4	-	2
Attacker1	1	4	4	1	5	-
Attacker2	-	4	4	1	5	-

Segment-based injection attacks against collaborative filtering recommender systems, ICDM 2005.

Gradient-based Attack

- Gradient-based Methods
  - White-Box Attack: Optimization

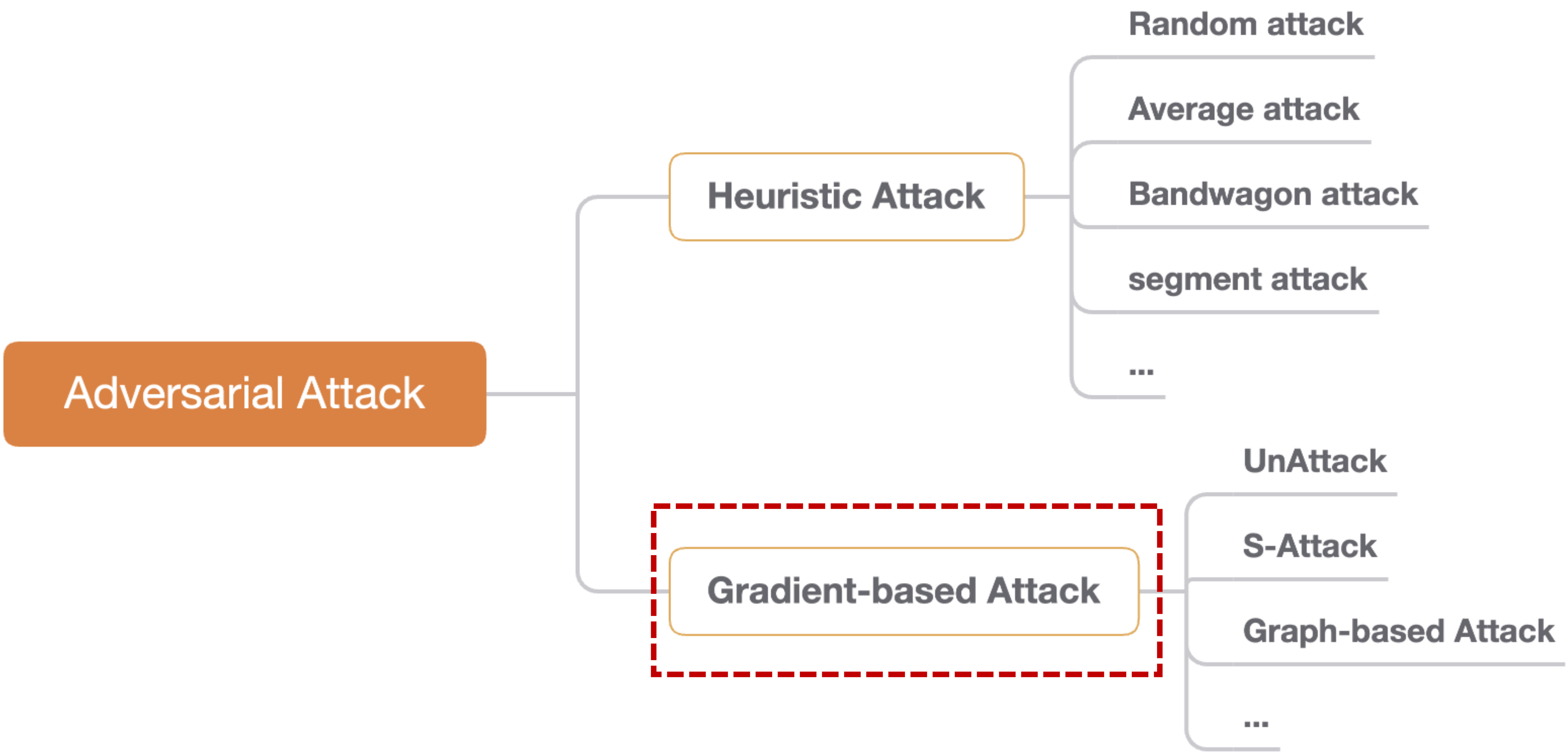




$$\min_{\widehat{U}} \mathcal{L}_{adv}(\theta^*), \quad \text{s.t.} \quad \theta^* = \arg\min_{\theta}(\mathcal{L}_{rec}(\boldsymbol{R}, \boldsymbol{O}_\theta) + \mathcal{L}_{rec}(\widehat{\boldsymbol{R}}, \boldsymbol{O}_\theta))$$

Data poisoning attacks on neighborhood-based recommender systems, ETT 2019.

Gradient-based Attack



UNAttack

- UNAttack
  - Optimize the ratings of fake users one by one rather than for all m fake users at the same time
  - Borrow the strategy from the ranking problem to construct pairwise loss function

$$p_{ui} = \sum_{v \in S(u,K) \cap U_i^+)} s_{uv} X_{vi}$$

$$loss_1 = \sum_{v \in S(u,K)} \sigma(\boxed{s_{uv}} - s_{uf})$$

$$loss_2 = \sum_{i \in L_u} \sigma(\boxed{p_{ui}} - p_{ut})$$

$$loss_u = (1-\lambda) loss_1 + \lambda loss_2$$

$$loss = \sum_{u \in U_t^-} loss_u$$

$$Minimize(F(X_f) = loss)$$

$$s.t. \ |X_f'| \leq z,$$

$$X_{fi} \in \{0, 1, \ldots, r_{max}\}$$

Make the fake user be in the top-K nearest neighbours of user, which can be expressed as $s_{uf} > s_{uv}$.

Data poisoning attacks on neighborhood-based recommender systems, ETT 2019.

UNAttack

- UNAttack
  - Choosing the optimal filler-items for fake users

$$X_f^{(t)} = Project(X_f^{(t-1)} - \eta \frac{\partial F(X_f)}{\partial X_f})$$

where $Project(x)$ is the project function that cuts each X_{fi} into the range $[0,1,..r_{max}]$.

$$\frac{\partial F(X_f)}{\partial X_f} = \sum_{u \in U_{\bar{t}}} (1-\lambda)\frac{\partial loss_1}{\partial X_f} + \lambda \frac{\partial loss_2}{\partial X_f}$$

Gradient

$$\frac{\partial (loss_1)}{\partial X_f} = \sum_{v \in S(u,k)} \frac{\partial \sigma(Q)}{\partial Q}\left(\boxed{\frac{\partial s_{uv}}{\partial X_f} - \frac{\partial s_{uf}}{\partial X_f}}\right)$$

$$\frac{\partial (loss_2)}{\partial X_f} = \sum_{i \in L_u} \sum_{v \in W} \frac{\partial \sigma(P)}{\partial P}\left(\frac{\partial s_{uv} X_{vi}}{\partial X_f} - \frac{\partial s_{uf} X_{ft}}{\partial X_f}\right)$$

similarity

$$\frac{\partial s_{uf}}{\partial X_f} = \frac{X_u}{\|X_u\| \|X_f\|} - \frac{X_u X_f}{\|X_u\| \|X_f\|} \frac{X_f}{\|X_f\|^2}$$

Data poisoning attacks on neighborhood-based recommender systems, ETT 2019.

UNAttack

- UNAttack

Give the target items the maximum ratings.

Algorithm 1. UNAttack

Input: Matrix $R_{m \times n}$
Parameter: λ, K, N, z, j
Output: j fake users

for each fake user f **do**
  Solve the problem in Equation 6 with current rating matrix R to get X_f
  Let $X_{ft} = r_{\max}$
  Select z items with highest value in X_{fi} as filler items.
  For each filler-items j, $X_{fj} \sim \mathcal{N}(\mu_j, \sigma_j^2)$
  $R_{m \times n} = R_{m \times n} \cup X_f$
end for

Inspired by the ranking problem, all items will be ranked according to X_{fi}, and top-z items with the highest values will be chosen as the filler-items.

The rating score assigned to each filler-item is drawn from a normal distribution of the normal users' rating data of this item.

Data poisoning attacks on neighborhood-based recommender systems, ETT 2019.

S-Attack

- Attack matrix factorization based recommender systems
  - Attacker's Goal: promote certain items availability of being recommended
  - Attacker's knowledge: fully (partial) observable dataset
  - Challenge:
    - User ratings are discrete
    - Excessive number of users

$$\arg\min_{X,Y} \sum_{(u,i)\in\mathcal{E}} \left(r_{ui} - \boldsymbol{x}_u^\top \boldsymbol{y}_i\right)^2 + \lambda \left(\sum_u \|\boldsymbol{x}_u\|_2^2 + \sum_i \|\boldsymbol{y}_i\|_2^2 \right)$$

$$\begin{aligned} \max\ & h(t) \\ \text{s.t.}\ & |\Omega_v| \le n+1, && \forall v \in \mathcal{M}, \\ & r_{vi} \in \{0, 1, \cdots, r_{max}\}, && \forall v \in \mathcal{M}, \forall i \in \Omega_v. \end{aligned}$$

Influence Function based Data Poisoning Attacks to Top-N Recommender Systems, WWW 2020.

S-Attack

- Step 1: Optimize one by one
- Step 2: Relax the discrete ratings to continuous

$$\boldsymbol{w}_v = [w_{vi}, i \in \Omega_v]^\top$$

$$r_{vi} \in \{0, 1, \cdots, r_{max}\} \Rightarrow w_{vi} \in [0, r_{max}] \Rightarrow w_{vi} \in \{0, 1, \cdots, r_{max}\}$$

Discrete | Continues | Discrete

Influence Function based Data Poisoning Attacks to Top-N Recommender Systems, WWW 2020.

S-Attack

- Step 3: Approximating the Hit Ratio
- Step 4: Determining the Set of Influential Users

$$\min_{\mathbf{w}_v} \mathcal{L}_{\mathcal{U}}(\mathbf{w}_v) = \sum_{u \in \mathcal{U}} \sum_{i \in \boxed{\Gamma_u}} g(\hat{r}_{ui} - \hat{r}_{ut}) + \eta \|\mathbf{w}_v\|_1$$
$$\text{s.t. } w_{vi} \in [0, r_{max}],$$

Top-k list

Influential Users

$$\min_{\mathbf{w}_v} \mathcal{L}_{\boxed{\mathcal{S}}}(\mathbf{w}_v) = \sum_{u \in \mathcal{S}} \sum_{i \in \Gamma_u} g(\hat{r}_{ui} - \hat{r}_{ut}) + \eta \|\mathbf{w}_v\|_1$$
$$\text{s.t. } w_{vi} \in [0, r_{max}].$$

Influence Function based Data Poisoning Attacks to Top-N Recommender Systems, WWW 2020.

Graph-Based Attack

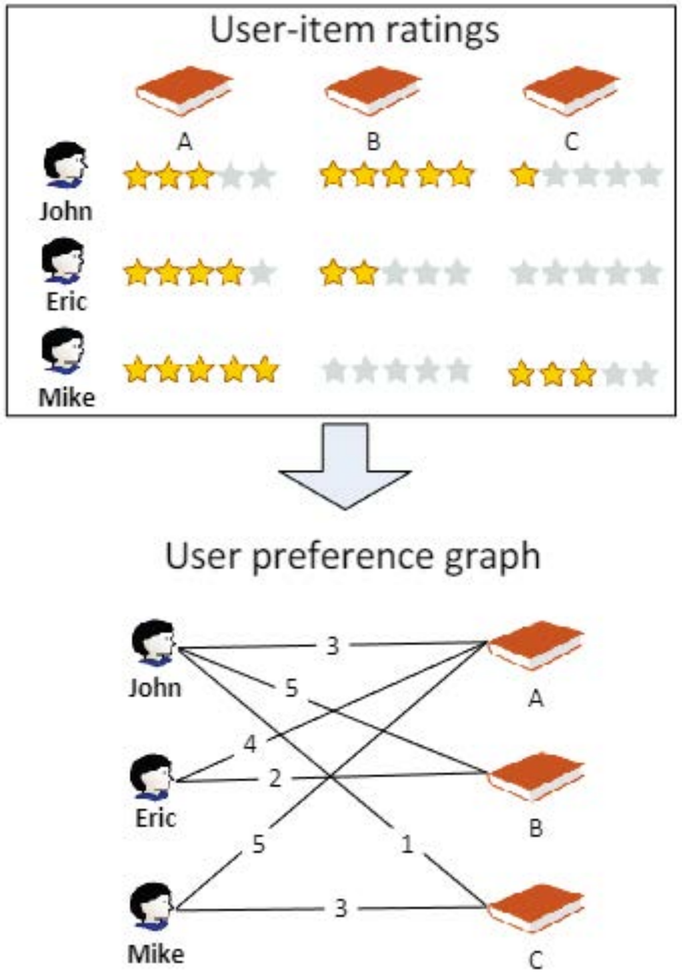
- Attack graph-based recommender systems
  - Attack using random walk algorithm

Random walk: $$p_u = (1-\alpha)\cdot Q \cdot p_u + \alpha \cdot e_u$$

$$Q_{xy} = \begin{cases} \frac{r_{xy}}{\sum_{z\in\Gamma_x} r_{xz}} & \text{if } (x,y)\in E \\ 0 & \text{otherwise} \end{cases}$$

Loss function: $$l_u = \sum_{i\in L_u} g(p_{ui} - p_{ut})$$

$$g(x) = \frac{1}{1+\exp(-x/b)}$$

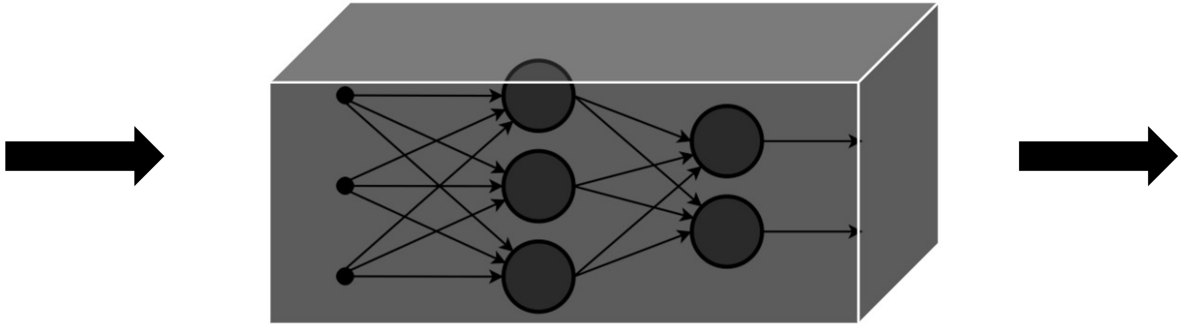


Poisoning Attacks to Graph-Based Recommender Systems, ACSAC 2018.

Black-Box Attack

- Black-Box Attack

Perturbed Data



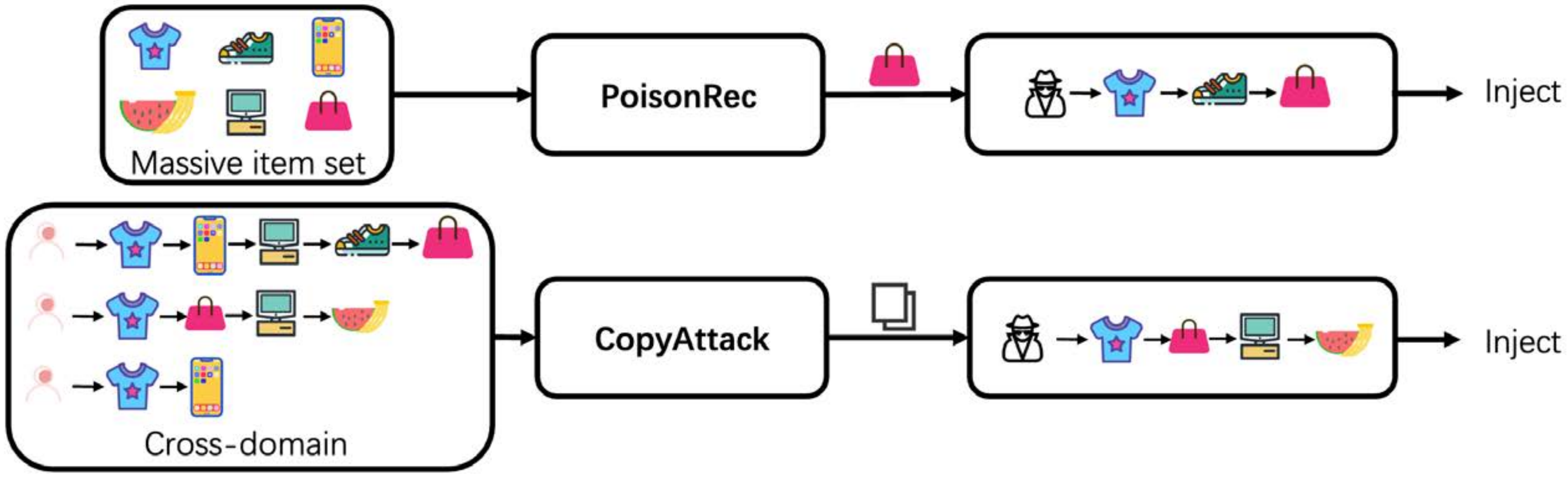
Promote/Demote Target Item

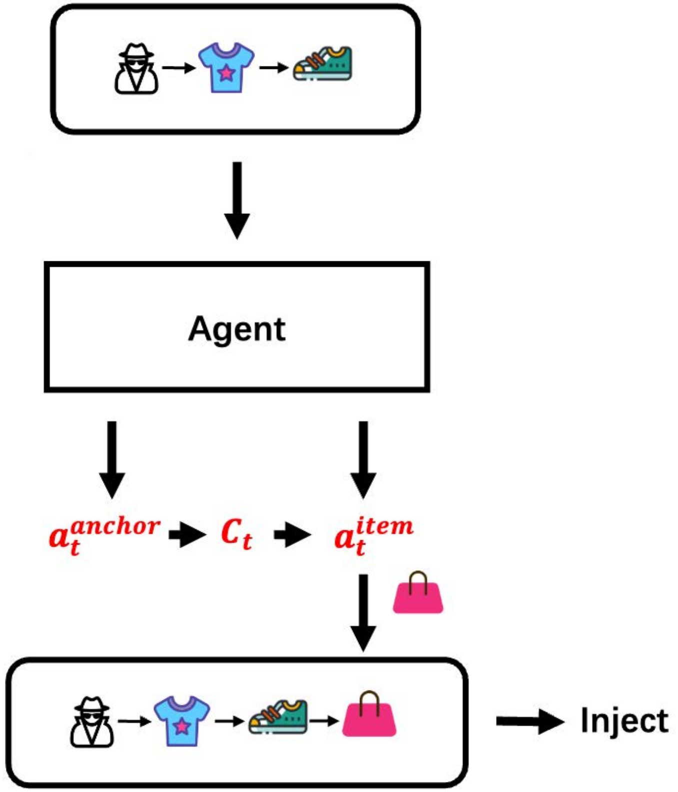
Reinforcement Learning-based Attack

- Challenges in existing attacking methods:
  - Model structure, parameters and training data are unknown
  - Unable to get user-item interactions
  - Black-box setting
    - Reinforcement Learning (RL) -- Query Feedback (Reward)

Reinforcement Learning-based Attack

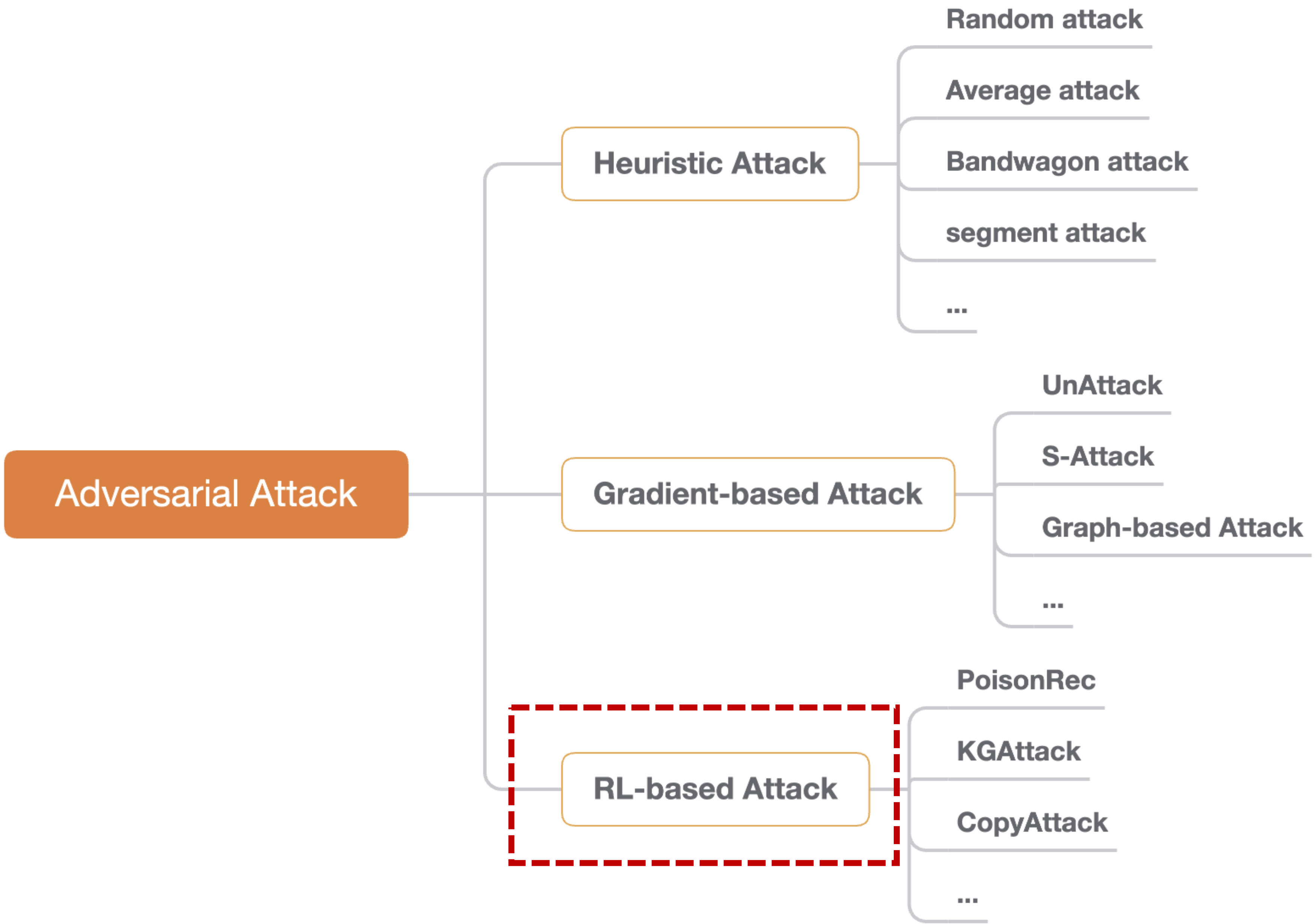
- Reinforcement Learning-based Methods
  - PoisonRec
  - KGAttack
  - CopyAttack





An Adaptive Data Poisoning Framework for Attacking Black-box Recommender Systems, ICDE 2020.
Attacking Black-box Recommendations via Copying Cross-domain User Profiles, ICDE 2021
Knowledge-enhanced Black-box Attacks for Recommendations, KDD 2022

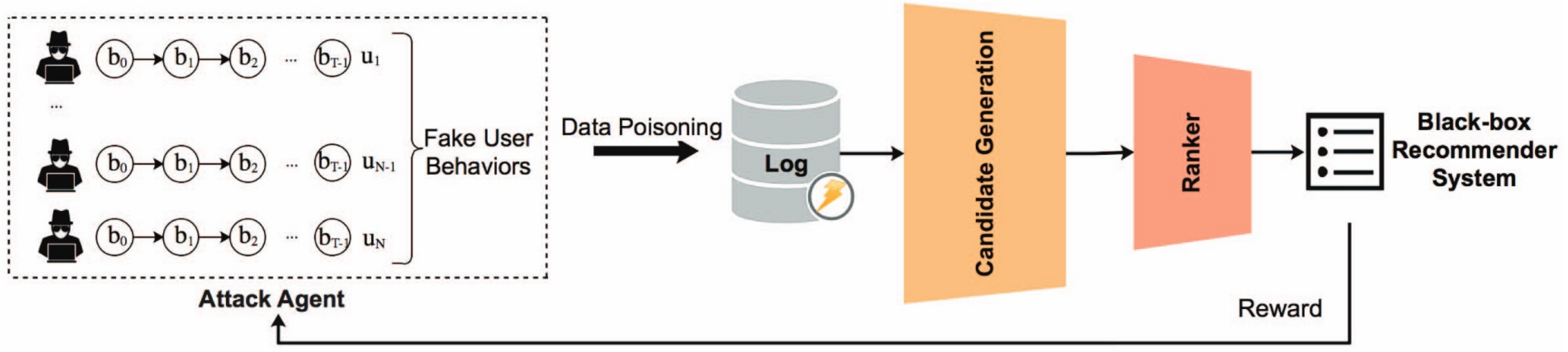
Reinforcement Learning-based Attack



PoisonRec

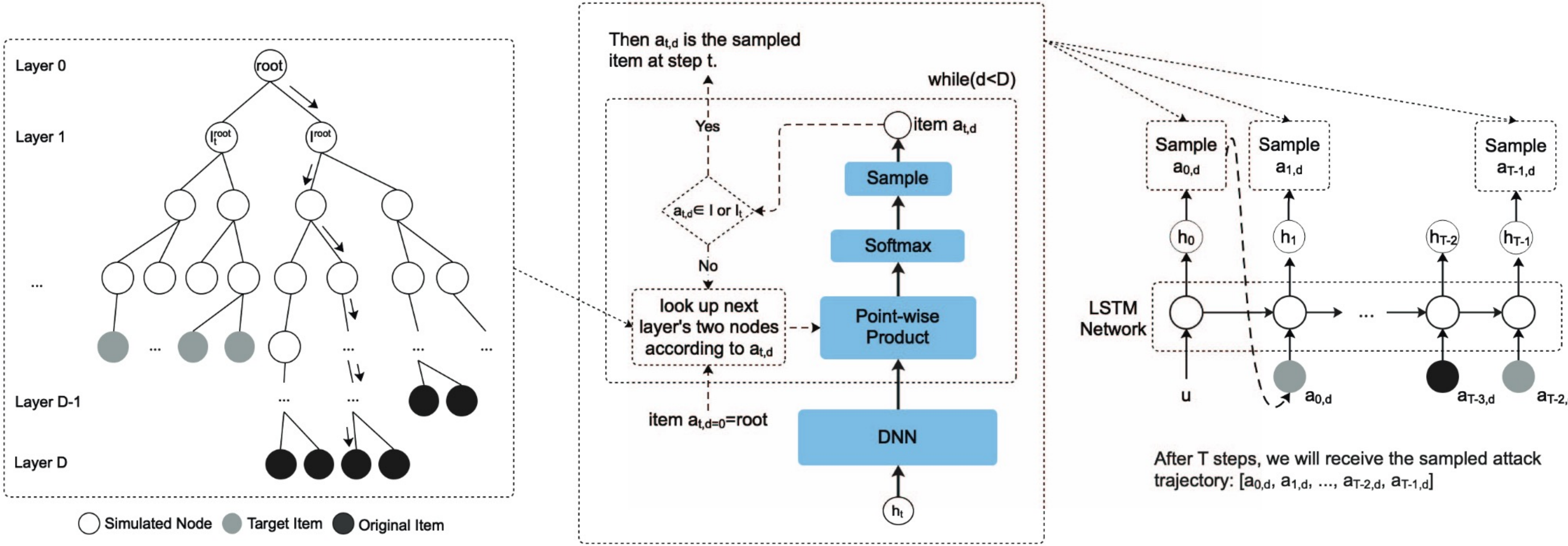
- Target: $$RecNum = \sum_u |L_u \cap I_t|$$

- DNN + PPO



PoisonRec

- Introduce (Biased Complete Binary Tree) BCBT to reduce action space



(a) The biased complete binary tree, BCBT

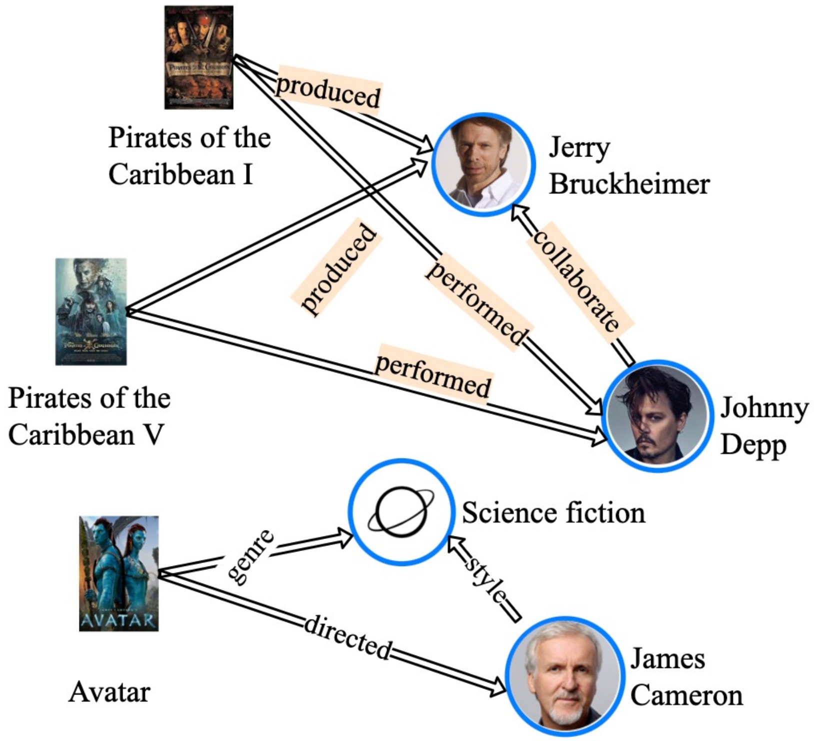
(b) The sampling process on BCBT

(c) The sampling process for a complete attack trajectory.

An Adaptive Data Poisoning Framework for Attacking Black-box Recommender Systems, ICDE 2020.

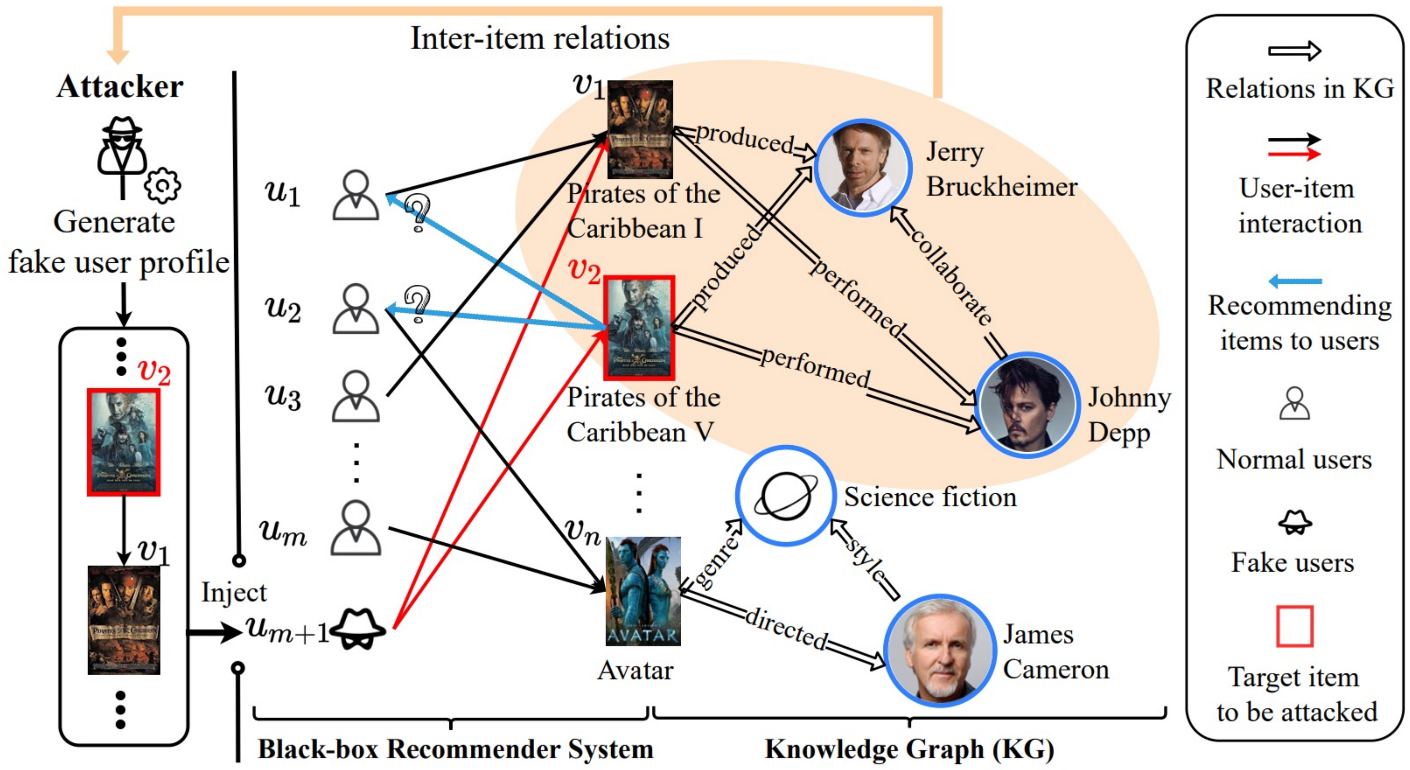
KGAttack

- Side-information: Knowledge Graph (KG)
  - Rich auxiliary knowledge: relations among items and real-world entities
  - The underlying relationships between Target items and other items



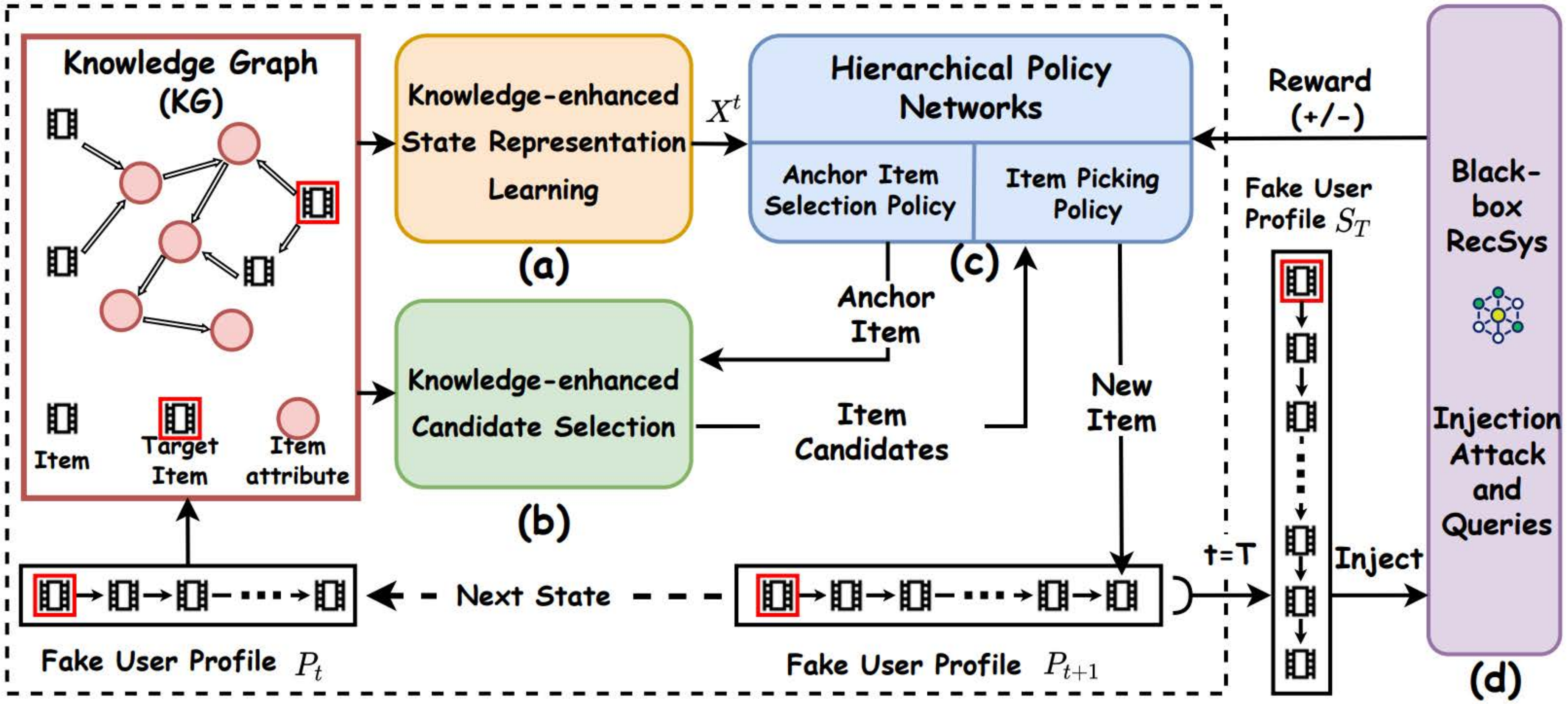
KGAttack

- Employs the KG to enhance the generation of fake user profiles from the massive item sets



KGAttack

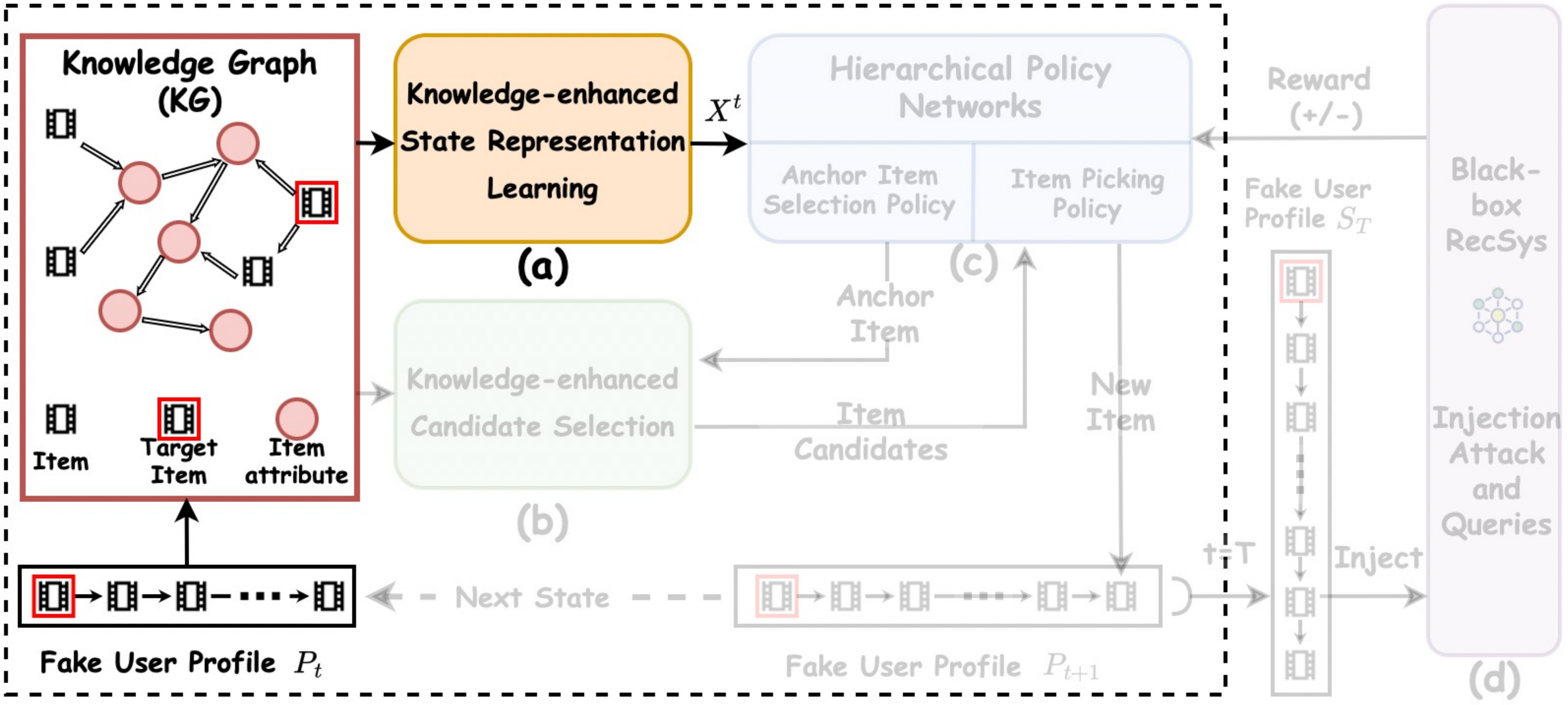
- Using KG to enhance the representation of state
- RL agent, generate user profiles



Knowledge-enhanced Black-box Attacks for Recommendations, KDD 2022

KGAttack

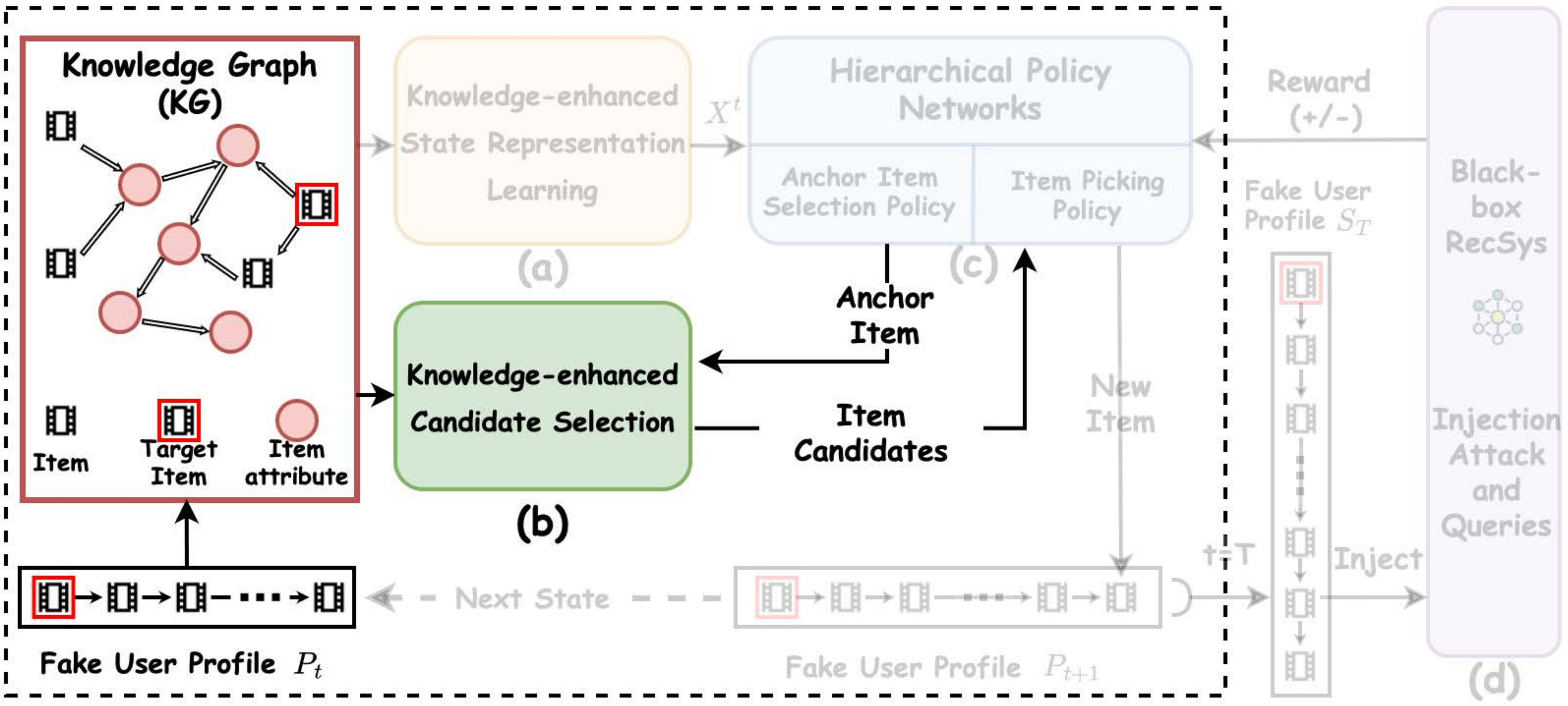
- (a): Using KG to enhance the representation of state



Knowledge-enhanced Black-box Attacks for Recommendations, KDD 2022

KGAttack

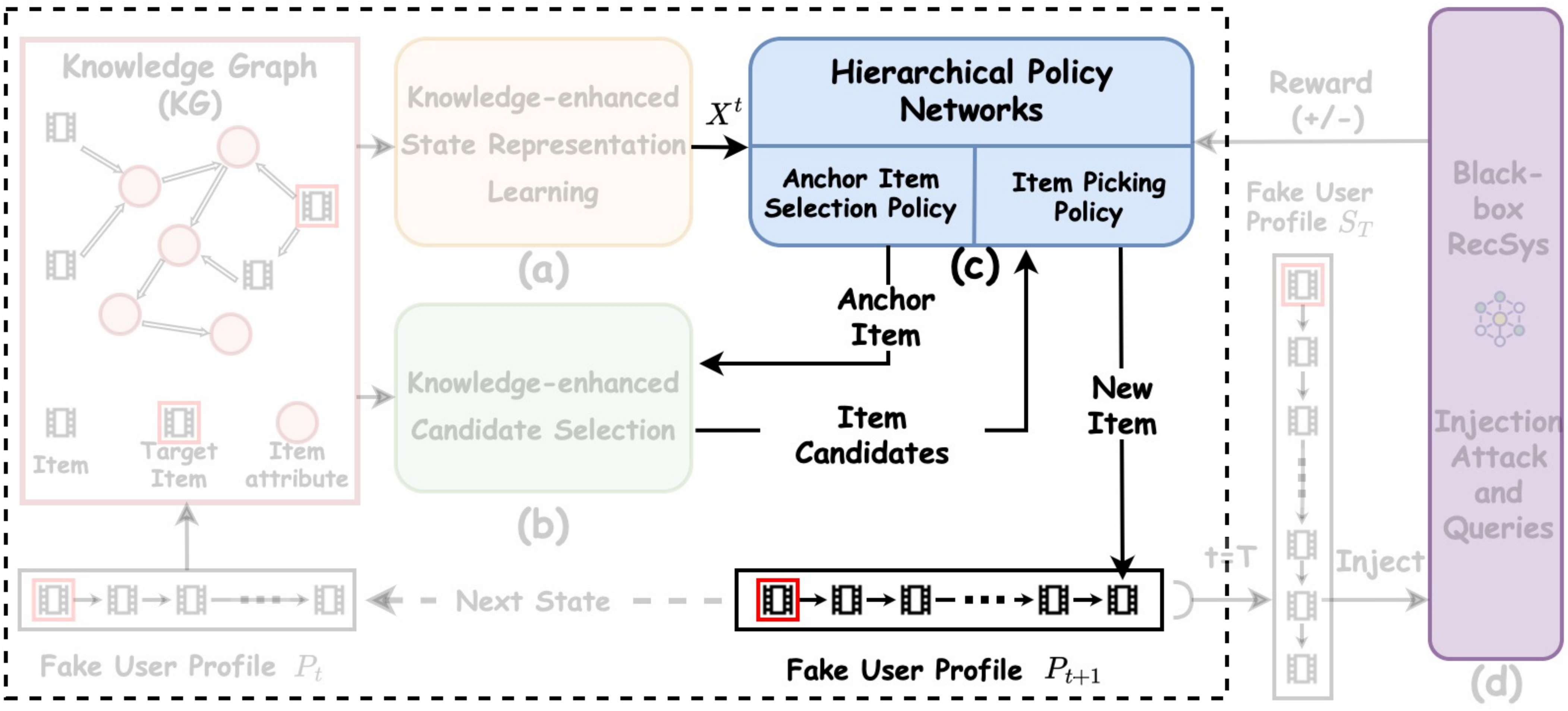
- (b): Using KG to localize relevant item candidates



Knowledge-enhanced Black-box Attacks for Recommendations, KDD 2022

KGAttack

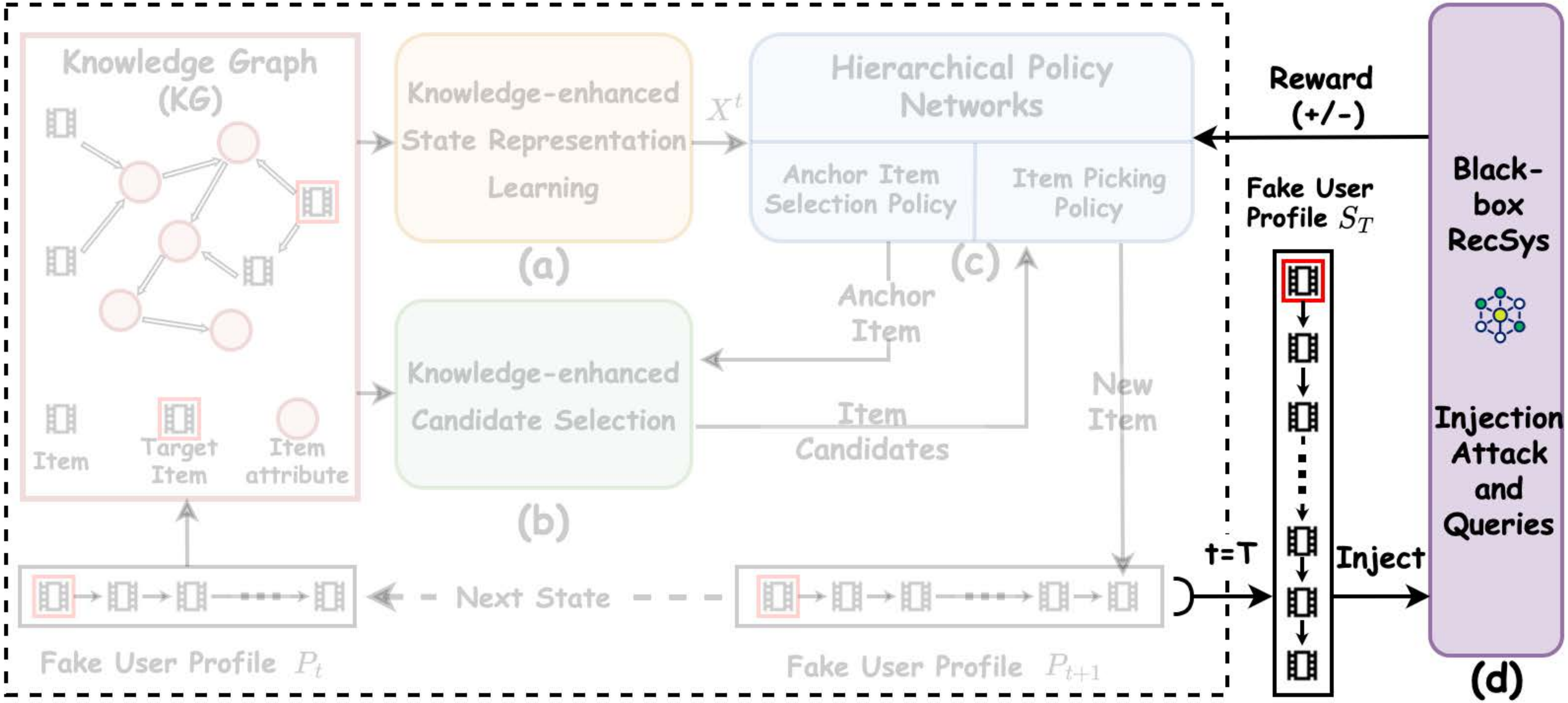
- (c): Using KG to localize relevant item candidates



Knowledge-enhanced Black-box Attacks for Recommendations, KDD 2022

KGAttack

- (d): Injection attacks and query



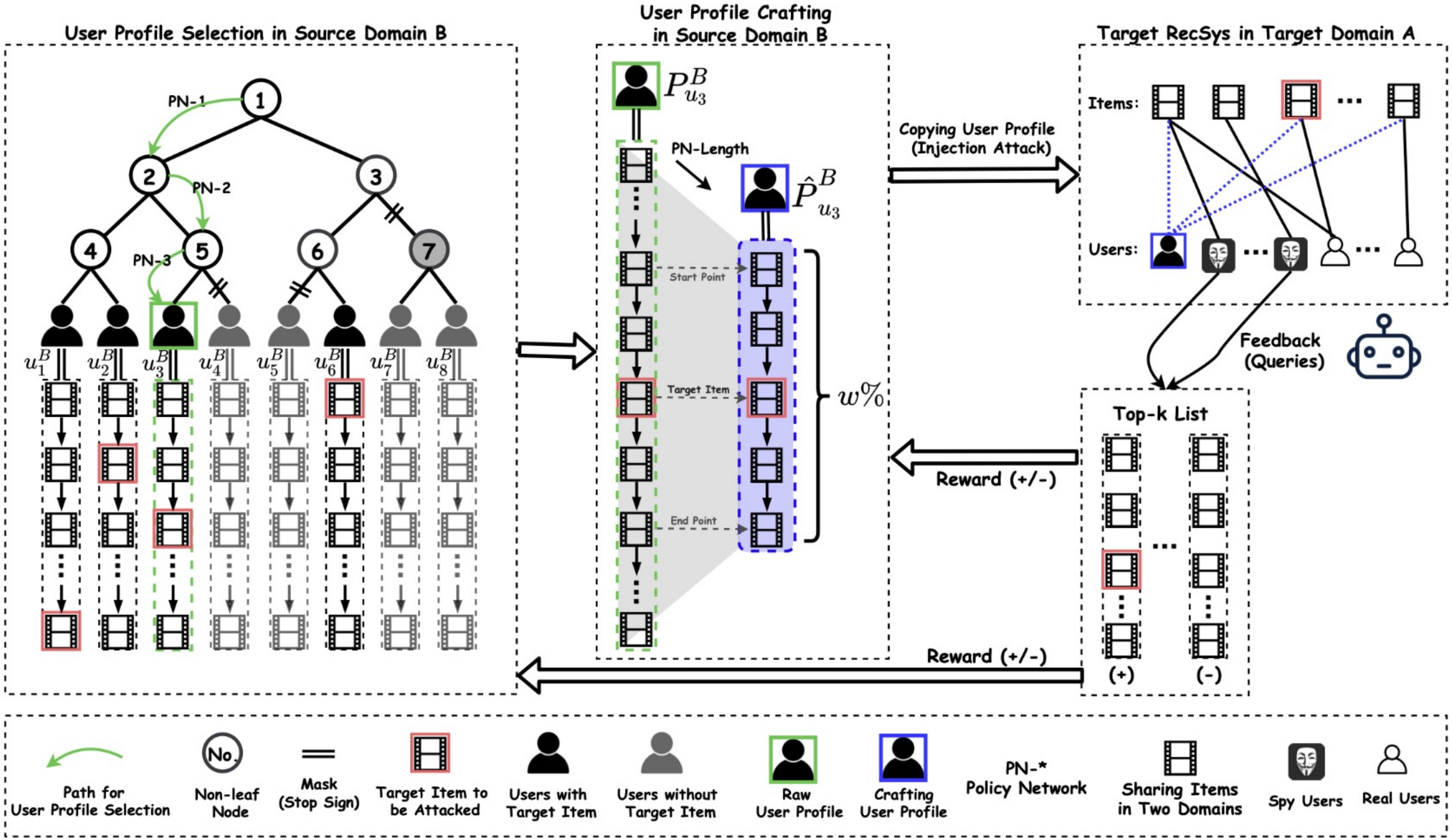
Knowledge-enhanced Black-box Attacks for Recommendations, KDD 2022

CopyAttack

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

Taobao JD.com

CopyAttack



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

CopyAttack

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

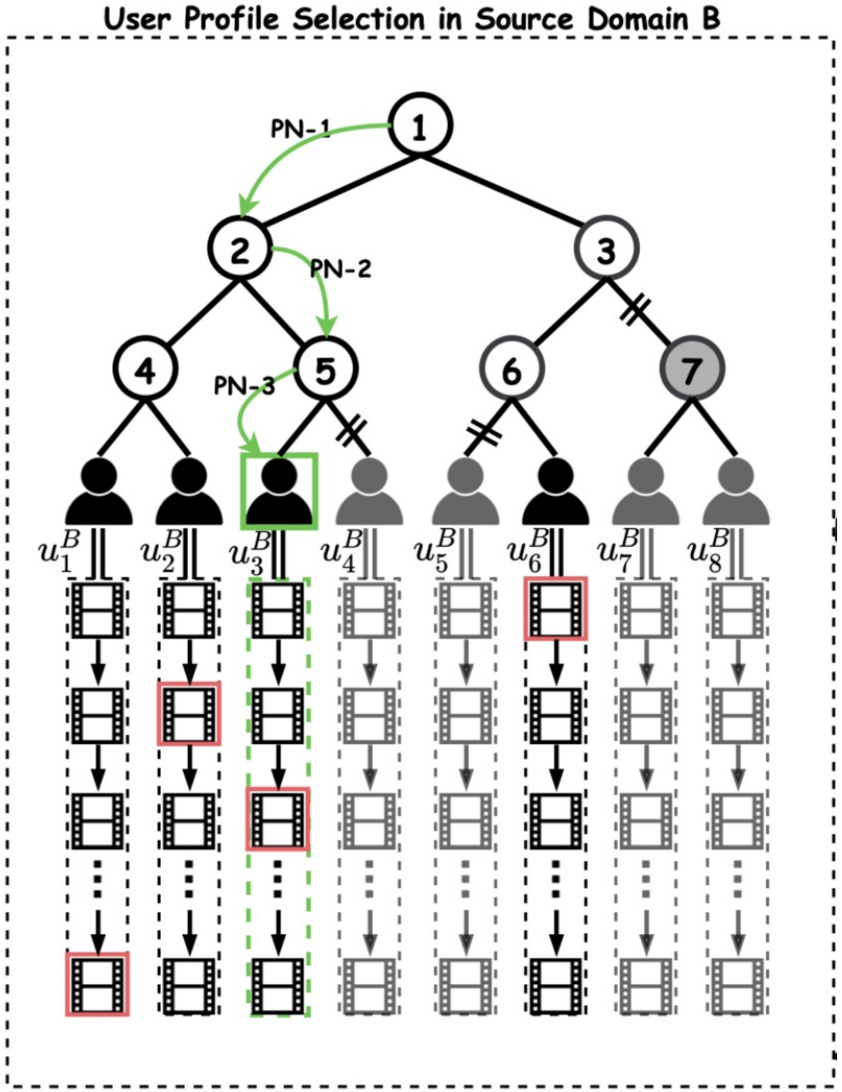
$$a_t^u = \left\{ a_{[t,1]}^u, a_{[t,2]}^u, \ldots, a_{[t,d]}^u \right\}$$

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

$$\mathbf{x}_{v_*} = RNN\left(\mathcal{U}_t^{B \to A}\right)$$

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

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



CopyAttack

- 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)

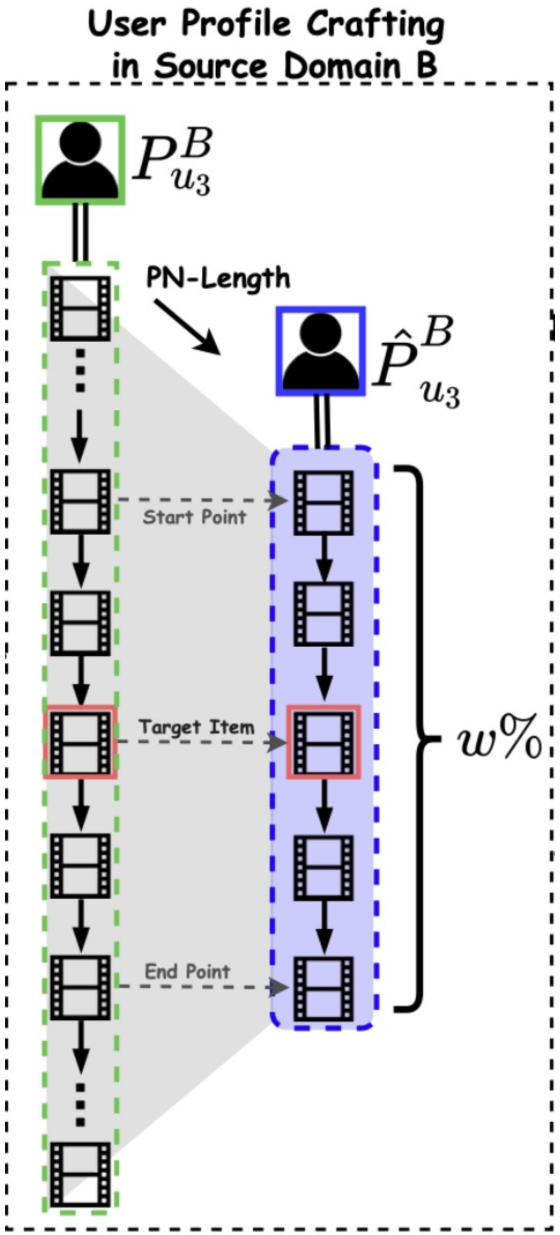
Example:

w = 50%

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

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

$$p^l(\cdot \mid s_t^l) = \mathrm{softmax}\big(MLP([\mathbf{p}_i^B \oplus \mathbf{q}_{v_*}^B] \mid \theta^l)\big)$$



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

Outline

Concepts and Taxonomy

Adversarial Attack

Adversarial Defense

Application

Adversarial Learning Surveys and Tools

Future directions

Detection

- Exceptions and outliers in the recommendation system
  - Discrepancies between user’s ratings and item’s average ratings
  - Spectrum-based features of series rate values of each user
  - Cluster instances
  - User behaviors
  - The process of learning users and items representations
  - The distribution of normal users’ behaviors over a partial dataset
  - . . .

Detection


Adversarial Defense
Detection
DegSim and RDMA
PPu and Du
TSGR, RSF, and TBR
...

Detection

- Detection of shilling attacks in online recommender systems
  - Detecting Process:
    - Extract the supposed characteristics, DegSim and RDMA

Degree of similarity with Top Neighbors:

$$\mathrm{Degsim_u} = \frac{\sum_{\mathrm{v}=1}^{\mathrm{k}} \mathrm{W_{u,v}}}{\mathrm{k}}$$

Rating Deviation from Mean Agreement:

$$RDMA_j = \frac{\sum_{i=0}^{N_j} \frac{|r_{i,j} - Avg_i|}{NR_i}}{N_j}$$

Preventing shilling attacks in online recommender systems, WIDM 2005

Detection

- Detection of shilling attacks via selecting patterns analysis
  - Detecting Process:
    - Extract the supposed characteristics, popularity profile and popularity distribution

A set of item popularity values of rated items: $PP_u = (d_{u,1}, d_{u,2}, \ldots, d_{u,N_u})$

Popularity distribution: $D_u = (p_{u,1}, p_{u,2}, \ldots, p_{u,d_{max}})$

Shilling Attack Detection in Recommender Systems via Selecting Patterns Analysis, IEICE 2016

Detection

- Detection of trust shilling attacks in recommender systems
  - Detecting Process:
    - Extract the supposed characteristics, TSGR, RSF, and TBR

$$\boxed{TSGR_i} = \frac{tg_i \cap tr_i}{tg_i \cup tr_i}$$

User i's trust similarity between trust givers and trust receivers

$$\boxed{PTBR_u} = \frac{pn_u}{N_u}$$

Positive Trust Behavior Ratio

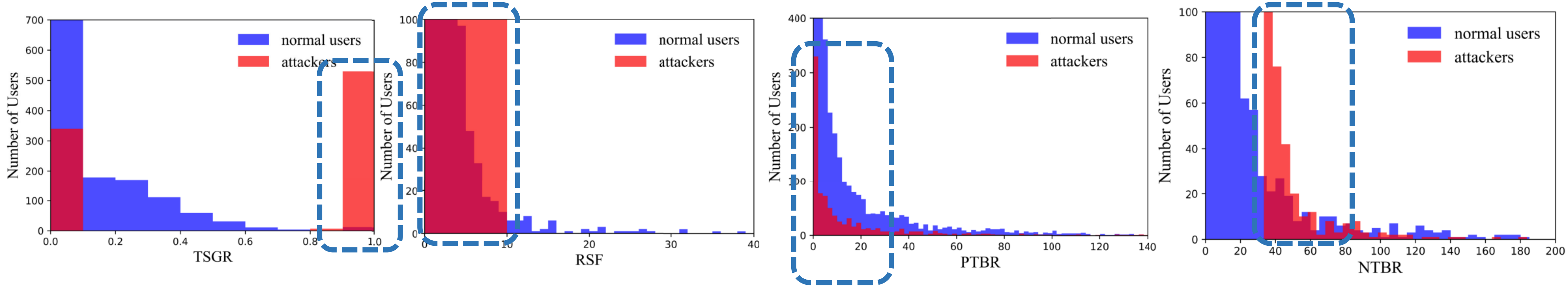
$$\boxed{NTBR_u} = \frac{nn_u}{N_u}$$

Negative Trust Behavior Ratio

Detection of Trust Shilling Attacks in Recommender Systems, IEICE 2022

Detection

- Normal vs. attackers distributions for each feature:



Detection of Trust Shilling Attacks in Recommender Systems, IEICE 2022

Adversarial Training

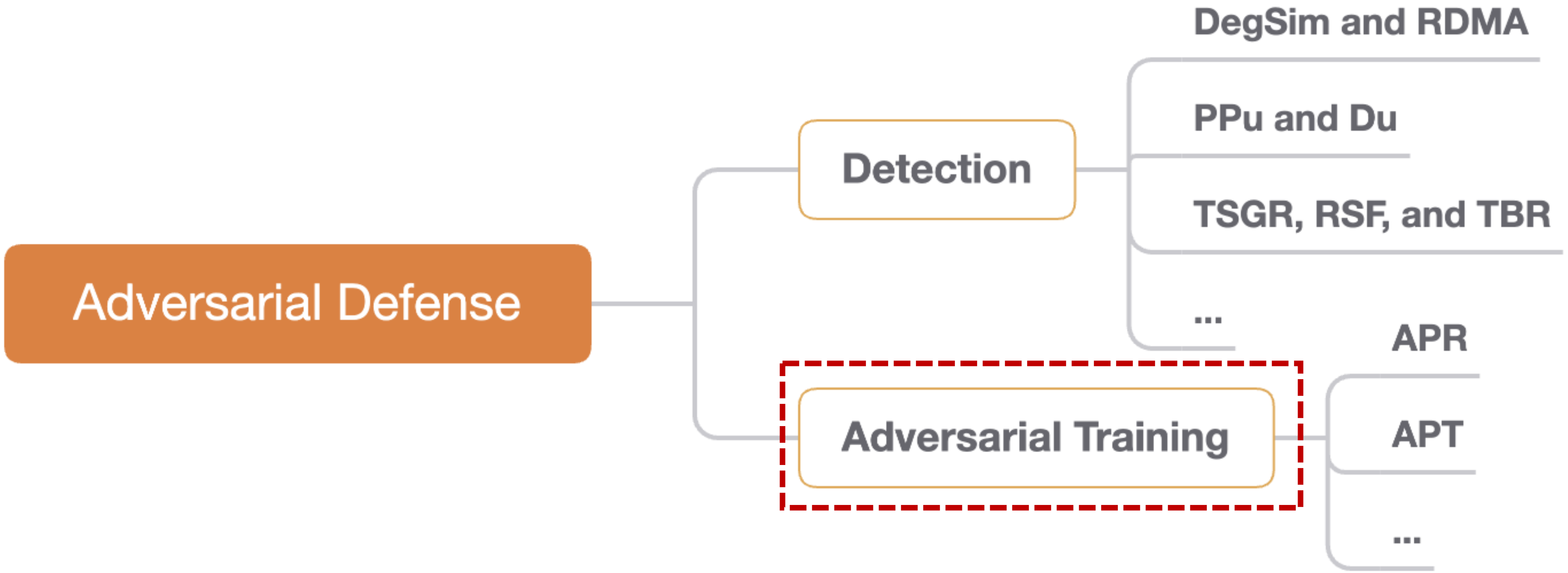
- Adversarial training contains two alternating processes:

  - Generating perturbations that can confuse a recommendation model

  - Training the recommendation model along with generated perturbations

$$\min_{\theta}\max_{\eta}\mathcal{L}(\mathcal{X}+\eta,\theta)$$

Adversarial Training



Adversarial Training

- Adversarial Personalized Ranking (APR)

Optimization objectives against noise:

$$\Delta_{adv} = \arg \max_{\Delta, ||\Delta|| \leq \epsilon} L_{BPR}(\mathcal{D}|\hat{\Theta} + \Delta)$$

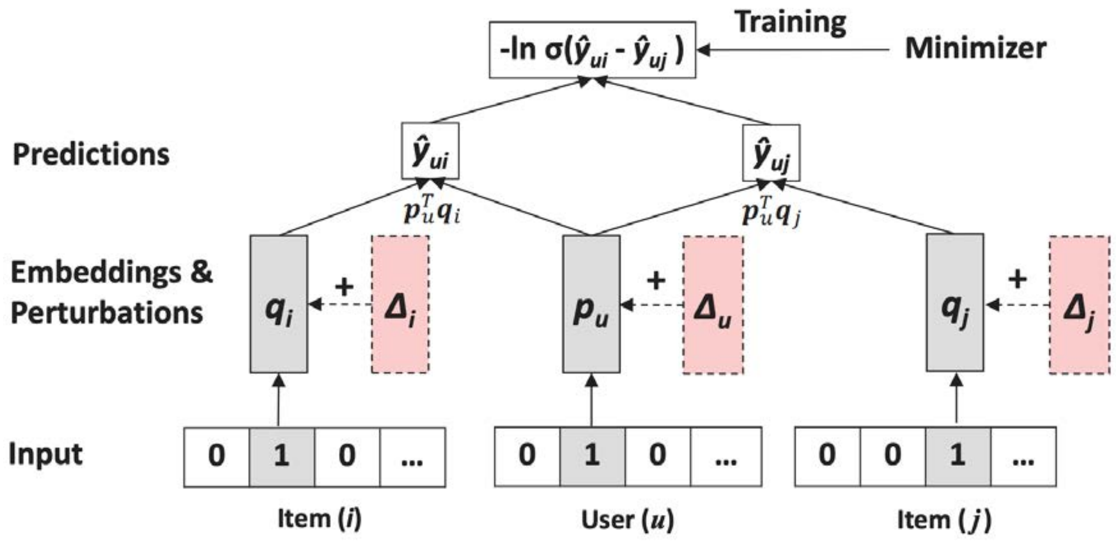
Adversarial Personalized Ranking (APR):

$$\mathrm{L}_{\mathrm{APR}}(\mathcal{D} \mid \Theta) = \mathrm{L}_{\mathrm{BPR}}(\mathcal{D} \mid \Theta) + \lambda \mathrm{L}_{\mathrm{BPR}}(\mathcal{D} \mid \Theta + \Delta_{\mathrm{adv}})$$

$$\text{where } \Delta_{\mathrm{adv}} = \arg \max_{\Delta, \|\Delta\| \leq \epsilon} \mathrm{L}_{\mathrm{BPR}}(\mathcal{D} \mid \hat{\Theta} + \Delta)$$

The training process of APR:

$$\Theta^*, \Delta^* = \arg\min_{\Theta} \max_{\Delta, ||\Delta|| \leq \epsilon} L_{BPR}(\mathcal{D}|\Theta) + \lambda L_{BPR}(\mathcal{D}|\Theta + \Delta)$$



Adversarial Training

- Adversarial poisoning training (APT)

$$\min_{\theta_R} \min_{\boxed{\mathcal{D}^*}, |\mathcal{D}^*|=n^*} \mathcal{L}(\mathcal{D} \cup \mathcal{D}^*, \theta_R)$$

$D^* = \{r_1^*, \ldots, r_{n*}^*\}$ is a set of $n*$ fake users dedicated to minimizing the empirical risk.

Algorithm 1: Adversarial Poisoning Training

Input: The epochs of training T, pre-training T_{pre}, and poisoning interval T_{inter}.

```
Randomly initialize the user set D* defined in Definition 3.1.   ①
for T_pre epochs do
    Do standard training on the dataset D;   ②
end
D' = D;
for T − T_pre epochs do
    for per T_inter epochs do
        Calculate the influence vector I according to Eq. 5;   ③
        for each ERM user in D* do
            Select m* items in Φ with probability
            exp(−tI_i) / Σ_{j∈Φ} exp(−tI_j) and rate the selected items with
            normal distribution (μ_i + r^+, σ_i) at random;   ④
        end
        D' = D ∪ D*;   ⑤
    end
    Do standard training on the dataset D';
end
```

Summary

- Adversarial Recommender System
  - Adversarial Attack
    - Heuristic Attack
      - Random attack
      - Average attack
      - Bandwagon attack
      - segment attack
      - ...
    - Gradient-based Attack
      - UnAttack
      - S-Attack
      - Graph-based Attack
      - ...
    - RL-based Attack
      - PoisonRec
      - KGAttack
      - CopyAttack
      - ...
  - Adversarial Defense
    - Detection
      - DegSim and RDMA
      - PPu and Du
      - TSGR, RSF, and TBR
      - ...
    - Adversarial Training
      - APR
      - APT
      - ...

Outline

- Concepts and Taxonomy
- Adversarial Attack
- Adversarial Defense
- **Application**
- Adversarial Learning Surveys and Tools
- Future directions

Application

- The application of adversarial training can help improve the trustworthiness and reliability of recommendation systems in various domains, including:
  - E-health recommendation
  - E-commercial recommendation
  - . . .

Outline

Concepts and Taxonomy

Adversarial Attack

Adversarial Defense

Application

Adversarial Learning Surveys and Tools

Future directions

Adversarial Learning Surveys

- Attack:
  - Zhang, Fuguo. "A survey of shilling attacks in collaborative filtering recommender systems." 2009
  - Gunes, Ihsan, et al. "Shilling attacks against recommender systems: A comprehensive survey." 2014
  - Si, Mingdan, and Qingshan Li. "Shilling attacks against collaborative recommender systems: a review." 2020

- Adversarial recommender systems:
  - Truong, Anh, Negar Kiyavash, and Seyed Rasoul Etesami. "Adversarial machine learning: The case of recommendation systems." 2018
  - Deldjoo, Yashar, Tommaso Di Noia, and Felice Antonio Merra. "A survey on adversarial recommender systems: from attack/defense strategies to generative adversarial networks." 2021

Adversarial Learning Tools

- RGRecSys (Ovaisi et al., 2022)

RGRecSys: A Toolkit for Robustness Evaluation of Recommender Systems, Ovaisi et al 2022.

Outline

Concepts and Taxonomy

Adversarial Attack

Adversarial Defense

Application

Adversarial Learning Surveys and Tools

Future directions

Future Directions

- Investigate vulnerability of different recommender systems
- Investigate vulnerability of Large Language Models in recommender systems
- Generate adversarial perturbations on user-item interactions for adversarial robust training
- Address open problems and challenges in robustness in recommendation

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

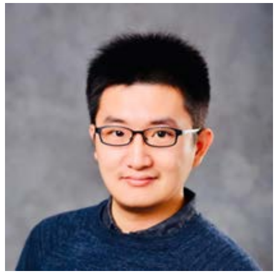
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

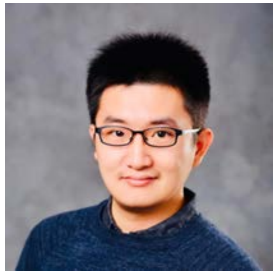
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

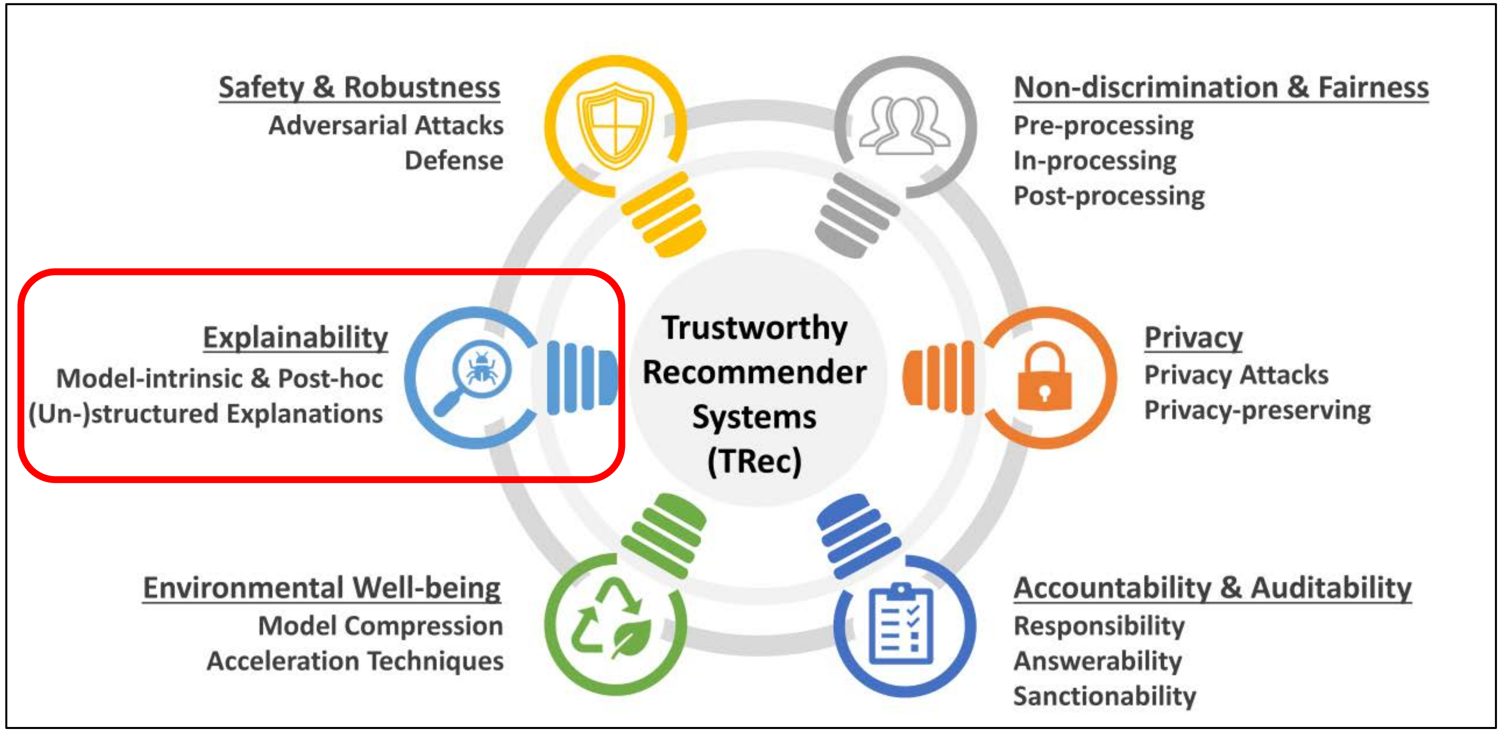
Future Directions



Xiangyu Zhao

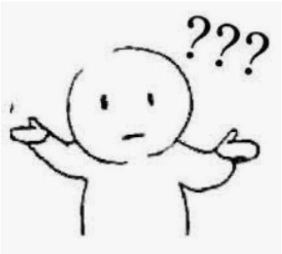
Explainability

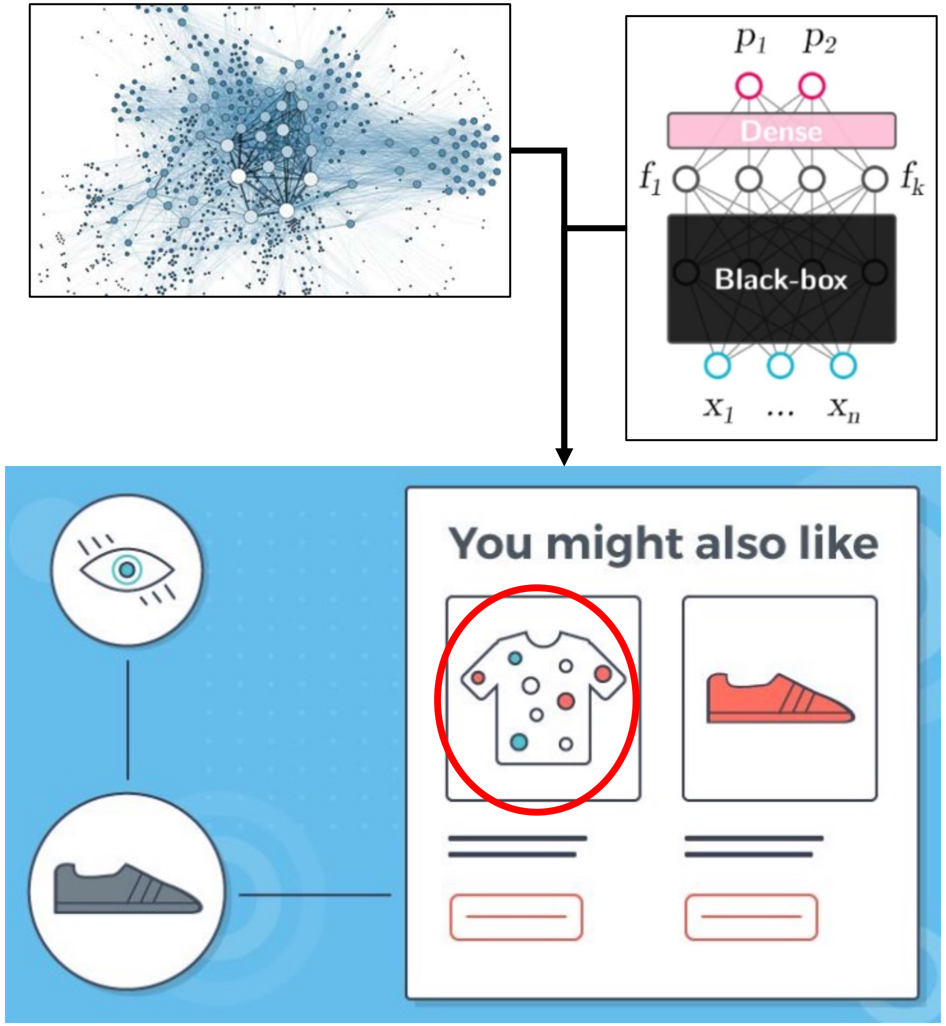
- **What's explainability in Rec, or to say explainable recommendations?**
  - It refers to the recommendation algorithms focusing on providing explanation for recommendation results



Explainability

- **Why do we need explainability in a trustworthy Rec system?**
  - Complicated modeling & Black-box module:

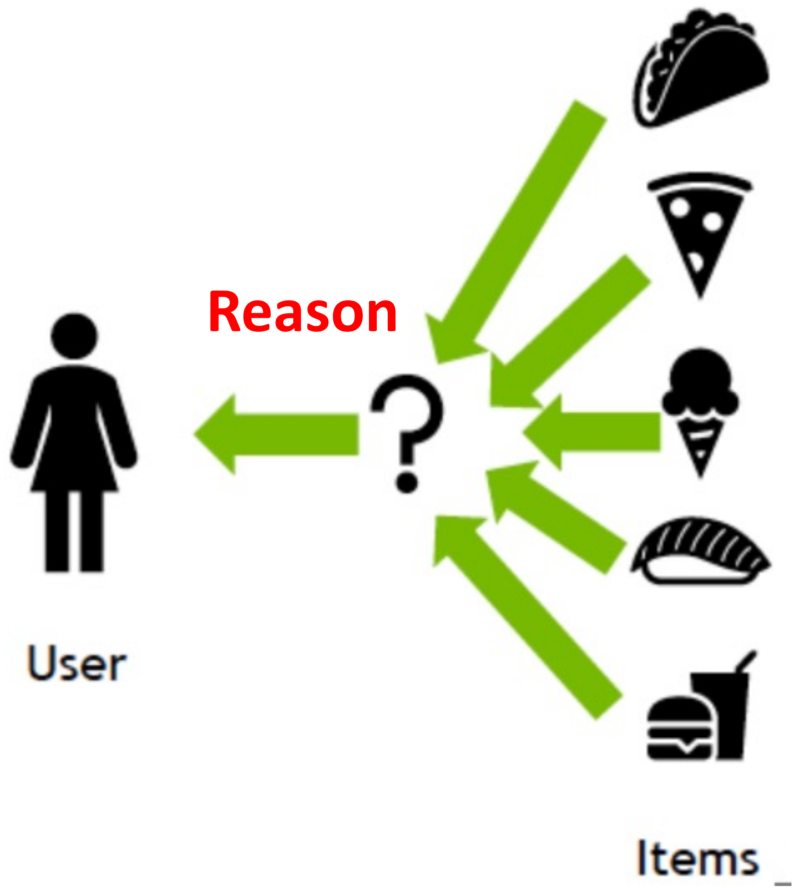




  - Why would you recommend this to me?
  - Similar style, same brand,
    or just a mis-recommendation?

Concepts

- **The ability to explain or to present in understandable terms to a human**



Explainability

METHODS

EVALUATIONS

APPLICATIONS

FUTURE DIRECTIONS

Taxonomy

- **How to produce explanations: model-intrinsic based (mostly used) or post-hoc**

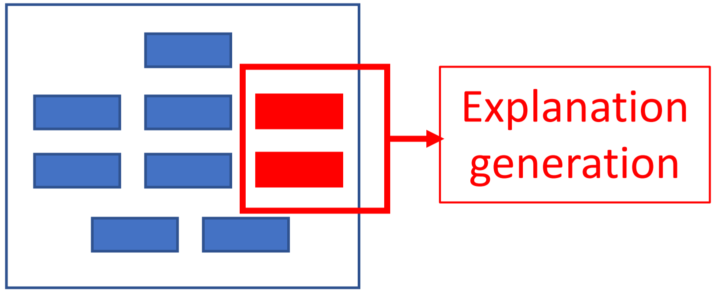
- **How the explanations are presented: structured or unstructured**

	Model-intrinsic based	**Post-Hoc**	***Characteristics***
Structured	[48, 114, 364, 389, 390, 396]	[280, 319]	Logical, Visible
Unstructured	[63, 64, 291]	[211, 315, 338]	Diversified, Fragmented
Focus	Model's reasoning process	Instances' relationship	-

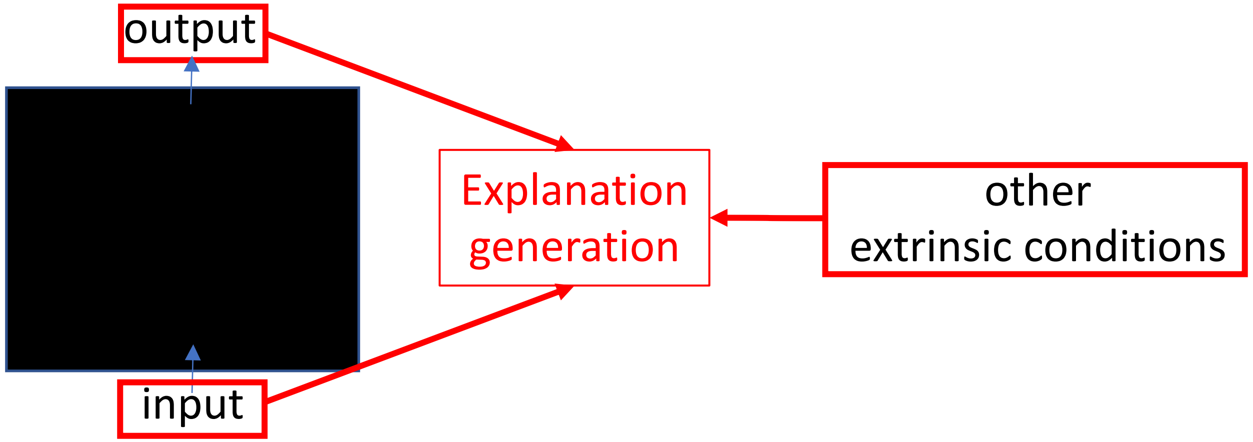
Note: Since some studies construct models from multiple perspectives at the same time, these different classifications are not completely antithetical

Taxonomy

- **The first criteria: How to produce explanations**
  - Model-intrinsic based methods: seek to derive explanations from the **intrinsic structure** of the model

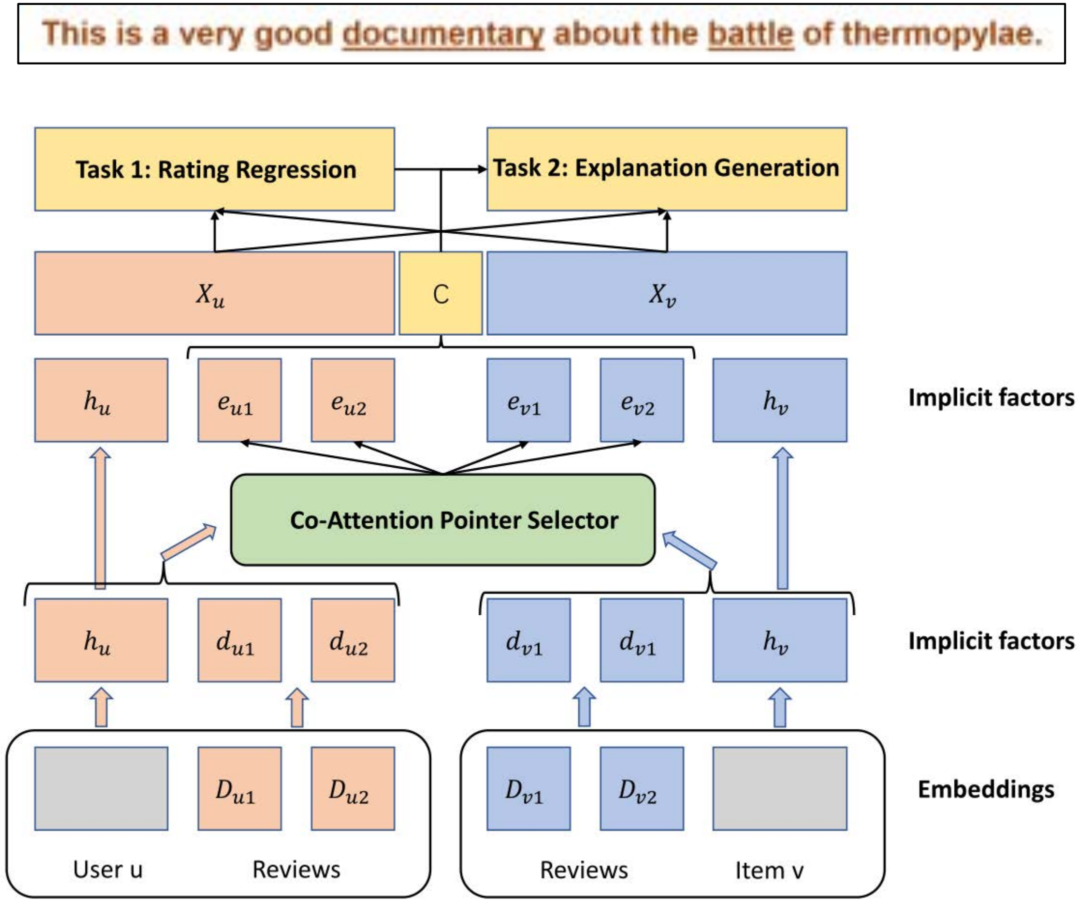


  - Post-hoc methods: provide explanations based only on the inputs, outputs and extrinsic conditions of the model



Model-intrinsic based methods

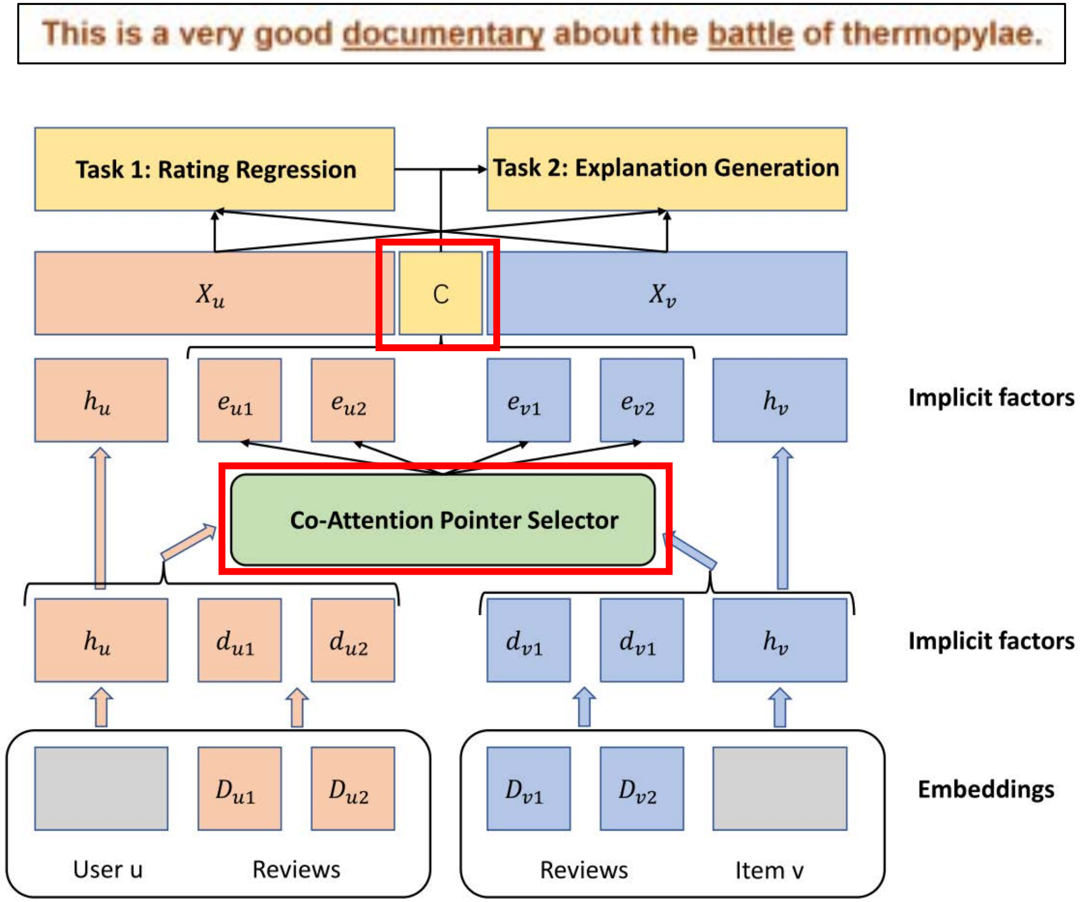
- **CAML**
  - The explanation is one of the major tasks and modeling goals
  - Only effective for the embedded models and cannot simply be reused in other models



[1] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. 2019. Co-Attentive Multi-Task Learning for Explainable Recommendation.. In IJCAI. 2137–2143.

Model-intrinsic based methods

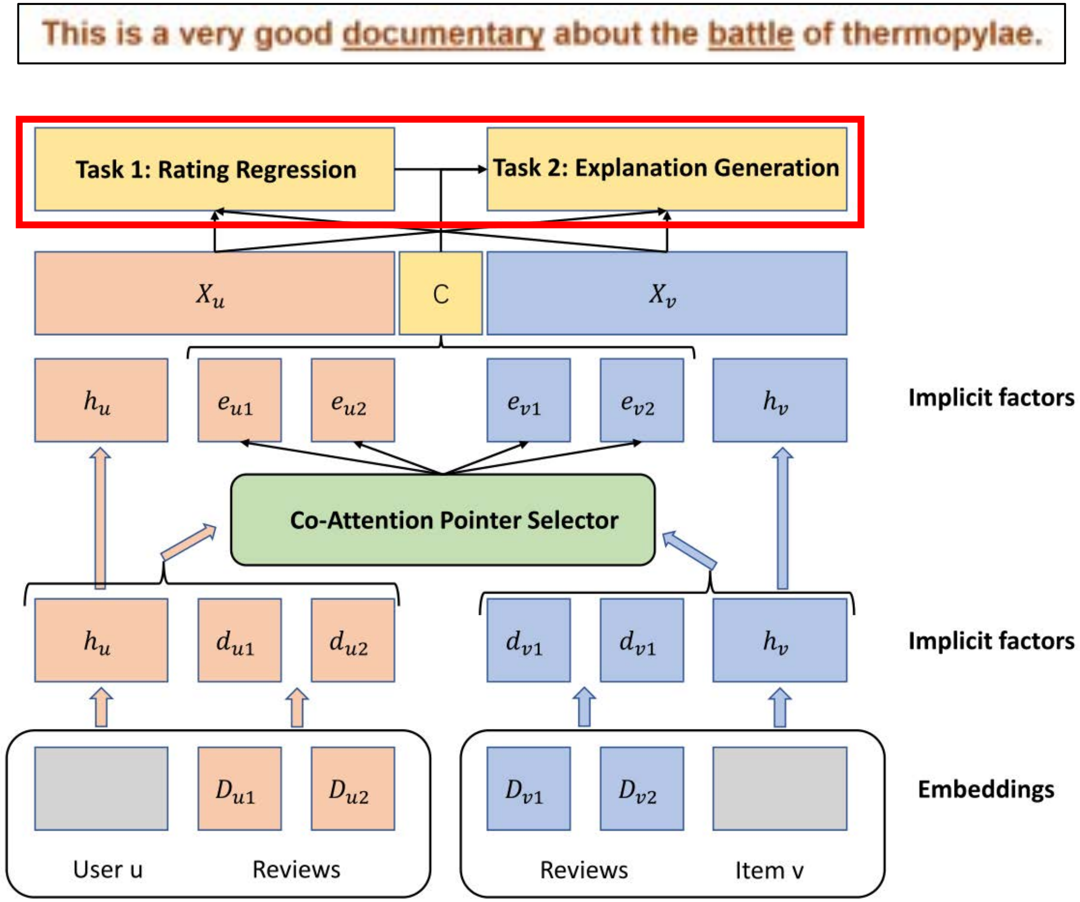
- **CAML**
  - The explanation is one of the major tasks and modeling goals
  - Only effective for the embedded models and cannot simply be reused in other models



[1] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. 2019. Co-Attentive Multi-Task Learning for Explainable Recommendation.. In IJCAI. 2137–2143.

Model-intrinsic based methods

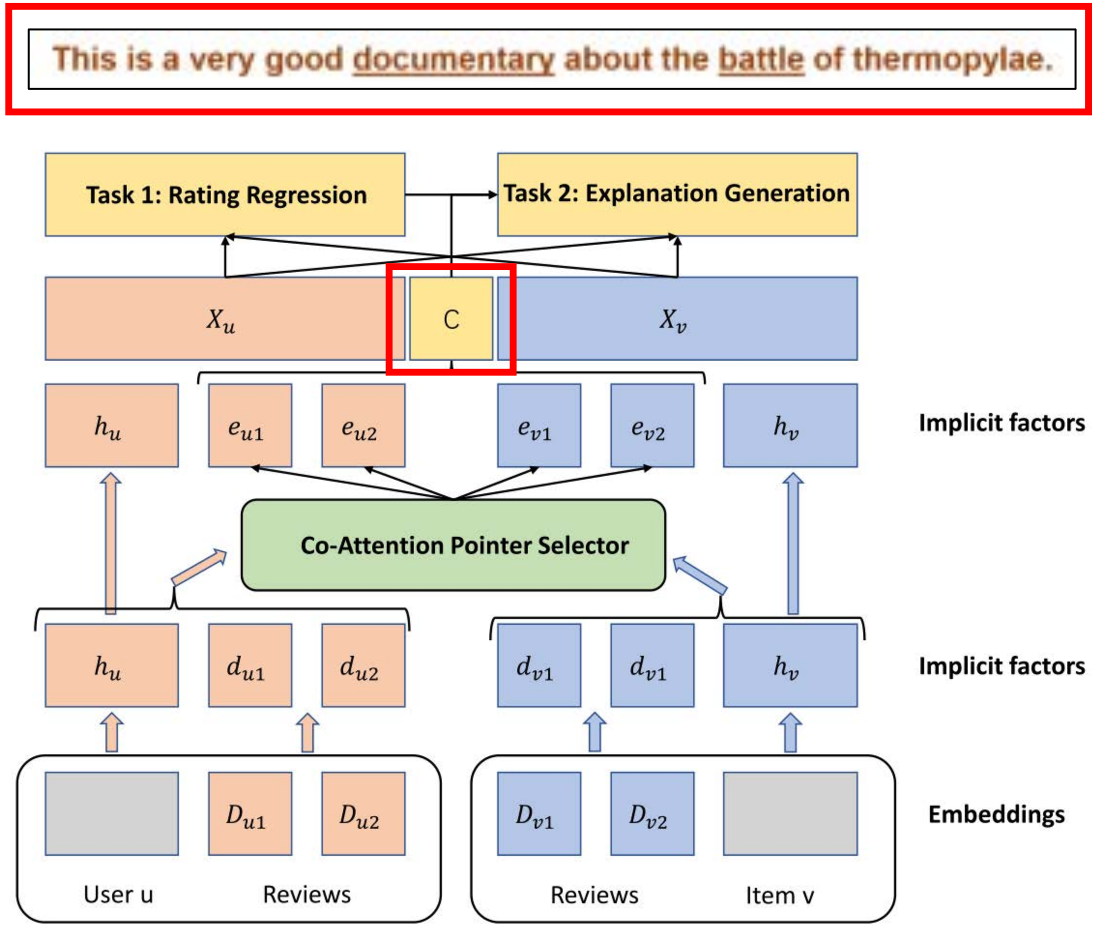
- **CAML**
  - The explanation is one of the major tasks and modeling goals
  - Only effective for the embedded models and cannot simply be reused in other models



[1] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. 2019. Co-Attentive Multi-Task Learning for Explainable Recommendation.. In IJCAI. 2137–2143.

Model-intrinsic based methods

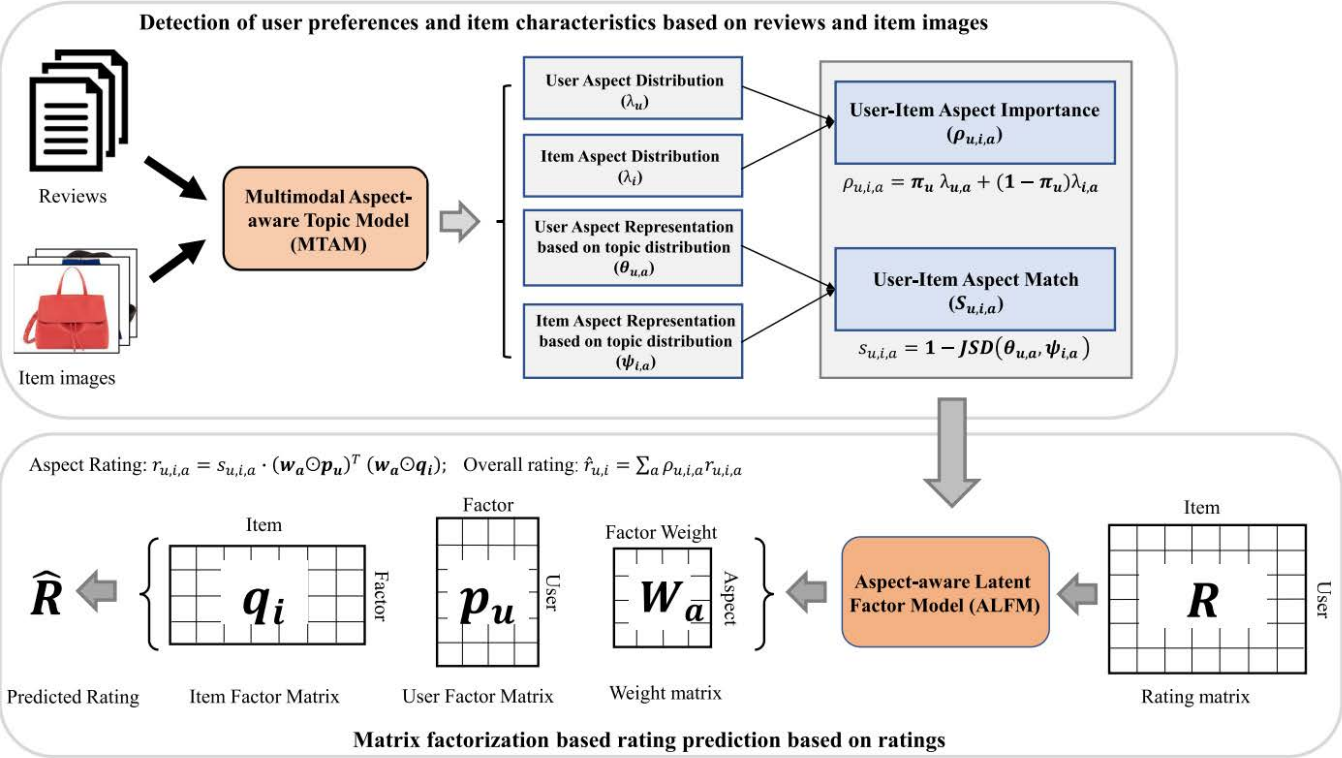
- **CAML**
  - The explanation is one of the major tasks and modeling goals
  - Only effective for the embedded models and cannot simply be reused in other models



[1] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. 2019. Co-Attentive Multi-Task Learning for Explainable Recommendation.. In IJCAI. 2137–2143.

Model-intrinsic based methods

- **MMALFM**



User_2397	Food	sauce, fried, bread, fresh, huge, flavor, shrimp, dessert, dish
	Ambience	nice, bar, atmosphere, location, friendly, inside, decor, staff, music
	Price	expensive, high, cheap, pricey, decent, pay, reasonable, priced, deal
	Service	table, server, friendly, minutes, nice, staff, asked, make, seated
	Misc.	never, give, restaurant, times, stars, friends, night, places, dinner
Item_137	Food	sauce, salad, fries, dish, cheese, dishes, burger, fresh, crab
	Ambience	bar, atmosphere, patio, area, inside, wine, small, cool, decor
	Price	price, worth, prices, better, bit, meal, sauce, dishes, quality
	Service	table, bar, friendly, wait, server, staff, minutes, beer, atmosphere
	Misc.	eat, dinner, Vegas, experience, wait, friends, times, never, give
Item_673	Food	nigiri, sake, tempura, shrimp, sauce, items, poke, crab, chef
	Ambience	atmosphere, friendly, bar, staff, inside, area, spot, monta, feel
	Price	price, worth, prices, nigiri, sake, tempura, items, lunch, special
	Service	service, table, server, friendly, minutes, staff, nice, asked, seated
	Misc.	restaurant, times, give, favorite, night, places, stars, friends, Vegas

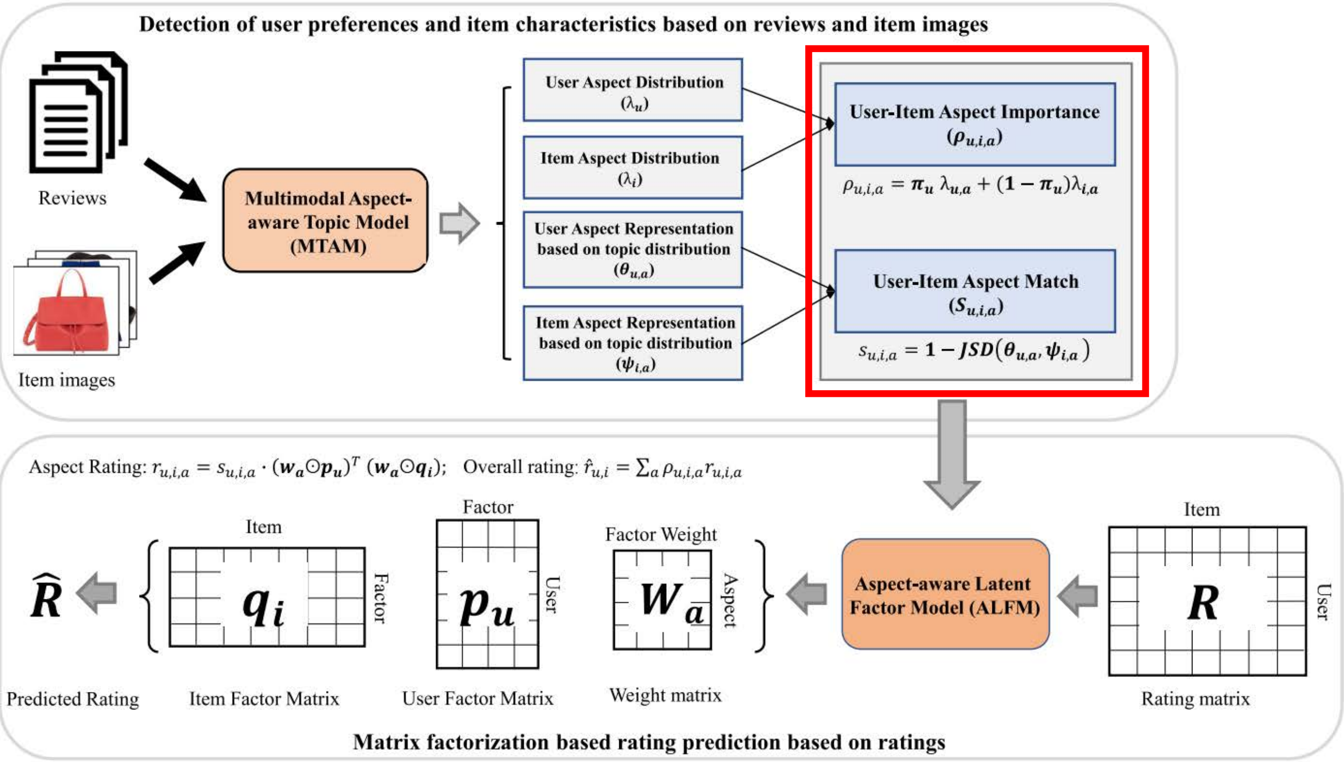
Table 6. Interpretation for Why the "User 2397" Rated "Item 137" and "Item 673" with 5 and 2, Respectively

Item	Aspect	Food	Ambience	Price	Service	Misc.
Item_137	Importance	0.3815	0.1034	0.0723	0.2038	0.2390
	Matching	0.5672	0.4523	0.5329	0.6021	0.7138
	Polarity	+	+	−	+	+
Item_673	Importance	0.3726	0.0794	0.0853	0.2076	0.2551
	Matching	0.1813	0.6535	0.4512	0.6018	0.7093
	Polarity	−	−	+	+	−

[1] Zhiyong Cheng, Xiaojun Chang, Lei Zhu, Rose C Kanjirathinkal, and Mohan Kankanhalli. 2019. MMALFM: Explainable recommendation by leveraging reviews and images. ACM Transactions on Information Systems (TOIS) 37, 2 (2019), 1–28.

Model-intrinsic based methods

- **MMALFM**



User_2397	Food	sauce, fried, bread, fresh, huge, flavor, shrimp, dessert, dish
	Ambience	nice, bar, atmosphere, location, friendly, inside, decor, staff, music
	Price	expensive, high, cheap, pricey, decent, pay, reasonable, priced, deal
	Service	table, server, friendly, minutes, nice, staff, asked, make, seated
	Misc.	never, give, restaurant, times, stars, friends, night, places, dinner
Item_137	Food	sauce, salad, fries, dish, cheese, dishes, burger, fresh, crab
	Ambience	bar, atmosphere, patio, area, inside, wine, small, cool, decor
	Price	price, worth, prices, better, bit, meal, sauce, dishes, quality
	Service	table, bar, friendly, wait, server, staff, minutes, beer, atmosphere
	Misc.	eat, dinner, Vegas, experience, wait, friends, times, never, give
Item_673	Food	nigiri, sake, tempura, shrimp, sauce, items, poke, crab, chef
	Ambience	atmosphere, friendly, bar, staff, inside, area, spot, monta, feel
	Price	price, worth, prices, nigiri, sake, tempura, items, lunch, special
	Service	service, table, server, friendly, minutes, staff, nice, asked, seated
	Misc.	restaurant, times, give, favorite, night, places, stars, friends, Vegas

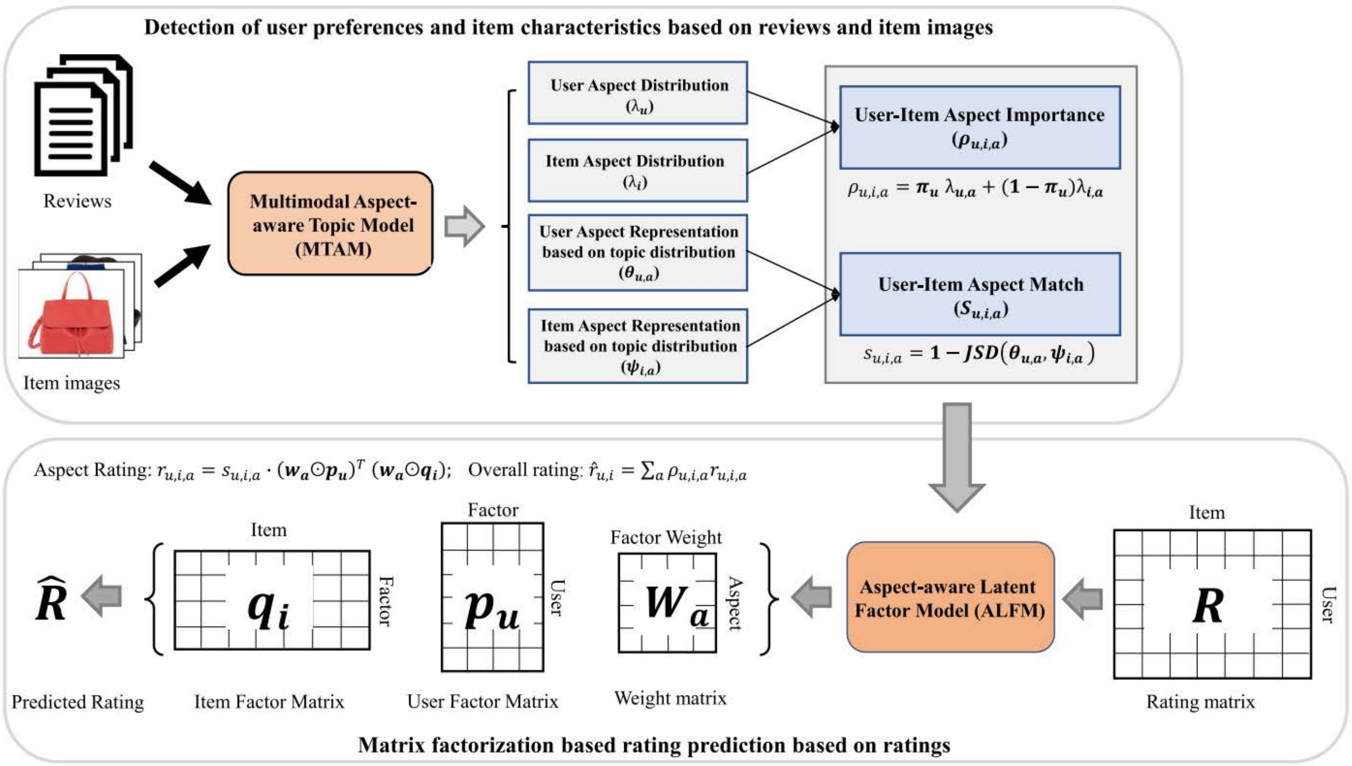
Table 6. Interpretation for Why the "User 2397" Rated "Item 137" and "Item 673" with 5 and 2, Respectively

Item	Aspect	Food	Ambience	Price	Service	Misc.
Item_137	Importance	0.3815	0.1034	0.0723	0.2038	0.2390
	Matching	0.5672	0.4523	0.5329	0.6021	0.7138
	Polarity	+	+	−	+	+
Item_673	Importance	0.3726	0.0794	0.0853	0.2076	0.2551
	Matching	0.1813	0.6535	0.4512	0.6018	0.7093
	Polarity	−	−	+	+	−

[1] Zhiyong Cheng, Xiaojun Chang, Lei Zhu, Rose C Kanjirathinkal, and Mohan Kankanhalli. 2019. MMALFM: Explainable recommendation by leveraging reviews and images. ACM Transactions on Information Systems (TOIS) 37, 2 (2019), 1–28.

Model-intrinsic based methods

- **MMALFM**



User_2397	Food	sauce, fried, bread, fresh, huge, flavor, shrimp, dessert, dish
	Ambience	nice, bar, atmosphere, location, friendly, inside, decor, staff, music
	Price	expensive, high, cheap, pricey, decent, pay, reasonable, priced, deal
	Service	table, server, friendly, minutes, nice, staff, asked, make, seated
	Misc.	never, give, restaurant, times, stars, friends, night, places, dinner
Item_137	Food	sauce, salad, fries, dish, cheese, dishes, burger, fresh, crab
	Ambience	bar, atmosphere, patio, area, inside, wine, small, cool, decor
	Price	price, worth, prices, better, bit, meal, sauce, dishes, quality
	Service	table, bar, friendly, wait, server, staff, minutes, beer, atmosphere
	Misc.	eat, dinner, Vegas, experience, wait, friends, times, never, give
Item_673	Food	nigiri, sake, tempura, shrimp, sauce, items, poke, crab, chef
	Ambience	atmosphere, friendly, bar, staff, inside, area, spot, monta, feel
	Price	price, worth, prices, nigiri, sake, tempura, items, lunch, special
	Service	service, table, server, friendly, minutes, staff, nice, asked, seated
	Misc.	restaurant, times, give, favorite, night, places, stars, friends, Vegas

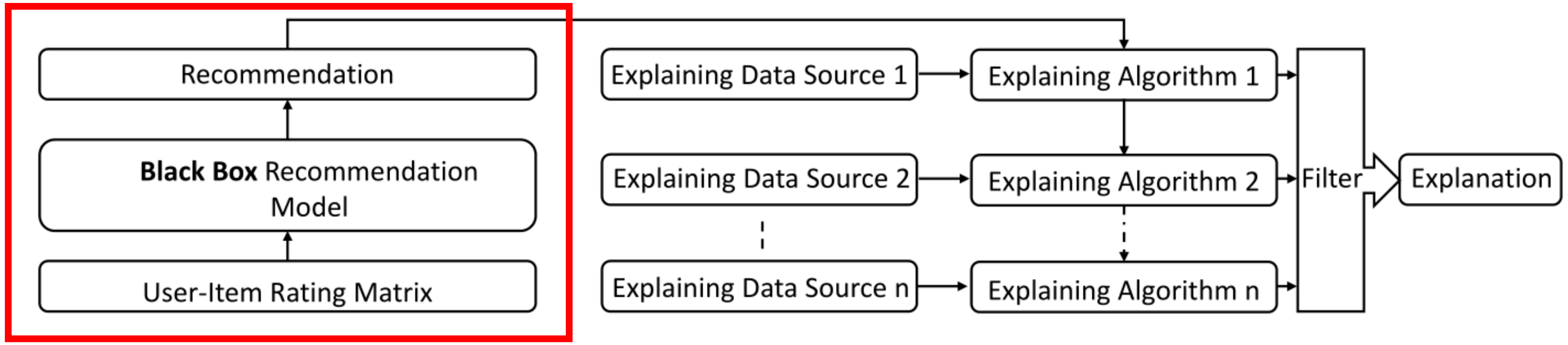
Table 6. Interpretation for Why the "User 2397" Rated "Item 137" and "Item 673" with 5 and 2, Respectively

Item	Aspect	Food	Ambience	Price	Service	Misc.
Item_137	Importance	0.3815	0.1034	0.0723	0.2038	0.2390
	Matching	0.5672	0.4523	0.5329	0.6021	0.7138
	Polarity	+	+	−	+	+
Item_673	Importance	0.3726	0.0794	0.0853	0.2076	0.2551
	Matching	0.1813	0.6535	0.4512	0.6018	0.7093
	Polarity	−	−	+	+	−

[1] Zhiyong Cheng, Xiaojun Chang, Lei Zhu, Rose C Kanjirathinkal, and Mohan Kankanhalli. 2019. MMALFM: Explainable recommendation by leveraging reviews and images. ACM Transactions on Information Systems (TOIS) 37, 2 (2019), 1–28.

Post-hoc methods

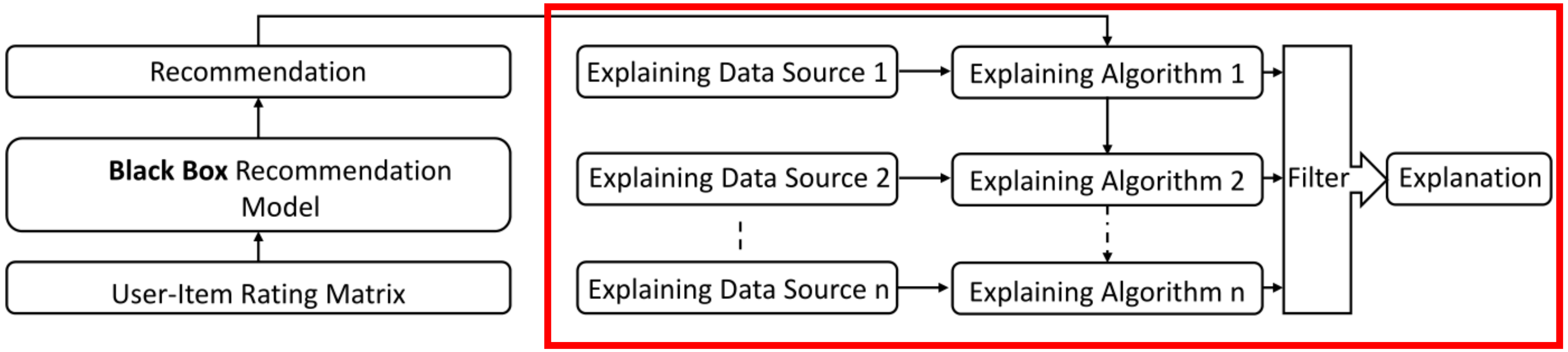
- **An example from Shmaryahu et al.**
  - It generates explanations directly from the recommendation and explaining data source



[1] Dorin Shmaryahu, Guy Shani, and Bracha Shapira. 2020. Post-hoc Explanations for Complex Model Recommendations using Simple Methods. In IntRS@ RecSys. 26–36.

Post-hoc methods

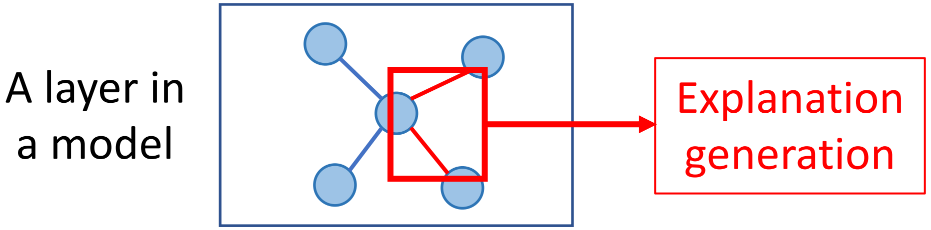
- **An example from Shmaryahu et al.**
  - It generates explanations directly from the recommendation and explaining data source



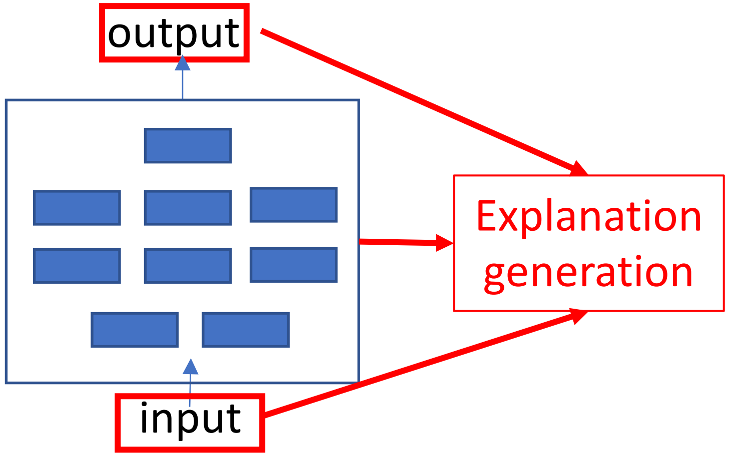
[1] Dorin Shmaryahu, Guy Shani, and Bracha Shapira. 2020. Post-hoc Explanations for Complex Model Recommendations using Simple Methods. In IntRS@ RecSys. 26–36.

Taxonomy

- **The second criteria: How the explanations are presented**
  - Structured methods: present explanations in the form of **logical reasoning** based on some particular structures, such as a graph, or a knowledge graph

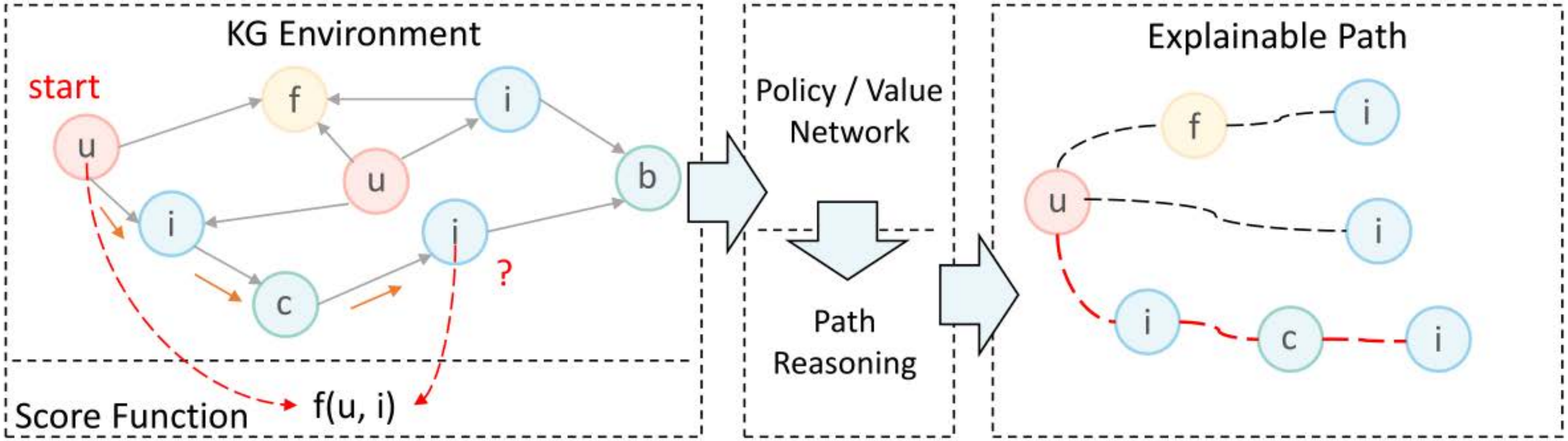


  - Unstructured methods: provide explanations based on the inputs, outputs and models, do not rely on, or explicitly rely on logical reasoning



Structured methods

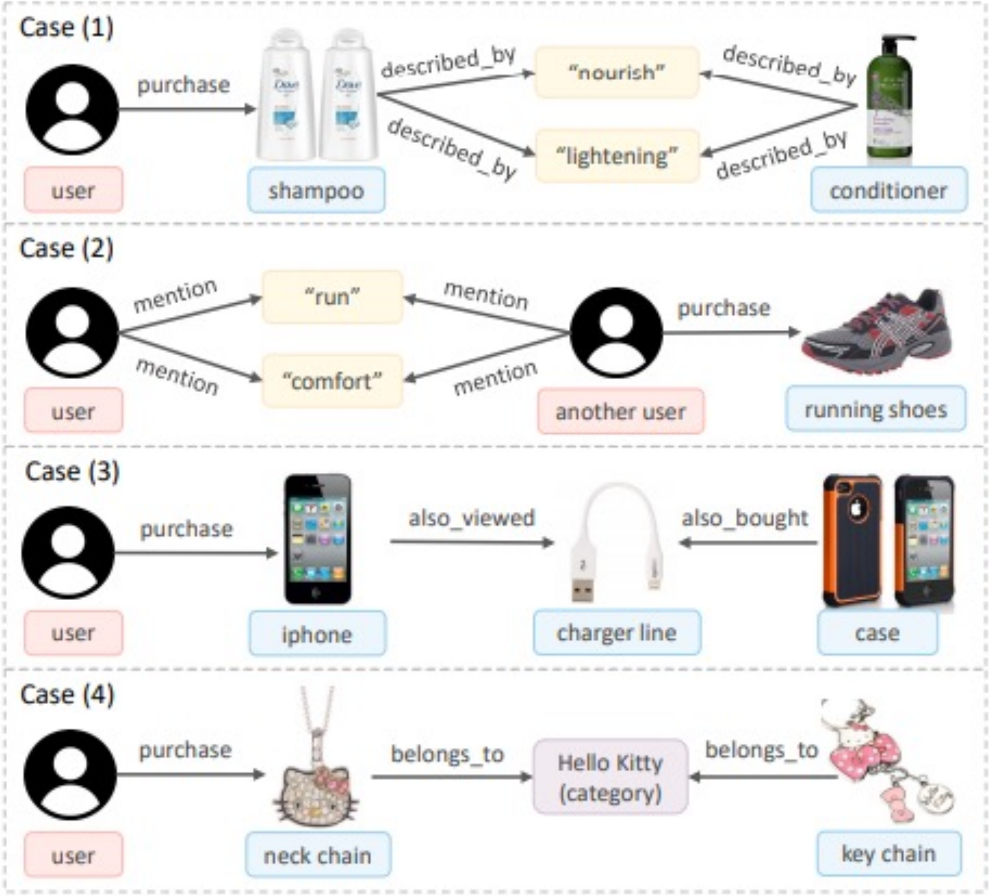
- **PGPR**
  - An explanation path graph generated with knowledge graph
  - Path definition: $p_k(e_0, e_k) = \left\{ e_0 \overset{r_1}{\leftrightarrow} e_1 \overset{r_2}{\leftrightarrow} \cdots \overset{r_k}{\leftrightarrow} e_k \right\}$



[1] Yikun Xian, Zuohui Fu, Shan Muthukrishnan, Gerard De Melo, and Yongfeng Zhang. 2019. Reinforcement knowledge graph reasoning for explainable recommendation. In Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval. 285–294.

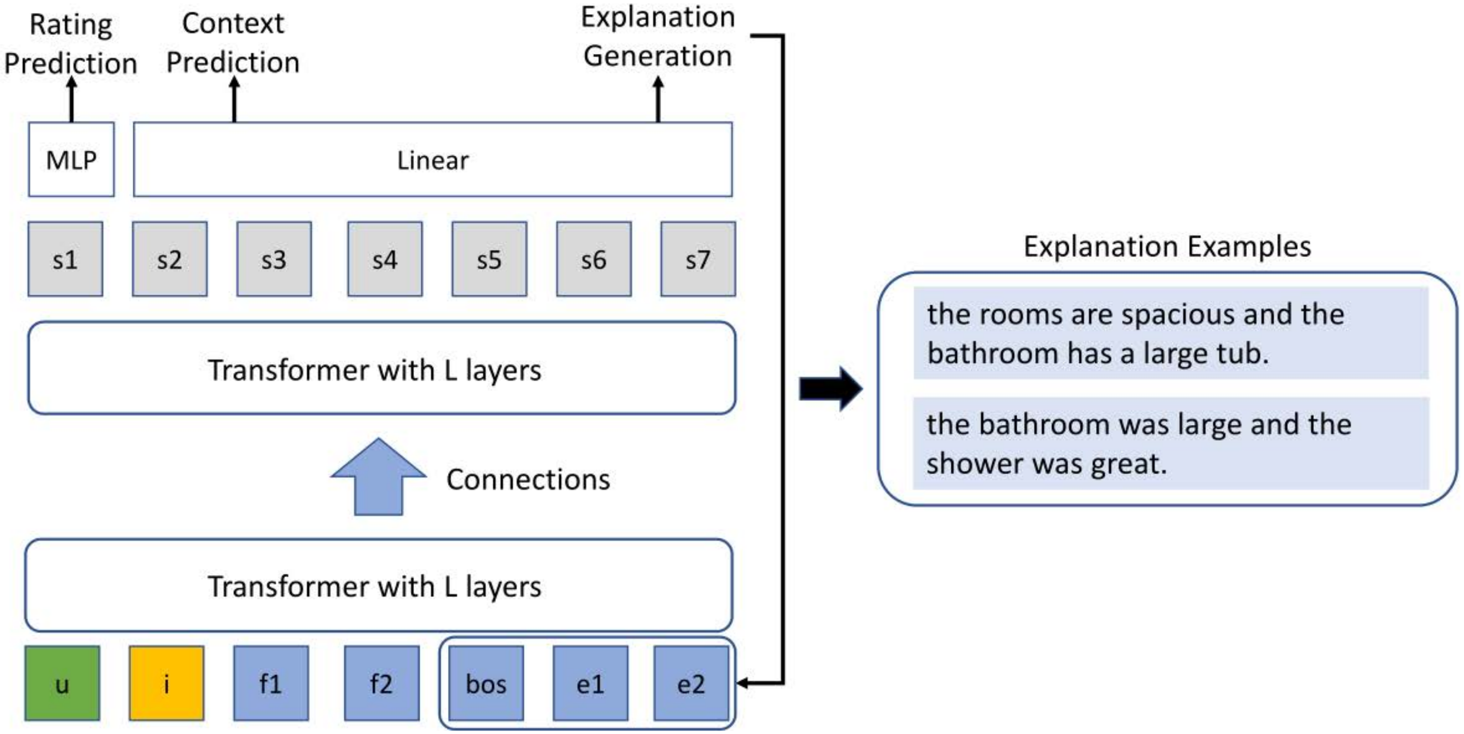
Structured methods

- **PGPR**
  - Explanation path



Unstructured methods

- **PETER**
  - Generate explanation sentence word by word
  - The final explanation is a sentence based on probability, not the sole reason deduced according to deterministic rules or structures



[1] Lei Li, Yongfeng Zhang, and Li Chen. 2021. Personalized transformer for explainable recommendation. arXiv preprint arXiv:2105.11601 (2021).

Unstructured methods

- **CountER**
  - It tries to use small changes in item aspects to reverse the decision

```
If the item had been slightly worse on [aspect(s)],
then it will not be recommended.
```

minimize Explanation Complexity
s.t., Explanation is Strong Enough

Matching-based:

Counterfactual reasoning:

[1] untao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual explainable recommendation. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management.

Unstructured methods

- **CountER**
  - It tries to use small changes in item aspects to reverse the decision

```
If the item had been slightly worse on [aspect(s)],
then it will not be recommended.
```

minimize Explanation Complexity
s.t., Explanation is Strong Enough

Matching-based:

Counterfactual reasoning:

[1] untao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual explainable recommendation. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management.

Explainability

METHODS

EVALUATIONS

APPLICATIONS

FUTURE DIRECTIONS

Taxonomy of research on evaluations

- **Evaluation perspectives**
  - Effectiveness
  - Transparency
  - Scrutability

- **Evaluation form**
  - Quantitative metrics
  - Case study
  - Real-world performance
  - Ablation Study

Taxonomy of Evaluation

- **Evaluation perspectives**
  - Effectiveness
  - Transparency
  - Scrutability

Evaluation perspective	Evaluation criteria	Related research
Effectiveness	Whether the explanations are useful to users? (e.g. Decision making, Recommendation results)	[8, 58, 337]
Transparency	Whether the explanations can reveal the working principles of the model?	[18, 144, 225]
Scrutability	Whether the explanations contribute to the prediction of the model?	[327, 347, 362]

Reference: Nava Tintarev and Judith Masthoff. 2011. Designing and evaluating explanations for recommender systems. In Recommender systems handbook. Springer, 479–510.

Taxonomy of Evaluation

- **Evaluation form**
  - **Quantitative:** ROUGE score, BLEU, USR, FMR...
  - **Case study:** Whether the explanation conforms to human logic
  - **Real-world performance:** The practical effects of the explanation
  - **Ablation study:** How algorithmic modules provide explanations and how these modules enhance the recommendation model

Evaluation form	Corresponding perspectives	Related research
Quantitative metrics	Effectiveness; Scrutability	[337, 338]
Case study	Effectiveness; Transparency	[225, 362, 396]
Real-world performance	Effectiveness; Scrutability; Transparency	[58, 347, 392]
Ablation Study	Effectiveness; Transparency	[64, 211, 327]

Explainability

METHODS

EVALUATIONS

APPLICATIONS

FUTURE DIRECTIONS

E-commercial Recommendation

Social Media

Explainability

METHODS

EVALUATIONS

APPLICATIONS

FUTURE DIRECTIONS

Natural Language Generation

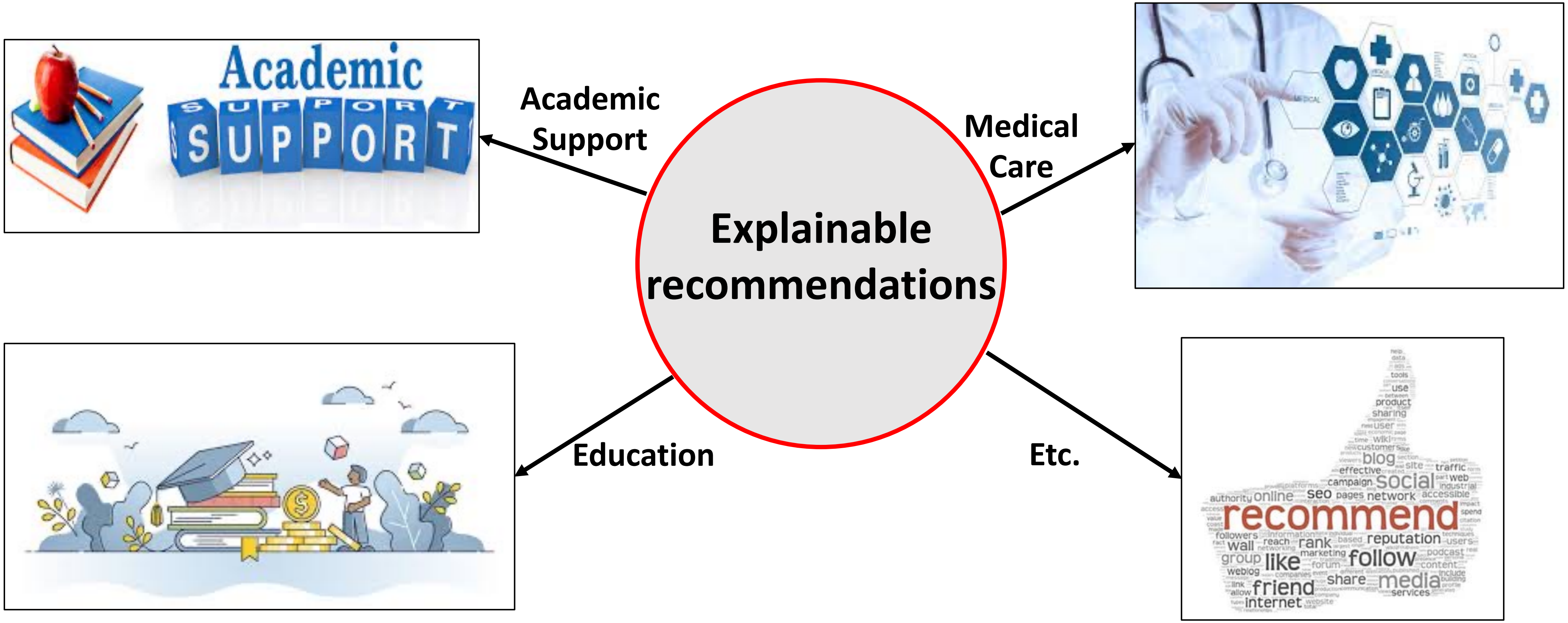
- **Templated based (now)**

I recommend Iron Man to you because you've seen The Avengers

- **Full paragraph interpretation generation (currently exist but their effectiveness has yet to improve)**

Since you've seen movies like The Avengers, and your recent interest is in the TV series, we recommend something similar for you: Agents of S.H.I.E.L.D.

Explainable recommendations in more fields



Summary

- **Concept of explainability in Rec**
  - The ability to explain or to present in understandable terms to a human
- **Taxonomy of methods**
  - How to produce explanations: model-intrinsic based (mostly used) or post-hoc
  - How the explanations are presented: structured or unstructured
- **Taxonomy of evaluations**
  - Evaluation perspectives: Effectiveness, Transparency, Scrutability
  - Evaluation forms: Quantitative, Case study, Real-world performance, Ablation study
- **Application**
  - E-commercial Recommendation
  - Social Media
- **Future directions**
  - Natural Language Generation for Explanation
  - Explainable recommendations in more fields

香港城市大學
City University of Hong Kong

THE HONG KONG POLYTECHNIC UNIVERSITY
香港理工大學

Trustworthy Recommender Systems

Wenqi Fan[1], Xiangyu Zhao[2], Lin Wang[1], Xiao Chen[1], Jingtong Gao[2], Qidong Liu[2], Shijie Wang[1]
[1]The Hong Kong Polytechnic University
[2]City University of Hong Kong

Coffee Break time, we will be back in 10-15 minutes

Website (Slides): https://advanced-recommender-systems.github.io/trustworthiness-tutorial/
Survey: A Comprehensive Survey on Trustworthy Recommender Systems, arXiv:2209.10117, 2022.

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

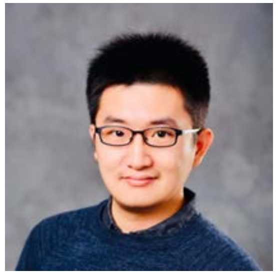
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Privacy

The era of big data

- ❑ Modern recommender systems, heavily rely on big data and even private data to train algorithms for obtaining high-quality recommendation performance.
- ❑ This raises huge concerns about the safety of private and sensitive data when recommendation algorithms are applied to safety-critical tasks such as finance and healthcare.

Privacy

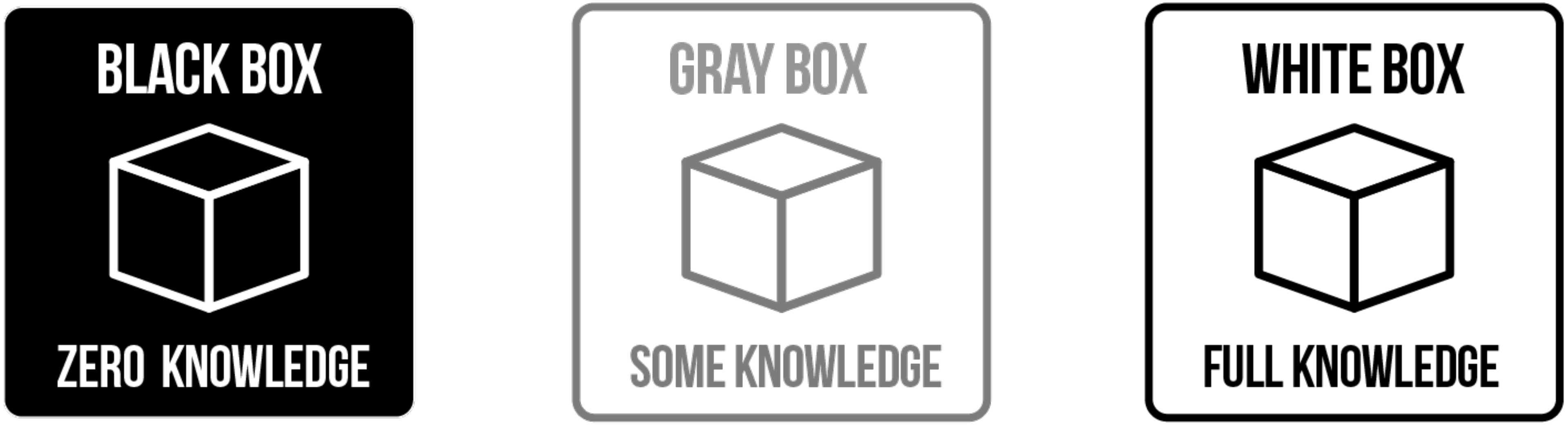
- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods.
- Applications
- Survey and Tools
- Future Directions

Privacy

- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods
- Applications
- Survey and Tools
- Future Directions

Privacy Attacks

Privacy Attacks aim to steal knowledge that is not intended to be shared, such as the sensitive information of users and model parameters.



Privacy Attacks

Privacy Attacks aim to steal knowledge that is not intended to be shared, such as the sensitive information of users and model parameters.

- Membership Inference Attacks (MIA)
- Property Inference Attacks (PIA)
- Reconstruction Attacks (RA)
- Model Extraction Attacks (MEA)

Privacy Preserving

Privacy Preserving, in order to defend against privacy attacks, privacy-preserving methods have been proposed based on different strategies, which can be broadly divided into five categories:

- Differential Privacy (DP)
- Federated Learning (FL)
- Adversarial Learning (AL)
- Anonymization
- Encryption

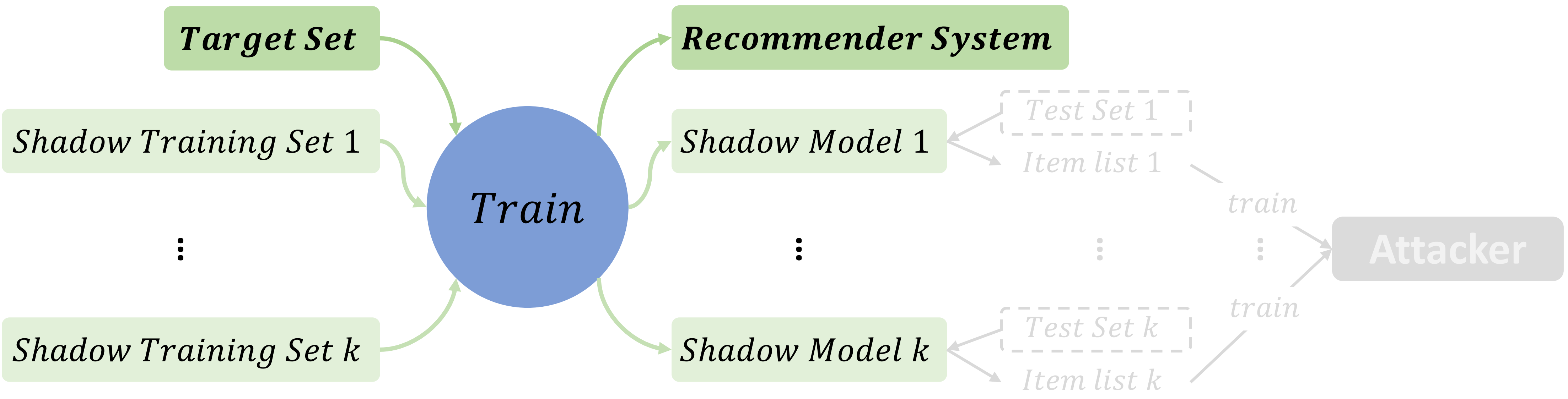
Privacy

- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods
- Applications
- Survey and Tools
- Future Directions

Privacy Attack Methods

	Taxonomy	Related methods
Privacy Attacks	Membership Inference Attacks	[79, 431]
	Property Inference Attacks	[14, 115, 277, 437]
	Reconstruction Attacks	[42, 90, 151, 257, 257, 303]
	Model Extraction Attacks	[418]

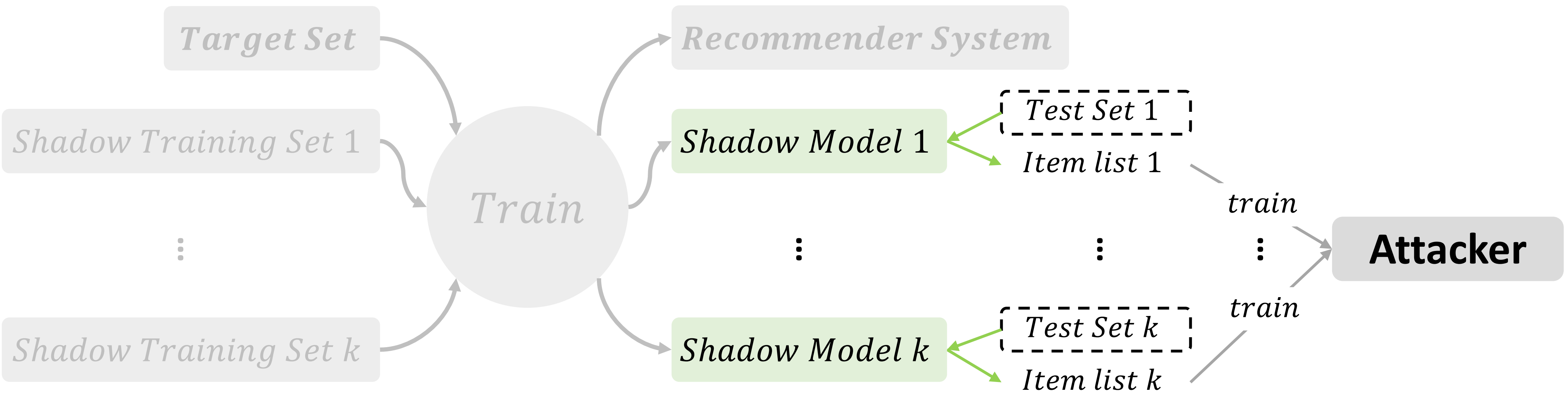
Membership Inference Attacks



Shadow training

Shokri R, et al. Membership inference attacks against machine learning models[C]// IEEE SP 2017.

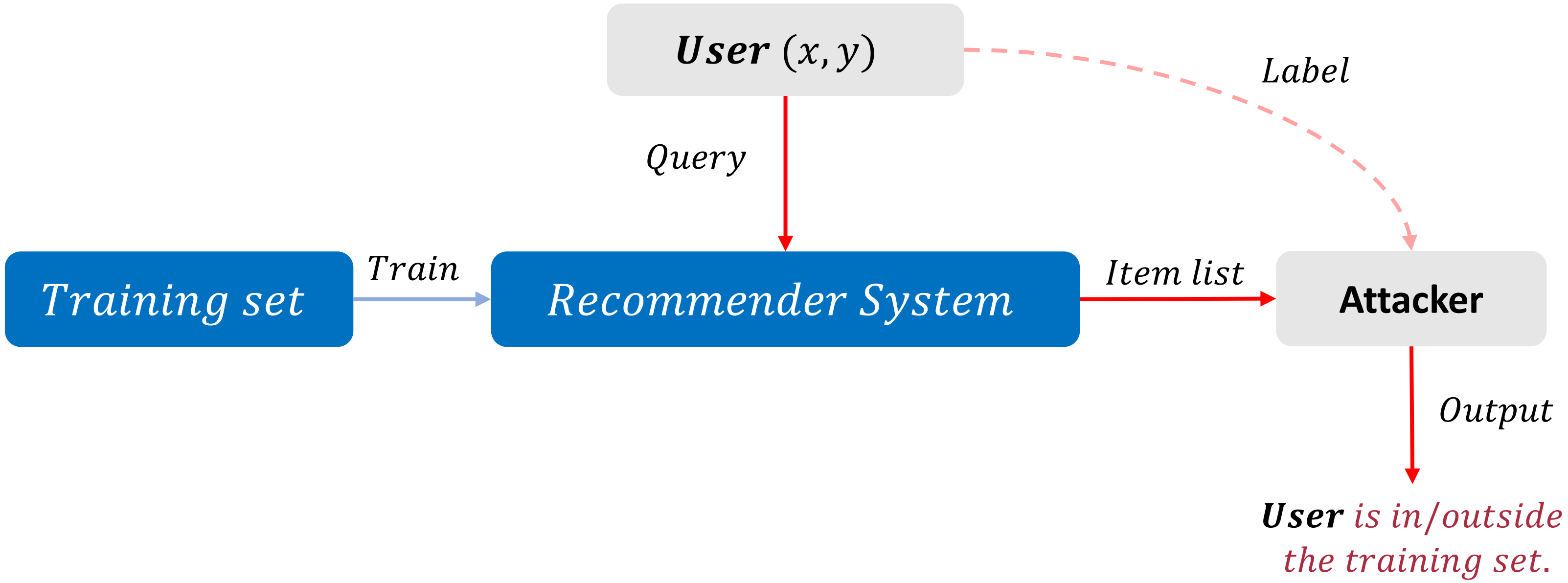
Membership Inference Attacks



Shadow training

Shokri R, et al. Membership inference attacks against machine learning models[C]// IEEE SP 2017.

Membership Inference Attacks



Membership Inference Attack

Shokri R, et al. Membership inference attacks against machine learning models[C]// IEEE SP 2017.

Membership Inference Attacks

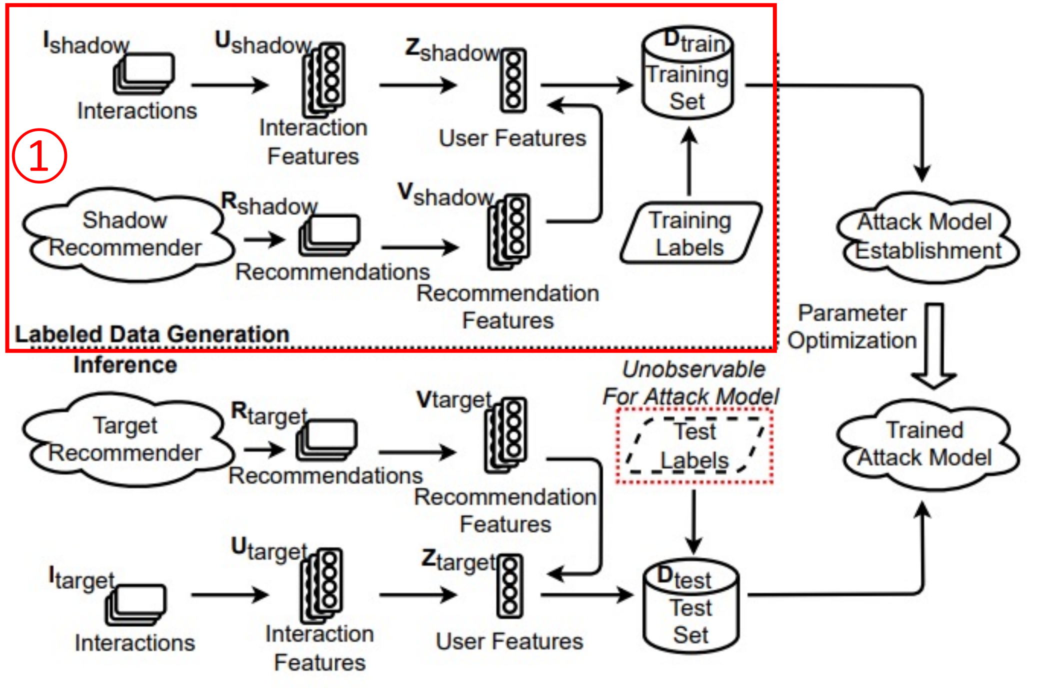


Figure 2: The framework of the membership inference attack against a recommender system.

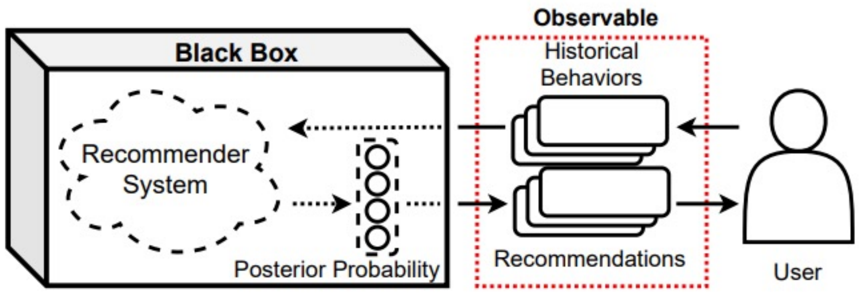


Figure 1: An example of recommender systems.

Membership Inference Attacks in Recommender Systems

Zhang M, et al. Membership inference attacks against recommender systems[C]//SIGSAC 2021.

Membership Inference Attacks

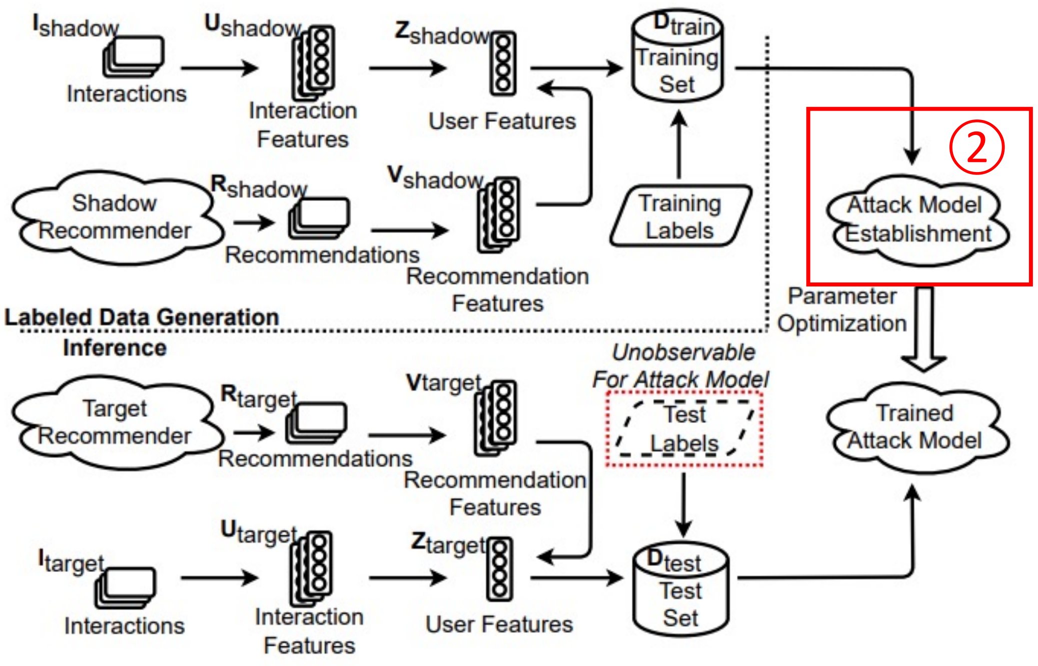


Figure 2: The framework of the membership inference attack against a recommender system.

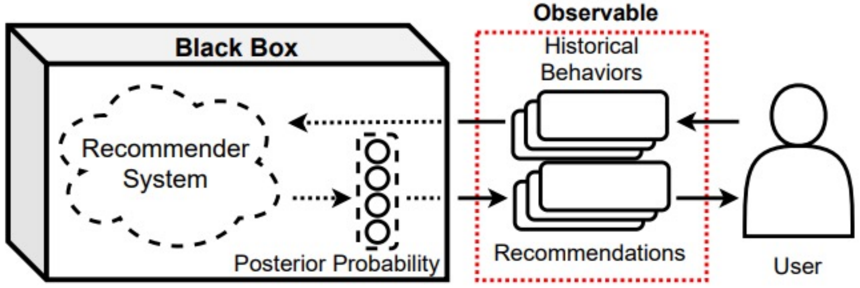


Figure 1: An example of recommender systems.

Membership Inference Attacks in Recommender Systems

Zhang M, et al. Membership inference attacks against recommender systems[C]//SIGSAC 2021.

Membership Inference Attacks

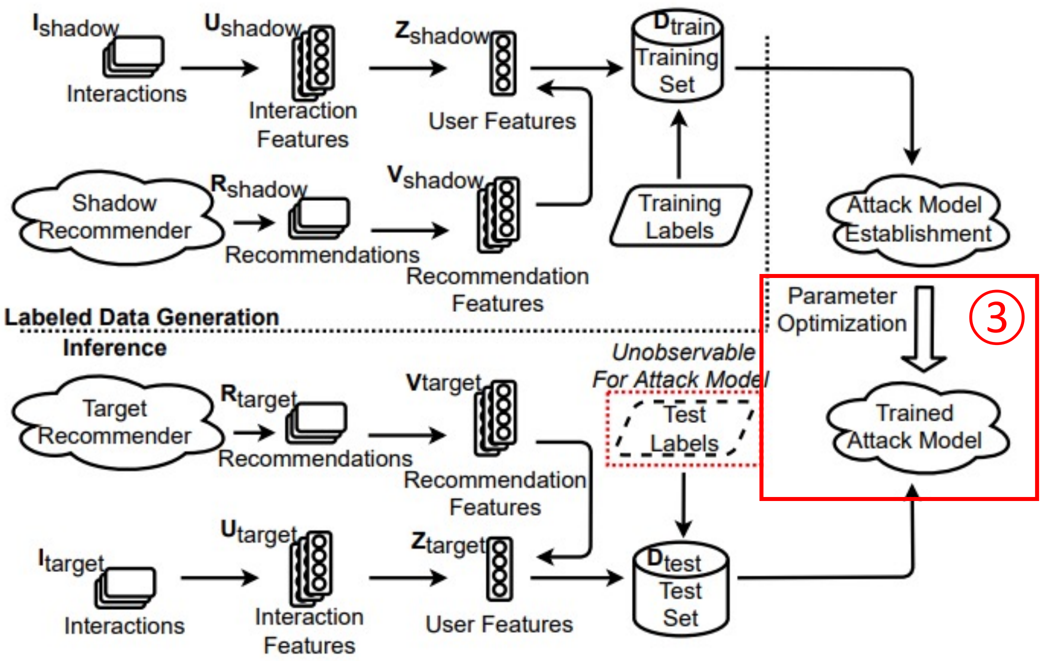


Figure 2: The framework of the membership inference attack against a recommender system.

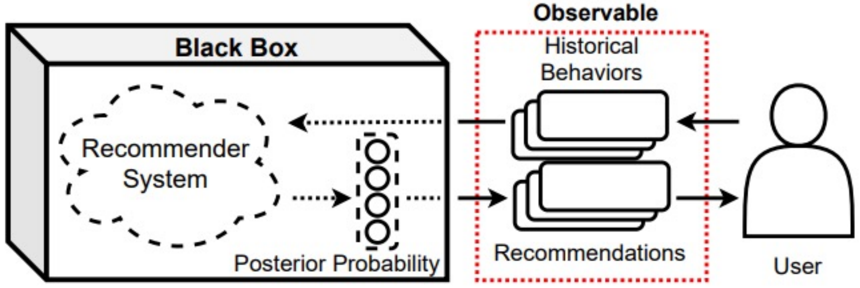
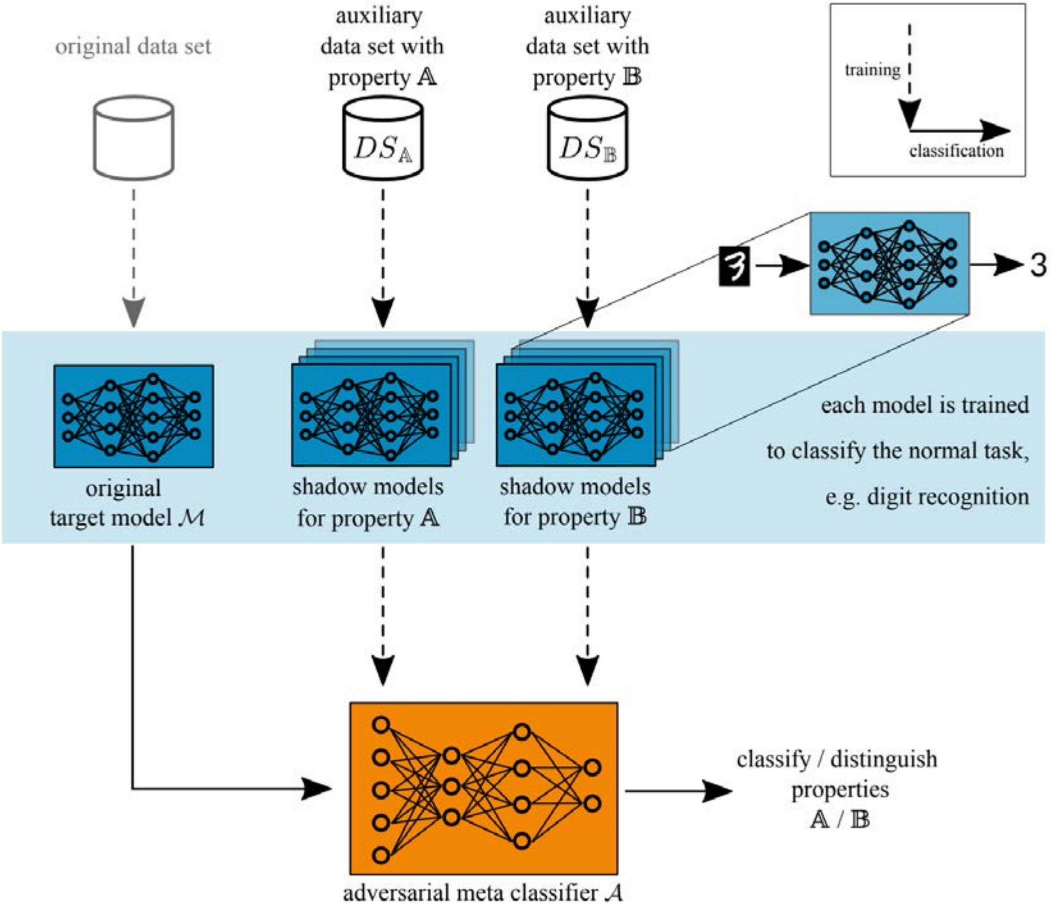


Figure 1: An example of recommender systems.

Membership Inference Attacks in Recommender Systems

Zhang M, et al. Membership inference attacks against recommender systems[C]//SIGSAC 2021.

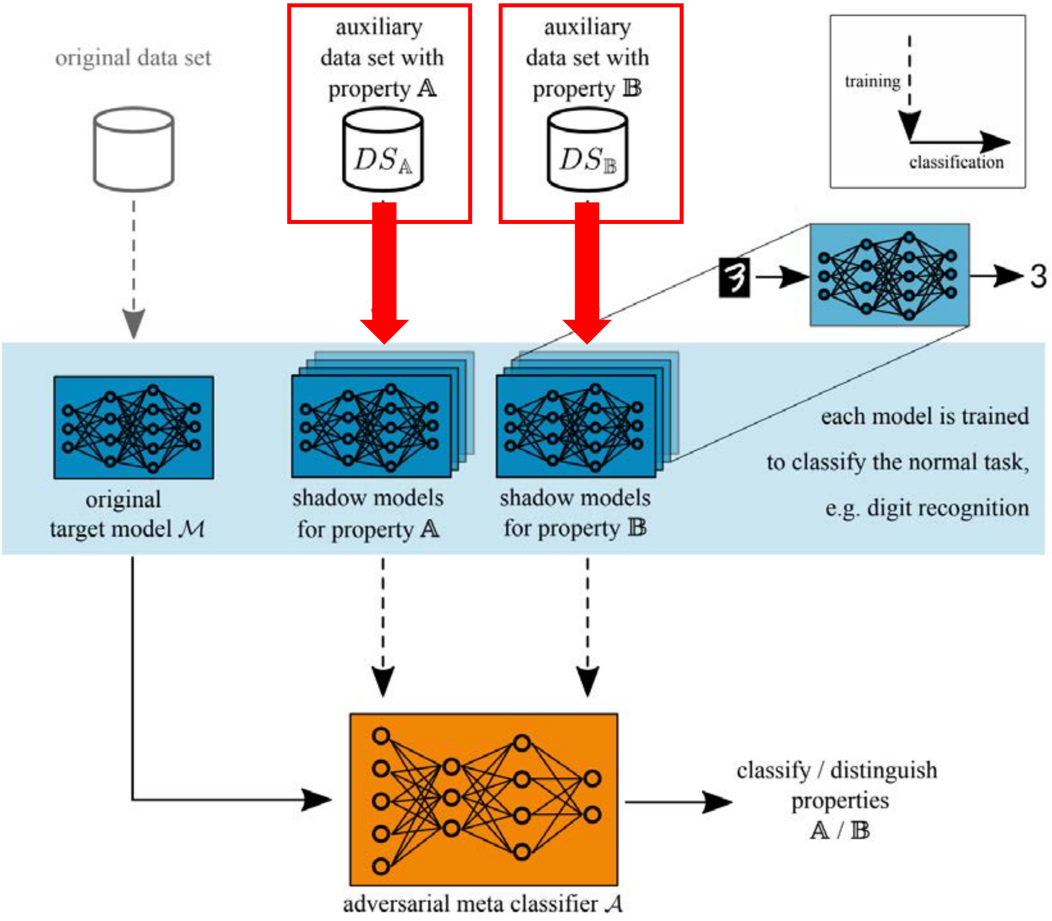
Property Inference Attacks



Using the auxiliary data with different property to train series shadow models.

Stock J, et al. Property Unlearning: A Defense Strategy Against Property Inference Attacks[J]. arXiv, 2022.

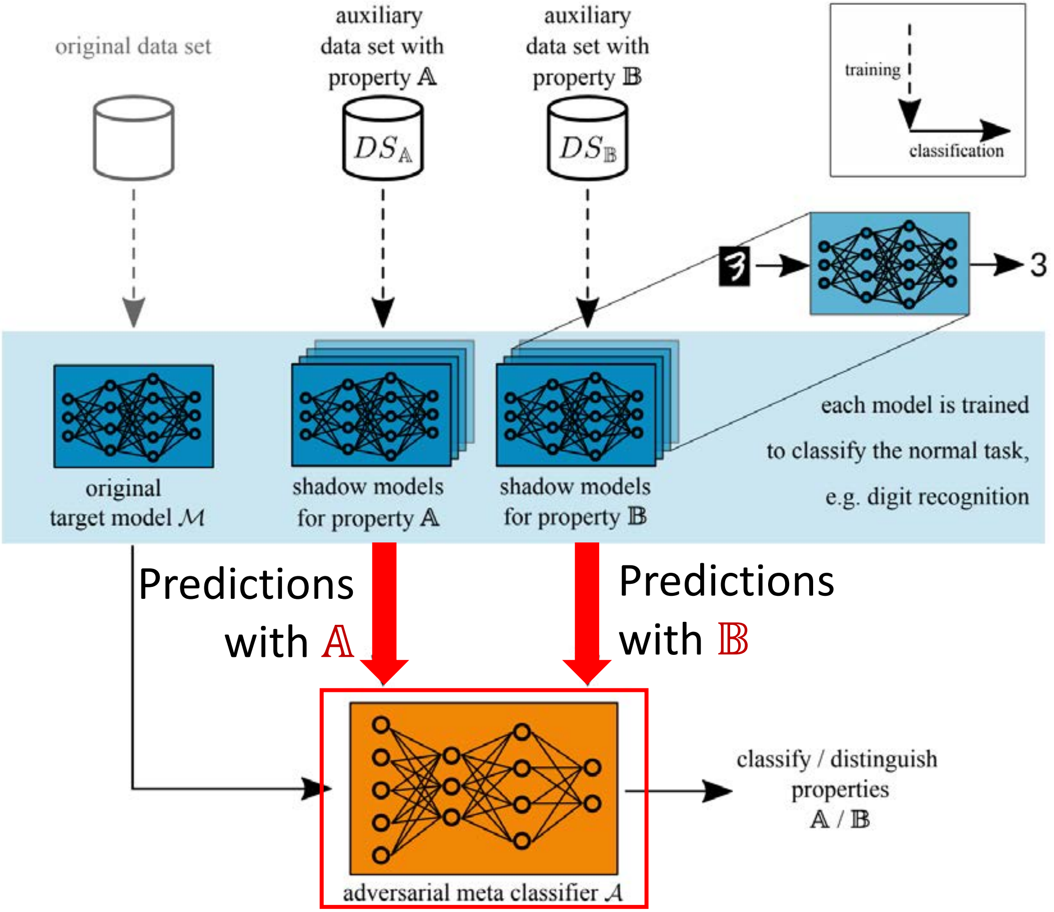
Property Inference Attacks



Using the auxiliary data with different property to train series shadow models.

Stock J, et al. Property Unlearning: A Defense Strategy Against Property Inference Attacks[J]. arXiv, 2022.

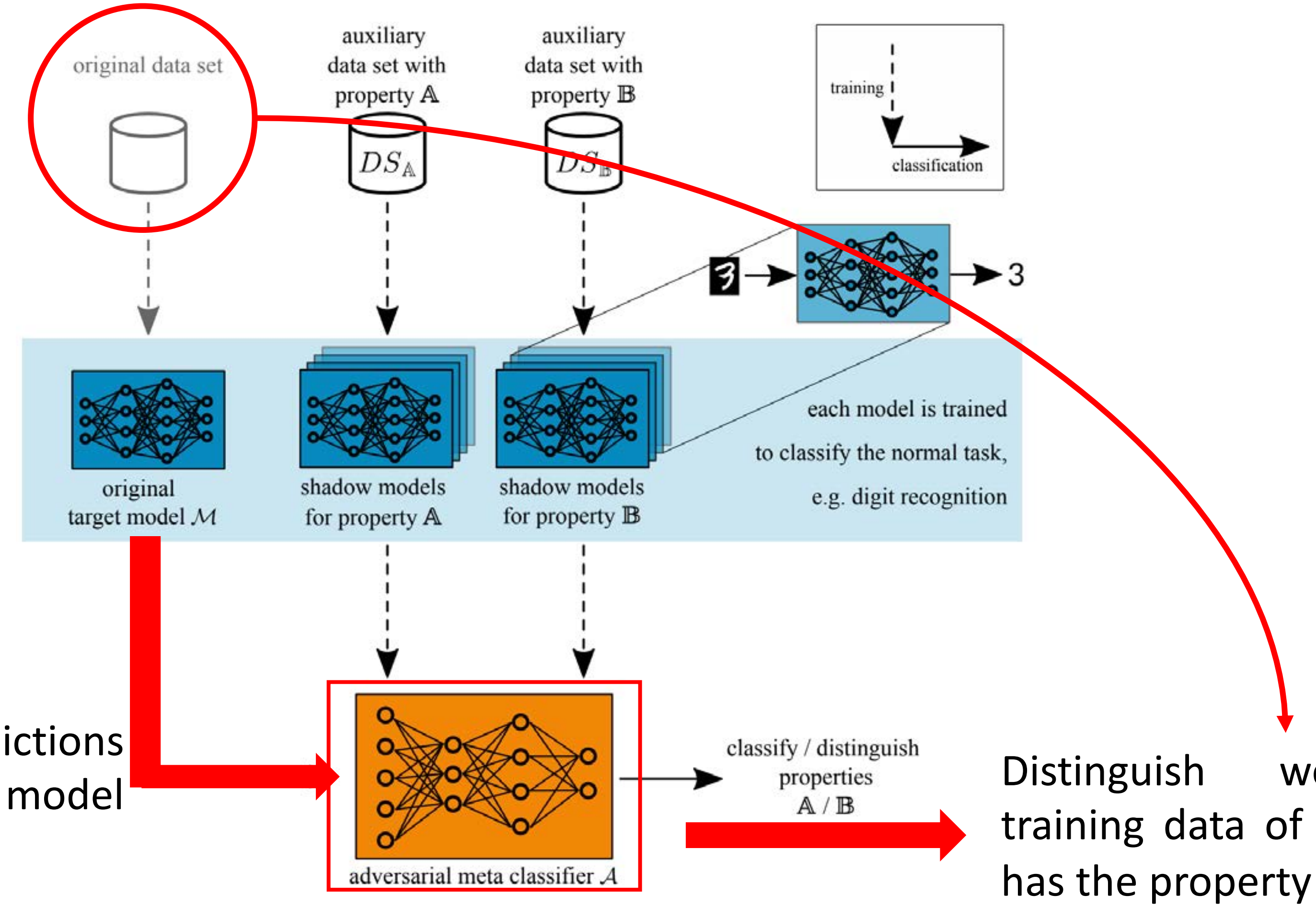
Property Inference Attacks



The predictions of the shadow models are used to train a classifier.

Stock J, et al. Property Unlearning: A Defense Strategy Against Property Inference Attacks[J]. arXiv, 2022.

Property Inference Attacks

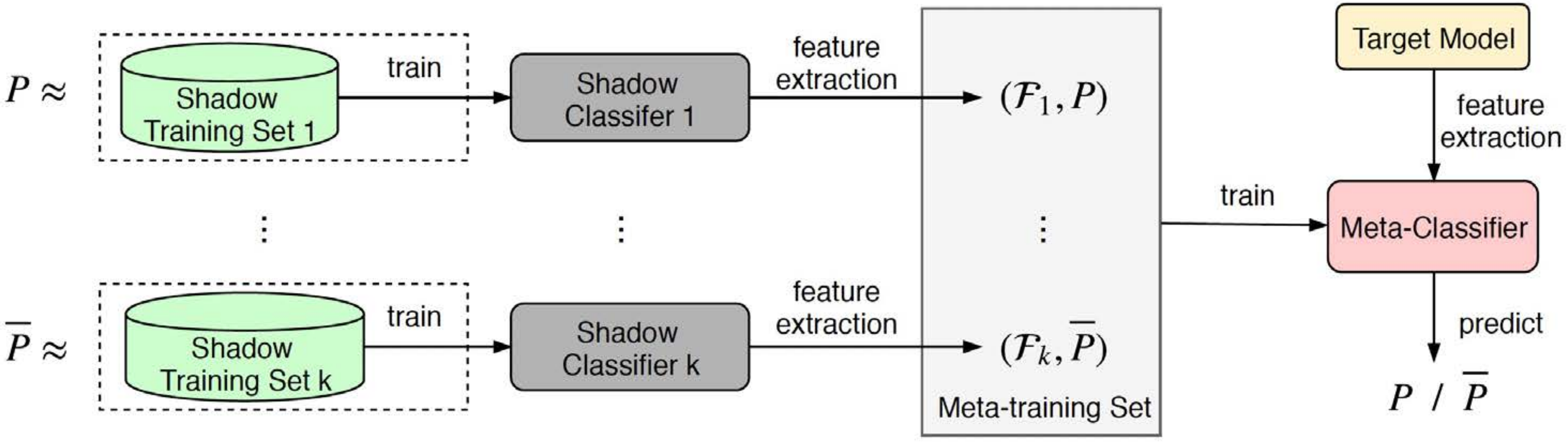


Predictions of target model

Distinguish weather the training data of target model has the property $\mathbb{A}/\mathbb{B}$ or not.

Stock J, et al. Property Unlearning: A Defense Strategy Against Property Inference Attacks[J]. arXiv, 2022.

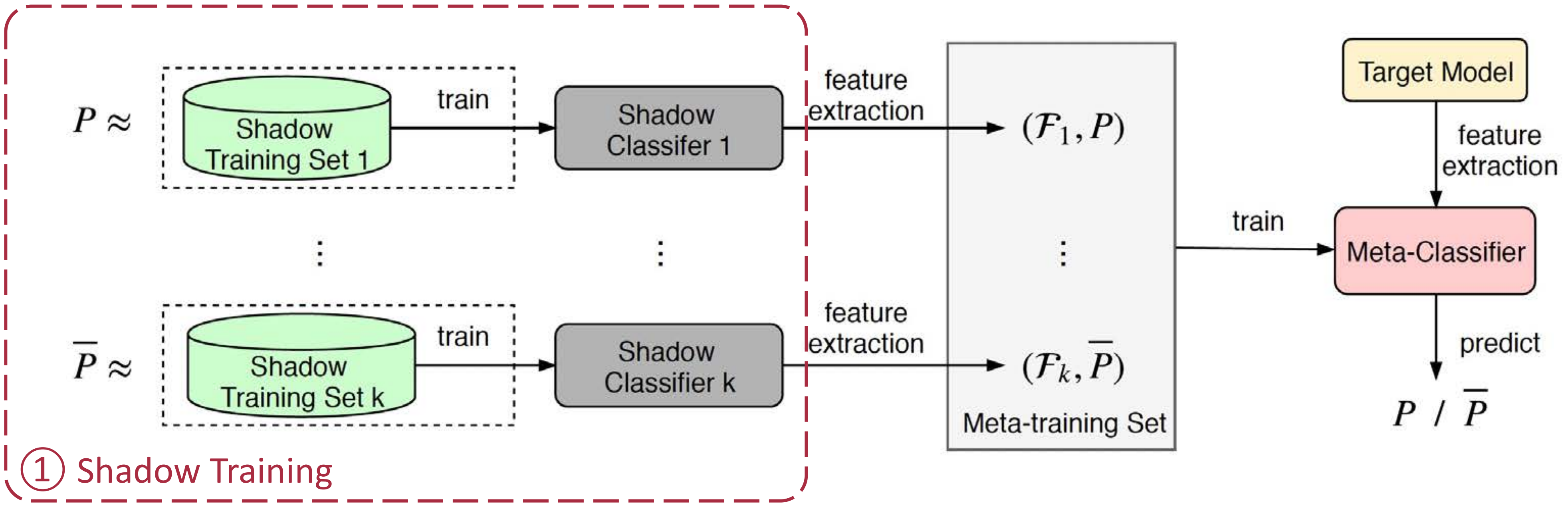
Property Inference Attacks



The workflow of the property inference attack

Ganju K, et al. Property inference attacks on fully connected neural networks using permutation invariant representations[C] 2018.

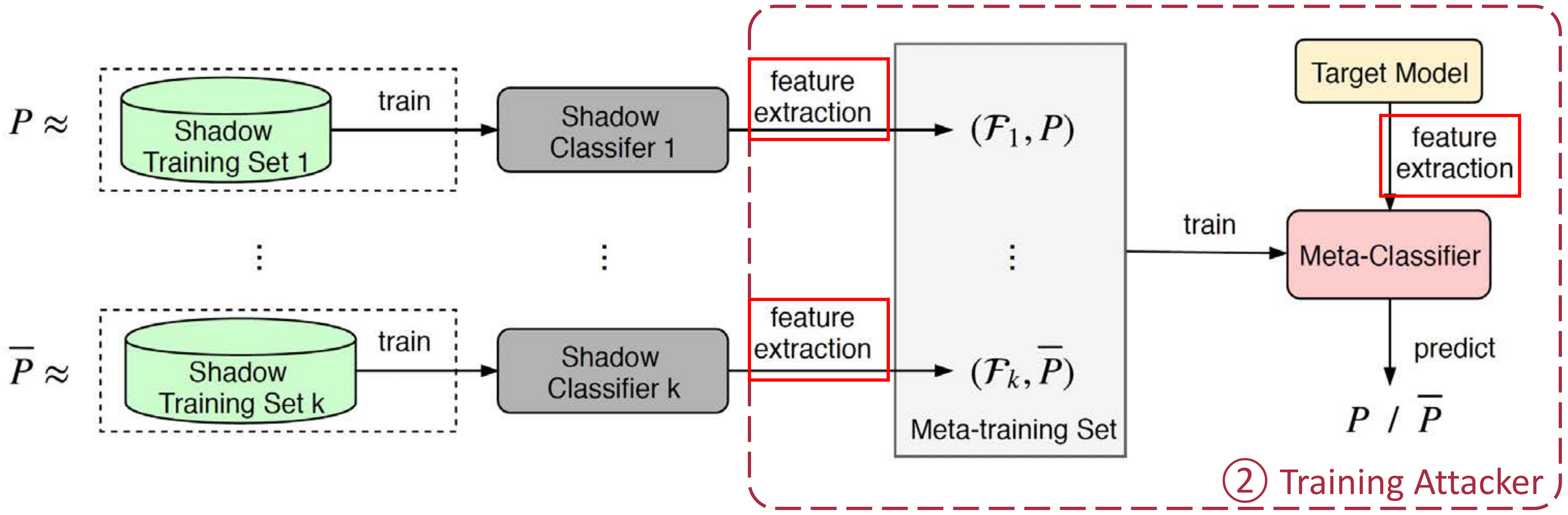
Property Inference Attacks



The workflow of the property inference attack

Ganju K, et al. Property inference attacks on fully connected neural networks using permutation invariant representations[C] 2018.

Property Inference Attacks



The workflow of the property inference attack

Ganju K, et al. Property inference attacks on fully connected neural networks using permutation invariant representations[C] 2018.

Property Inference Attacks

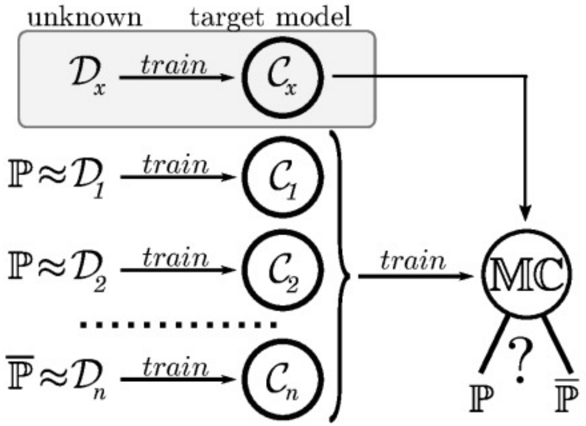


Fig. 1. Attack methodology: the target training set $\mathcal{D}_x$ produced $\mathcal{C}_x$. Using several training sets $\mathcal{D}_1, \ldots, \mathcal{D}_n$ with or without a specific property, we build $\mathcal{C}_1, \ldots, \mathcal{C}_n$, namely the training set for the meta-classifier $\mathbb{MC}$ that will classify $\mathcal{C}_x$.

Input:
$\boldsymbol{\mathcal{D}}$: the array of training sets
$\boldsymbol{l}$: the array of labels, where each $l_i \in \{\mathbb{P}, \overline{\mathbb{P}}\}$
Output: The meta-classifier $\mathbb{MC}$

TrainMC($\boldsymbol{\mathcal{D}}$,$\boldsymbol{l}$)
begin
  $\mathcal{D}_\mathcal{C} = \{\emptyset\}$
  foreach $\mathcal{D}_i \in \boldsymbol{\mathcal{D}}$ **do**
    $\mathcal{C}_i \leftarrow \text{train}(\mathcal{D}_i)$
    $\mathcal{F}_{\mathcal{C}_i} \leftarrow \text{getFeatureVectors}(\mathcal{C}_i)$
    foreach $\boldsymbol{a} \in \mathcal{F}_{\mathcal{C}_i}$ **do**
      $\mathcal{D}_\mathcal{C} = \mathcal{D}_\mathcal{C} \cup \{\boldsymbol{a}, l_i\}$
    end
  end
  $\mathbb{MC} \leftarrow \text{train}(\mathcal{D}_\mathcal{C})$
  return $\mathbb{MC}$
end

Algorithm 1: Training of the meta-classifier

Using the shadow training to train a meta-classifier(attacker)

Ateniese G, et al. Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers[J]. Int. J. Netw. Secur, 2015.

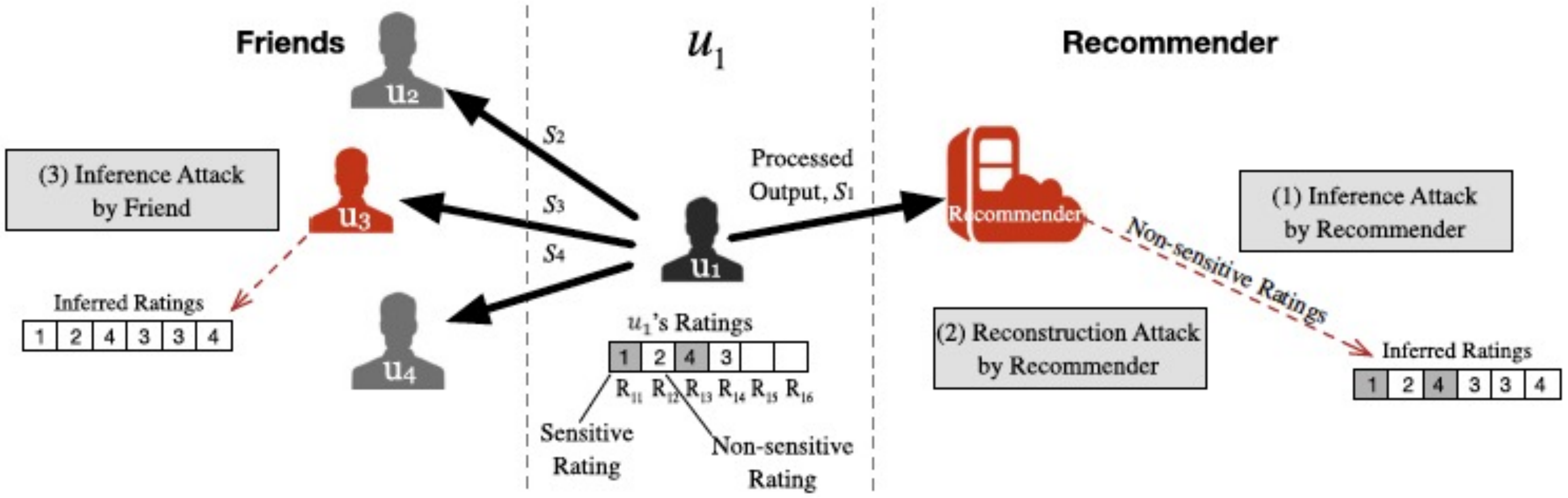
Reconstruction Attacks

Recover the face image given the person’s name and the class confidence of a facial recognition system

Fredrikson, Matt, et al. "Model inversion attacks that exploit confidence information and basic countermeasures." 2015.

Reconstruction Attacks

Reconstruction attacks in recommender systems



Using the social, public information to reconstruct the **sensitive items** of the user.

Meng X, et al. Towards privacy preserving social recommendation under personalized privacy settings. WWW 2019.

Reconstruction Attacks

Reconstruction attacks in recommender systems

Algorithm 1: RelatedItemsListInference

Input: Set of target items $\mathcal{T}$, set of auxiliary items $\mathcal{A}$, scoring function : $\mathbb{R}^{|\mathcal{A}|} \rightarrow \mathbb{R}$
Output: Subset of items from $\mathcal{T}$ which are believed by the attacker to have been added to the user's record

$inferredItems = \{\}$
foreach *observation time* τ **do**
  Δ = observation period beginning at τ
  N_Δ = delta matrix containing changes in positions of items from $\mathcal{T}$ in lists associated with items from $\mathcal{A}$
  foreach *target item* t *in* N_Δ **do**
    $scores_t = \text{ScoreFunction}(N_\Delta[t])$
    if $scores_t \geq threshold$ *and* $t \notin \mathcal{A}$ **then**
      $inferredItems = inferredItems \cup \{t\}$
return $inferredItems$

Using the Auxiliary information to reconstruct the sensitive items of the user.

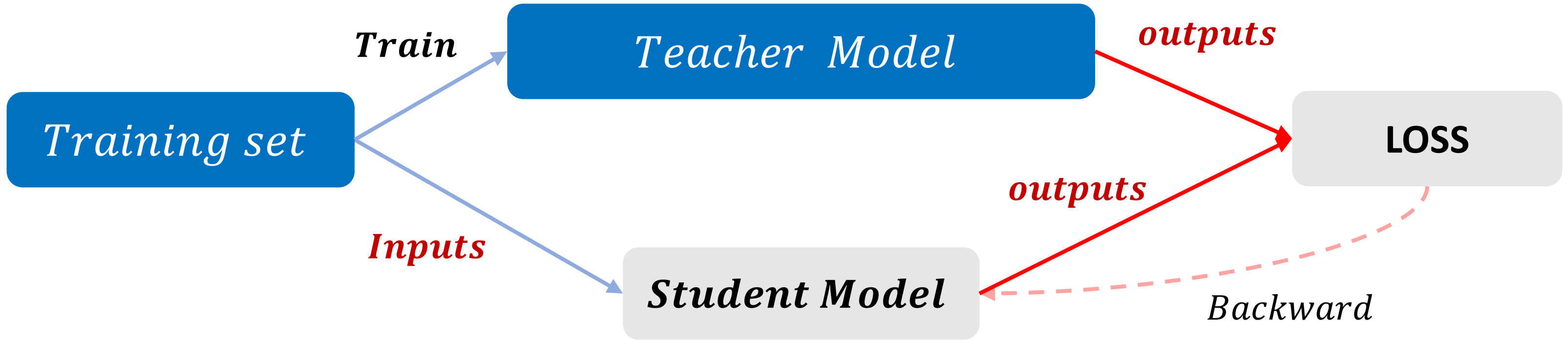
Auxiliary information:

- Users publicly rate or comment on items
- Users revealing partial information about themselves via third-party sites.
- Data from other sites which are not directly tied to the user's transactions on the target site but leak partial information about them.

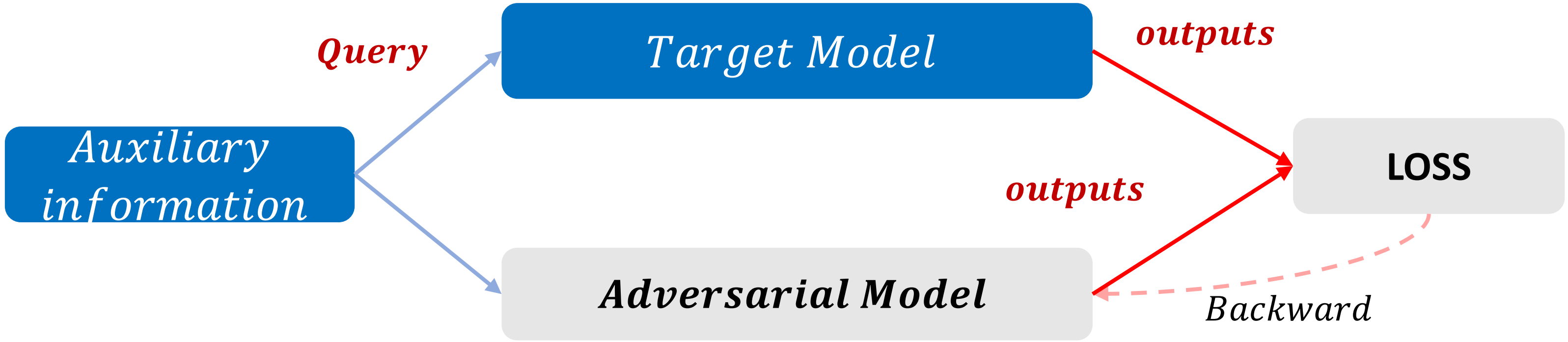
J. A. Calandrino, et al, "You Might Also Like:" Privacy Risks of Collaborative Filtering," 2011 IEEE SP.

Model Extraction Attacks

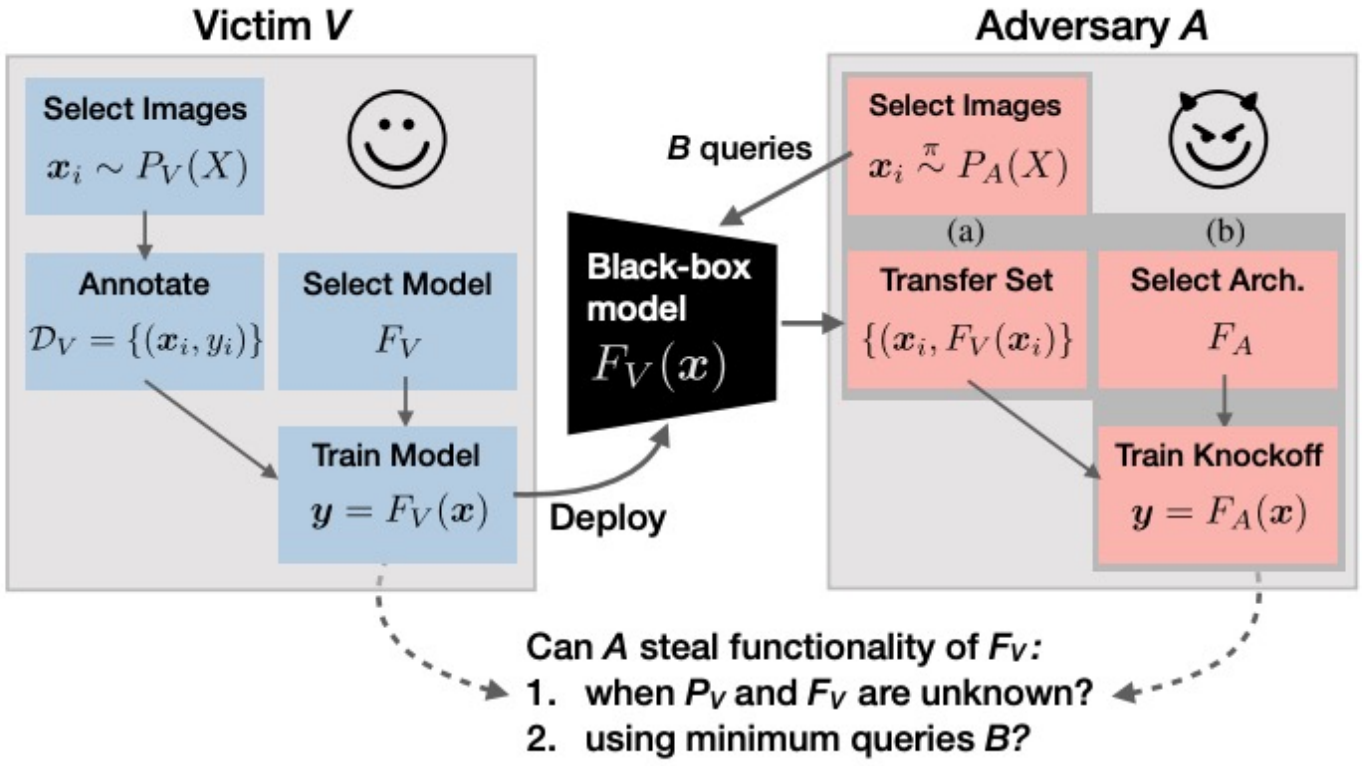
- Knowledge Distillation



- Model Extraction Attacks



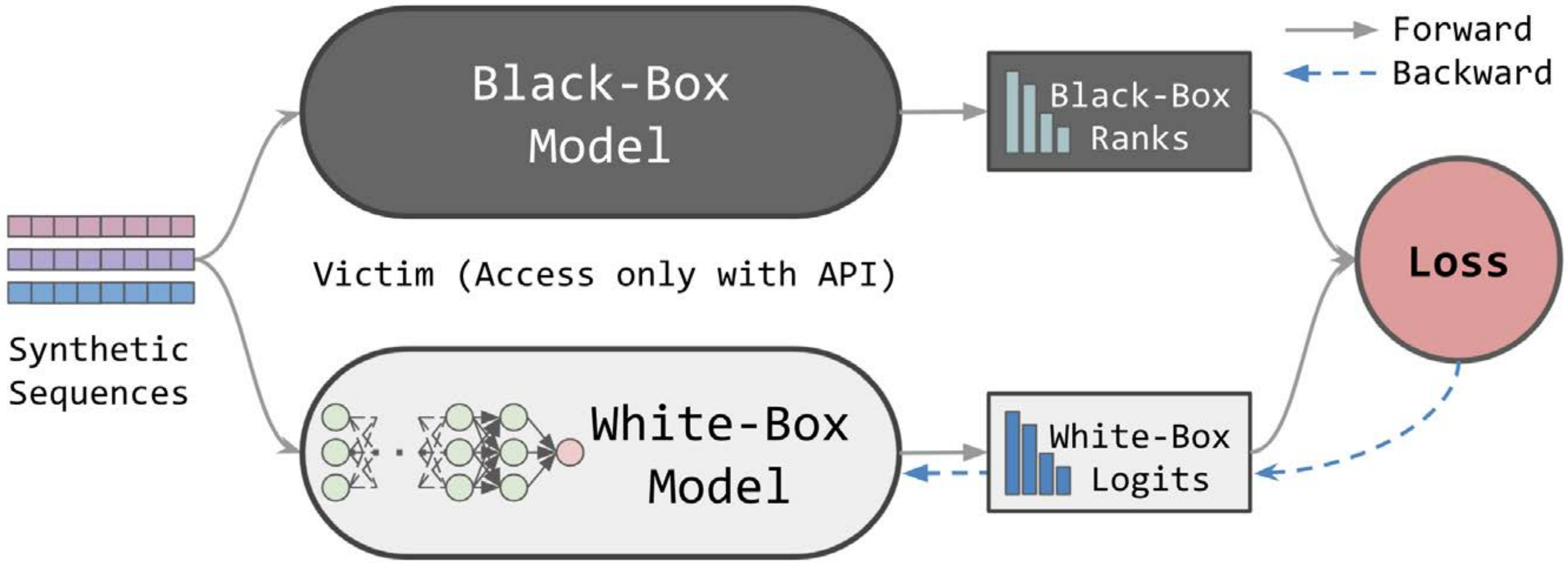
Model Extraction Attacks



The **Adversary A** steal the knowledge of the black-box model by B queries

Orekondy T, Schiele B, Fritz M. Knockoff nets: Stealing functionality of black-box models. CVPR, 2019.

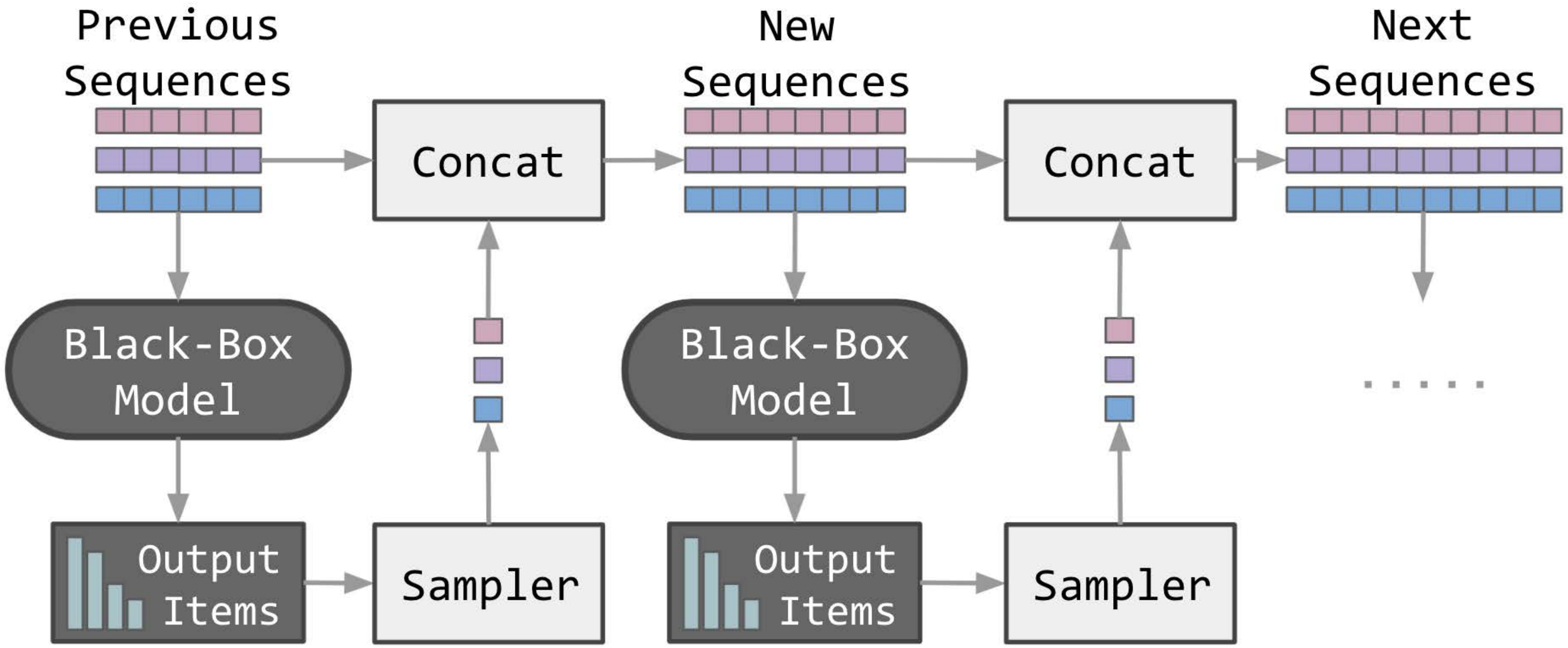
Model Extraction Attacks



Workflow of Model Extraction Attack

Yue Z, et al. Black-box attacks on sequential recommenders via data-free model extraction[C] RecSys, 2021.

Model Extraction Attacks



Synthetic Sequences Generation

Yue Z, et al. Black-box attacks on sequential recommenders via data-free model extraction[C] RecSys, 2021.

Summary of Attacks

- **Membership Inference Attacks** (MIA) aim to identity whether **the target user is used to train** the target recommender system.

- **Property Inference Attacks** (PIA) aim at **stealing global properties** of the training data in the target recommender system.

- **Reconstruction Attacks** (RA), aim to **infer private information** or labels on training data.

- **Model Extraction Attacks** (MEA), aims to **steal the parameters and structure** of a target model and create a new replacement model that behaves similarly to the target model.

Privacy

- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods
- Applications
- Survey and Tools
- Future Directions

Privacy-preserving Methods

	Taxonomy	Representative Methods
Privacy-preserving Methods	Differential Privacy	[45, 46, 395, 429, 432, 459]
	Federated Learning	[111, 138, 160, 218, 284, 376, 378]
	Adversarial Learning	[22, 208, 229, 295, 352]
	Anonymization & Encryption	[53, 163, 281, 302, 360, 402, 413, 430]

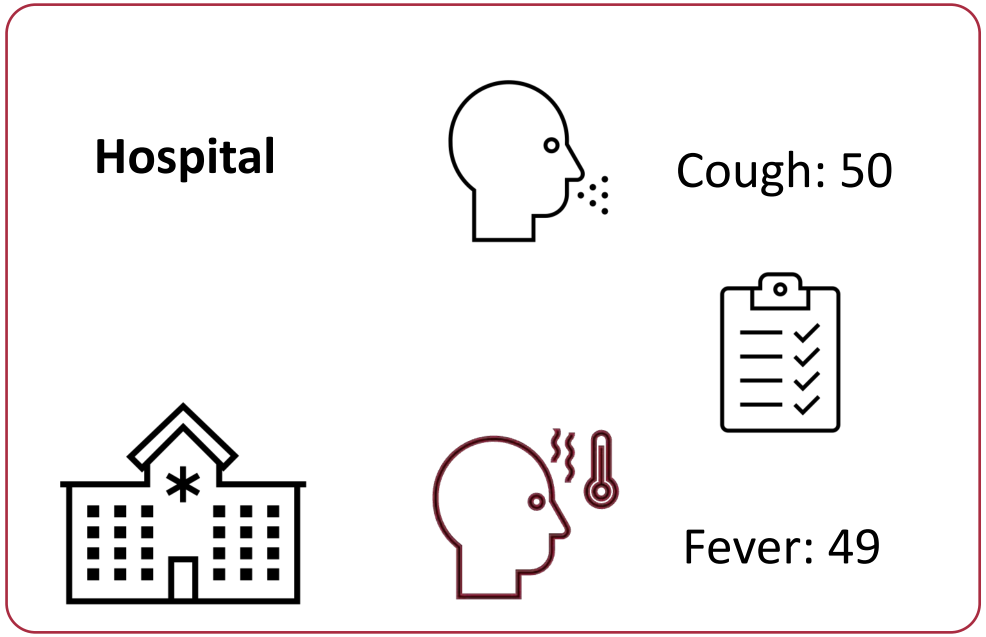
Differential Privacy

Given $\epsilon > 0$ and $\delta \geq 0$, a randomized mechanism $\mathcal{M}$ satisfies (ϵ, δ)-differential privacy, if for any adjacent datasets $\boldsymbol{D}$ and $\boldsymbol{D'} \in \mathbf{R}$ and for any subsets of outputs $\mathcal{S}$, the following equation is met:

$$P(\mathcal{M}(D) \in \mathcal{S}) \leq e^{\epsilon} P(\mathcal{M}(D') \in \mathcal{S}) + \delta$$

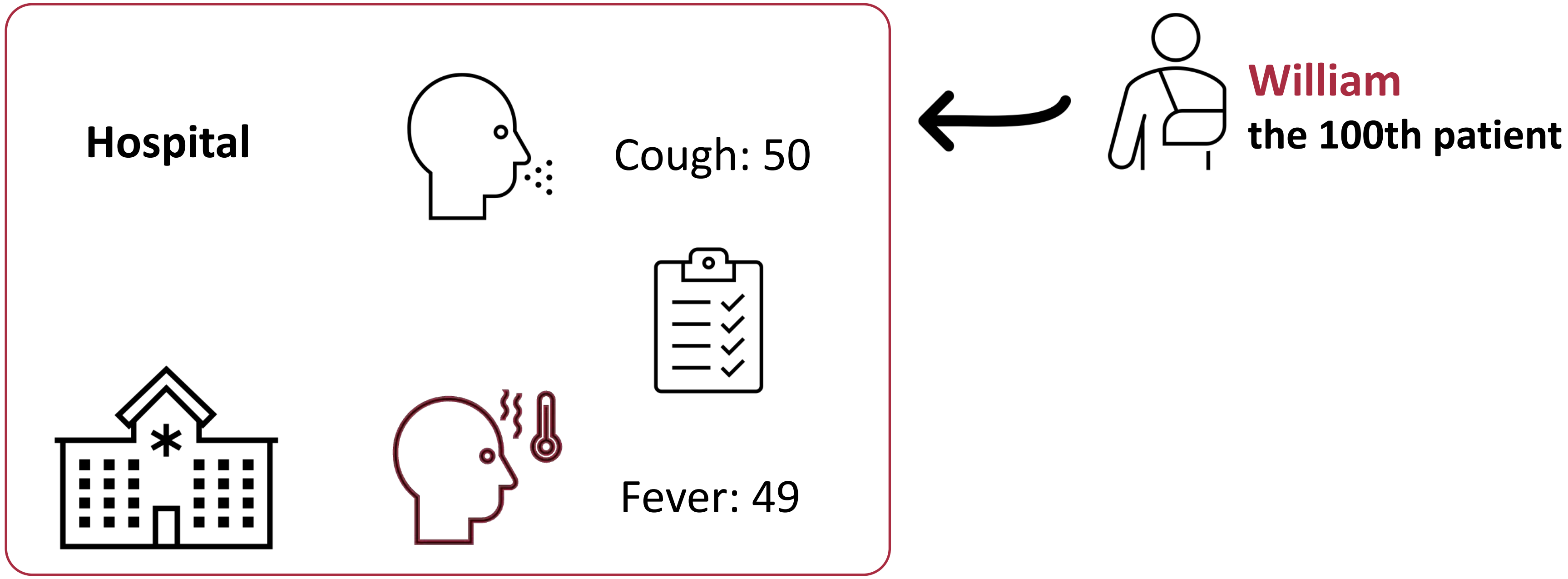
ϵ is the **privacy budget,** the smaller ϵ is, the better the privacy protection is, but more noise is added, and the data utility decreases.

Differential Privacy



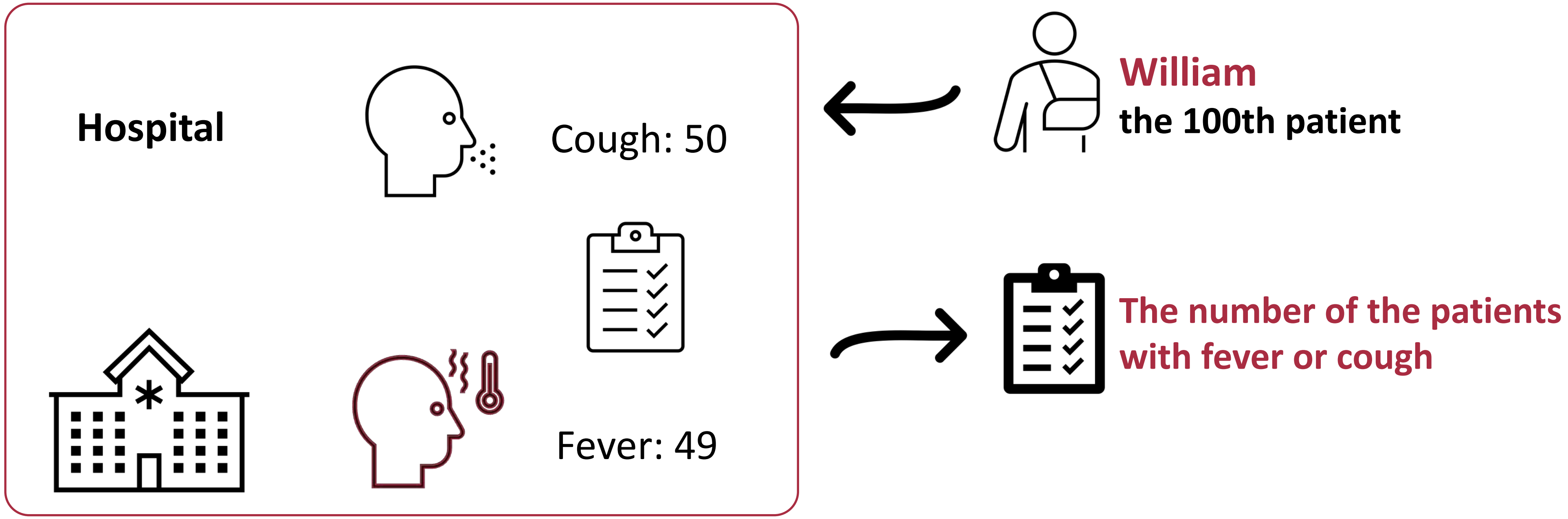
J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy



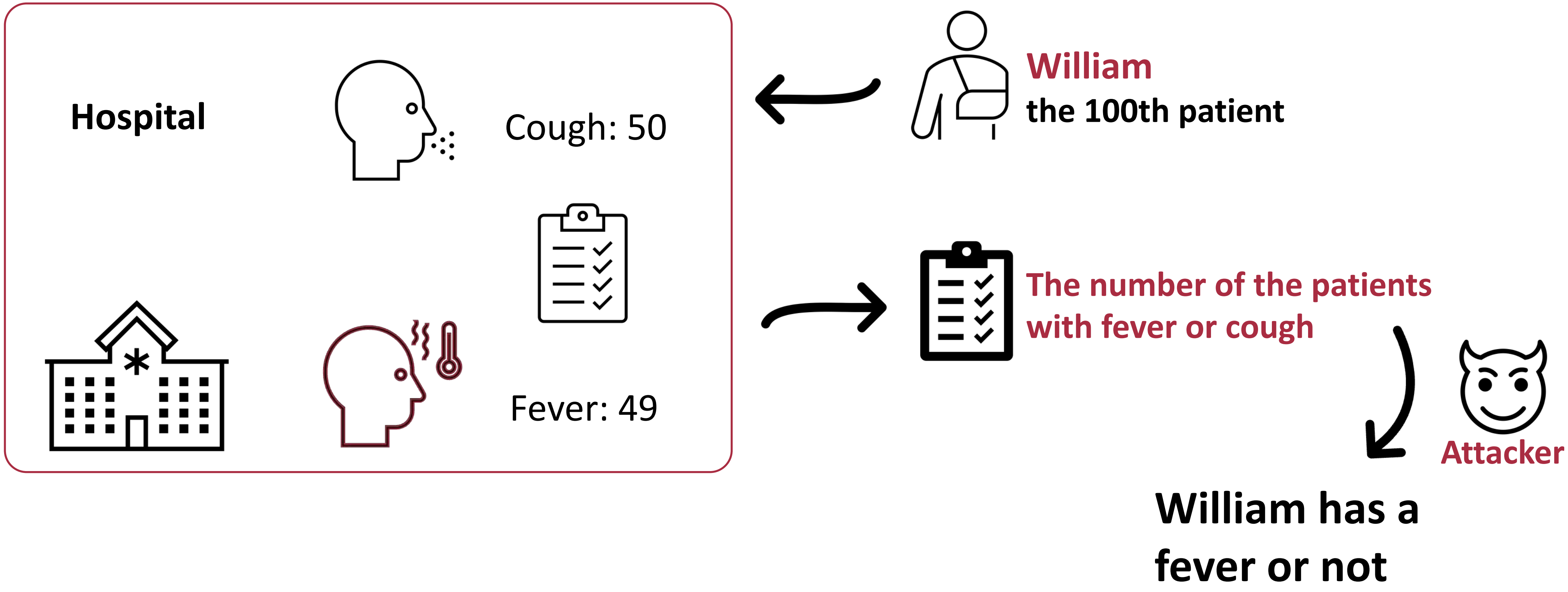
J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy



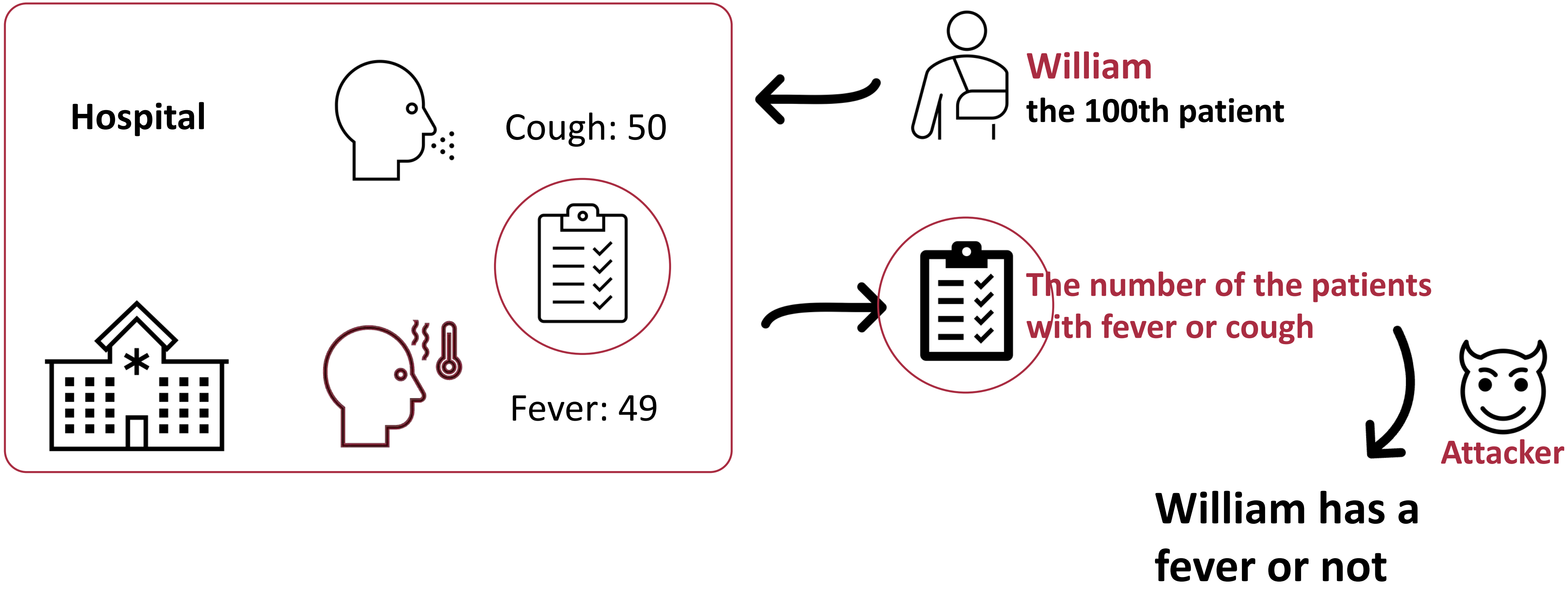
J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy



J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy



J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy

Before

After

Differential Privacy makes them **similar enough** so that the attack can not infer which illness William has.

J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy

Transform the rating matrix to the cross domain, which could meet the Differential Privacy requirements.

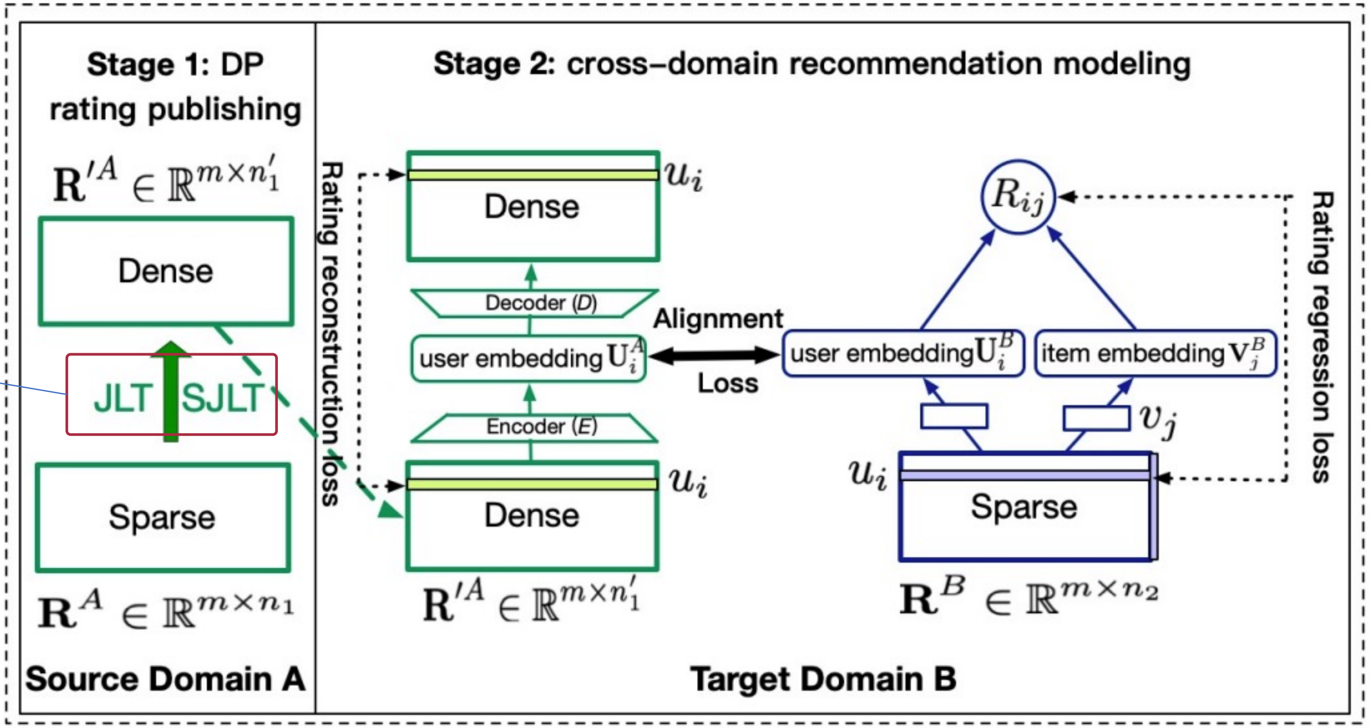


Figure 1: Framework of PriCDR.

Chen C, et al. Differential Private Knowledge Transfer for Privacy-Preserving Cross-Domain Recommendation. WWW 2022.

Federated Learning

Devices with local recommender systems and users’ data

Q. Yang, et al. Federated machine learning: Concept and applications. TIST, 2019.

Federated Learning

Global server with global recommendation model

Devices with local recommender systems and users' data

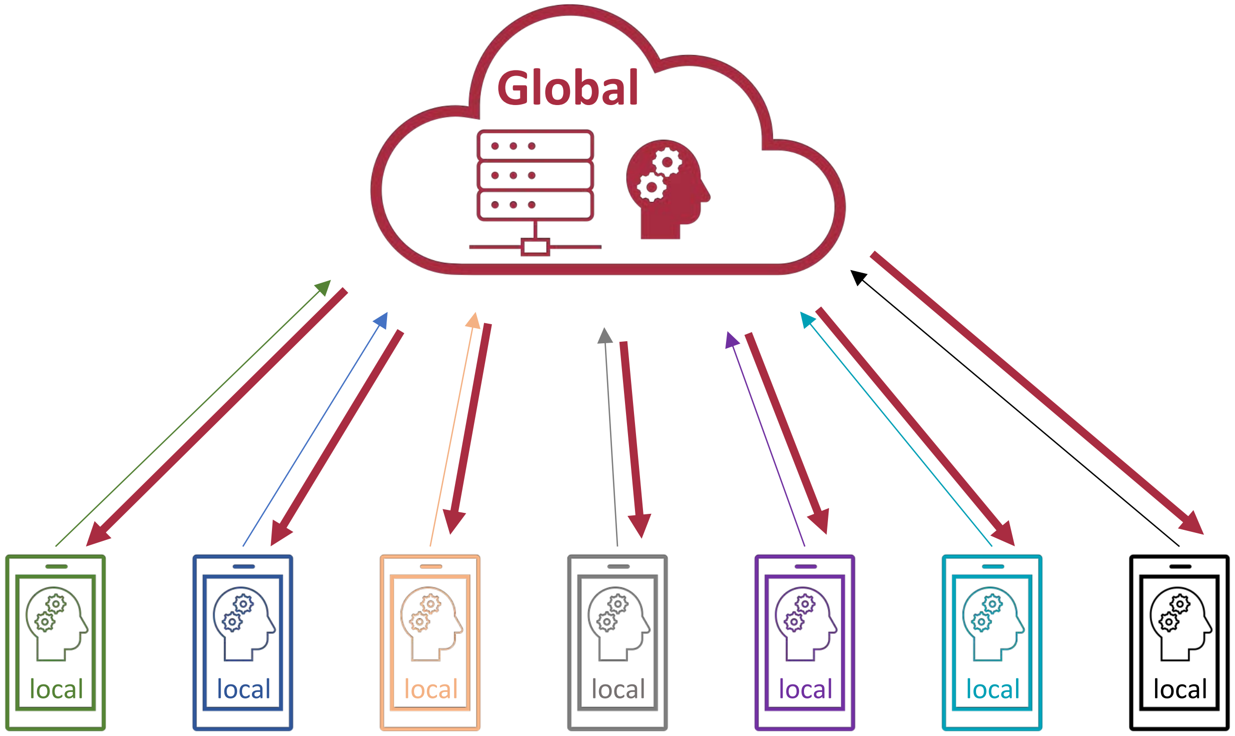
Q. Yang, et al. Federated machine learning: Concept and applications. TIST, 2019.

Federated Learning

Global server with global recommendation model

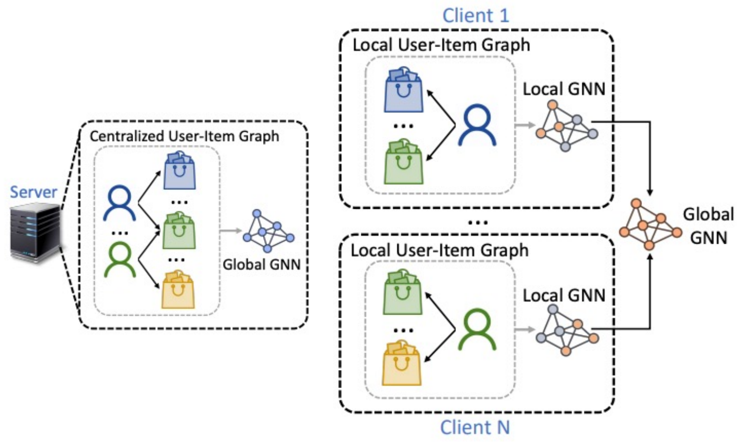
Gradients

Devices with local recommender systems and users' data



Q. Yang, et al. Federated machine learning: Concept and applications. TIST, 2019.

Federated Learning



(a) Centralized learning. (b) Decentralized learning.

Figure 1: Comparisons between centralized and decentralized training of GNN based recommendation models.

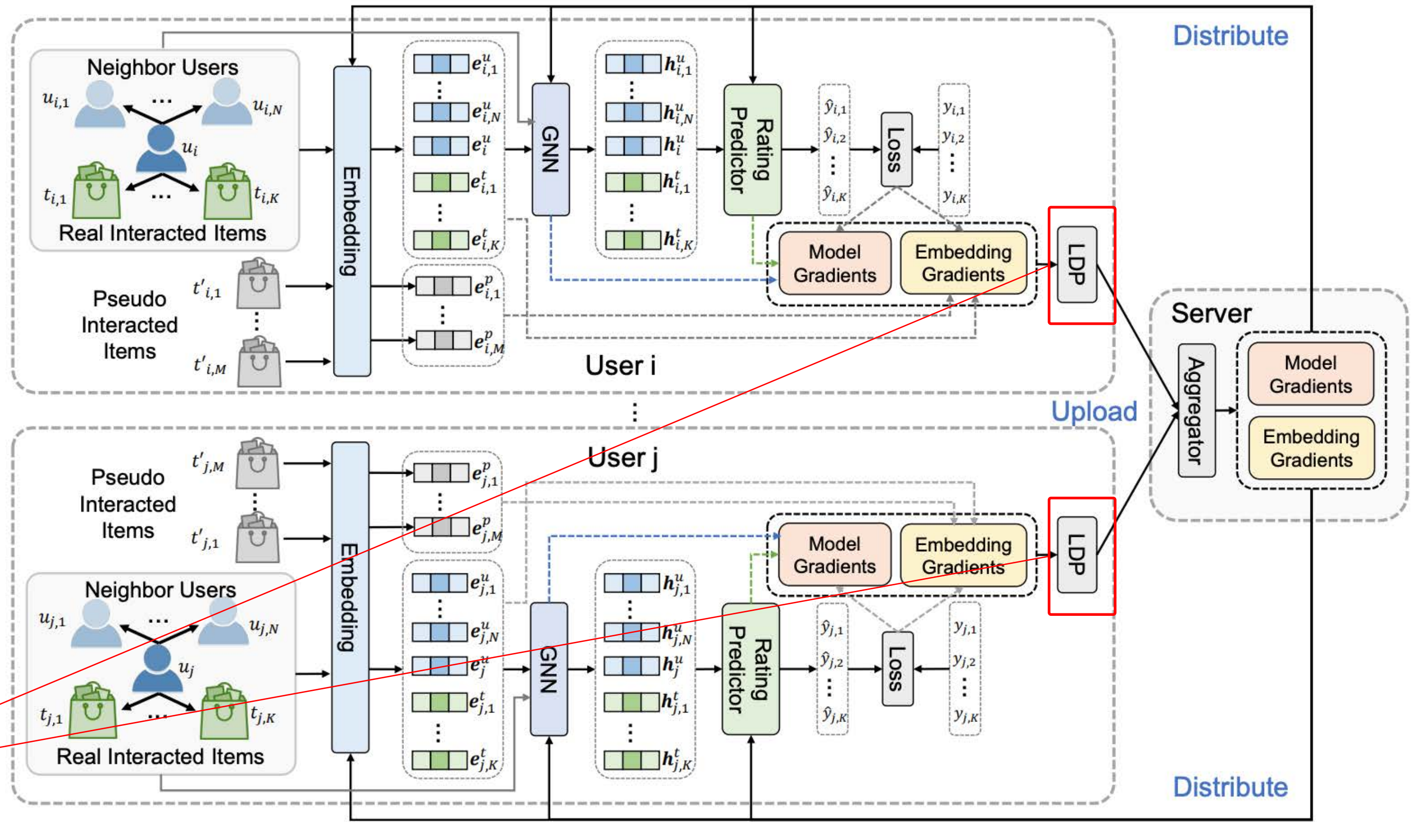
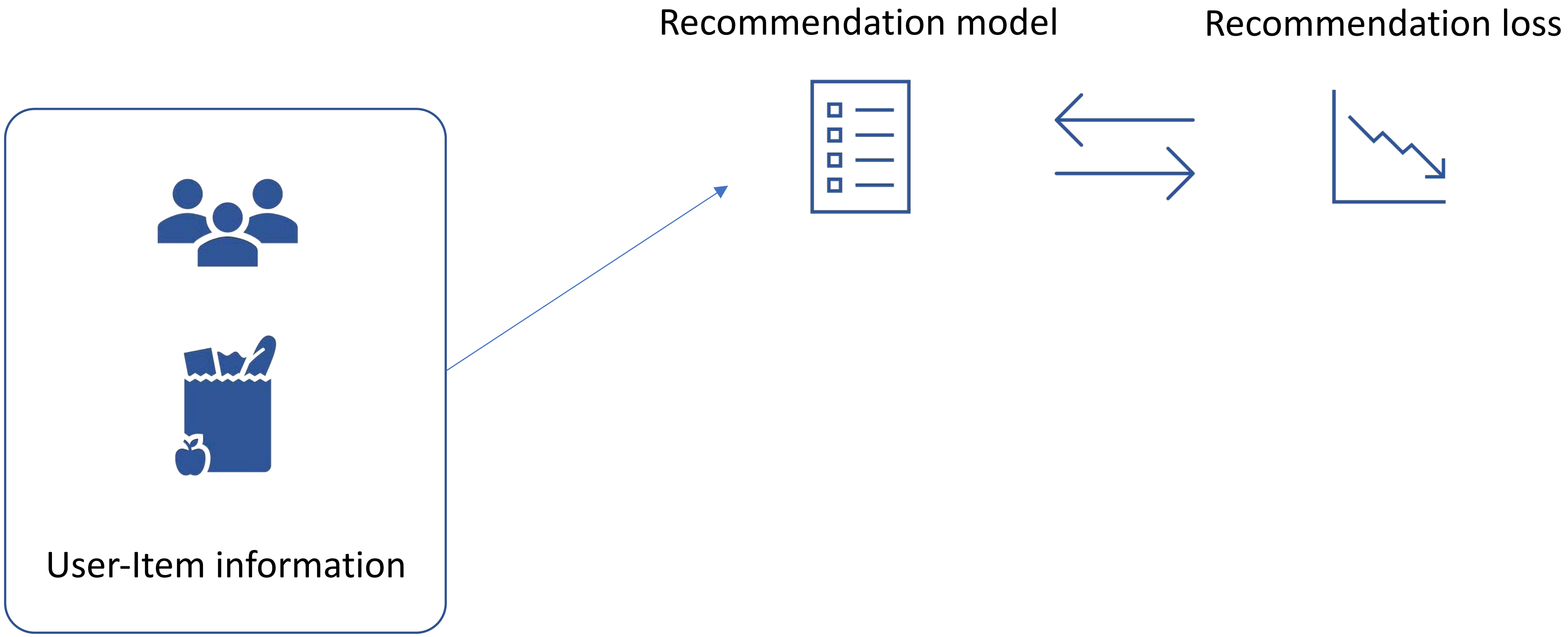


Figure 2: The framework of our *FedGNN* approach.

Before uploading, the gradients are privacy processed by Differential Privacy.

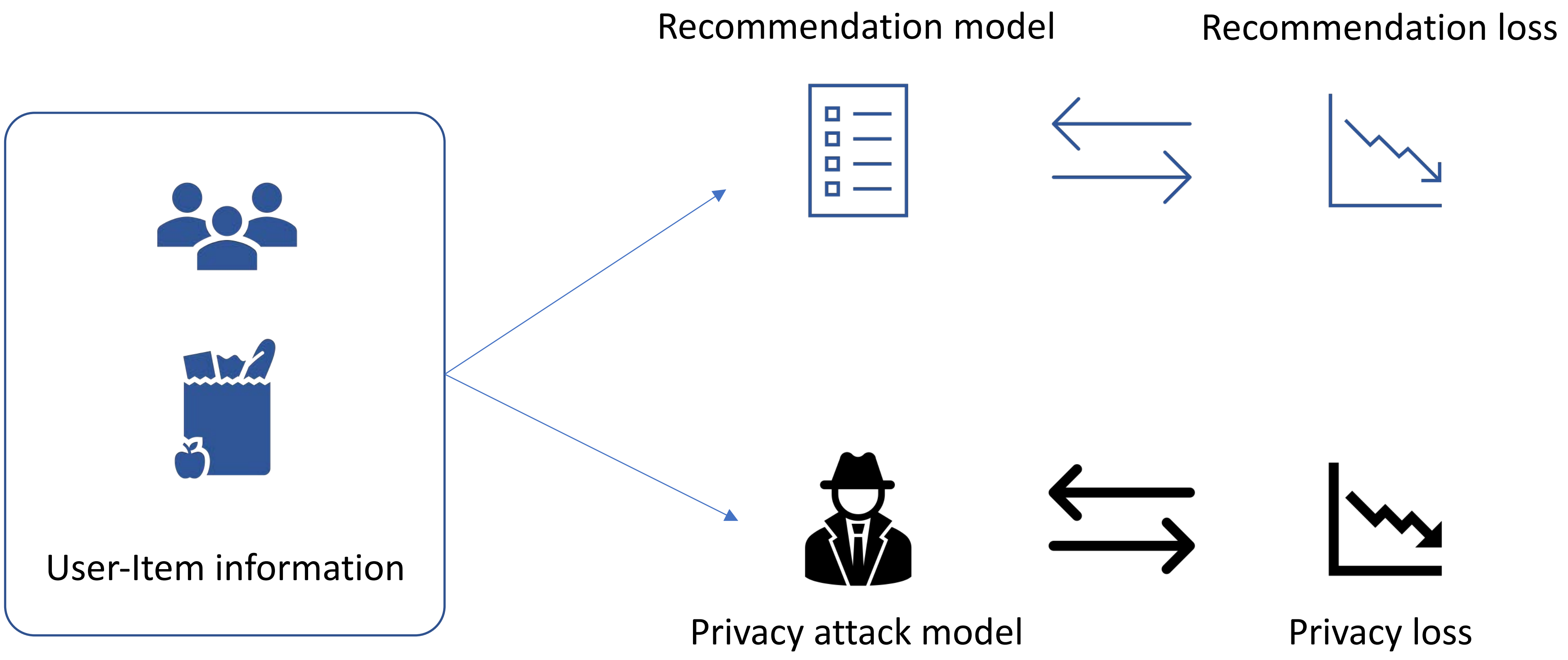
Wu C, et al. Fedgnn: Federated graph neural network for privacy-preserving recommendation. arXiv, 2021.

Adversarial Learning



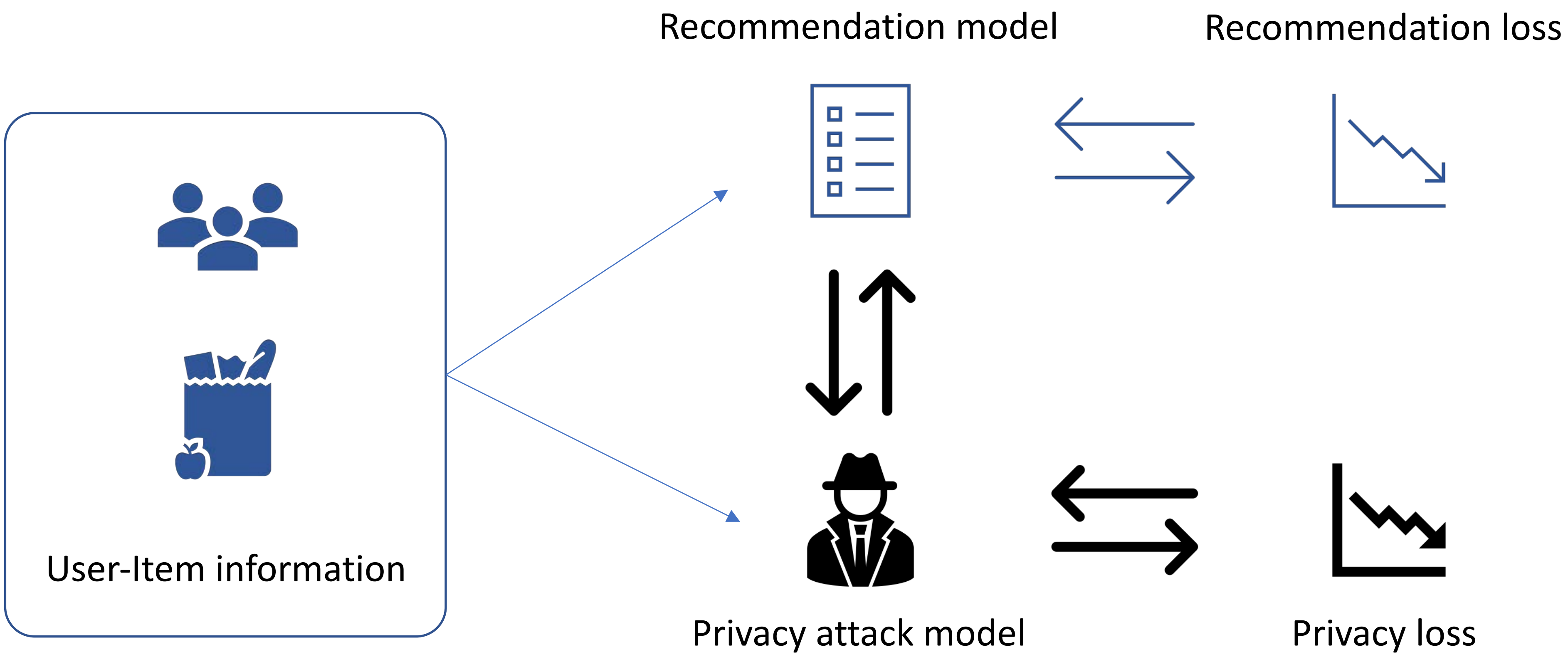
L. Huang, et al. Adversarial machine learning. the 4th ACM workshop on Security and artificial intelligence, 2011.

Adversarial Learning



L. Huang, et al. Adversarial machine learning. the 4th ACM workshop on Security and artificial intelligence, 2011.

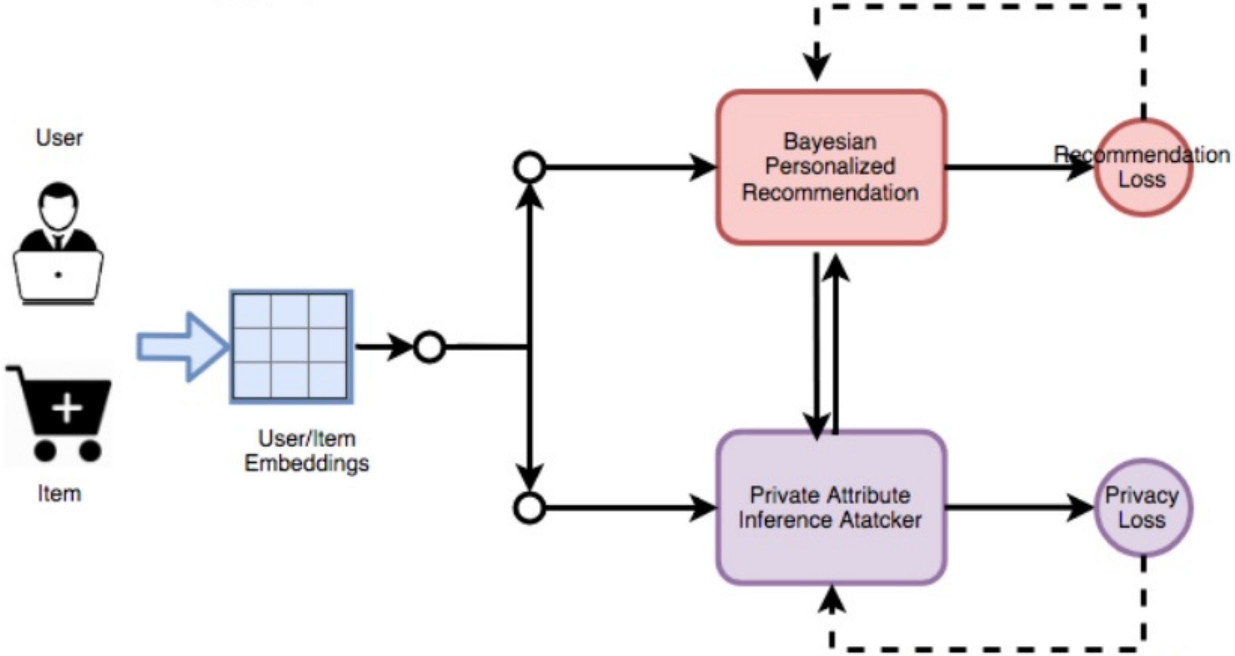
Adversarial Learning



L. Huang, et al. Adversarial machine learning. the 4th ACM workshop on Security and artificial intelligence, 2011.

Adversarial Learning

$$\min_{\theta_R} \frac{1}{N}\sum_{h=1}^{N}\left[\sum_{(h,j,k)\in\mathcal{D}_h} -\ln\sigma\big((\hat{y}_{hj}(\theta_R)-\hat{y}_{hk}(\theta_R)).\,g(h,j,k)\big) - \alpha\left[\frac{1}{T}\sum_{t=1}^{T}\mathcal{L}_{D_P^t}(\hat{p}_{h,t},p_{h,t})\right]\right] + \lambda\Omega(\theta)$$



$$\min_{\{\theta_P^t\}^T\_\{t=1\}} \frac{1}{N}\sum_{h=1}^{N}\left[\frac{1}{T}\sum_{t=1}^{T}\mathcal{L}_{D_P^t}(\hat{p}_{h,t},p_{h,t})\right]$$

Model structure

$$\underbrace{\min_{\theta_R}\left(\mathcal{L}_{D_R}\ \overbrace{-\alpha\max_{\{\theta_P^t\}_{t=1}^{T}}\mathcal{L}_{D_P}}^{\text{private-attribute attacker}}\right)}_{\text{privacy-aware recommendation system}}$$

Objective Function

Beigi G, et al. Privacy-aware recommendation with private-attribute protection using adversarial learning. 2020.

Anonymization

Anonymization aim to prevent the public data from being linked to individual identities of people.

Zip	Age	Disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Viral infection
130▪	3▪	Cancer
130▪	3▪	Cancer

▪ denotes a suppressed value.

Quasi-identifiers Sensitive attributes

https://media13.connectedsocialmedia.com/intel/01/9768/Using_Data_Anonymization_Enhance_Cloud_Security.pdf

Anonymization

Anonymization aim to prevent the public data from being linked to individual identities of people.

Zip	Age	Disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Viral infection
130▪	3▪	Cancer
130▪	3▪	Cancer

▪ denotes a suppressed value.

Quasi-identifiers

k-Anonymity (k=2)

https://media13.connectedsocialmedia.com/intel/01/9768/Using_Data_Anonymization_Enhance_Cloud_Security.pdf

Anonymization

Anonymization aim to prevent the public data from being linked to individual identities of people.

Zip	Age	Disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Viral infection
130▪	3▪	Cancer
130▪	3▪	Cancer

▪ denotes a suppressed value.

Quasi-identifiers

k-Anonymity (k=2)

Zip	Age	Disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Heart disease
130▪	2▪	Cancer
130▪	2▪	Cancer
130▪	2▪	Viral infection
130▪	2▪	Viral infection
130▪	3▪	Viral infection
130▪	3▪	Viral infection
130▪	3▪	Cancer
130▪	3▪	Cancer

▪ denotes a suppressed value.

Sensitive attributes

l-Diversity (l=2)

https://media13.connectedsocialmedia.com/intel/01/9768/Using_Data_Anonymization_Enhance_Cloud_Security.pdf

Encryption

Encryption techniques make data unreadable to those who do not have the key to decrypt it.

Encryption

Using the noise to encrypt sensitive data.

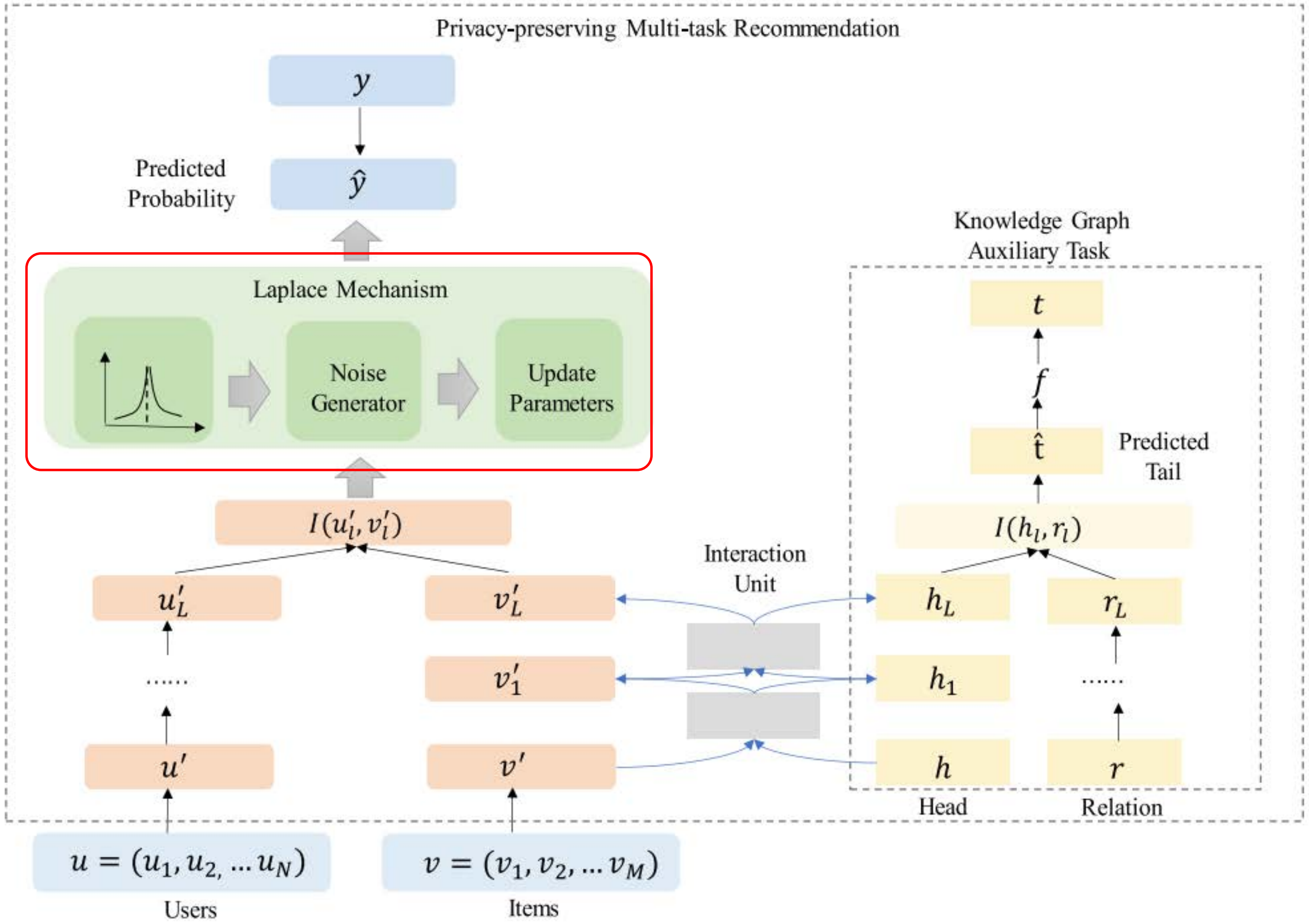


FIGURE 1. **A privacy-preserving multi-task framework for knowledge graph enhanced recommendation.**

Yu B, et al. A privacy-preserving multi-task framework for knowledge graph enhanced recommendation. IEEE Access, 2020.

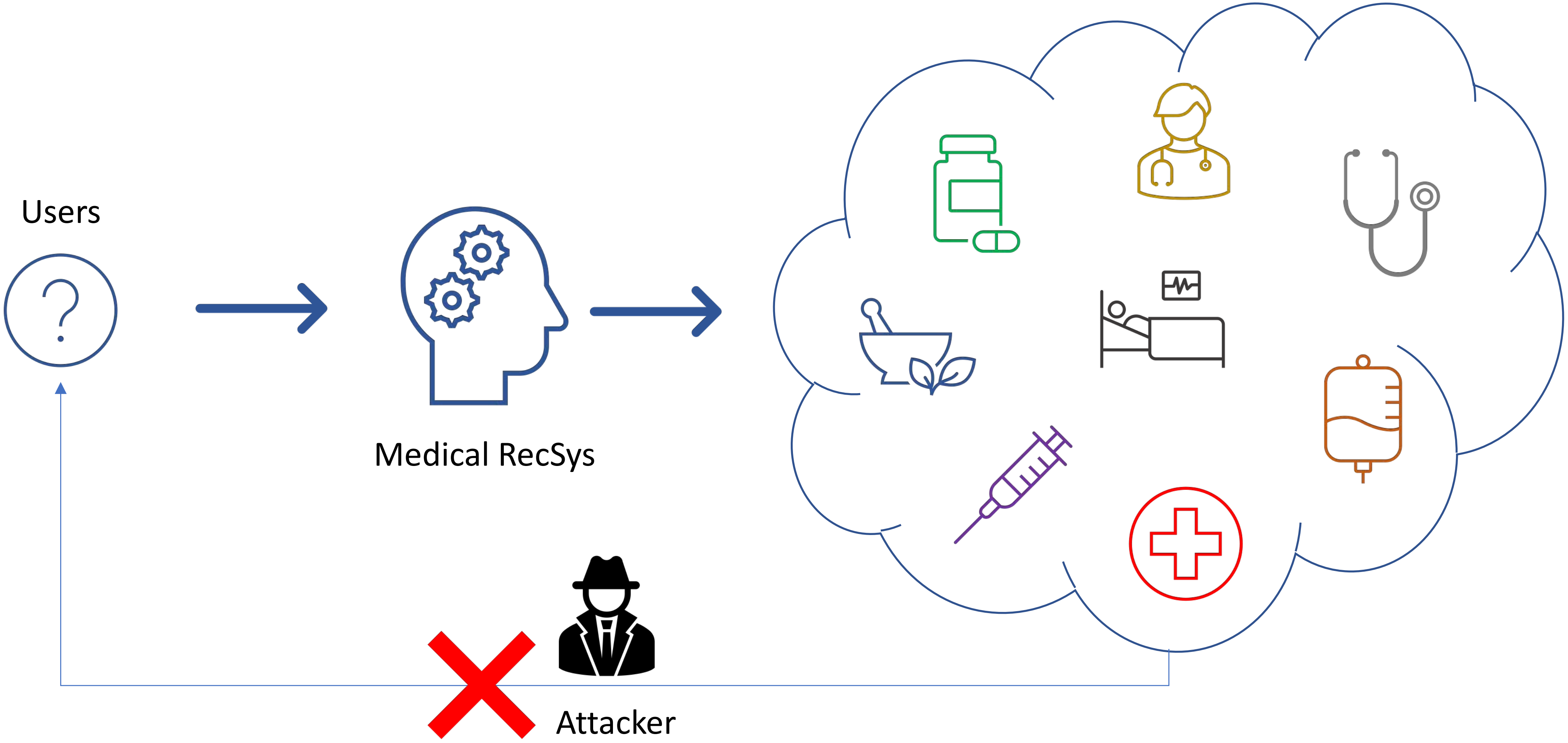
Summary of Privacy Preserving

- **Differential Privacy (DP)** is a common way to **preserve membership inference attacks**, which can provide strict statistical guarantees for data privacy.
- **Federated Learning (FL)** isolates users' data and the cloud server by **only transferring the gradients** between them.
- **Adversarial Learning (AL)** can be formulated as the **minimax simultaneous optimization** of recommendation and privacy attacker models.
- **Anonymization** makes the privacy **attributes of users impossible to be correlated** with individual identities of people.
- **Encryption** techniques **prevent people who do not have the authorization** from any useful information.

Privacy

- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods
- Applications
- Survey and Tools
- Future Directions

Private medical RecSys



Private medical RecSys

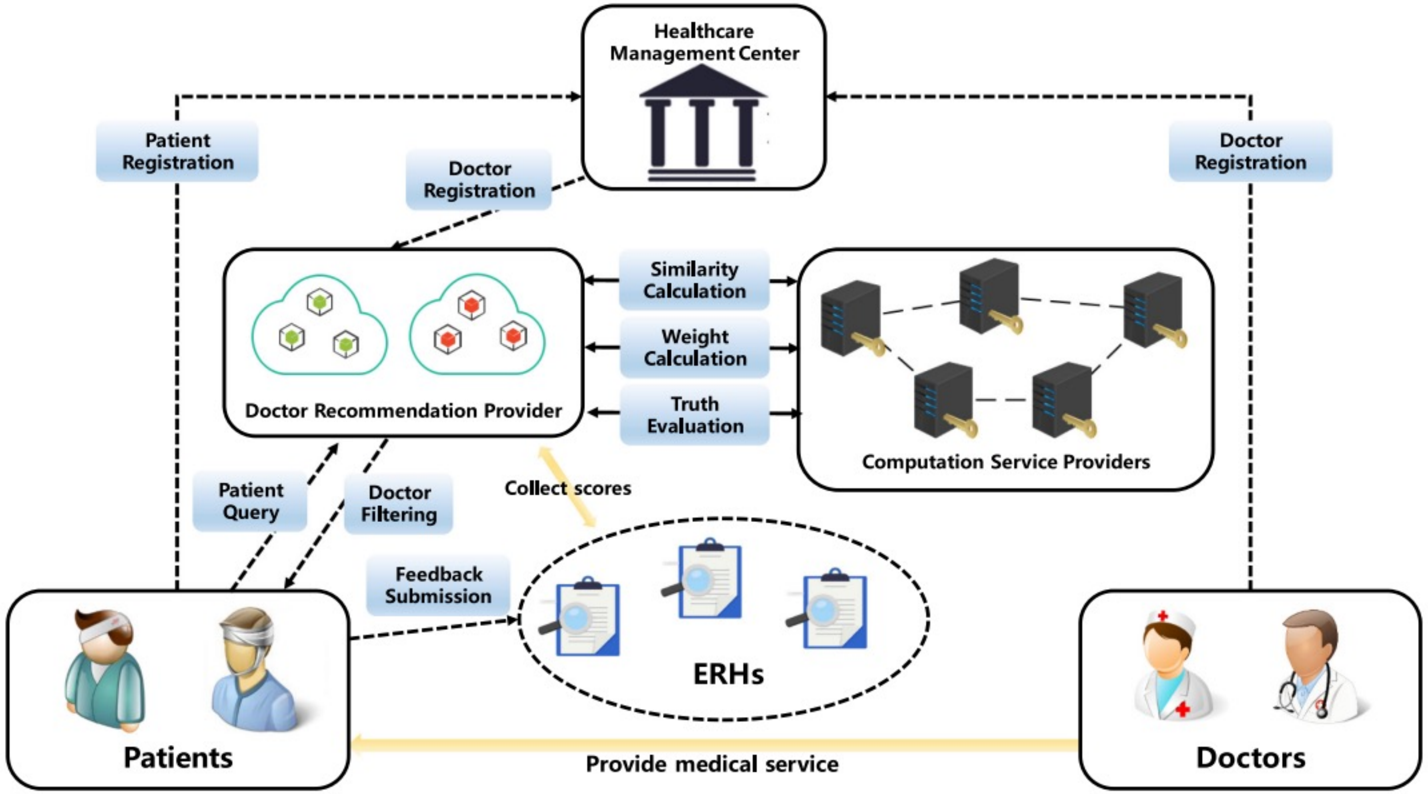
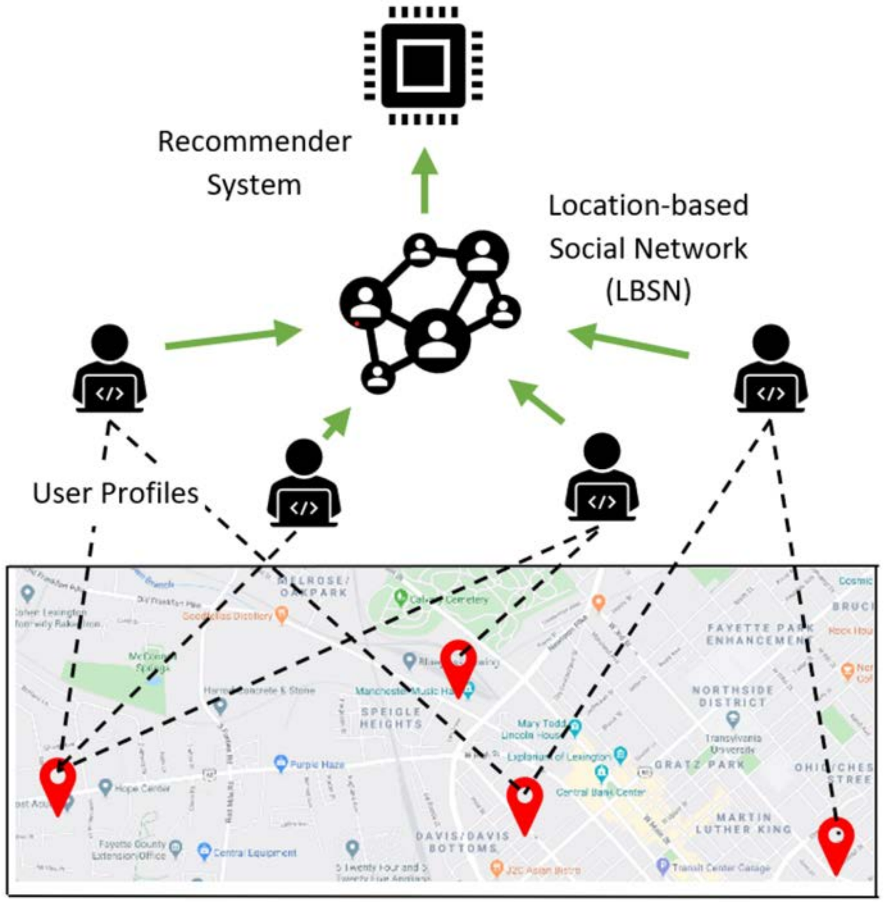


Fig. 1. System model.

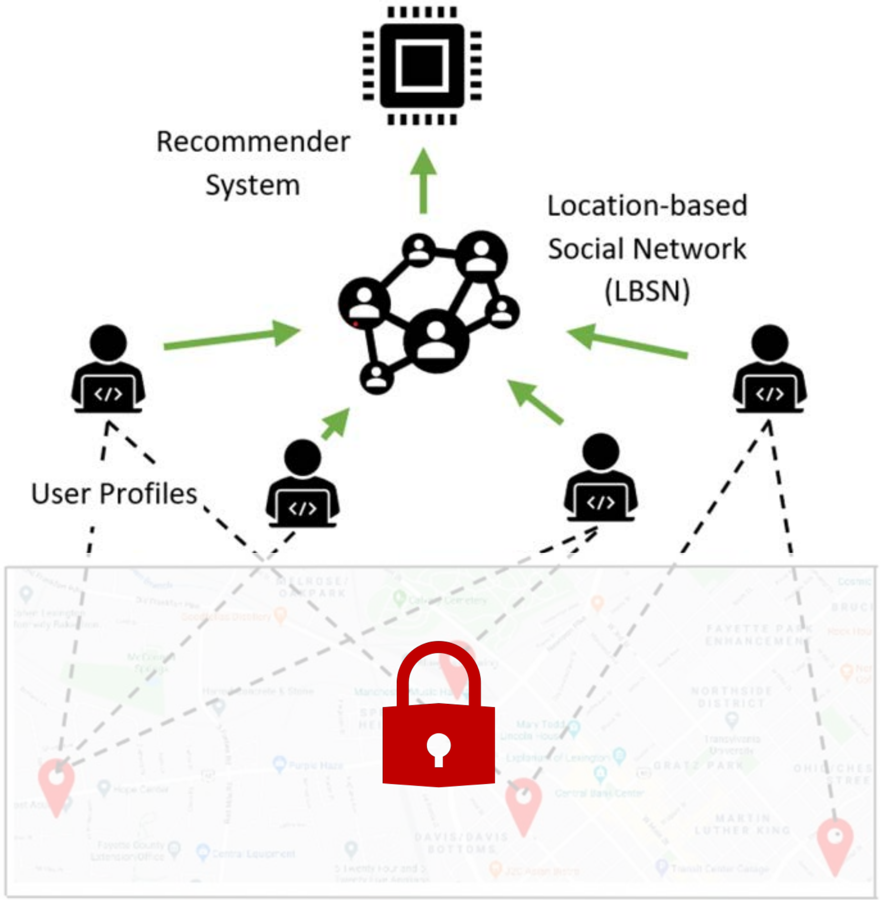
Cong Peng, et al. 2021. EPRT: An Efficient Privacy-Preserving Medical Service Recommendation and Trust Discovery Scheme for eHealth System. ACM Trans. Internet Technol. 2021.

Location-private RecSys



Cui L, Wang X. A Cascade Framework for Privacy-Preserving Point-of-Interest Recommender System[J]. 2022.

Location-private RecSys



Cui L, Wang X. A Cascade Framework for Privacy-Preserving Point-of-Interest Recommender System[J]. 2022.

Privacy

- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods
- Applications
- Survey and Tools
- Future Directions

Surveys

Privacy in recommender systems

- Erfan Aghasian, Saurabh Garg, and James Montgomery. 2018. User's Privacy in Recommendation Systems Applying Online Social Network Data, A Survey and Taxonomy. arXiv preprint arXiv:1806.07629 (2018).

- Weiming Huang, Baisong Liu, and Hao Tang. 2019. Privacy protection for recommendation system: a survey. In Journal of Physics: Conference Series.

Privacy in machine learning

- Fatemehsadat Mireshghallah, Mohammadkazem Taram, Praneeth Vepakomma, Abhishek Singh, Ramesh Raskar, and Hadi Esmaeilzadeh. 2020. Privacy in deep learning: A survey. arXiv preprint arXiv:2004.12254 (2020).

- Maria Rigaki and Sebastian Garcia. 2020. A survey of privacy attacks in machine learning. arXiv preprint arXiv:2007.07646 (2020).

Tools

Differential privacy

- Facebook Opacus
- TensorFlow-Privacy
- OpenDP
- Diffpriv
- Diffprivlib

Homomorphic Encryption

- Awesome HE
- TF Encrypted

Federated learning

- TFF
- FATE
- FedML
- LEAF

Privacy

- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods
- Applications
- Survey and Tools
- Future Directions

Future Directions

- **Privacy and performance trade-off**

Depending on different task requirements, how to protect privacy with minimal performance cost may be a continuous research direction.

- **Comprehensive privacy protection**

It is still challenging to combine different privacy protection approaches without degrading the recommendation performance.

- **Defence against shadow training**

The training method provides vital support to the privacy attacks but is indeed trained under reasonable assumptions.

Summary

- **Privacy Attacks**
  - Membership Inference Attacks (MIA)
  - Property Inference Attacks (PIA)
  - Reconstruction Attacks (RA)
  - Model Extraction Attacks (MEA)
- **Privacy Preserving**
  - Differential Privacy (DP)
  - Federated Learning (FL)
  - Adversarial Learning (AL)
  - Anonymization
  - Encryption

For more information, please refer to our survey:

A Comprehensive Survey on Trustworthy Recommender Systems

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

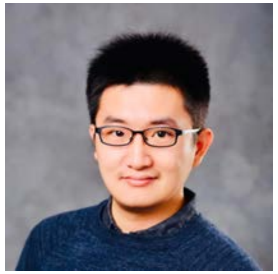
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Trustworthy Recommender Systems

Introduction

Wenqi Fan

→

Non-discrimination & Fairness

→

Xiao Chen

→

Safety & Robustness

Shijie Wang

→

Explainability

Jingtong Gao

→

Privacy

Lin Wang

→

Environmental Well-being

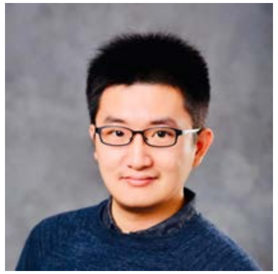
Accountability & Auditability

Qidong Liu

→

Dimension Interactions

Future Directions



Xiangyu Zhao

Background

- Environmental Well-being
  - Advanced RS models benefit many aspects of society.
  - Advanced RS models cost much resources.

- Relation with Trustworthy
  - Environmental-friendly RS can be widely adopted.

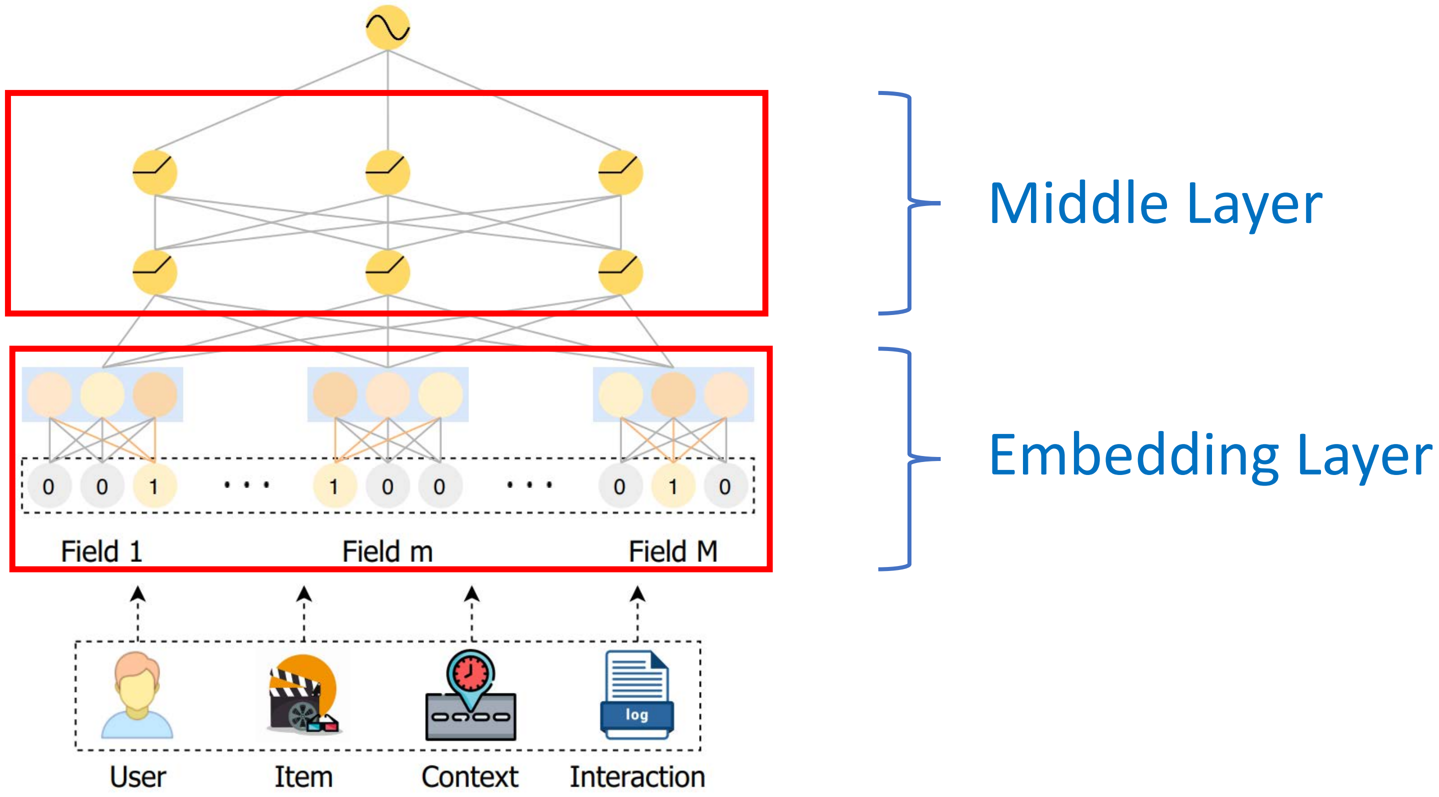
Model Compression

Acceleration Techniques

Model Compression

- Concepts:
  - **Model Compression**

  ➡ Save Storage Resources
  - Acceleration Technique
- Taxonomy
  - Embedding Layer
  - Middle Layer



Model Compression

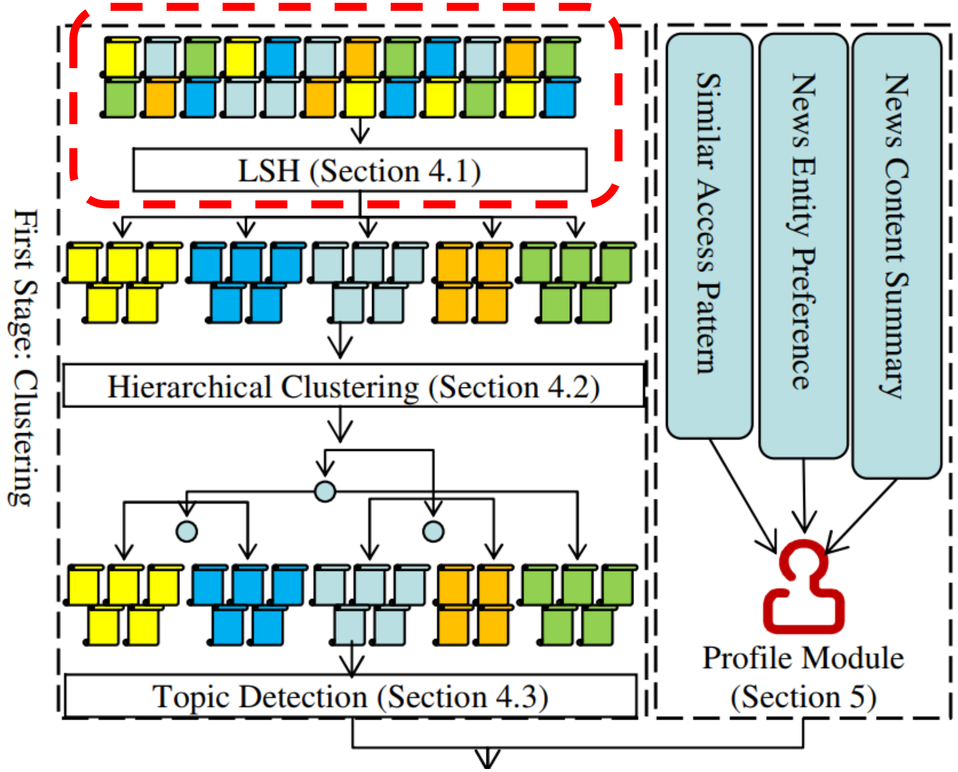
- Model Compression
  - Hash
    - Data-independent Methods
    - Data-dependent Methods
  - Quantization
  - Knowledge Distillation
  - Neural Architecture Search
  - Others

$$x \in \{0,1\}^n \xrightarrow{h(\cdot)} y \in \{0,1\}^m$$

The hash function $h(\cdot)$ shrink the vocabulary size from n to m, where $n \gg m$. Thus, the embedding table is compressed.

Hash

- **Data-independent Method**
  - The hash function $h(\cdot)$ is pre-defined without considering the dataset.
  - ✓ Advantage: time-saving



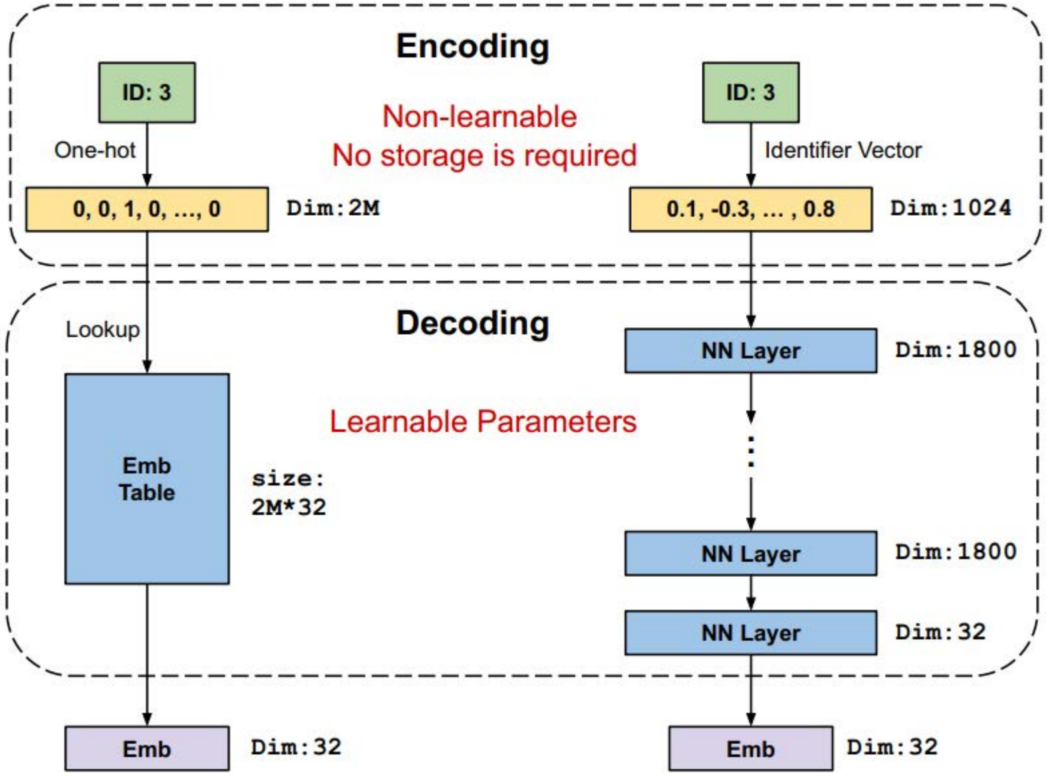
- **SCENE** – SIGIR'11
  - A two-stage news recommendation.
  - Make use of the **Locality Sensitivity Search (LSH)** to cluster similar news items, which can shrink the item embedding table.

SCENE : A Scalable Two-Stage Personalized News Recommendation System, SIGIR, 2011

Hash

- **Data-dependent Method**
  - The hash function $h(\cdot)$ is learned for the specific dataset.
  - ✓ Advantage: better performance

- **DHE** – KDD'21
  - Encode the feature value to a unique identifier with multiple hash functions.
  - Convert the unique identifier to an embedding with nn.
  - It substitutes embedding layer with hash functions and nn.



Learning to Embed Categorical Features without Embedding Tables for Recommendation, KDD, 2021

Model Compression

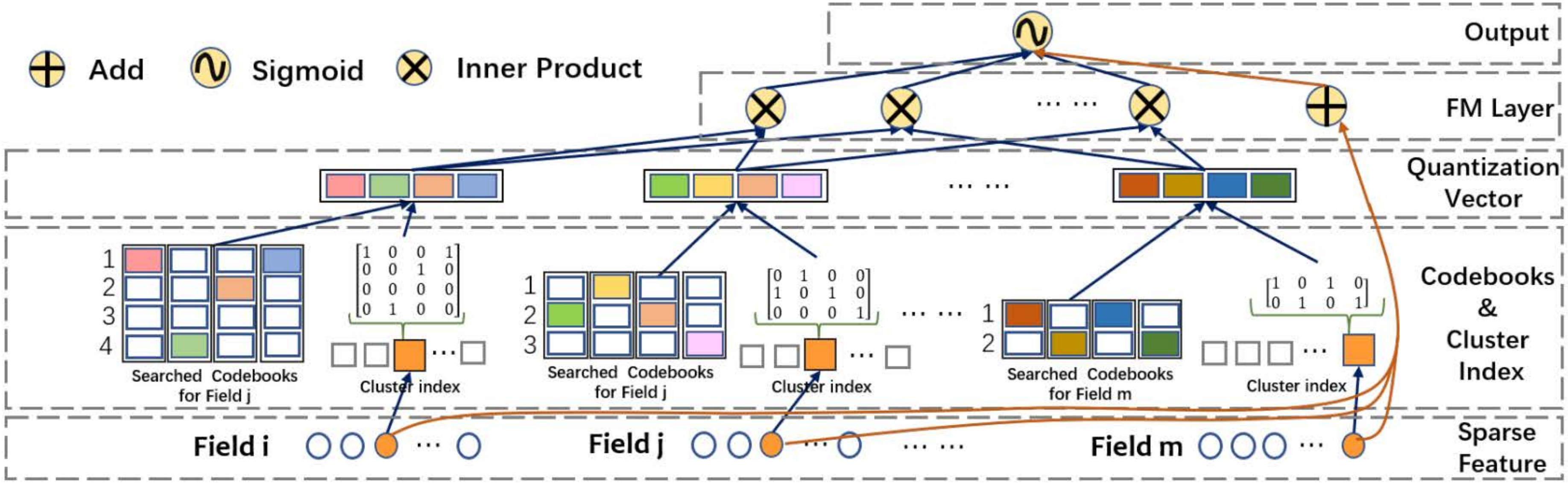
- Model Compression
  - Hash
  - Quantization
    - Product Quantization
    - Additive Quantization
    - Compositional Quantization
  - Knowledge Distillation
  - Neural Architecture Search
  - Others

$$\mathbf{q}_i = f(c^1_{w_i^1}, c^2_{w_i^2}, \ldots, c^B_{w_i^B})$$

The embedding of one feature value can be represented by its cluster center (Codeword w). To enhance the representation ability, an embedding is quantized to several sub-vectors (Codebook B). $f(\cdot)$ is the composing function.

Quantization

- **Product Quantization (PQ)**
  - PQ is a type of quantization method that composes quantized vectors by product.
- **xLightFM** – SIGIR’21
  - An end-to-end quantization-based factorization machine for the first time.
  - Search the quantized vectors in codebooks for each feature field.



xLightFM: Extremely Memory-Efficient Factorization Machine, SIGIR, 2021

Quantization

- **Additive Quantization (AQ)**
  - AQ is a type of quantization method that composes quantized vectors by add operation.
- **Anisotropic Additive Quantization** – AAAI'22
  - Design a new objective function for additive function by anisotropic loss function.
  - Achieve a lower approximation error than PQ.

Anisotropic Additive Quantization Problem:

$$\min_{C^{(1)},\ldots,C^{(M)}} \sum_{i=1}^{n} \min_{\tilde{\boldsymbol{x}}_i \in \sum_{m=1}^{M} C^{(m)}_{i_m(x_i)}} \underbrace{h_{i,\parallel} \left\| \boldsymbol{r}_\parallel \left(\boldsymbol{x}_i, \tilde{\boldsymbol{x}}_i\right)\right\|^2}_{\text{Parallel residual error}} + \underbrace{h_{i,\perp} \left\| \boldsymbol{r}_\perp \left(\boldsymbol{x}_i, \tilde{\boldsymbol{x}}_i\right)\right\|^2}_{\text{orthogonal residual error}} .$$

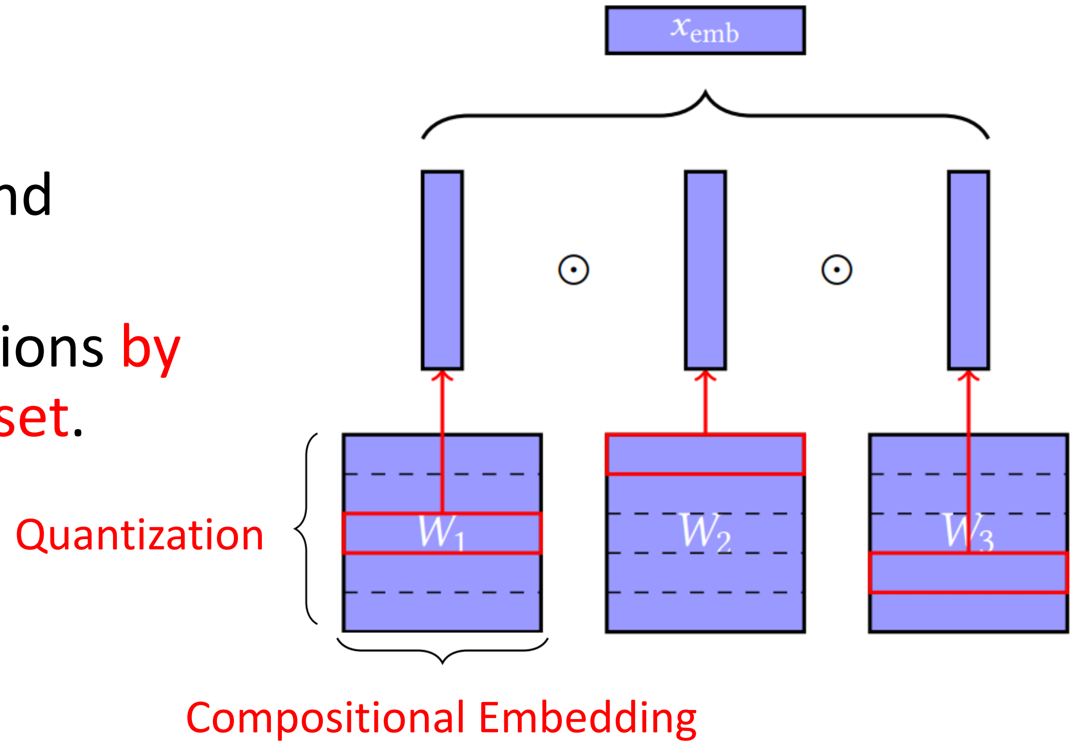
The objective function:

$$\begin{aligned} L^{(i)}(\boldsymbol{C}, \boldsymbol{b_i}) :&= h_{i,\parallel} \left\|\boldsymbol{r}_\parallel\right\|^2 + h_{i,\perp} \left\|\boldsymbol{r}_\perp\right\|^2 \\ &= \tilde{\boldsymbol{x}}_i^\top \left(\left(h_{i,\parallel} - h_{i,\perp}\right) \frac{\boldsymbol{x}_i \boldsymbol{x}_i^\top}{\left\|\boldsymbol{x}_i\right\|^2} + h_{i,\perp} \boldsymbol{I} \right) \tilde{\boldsymbol{x}}_i \\ &\quad - 2 h_{i,\parallel} \boldsymbol{x}_i^\top \tilde{\boldsymbol{x}}_i + h_{i,\parallel} \left\|\boldsymbol{x}_i\right\|^2 . \end{aligned}$$

Anisotropic Additive Quantization for Fast Inner Product Search, AAAI, 2022

Quantization

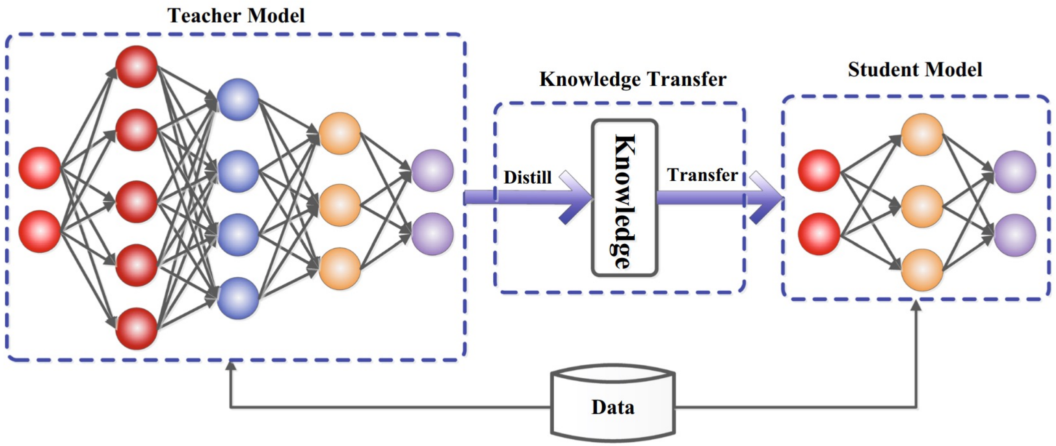
- **Compositional Embedding**
  - The main idea of compositional embedding is to generate meta embedding for each feature based on their characteristics.
- **Compositional Embeddings** – KDD'20
  - Reduce the embedding size in an end-to-end scheme.
  - Split the embedding table into several sections by complementary partitions of the category set.



Compositional Embeddings Using Complementary Partitions for Memory-Efficient Recommendation Systems, KDD, 2020

Model Compression

- Model Compression
  - Hash
  - Quantization
  - Knowledge Distillation
    - Response-based
    - Feature-based
  - Neural Architecture Search
  - Others



KD aims to use a smaller model (Student Model) to approximate the capacity of the original big model (Teacher Model).

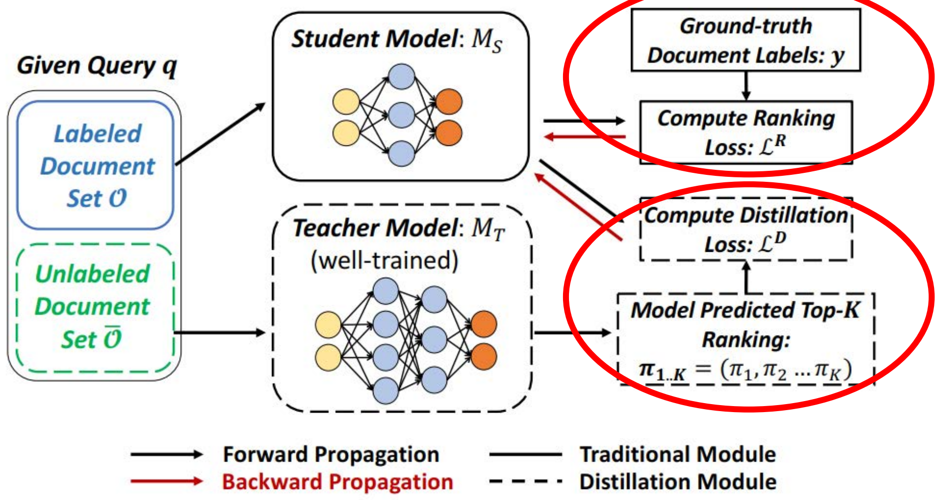
Knowledge Distillation: A Survey, IJCV, 2021

Knowledge Distillation

- **Response-based**
  - Transfer knowledge via the output layer of the teacher model.

$$\mathcal{L}_{res} = \mathcal{L}_R(z_t, z_s)$$

- **Ranking Distillation** – KDD'18
  - RD generates additional top-K training data and labels from unlabeled data set.



Ranking Distillation: Learning Compact Ranking Models With High Performance for Recommender System, KDD, 2018

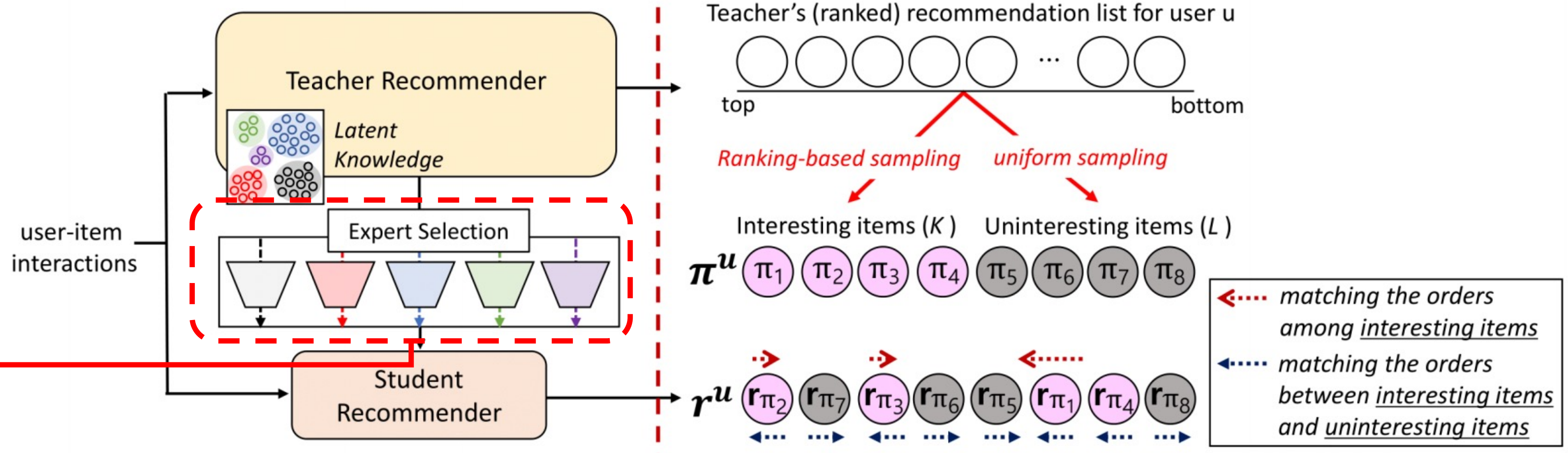
Knowledge Distillation

- **Feature-based**
  - Transfer knowledge in the intermediate layers of the teacher model.

$$\mathcal{L}_{feat} = \mathcal{L}_F(f_t(x), f_s(x))$$

- **DE-RRD** – CIKM'20
  - Adopt multiple experts and propose an expert selection strategy to distill the knowledge.

$$\mathcal{L}(u) = \|h_t(u) - E(h_s(u))\|_2$$



DE-RRD: A Knowledge Distillation Framework for Recommender System, CIKM, 2020

Model Compression

- Model Compression
  - Hash
  - Quantization
  - Knowledge Distillation
  - Neural Architecture Search
    - Embedding Dimension Search
    - Automated Feature Selection
  - Others

$$\min_{\mathcal{A}} \mathcal{L}_{valid}(\mathcal{W}^*(\mathcal{A}), \mathcal{A}),$$

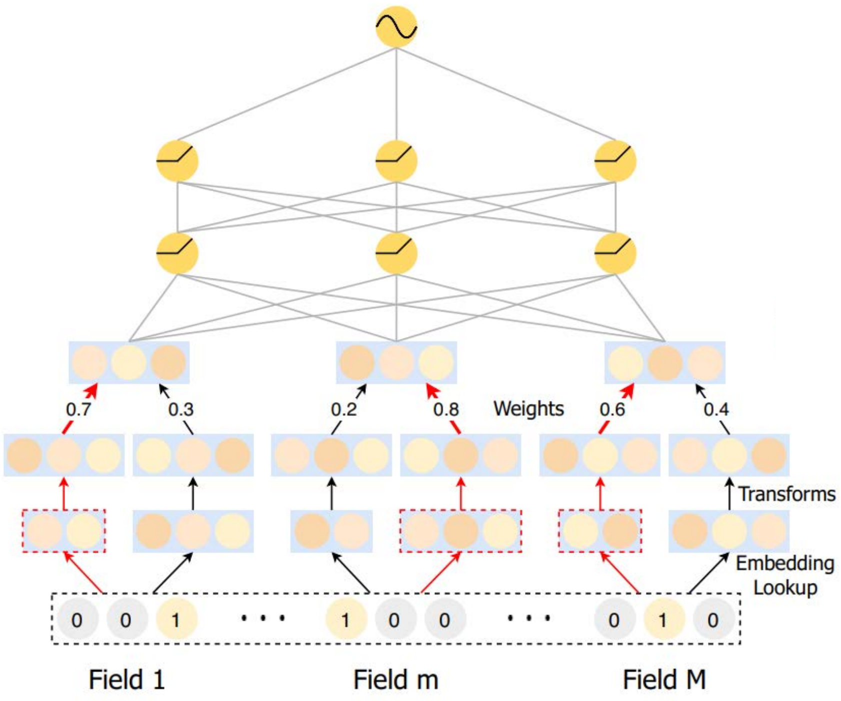
$$s.t.\ \mathcal{W}^*(\mathcal{A}) = arg\min_{\mathcal{W}} \mathcal{L}_{train}(\mathcal{W}, \mathcal{A}).$$

NAS aims to search for the optimal architecture for deep models, which can prune the redundant parameters.

Neural Architecture Search

- **Embedding Dimension Search**
  - Search for optimal and minimal embedding size for each feature, which can compress the embedding layer efficiently.

- **AutoDim** – WWW'21
  - An end-to-end differentiable framework that can calculates the weights over various dimensions.
  - Derive the final architecture according to the maximal weights and retrain the whole model.

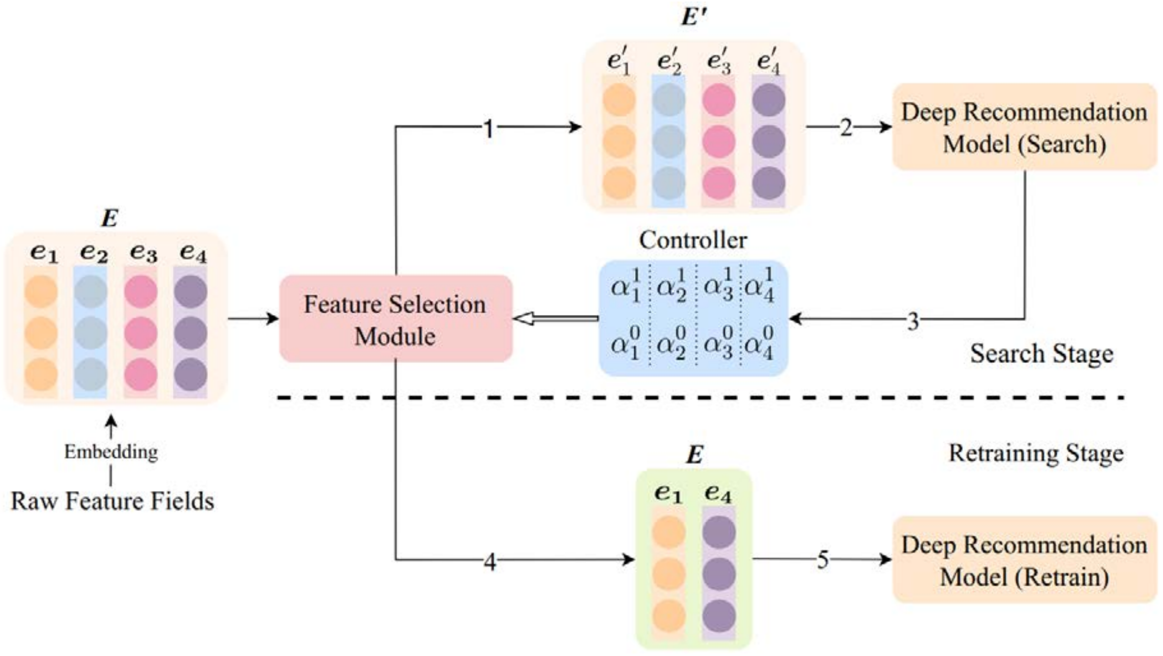


AutoDim: Field-aware Embedding Dimension Search in Recommender Systems, WWW, 2021

Neural Architecture Search

- **Automated Feature Selection**
  - Decrease the number of input features by automated feature selection.

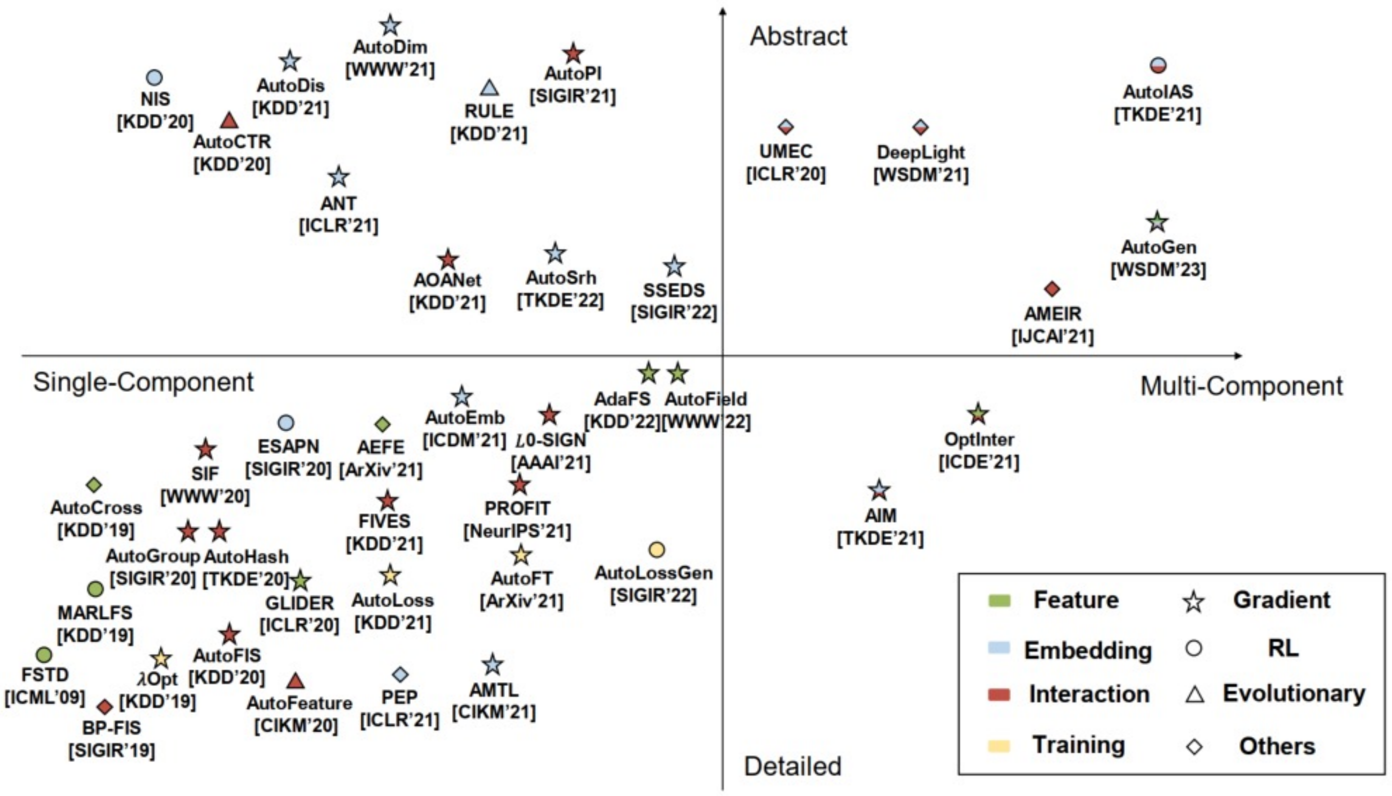
- **AutoField** – WWW'22
  - Equips with a controlling architecture to calculate the drop and select probability of each feature field.
  - Retrain the RS model according to the drop and select probability.



AutoField: Automating Feature Selection in Deep Recommender Systems, WWW, 2022

Neural Architecture Search

- **Survey for AutoML RS**
  - More recent and detailed NAS related works can be found in this survey.



A Comprehensive Survey on Automated Machine Learning for Recommendations, arXiv, 2023

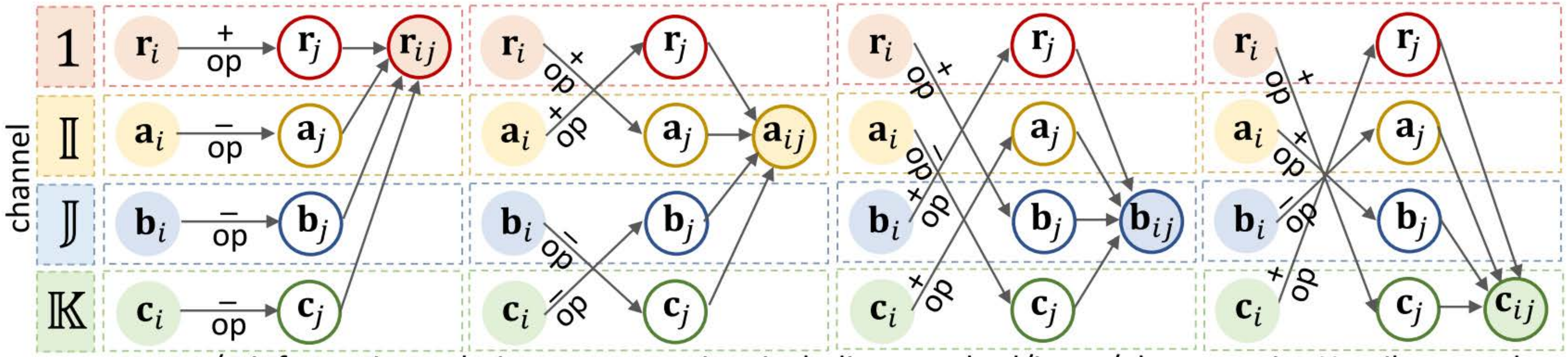
Model Compression

- Model Compression
  - Hash
  - Quantization
  - Knowledge Distillation
  - Neural Architecture Search
  - Others

Others

- **QFM** – TNNLS'21
  - Adopt quaternion representations to substitute the real-valued representation vectors.
  - Parameterize the feature interaction schemes as quaternion-valued functions in the hypercomplex space.

$$q^{\diamond} = r\mathbb{1} + a\mathbb{I} + b\mathbb{J} + c\mathbb{K}$$



+/-: information polarity op: operation, including standard/inner/element-wise Hamilton product

Quaternion Factorization Machines: A Lightweight Solution to Intricate Feature Interaction Modeling, TNNLS, 2021

Conclusion

- Hash, quantization and NAS methods focus on shrinking the embedding layer.
- KD can lightweight the whole model.

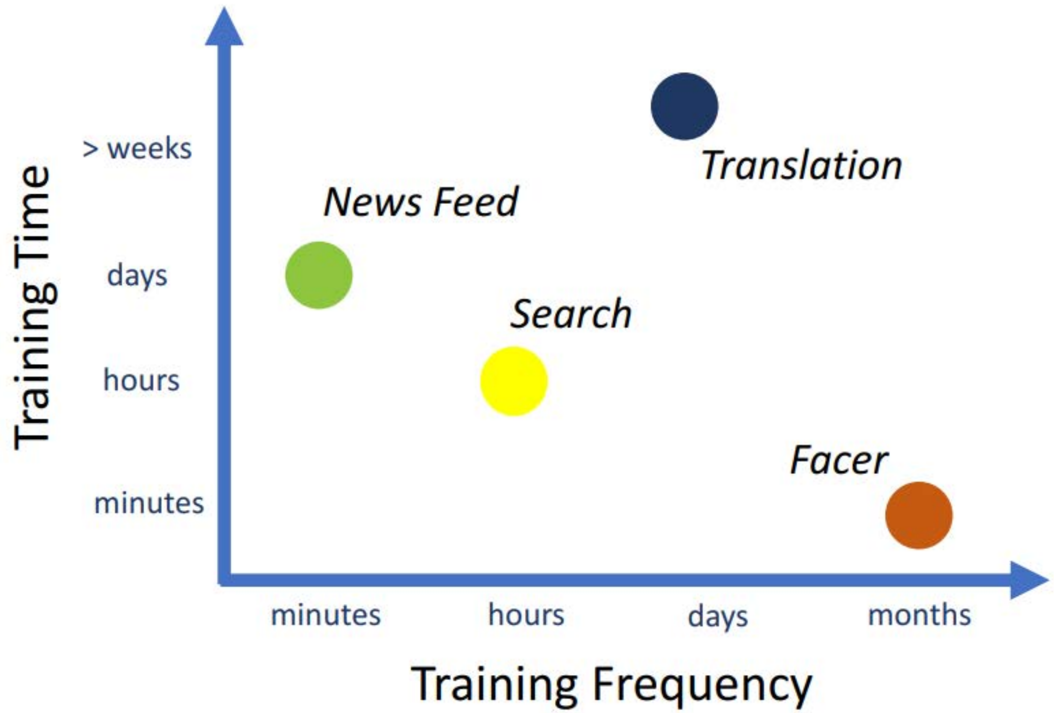
	Embedding Layer	**Middle Layer**
Hash	[80, 209, 307, 438, 456], [184, 227, 313, 355, 422]	[307, 355]
Quantization	[173, 226, 228, 234, 385, 394], [56, 142, 222, 241, 312, 354, 428]	[222, 354, 385]
Knowledge Distillation	[60, 182, 203, 342, 358], [52, 183, 194, 388, 457]	[60, 182, 203, 342, 358], [52, 183, 194, 388, 457]
Neural Architecture Search	[66, 237, 242, 401, 445, 448], [56, 175, 232, 239, 366]	[52, 326]
Others	[128, 311, 332]	[55, 311, 332]

Acceleration Techniques

- Concepts:
  - Model Compression
  - **Acceleration Technique**

➡ Save Computation Resources

- Taxonomy
  - Training Stage
  - Inference Stage



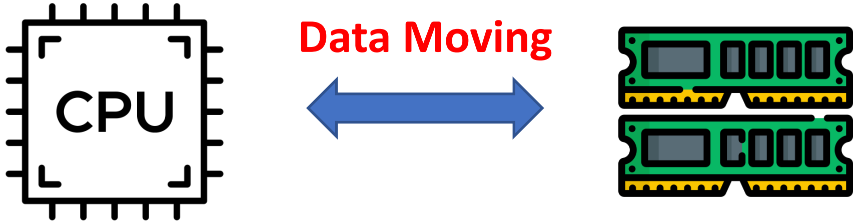
Memory-based Challenge: Difficulty of data access by computation units

Computation-based Challenge: Huge and complex computation

Understanding Training Efficiency of Deep Learning Recommendation Models at Scale, HPCA, 2021

Acceleration Techniques

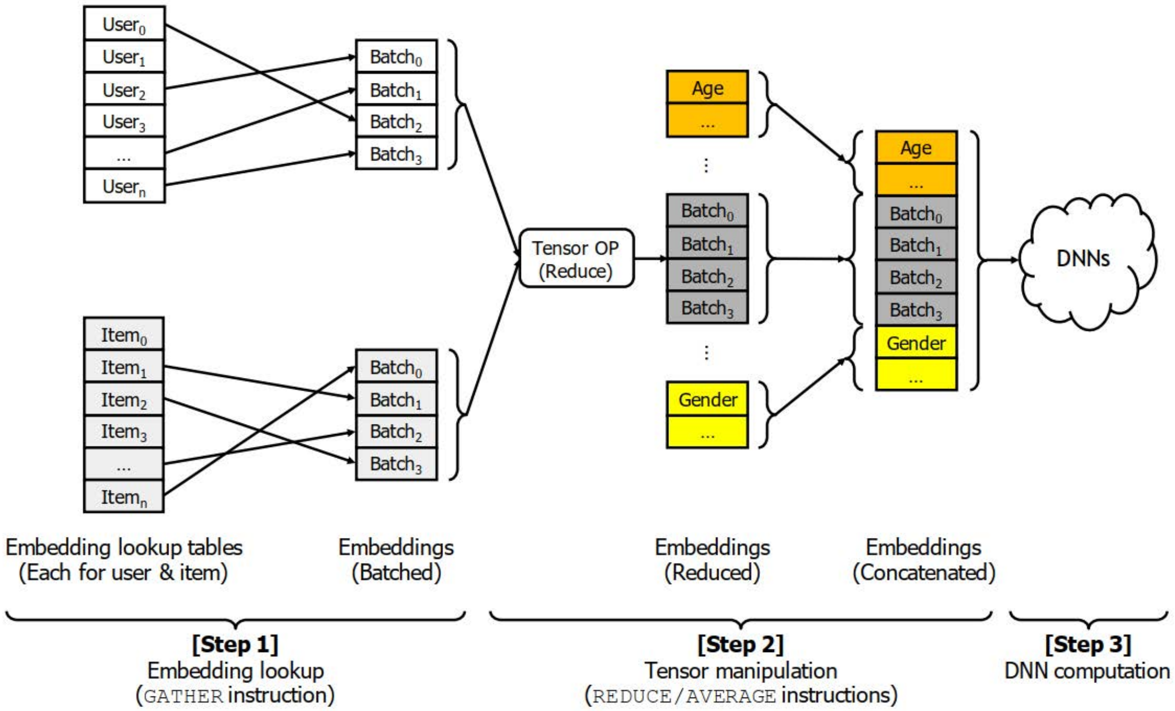
- Acceleration Techniques
  - Hardware-related
    - Near/In Memory Computing
    - Cache Optimization
    - CPU-GPU Co-design
  - Software-related



The computing units advance much, while memory techniques improve slowly. Such gap causes the problem of **memory wall**. Hardware-related methods aim to **optimize data moving** between the storage device and computing units.

Hardware-related

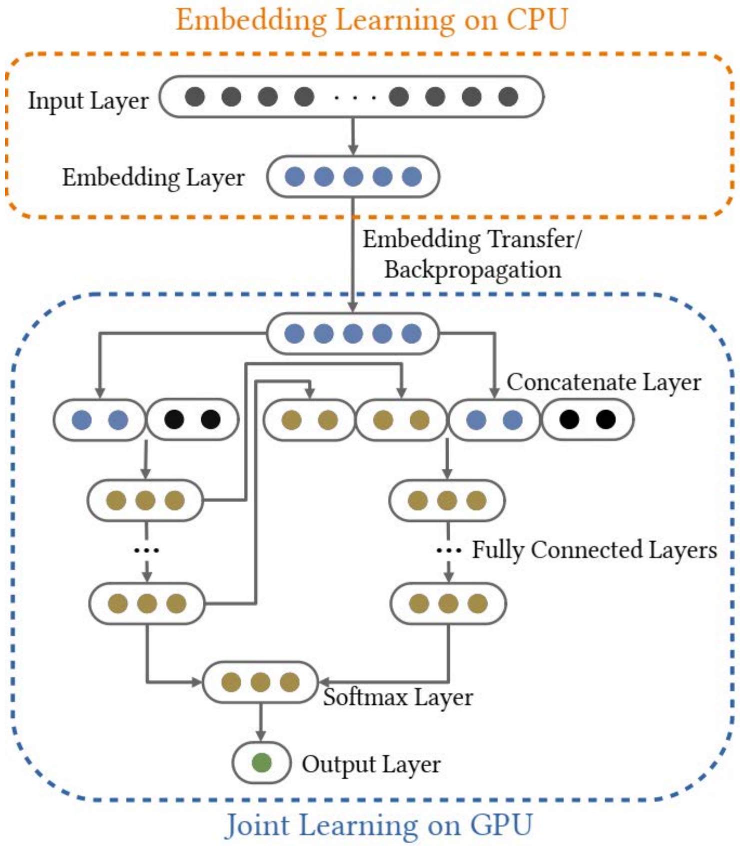
- **Near/In Memory Computing**
  - Put computing units closer to the memory, which can lower the distance of data moving and thus reduce latency.
- **TensorDIMM** – MICRO’19
  - The first to explore architectural solutions for sparse embedding layer.
  - Propose a runtime system to utilize the TensorDIMM for tensor operations.



TensorDIMM: A Practical Near-Memory Processing Architecture for Embeddings and Tensor Operations in Deep Learning, MICRO, 2019

Hardware-related

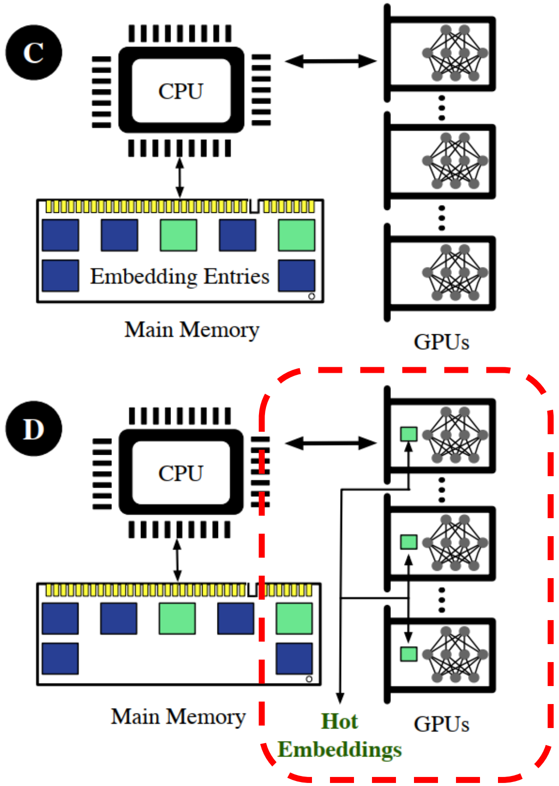
- **Cache Optimization**
  - Optimize the cache allocation mechanism to store the frequently accessed data on the memory device.
- **AIBox** – CIKM'19
  - Partition the model into two parts:
    - (1) Memory-intensive part: Embedding Learning on CPU.
    - (2) Computation-intensive part: Joint Learning on GPU.
  - Leverage SSDs as a secondary storage to cache the embedding table and employ NVLink to reduce GPU data transfer.



AIBox: CTR Prediction Model Training on a Single Node, CIKM, 2019

Hardware-related

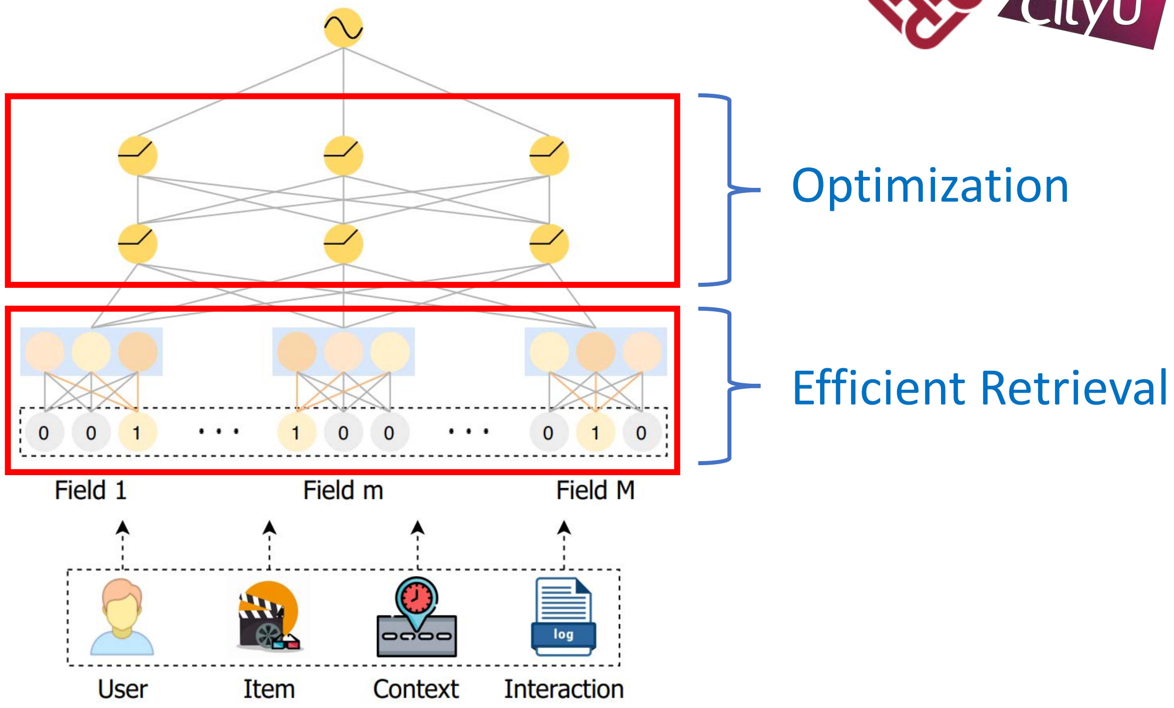
- **CPU-GPU Co-design**
  - Due to huge embedding tables, the embedding part is often stored and processed on CPU and DNN part on CPU. CPU-GPU co-design reduces the communication costs between CPU and GPU.
- **FAE** – VLDB'22
  - Utilize the scarce GPU memory to store the highly accessed embeddings, so it can reduce the data transfers from CPU to GPU.
  - Determine the access pattern of each embeddings by sampling of the input dataset.



Accelerating Recommendation System Training by Leveraging Popular Choices, VLDB, 2022

Acceleration Techniques

- Acceleration Techniques
  - Hardware-related
  - Software-related
    - Optimization
    - Efficient Retrieval



Some designed accelerators for middle layers focus on handling computation challenges. By comparison, embedding layer also needs acceleration.

Software-related

- **Optimization**
  - Accelerate training recommendation models by optimizing its training process.

- **CowClip** – AAAI'23
  - Large batch can speed up training, but suffers from the loss of accuracy.
  - Develop the adaptive column-wise clipping to stabilize the training process under large batch setting.

Algorithm 1 Adaptive Column-wise Clipping(CowClip)

Input: CowClip coefficient r and lower-bound ζ, number of steps T, batch size b, learning rate for dense and embedding η, η_e, optimizer $\texttt{Opt}(\cdot)$

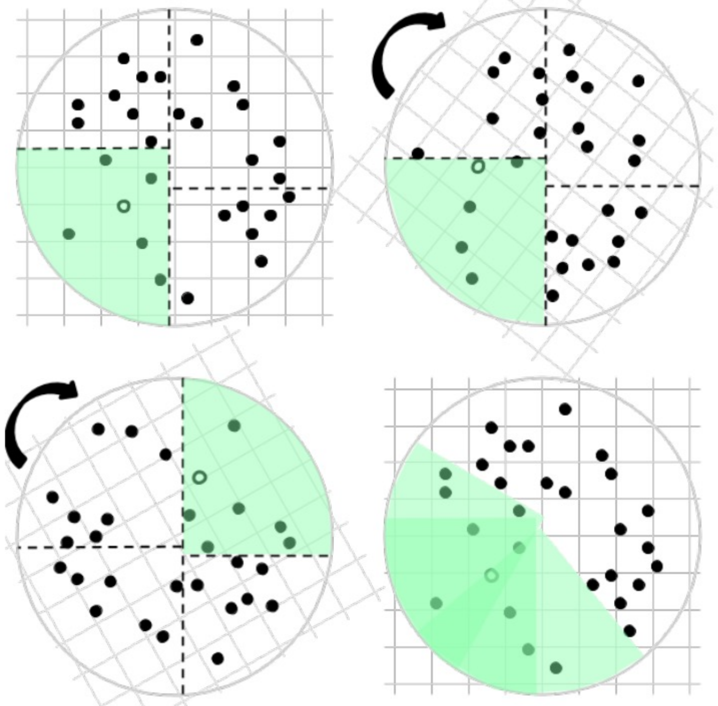
for $t \leftarrow 1$ to T **do**
  Draw b samples B from $\mathcal{D}$
  $\boldsymbol{g}_t, \boldsymbol{g}_t^e \leftarrow \frac{1}{b} \sum_{x \in B} \nabla L(x, w_t, w_t^e)$
  $w_{t+1} \leftarrow \eta \cdot \texttt{Opt}(w_t, \boldsymbol{g}_t)$ // Update dense weights
  for each field and each column in the field **do**
    $n_{\boldsymbol{g}} \leftarrow \|\boldsymbol{g}_t^e[\mathrm{id}_k^{\mathrm{f}_j}]\|$
    $\texttt{cnt} \leftarrow |\{x \in B | \mathrm{id}_k^{\mathrm{f}_j} \in x\}|$ // Number of occurrence
    $\texttt{clip\_t} \leftarrow \texttt{cnt} \cdot \max\{r \cdot \|w_t^e[\mathrm{id}_k^{\mathrm{f}_j}]\|, \zeta\}$ // Clip norm threshold
    $\boldsymbol{g}_c \leftarrow \min\{1, \frac{\texttt{clip\_t}}{n_{\boldsymbol{g}}}\} \cdot \boldsymbol{g}_t^e[\mathrm{id}_k^{\mathrm{f}_j}]$ // Gradient clipping
    $w_t^e[\mathrm{id}_k^{\mathrm{f}_j}] \leftarrow \eta_e \cdot \texttt{Opt}(w_t^e[\mathrm{id}_k^{\mathrm{f}_j}], \boldsymbol{g}_c)$ // Update the id embedding

CowClip: Reducing CTR Prediction Model Training Time from 12 hours to 10 minutes on 1 GPU, AAAI, 2023

Software-related

- **Efficient Retrieval**
  - In industrial, train user and item embeddings offline to represent their preference and attributes, then get recommending list by Embedding-Based Retrieval (EBR) online.

- **Improved KD-Tree** – KDD'19
  - Prove that a kd-tree based on the randomly rotated data can have the same accuracy as RP-tree.
  - Propose a improved kd-tree based on RP-tree with $O(d \log d + \log n)$ query time and guarantee the search accuracy.



Revisiting kd-tree for Nearest Neighbor Search, KDD, 2019

Conclusion

- NMC and Efficient Retrieval are mainly for accelerating inference.
- Cache Optimization, CPU-GPU Co-design and Optimization aim to accelerate training process to save energy.

		Training	Inference
Hardware-related	Near/In Memory Computing	[196]	[78, 164, 190, 195, 367, 371]
	Cache Optimization	[135, 165, 403, 442]	[93, 397]
	CPU-GPU Co-design	[4, 5, 197, 308, 441, 450]	-
Software-related	Optimization	[128, 137, 146, 411, 454]	[140, 141]
	Efficient Retrieval	-	[81, 113, 191, 287], [238, 263, 339, 400]

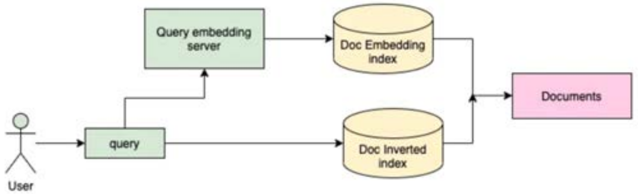
Applications

- **Large Language model**:
  - The emergence of LLMs urge recommendation to step into large model period. The environmental well-being is a vital issue.

ChatGPT

- **Edge Computation**:
  - The combination between edge computation and RS help decrease the latency of service and communication costs.

- **Embedding-based Retrieval Systems**:
  - An efficient EBR system should meet trade-off of three key points: memory, latency and accuracy.



Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

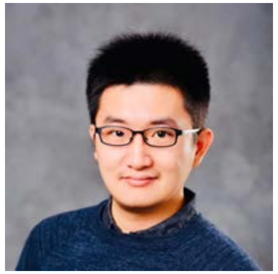
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Background

- Accountability & Auditability
  - What extent users can **trust** the RS
  - Who is **responsible** for the devastating effects brought by RS

responsible

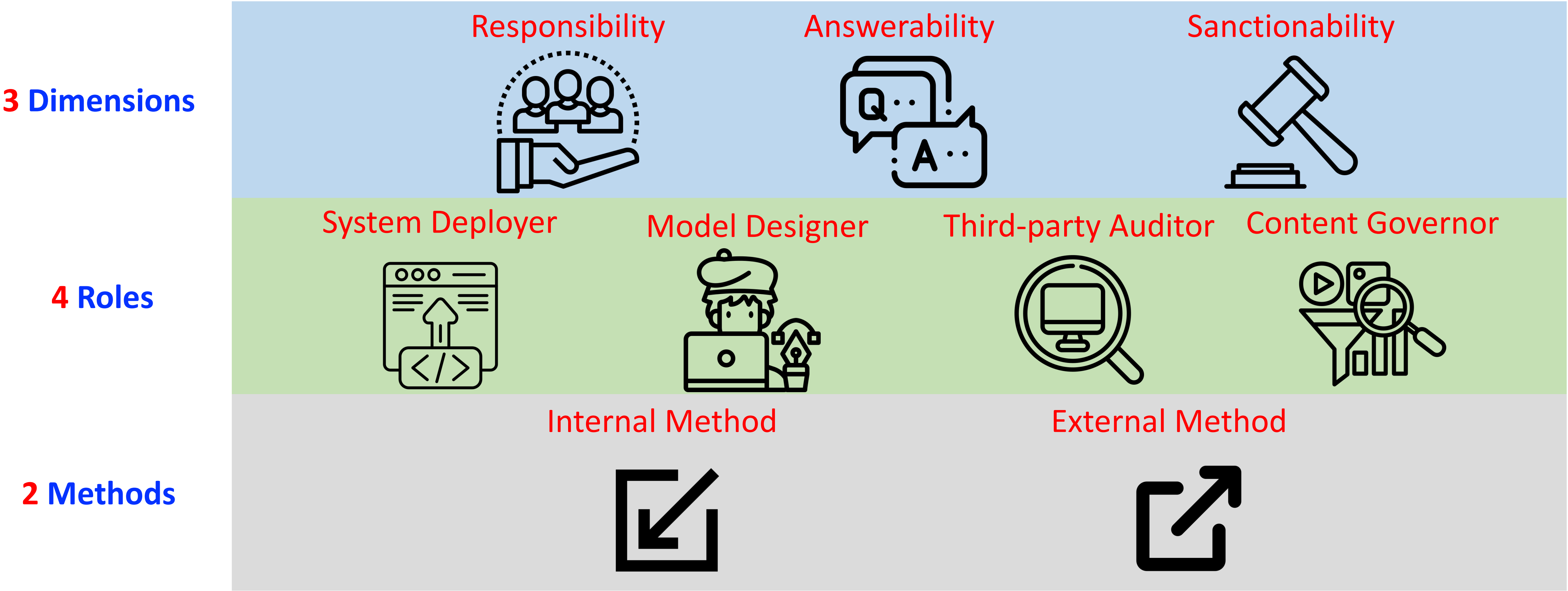
Recommending Videos

trust

Disturbed YouTube for Kids: Characterizing and Detecting Inappropriate Videos Targeting Young Children, ICWSM, 2020

Background

- Accountability & Auditability



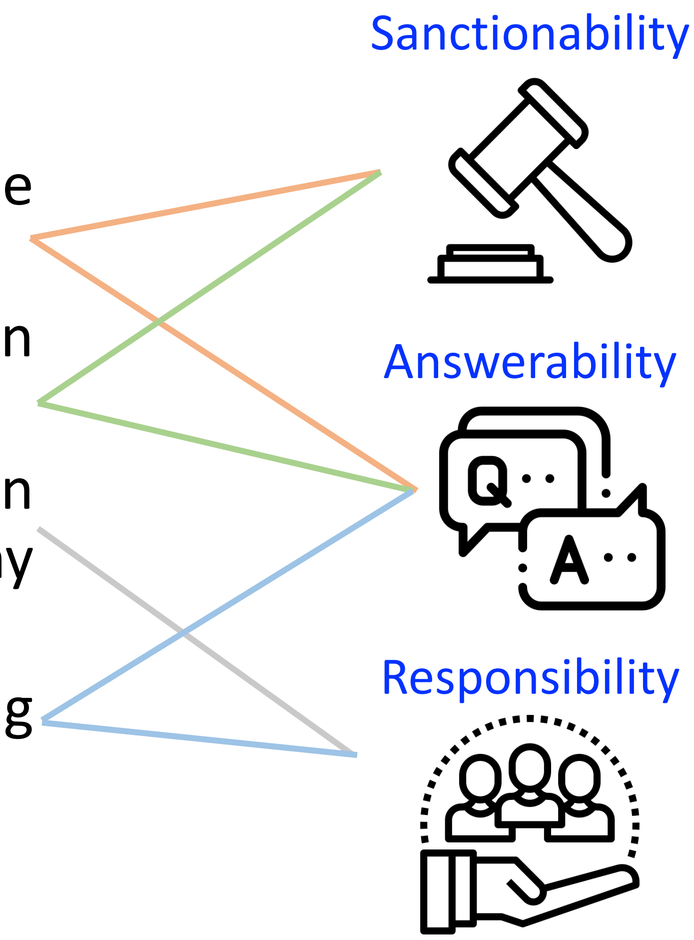
Accountability

- **Three Dimensions of RS Accountability**
  - **Responsibility**: If a user accepts an uncomfortable or illegal recommendation, accountability requires recommender systems to know which part of the system should be blamed.
  - **Answerability**: If an recommender system is accountable, it can reveal the reasons when recommender system has a bad effect.
  - **Sanctionability**: Sanctionability refers that recommender systems should punish and mend the parts that cause harmful impacts.

Accountability

- **Four roles for an accountable RS**

  - **Content Governors**: responsible for examining the facticity and noxiousness of "items" in an RS.
  - **Model Designers**: build the recommendation models for service.
  - **System Deployers**: deploy recommendation models online and check the possible trustworthy problems.
  - **Third-party Auditors**: are responsible for pointing out existing and potential problems in RS.

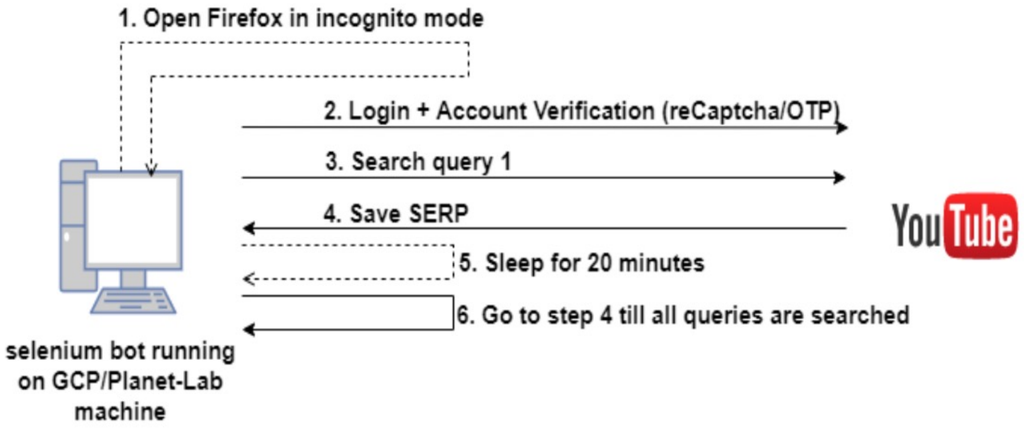


Auditability

- **External Audits**
  - External audits regard recommendation models as a black box, and utilize input and output data from recommender systems to evaluate the algorithm.

- Three procedures for audits:

  1. Collect publicly available data from YouTube.
  2. Classify normal and bad videos (such as radicalized videos) by manual annotations or well-trained classifiers.
  3. Analyze the annotated data to probe problems



Measuring Misinformation in Video Search Platforms: An Audit Study on YouTube, CSCW, 2020

Auditability

- **Internal Audits**
  - Internal audits examine the problems with access to training data.

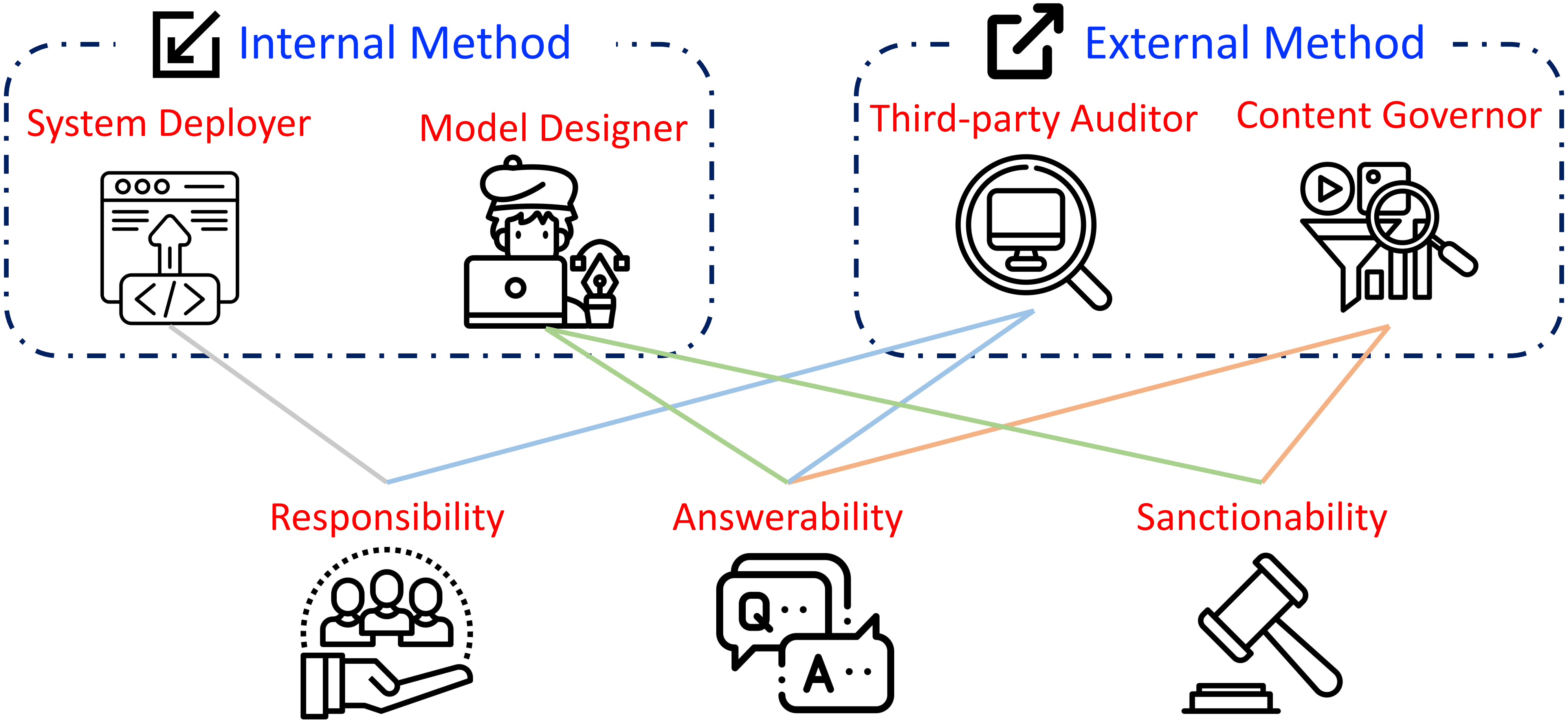
- Model Designers:
  1. Enhance explainability for recommendation models.
  2. Achieve reproducibility of recommendation models.

- System Deployers:
  - Five-step audit method: scoping, mapping, artifact collection, testing, and reflection.

Building and auditing fair algorithms: A case study in candidate screening, FAccT, 2021

Conclusion

- Accountability & Auditability



Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

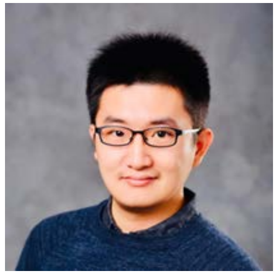
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

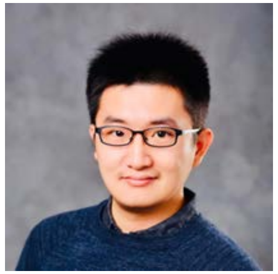
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Interactions

The ideal TRec systems would possess all of six features and advantages

However, it is challenging to consider the modeling of multiple features simultaneously…

Interactions

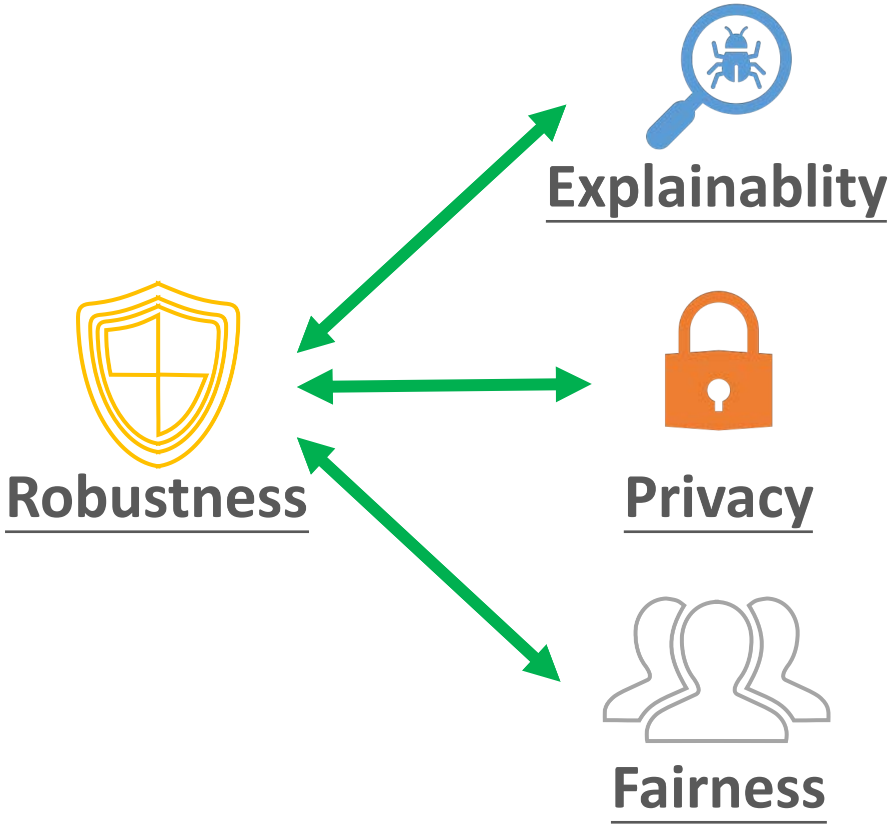
Why? Because these features may have many varying levels of interdependence, and even conflict in some aspects

So here we focus on the **interactions between dimensions with extensive and close ties to other dimensions**

Interactions

- **Interactions with Robustness**
- Interactions with Fairness
- Interactions with Explainability

Interactions with Robustness



These relations are particularly evident in adversarial attacks and robust training

How to use positive dimensions and maintain the balance between conflicting dimensions is important

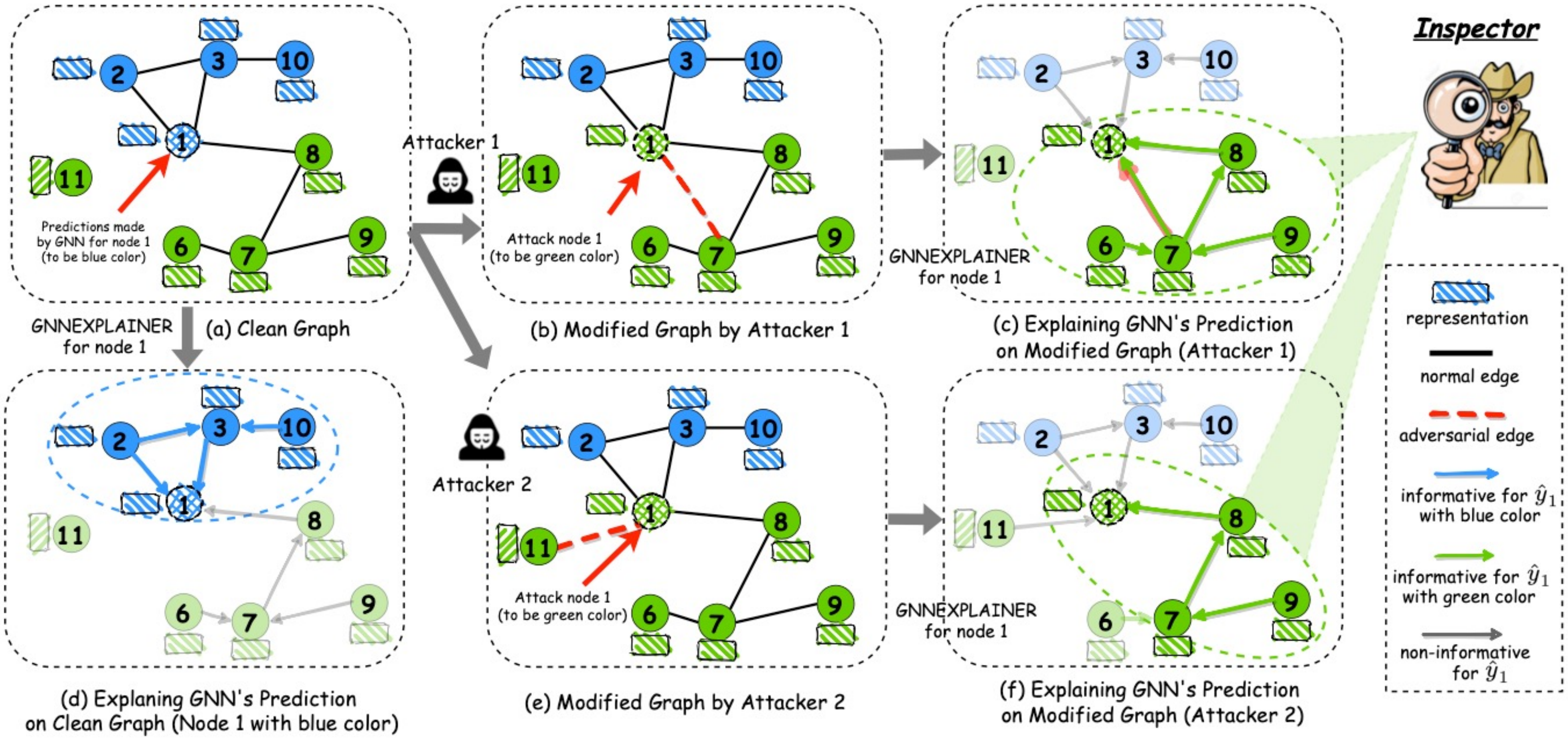
Robustness ⟷ Explainability

- **GEAttack: Jointly Attacking Graph Neural Network and its Explanations**
  - Propose **GEAttack** to jointly attack a graph neural network method and its explanations
  - Investigate interactions between adversarial attacks (robustness) and explainability for the trustworthy GNNs

[1] Wenqi Fan, Han Xu, Wei Jin, Xiaorui Liu, Xianfeng Tang, Suhang Wang, Qing Li, Jiliang Tang, Jianping Wang, and Charu Aggarwal. 2023. Jointly Attacking Graph Neural Network and its Explanations. In 2023 IEEE 39th International Conference on Data Engineering (ICDE). IEEE.

GEAttack - Motivation

- Jointly attack a graph neural network method and its explanations



GEAttack - Problem

- **Problem:** *Given* $G = (\mathbf{A}, \mathbf{X})$, *target (victim) nodes* $v_i \subseteq V_t$ *and specific target label* $\hat{y}_i$, *the attacker aims to select adversarial edges to composite a new graph* $\hat{\mathbf{A}}$ *which fulfills the following two goals: (1) The added adversarial edges can change the GNN's prediction to a specific target label:* $\hat{y}_i = \arg\max_c f_\theta(\hat{\mathbf{A}}, \mathbf{X})^c_{v_i}$; *and (2) The added adversarial edges will not be included in the subgraph generated by explainer:* $\hat{\mathbf{A}} - \mathbf{A} \notin \mathbf{A}_S$.
- The framework under attack:

Node Classification

Two-layer GCN model

$$f_\theta(\mathbf{A}, \mathbf{X}) = \mathrm{softmax}(\tilde{\mathbf{A}}\,\sigma(\tilde{\mathbf{A}}\,\mathbf{X}\,\mathbf{W}_1)\,\mathbf{W}_2)$$

$$\min_\theta \mathcal{L}_{\text{GNN}}(f_\theta(\mathbf{A}, \mathbf{X})) := \sum_{v_i \in V_L} \ell\left(f_\theta(\mathbf{A}, \mathbf{X})_{v_i}, y_i\right) \quad (1)$$

$$= -\sum_{v_i \in V_L} \sum_{c=1}^{C} \mathbb{I}[y_i = c] \ln(f_\theta(\hat{\mathbf{A}}, \mathbf{X})^c_{v_i})$$

GNNExplainer

$$\max_{(\mathbf{A}_S, \mathbf{X}_S)} MI\left(Y, (\mathbf{A}_S, \mathbf{X}_S)\right)$$

$$\rightarrow \min_{(\mathbf{A}_S, \mathbf{X}_S)} H(Y | \mathbf{A} = \mathbf{A}_S, \mathbf{X} = \mathbf{X}_S)$$

$$\approx \min_{(\mathbf{A}_S, \mathbf{X}_S)} -\sum_{c=1}^{C} \mathbb{I}[\hat{y}_i = c] \ln f_\theta(\mathbf{A}_S, \mathbf{X}_S)^c_{v_i}$$

Adversarial Edges

$$\min_{\mathbf{M}_A} \mathcal{L}_{\text{Explainer}}(f_\theta, \mathbf{A}, \mathbf{M}_A, \mathbf{X}, v_i, \hat{y}_i)$$

$$\rightarrow \max_{\mathbf{M}_A} \sum_{c=1}^{C} \mathbb{I}[\hat{y}_i = c] \ln f_\theta(\mathbf{A} \odot \sigma(\mathbf{M}_A), \mathbf{X})^c_{v_i}$$

GEAttack - Method

- Graph Attack:

$$\min_{\hat{\mathbf{A}}} \mathcal{L}_{\text{GNN}}(f_\theta(\hat{\mathbf{A}}, \mathbf{X})_{v_i}, \hat{y}_i) := -\sum_{c=1}^{C} \mathbb{I}[\hat{y}_i = c] \ln(f_\theta(\hat{\mathbf{A}}, \mathbf{X})_{v_i}^c)$$

Perturbation budget: $\|\mathbf{E}'\| = \|\hat{\mathbf{A}} - \mathbf{A}\|_0 \leq \Delta.$

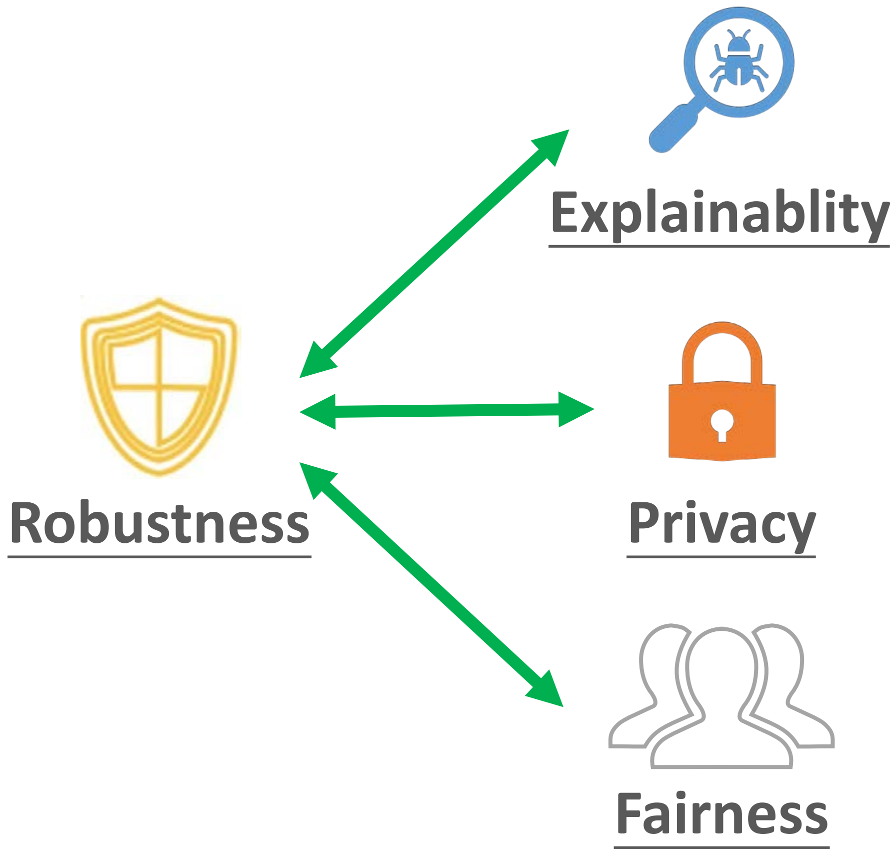
- GNNExplainer Attack:

$$\min_{\hat{\mathbf{A}}} \sum_{v_j \in \mathcal{N}(v_i)} \mathbf{M}_A^T[i,j] \cdot \mathbf{B}[i,j].$$

where $\mathbf{B} = \mathbf{1}\mathbf{1}^T - \mathbf{I} - \mathbf{A}$. $\mathbf{I}$ is an identity matrix, and $\mathbf{1}\mathbf{1}^T$ is all-ones matrix. $\mathbf{1}\mathbf{1}^T - \mathbf{I}$ corresponds to the fully-connected graph. When t is 0, $\mathbf{M}_A^0$ is randomly initialized; while t is larger than 0, $\mathbf{M}_A^t$ is updated with step-size η as follows:

$$\mathbf{M}_A^t = \mathbf{M}_A^{t-1} - \eta \nabla_{\mathbf{M}_A^{t-1}} \mathcal{L}_{\text{Explainer}}(f_\theta, \hat{\mathbf{A}}, \mathbf{M}_A^{t-1}, \mathbf{X}, v_i, \hat{y}_i).$$

More works…



- **Zheng et al.** -> An additive causal model for disentangling user interest and conformity which **Ensures robustness and explainability in recommendation**
- **Bilge et al.** -> **Robust recommendation algorithms** based on collaborative filtering **with privacy enhancement**
- **Zhang et al.** -> A **robust model to combat the attacks** and **ensure the fairness** of the recommender system

[1] Yu Zheng, Chen Gao, Xiang Li, Xiangnan He, Yong Li, and Depeng Jin. 2021. Disentangling user interest and conformity for recommendation with causal embedding. In Proceedings of the Web Conference 2021. 2980–2991.
[2] Alper Bilge, Ihsan Gunes, and Huseyin Polat. 2014. Robustness analysis of privacy-preserving model-based recommendation schemes. Expert Systems with Applications 41, 8 (2014), 3671–3681.
[3] Shijie Zhang, Hongzhi Yin, Tong Chen, Quoc Viet Nguyen Hung, Zi Huang, and Lizhen Cui. 2020. Gcn-based user representation learning for unifying robust recommendation and fraudster detection. In Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 689–698.

Interactions

- Interactions with Robustness
- **Interactions with Fairness**
- Interactions with Explainability

Fairness ↔ Explainability

- **CEF : Counterfactual Explainable Fairness Framework:**
  - Try to explain the recommendation unfairness based on a counterfactual reasoning paradigm
  - An explainability score in terms of the fairness-utility trade-off for feature-based explanation ranking
  - Select the top ones as fairness explanations

[1] Yingqiang Ge, Juntao Tan, Yan Zhu, Yinglong Xia, Jiebo Luo, Shuchang Liu, Zuohui Fu, Shijie Geng, Zelong Li, and Yongfeng Zhang. 2022. Explainable Fairness in Recommendation. arXiv preprint arXiv:2204.11159 (2022).

CEF: Method

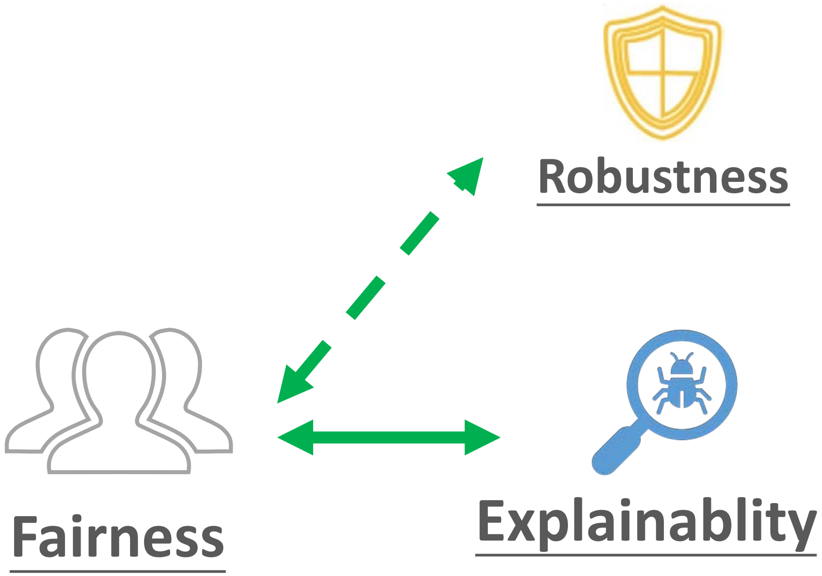
- Overall procedure:

User review information → User-feature matrix and item-feature matrix → Feature-aware recommendation systems → Counterfactual explanations for fairness

- The explainability score (ES):
  - Proximity: the degree of perturbation
  - Validity: the degree of influence on fairness

$$ES = Validity - \beta \cdot Proximity,$$

More works…



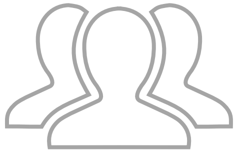
- **Chen et al.** -> Research on **fairness** and analyzes the **explainability** of the model at the same time
- **Fu et al.** -> A **fairness-aware explainable recommendation model**

[1] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and debias in recommender system: A survey and future directions. ArXiv preprint abs/2010.03240 (2020). https://arxiv.org/abs/2010.03240
[2] Zuohui Fu, Yikun Xian, Ruoyuan Gao, Jieyu Zhao, Qiaoying Huang, Yingqiang Ge, Shuyuan Xu, Shijie Geng, Chirag Shah, Yongfeng Zhang, et al . 2020. Fairness-aware explainable recommendation over knowledge graphs. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 69–78.

Interactions

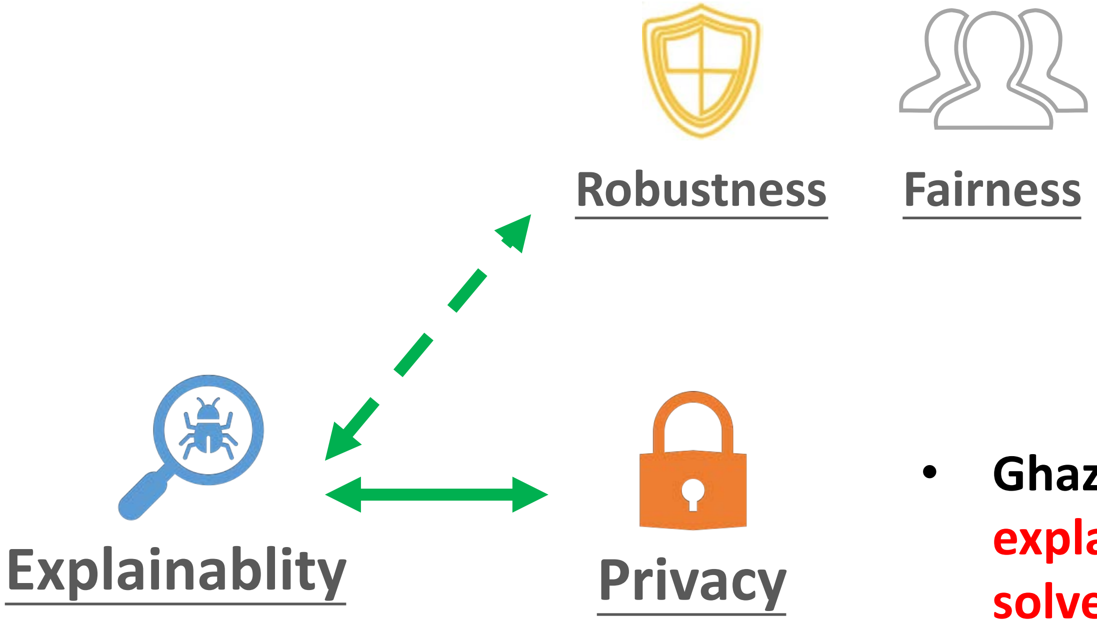
- Interactions with Robustness

- Interactions with Fairness



- **Interactions with Explainability**

Interactions with Explaianablity



- **Ghazimatin et al.** -> Provide a new **counterfactual explanation mechanism** for recommendation, which **also solved the privacy exposure problem**

[1] Azin Ghazimatin, Oana Balalau, Rishiraj Saha Roy, and Gerhard Weikum. 2020. PRINCE: Provider-side interpretability with counterfactual explanations in recommender systems. In Proceedings of the 13th International Conference on Web Search and Data Mining. 196–204.

Summary

- **Interaction is challenging -> Consider the modeling of multiple features simultaneously**

- **We focus on the interactions between dimensions with extensive and close ties to other dimensions**

- **Three mainly considered interactions:**
  - Interactions with Robustness
  - Interactions with Fairness
  - Interactions with Explainability

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

Jingtong Gao

Privacy

Lin Wang

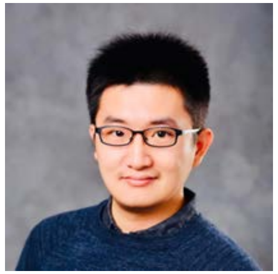
Environmental Well-being

Accountability & Auditability

Qidong Liu

Dimension Interactions

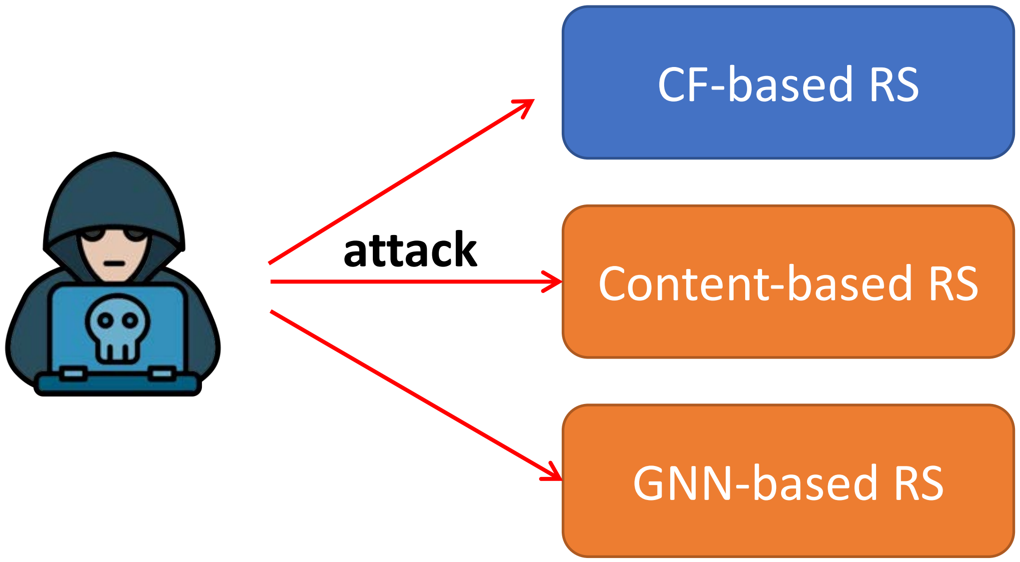
Future Directions



Xiangyu Zhao

Future Directions in Six Dimensions

- **Robustness**
  - ***Research on other RS models:*** more robust-related researches can investigate other RS models in the future, such as GNN-based RS and content-based RS, but not only the CF-based RS model.
  - ***Adversarial robust training methods***: generate adversarial perturbations on user-item interactions, instead of only on parameter space.



Future Directions in Six Dimensions

- **Non-discrimination & Fairness**
  - ***Consensus on fairness definitions***: (1) priority of fairness objectives; (2) suitable fairness metrics; (3) multiple fairness notions.
  - ***Trade-off between fairness and utility***: design a trade-off mechanism so that the decision–makers can make a better balance.

- **Privacy**
  - ***Comprehensive privacy protection***: propose a comprehensive privacy protection framework to protect against multiple privacy attacks.
  - ***Defence against shadow training***: investigating how to defend against shadow training methods is crucial for privacy protection, because most attack methods use it to train attackers.

Future Directions in Six Dimensions

- **Explainability**
  - ***Natural Language Generation for Explanation***: explore the explainable RS with natural language sentences to be more user-friendly.
  - ***Explainable recommendations in more fields***: except for e-commerce, develop explainable recommendations for healthcare, education and etc.

Item: Last Stand of the 300 User interest: war, history, documentary

(a) Post-hoc	Alice and 7 of your friends like this.
	Because you watched Spartacus, we recommend Last Stand of the 300.
(b) Embedded-F	You might be interested in documentary, on which this item performs well.
(c) Embedded-S	I agree with several others that this is a good companion to the movie.
(d) Joint	**This is a very good movie.**
(e) Ours	**This is a very good documentary about the battle of thermopylae.**

Pre-defined template Retrieved from explanations written by others **Generated by RNNs**

Co-Attentive Multi-Task Learning for Explainable Recommendation, IJCAI, 2019

Future Directions in Six Dimensions

- **Environmental Well-being**
  - ***Cost measurement for RS***: develop a framework to measure and predict the energy consumption for recommender systems specifically.
  - ***Trade-off between consumption and accuracy***: design a trade-off mechanism to produce the highest utility for RS.

- **Accountability & Auditability**
  - ***Combination of many accountability aspects***: design the auditability method to consider multiple accountability aspects, simultaneously.

Future Directions in Other Dimensions

- **Interactions among different dimensions**
  - Explore multiple aspects combinations to reach more requests of trustworthy dimensions.
  - Resolve the conflicts between several directions to avoid ruin the efforts for trustworthiness.

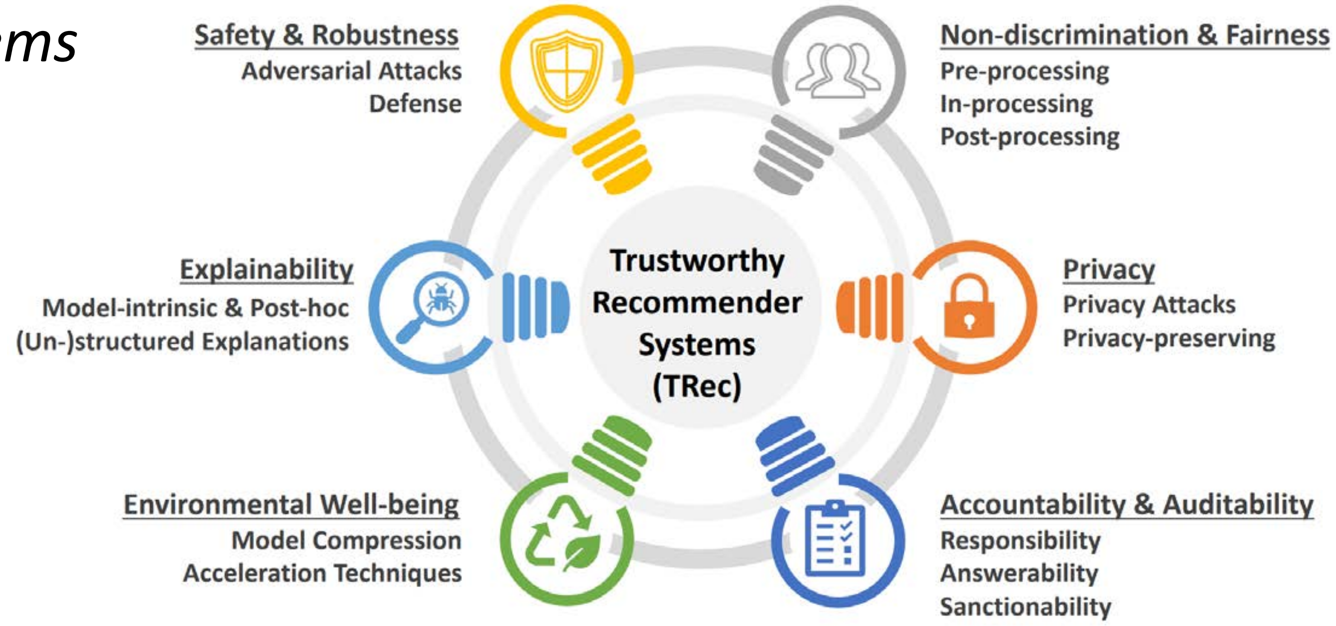
Future Directions in Other Dimensions

- **Other Dimensions to achieve TRec**
  - ***Security***: In medication or industrial scenes, the RS will affect human decisions directly, and any improper decision can cause uncountable losses to life and property.
  - ***Controllability***: controllability can help stop harmful recommendations and minimize the horrible effects, when a recommender system causes a devastating effect

- **Technology Ecosystem for TRec**
  - Develop an integrated technology ecosystem, including datasets, metrics, toolkits, etc., to be convenient for the TRec researches

Conclusion

- **Six of the most critical dimensions for TRec**
  - ✓ ***safety & robustness, non-discrimination & fairness, explainability, privacy, environmental well-being, and accountability & auditability***.
  - *Concepts an& Taxonomy*
  - *Summary of the Representative Methods*
  - *Applications in Real-world Systems*
  - *Surveys & Tools*
  - *Future Directions*



Q&A

Wenqi Fan
The Hong Kong Polytechnic University

Xiangyu Zhao
City University of Hong Kong

A Comprehensive Survey on Trustworthy Recommender Systems

https://arxiv.org/pdf/2209.10117.pdf