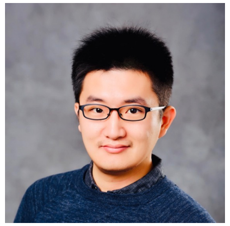
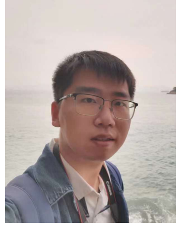
Trustworthy Recommender Systems

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

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

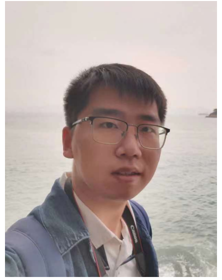
Jingtong Gao

Privacy

Lin Wang

Environmental Well-being

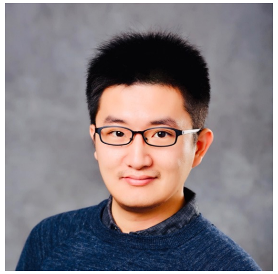
Accountability & Auditability



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

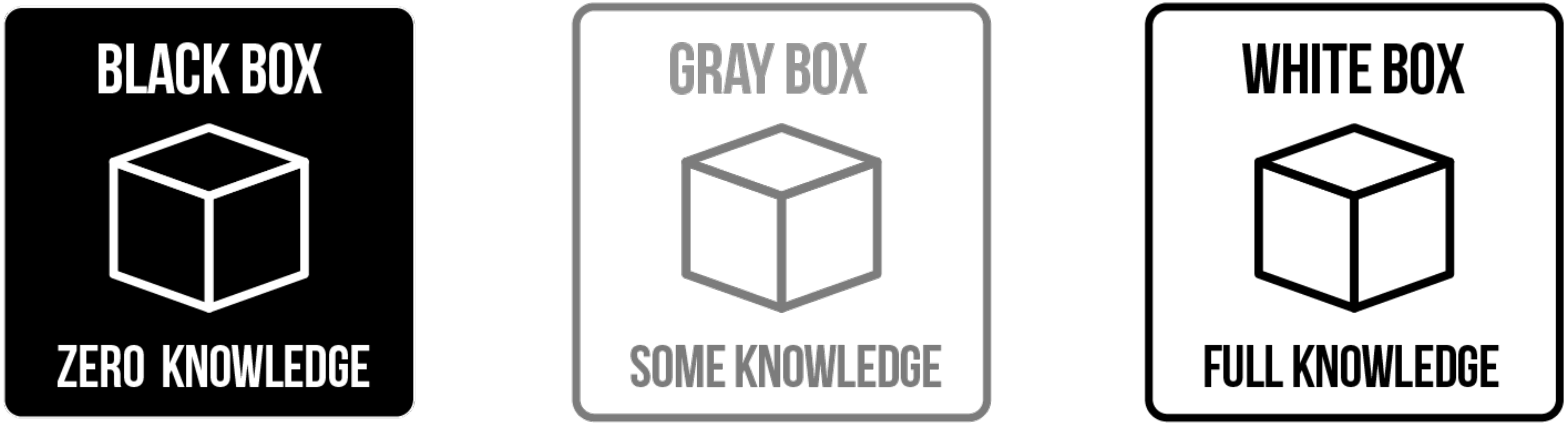
- Concepts and Taxonomy
- Privacy Attack Methods
- Privacy-preserving Methods.
- Applications
- Survey and Tools
- Future Directions

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

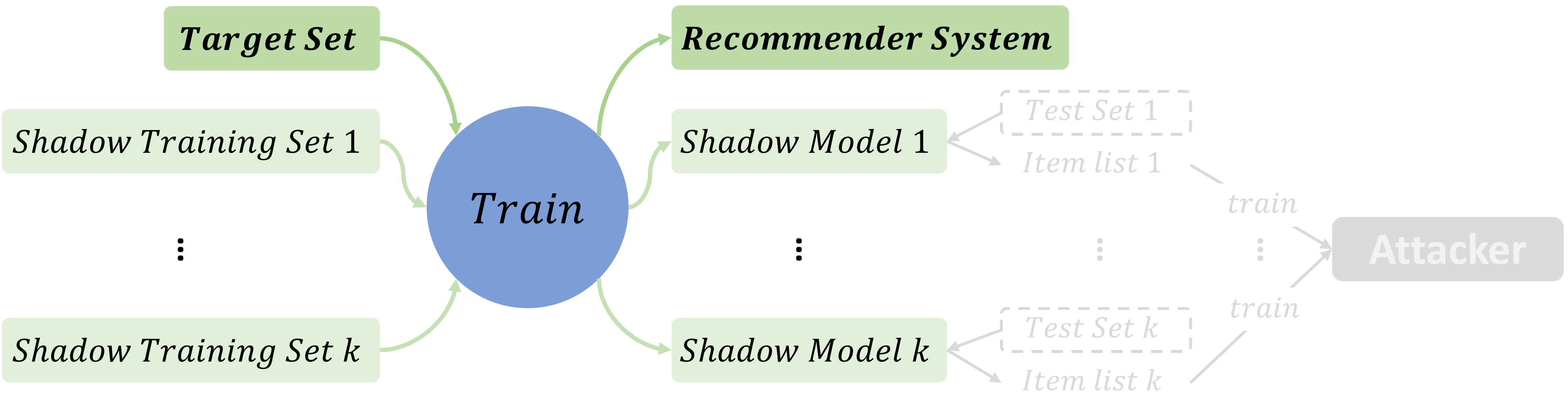
Privacy

- 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]

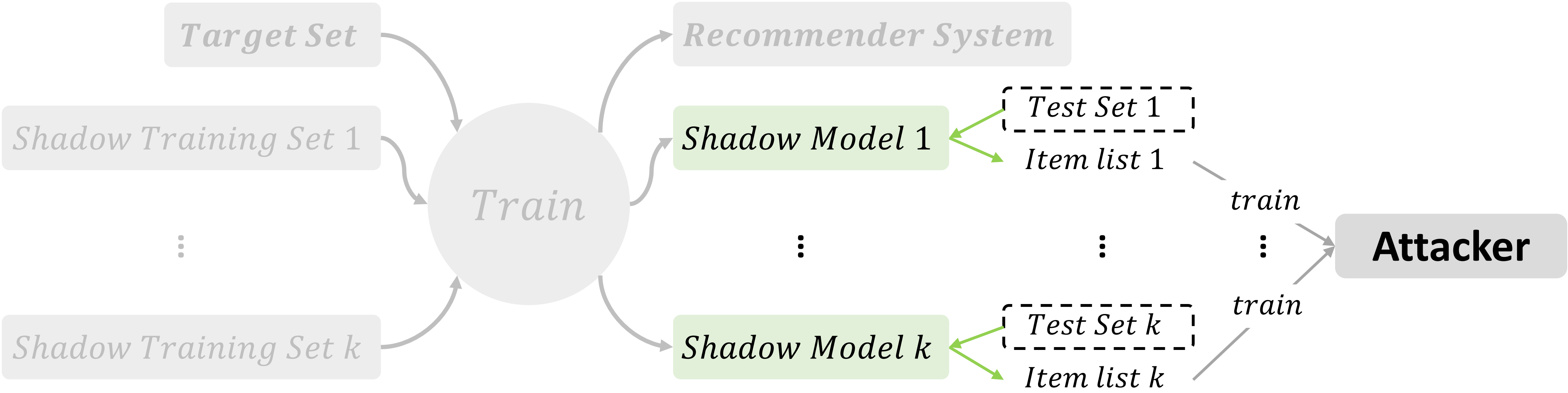
Membership Inference Attacks



Shadow training

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

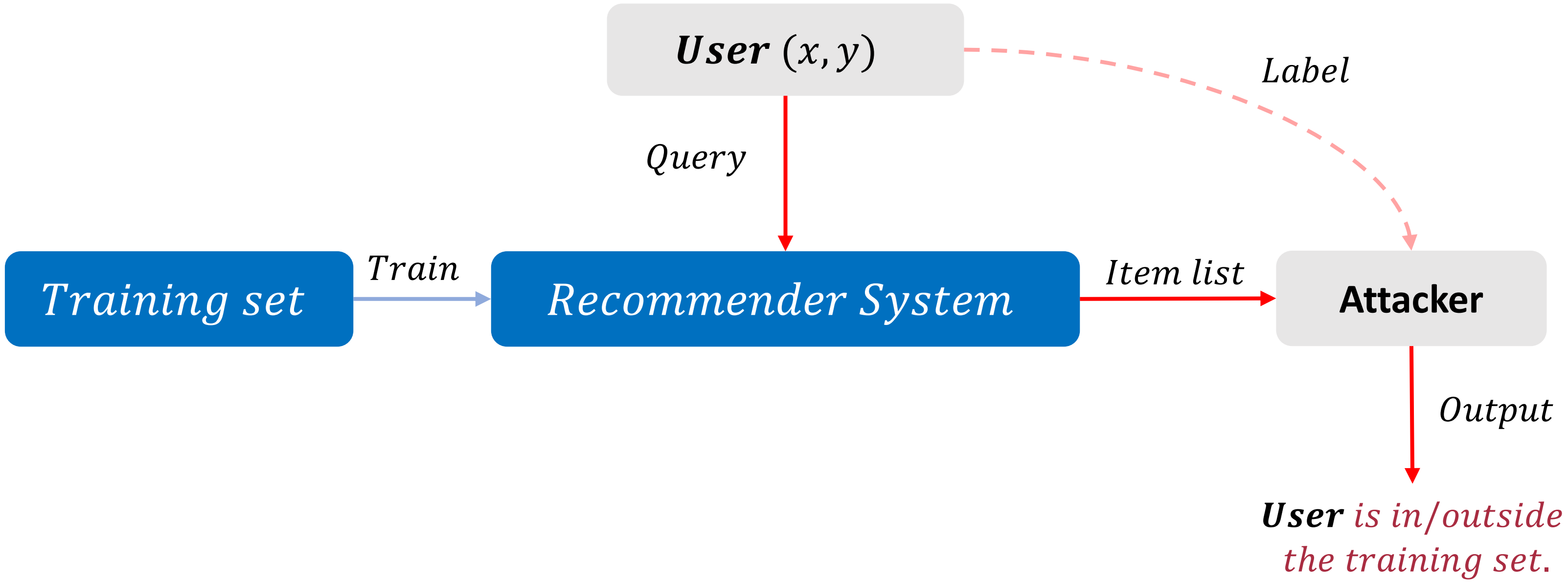
Membership Inference Attacks



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

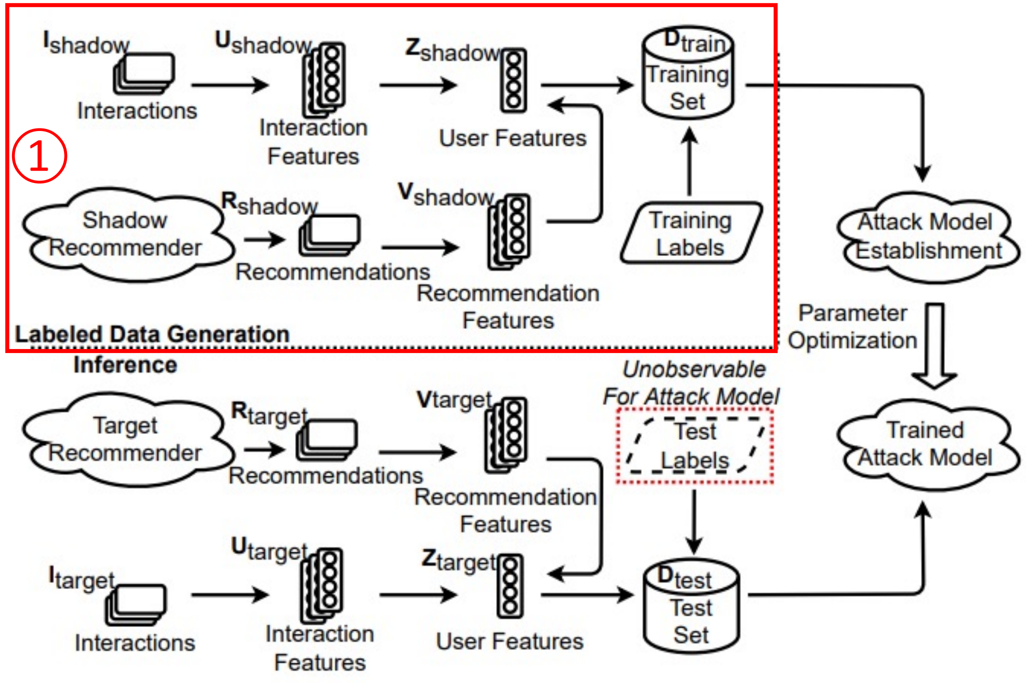


Figure 2: The framework of the membership inference attack against a recommender system.

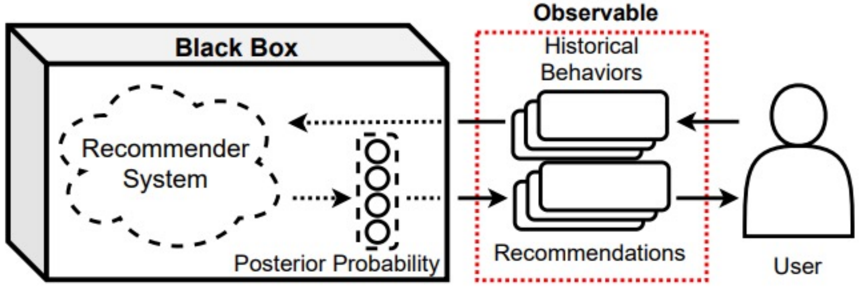


Figure 1: An example of recommender systems.

Membership Inference Attacks in Recommender Systems

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

Membership Inference Attacks

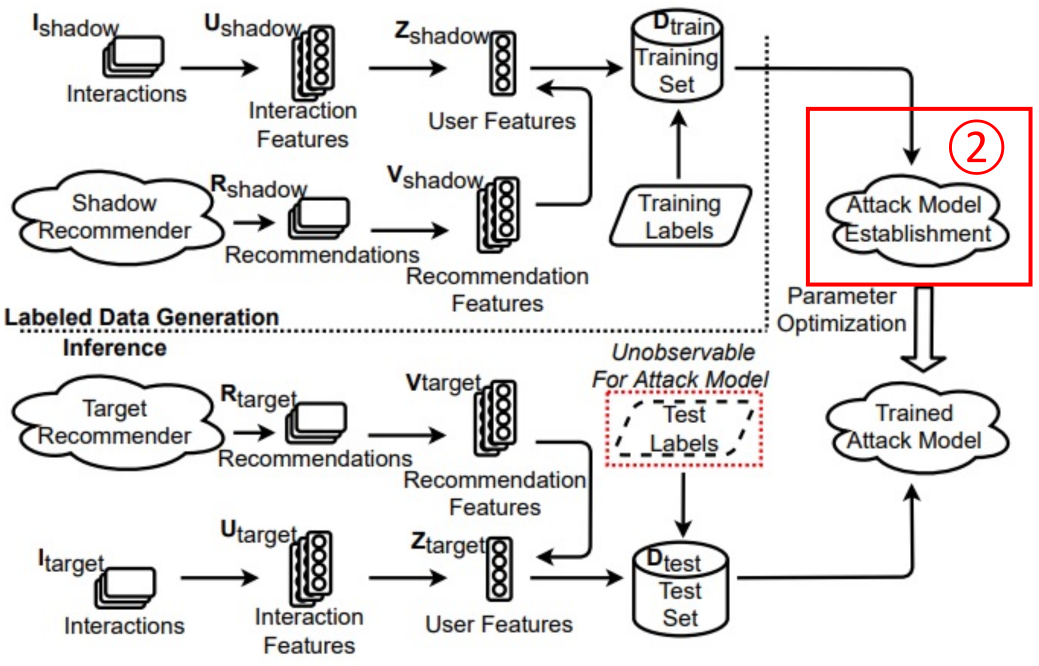


Figure 2: The framework of the membership inference attack against a recommender system.

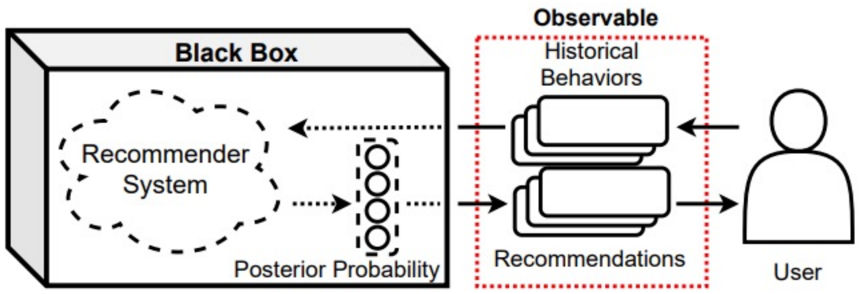


Figure 1: An example of recommender systems.

Membership Inference Attacks in Recommender Systems

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

Membership Inference Attacks

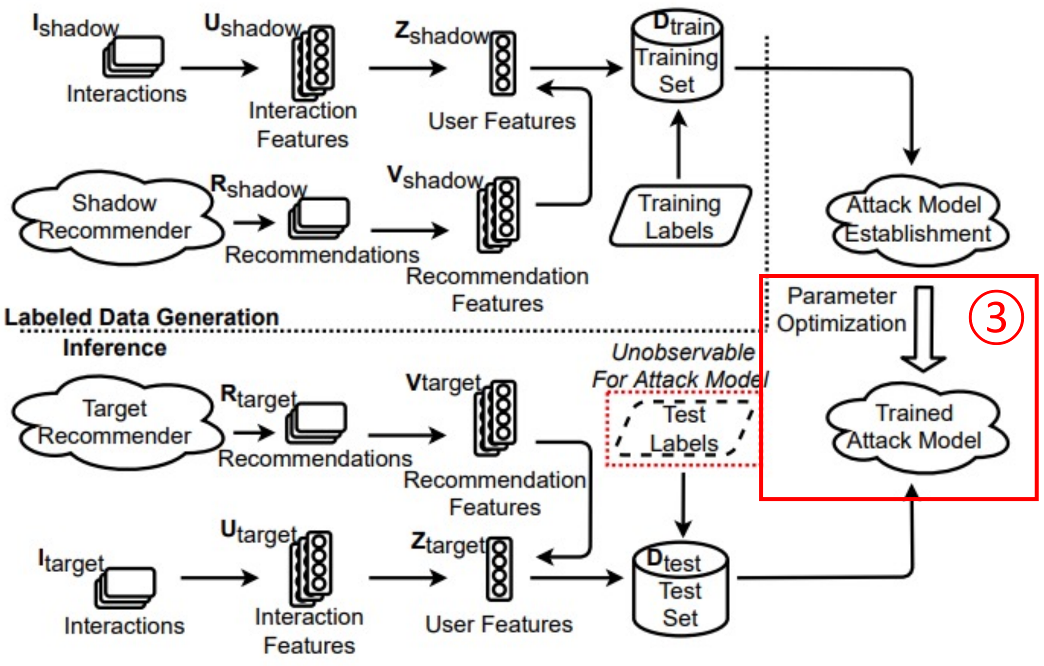


Figure 2: The framework of the membership inference attack against a recommender system.

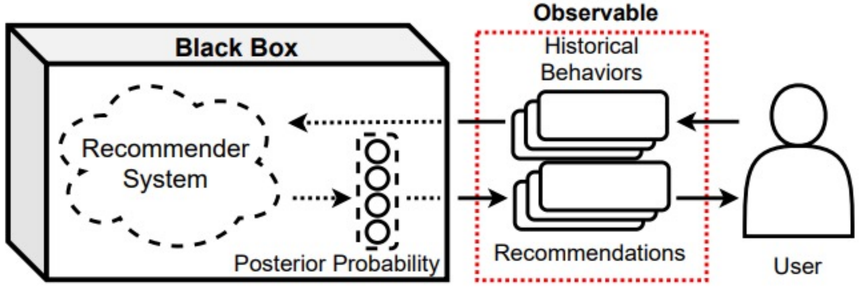
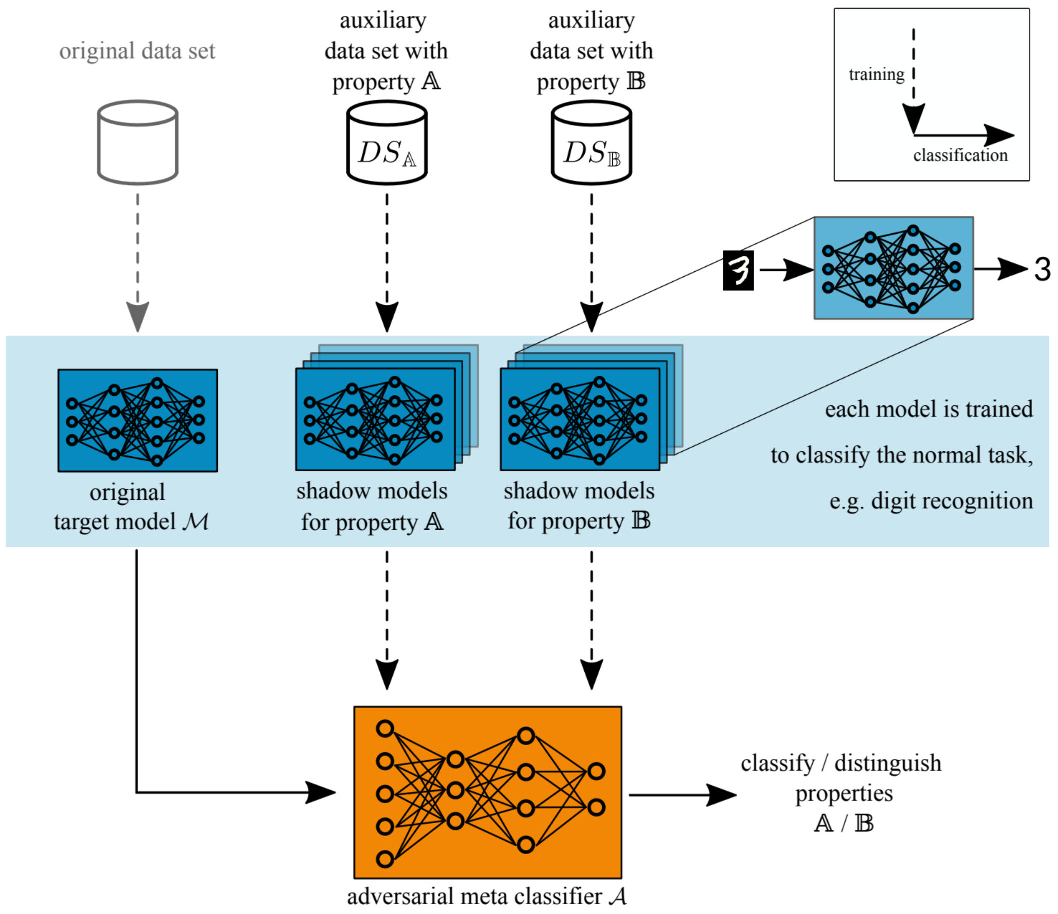


Figure 1: An example of recommender systems.

Membership Inference Attacks in Recommender Systems

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

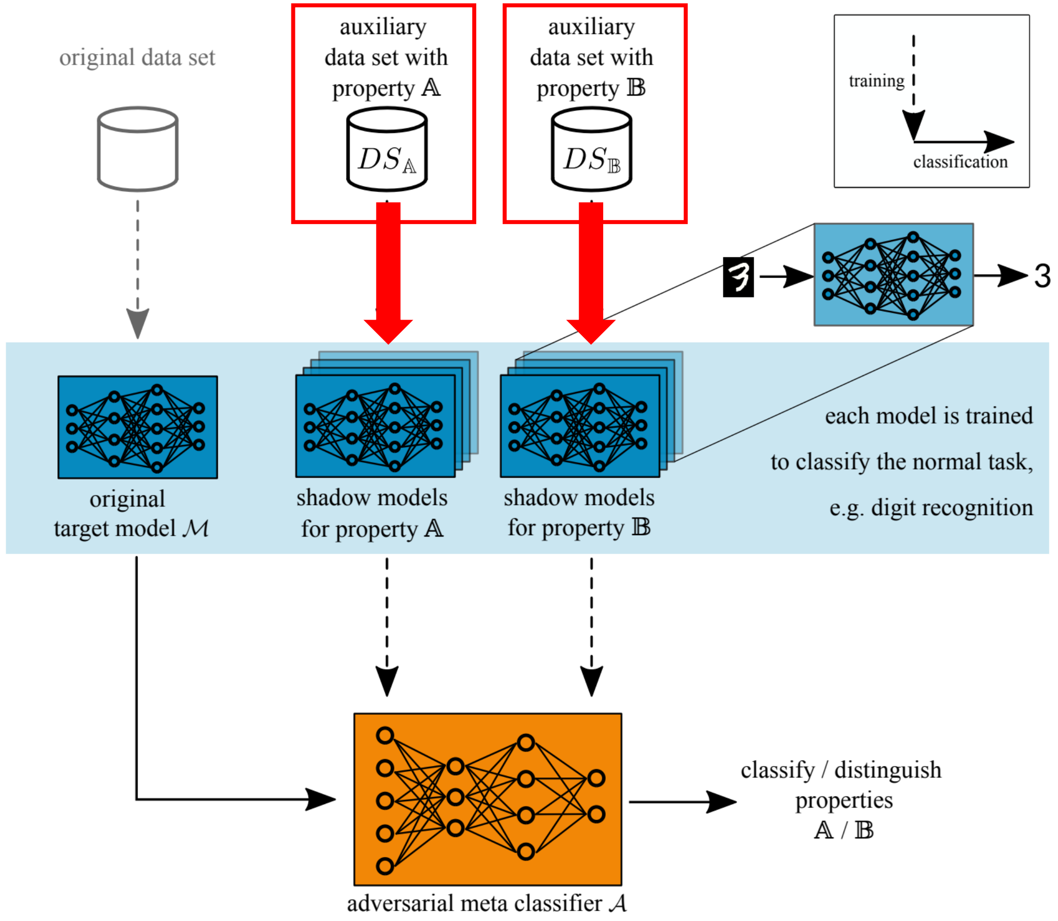
Property Inference Attacks



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.

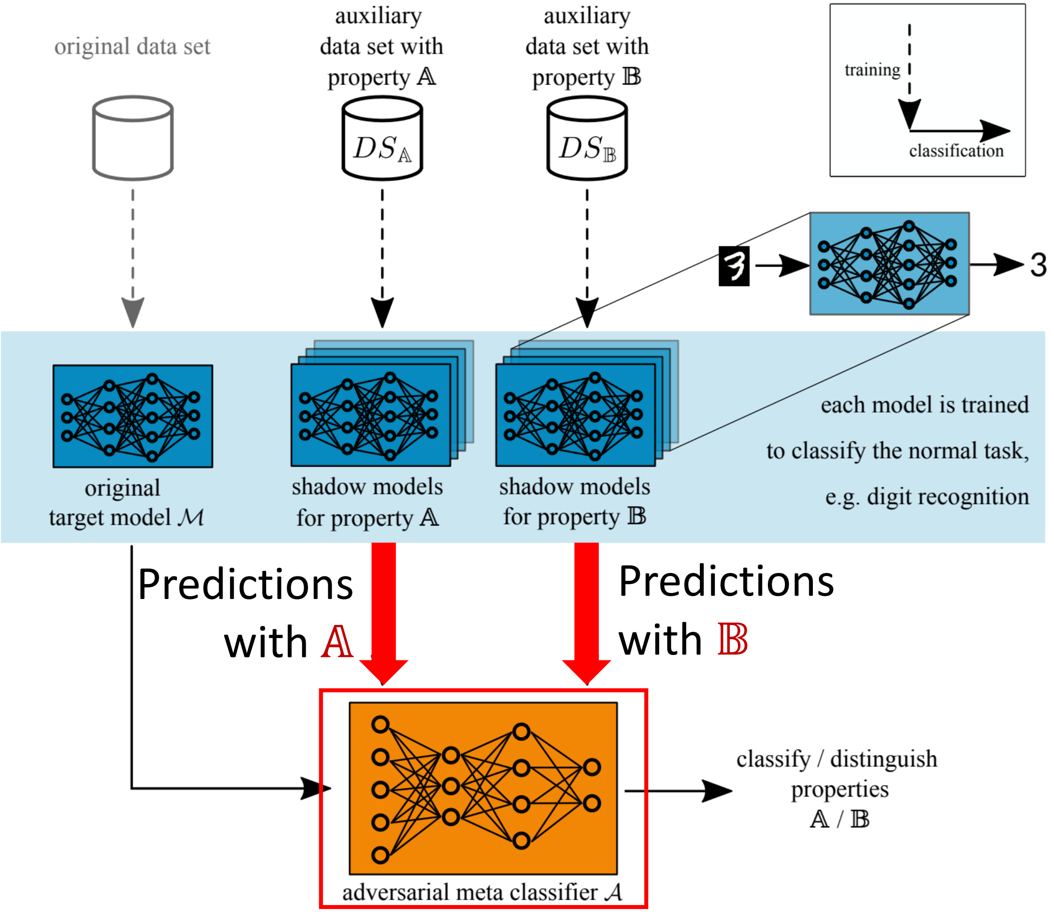
Property Inference Attacks



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.

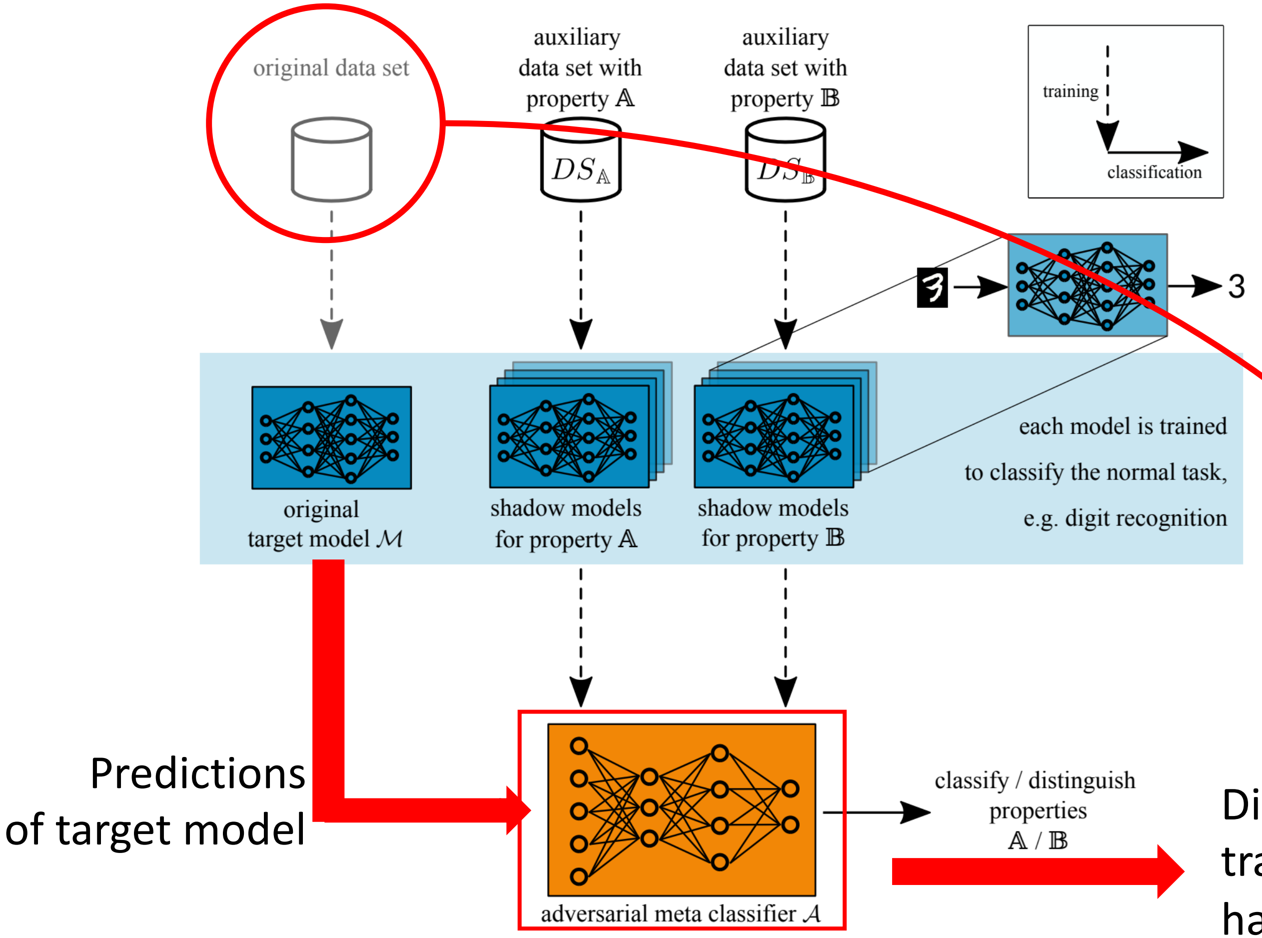
Property Inference Attacks



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.

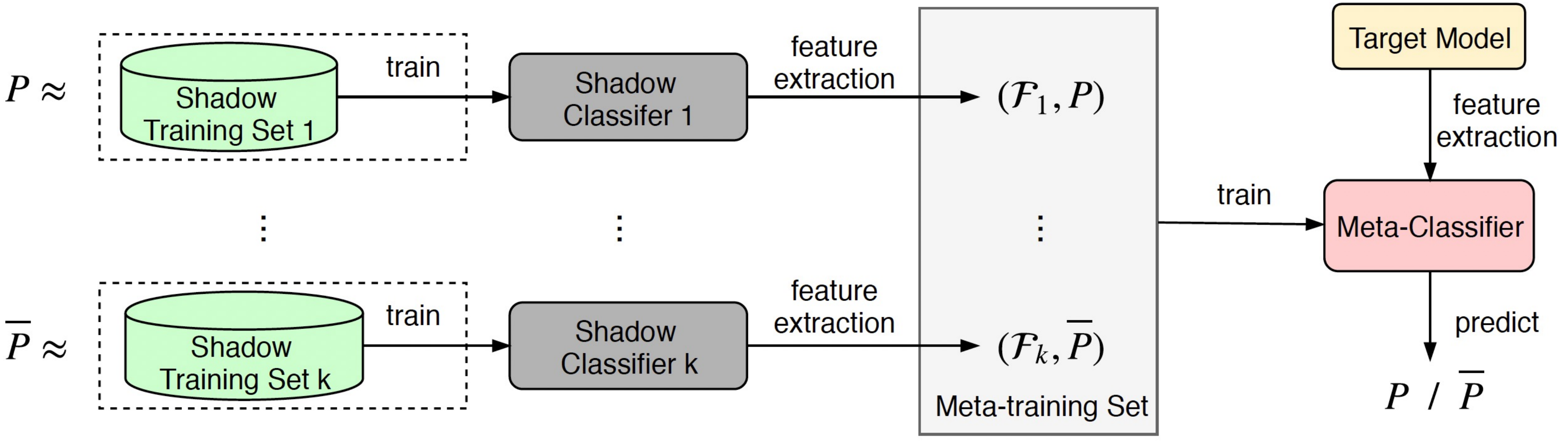
Property Inference Attacks



Distinguish weather the training data of target model has the property 𝔸/𝔹 or not.

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

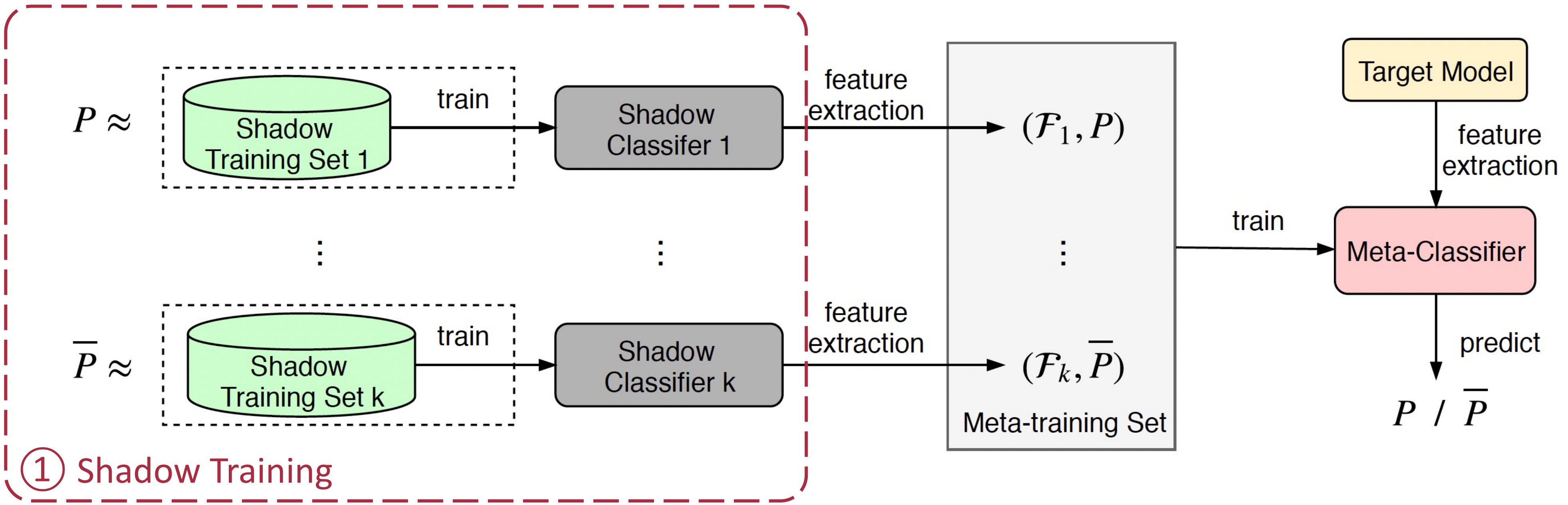
Property Inference Attacks



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.

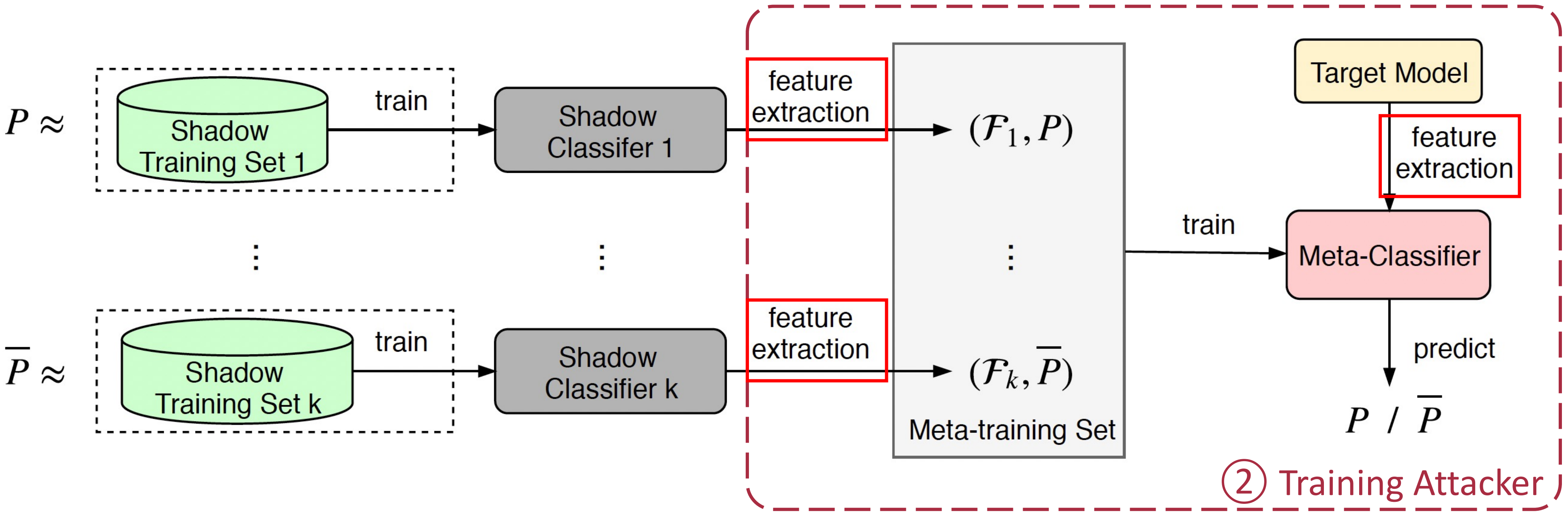
Property Inference Attacks



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

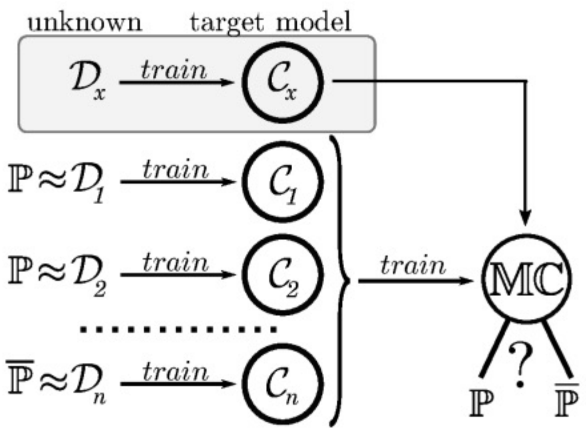


Fig. 1. Attack methodology: the target training set $\mathcal{D}_x$ produced $\mathcal{C}_x$. Using several training sets $\mathcal{D}_1, \dots, \mathcal{D}_n$ with or without a specific property, we build $\mathcal{C}_1, \dots, \mathcal{C}_n$, namely the training set for the meta-classifier $\mathbb{MC}$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

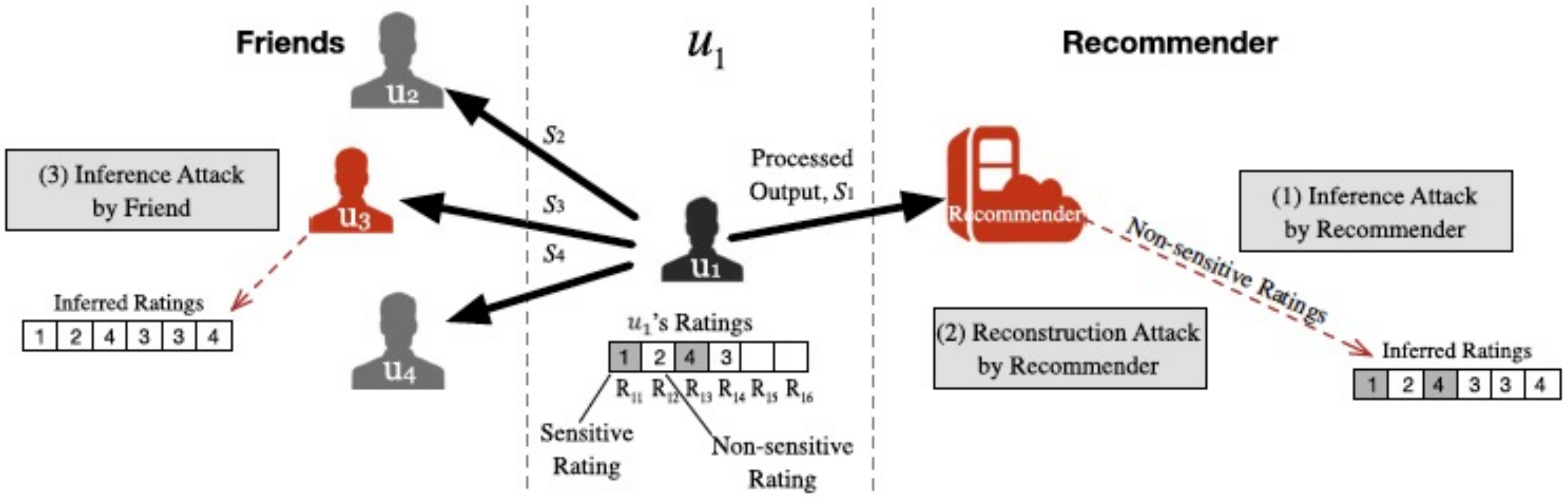
Reconstruction Attacks

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**

    $scores_t = \textsc{ScoreFunction}(N_\Delta[t])$

    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.

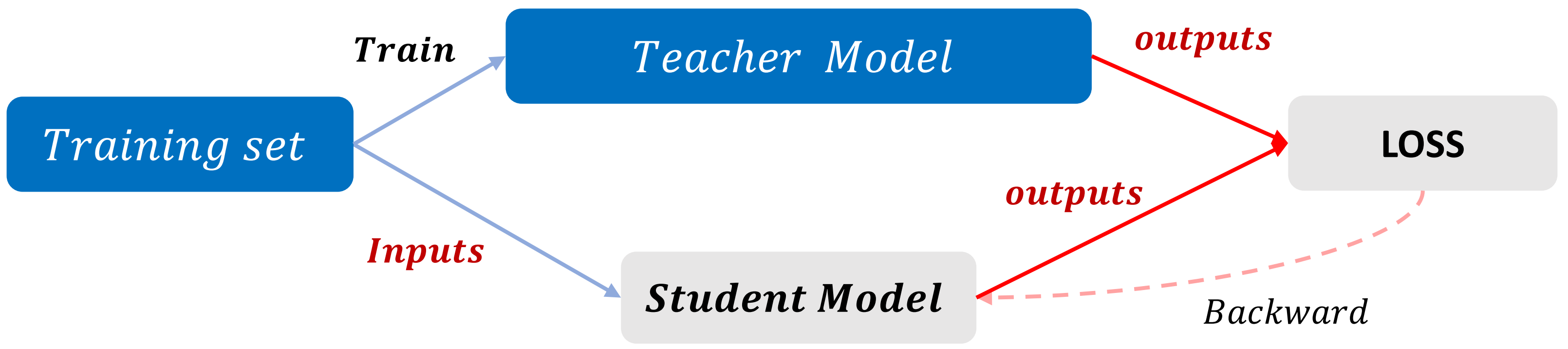
Auxiliary information:

- 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.

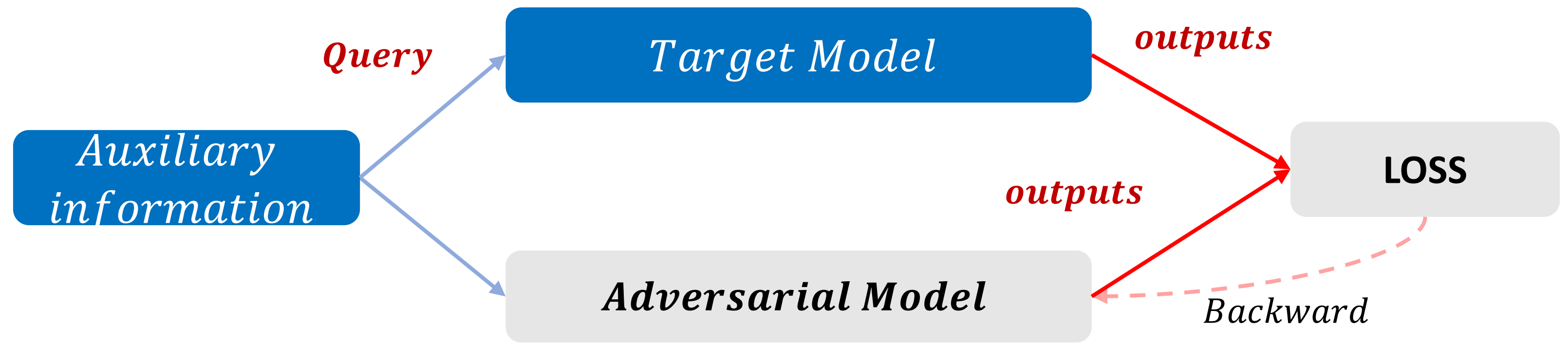
J. A. Calandrino, et al, "You Might Also Like:" Privacy Risks of Collaborative Filtering," 2011 IEEE SP.

Model Extraction Attacks

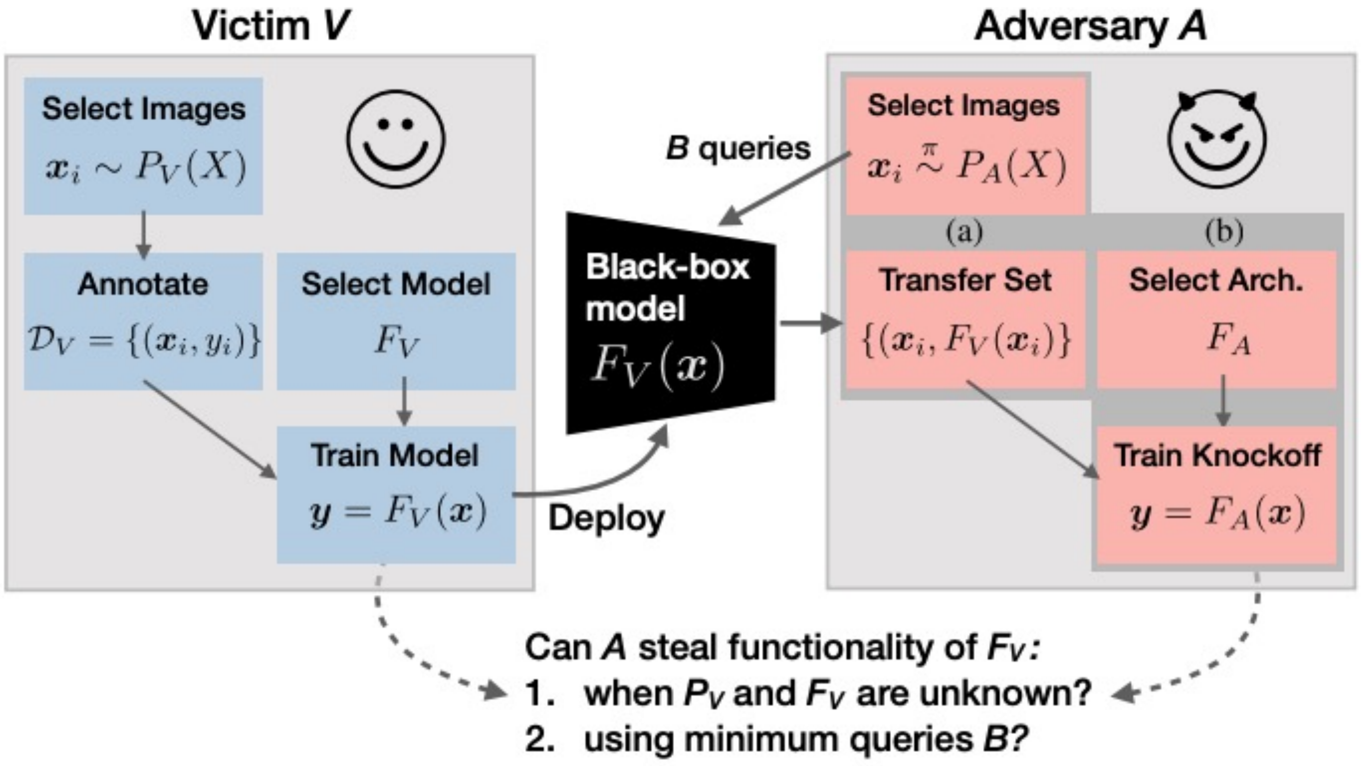
- Knowledge Distillation



- Model Extraction Attacks



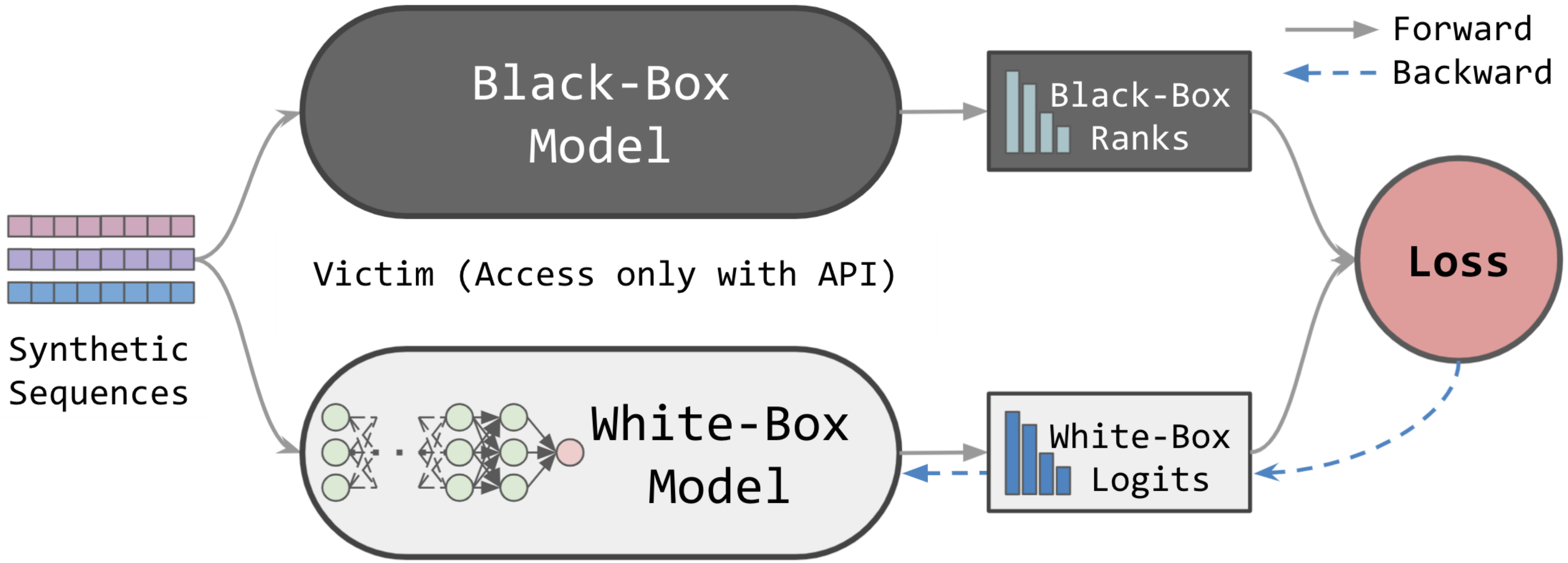
Model Extraction Attacks



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.

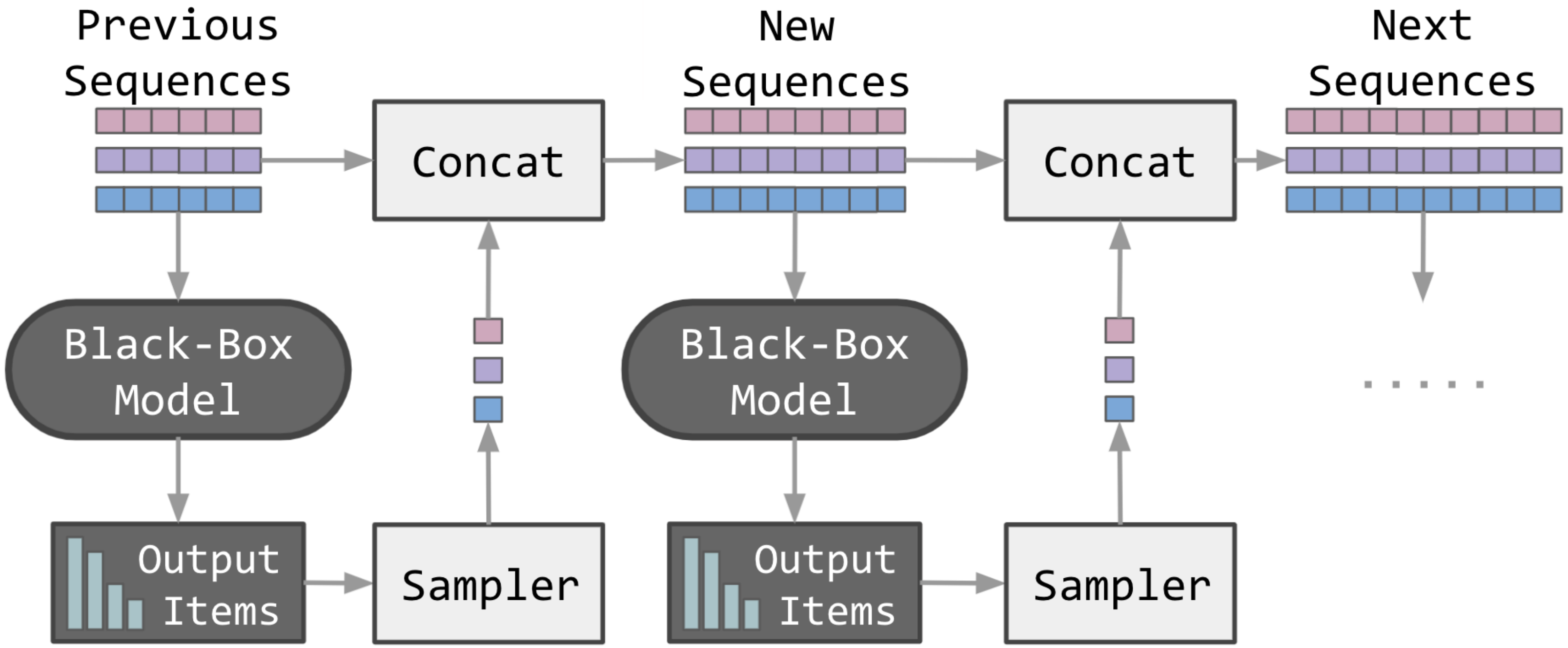
Model Extraction Attacks



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]

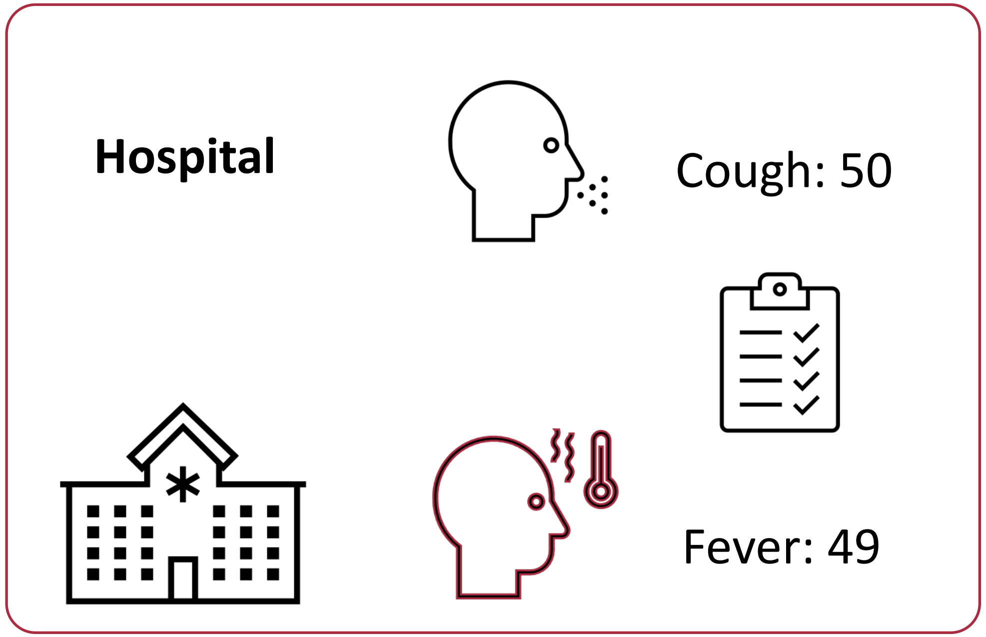
Differential Privacy

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:

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

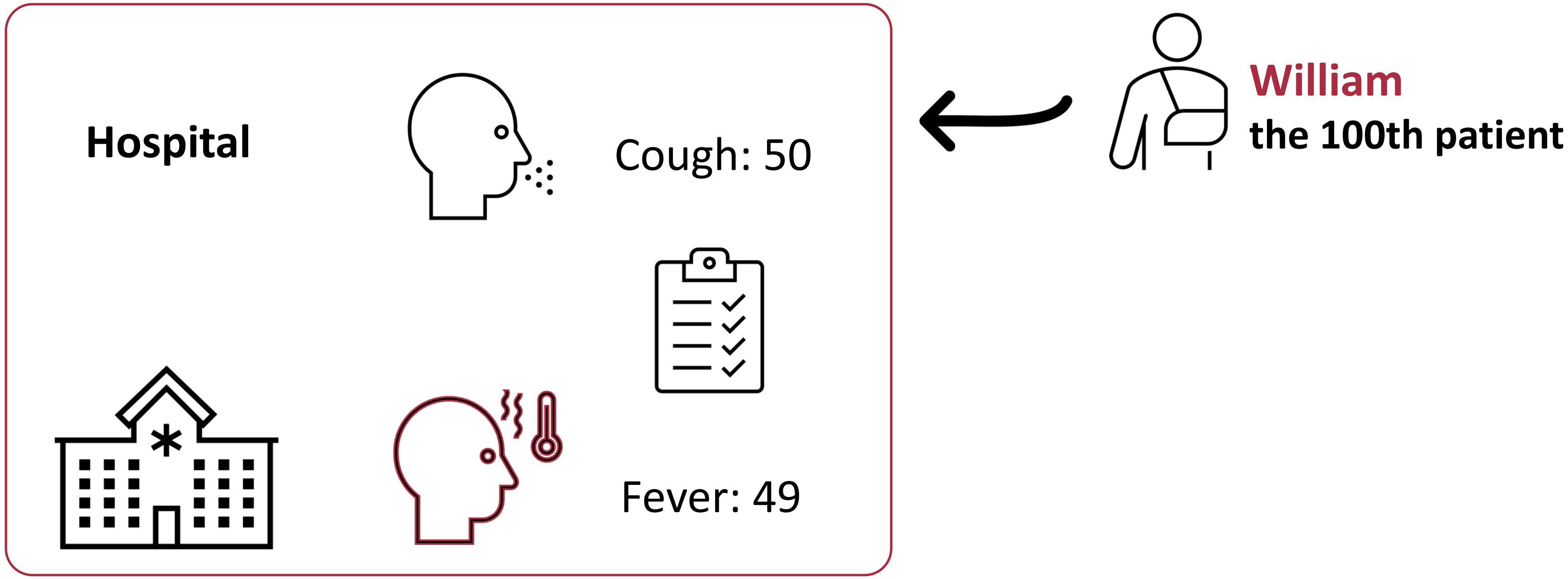
ϵ is the **privacy budget,** the smaller ϵ is, the better the privacy protection is, but more noise is added, and the data utility decreases.

Differential Privacy



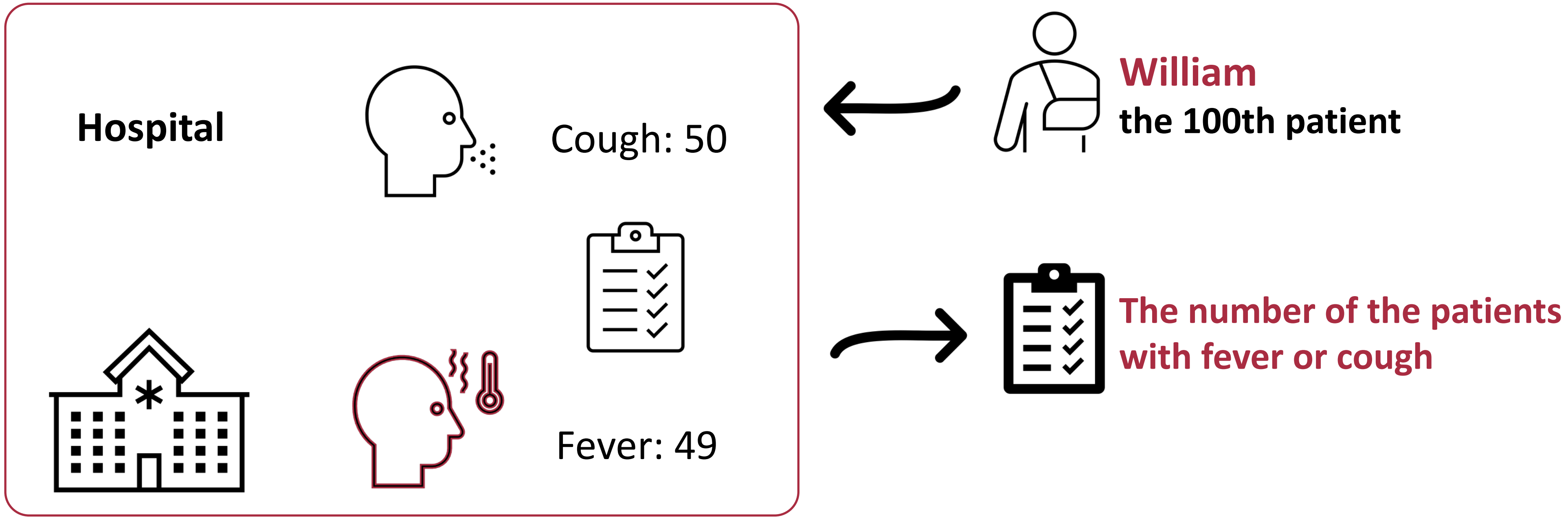
J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy



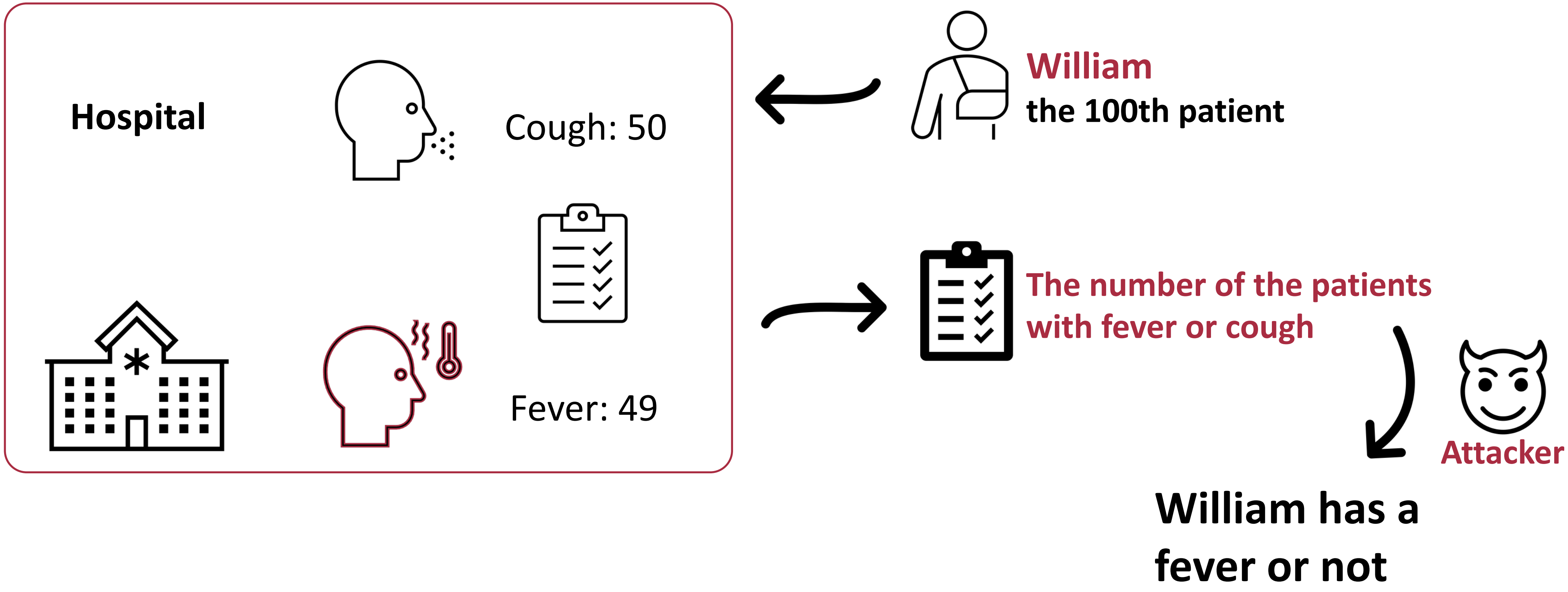
J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

Differential Privacy



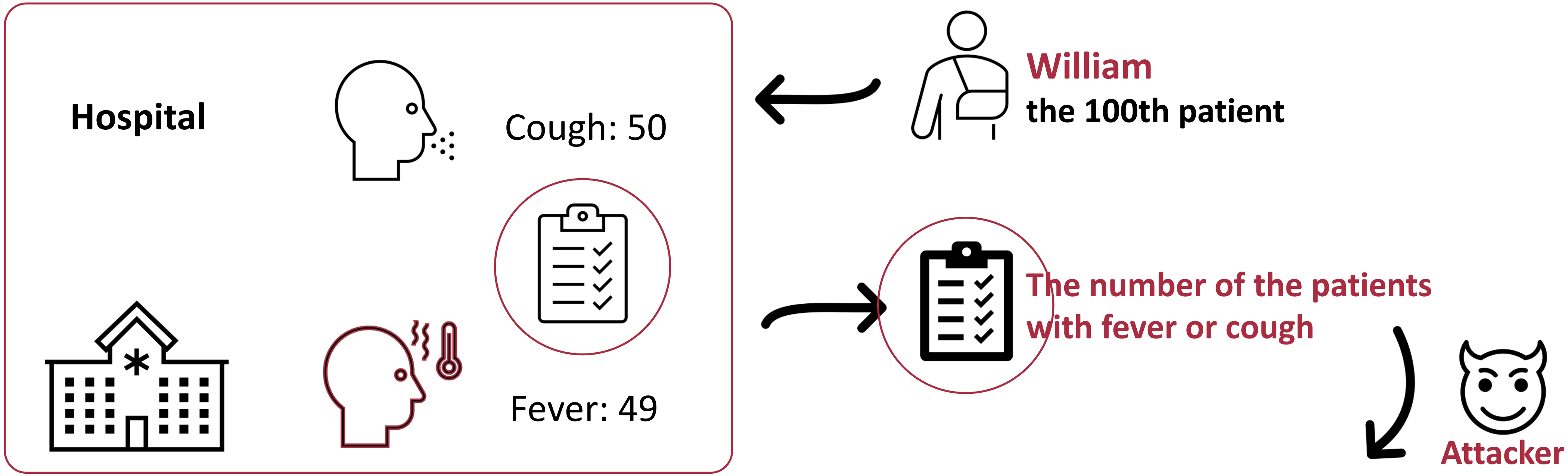
J. Chen, et al. Differential privacy protection against membership inference attack on machine learning for genomic data. the Pacific Symposium, 2021.

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



William has a fever or not

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.

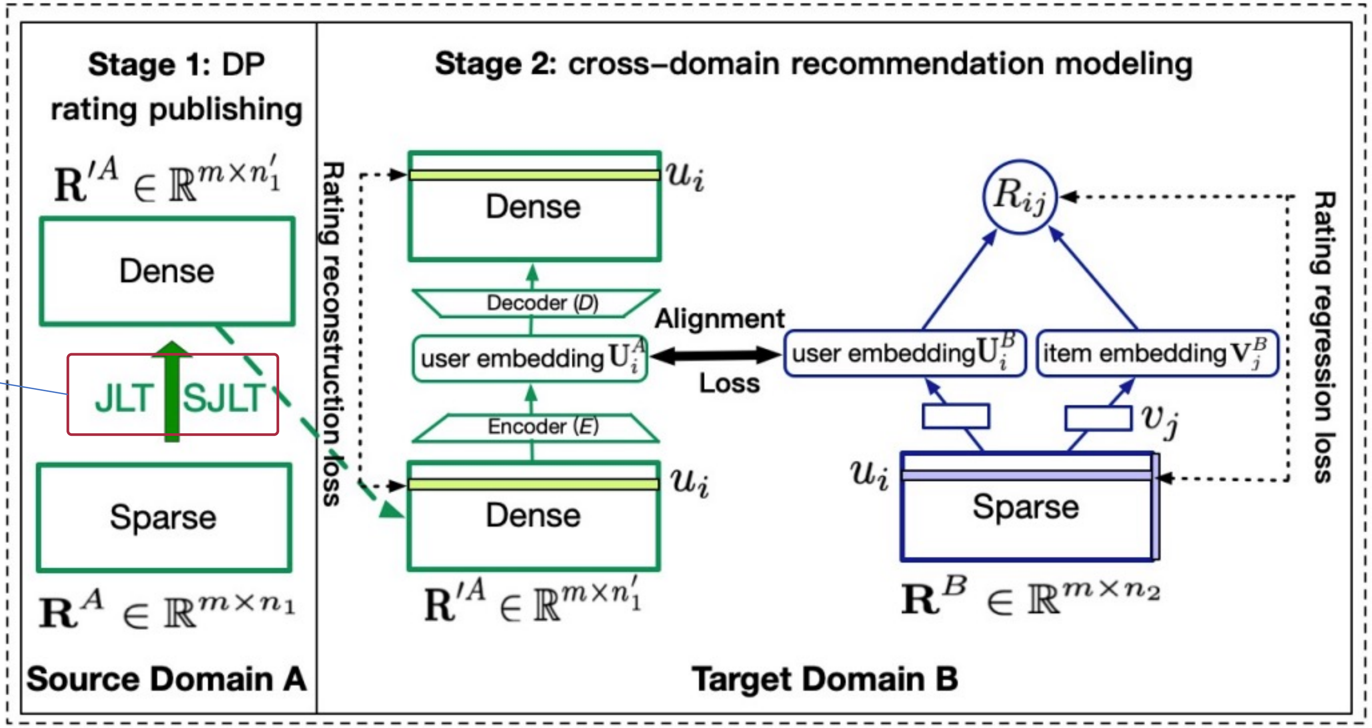


Figure 1: Framework of PriCDR.

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

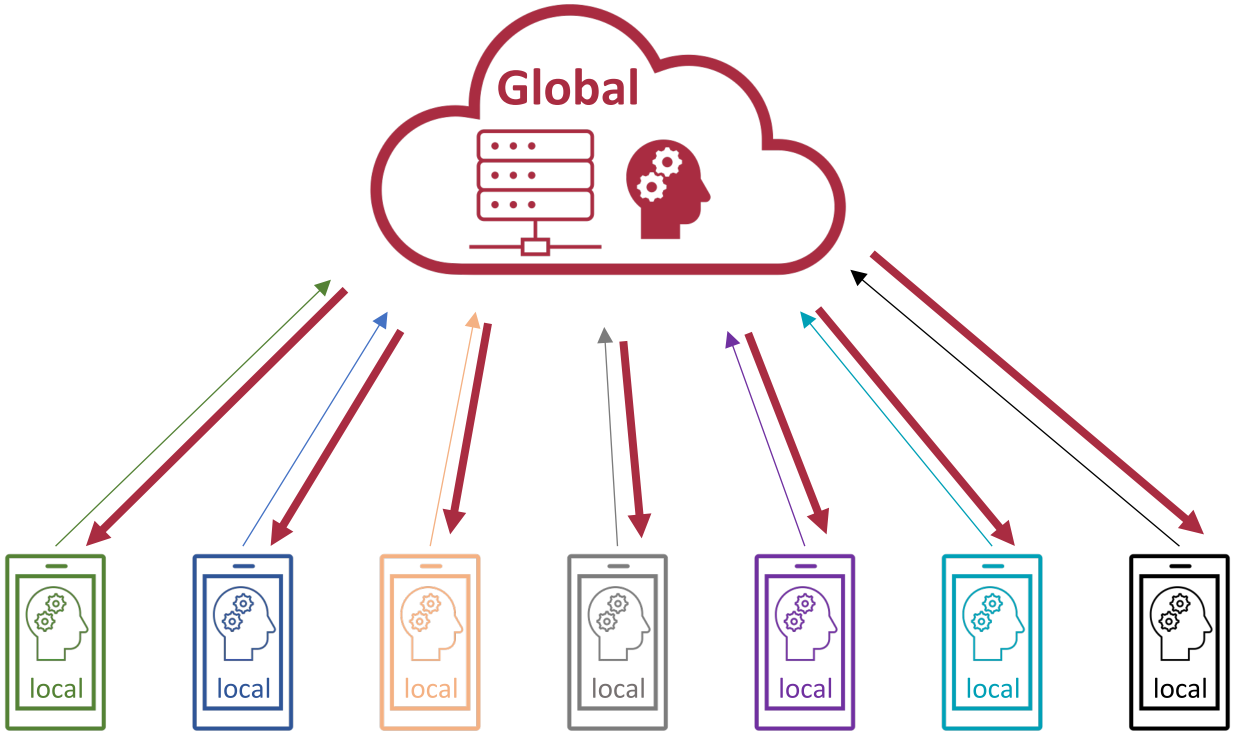
Q. Yang, et al. Federated machine learning: Concept and applications. TIST, 2019.

Federated Learning

Global server with global recommendation model

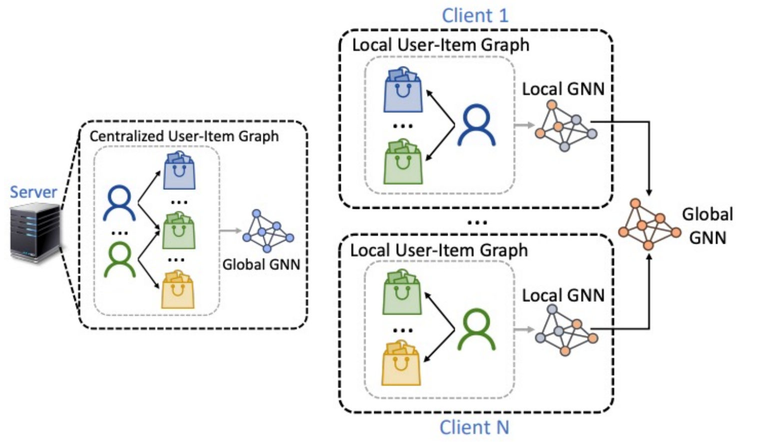
Gradients

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.

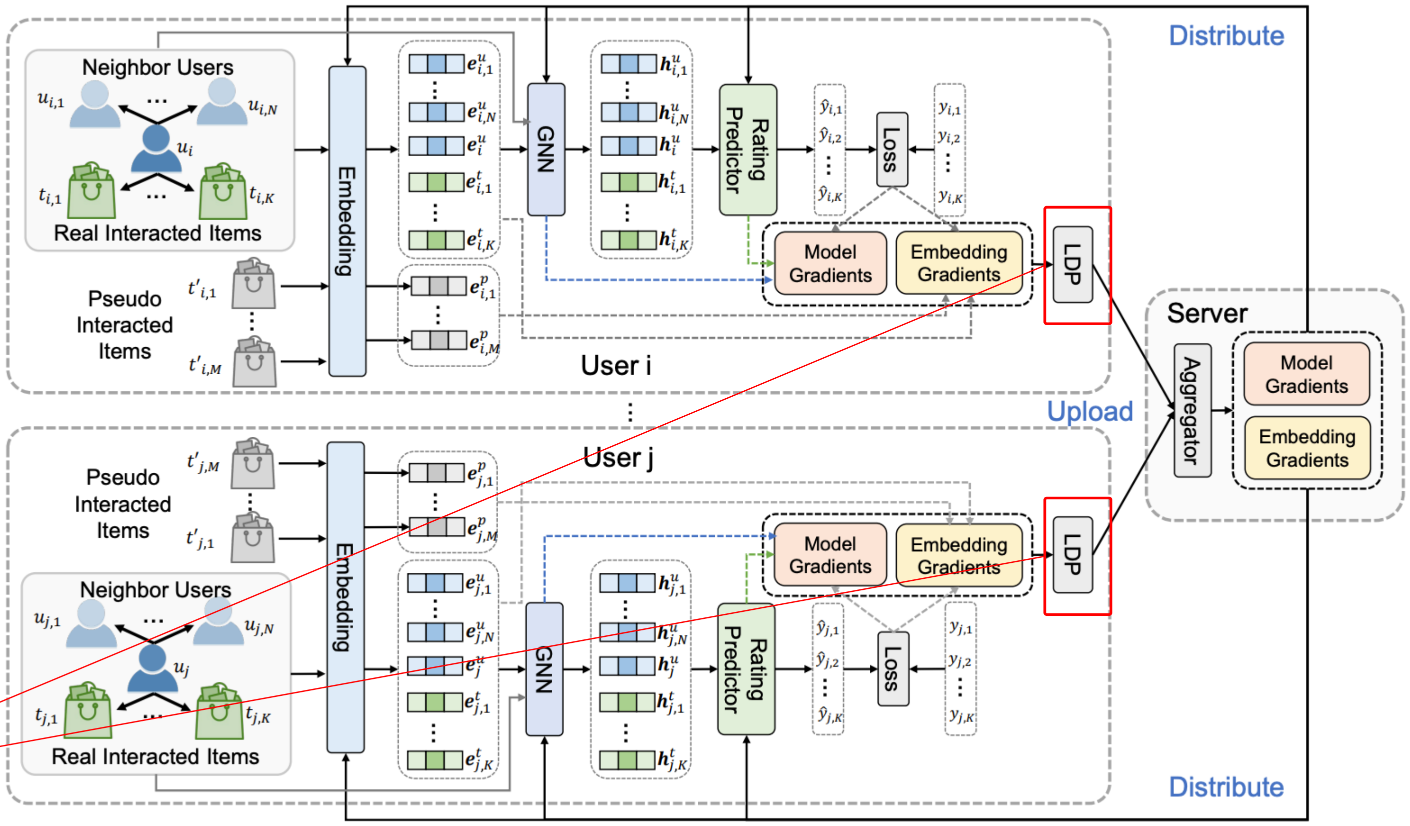
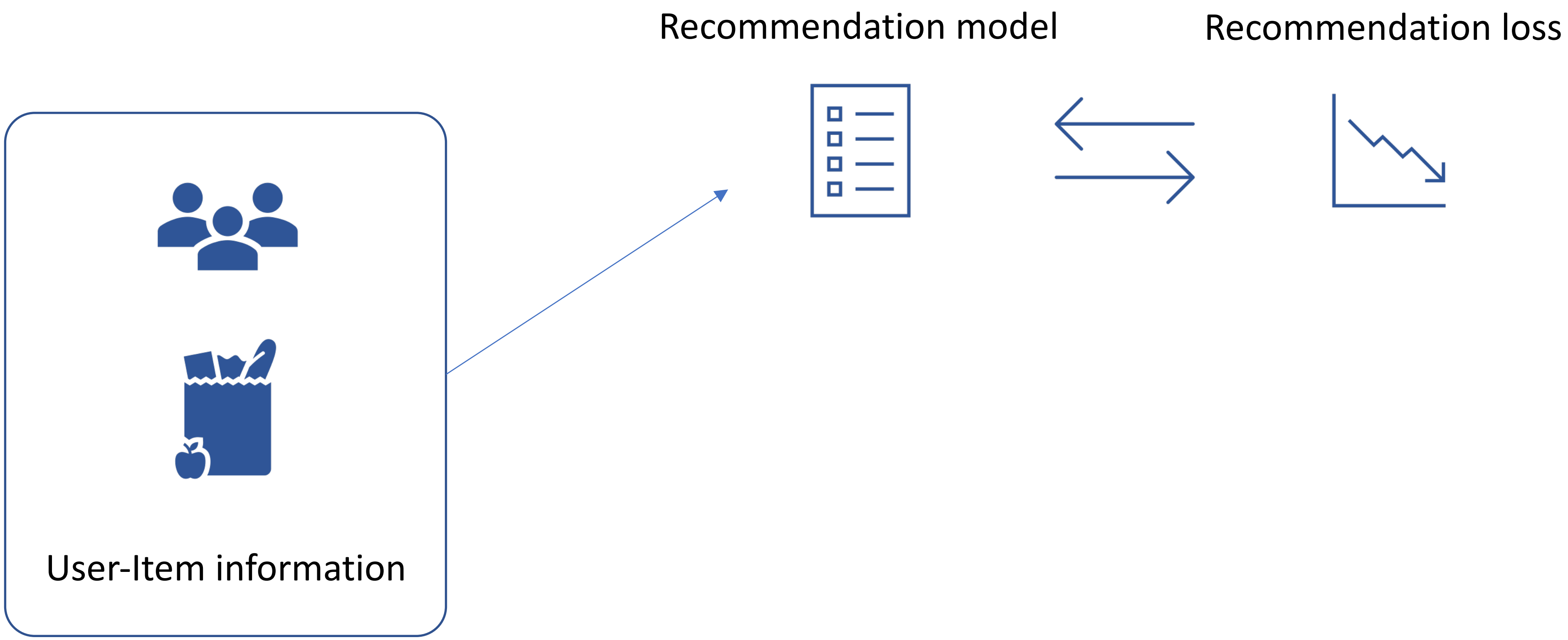


Figure 2: The framework of our *FedGNN* approach.

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

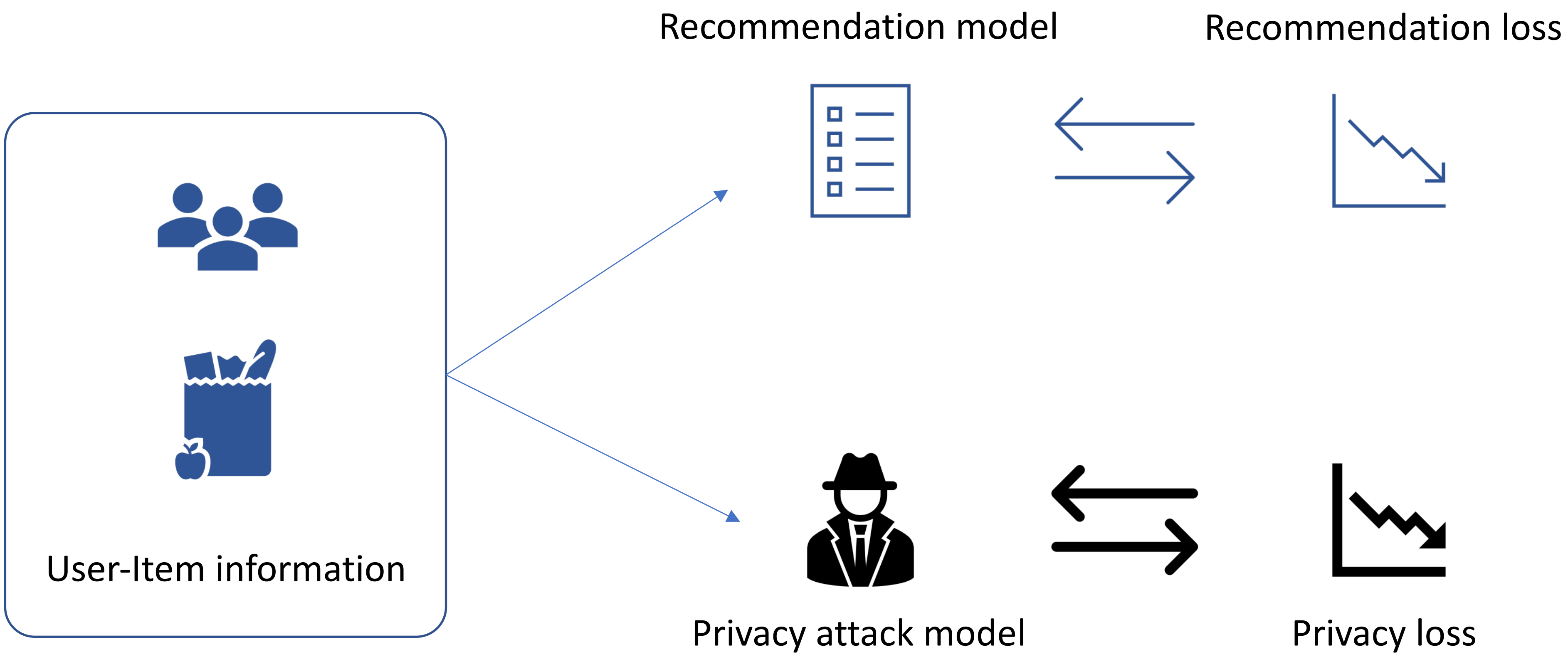
Wu C, et al. Fedgnn: Federated graph neural network for privacy-preserving recommendation. arXiv, 2021.

Adversarial Learning



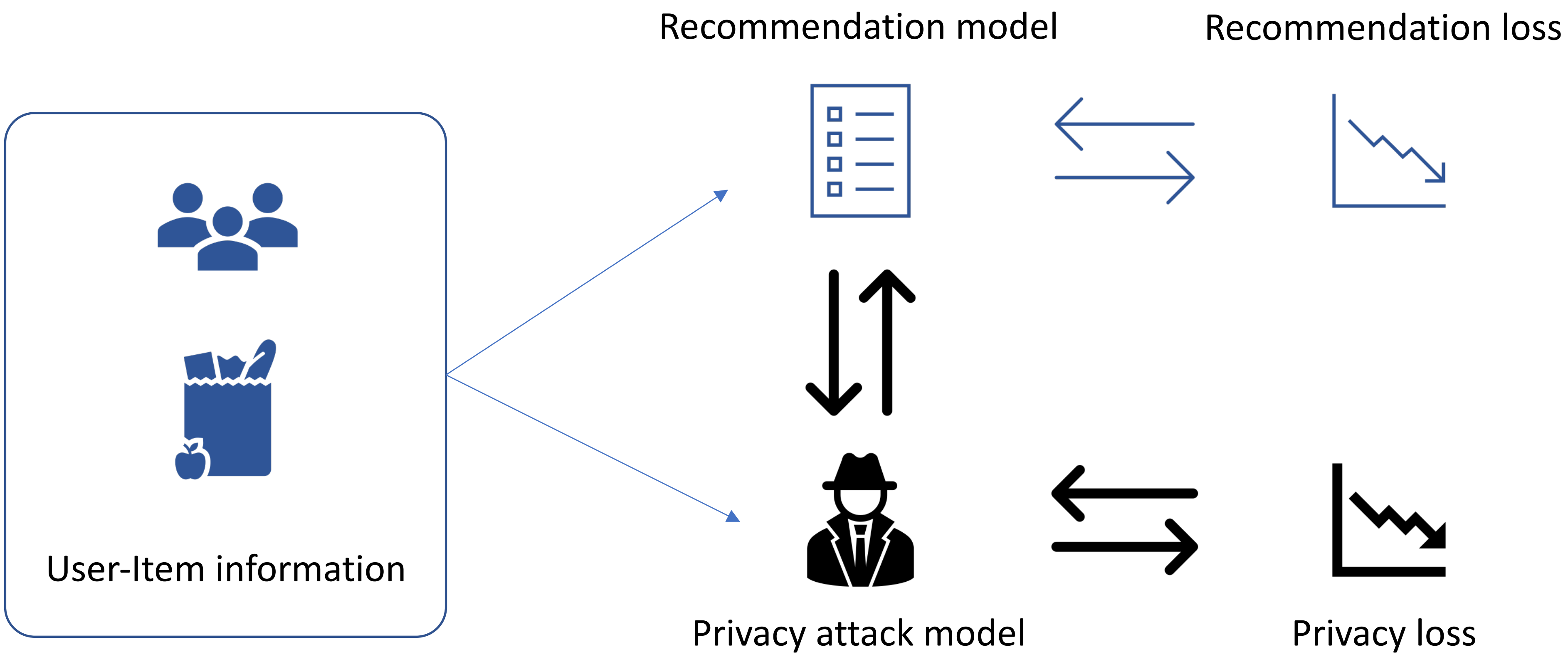
L. Huang, et al. Adversarial machine learning. the 4th ACM workshop on Security and artificial intelligence, 2011.

Adversarial Learning



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

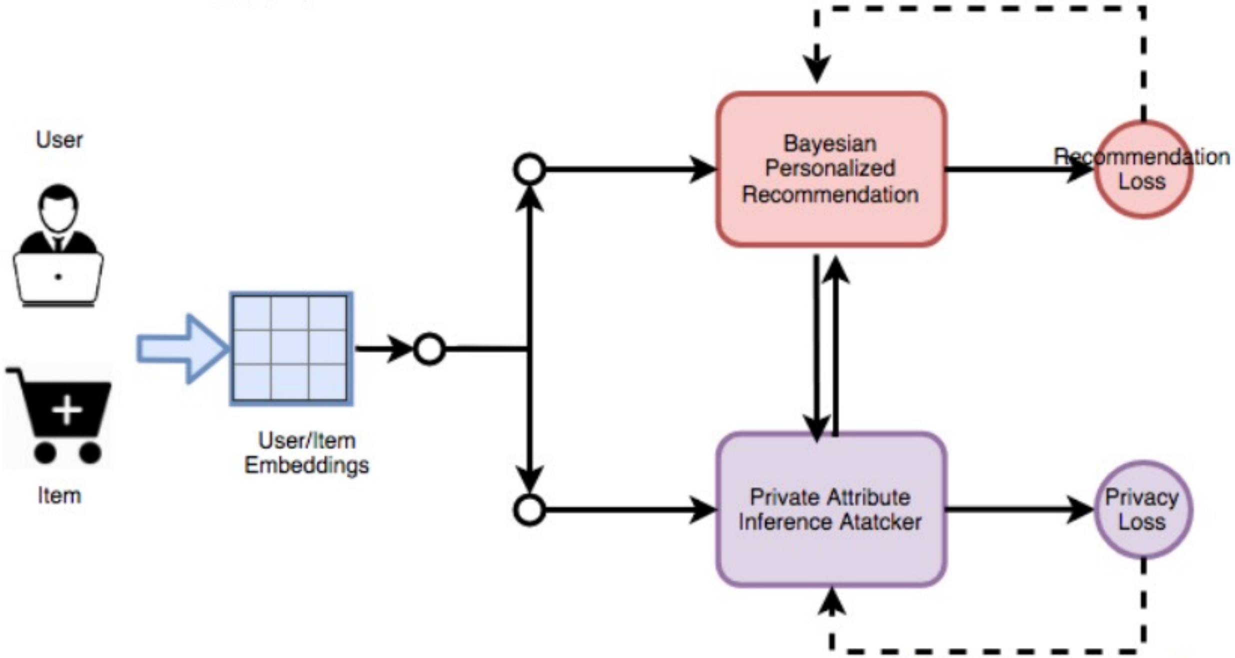
Adversarial Learning



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

Adversarial Learning

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



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

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

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)

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

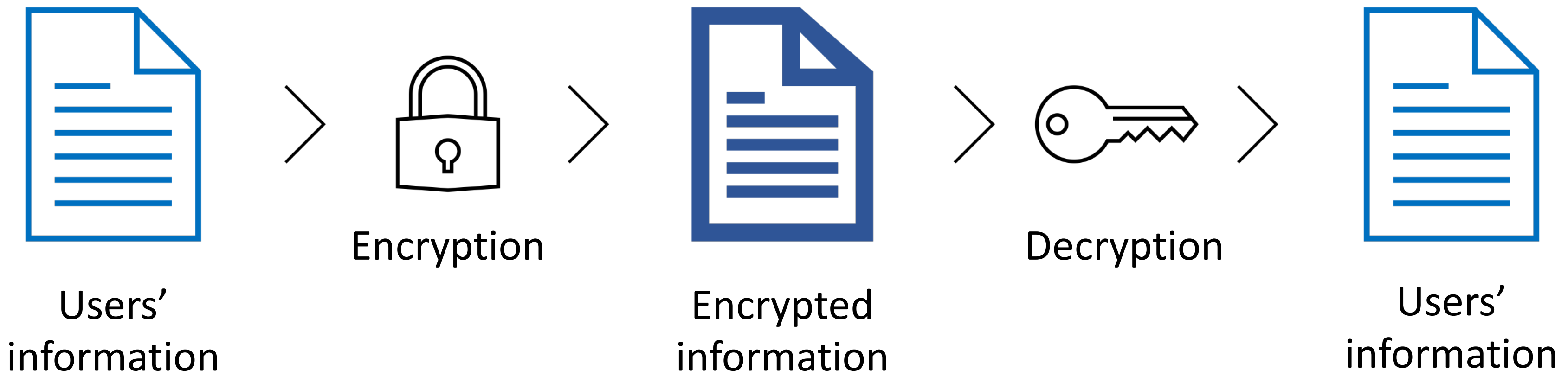
▪ denotes a suppressed value.

Sensitive attributes

l-Diversity (l=2)

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.

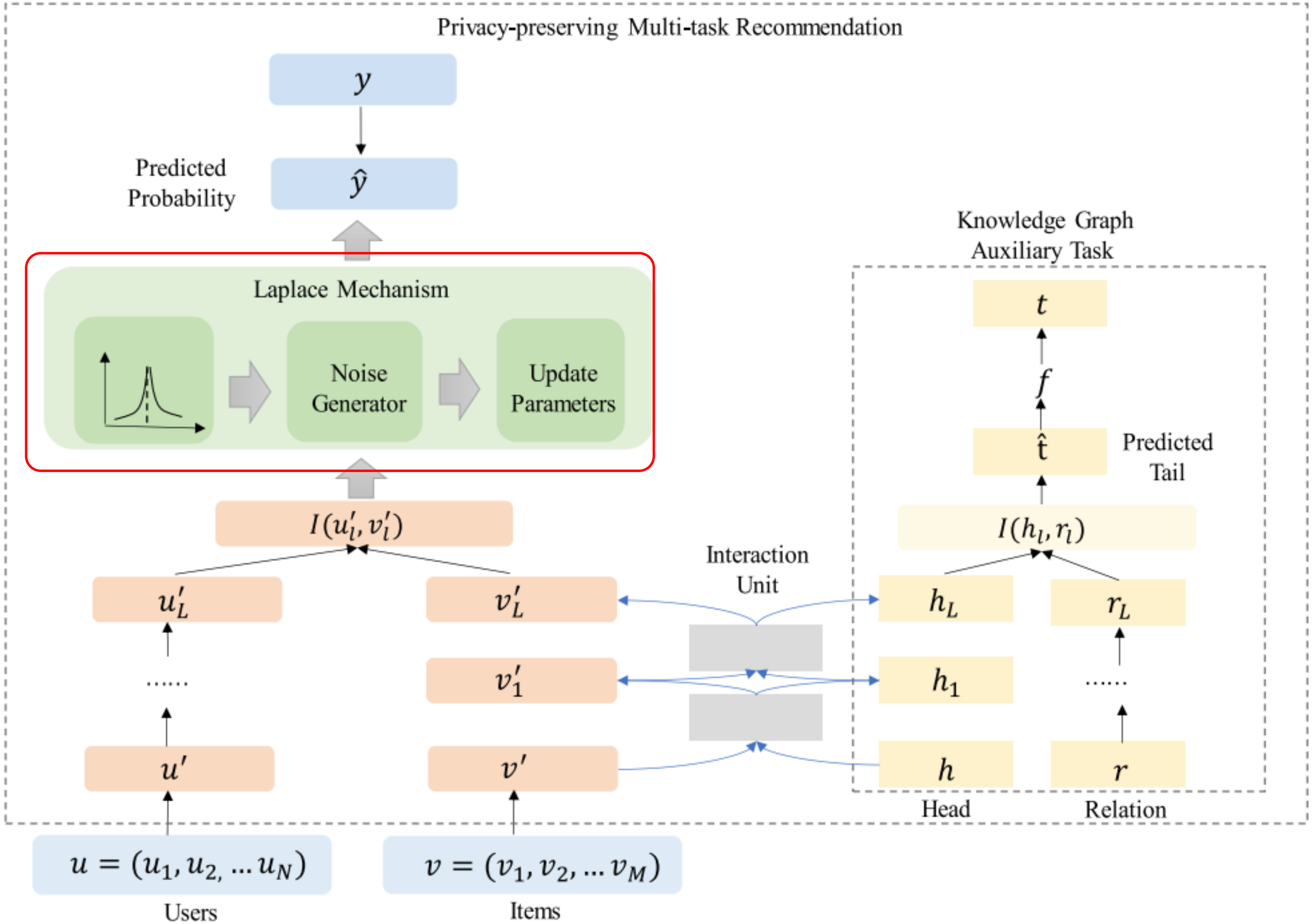


FIGURE 1. A privacy-preserving multi-task framework for knowledge graph enhanced recommendation.

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

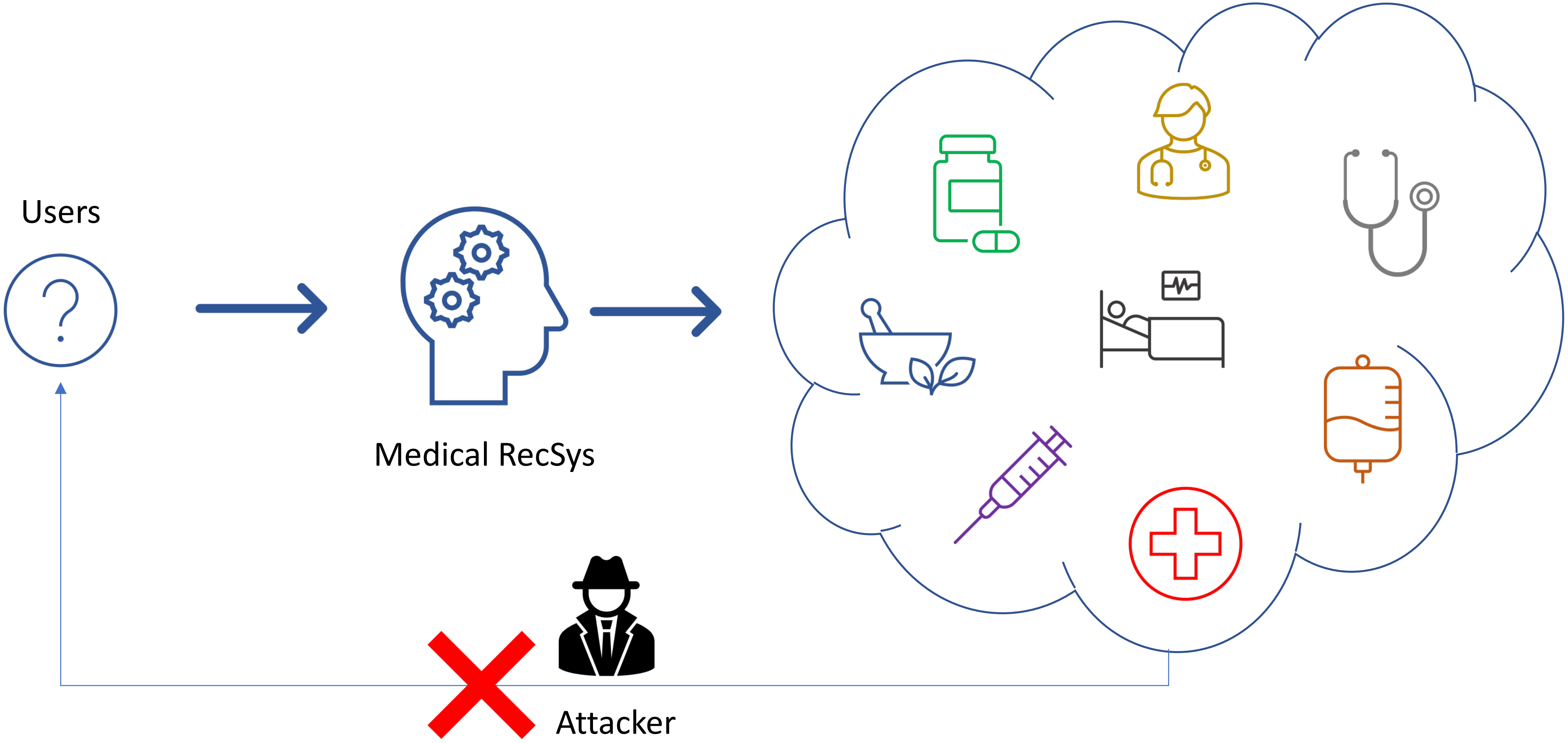
Summary of Privacy Preserving

- **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

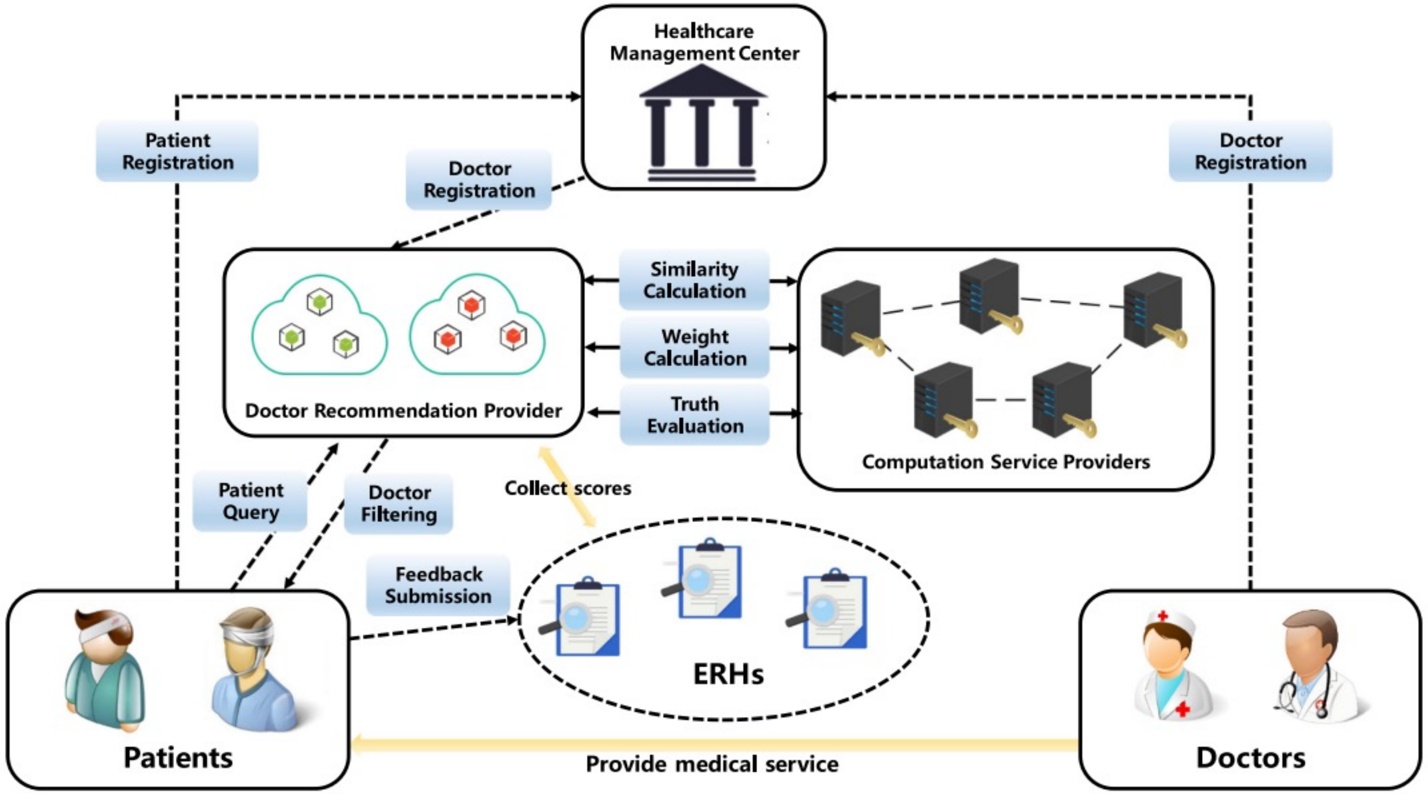
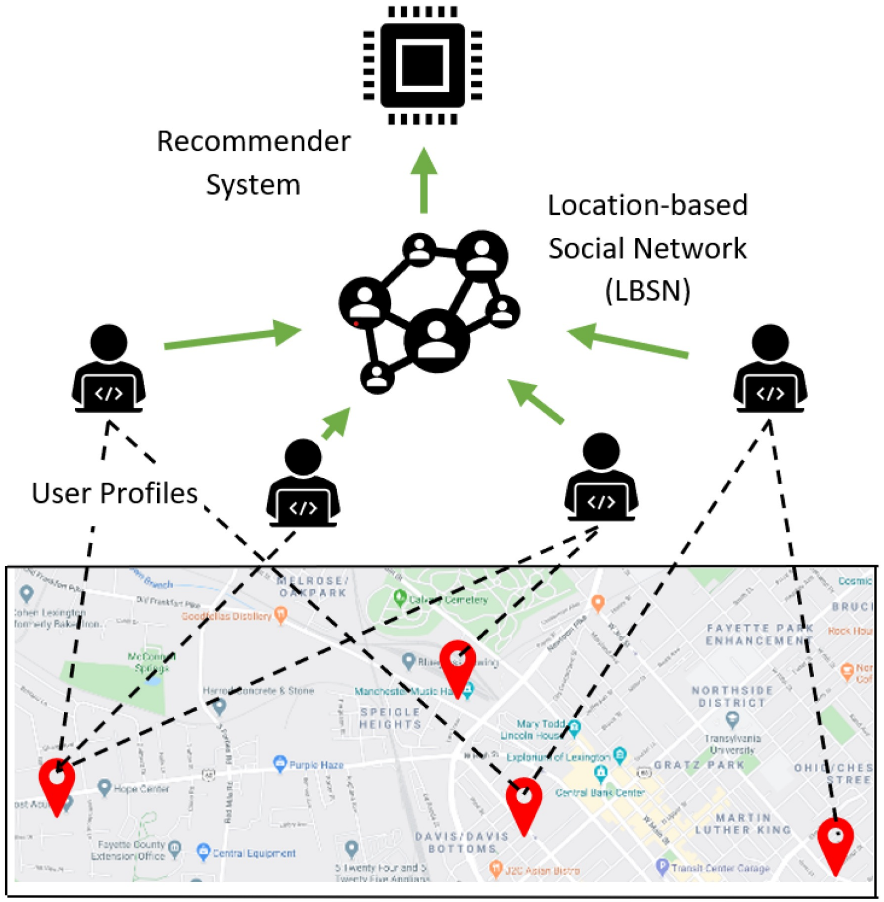


Fig. 1. System model.

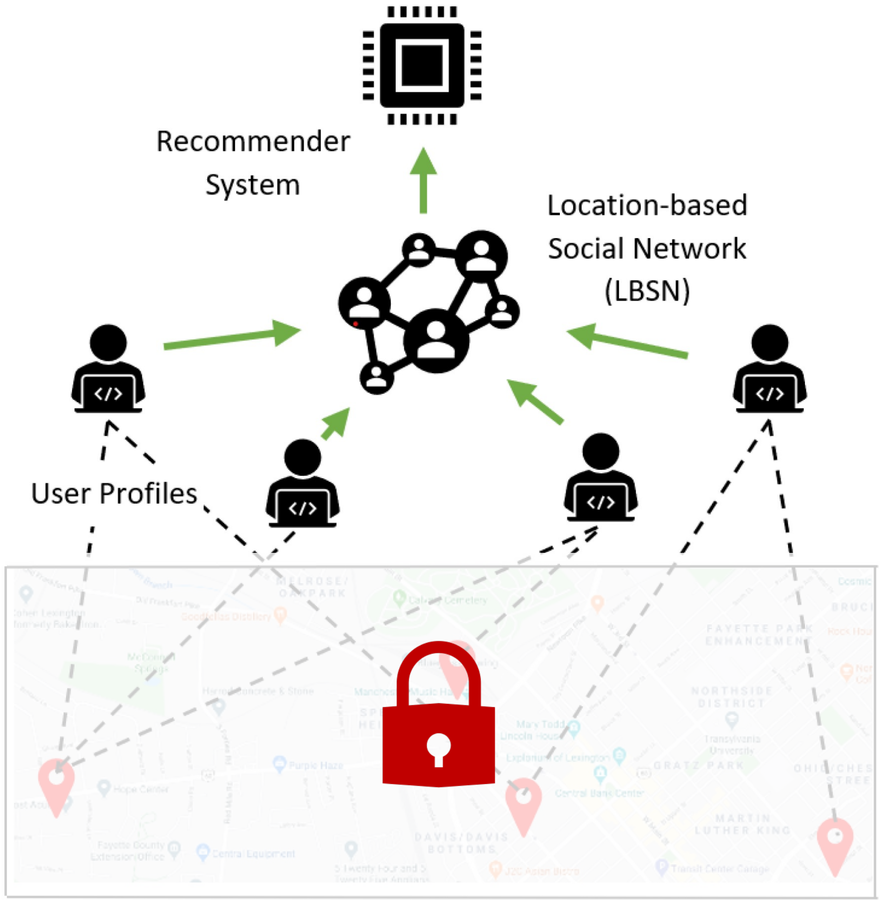
Cong Peng, et al. 2021. EPRT: An Efficient Privacy-Preserving Medical Service Recommendation and Trust Discovery Scheme for eHealth System. ACM Trans. Internet Technol. 2021.

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

Federated learning

- TFF
- FATE
- FedML
- LEAF

Homomorphic Encryption

- Awesome HE
- TF Encrypted

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

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

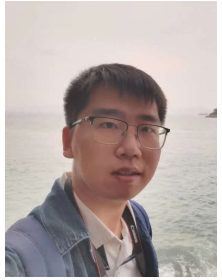
Jingtong Gao

Privacy

Lin Wang

Environmental Well-being

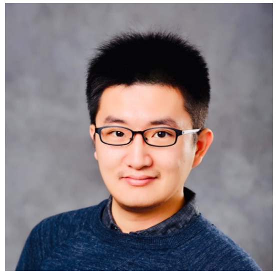
Accountability & Auditability



Qidong Liu

Dimension Interactions

Future Directions

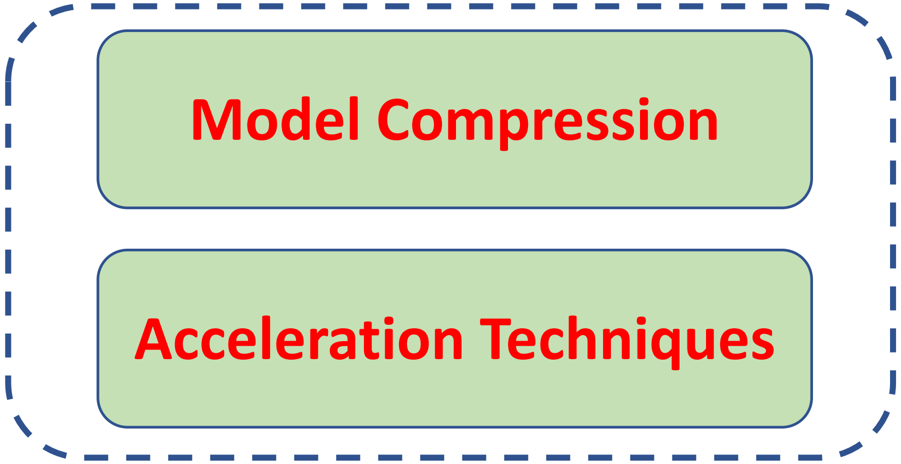


Xiangyu Zhao

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.

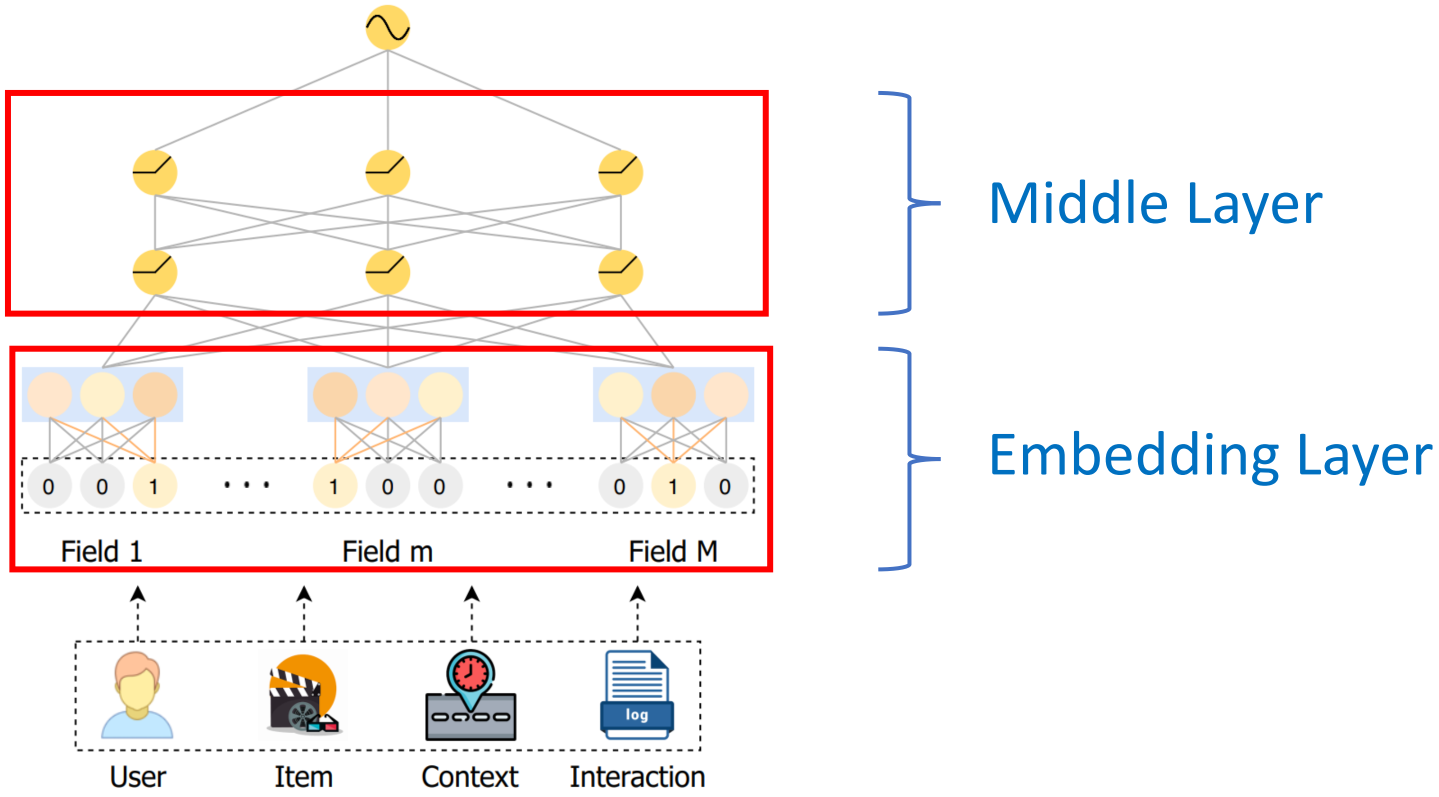


Model Compression

- 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

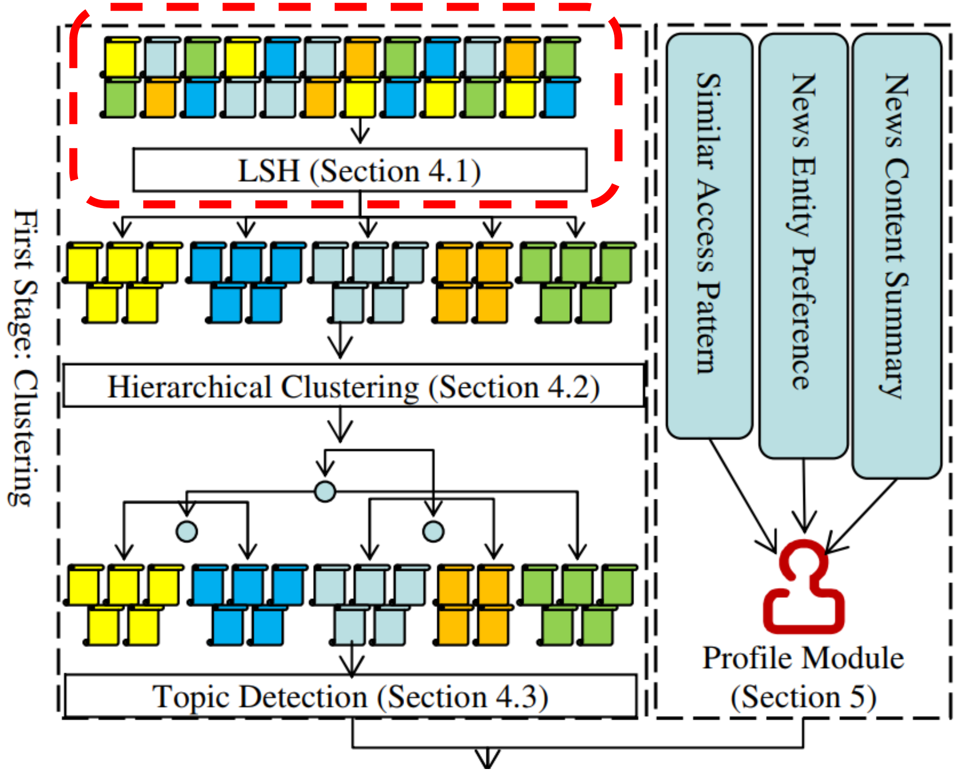
$$x \in \{0,1\}^n \xrightarrow{h(\cdot)} y \in \{0,1\}^m$$

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.

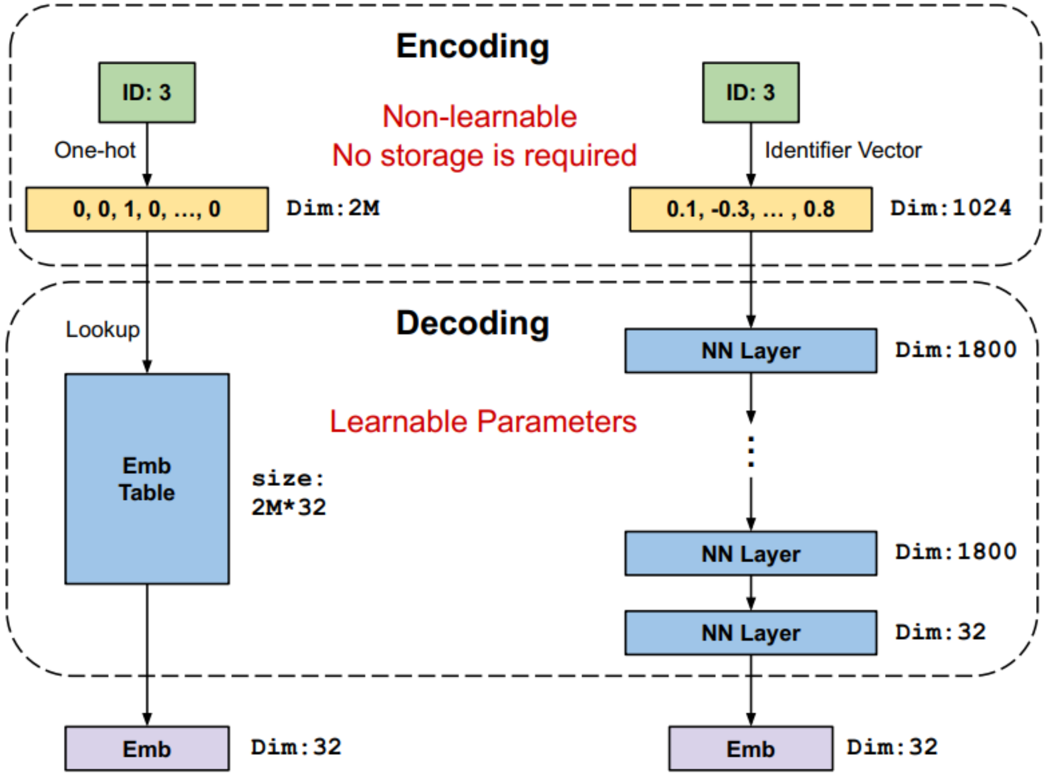


SCENE : A Scalable Two-Stage Personalized News Recommendation System, SIGIR, 2011

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

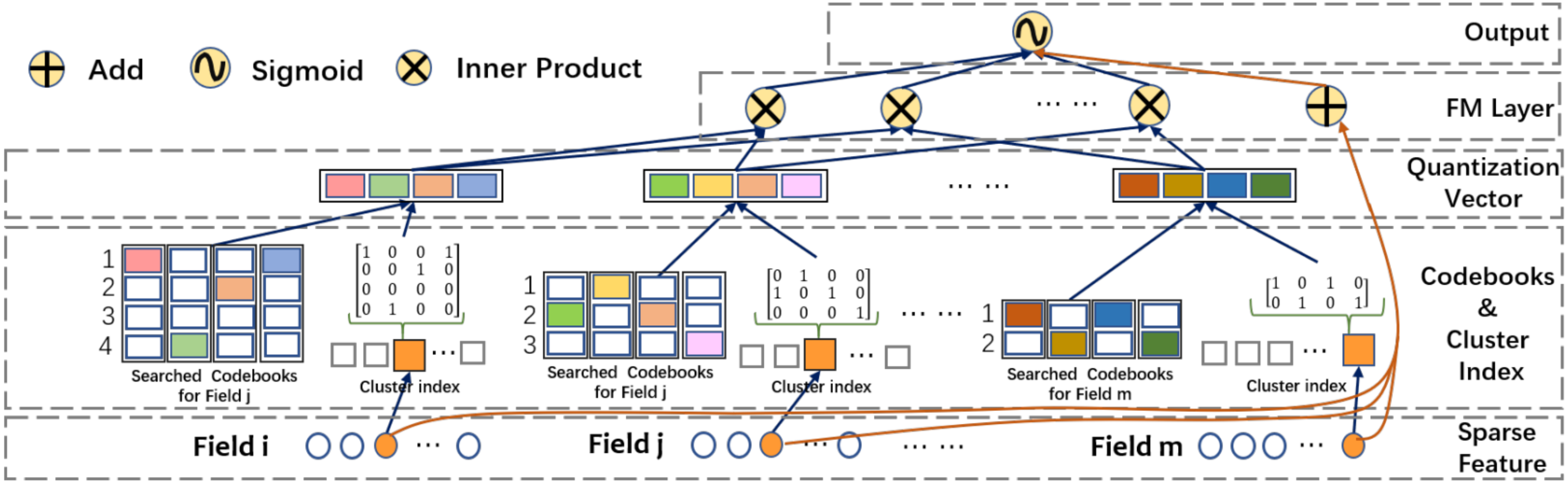
- Model Compression
  - Hash
  - Quantization
    - Product Quantization
    - Additive Quantization
    - Compositional Quantization
  - Knowledge Distillation
  - Neural Architecture Search
  - Others

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

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:

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

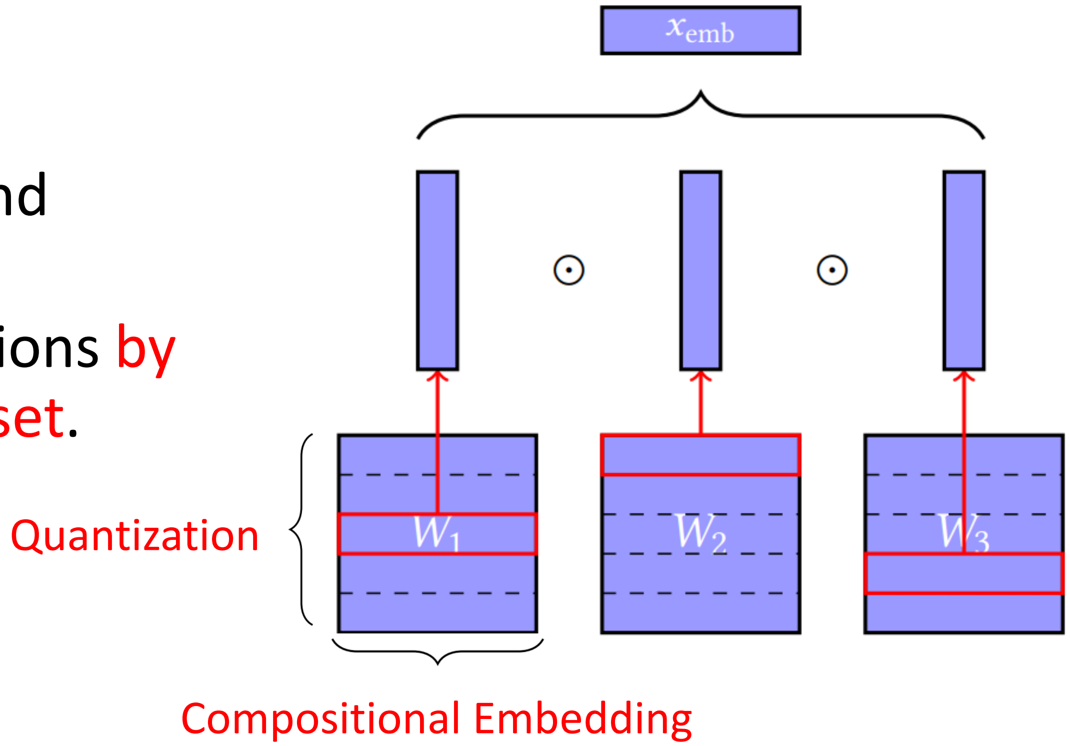
The objective function:

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

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

Quantization

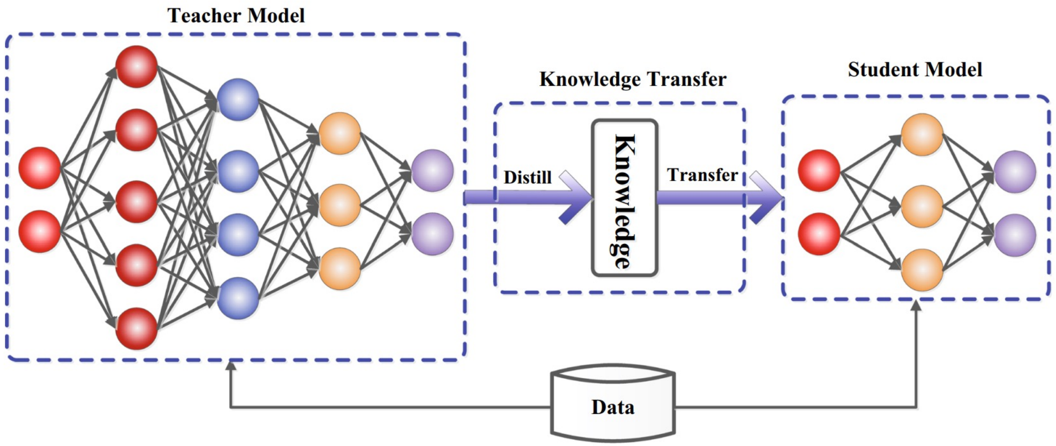
- **Compositional Embedding**
  - The main idea of compositional embedding is to generate meta embedding for each feature based on their characteristics.
- **Compositional Embeddings** – KDD'20
  - Reduce the embedding size in an end-to-end scheme.
  - Split the embedding table into several sections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

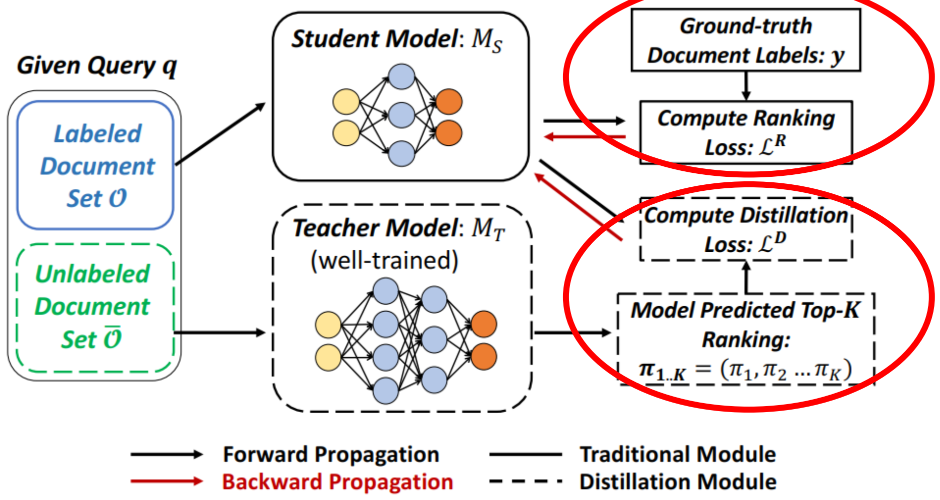
Knowledge Distillation: A Survey, IJCV, 2021

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

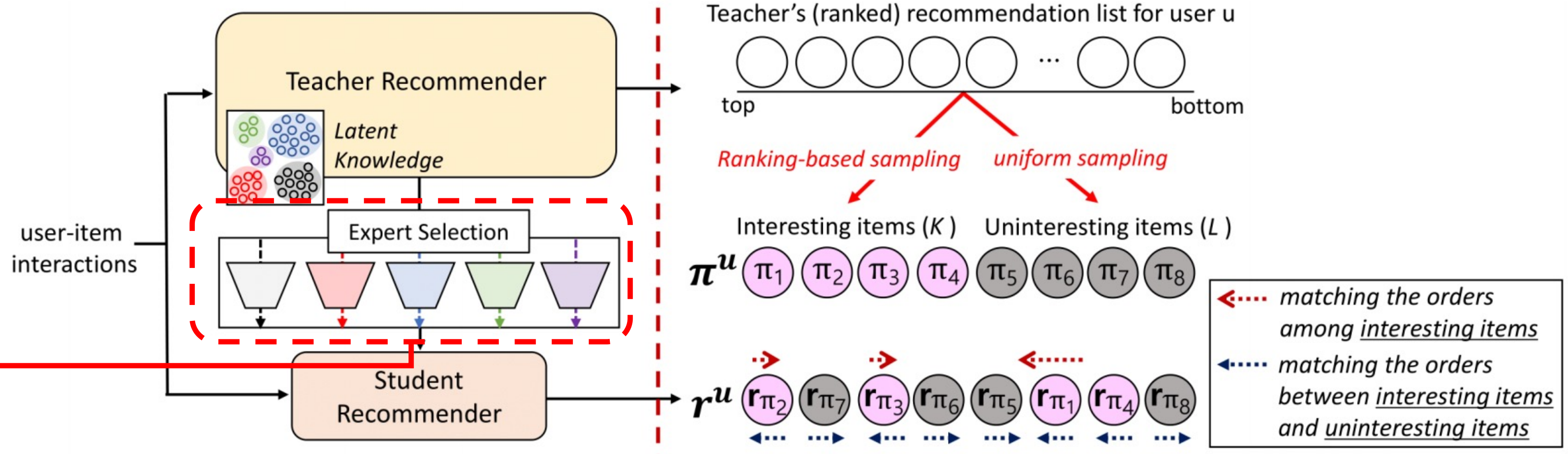
Knowledge Distillation

- **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

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

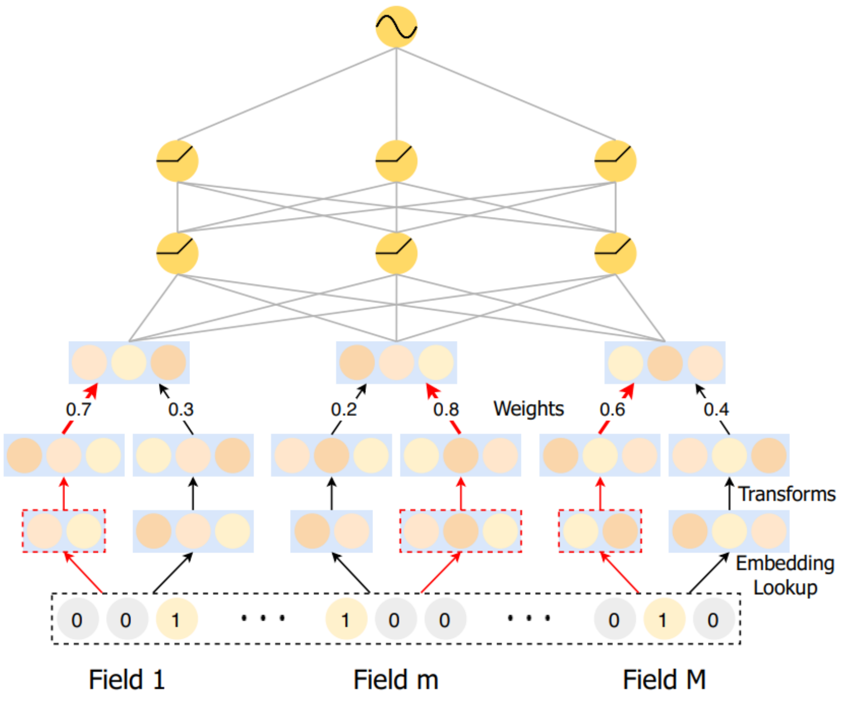
$$s.t.\ \mathcal{W}^*(\mathcal{A}) = arg\min_{\mathcal{W}} \mathcal{L}_{train}(\mathcal{W}, \mathcal{A})$$

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.

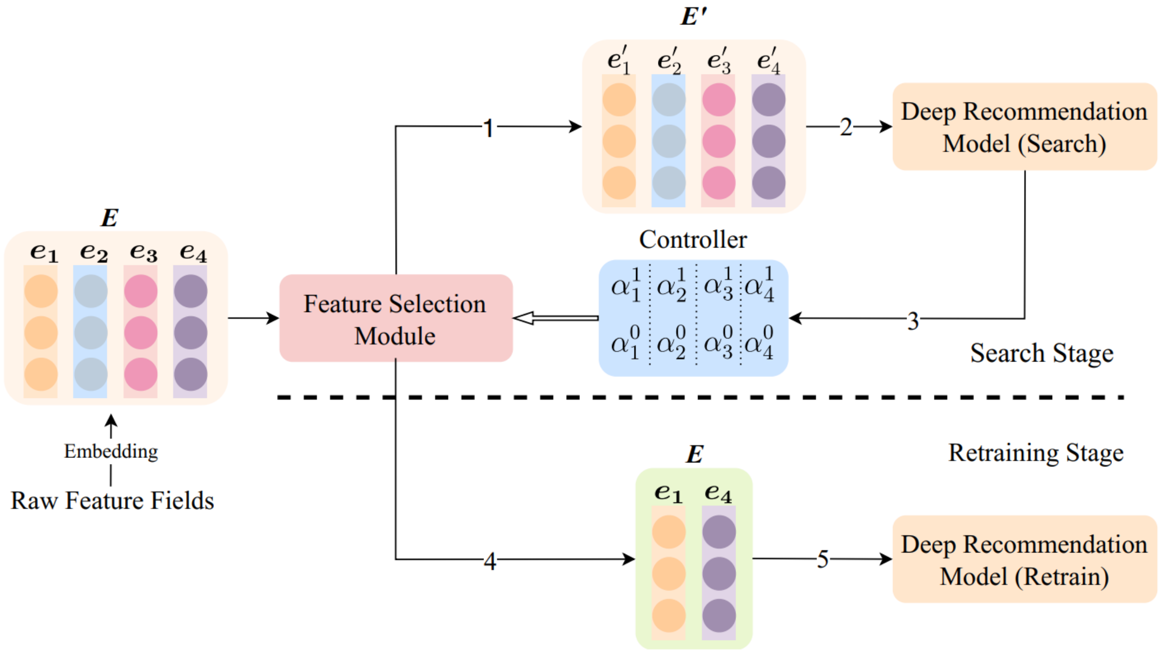


AutoDim: Field-aware Embedding Dimension Search in Recommender Systems, WWW, 2021

Neural Architecture Search

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

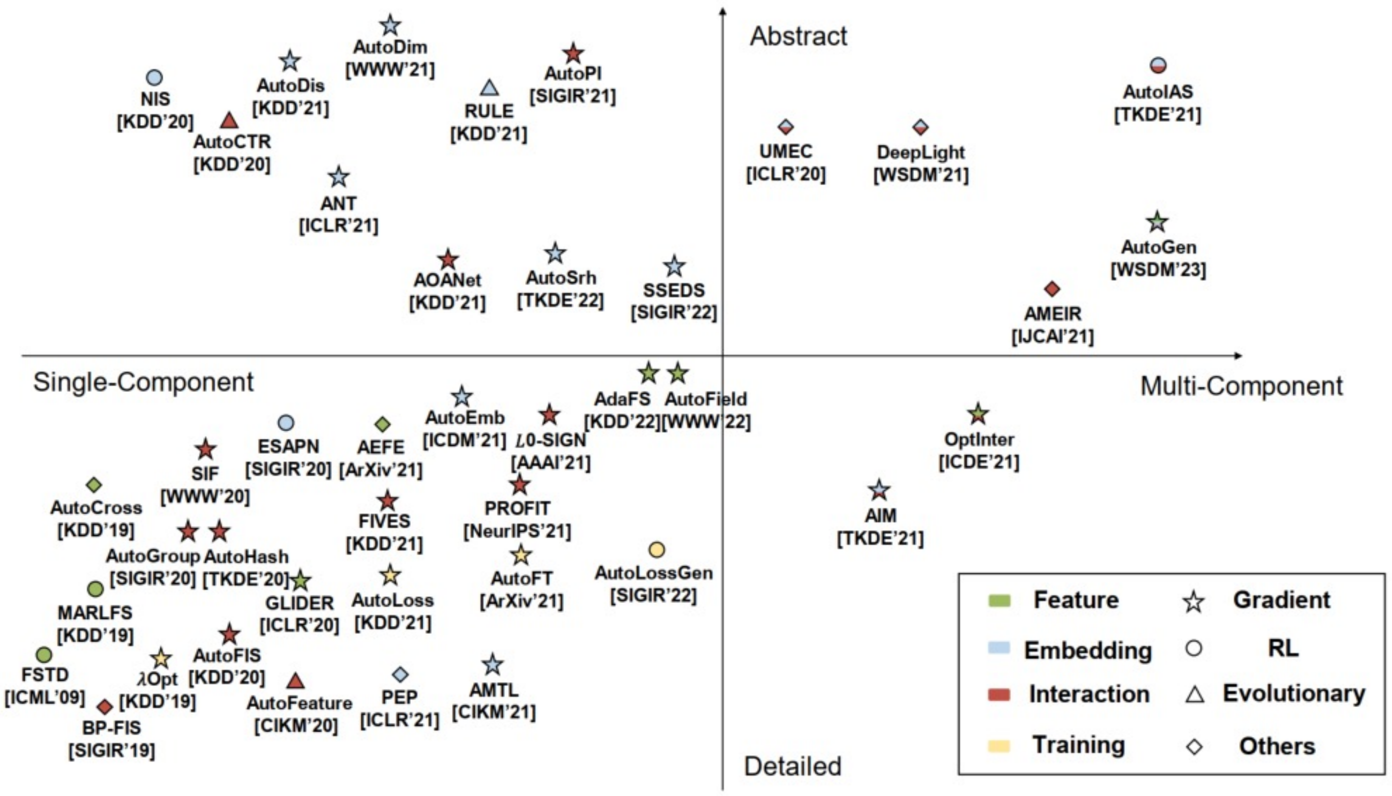
- **AutoField** – WWW'22
  - Equips with a controlling architecture to calculate the drop and select probability of each feature field.
  - Retrain the RS model according to the drop and select probability.



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

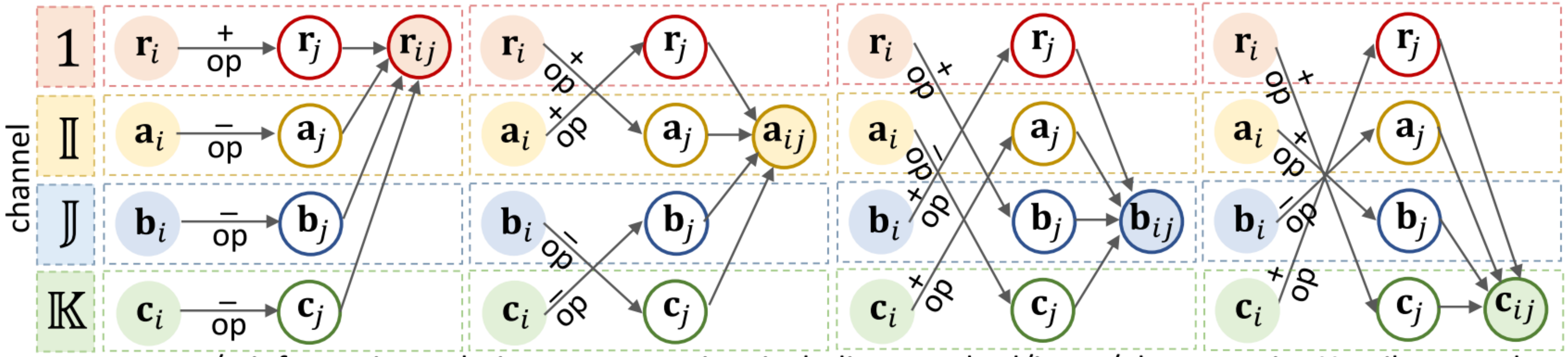
Model Compression

- 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.

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



+/-: 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.

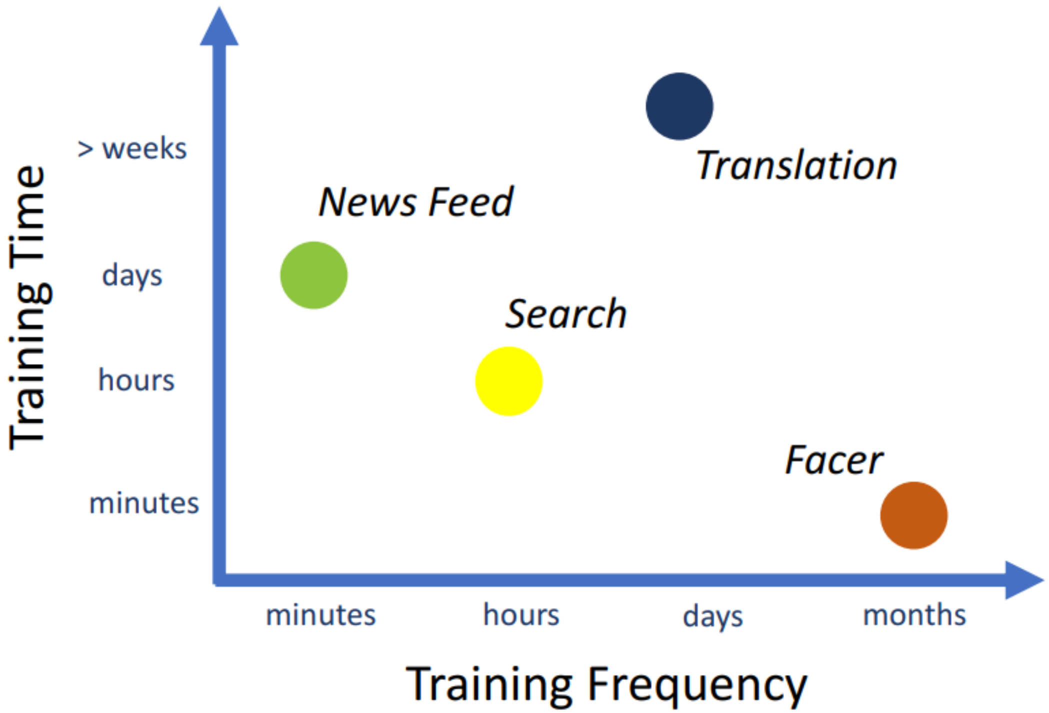
	Embedding Layer	**Middle Layer**
Hash	[80, 209, 307, 438, 456], [184, 227, 313, 355, 422]	[307, 355]
Quantization	[173, 226, 228, 234, 385, 394], [56, 142, 222, 241, 312, 354, 428]	[222, 354, 385]
Knowledge Distillation	[60, 182, 203, 342, 358], [52, 183, 194, 388, 457]	[60, 182, 203, 342, 358], [52, 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



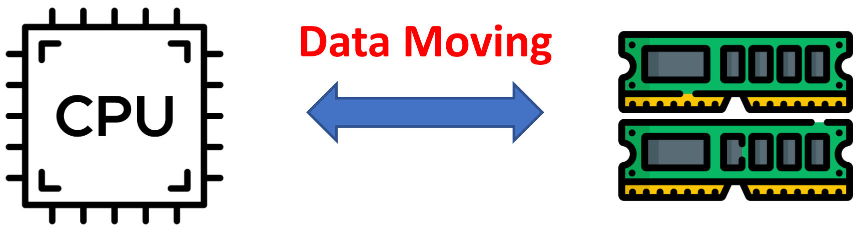
Memory-based Challenge: Difficulty of data access by computation units

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

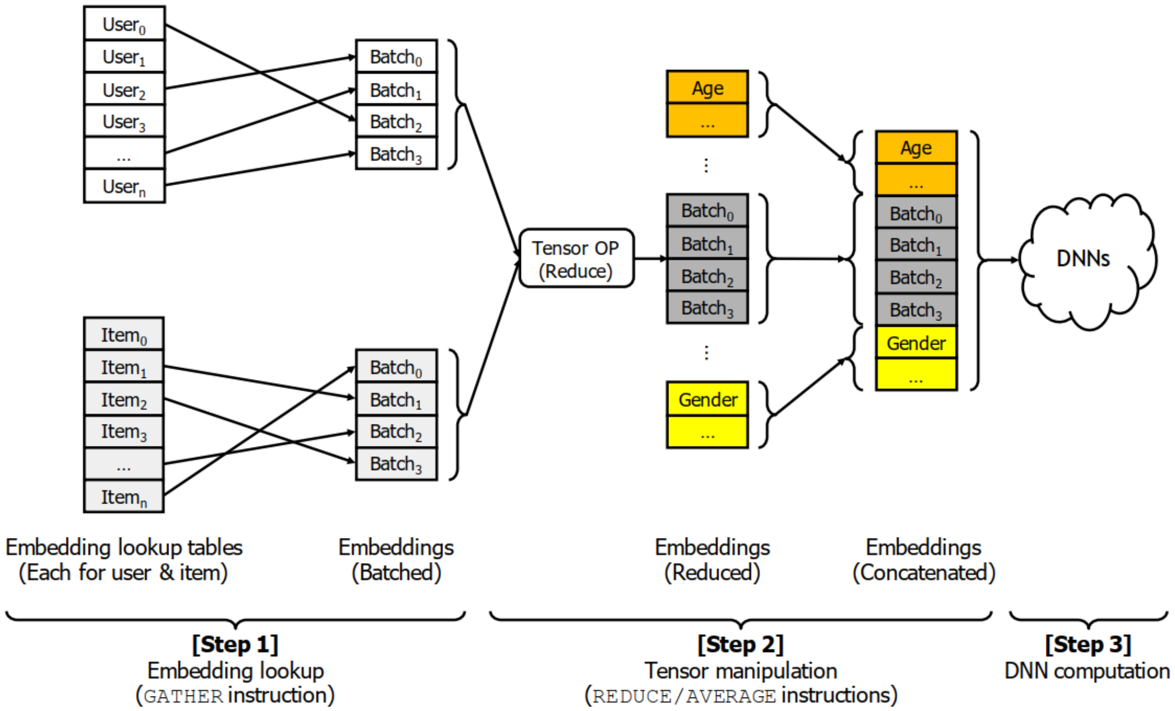


The computing units advance much, while memory techniques improve slowly. Such gap causes the problem of **memory wall**. Hardware-related methods aim to **optimize data moving** between the storage device and computing units.

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.

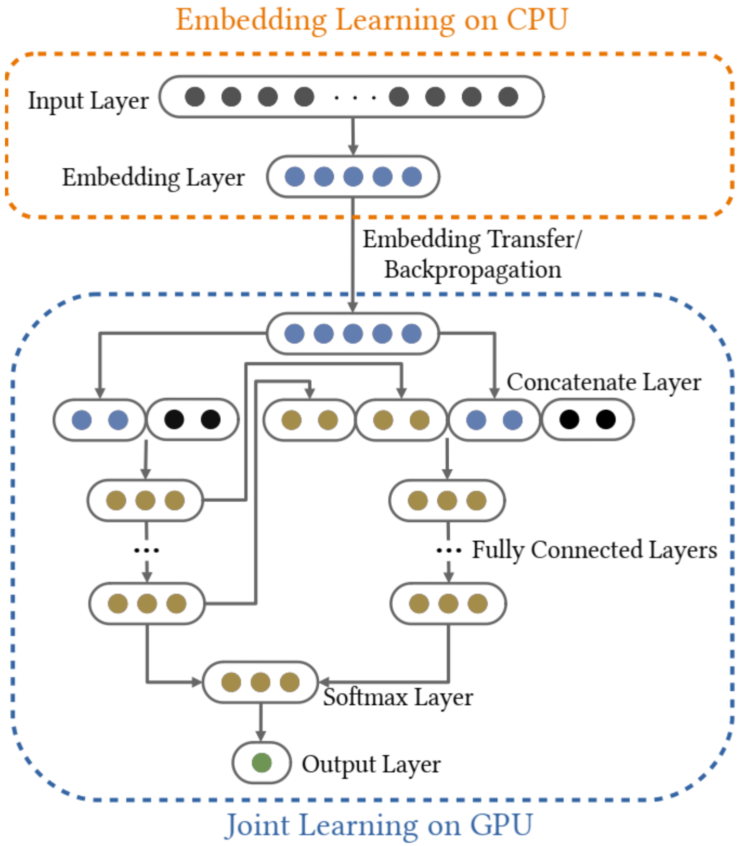
- **TensorDIMM** – MICRO’19
  - The first to explore architectural solutions for sparse embedding layer.
  - Propose a runtime system to utilize the TensorDIMM for tensor operations.



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

Hardware-related

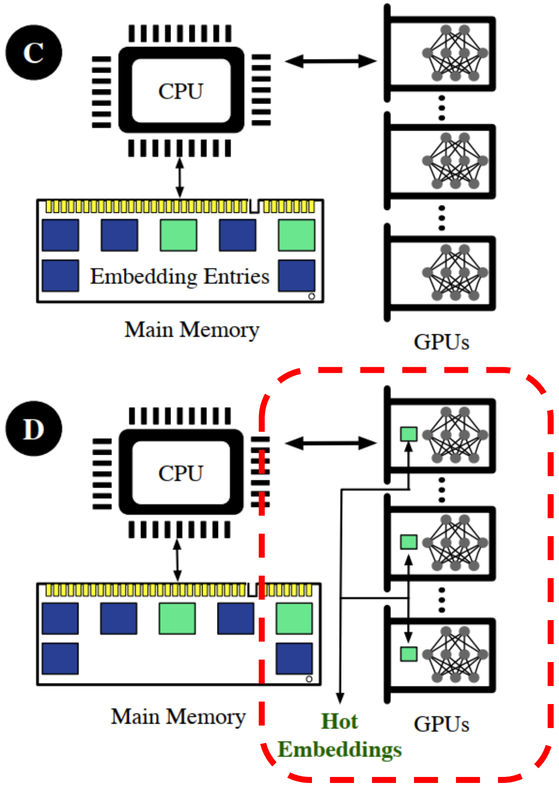
- **Cache Optimization**
  - Optimize the cache allocation mechanism to store the frequently accessed data on the memory device.
- **AIBox** – CIKM'19
  - Partition the model into two parts:
    - (1) Memory-intensive part: Embedding Learning on CPU.
    - (2) Computation-intensive part: Joint Learning 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

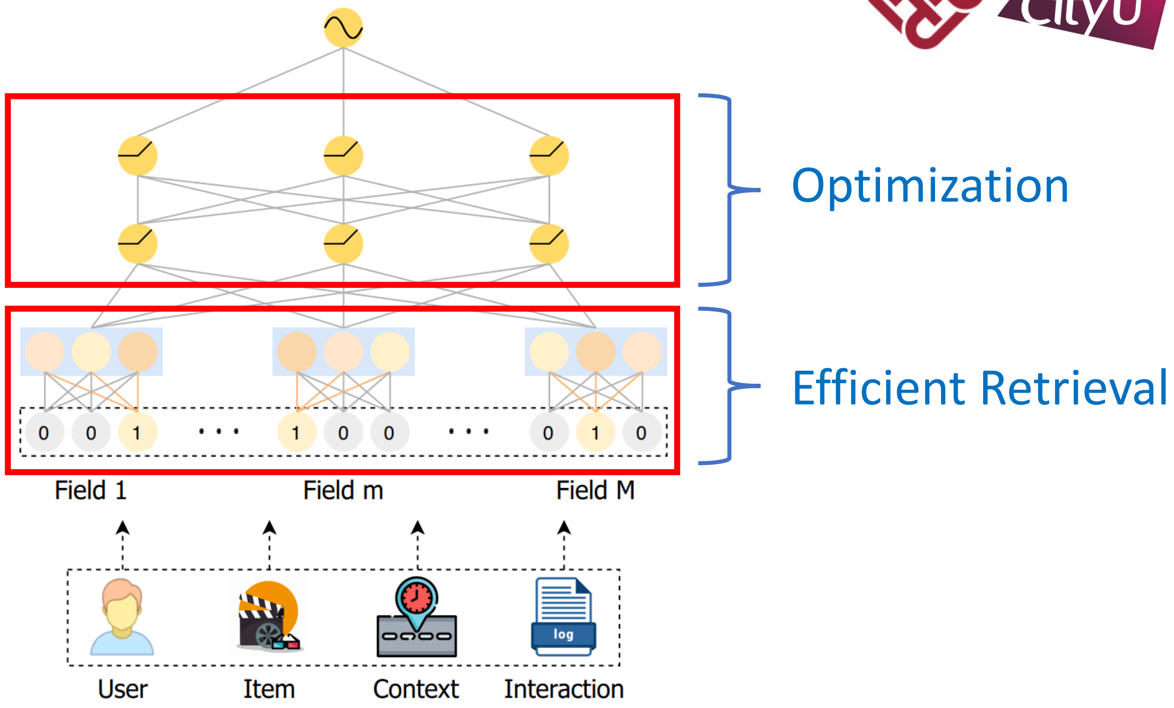
- **CPU-GPU Co-design**
  - Due to huge embedding tables, the embedding part is often stored and processed on CPU and DNN part on CPU. CPU-GPU co-design reduces the communication costs between CPU and GPU.
- **FAE** – VLDB'22
  - Utilize the scarce GPU memory to store the highly accessed embeddings, 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 $\texttt{Opt}(\cdot)$

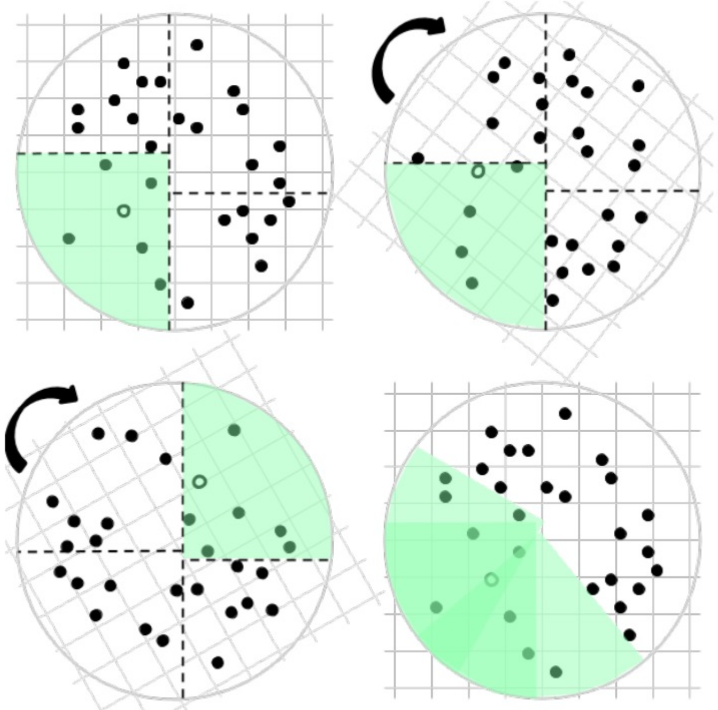
for $t \leftarrow 1$ to T **do**
  Draw b samples B from $\mathcal{D}$
  $\boldsymbol{g}_t, \boldsymbol{g}_t^e \leftarrow \frac{1}{b}\sum_{x \in B} \nabla L(x, w_t, w_t^e)$
  $w_{t+1} \leftarrow \eta \cdot \texttt{Opt}(w_t, \boldsymbol{g}_t)$ // Update dense weights
  for each field and each column in the field **do**
    $n_{\boldsymbol{g}} \leftarrow \|\boldsymbol{g}_t^e[\mathrm{id}_k^{\mathrm{f}_j}]\|$
    $\texttt{cnt} \leftarrow |\{x \in B \mid \mathrm{id}_k^{\mathrm{f}_j} \in x\}|$ // 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.

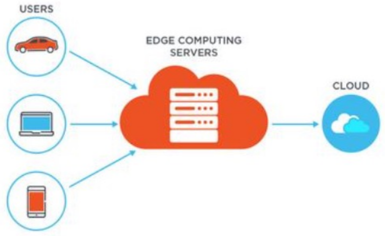
		Training	Inference
Hardware-related	Near/In Memory Computing	[196]	[78, 164, 190, 195, 367, 371]
	Cache Optimization	[135, 165, 403, 442]	[93, 397]
	CPU-GPU Co-design	[4, 5, 197, 308, 441, 450]	-
Software-related	Optimization	[128, 137, 146, 411, 454]	[140, 141]
	Efficient Retrieval	-	[81, 113, 191,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

Applications

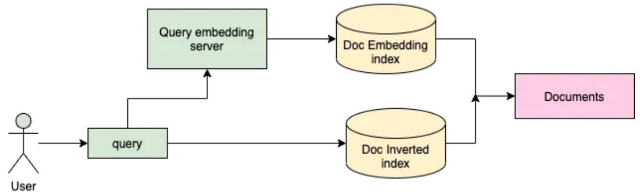
- **Large Language model**:
  - The emergence of LLMs urge recommendation to step into large model period. The environmental well-being is a vital issue.

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

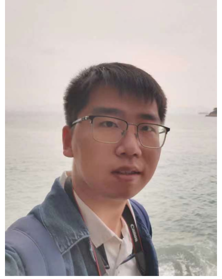
Jingtong Gao

Privacy

Lin Wang

Environmental Well-being

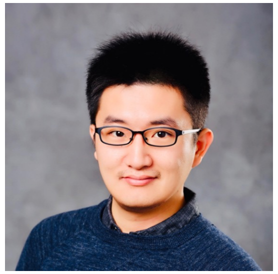
Accountability & Auditability



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

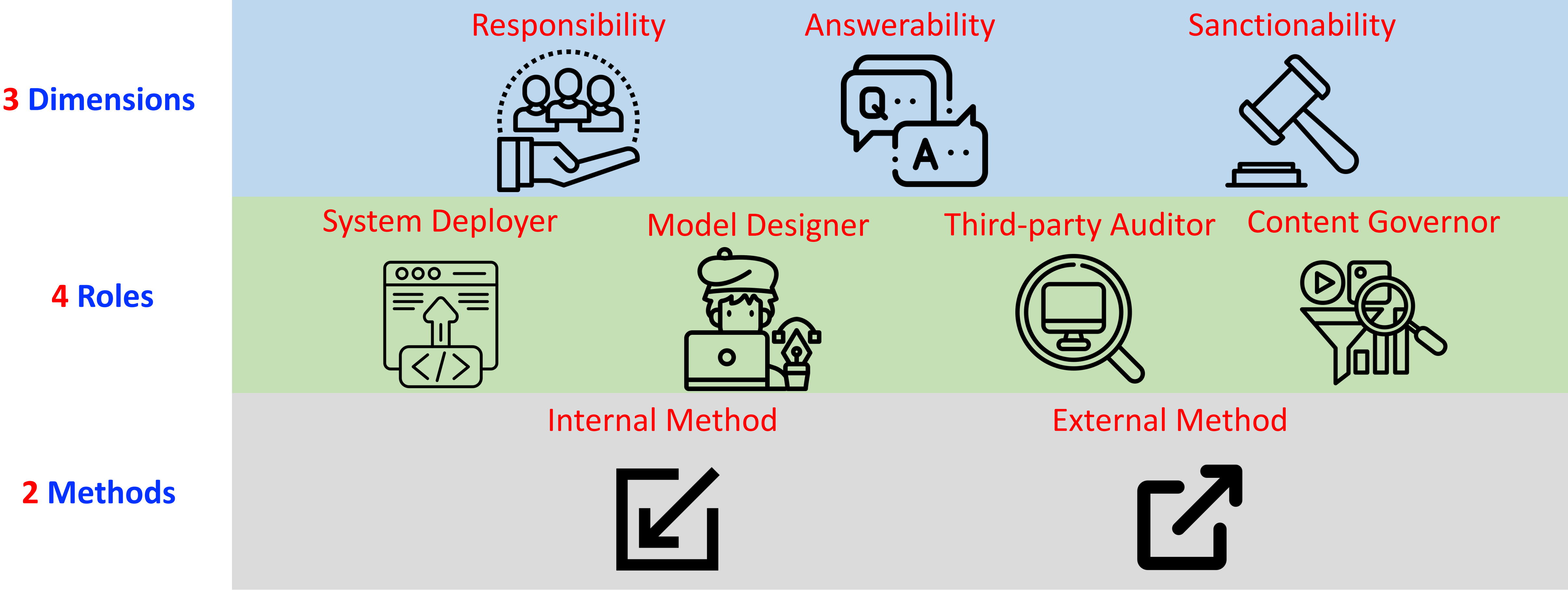
Recommending Videos

trust

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

Background

- Accountability & Auditability

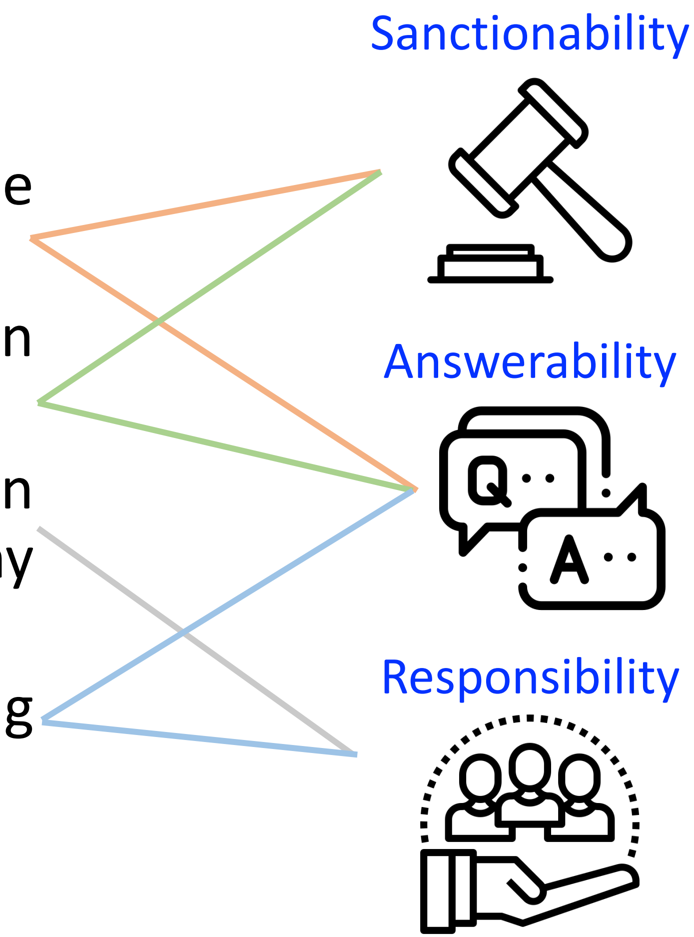


Accountability

- **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

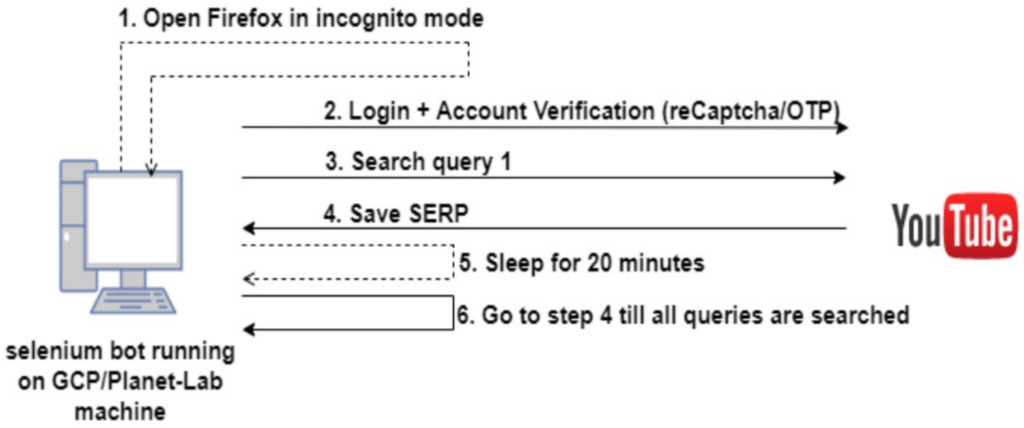
- **Four roles for an accountable RS**
  - **Content Governors**: responsible for examining the facticity and noxiousness of "items" in an RS.
  - **Model Designers**: build the recommendation models for service.
  - **System Deployers**: deploy recommendation models online and check the possible trustworthy 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

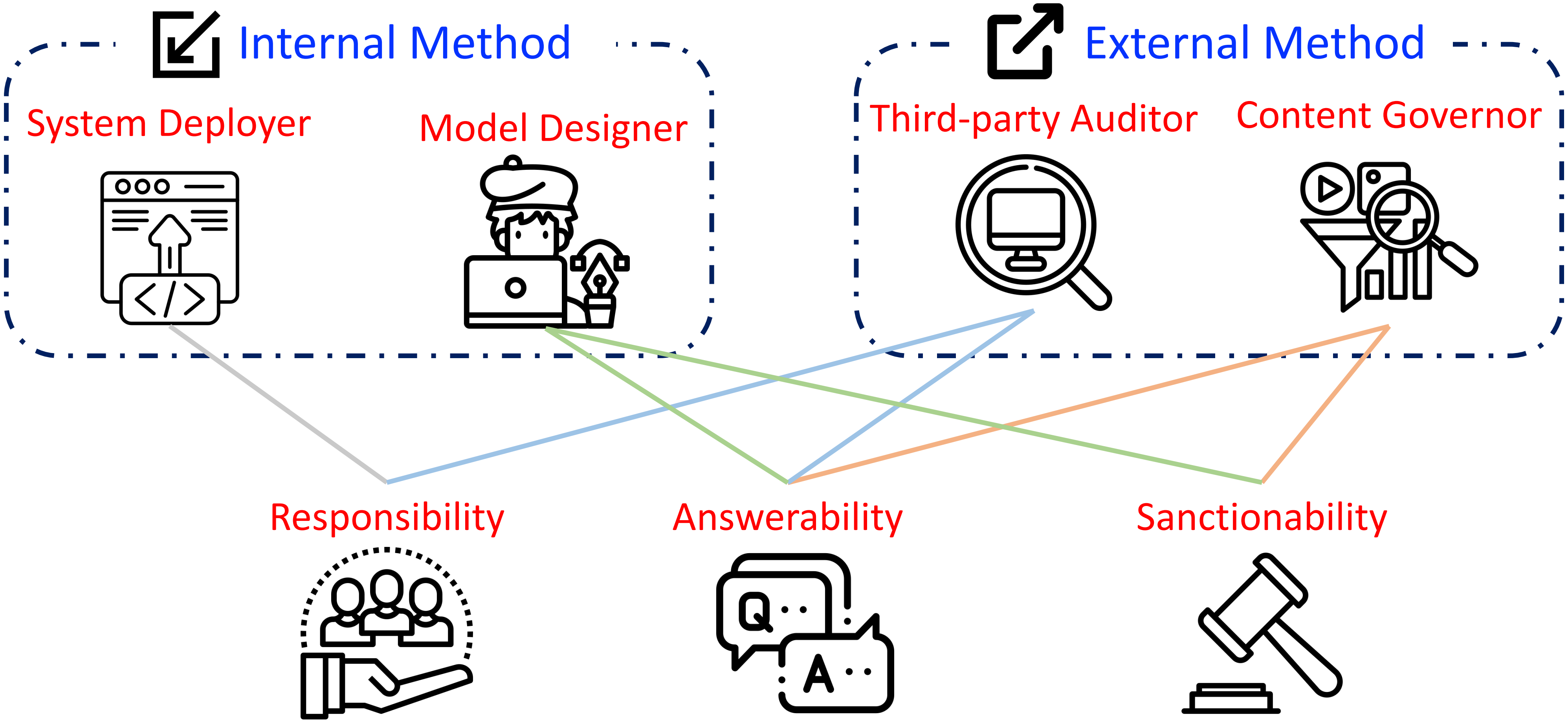
Auditability

- **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

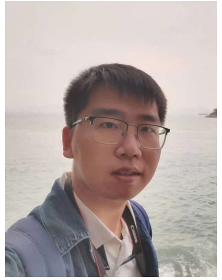
Jingtong Gao

Privacy

Lin Wang

Environmental Well-being

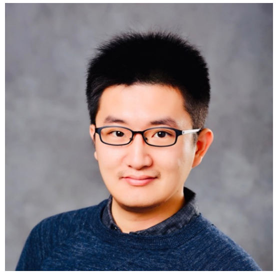
Accountability & Auditability



Qidong Liu

Dimension Interactions

Future Directions



Xiangyu Zhao

Trustworthy Recommender Systems

Introduction

Wenqi Fan

Non-discrimination & Fairness

Xiao Chen

Safety & Robustness

Shijie Wang

Explainability

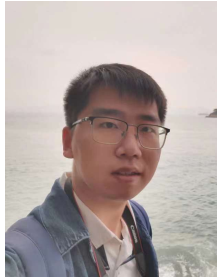
Jingtong Gao

Privacy

Lin Wang

Environmental Well-being

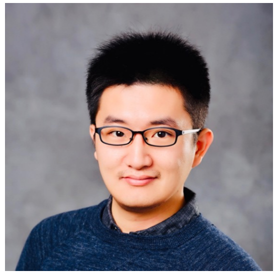
Accountability & Auditability



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

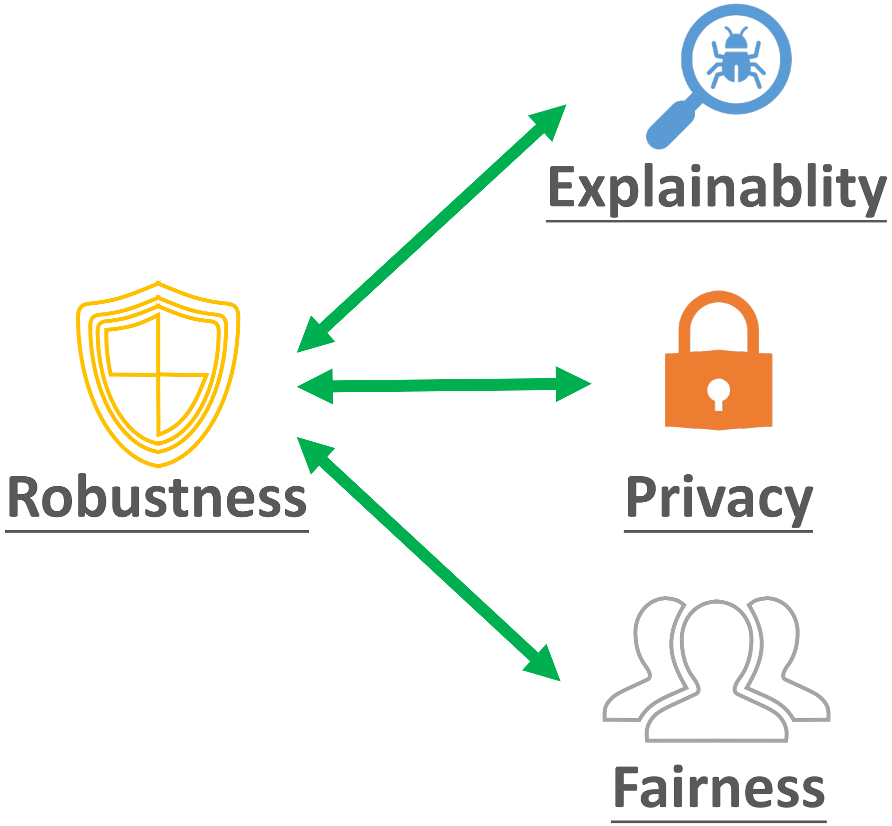
Why? Because these features may have many varying levels of interdependence, and even conflict in some aspects

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

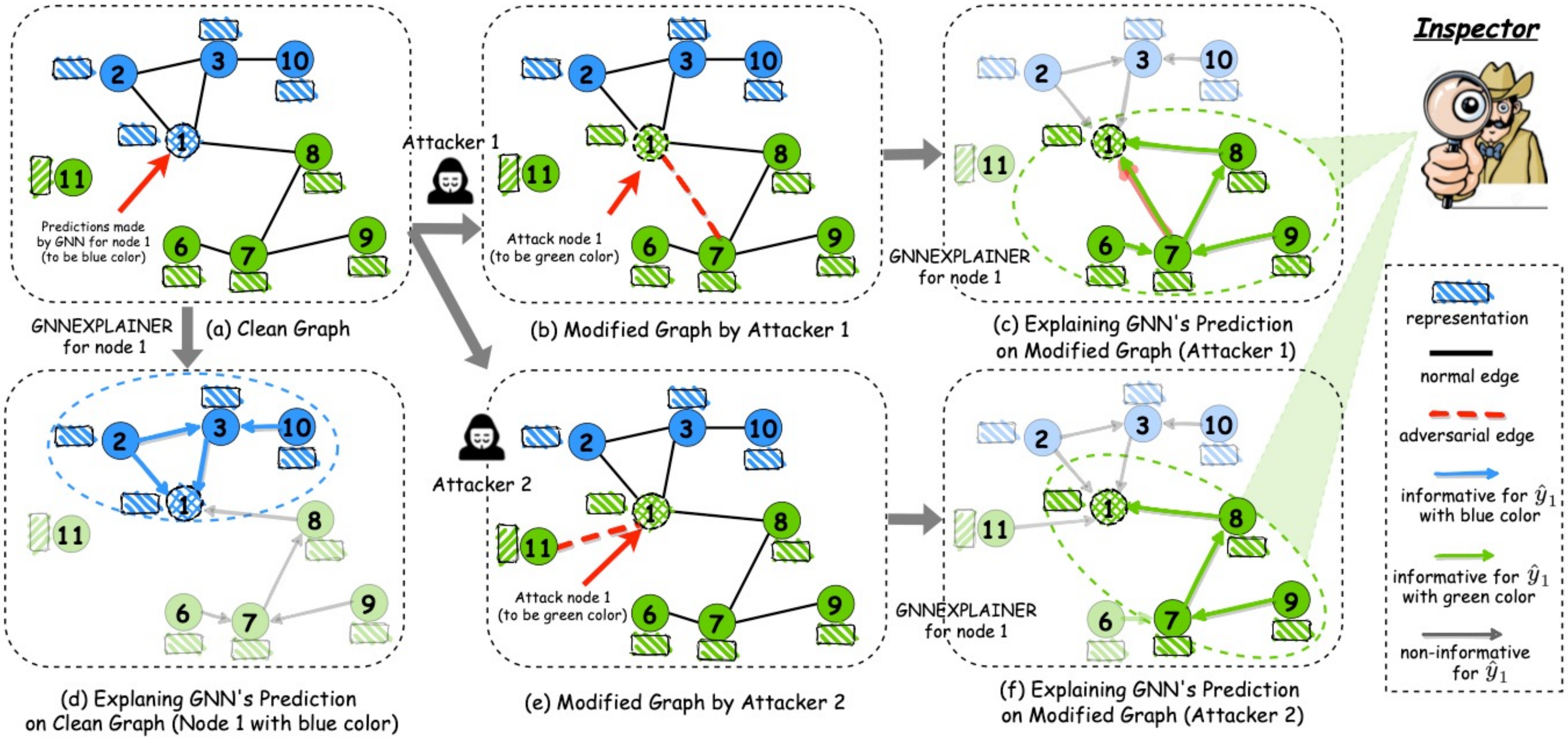
Robustness ↔ Explainability

- **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

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

$$\begin{aligned}\min_\theta \; \mathcal{L}_{\text{GNN}}(f_\theta(\mathbf{A}, \mathbf{X})) &:= \sum_{v_i \in V_L} \ell\left(f_\theta(\mathbf{A}, \mathbf{X})_{v_i}, y_i\right) \qquad (1)\\ &= -\sum_{v_i \in V_L} \sum_{c=1}^{C} \mathbb{I}[y_i = c] \ln(f_\theta(\hat{\mathbf{A}}, \mathbf{X})^c_{v_i})\end{aligned}$$

GNNExplainer

$$\begin{aligned}&\max_{(\mathbf{A}_S, \mathbf{X}_S)} MI\left(Y, (\mathbf{A}_S, \mathbf{X}_S)\right)\\ \rightarrow &\min_{(\mathbf{A}_S, \mathbf{X}_S)} H(Y \mid \mathbf{A} = \mathbf{A}_S, \mathbf{X} = \mathbf{X}_S)\\ \approx &\min_{(\mathbf{A}_S, \mathbf{X}_S)} -\sum_{c=1}^{C} \mathbb{I}[\hat{y}_i = c] \ln f_\theta(\mathbf{A}_S, \mathbf{X}_S)^c_{v_i}\end{aligned}$$

Adversarial Edges $\Rightarrow$

$$\begin{aligned}&\min_{\mathbf{M}_A} \mathcal{L}_{\text{Explainer}}(f_\theta, \mathbf{A}, \mathbf{M}_A, \mathbf{X}, v_i, \hat{y}_i)\\ \rightarrow &\max_{\mathbf{M}_A} \sum_{c=1}^{C} \mathbb{I}[\hat{y}_i = c] \ln f_\theta(\mathbf{A} \odot \sigma(\mathbf{M}_A), \mathbf{X})^c_{v_i}\end{aligned}$$

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

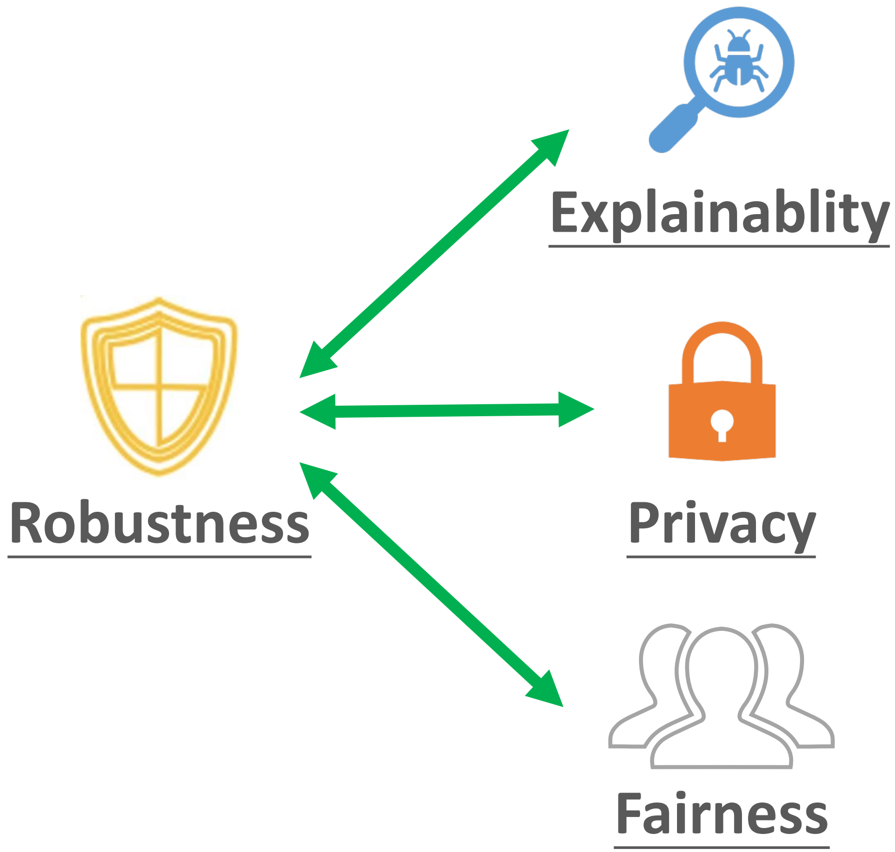
- GNNExplainer Attack:

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

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…



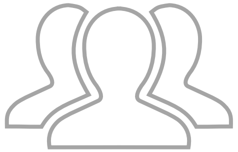
- **Zheng et al.** -> An additive causal model for disentangling user interest and conformity which **Ensures robustness and explainability in recommendation**
- **Bilge et al.** -> **Robust recommendation algorithms** based on collaborative filtering **with privacy enhancement**
- **Zhang et al.** -> 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

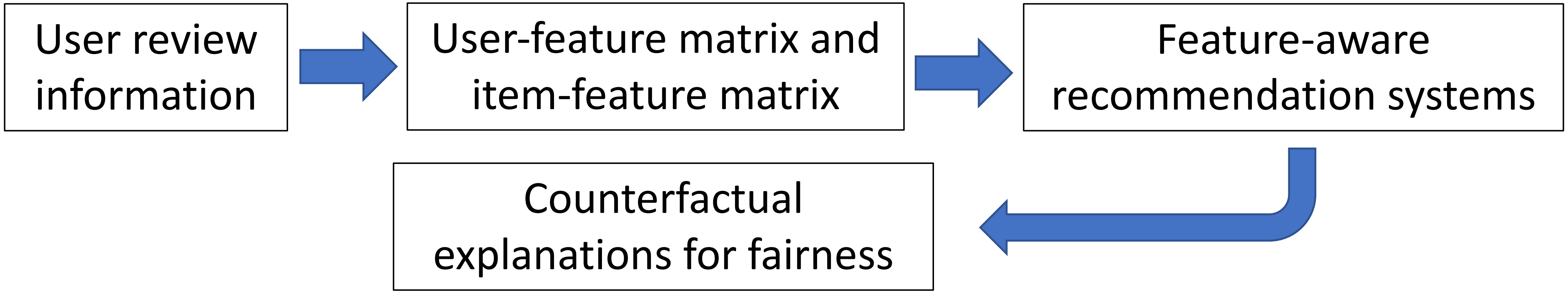
Fairness ↔ Explainability

- **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

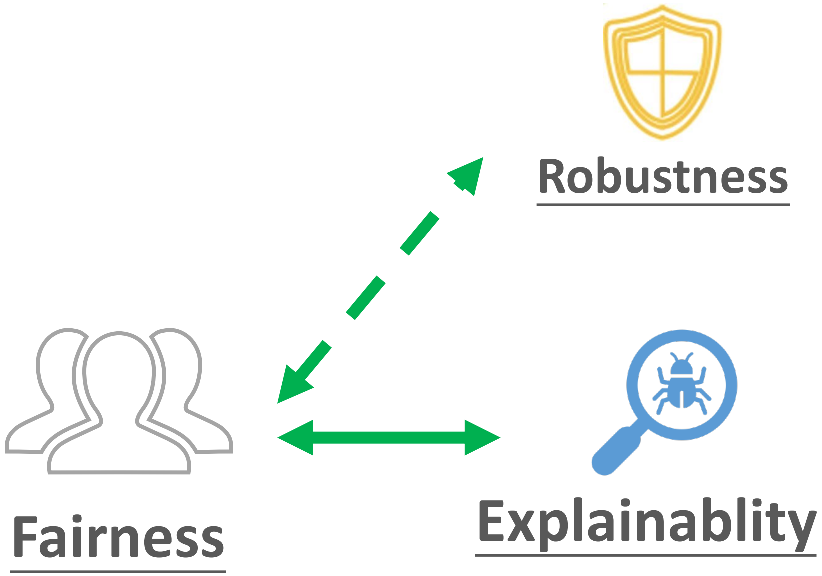
- Overall procedure:



- 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**

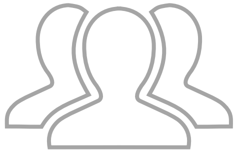
[1] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and debias in recommender system: A survey and future directions. ArXiv preprint abs/2010.03240 (2020). https://arxiv.org/abs/2010.03240

[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

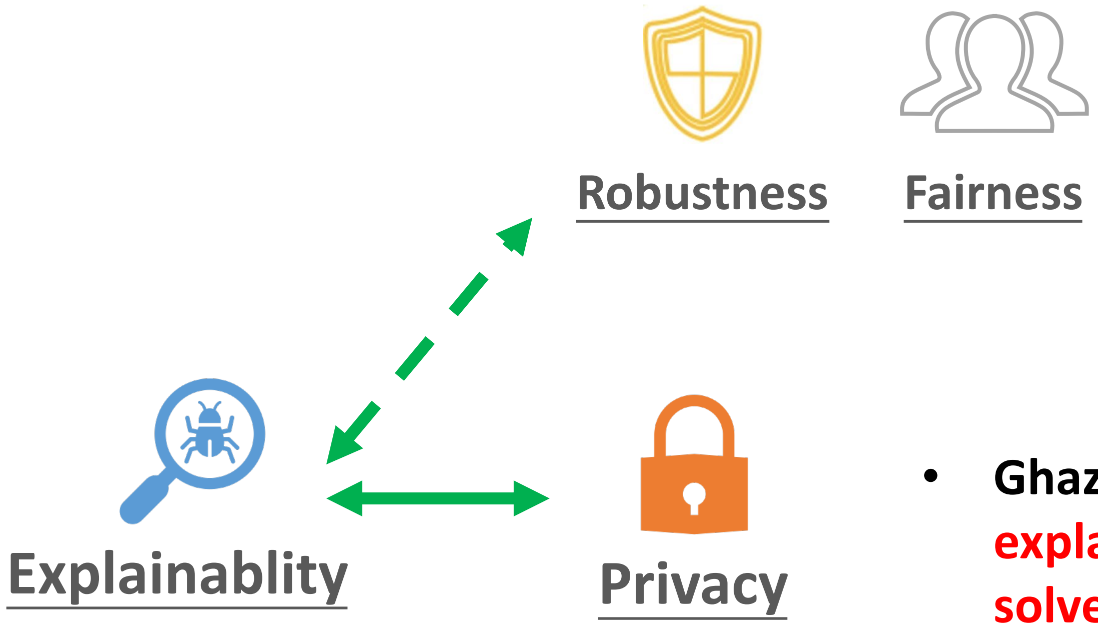
- Interactions with Robustness

- 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

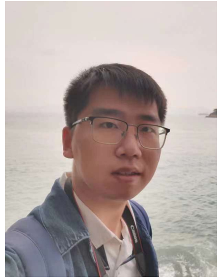
Jingtong Gao

Privacy

Lin Wang

Environmental Well-being

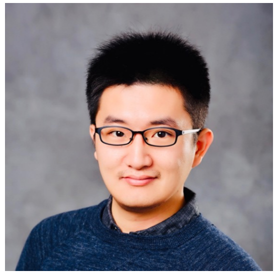
Accountability & Auditability



Qidong Liu

Dimension Interactions

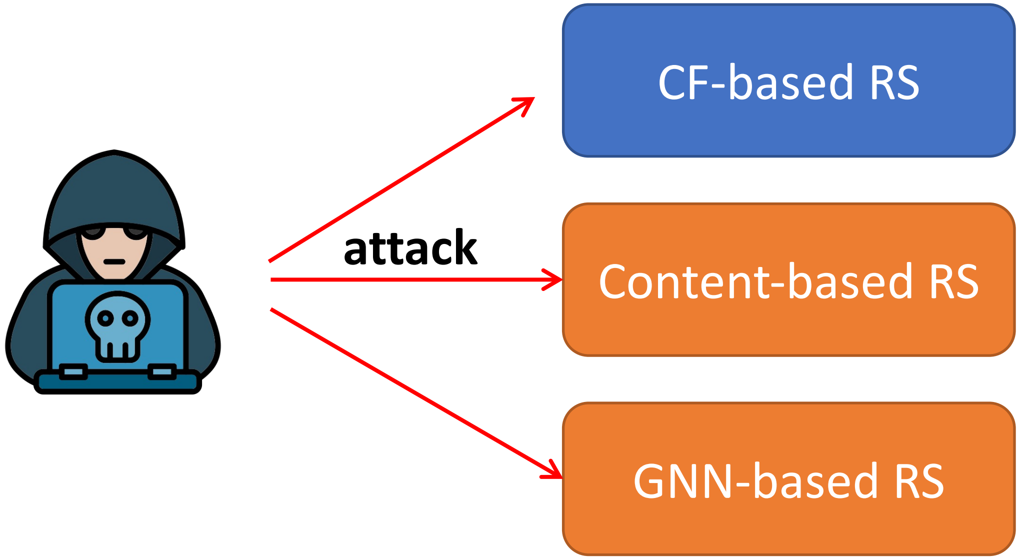
Future Directions



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.

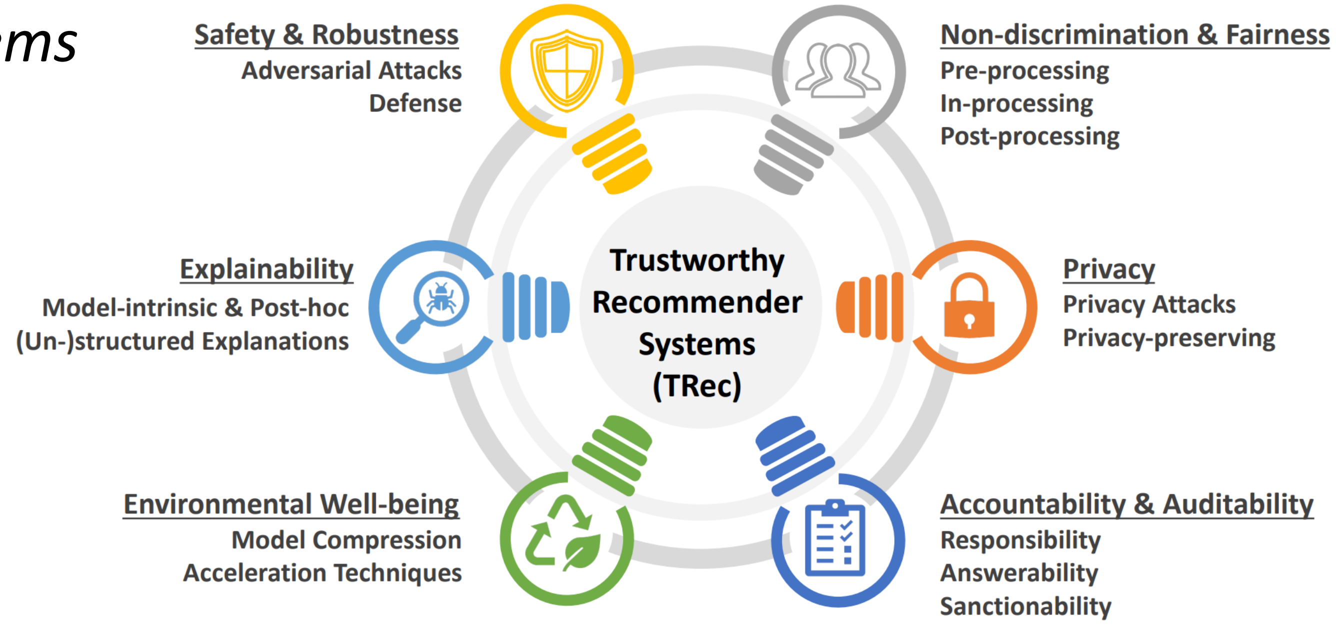
Future Directions in Other Dimensions

- **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

Dr. Wenqi Fan
The Hong Kong
Polytechnic University

Dr. Xiangyu Zhao
City University of
Hong Kong

A Comprehensive Survey on Trustworthy Recommender Systems
https://arxiv.org/pdf/2209.10117.pdf

WWW'2023 Tutorial Website (Slides)

https://advanced-recommender-systems.github.io/trustworthiness-tutorial/