Towards Secure and Robust Recommender Systems: A Data-Centric Perspective

Zongwei Wang[1], Junliang Yu[2], Tong Chen[2], Hongzhi Yin[2], Shazia Sadiq[2], Min Gao[1]
[1]Chongqing University, China,
[2]The University of Queensland, Australia

Website: https://secure-robust-recsys.github.io/

Presenters

Zongwei Wang

- A second-year PhD student

School of Big Data and Software Engineering, CQU

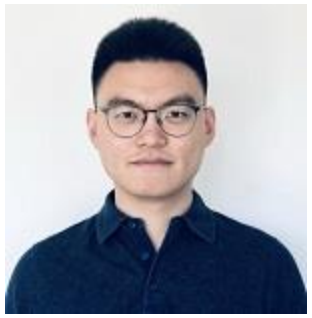
zongwei@cqu.edu.cn

Dr. Junliang Yu

- Lecturer

School of Information Technology and Electrical Engineering, UQ

jl.yu@uq.edu.au



Dr. Tong Chen

- Lecturer

School of Information Technology and Electrical Engineering, UQ

tong.chen@uq.edu.au

Outlines

1. Introduction

2. Securing Data Integrity

3. Preserving Data Privacy

4. Managing Data Noise

5. Limitations and Opportunities

6. Toolkit

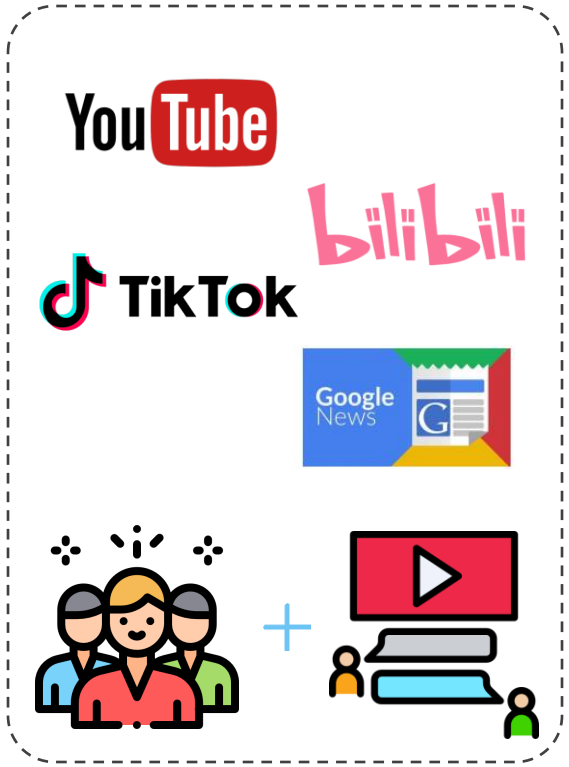
1. Introduction

Different Platforms Use Recommender System

1. Introduction

Recommender Systems Find Widespread Application in the Following Fields:

E-Commerce



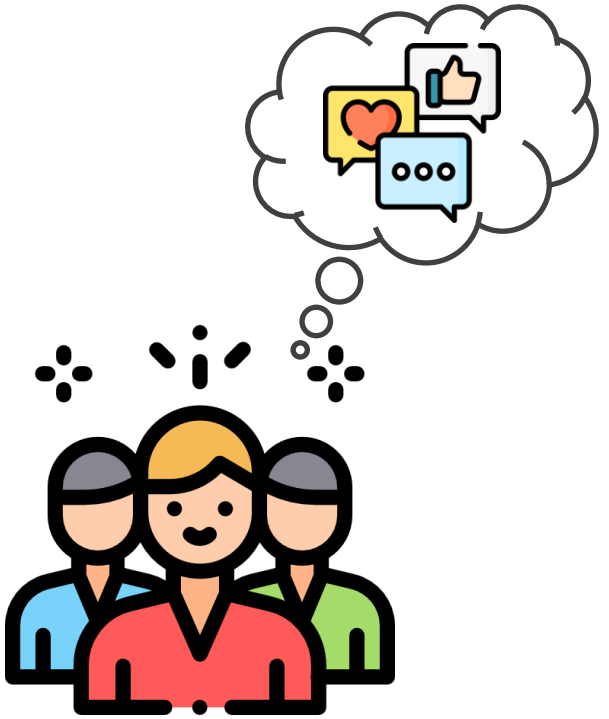
Video Streaming

Social Media

1. Introduction

Recommend Item X

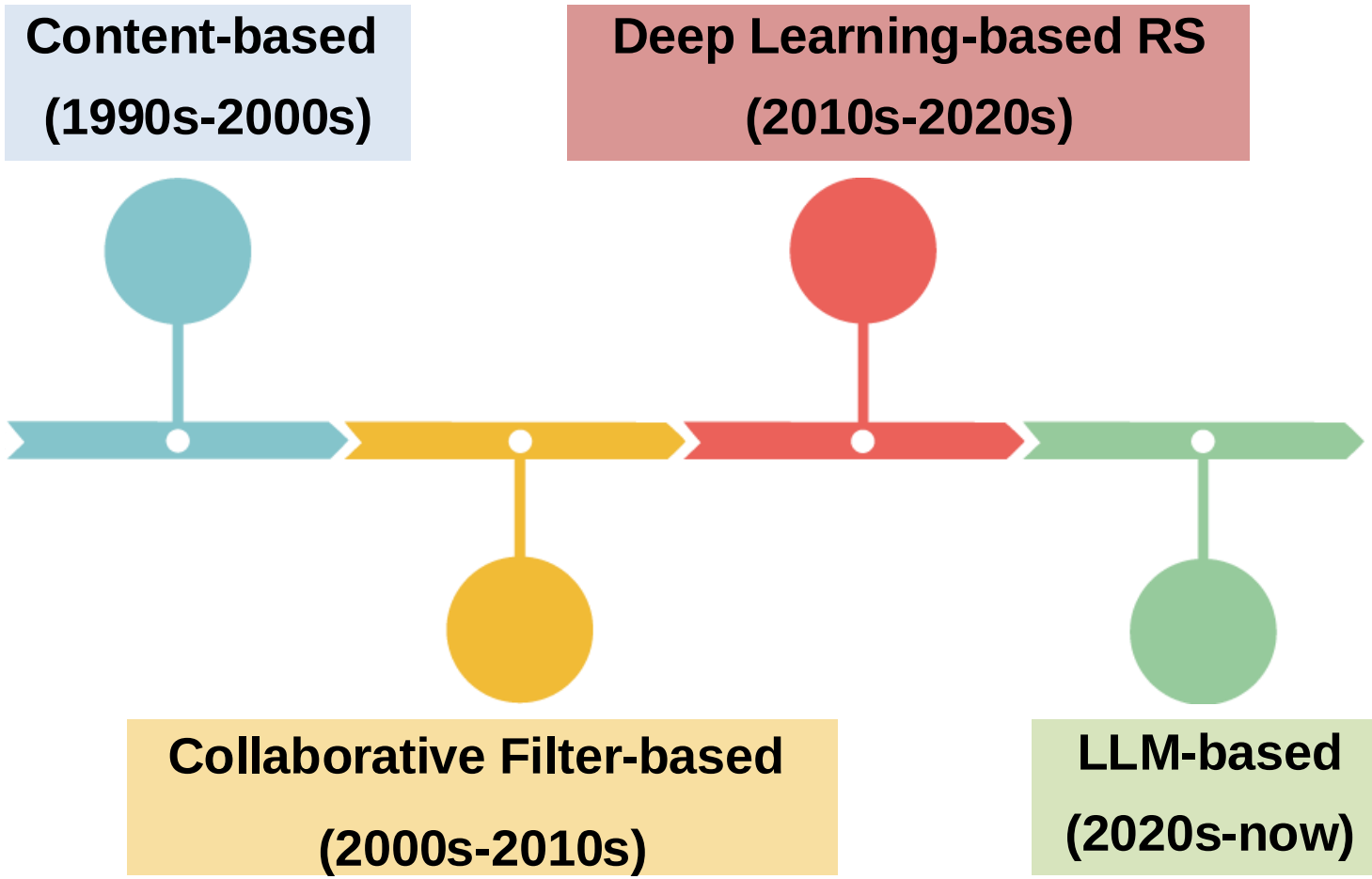
to Users



Items: Products, Movies, Friends, Videos, …

Large Number of Users

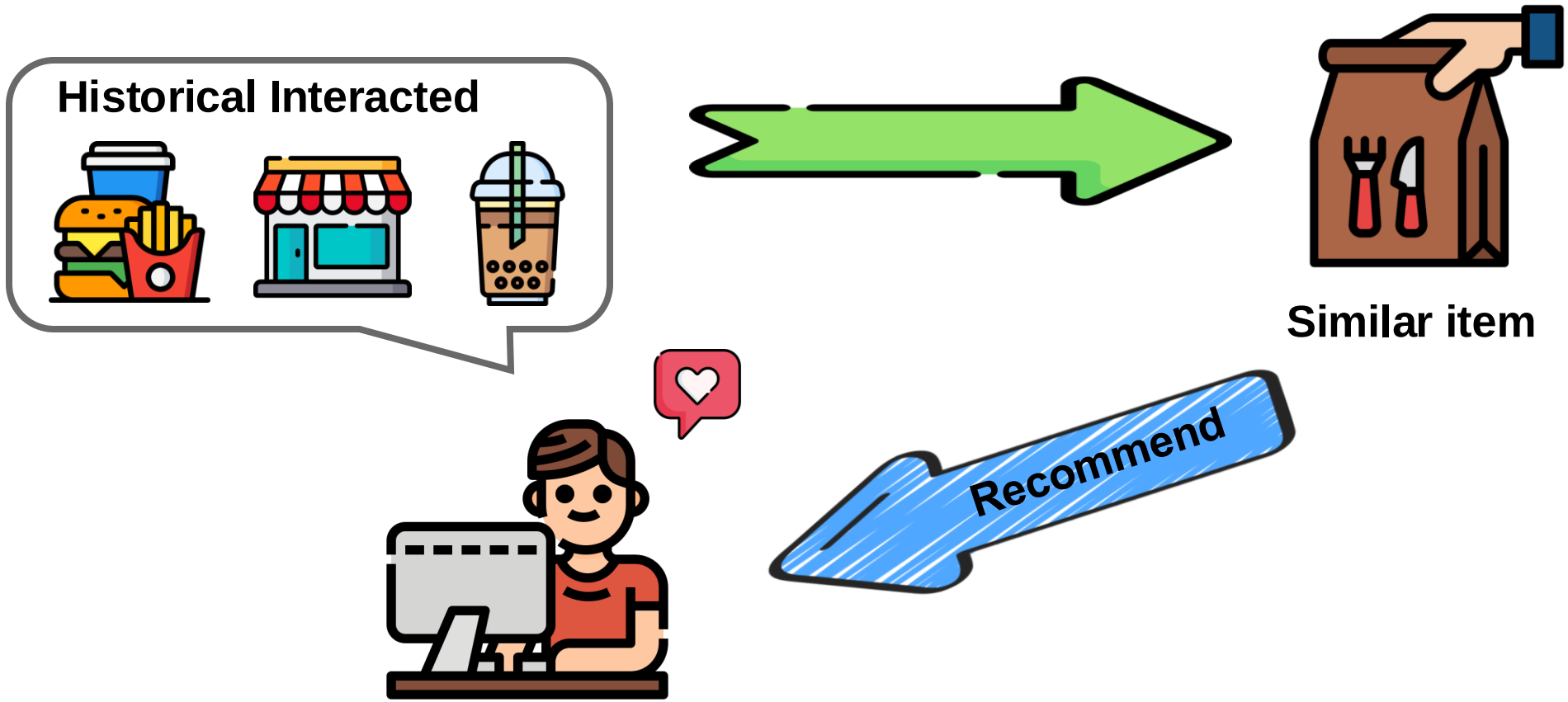
1. Introduction



1. Introduction

Content-based RS (1990s-2000s)

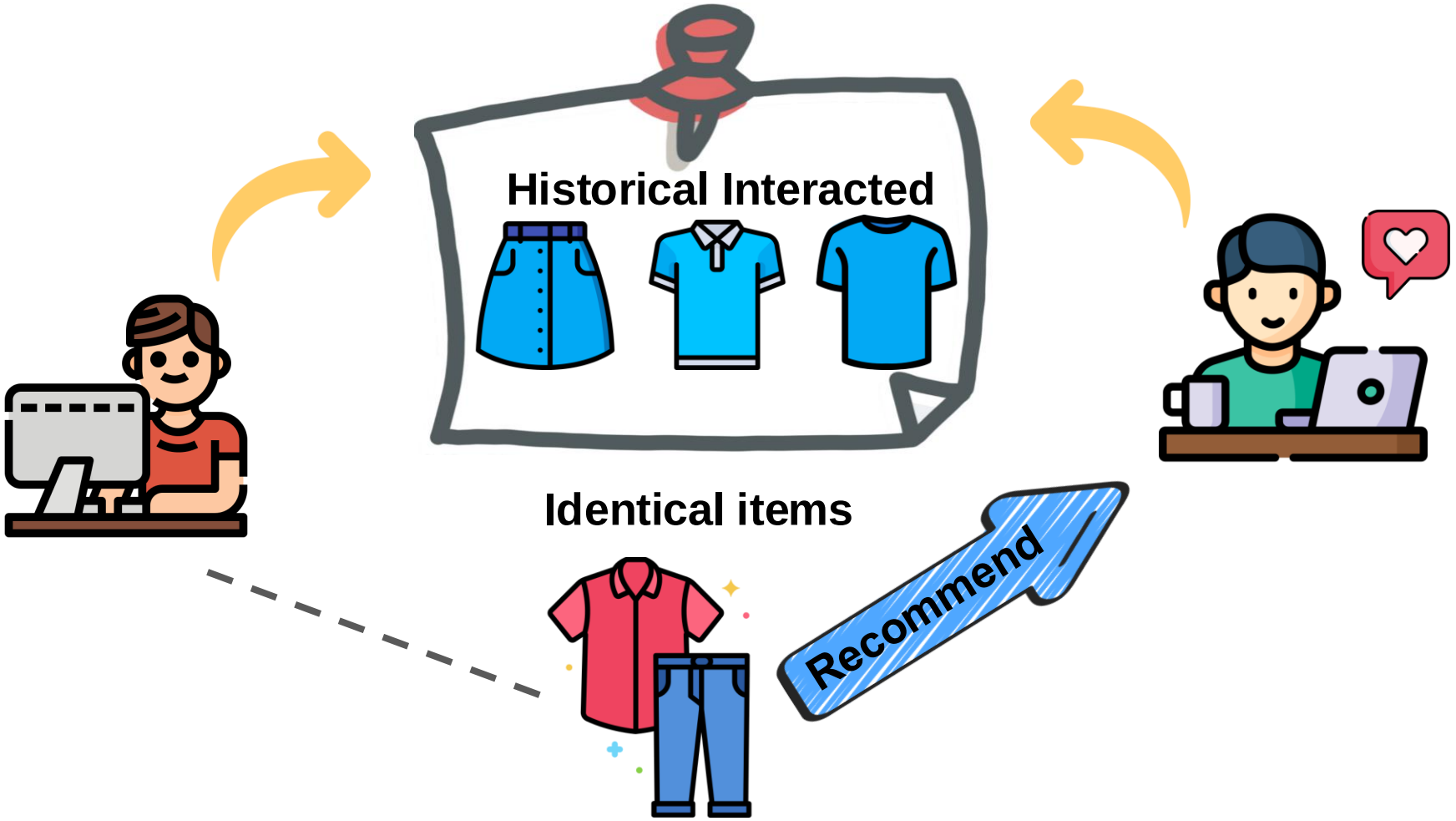
Users are likely to prefer items whose **content is similar to** those they have interacted with before.



1. Introduction

Collaborative Filter-based RS (2000s-2010s)

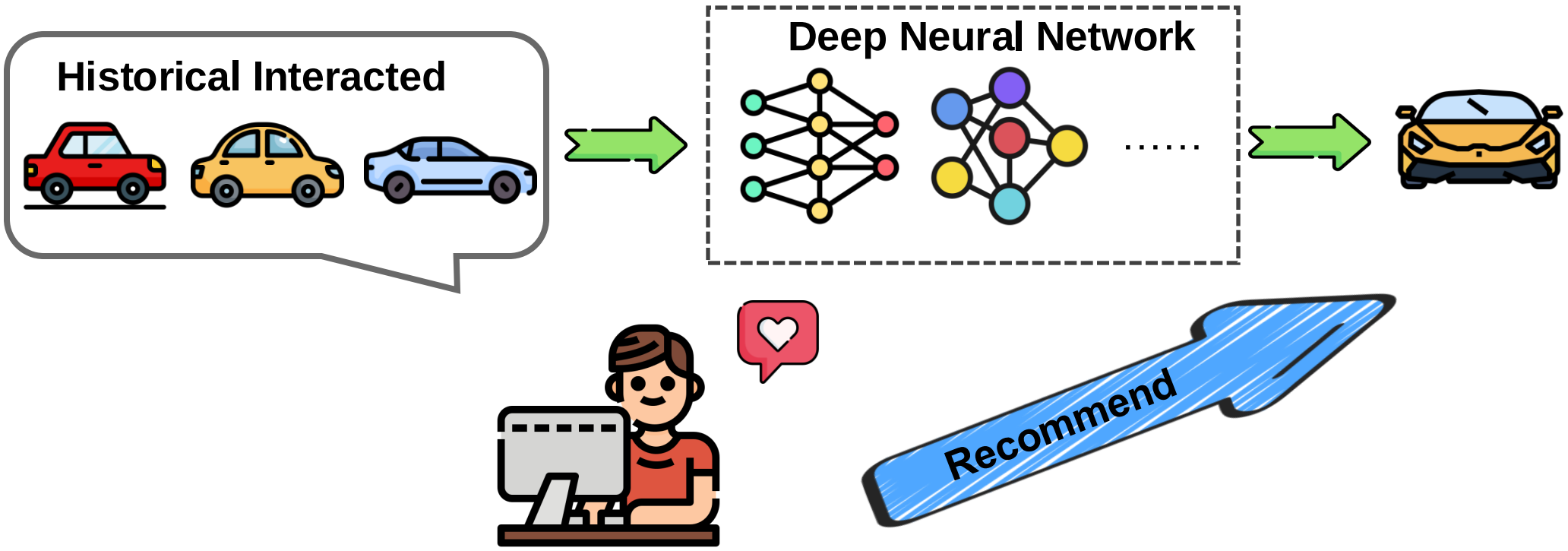
Users with **similar behaviours** tend to have similar preferences.



1. Introduction

Deep Learning-based RS (2010s-2020s)

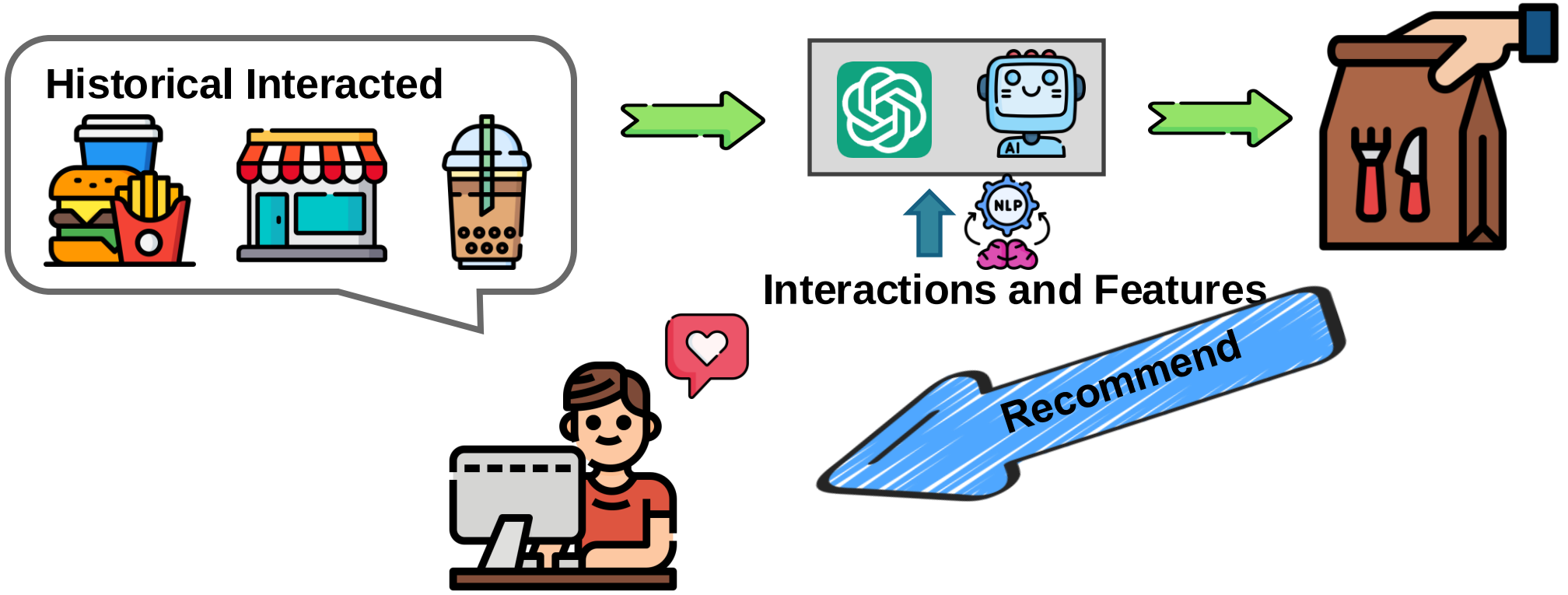
Applying **deep learning techniques** to uncover user preferences.



1. Introduction

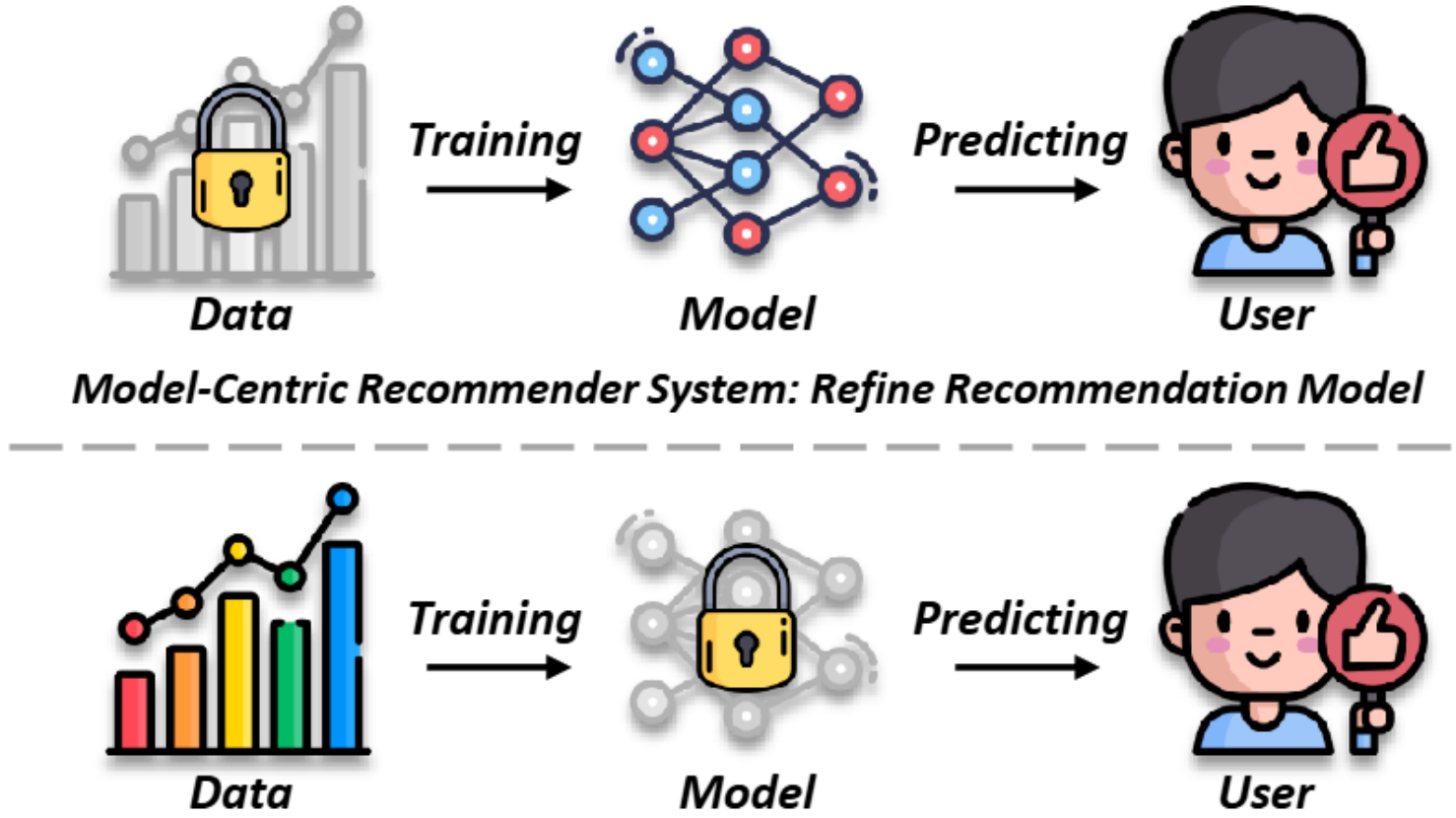
LLM-based RS (2020s-now)

Leveraging the powerful reasoning capabilities of **large language models** for recommendations.



1. Introduction

Model-Centric RS to Data-Centric RS



Lai, Riwei, Rui Chen, and Chi Zhang. "A survey on data-centric recommender systems." *arXiv preprint arXiv:2401.17878* (2024).

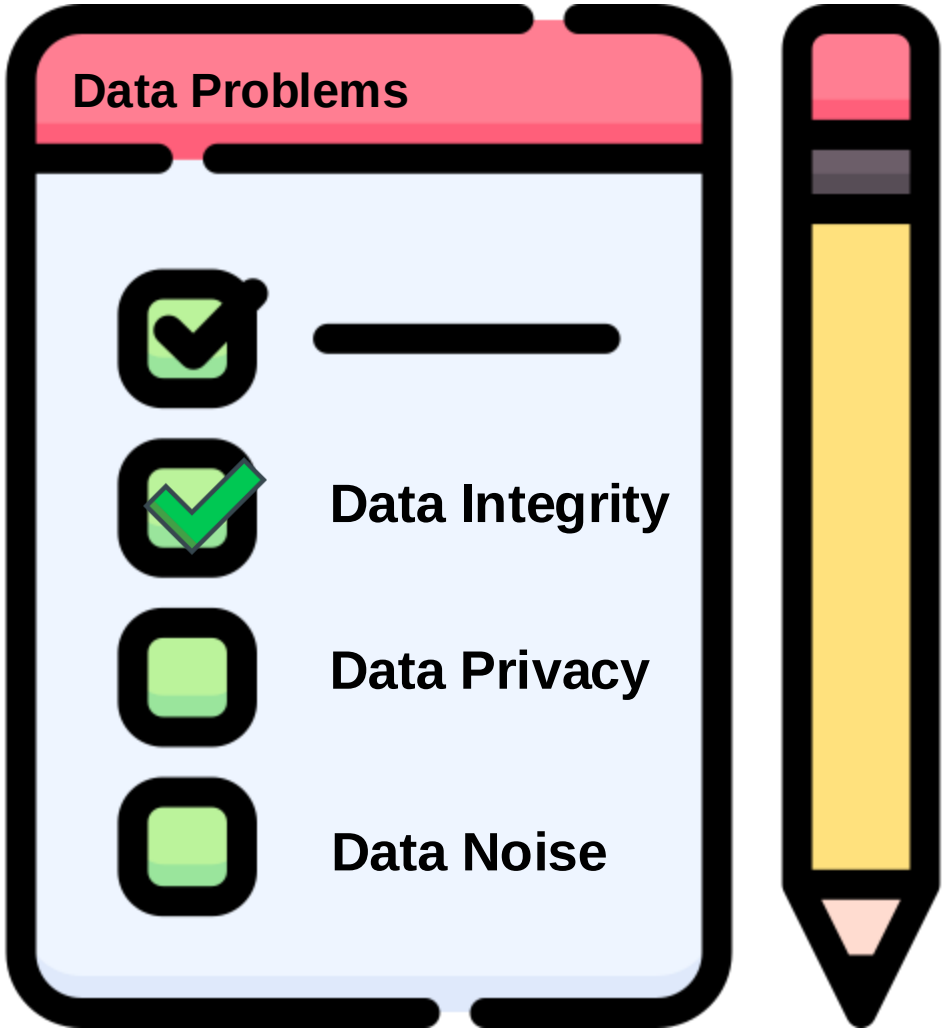
1. Introduction

Data Issues Appear in Data-Centric RS Pipeline

Data Collection	Data Preprocessing	Data Training	Data Monitoring
Multi-Source Data (Behavior or contextual data)	Data Clean (Missing or abnormal values)	Noise-Resistant model (Robustness enhancement)	Anomaly Detection (Deviations from expected patterns)

1. Introduction

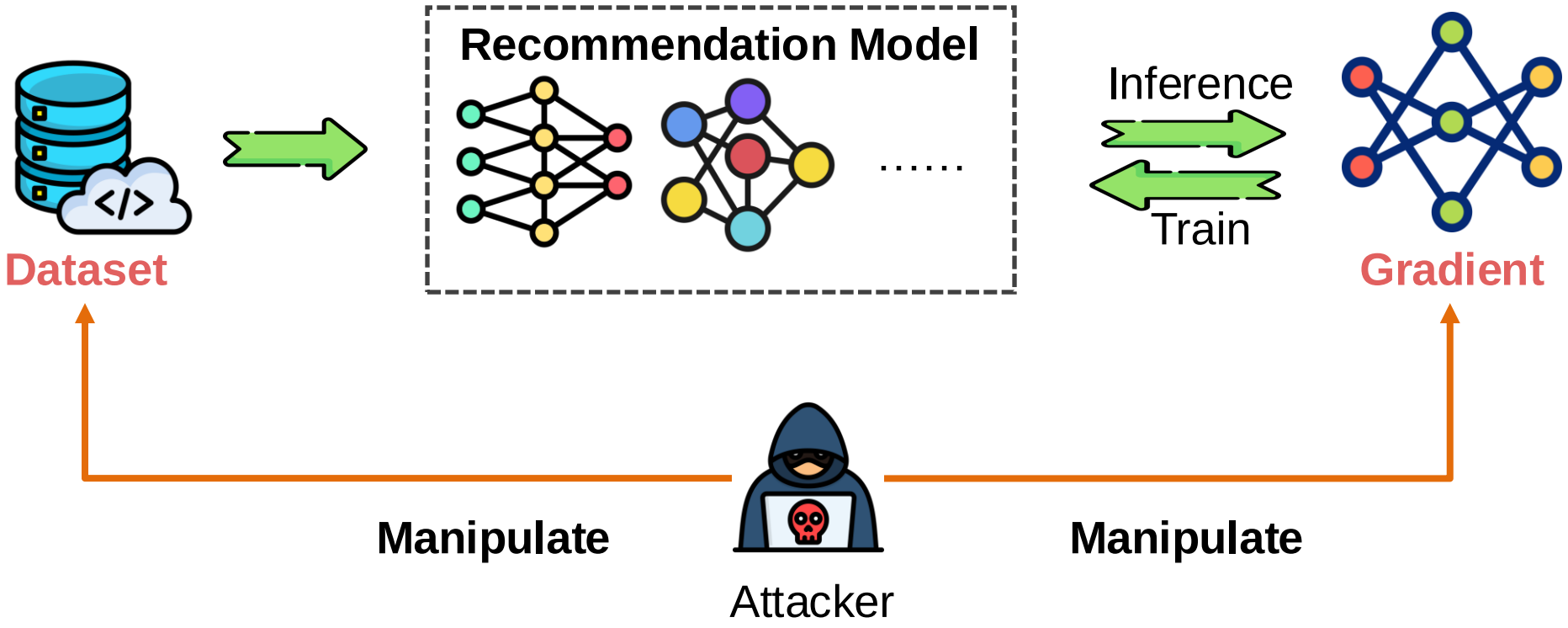
Data Integrity



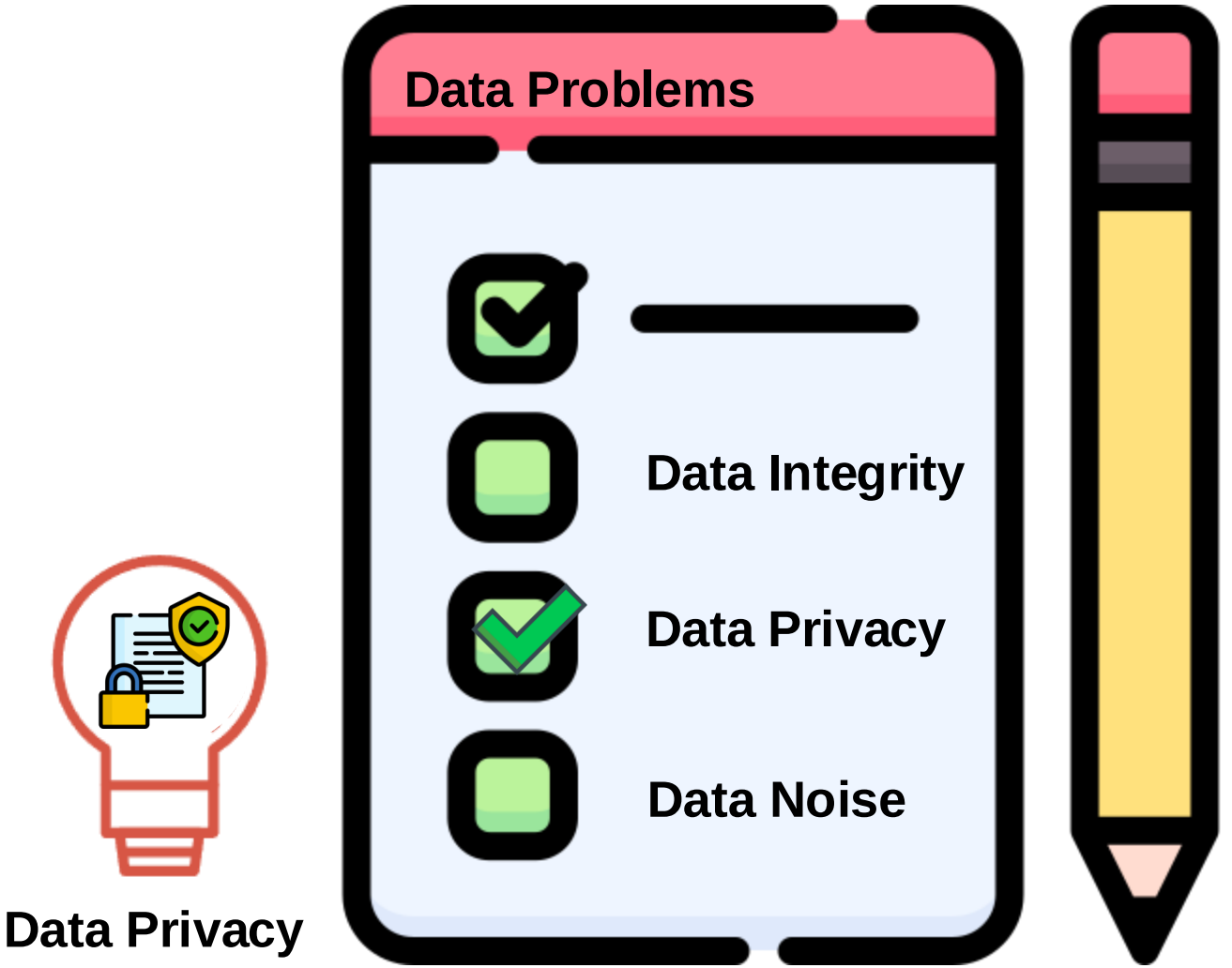
1. Introduction

Data Integrity Problem

Attackers target this data by corrupting training datasets or manipulating model gradients, ultimately distorting recommendation results.



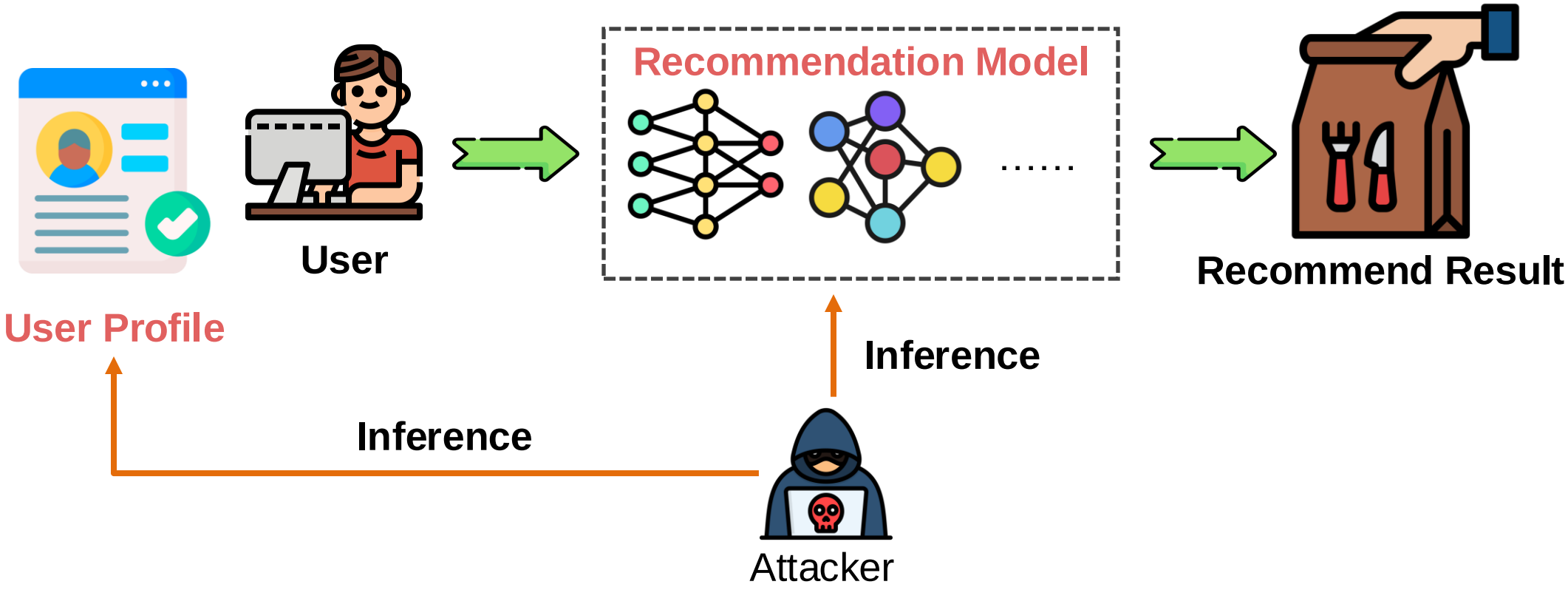
1. Introduction



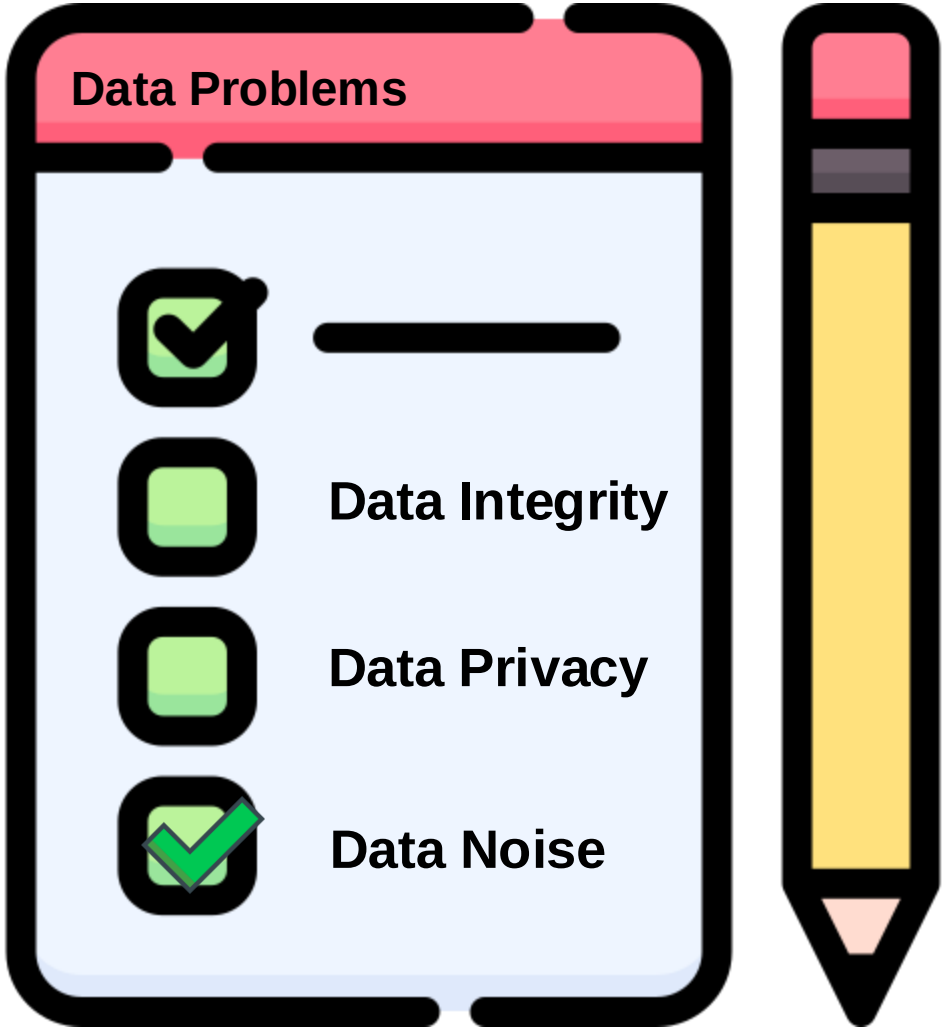
1. Introduction

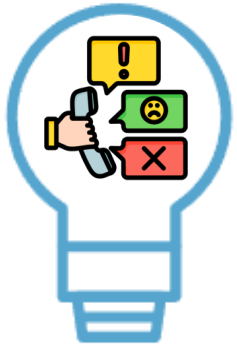
Data Privacy Problem

Attackers leverage the extensive data generated by recommender systems to uncover sensitive information.



1. Introduction



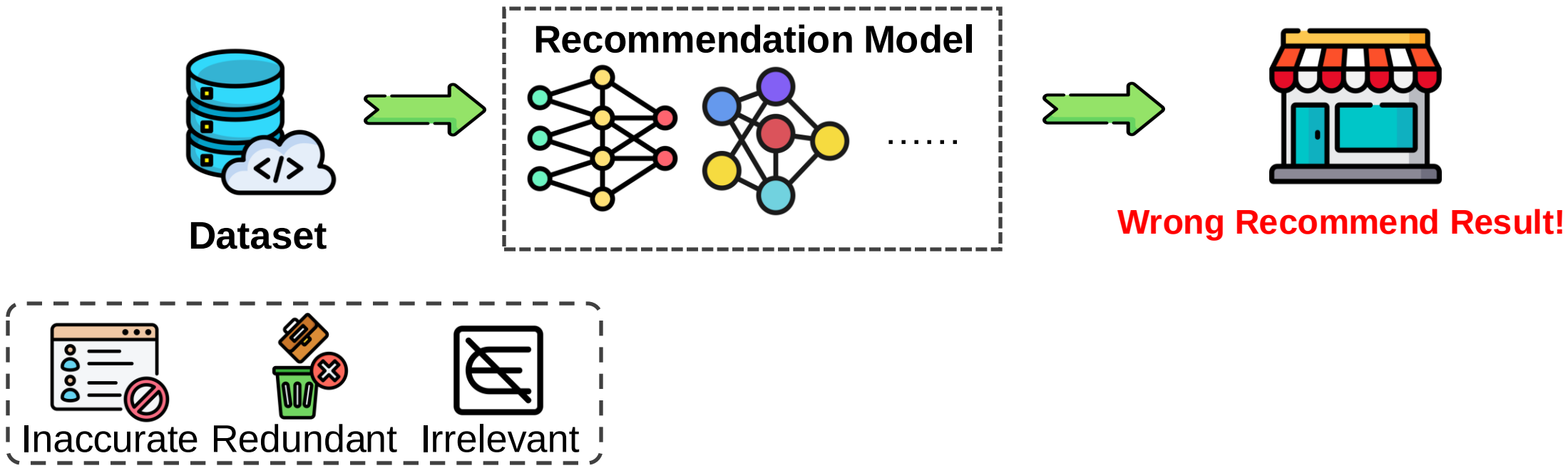


Data Noise

1. Introduction

Data Noise Problem

Noise in data, caused by inaccurate, redundant, or irrelevant information, can severely degrade system performance.



Outlines

2.1 Manipulating Data Integrity

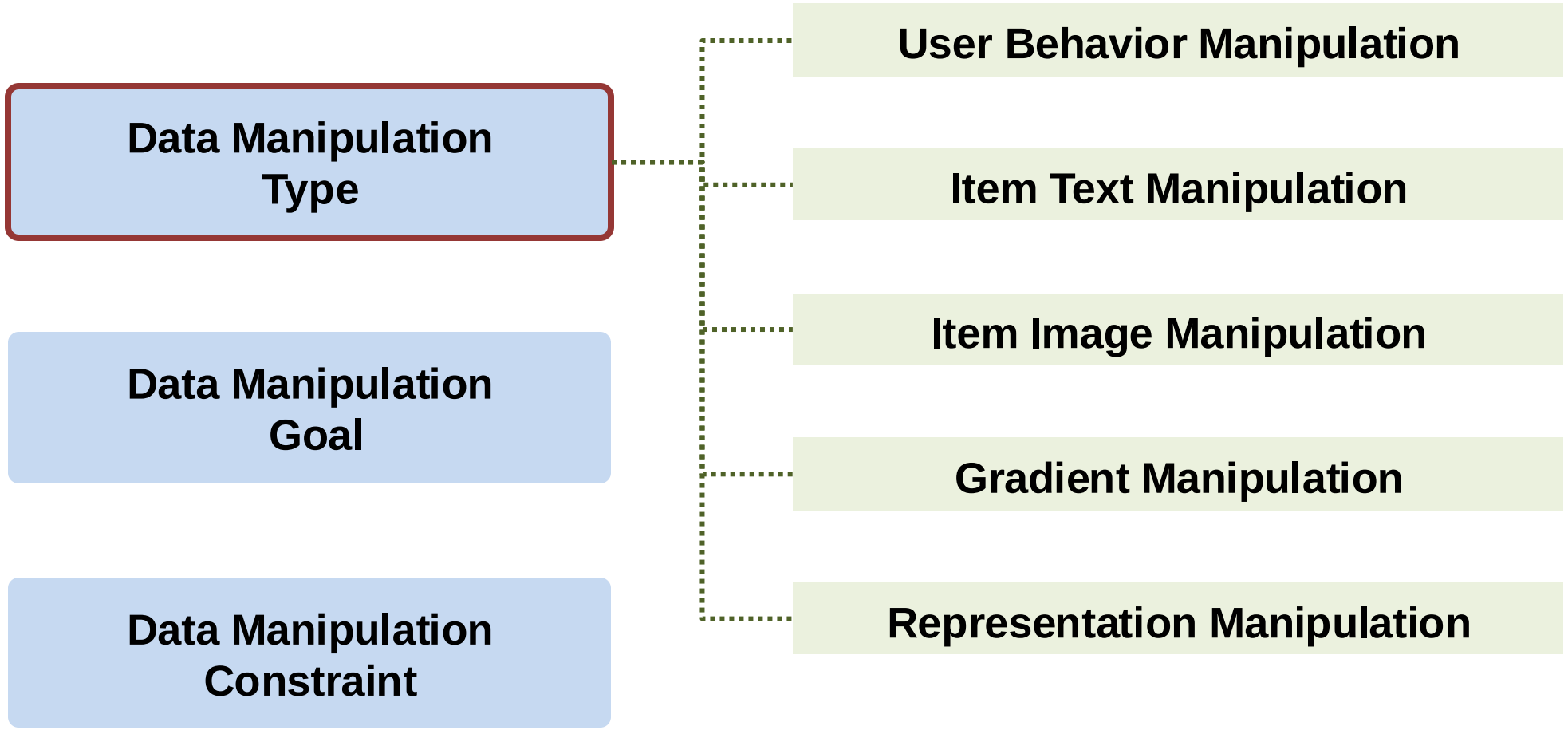
Key Dimensions of How Attackers Manipulate Data Integrity

Data Manipulation Type

Data Manipulation Goal

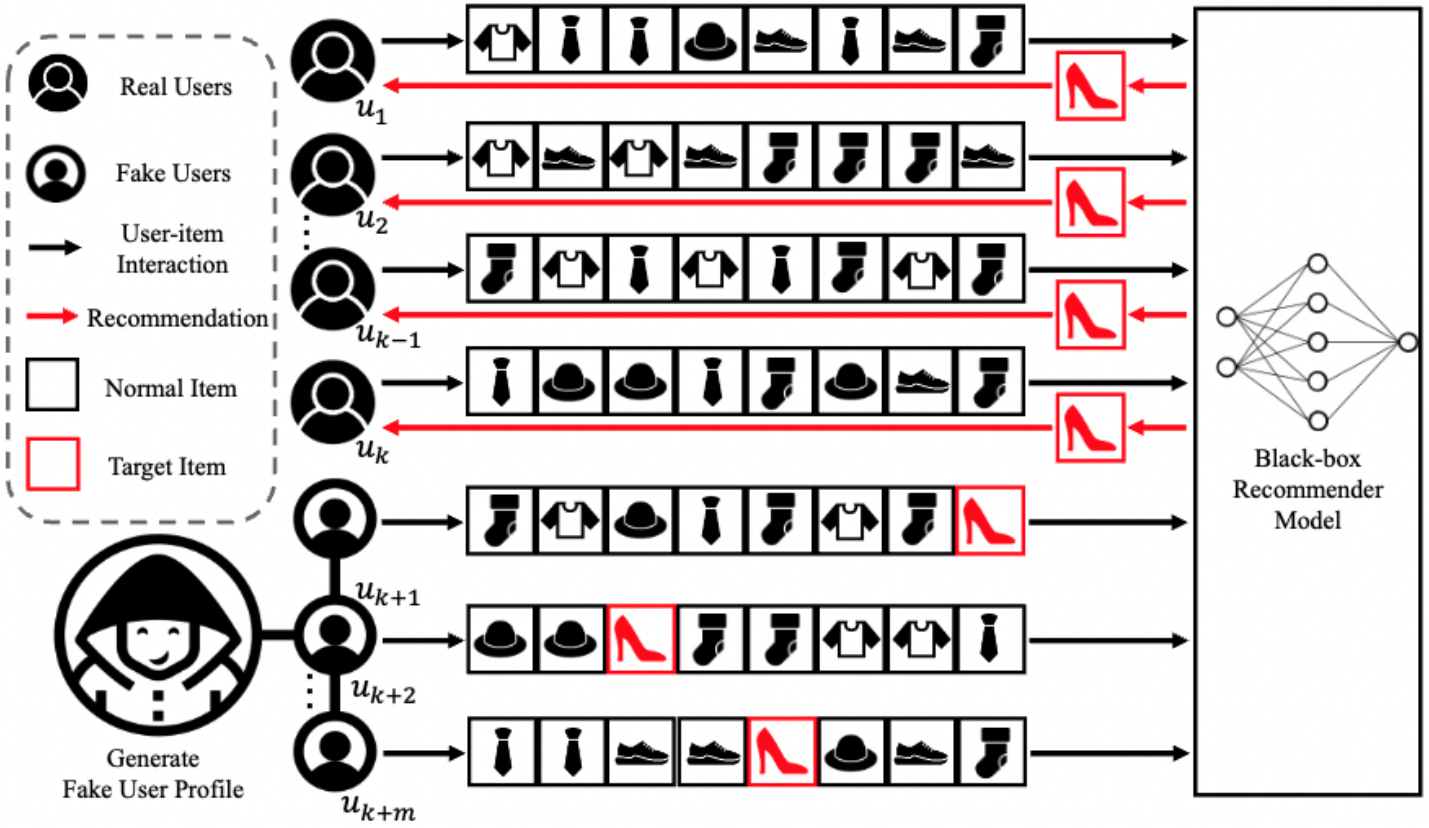
Data Manipulation Constraint

2.1 Manipulating Data Integrity



2.1.1 User Behavior Manipulation

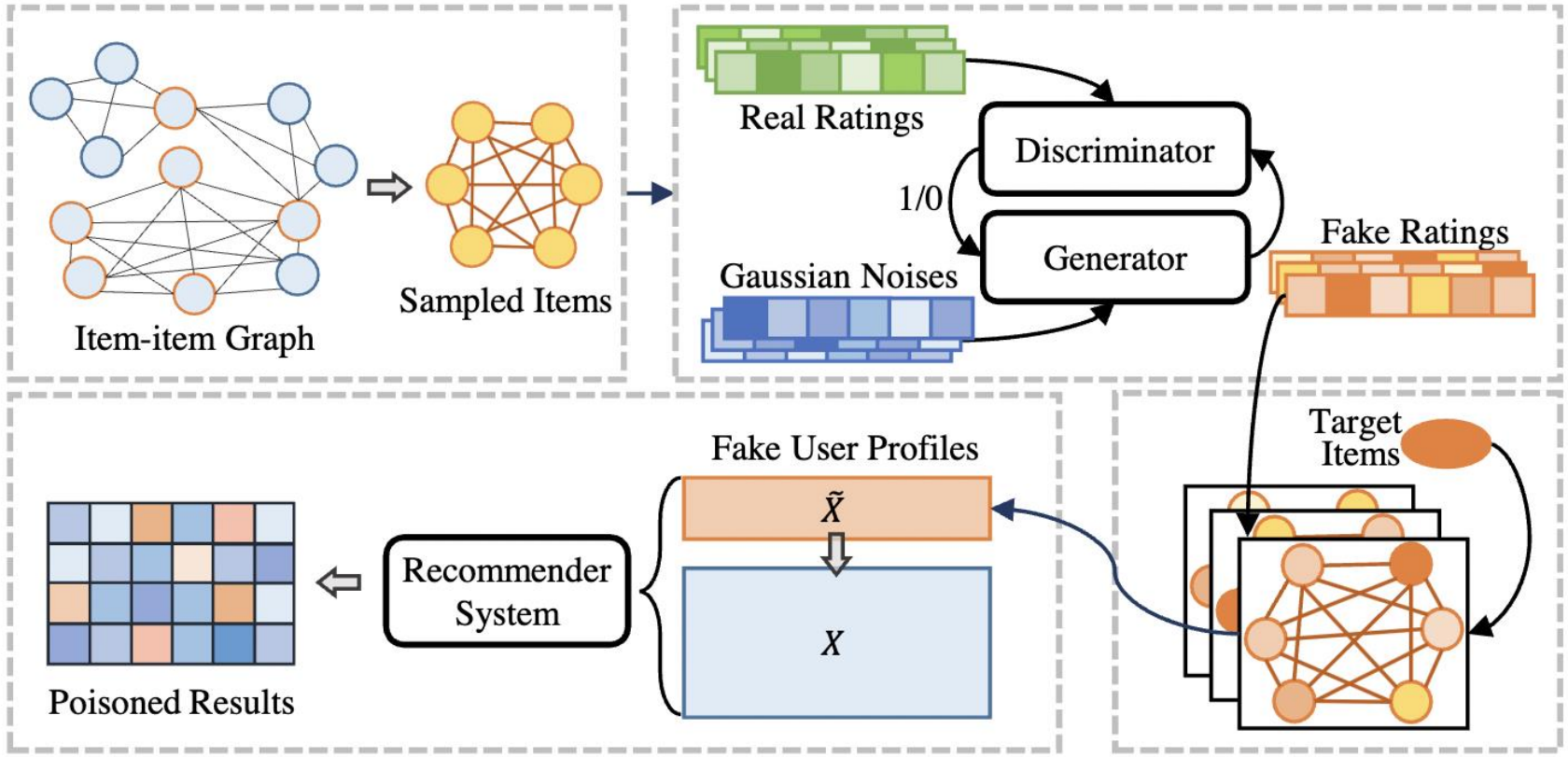
User Behavior Manipulation: Attackers manipulates real user accounts and inject fake behaviors (clicks, purchases, ratings).



Wang Y, Liu Y, Wang Q, et al. Poisoning self-supervised learning based sequential recommendations[C]//Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2023: 300-310.

2.1.1 User Behavior Manipulation

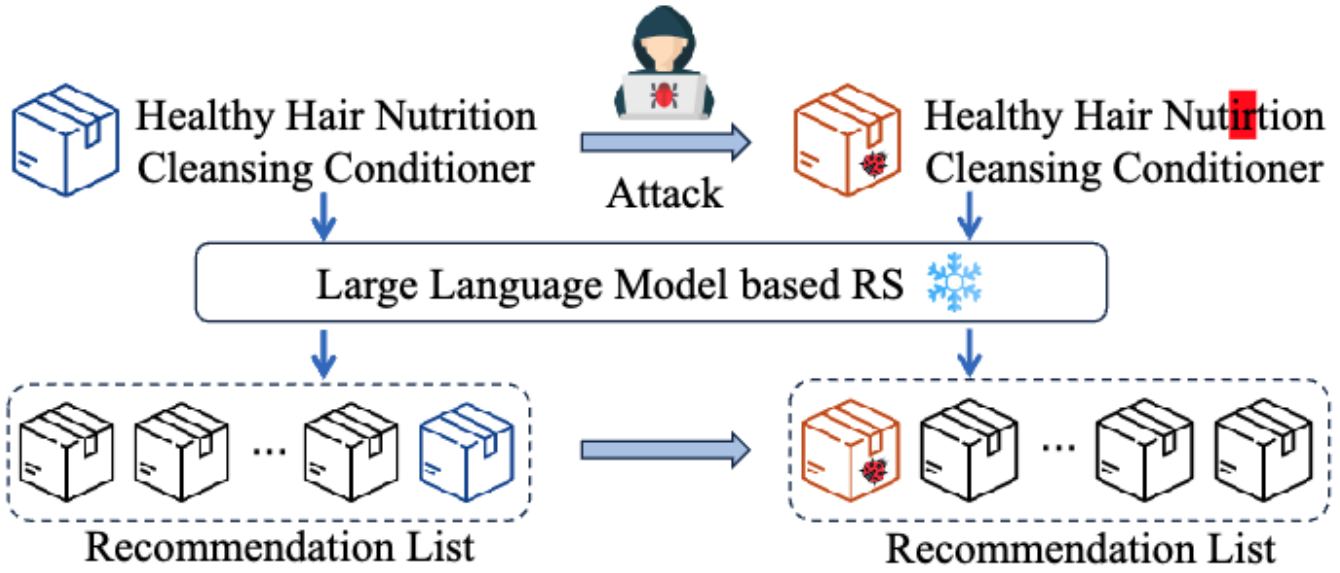
Example (Rating): GOAT performs poisoning attacks on the ratings of graph data.



Wu, Fan, et al. "Ready for emerging threats to recommender systems? A graph convolution-based generative shilling attack." Information Sciences 578 (2021): 683-701.

2.1.1 Item Text Manipulation

Item Text Manipulation: Attackers modify chars or sentences of the description of items.



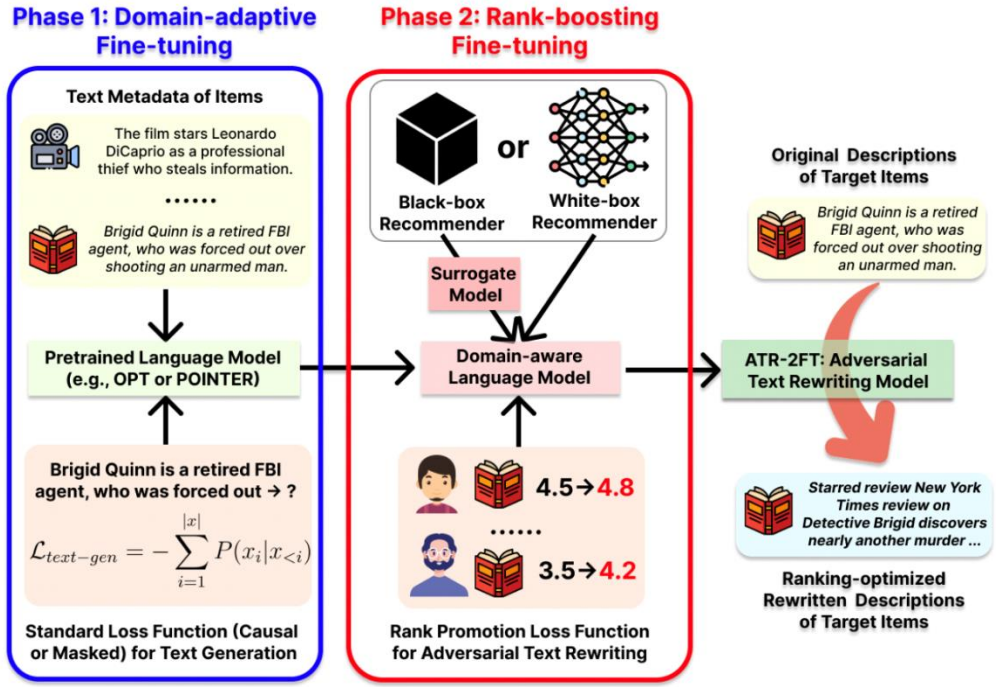
Zhang, Jinghao, et al. "Stealthy attack on large language model based recommendation." ACL 2024.

2.1.1 Item Text Manipulation

Example (Rewriting): TextRecAttack and ATR2FT employ text rewriting attack on text-aware recommendation.

Prompt 1: You are a marketing expert that helps to promote the product selling. Rewrite the product title in <MaxLen> words to keep its body the same but more attractive to customers: <ItemTitle>.
Prompt 2: Here is a basic title of a product. Use your creativity to transform it into a catchy and unique title in <MaxLen> words that could attract more attention: <ItemTitle>.
Prompt 3: Rewrite this product's title by integrating positive and appealing words, making it more attractive to potential users without altering its original meaning (in <MaxLen> words): <ItemTitle>.

TextRecAttack

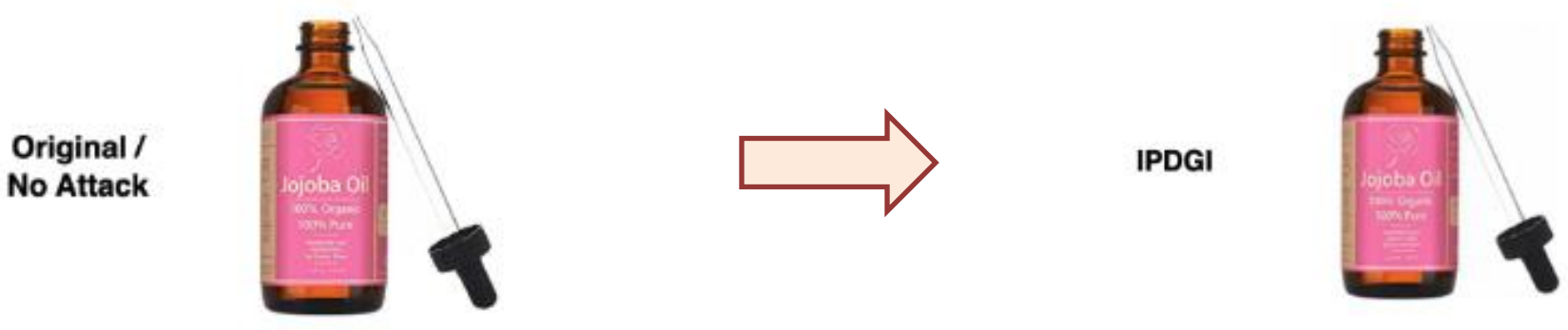


ATR2FT

Zhang, Jinghao, et al. "Stealthy attack on large language model based recommendation." ACL 2024.
Oh, Sejoon,et al. "Adversarial Text Rewriting for Text-aware Recommender Systems." Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. 2024.

2.1.1 Item Image Manipulation

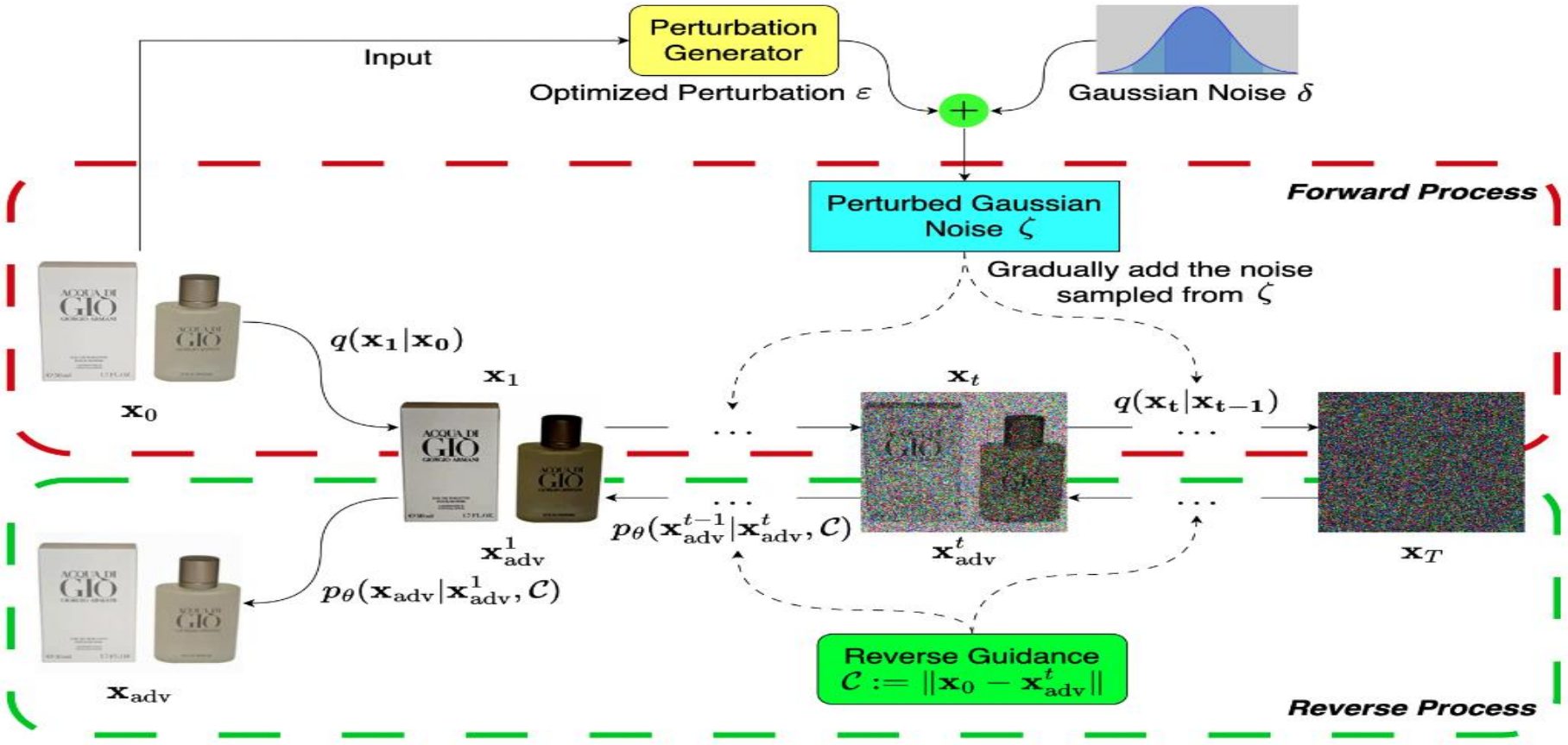
Item Image Manipulation: generates adversarial samples designed to promote the exposure rates of target items



Chen L, Yuan W, Chen T, et al. Adversarial item promotion on visually-aware recommender systems by guided diffusion[J]. ACM Transactions on Information Systems, 2024, 42(6): 1-26.

2.1.1 Item Image Manipulation

Example (Diffusion Model) : IPDGI generates adversarial samples designed to promote the exposure rates of target items

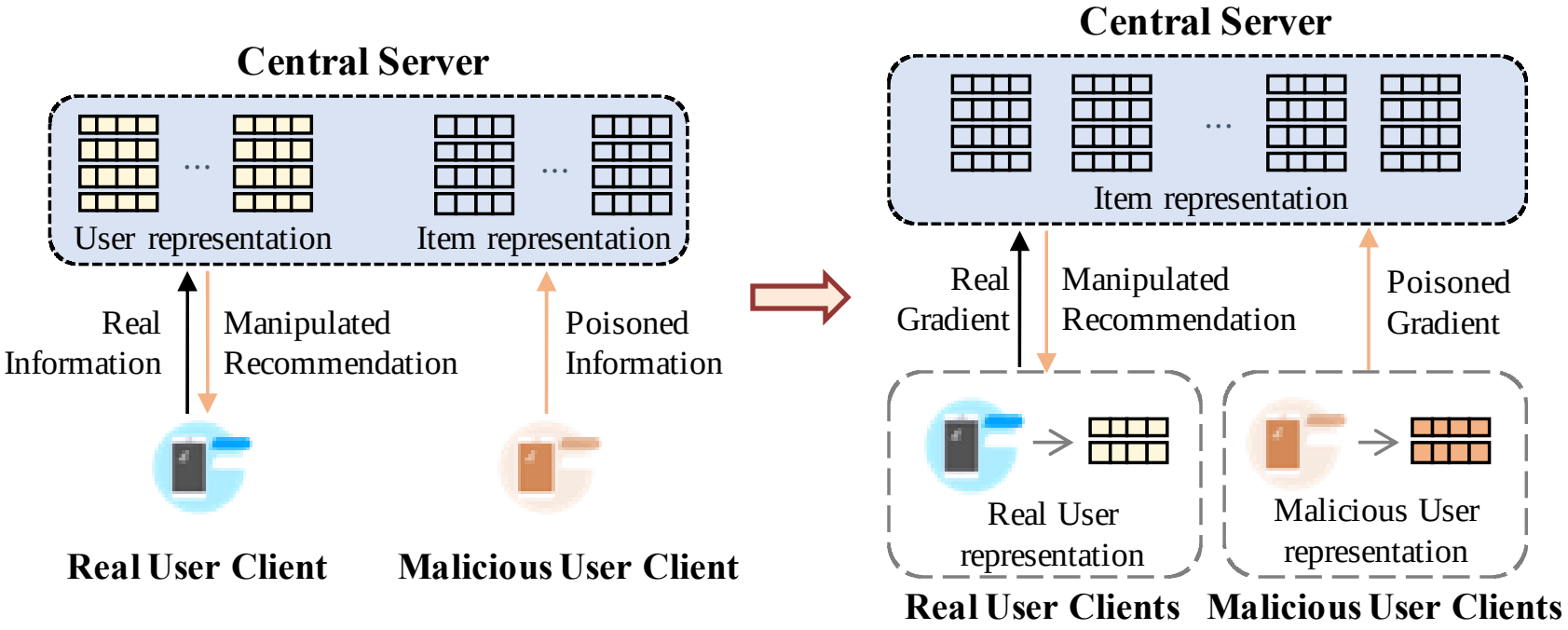


Add loss

Chen L, Yuan W, Chen T, et al. Adversarial item promotion on visually-aware recommender systems by guided diffusion[J]. ACM Transactions on Information Systems, 2024, 42(6): 1-26.

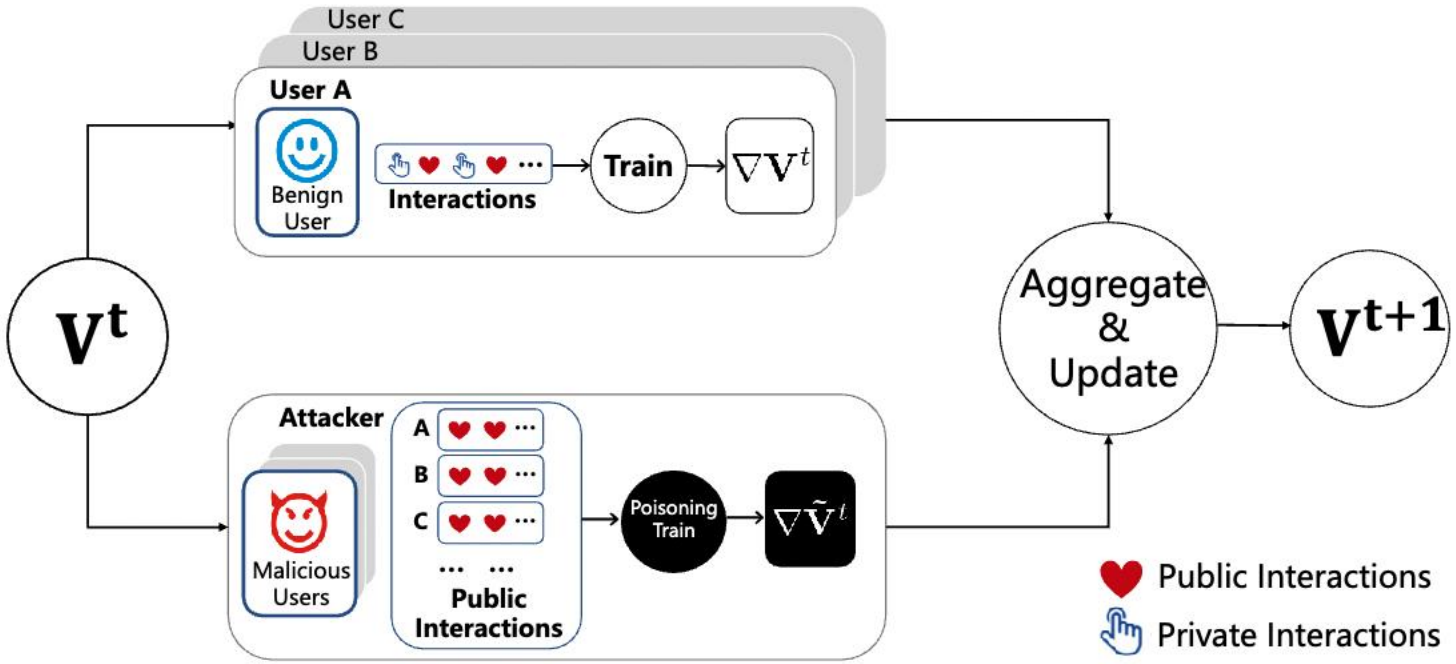
2.1.1 Gradient Manipulation

Gradient Manipulation: User Clients upload poisonous gradient into central server in decentralized recommendation (e.g., Federated RS) .



2.1.1 Data Manipulation Type

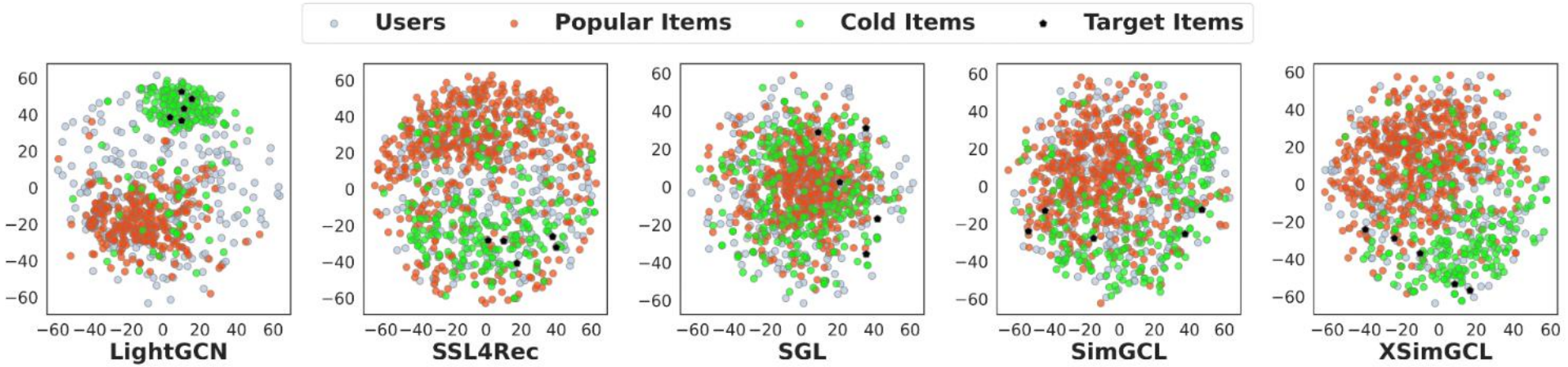
Example(Gradient Aggregation): FedRecAttack controls user clients uploads poisonous gradient into central server.



Rong, Dazhong, et al. "Fedrecattack: Model poisoning attack to federated recommendation." 2022 IEEE 38th International Conference on Data Engineering (ICDE). IEEE, 2022.

2.1.1 Representation Manipulation

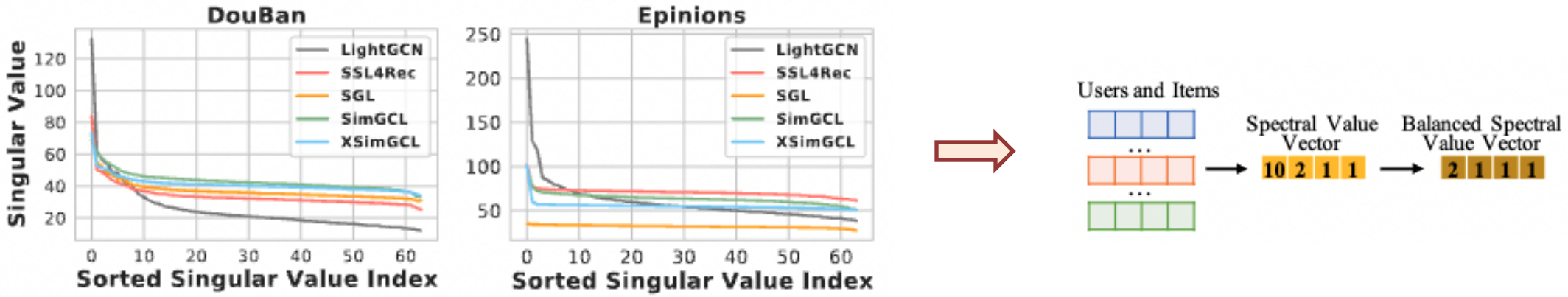
Representation Manipulation: Attackers control the user/items embedding representation to achieve goal.



Wang, Zongwei, et al. "Unveiling Vulnerabilities of Contrastive Recommender Systems to Poisoning Attacks." Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2024.

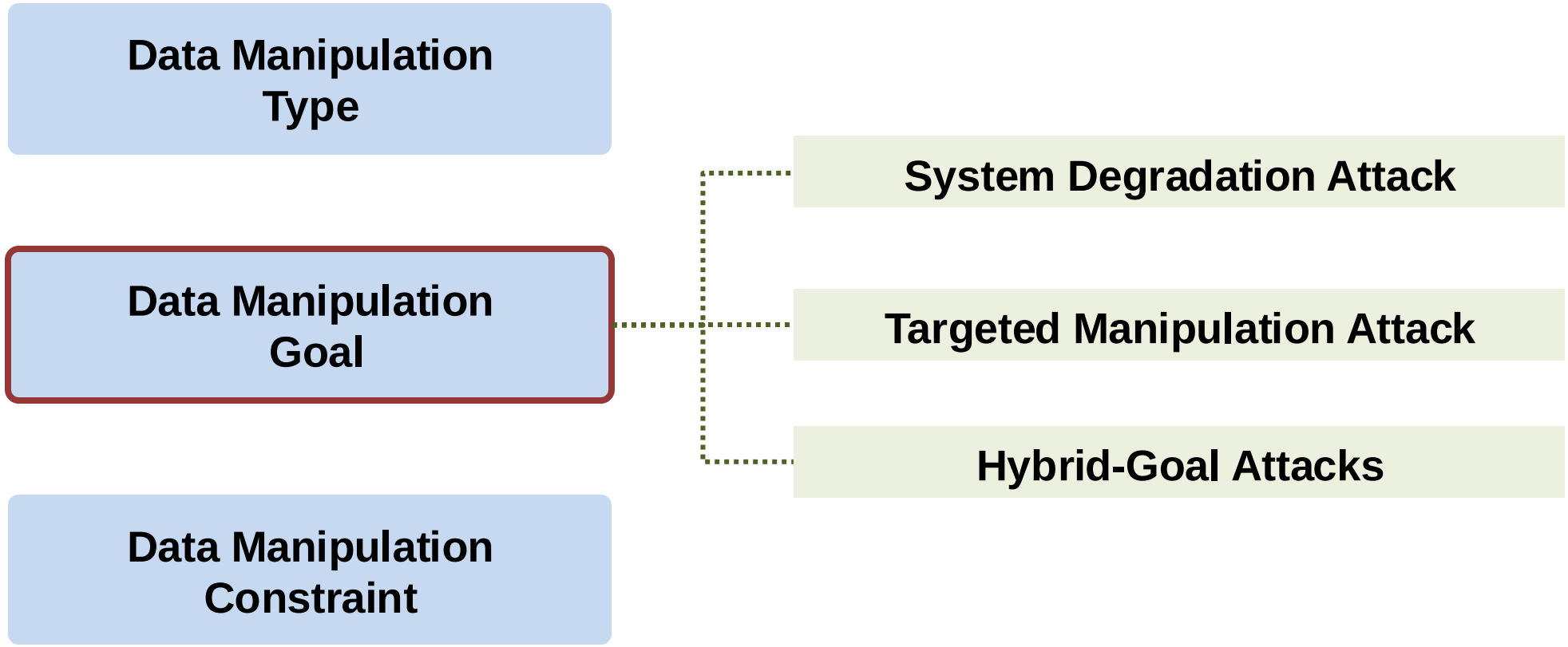
2.1.1 Representation Manipulation

Example (Representation Spectral Values): CLeaR controls the the spectral value of user/items embedding representation into a smooth distribution.



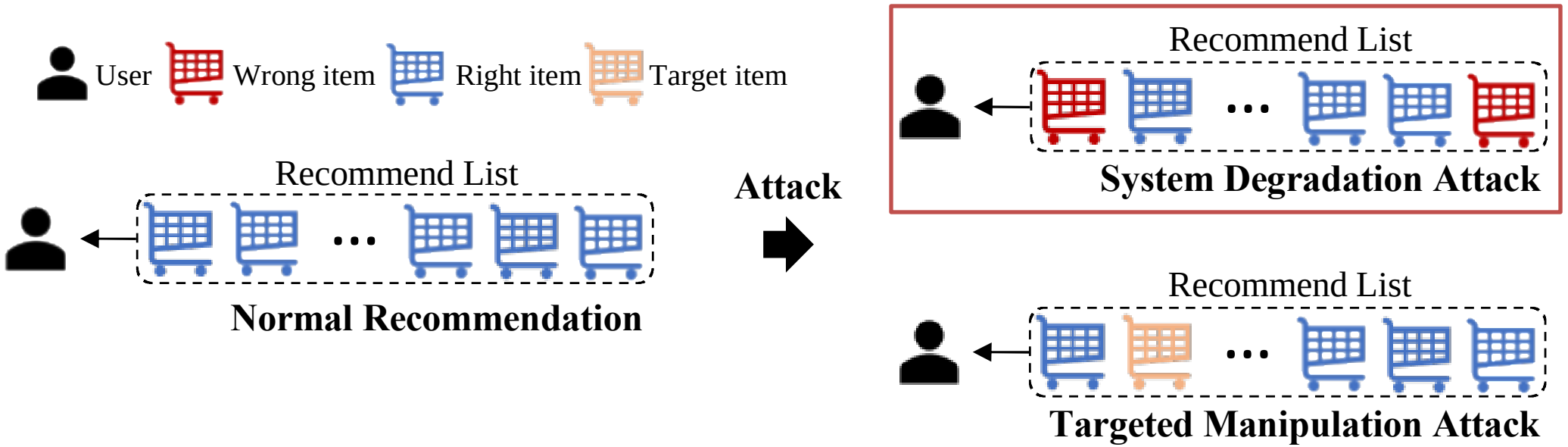
Wang, Zongwei, et al. "Unveiling Vulnerabilities of Contrastive Recommender Systems to Poisoning Attacks." Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2024.

2.1 Manipulating Data Integrity



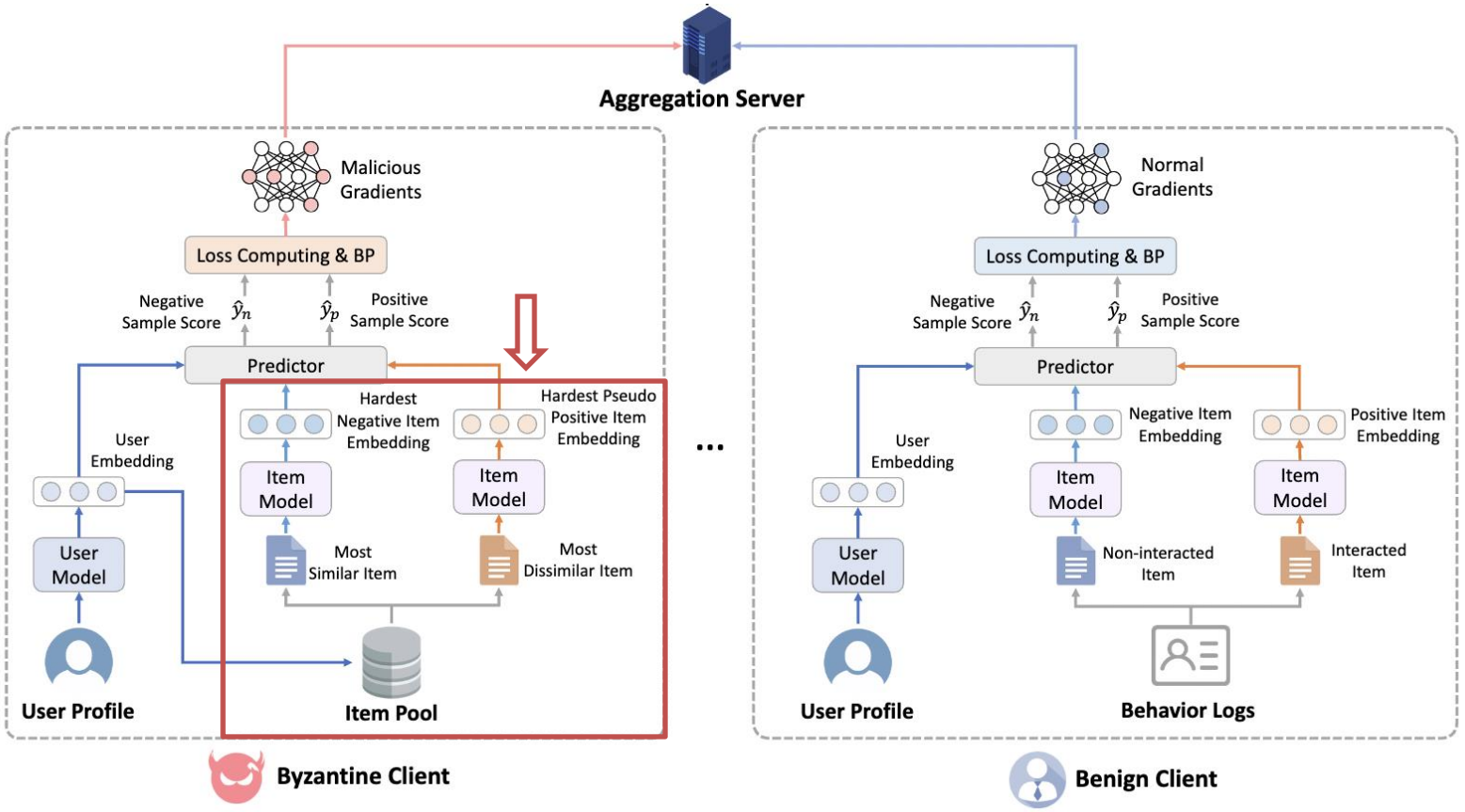
2.1.2 System Degradation Attack

System Degradation Attack (a.k.a untargeted attack), is crafted with the broad objective of undermining the entire system performance.



2.1.2 System Degradation Attack

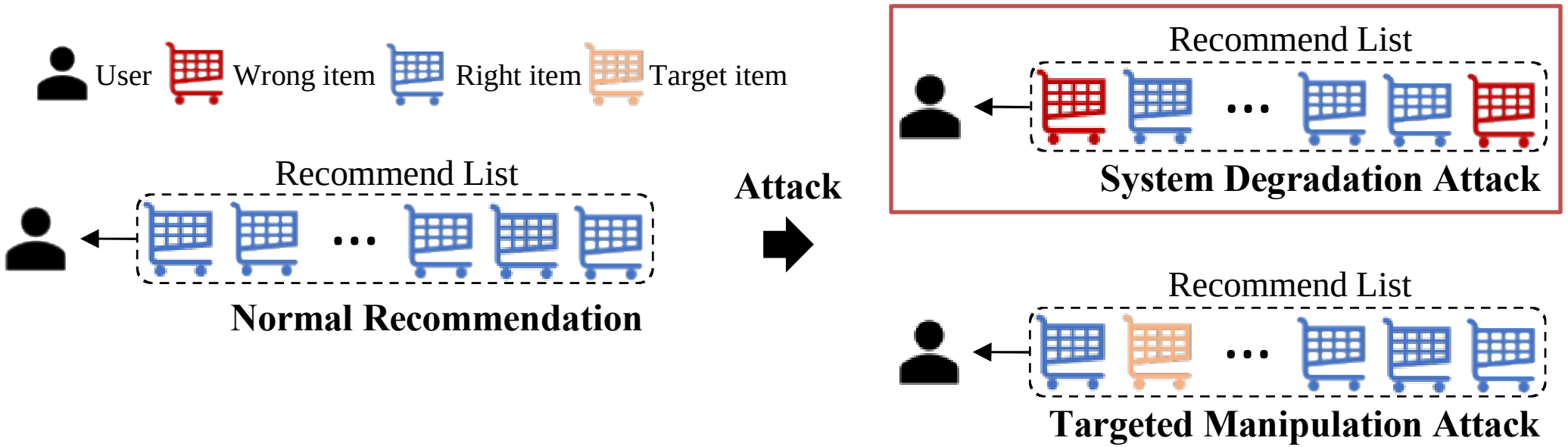
Example（Hardest Sample）：FedAttack selects items most and least relevant to user embeddings as the hardest negative and positive samples.



Wu, Chuhan, et al. "FedAttack: Effective and covert poisoning attack on federated recommendation via hard sampling." Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2022.

2.1.2 System Degradation Attack

System Degradation Attack (a.k.a untargeted attack), is crafted with the broad objective of undermining the entire system performance.



2.1.2 Targeted Manipulation Attack

Two Common Ways of Targeted Manipulation Attacks

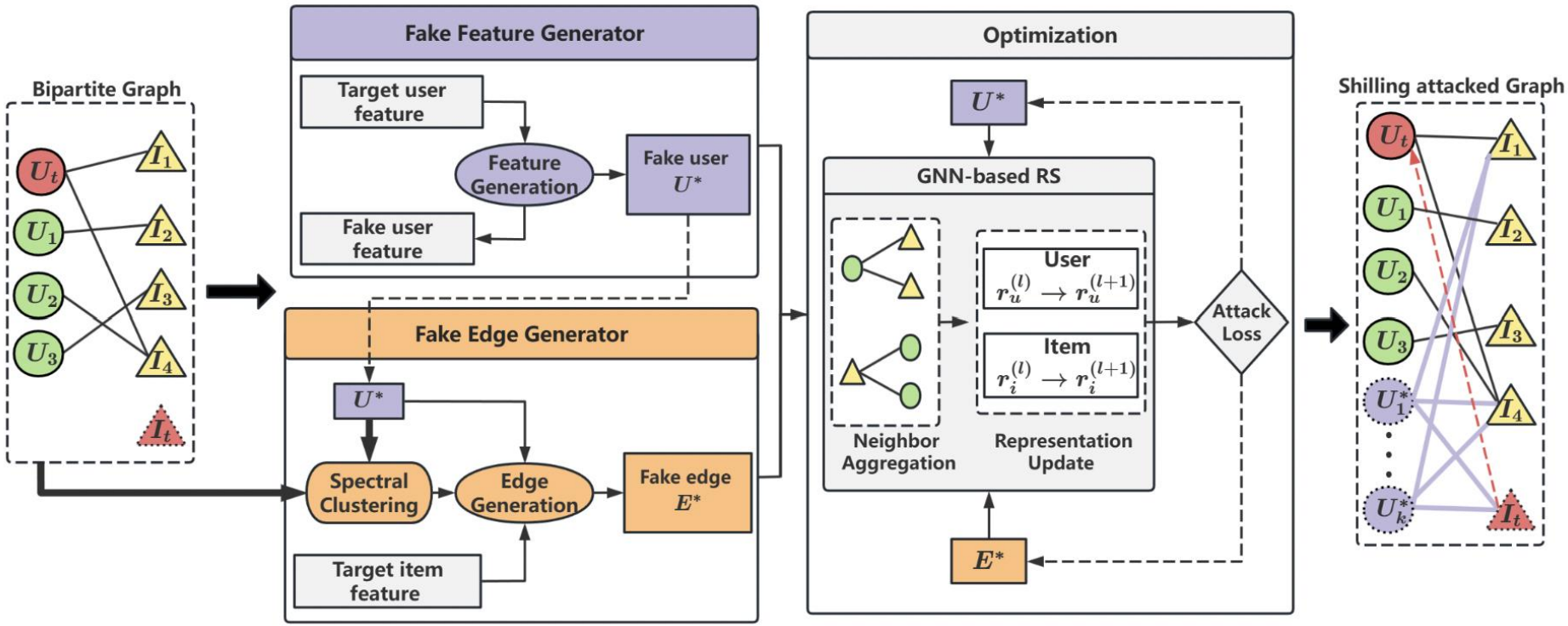
Example1 (Visibility of Target Items): PoisonRec focuses on maximizing the frequency of targeted items appearing in top-K recommendation lists.

Example2 (Close to User Preferred Items): GTA evaluates user intent by predicting the user's most preferred item and then adjusts the target item to closely resemble these favored items.

Song, Junshuai, et al. "Poisonrec: an adaptive data poisoning framework for attacking black-box recommender systems." 2020 IEEE 36th international conference on data engineering (ICDE). IEEE, 2020.
Wang, Zhiye, et al. "Revisiting data poisoning attacks on deep learning based recommender systems." 2023 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2023.

2.1.2 Targeted Manipulation Attack

Example (Close to Targeted User Group): AutoAttack crafts malicious profiles that closely mimic the targeted user group. It enables an influence on the designated group while concurrently minimizing effects on other users.



Guo, Sihan, Ting Bai, and Weihong Deng. "Targeted shilling attacks on gnn-based recommender systems." Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 2023.

2.1.2 Hybrid-Goal Attacks

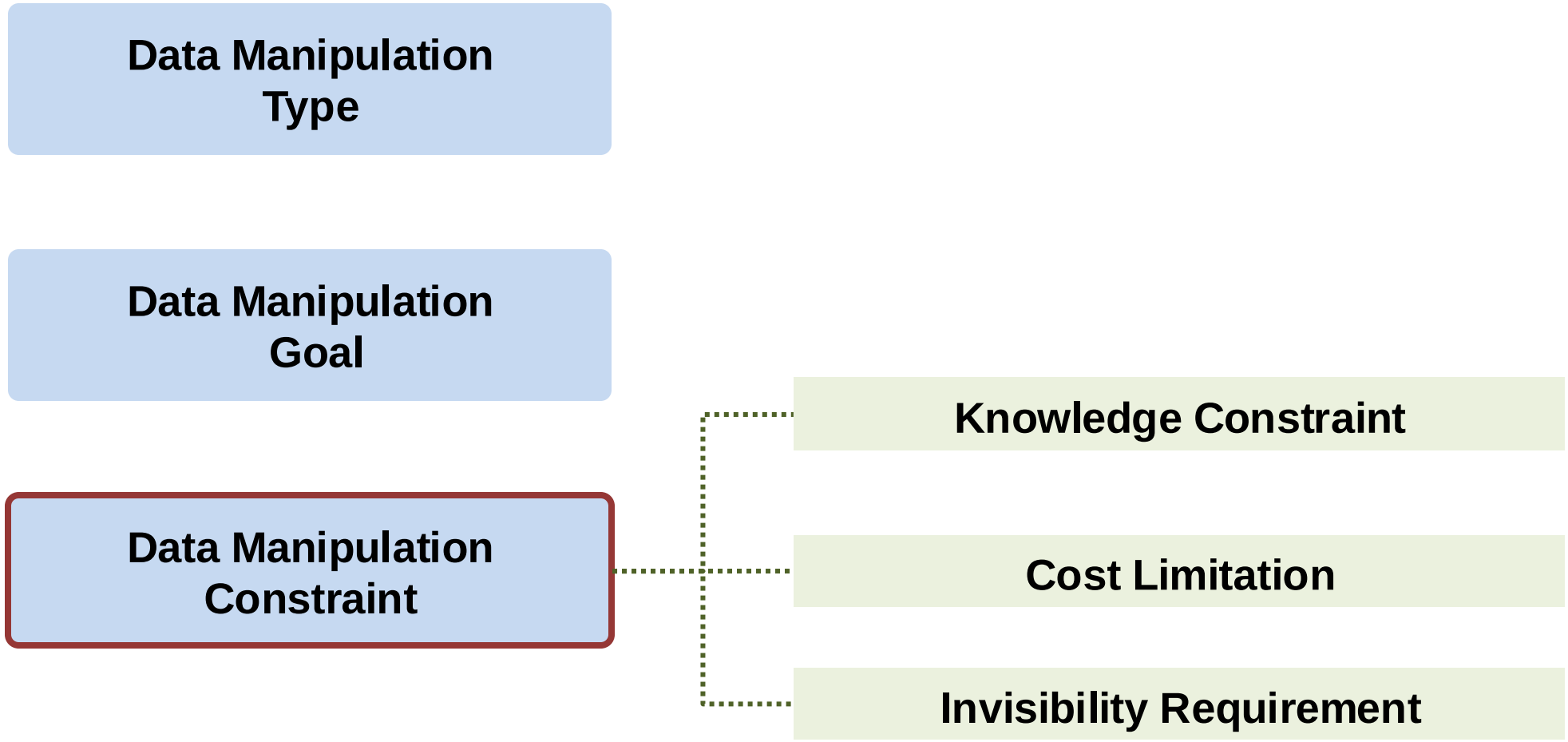
Hybrid-Goal Attack: Some attacks achieve a balance between these two goals using weighted coefficients, such as **SGLD, CD-Attack**.

$$\mathrm{R} = \alpha \mathrm{R}_{\mathrm{target}} + \beta \mathrm{R}_{\mathrm{untarget}}$$

The potential for hybrid attacks has not been fully explored, which does not fully represent the diversity of real-world attackers' intentions.

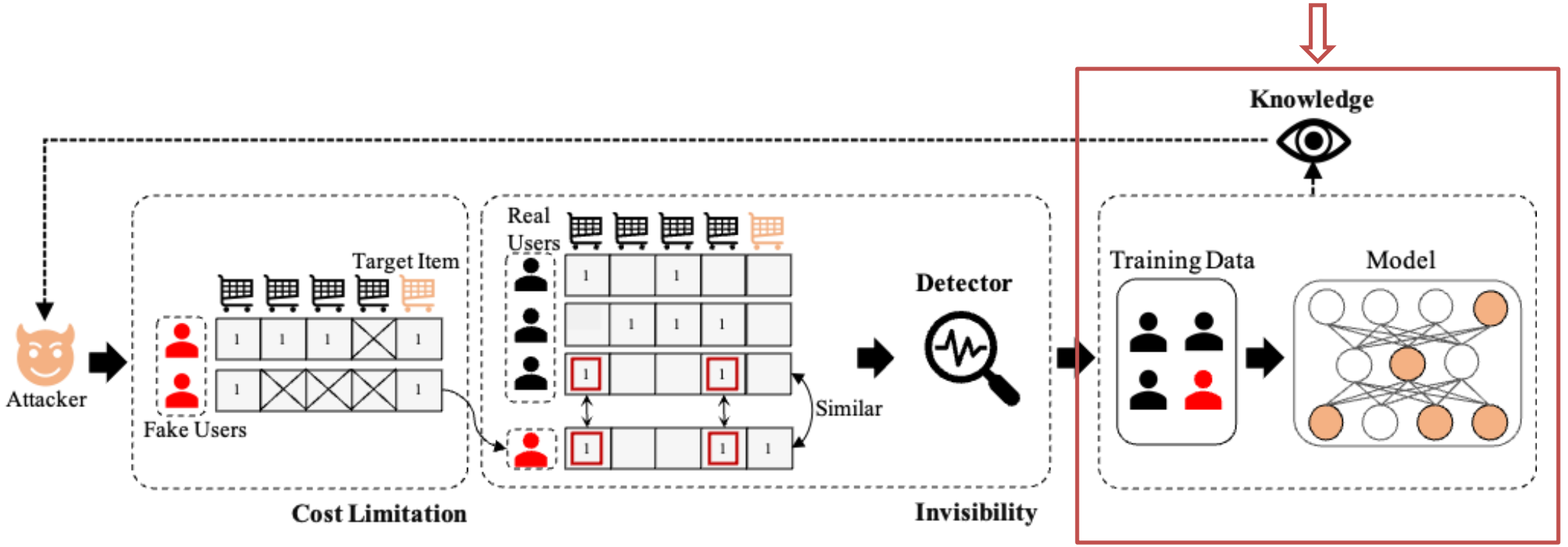
Li, Bo, et al. "Data poisoning attacks on factorization-based collaborative filtering." Advances in neural information processing systems 29 (2016).
Chen, Huiyuan, and Jing Li. "Data poisoning attacks on cross-domain recommendation." Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019.

2.1 Manipulating Data Integrity



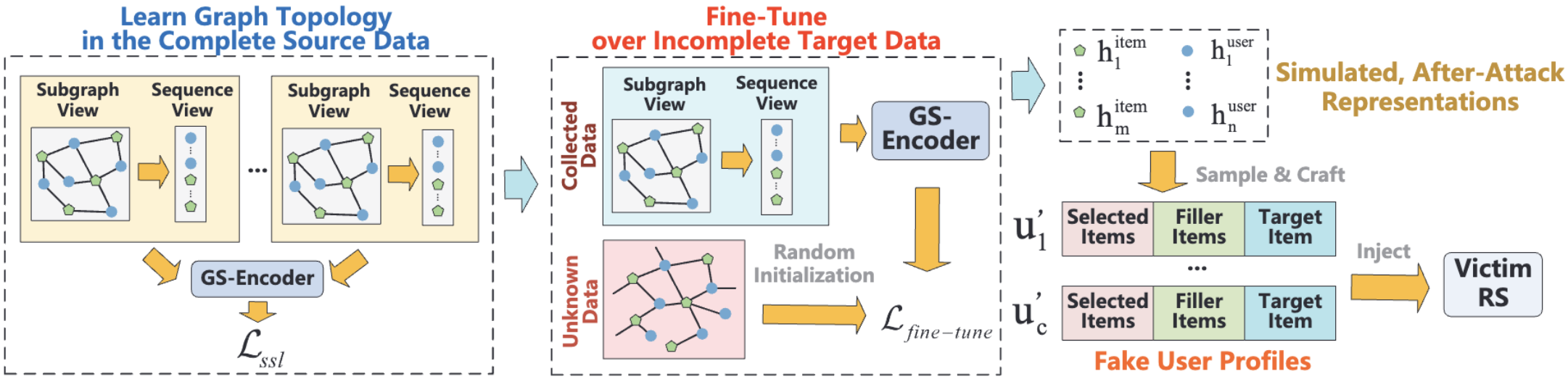
2.1.3 Knowledge Constraint

Knowledge Constraint: Attackers typically view RS as complex 'black boxes,' with limited transparency regarding the data used for training and the specifics of the underlying algorithms.



2.1.3 Knowledge Constraint

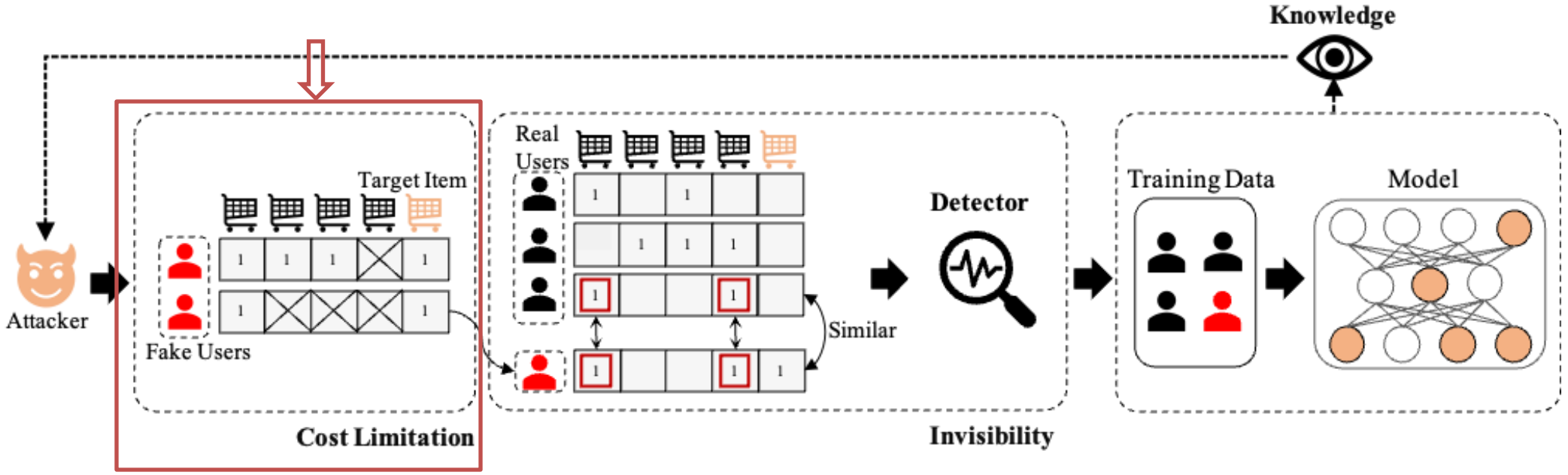
Example (Lack of Data Knowledge): PCAttack pre-trains a simulator model on multiple data sources. A small portion of the target data is then input into the simulator model for finetuning.



Zeng, Meifang, et al. "Practical cross-system shilling attacks with limited access to data." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 37. No. 4. 2023.

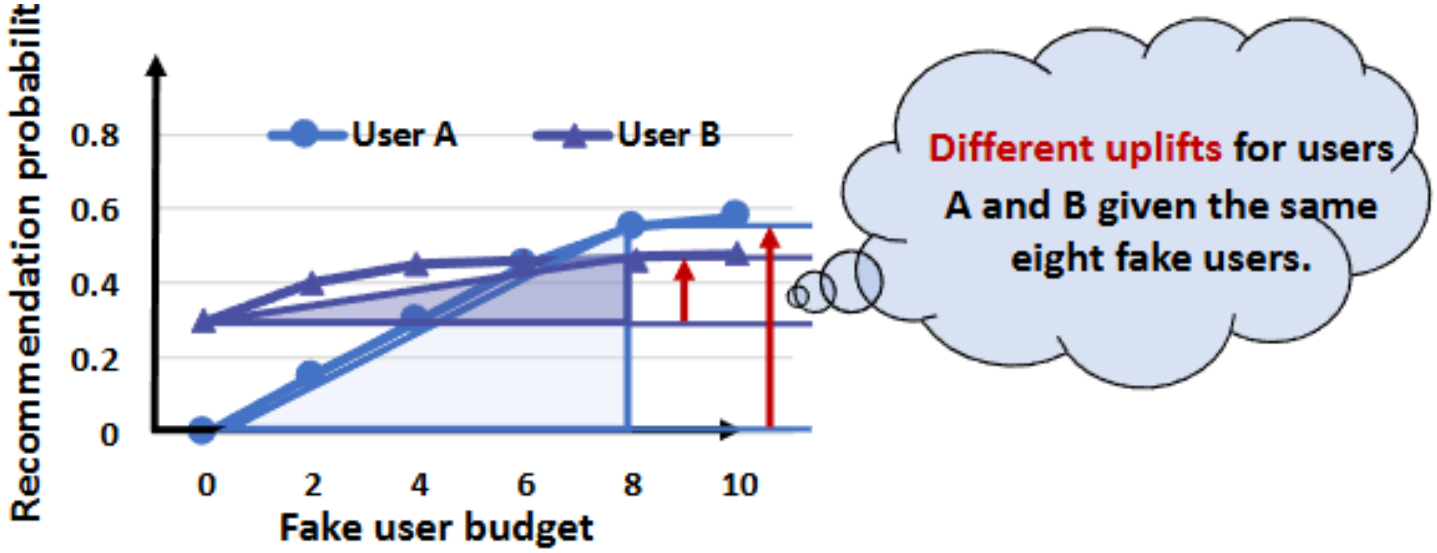
2.1.3 Cost Limitation

Cost Limitation: The insertion of poisoned data into the system requires a cost-effective analysis to ensure economic viability.



2.1.3 Cost Limitation

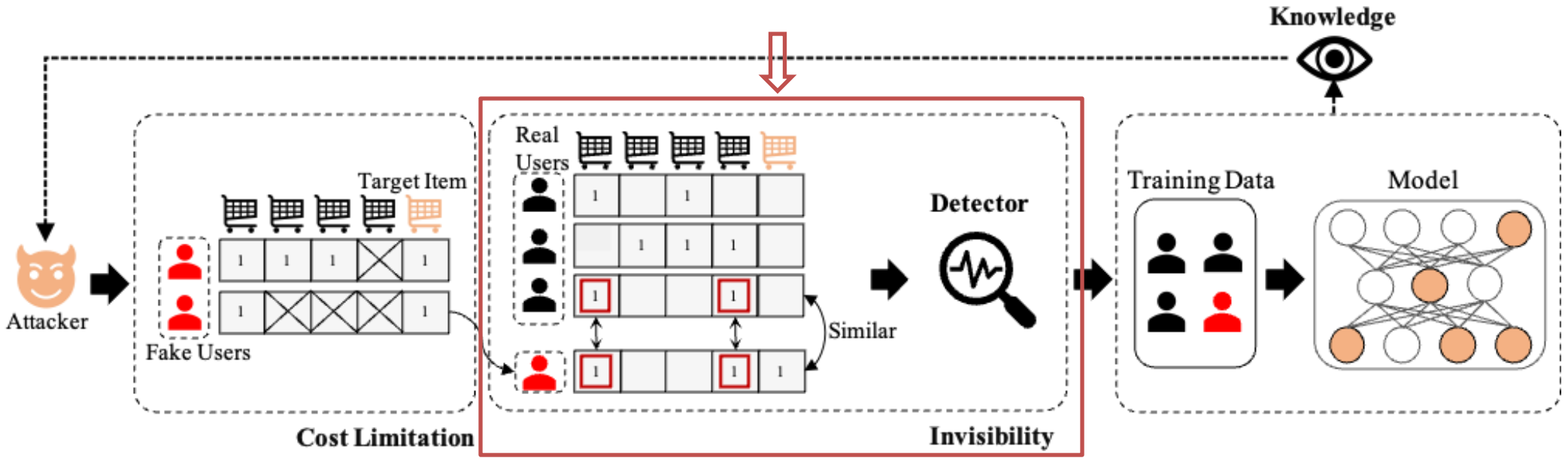
Example (Injection Budgets): UBA uses uplift model to estimate the treatment effect on each target user and optimizes the allocation of fake user budgets to maximize the attack performance.



Wang, Wenjie, et al. "Uplift Modeling for Target User Attacks on Recommender Systems." *Proceedings of the ACM on Web Conference 2024*. 2024.

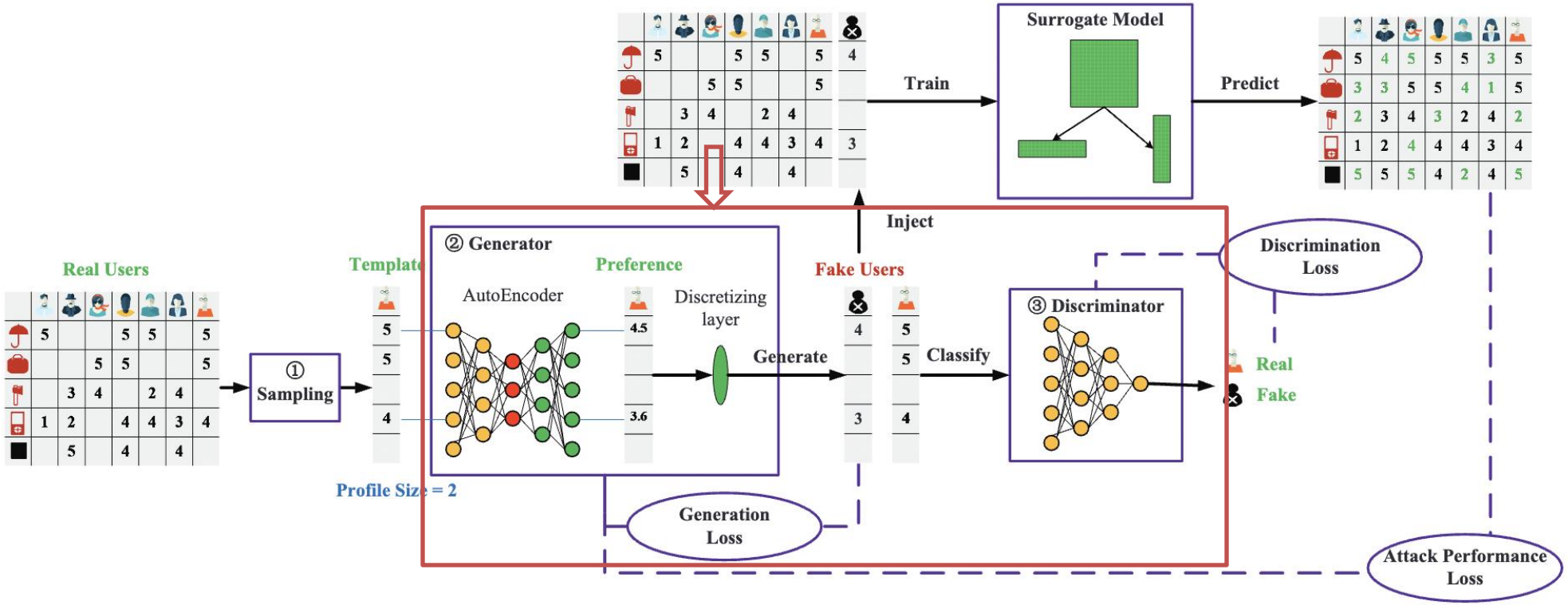
2.1.3 Invisibility Requirement

Invisibility Requirement: Poisoned data should be intricately crafted to blend seamlessly with legitimate data, making it challenging for the typical defense mechanisms within RS to detect the anomalies.



2.1.3 Invisibility Requirement

Example (GANS) : AUSH uses generative adversarial networks enhance the resemblance of generated user profiles to real user profiles.



Lin, Chen, et al. "Shilling black-box recommender systems by learning to generate fake user profiles." IEEE Transactions on Neural Networks and Learning Systems 35.1 (2022): 1305-1319.

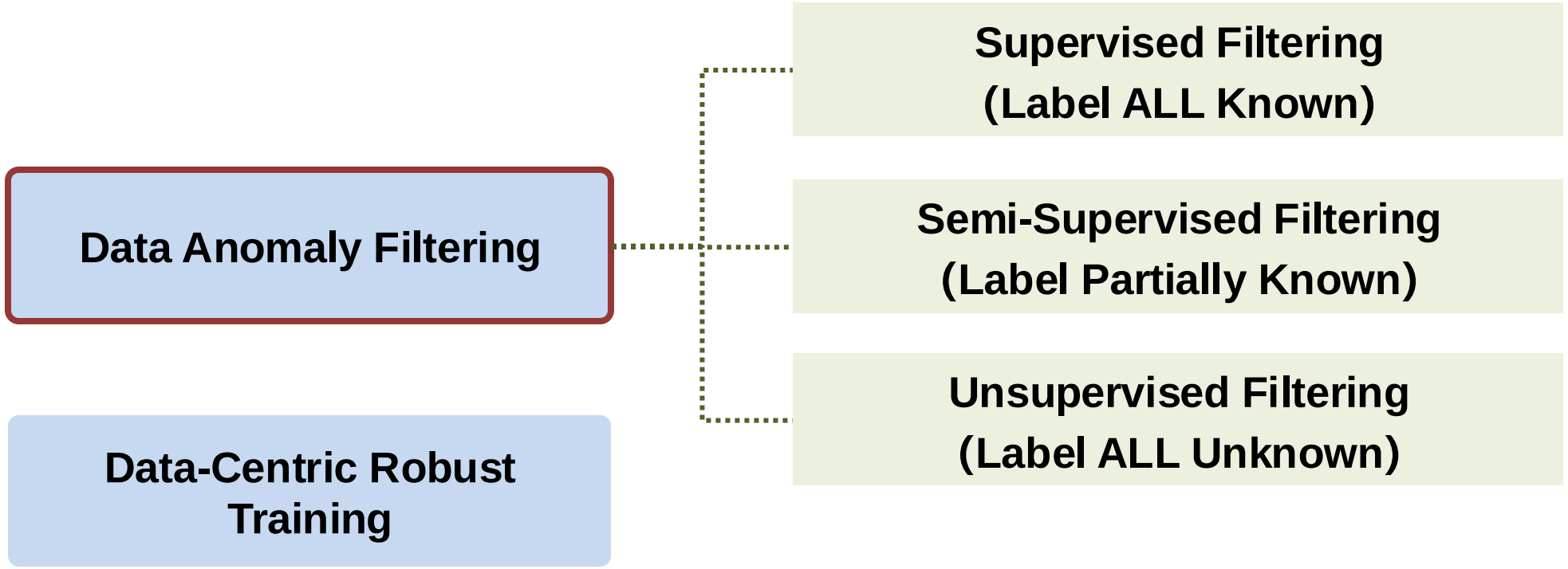
2.2 Data Integrity Protection

Two Stages of Data Integrity Protection

Data Anomaly Filtering

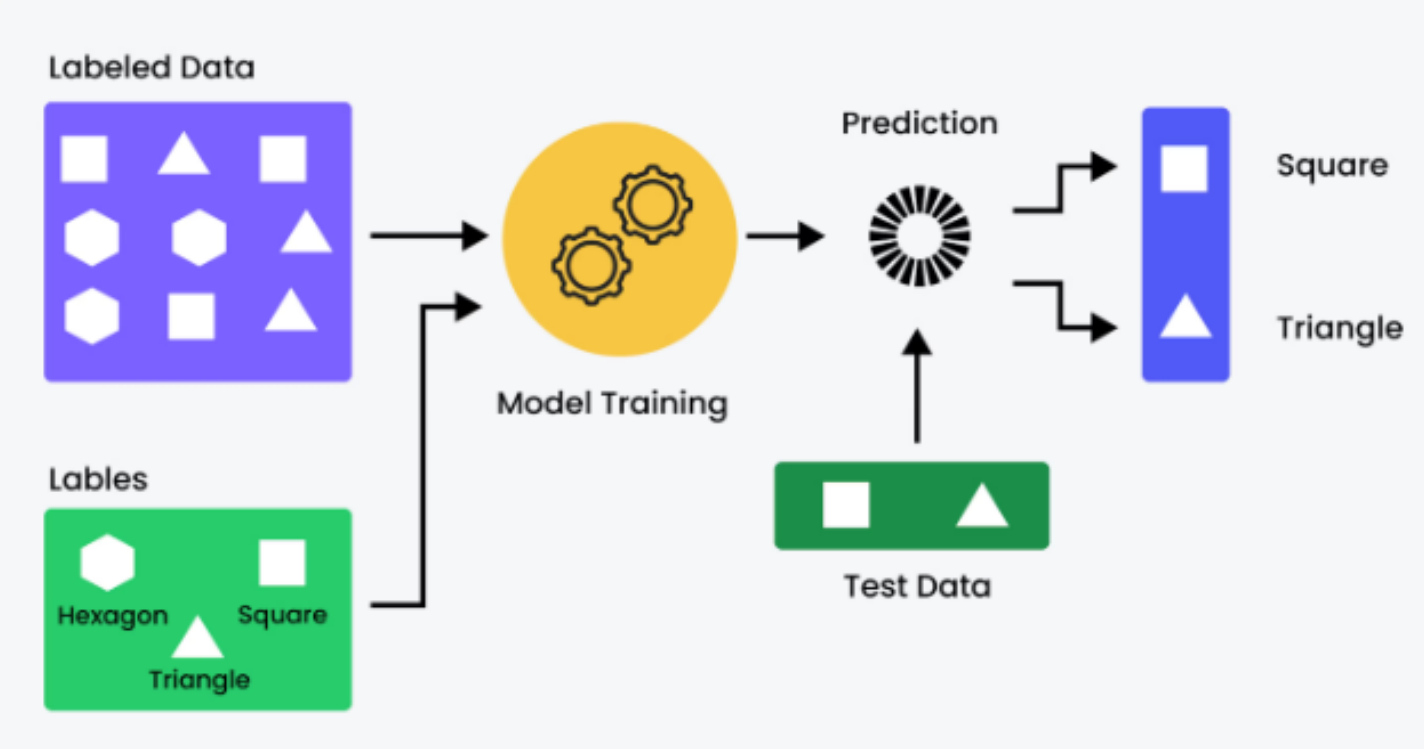
Data-Centric Robust Training

2.2 Protecting Data Integrity



2.2.1 Supervised Filtering

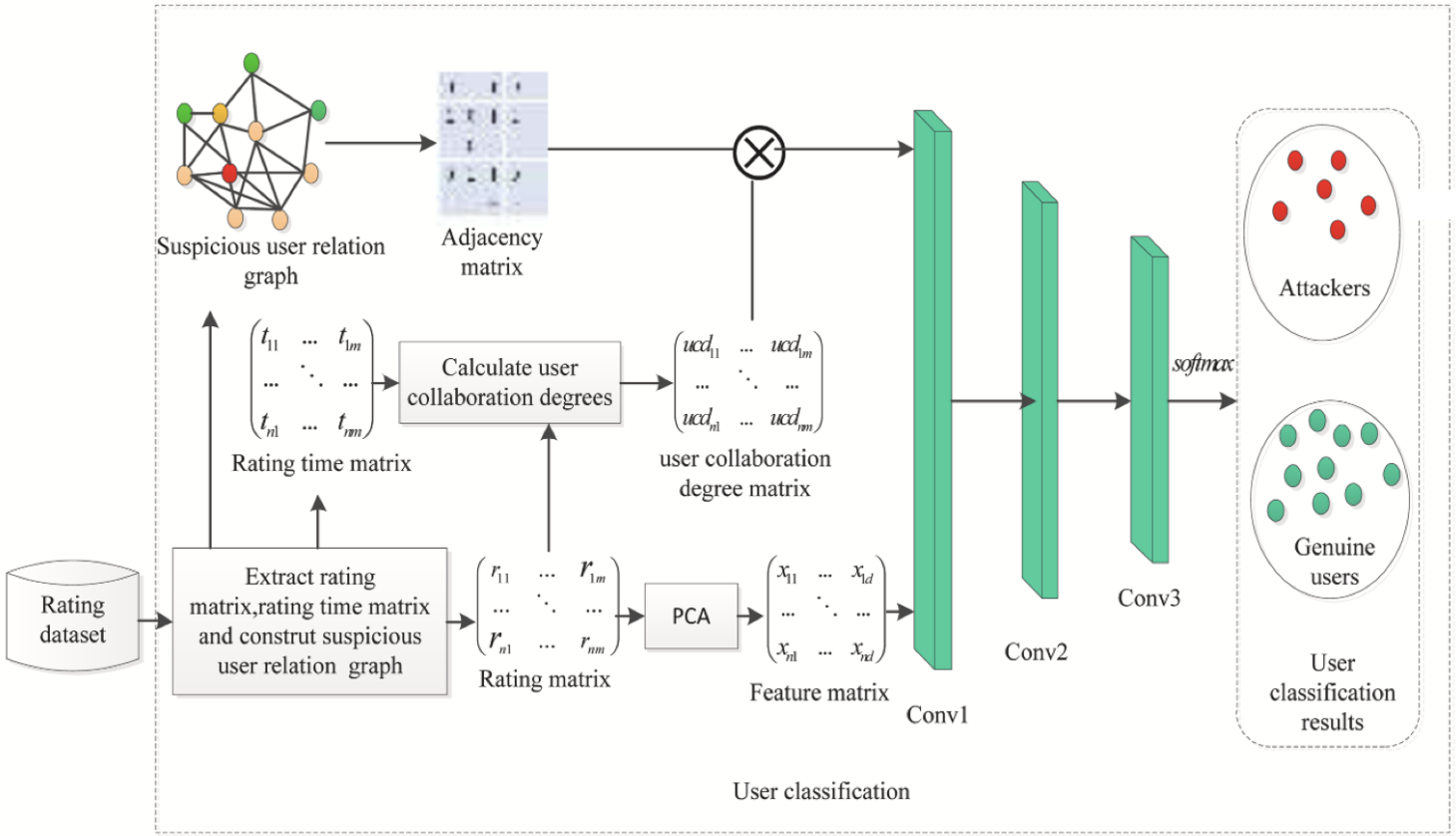
Supervised Filtering (Label ALL Known): Defenders leverage labeled historical data to train models that can discern and isolate malicious entries based on learned patterns.



https://maddevs.io/blog/supervised-learning-explained/

2.2.1 Supervised Filtering

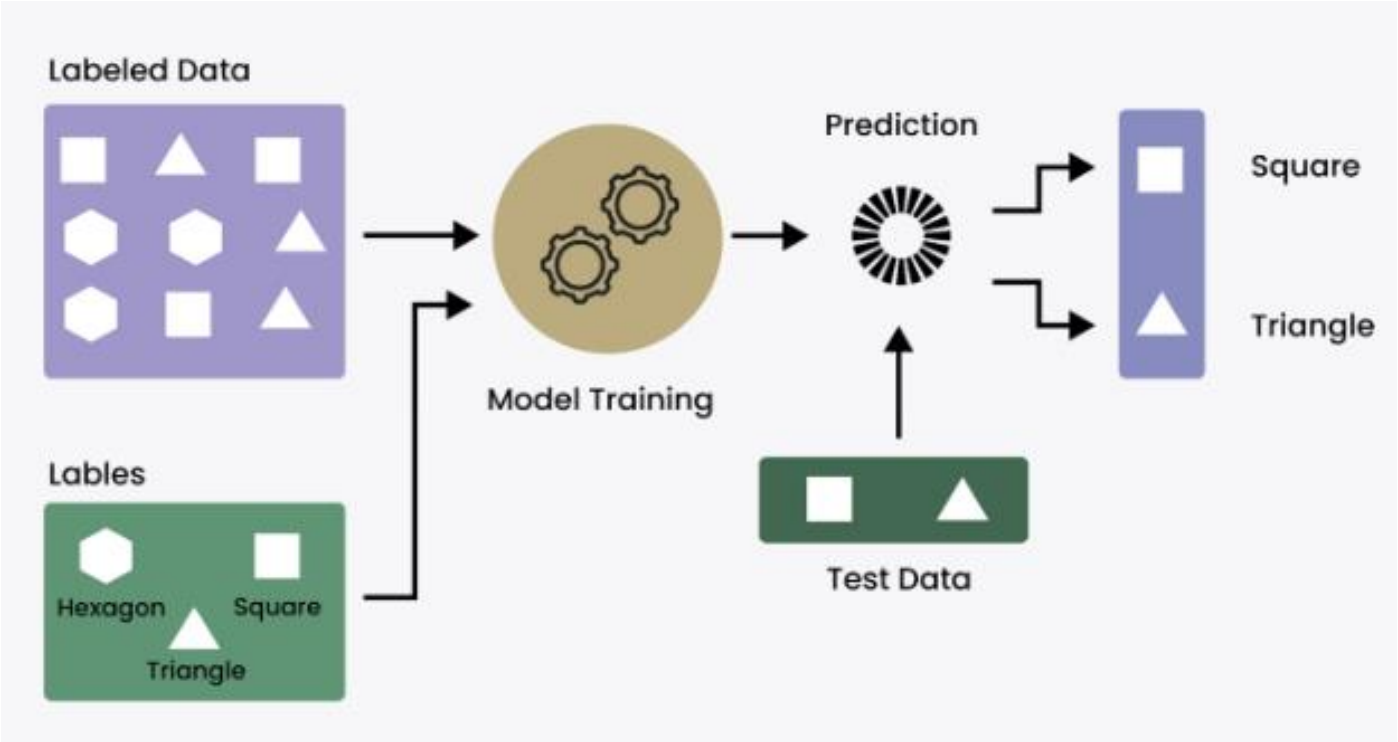
Example (GCN): DHAGCN combines user behavior features with GCN to detect hybrids of model-generative and group attacks.



Shilei Wang, Peng Zhang, Hui Wang, Hongtao Yu, and Fuzhi Zhang. 2022. Detecting shilling groups in online recommender systems based on graph convolutional network. Information Processing & Management 59, 5 (2022), 103031.

2.2.1 Semi-Supervised Filtering

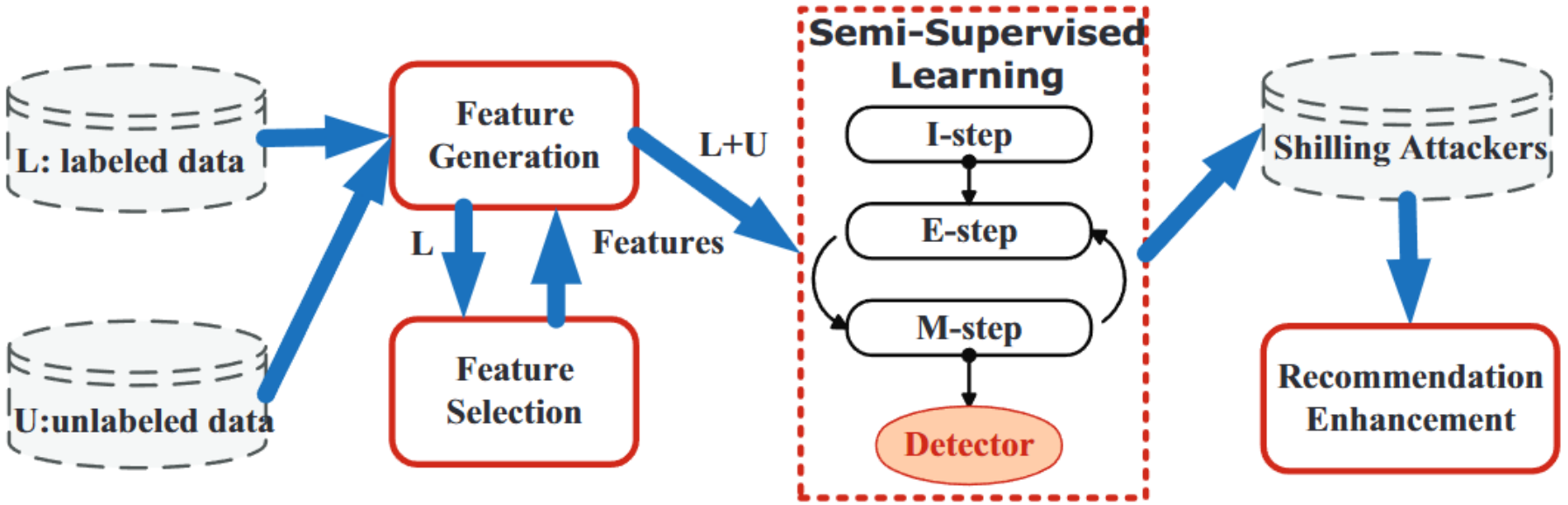
Semi-Supervised Filtering (Label Partially Known): Defenders leverage labeled historical data to train models that can discern and isolate malicious entries based on learned patterns.



https://maddevs.io/blog/supervised-learning-explained/

2.2.1 Semi-Supervised Filtering

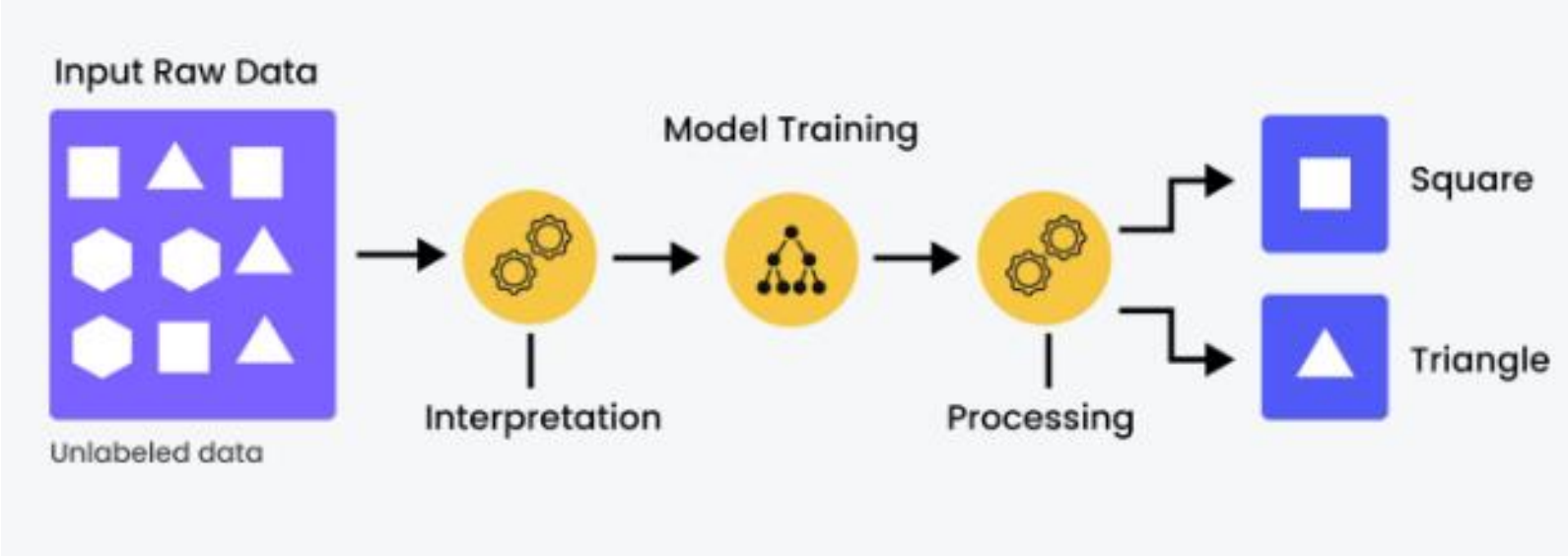
Example (Bayesian) : HySAD introduces MC-Relief to select effective detection metrics, and uses EM algorithm to precisely separate attackers.



Wu, Zhiang, et al. "HySAD: A semi-supervised hybrid shilling attack detector for trustworthy product recommendation." Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. 2012.

2.2.1 Unsupervised Filtering

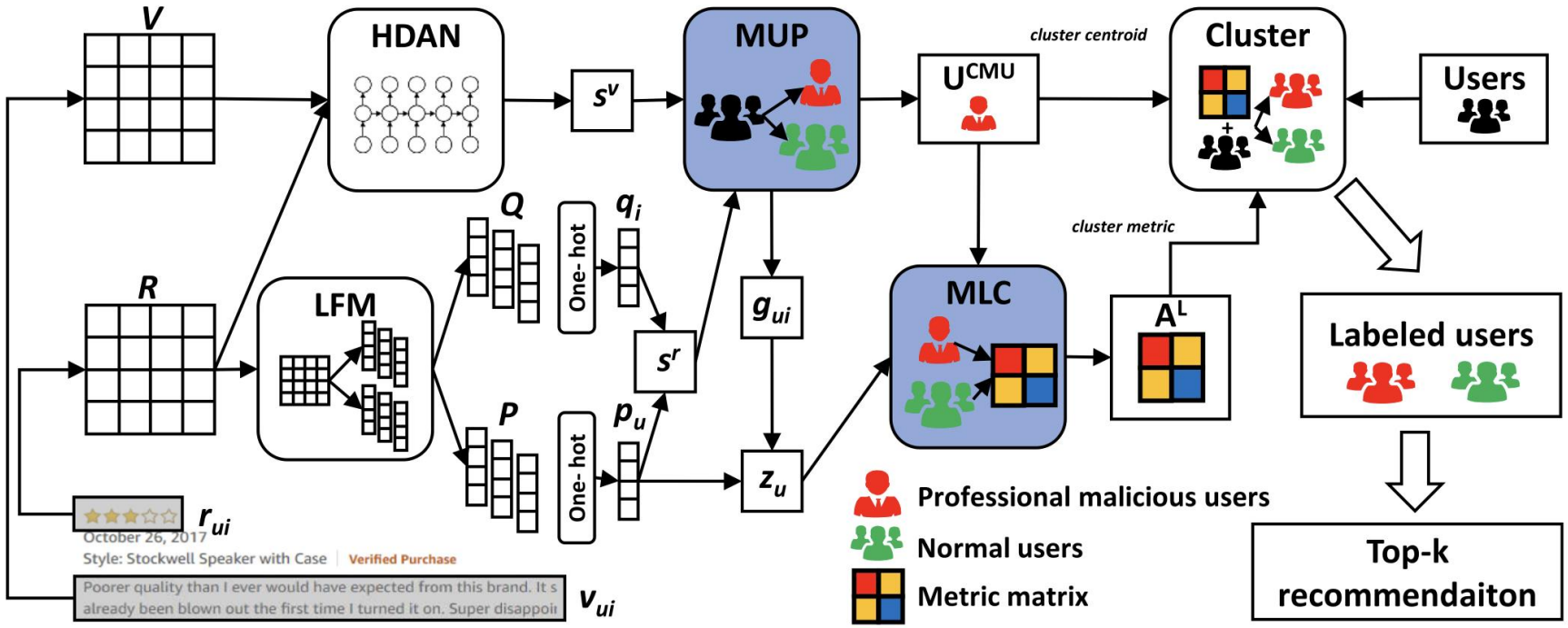
Unsupervised Filtering (Label ALL Unknown): this type of methods are applied in scenarios lacking labeled data, which involves segregating data into various clusters without prior knowledge of their properties.



https://maddevs.io/blog/supervised-learning-explained/

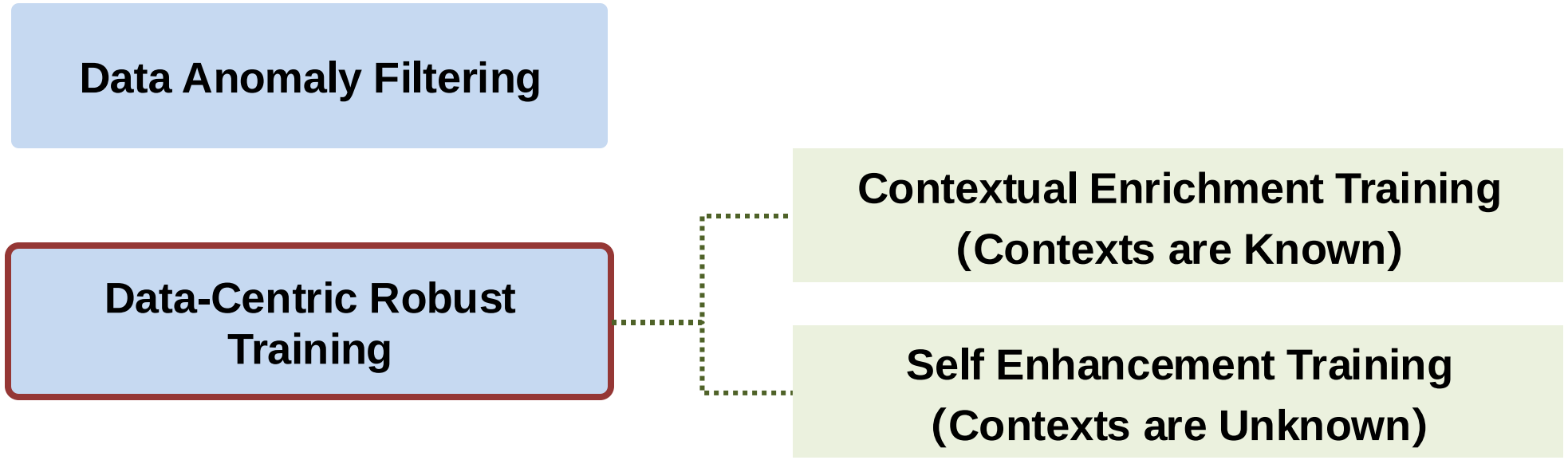
2.2.1 Unsupervised Filtering

Example (Clustering) : MMD utilizes metric learning to capture sentiment gaps of malicious users and true users.



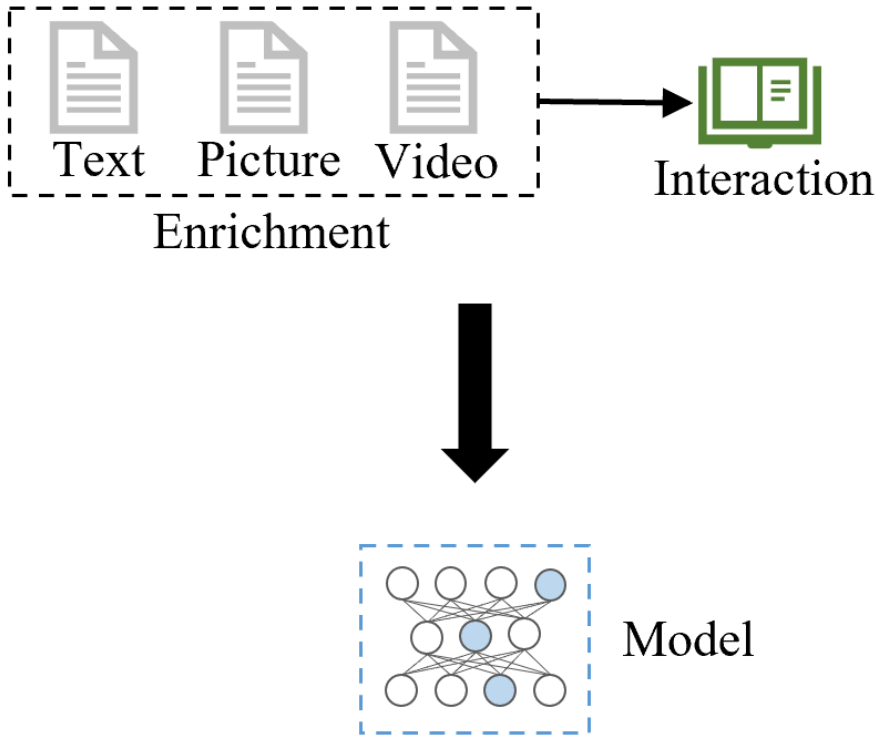
Yuanbo Xu, Yongjian Yang, En Wang, Fuzhen Zhuang, and Hui Xiong. 2020. Detect professional malicious user with metric learning in recommender systems. IEEE Transactions on Knowledge and Data Engineering 34, 9 (2020), 4133–4146.

2.2 Protecting Data Integrity



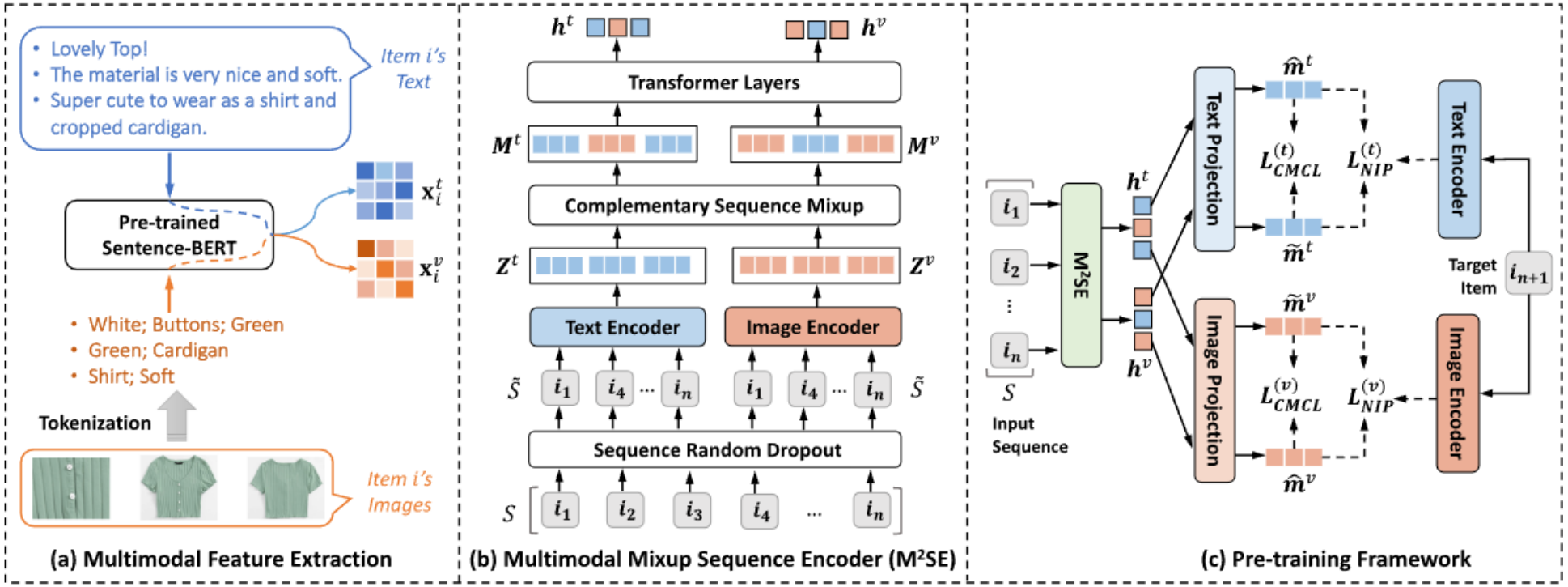
2.2.2 Contextual Enrichment Training

Contextual Enrichment Training (Contexts are Known): defenders expand the training dataset with a wide variety of additional signals such as user attributes, textual content, images, and social relations.



2.2.2 Contextual Enrichment Training

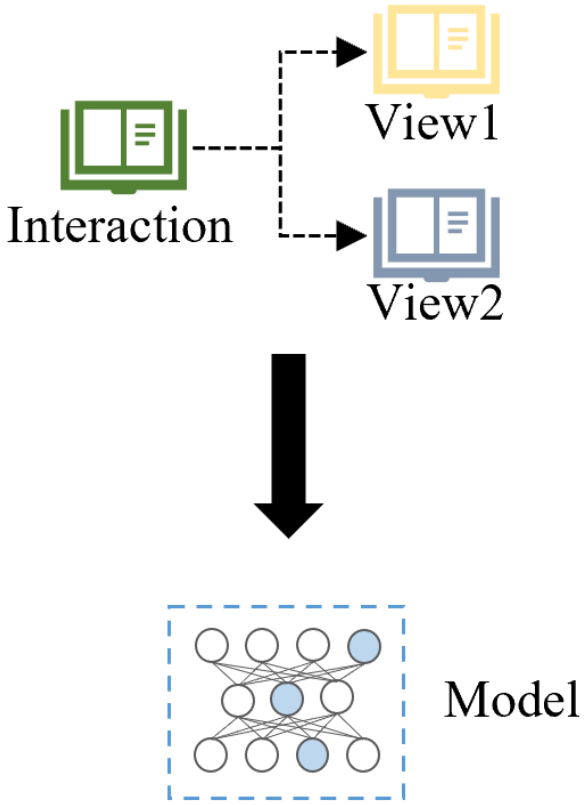
Example (Modality Information): MP4SR improves the modeling of user intentions by incorporating temporal periodicity information.



Zhang, Lingzi, et al. "Multimodal Pre-training for Sequential Recommendation via Contrastive Learning." ACM Transactions on Recommender Systems 3.1 (2024): 1-23.

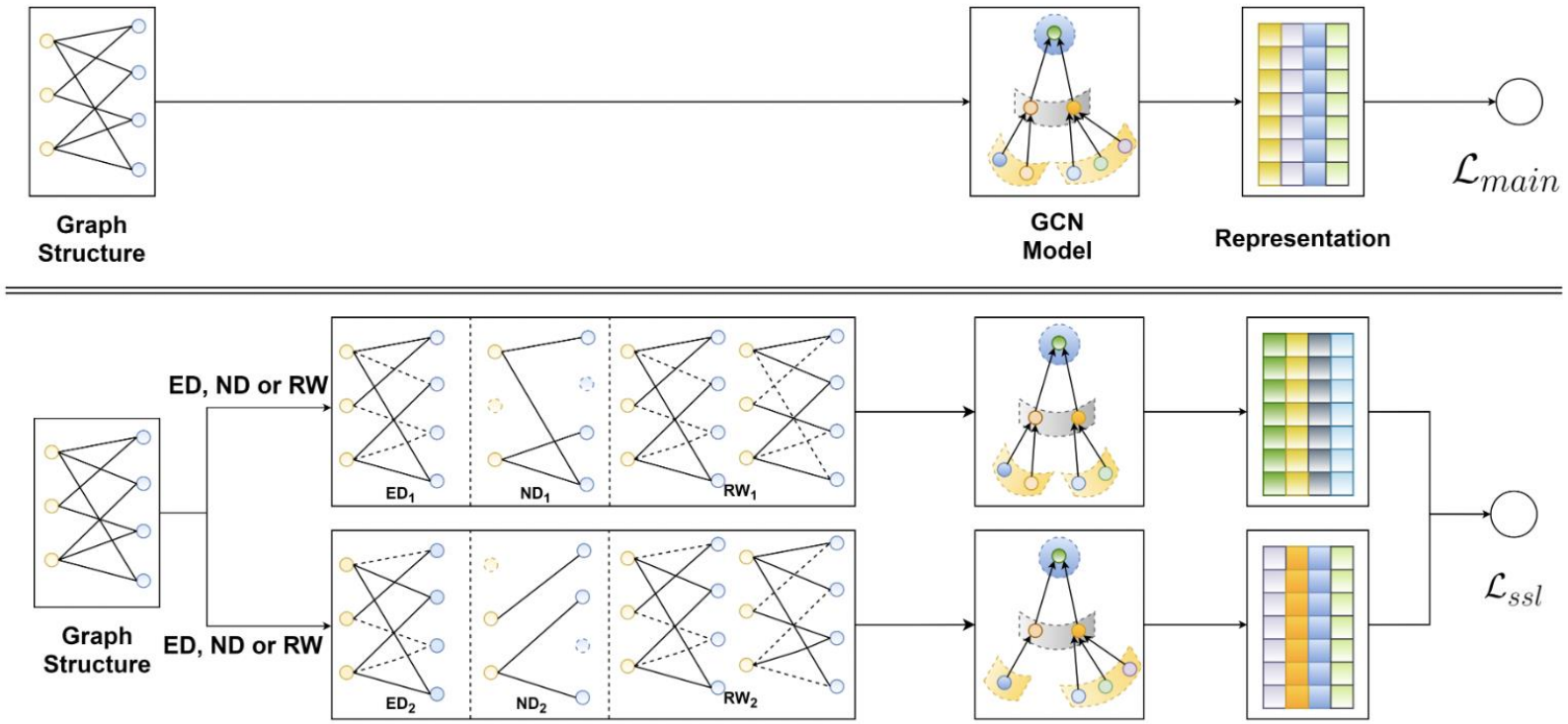
2.2.2 Self Enhancement Training

Self Enhancement Training (Contexts are Unknown): Defenders enhance model training by creating new data signals without additional information.



2.2.2 Self Enhancement Training

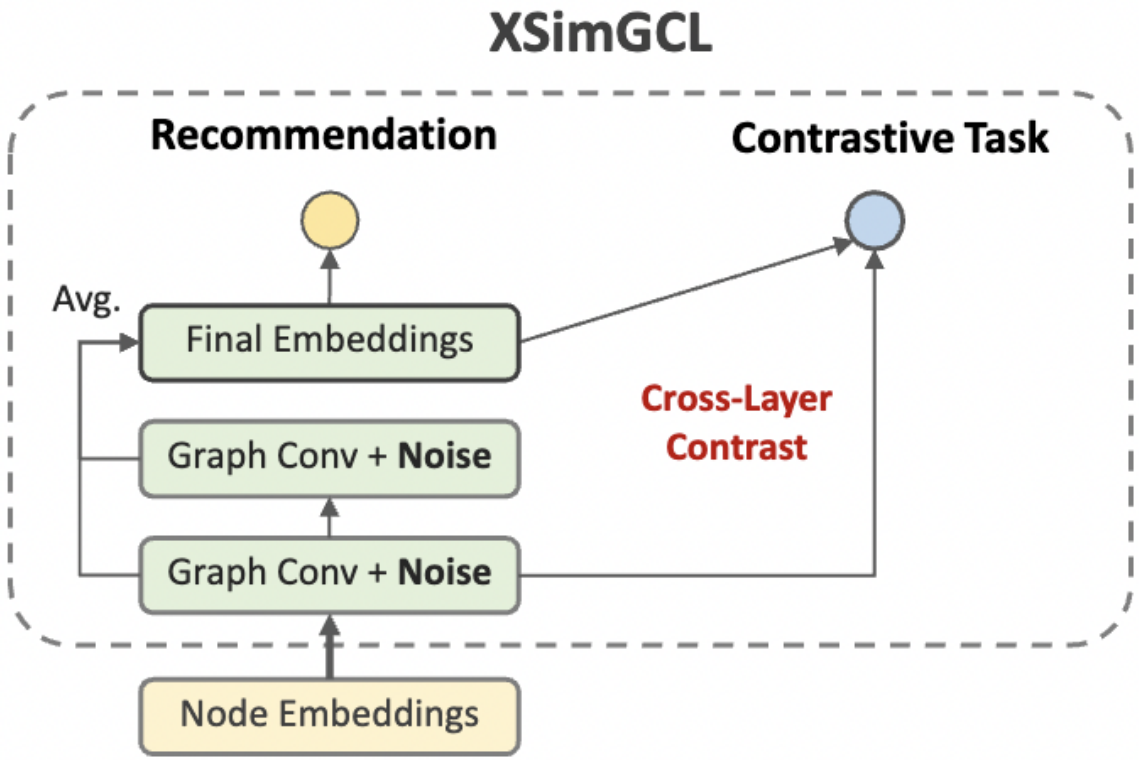
Example1 (Data-Level Self-Augmentation): SGL generates multiple views of a node, maximizing the agreement between different views of the same node compared to that of other nodes.



Wu, Jiancan, et al. "Self-supervised graph learning for recommendation." *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 2021.

2.2.2 Self Enhancement Training

Example2 (Representation-Level Self-Augmentation): XSimGCL employs a simple yet effective noise-based embedding augmentation to generate views for contrastive learning.



Yu, Junliang, et al. "XSimGCL: Towards extremely simple graph contrastive learning for recommendation." *IEEE Transactions on Knowledge and Data Engineering* 36.2 (2023): 913-926.

Outlines

3.1 Inferencing Data Privacy

Main Types Attackers Inference Privacy

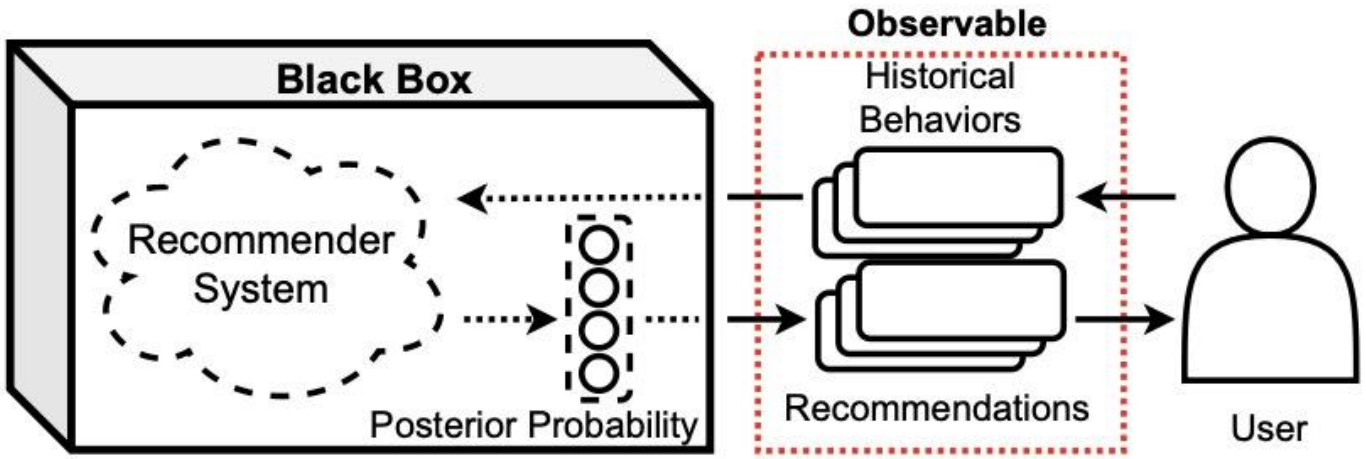
Membership Data Privacy

Attribute Data Privacy

Model Information Privacy

3.1.1 Membership Data Privacy

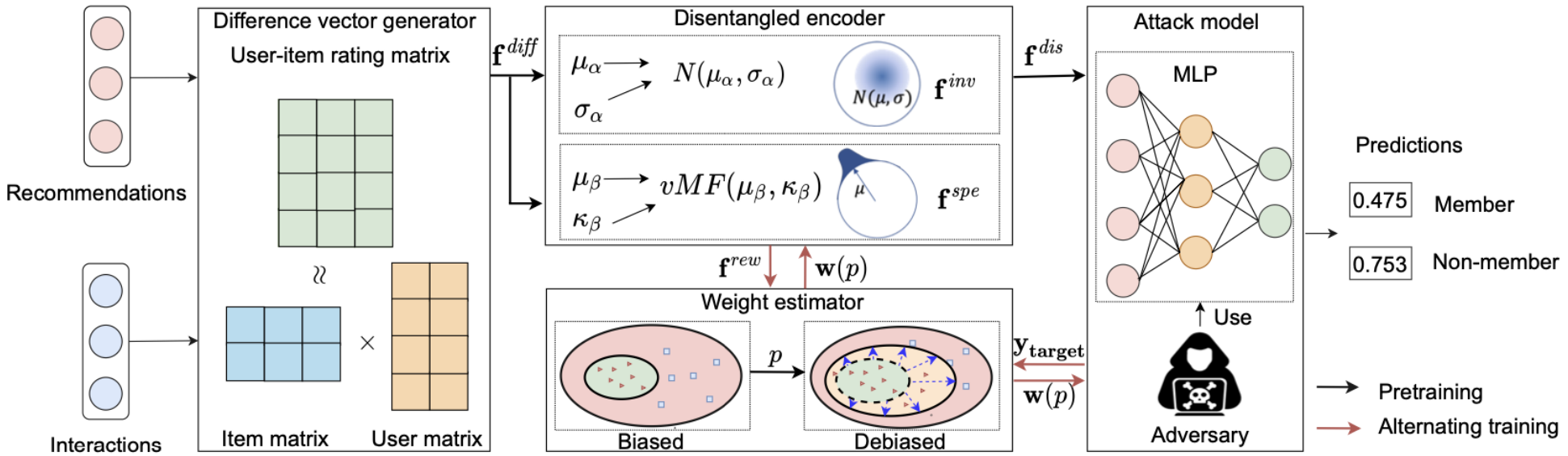
Membership Data Privacy: An attacker can infer the membership of a user, i.e., distinguish member users whose data was used for training the recommender system from non-member users of the model.



Zhang, Minxing, et al. "Membership inference attacks against recommender systems." Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. 2021.

3.1.1 Membership Data Privacy

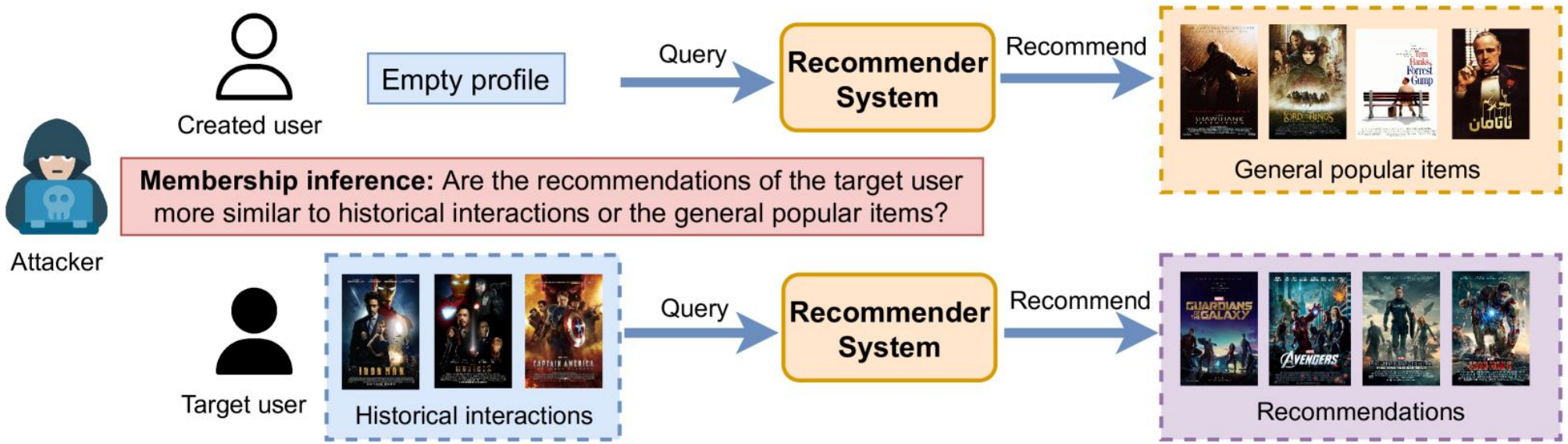
Example1 (Debias Learning): DL-MIA contains four main components: (i) a difference vector generator, (ii) a disentangled encoder, (iii) a weight estimator, and (iv) an attack model.



Wang, Zihan, et al. "Debiasing learning for membership inference attacks against recommender systems." Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2022.

3.1.1 Membership Data Privacy

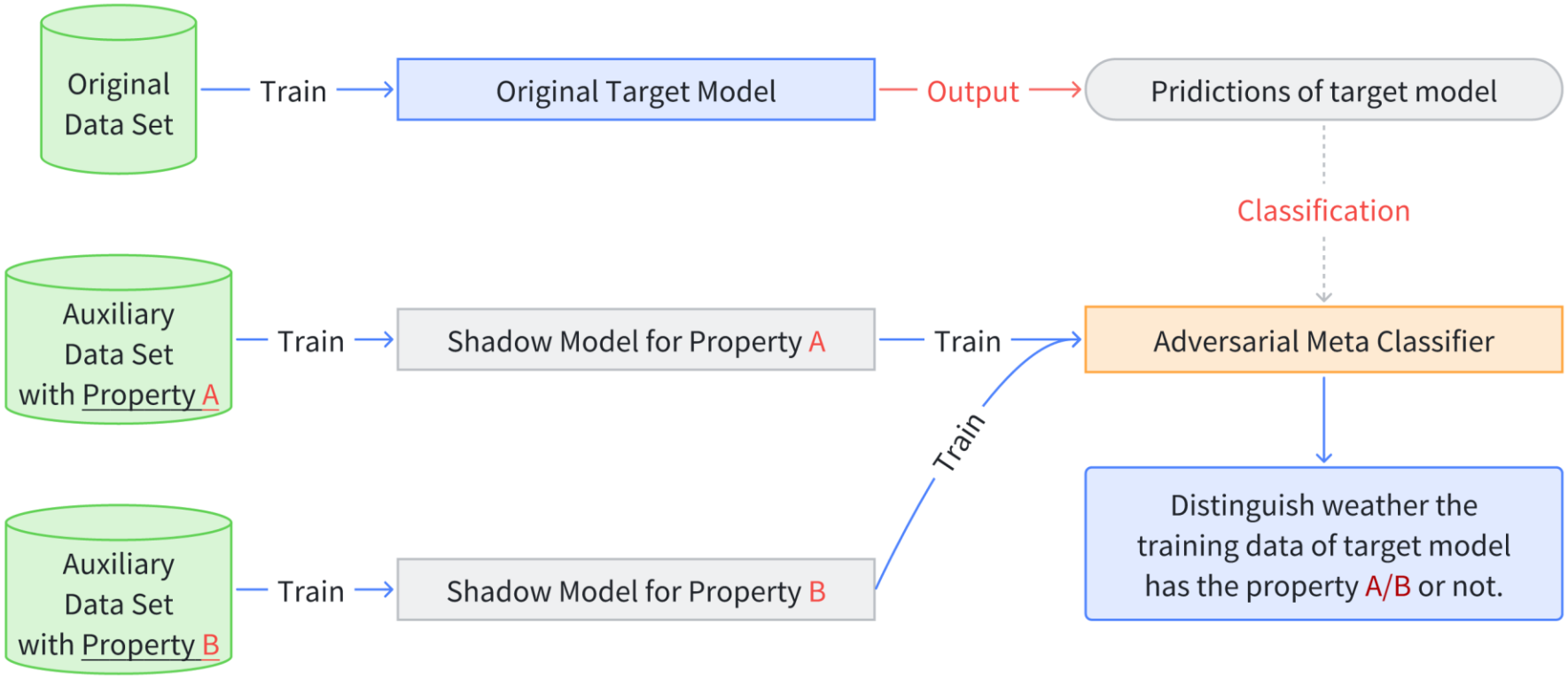
Example2 (Shadow-free): Shadow-free MIA examines whether the recommendations of the target user are more similar to his historical interactions or the general popular items to determine the membership status of the target user.



Chi X, Zhang X, Wang Y, et al. Shadow-free membership inference attacks: recommender systems are more vulnerable than you thought[C]//Proceedings of the Thirty-third International Joint Conference on Artificial Intelligence. 2024.

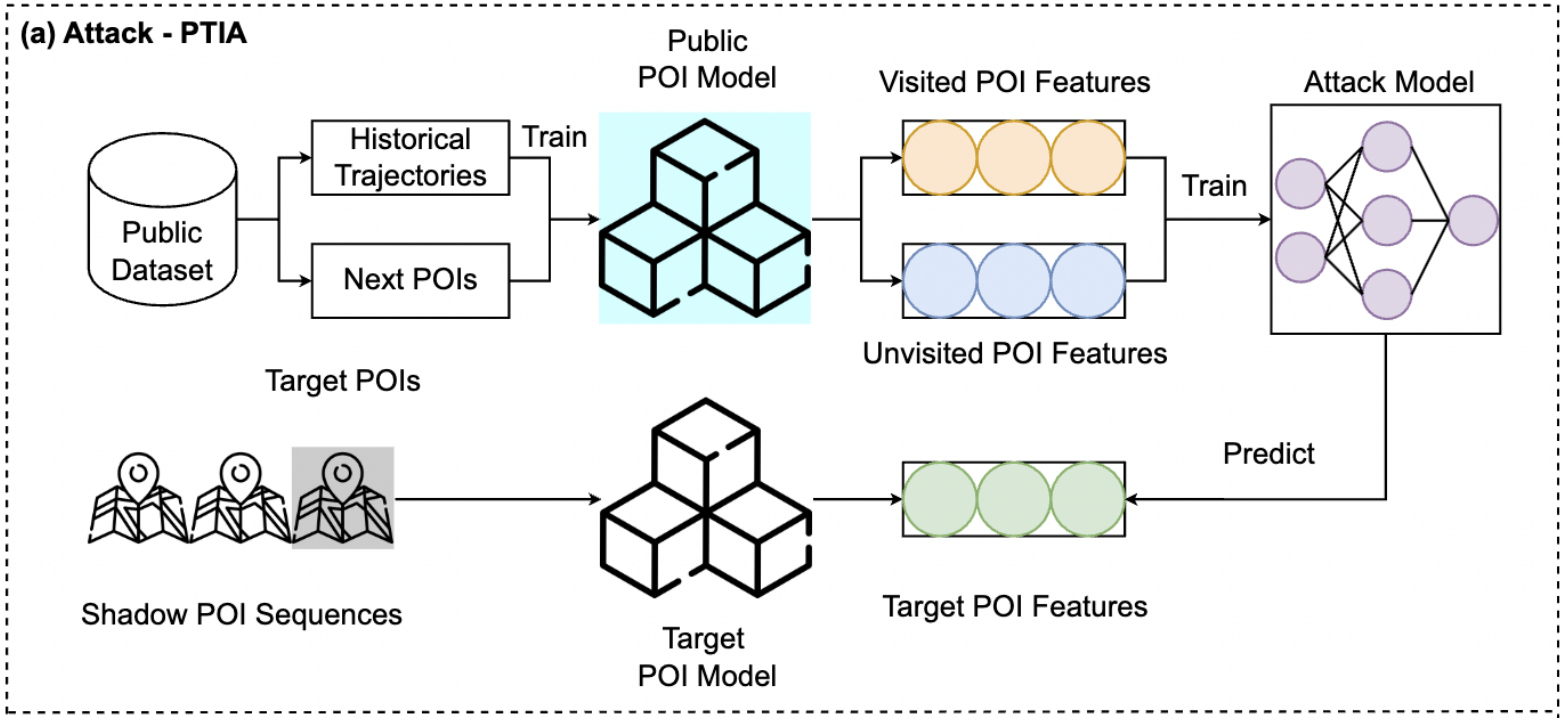
3.1.2 Attribute Data Privacy

Attribute Data Privacy: Attackers attempt to infer specific attribute information about the training data, such as personal characteristics or data-specific properties.



3.1.2 Attribute Data Privacy

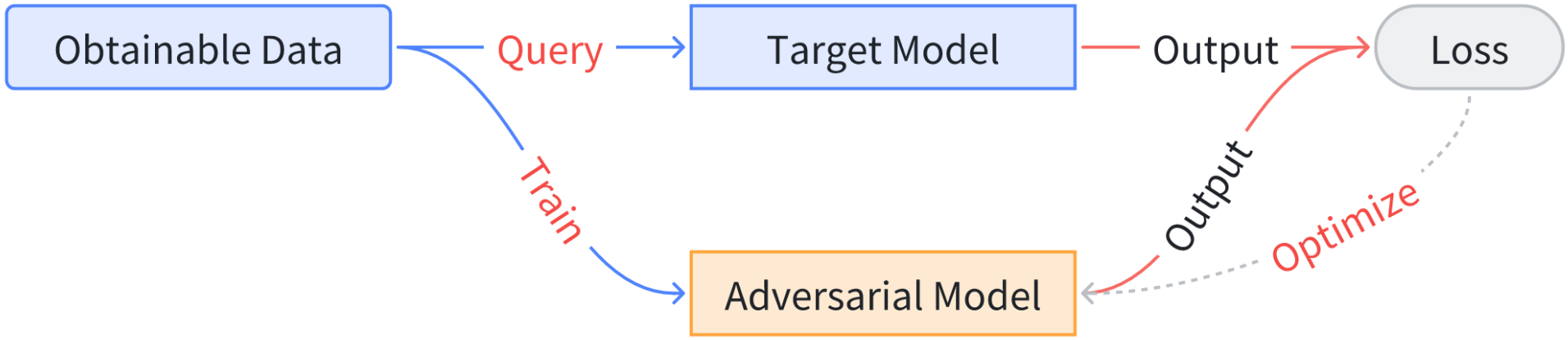
Example (POI Information Privacy): PTIA demonstrates the next POI recommendation can be not only represented by their own embeddings but also inferred from the embeddings of related POIs.



Long J, Chen T, Ye G, et al. Physical Trajectory Inference Attack and Defense in Decentralized POI Recommendation[C]//Proceedings of the ACM on Web Conference 2024. 2024: 3379-3387.

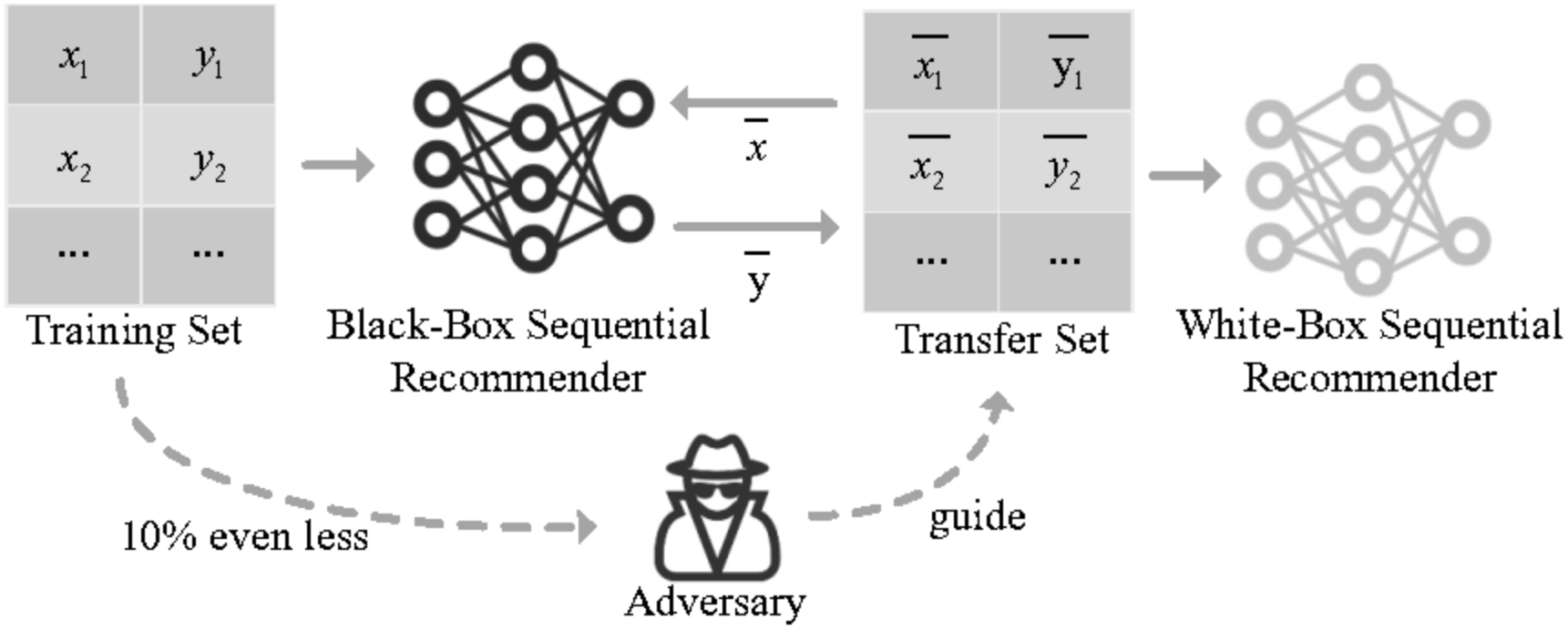
3.1.3 Model Information Privacy

Model Information Privacy: Attacker tries to extract the entire or partial model, its parameters, or the knowledge it has learned in a recommender system.



3.1.3 Model Information Privacy

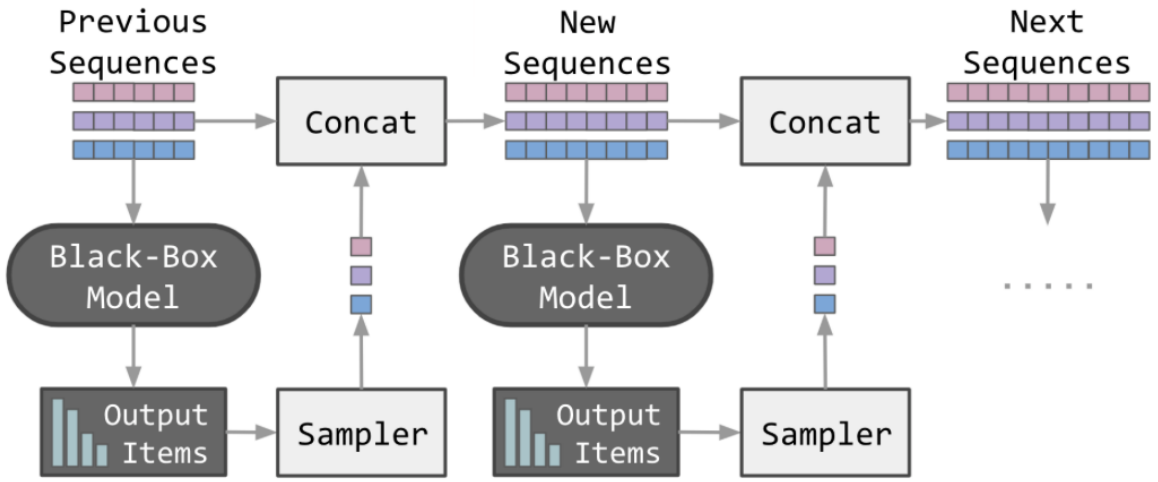
Example1 (Knowledge Distillation)： **MEA** uses few-shot raw data to obtain a white-box model.



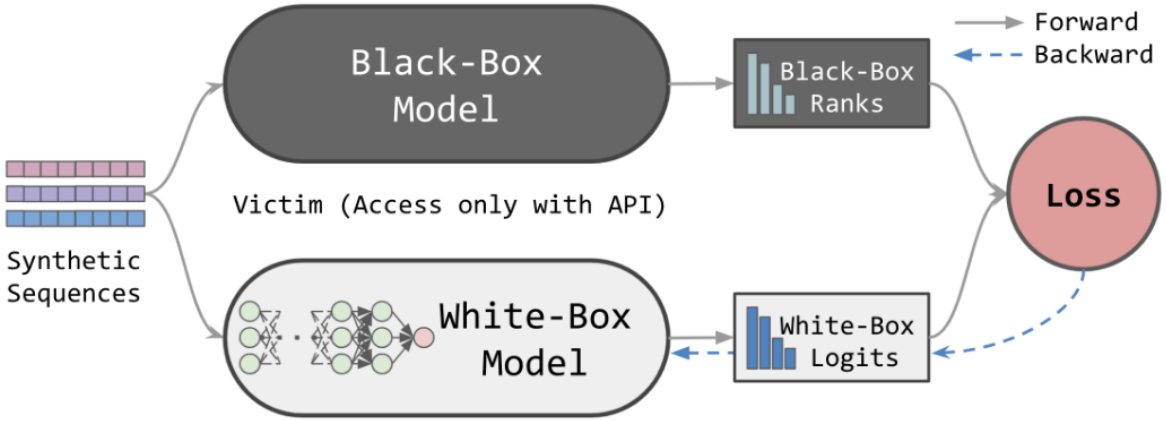
Zhang H, Liu F. Few-shot Model Extraction Attacks against Sequential Recommender Systems[J] arXiv, 2024.

3.1.3 Model Information Privacy

Example2 (Knowledge Distillation): Data-free MEA merely needs limited-budget synthetic data generation and knowledge distillation.



(1) Autoregressive Data Generation



(2) Workflow of Model Extraction Attack on Sequential Recommender

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

3.2 Data Privacy-Preserving Methods

Main Ways Defenders Alleviate Data Noise

Differential Privacy

Federated Learning

Adversarial Learning

3.2.1 Differential Privacy

Differential Privacy: A common way to preserve membership inference attacks, which can provide strict statistical guarantees for data privacy.

A randomized algorithm A is (ε, δ)-differentially private if for any pair of neighboring datasets D and D′ differing in one record, and for all S that is a subset of the range of A, we have

$$\mathbf{Pr}\left(\mathcal{A}(D) \in \mathcal{S}\right) \leq e^{\varepsilon}\mathbf{Pr}\left(\mathcal{A}(D') \in \mathcal{S}\right) + \delta,$$

where the probability is over the randomness of A. Usually, ε is assumed to be a small constant, and $\delta \ll 1/|D|$ where |D| is the size of dataset D.

3.2.1 Differential Privacy

Example: EANA modifies DP-SGD. It adds noise only to embedding parameters with non-zero gradients during training.

Algorithm 2 EANA

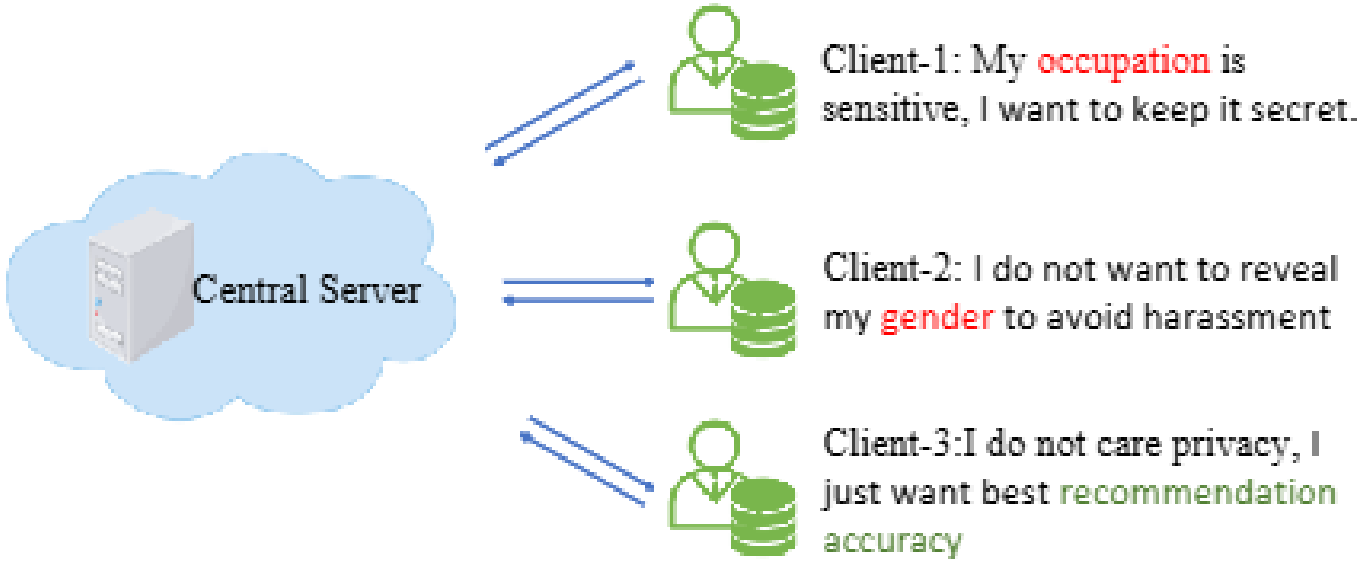
Require: Data set $D = \{d_1, \cdots, d_n\}$, model with p non-embedding parameters and e embedding parameters, loss function: $\ell : \mathbb{R}^{p+e} \times D \to \mathbb{R}$, gradient ℓ_2-norm bound: L, number of iterations: T, noise variance: σ^2, learning rate: η.

1: Randomly initialize θ_0.
2: **for** $t = 0, \ldots, T-1$ **do**
3: Randomly sample a batch of examples $B \subseteq D$.
4: $g_t \leftarrow \frac{1}{|B|} \sum_{d \in B} \text{clip}\left(\nabla \ell(\theta_t; d), L\right) + \delta_E \odot \mathcal{N}\left(0, \sigma^2 \mathbb{I}\right)$ where $\text{clip}\left(x, L\right) = \min\{1, L/\|x\|_2\} \cdot x$, $\delta_E \in \{0,1\}^{p+e}$ is 1 for embedding parameters and 0 elsewhere, and $\odot$ represents the element-wise product between two vectors.
5: $\theta_{t+1} \leftarrow \theta_t - \eta g_t$.
6: **end for**
7: **return** θ_T.

Ning L, Chien S, Song S, et al. EANA: reducing privacy risk on large-scale recommendation models[C] RecSys, 2022.

3.2.2 Federated Learning

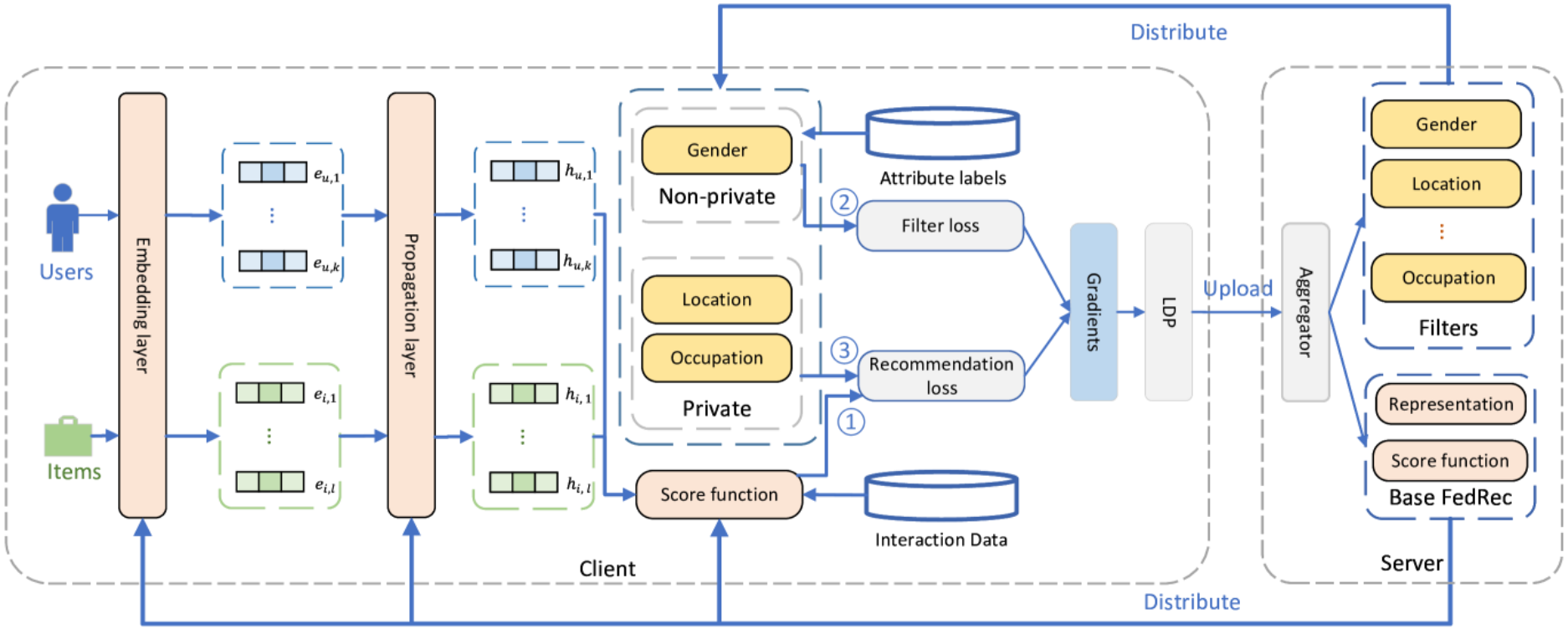
Federated Learning: isolates users' data and the cloud server by only transferring the gradients between them.



Hu Q, Song Y. User Consented Federated Recommender System Against Personalized Attribute Inference Attack[C]//Proceedings of the 17th ACM International Conference on Web Search and Data Mining. 2024: 276-285.

3.2.2 Federated Learning

Example: UC-FedRec ① train the recommendation part as the primary goal; ② train attribute distribution estimator on personal non-private attributes; ③utilize estimators to eliminate private information in the representation model.

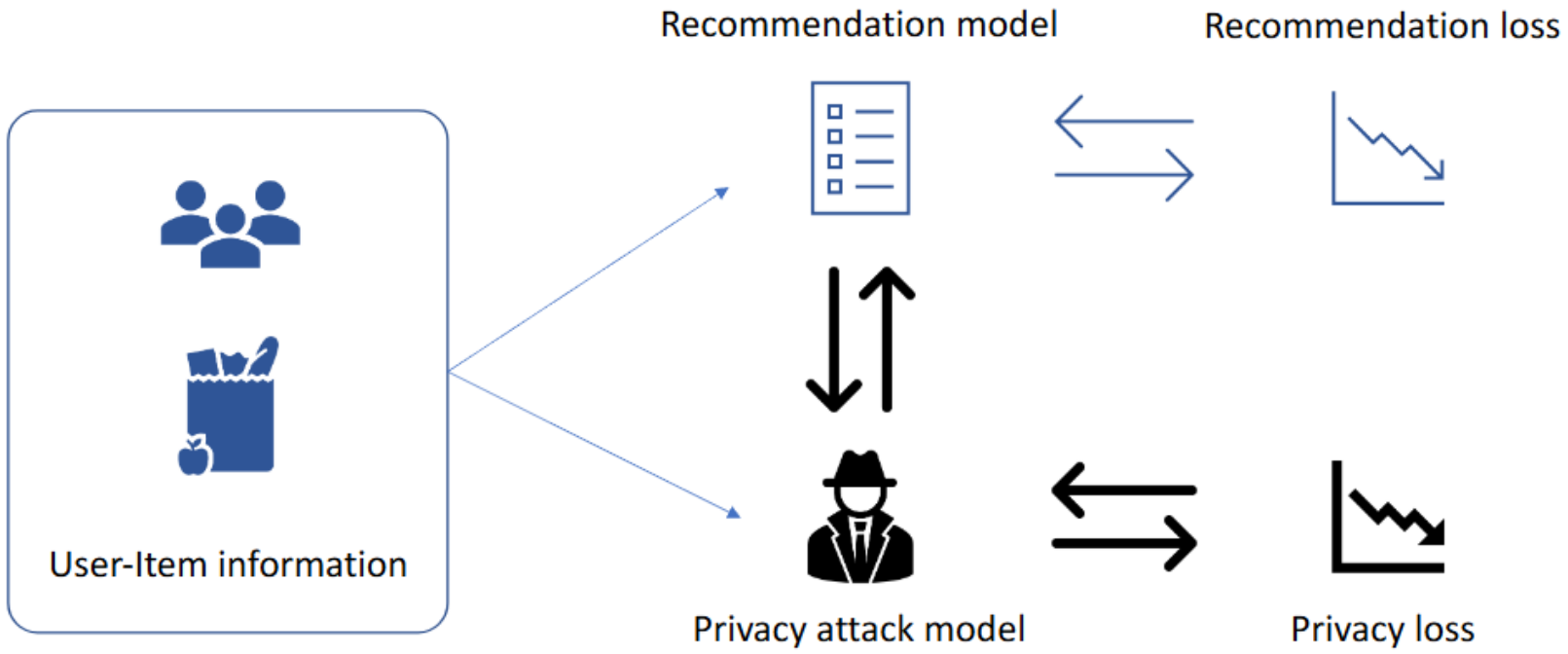


Hu Q, Song Y. User Consented Federated Recommender System Against Personalized Attribute Inference Attack[C]//Proceedings of the 17th ACM International Conference on Web Search and Data Mining. 2024: 276-285.

75

3.2.3 Adversarial Learning

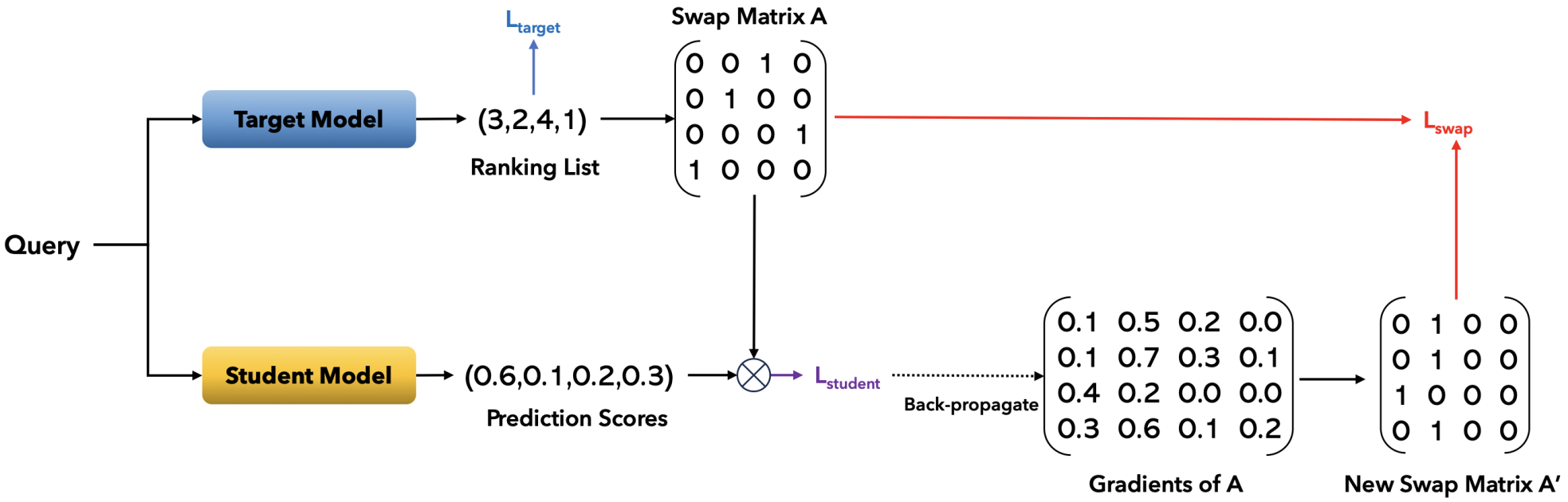
Adversarial Learning: minimizes the loss of the protected target model while maximizing the cost of the attacker.



Fan, Wenqi, et al. "Trustworthy Recommender Systems: Foundations and Frontiers." *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2023.

3.2.3 Adversarial Learning

Example: GRO minimizes the loss of the protected target model while maximizing the loss of the attacker's surrogate model.



Zhang S, Yin H, Chen H, et al. Defense Against Model Extraction Attacks on Recommender Systems[C] WSDM, 2024.

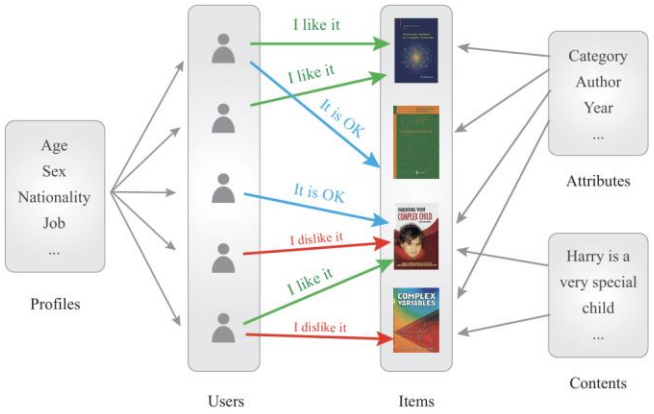
Outlines

4.1 Origins and Types of Data Noises

Noise in User Behavior Data

Noise in Item

Data Noise is Everywhere

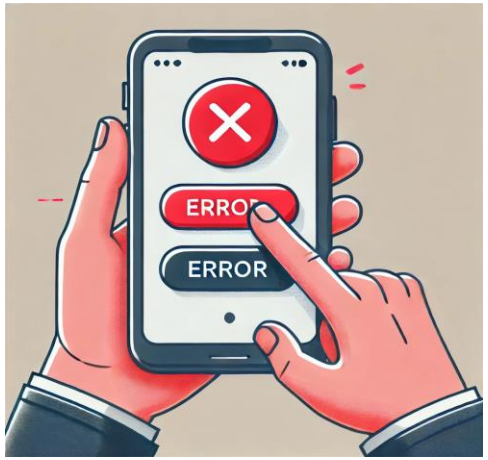


Noise from RS Itself

External Noise

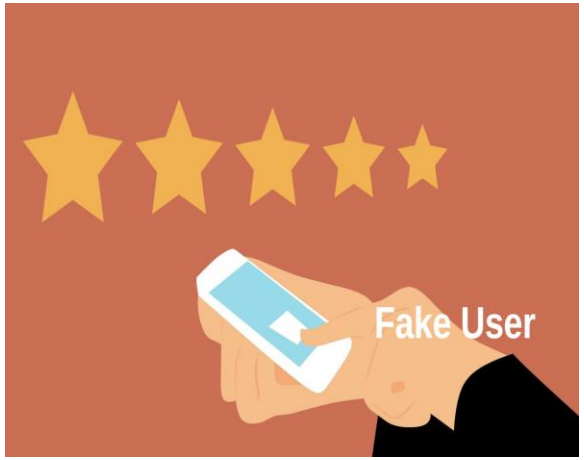
4.1 Noise in User Behavior Data

Mis-Operation Noise



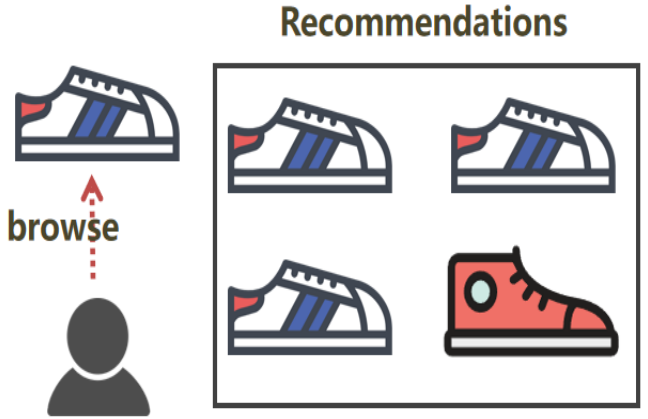
Unintentional clicks, or random operations

Malicious Behavior Noise



Fake orders or false reviews

Redundant Data Noise



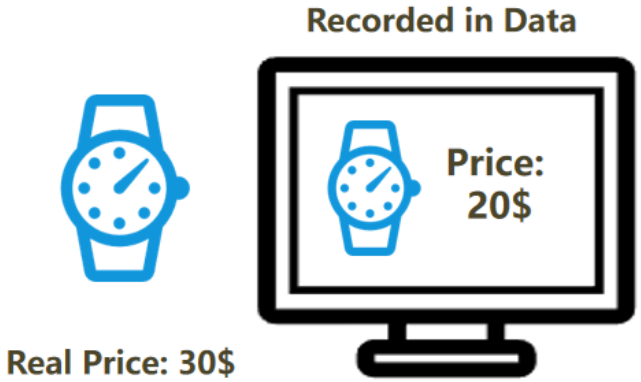
Duplicate or redundant information

4.1 Noise in Items

Descriptive Noise

The item's description information (such as title, tag, etc.) is inaccurate or outdated.

Attribute Noise

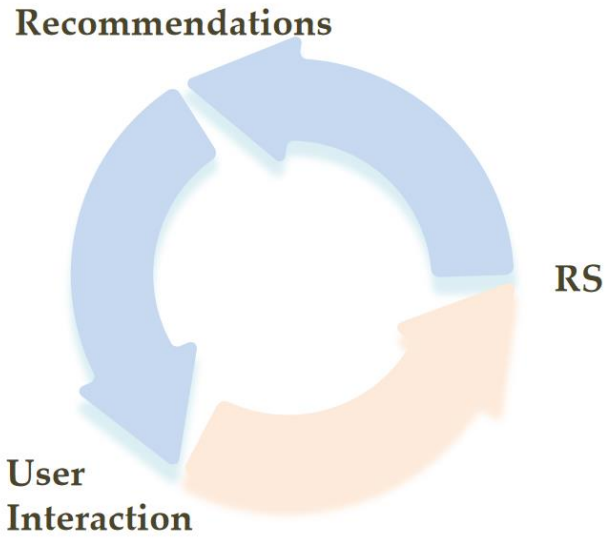


The actual attributes of the item, such as price and rating, are inconsistent with those recorded ones.

Content Update Lag

Item information changes are not updated in a timely manner.

4.1 Noise From System



One characteristic of RS is the feedback loop — the exposure mechanism of the RS determines user behaviors, which are then fed back as training data for the RS. This feedback loop not only creates biases but also intensifies them over time, resulting in the 'rich get richer' Matthew effect.

Abnormal Data Collection: noise caused by platform technical failures, network problems, or data transmission errors.

4.1 Noise From External Noise

Market Fluctuation Noise

External factors, such as the economy and politics, can affect consumers' purchasing behavior.

Emergency Noise

Unexpected situations can lead to drastic changes in user behavior.

User Emotional Noise

Temporary emotional changes in users may cause their behavior to deviate from the normal state.

4.2 Data Denoising Methods

Main Ways of Defenders Alleviating Data Noise

- **Data Cleaning**
- **Data Adaptive Correction**
- **Data Information Augmentation**
- **Model Enhancement**

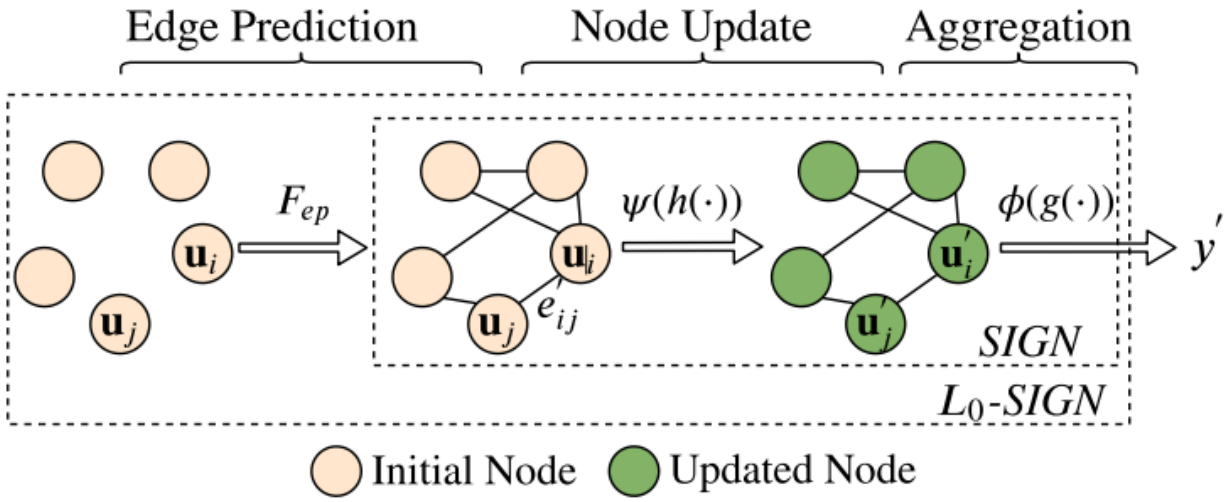
84

4.2.1 Data Cleaning

Data Cleaning: Identify and remove unreliable or harmful information from raw data. This method typically employs noise detection techniques, such as anomaly detection or rule-based removal, to identify errors or invalid information.

4.2.1 Data Cleaning

Example1: L0-SIGN detects Beneficial Feature Interactions in RS.



1. L0 Edge Prediction Model:

• Predicts the existence of beneficial feature interactions using a graph-based structure.

• L0 activation regularization helps in removing unbeneficial interactions, ensuring only relevant ones are retained.

2. SIGN (Statistical Interaction Graph Neural Network):

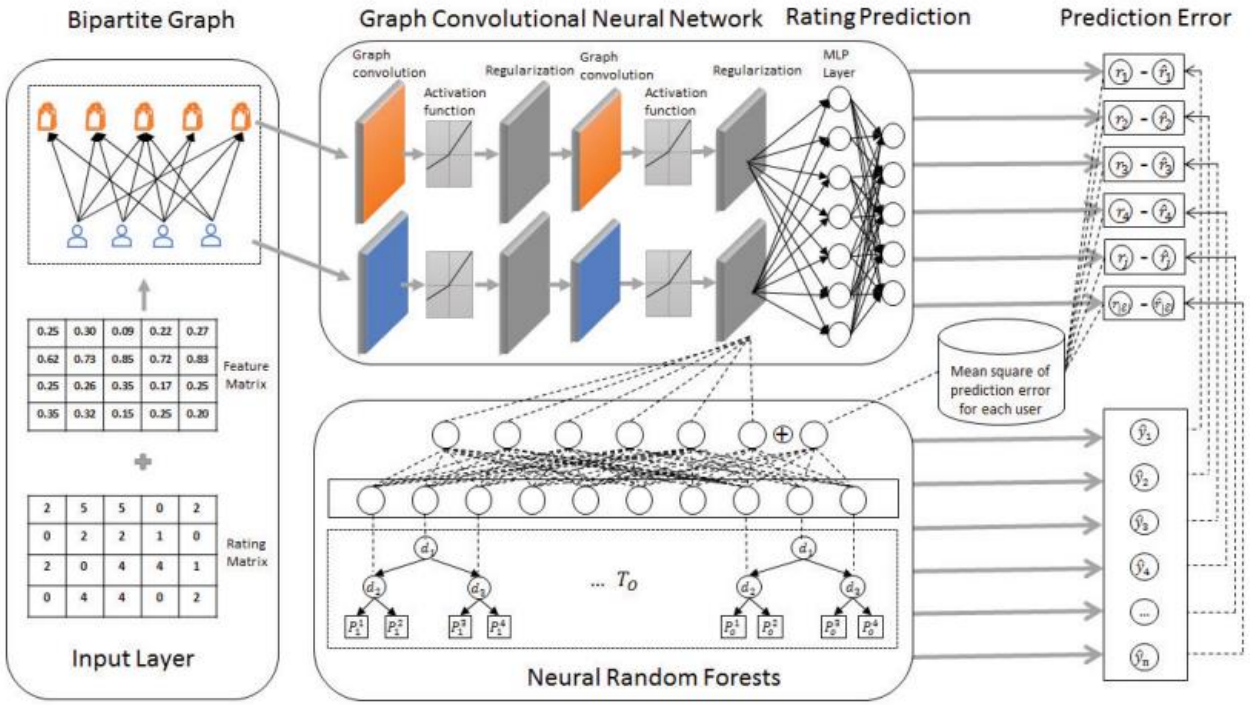
• Utilizes the detected beneficial feature interactions to make accurate recommendation predictions.

• Effectively models the relationships between features using GNNs for better accuracy.

Su Y, Zhang R, Erfani S, et al. Detecting beneficial feature interactions for recommender systems[C]//Proceedings of the AAAI conference on artificial intelligence. 2021.

4.2.1 Data Cleaning

Example2: Graphfi fulfills fraudster detection through a GCN.



1. GCN: User preference learning

- Utilize the user-item interaction graph to capture user preferences and reliability.
- Update user representations by integrating node features and graph structure.

2. NRF: Fraudulent user detection

- Detect fraudsters by analyzing prediction errors and classifying users.
- Adjusts the impact of user ratings on the recommendation process.

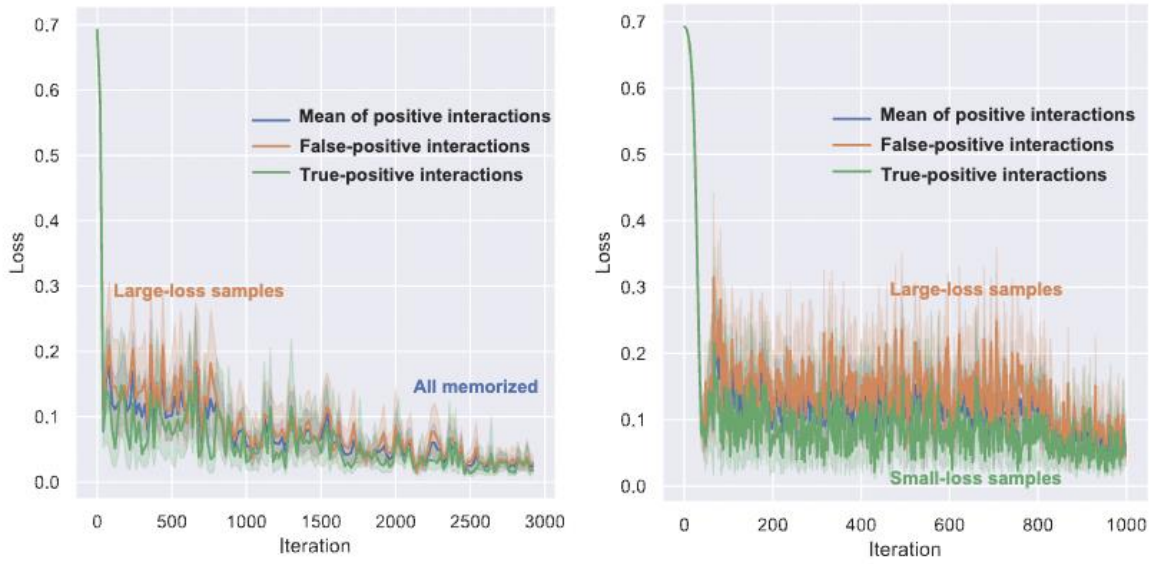
Zhang S, Yin H, Chen T, et al. Gcn-based user representation learning for unifying robust recommendation and fraudster detection[C]//Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 2020: 689-698.

4.2.2 Data Adaptive Correction

Data Adaptive Correction: Minimize the impact of noise on recommendation results by dynamically adjusting system behavior. This approach is usually based on a feedback mechanism that gradually adjusts the weights and parameters of the model to address the noise in the data.

4.2.2 Data Adaptive Correction

Example1: T-CE identifies samples with high loss as more likely to be noisy



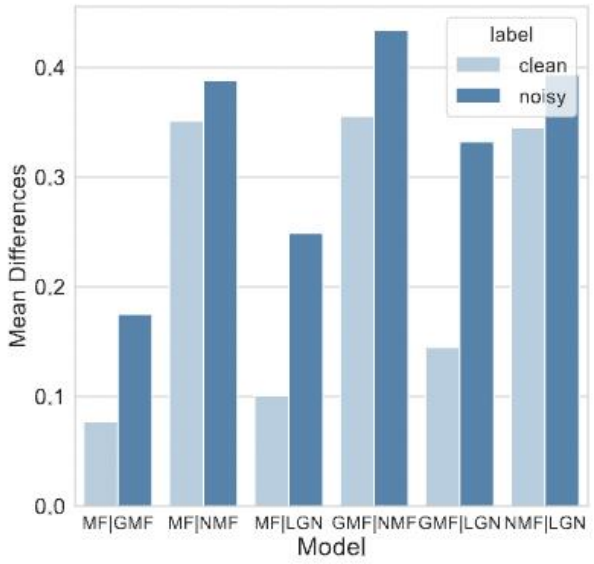
(a) Whole training process (b) Early training stages

$$\mathcal{L}_{T\text{-}CE}(u,i) = \begin{cases} 0, & \mathcal{L}_{CE}(u,i) > \tau \wedge \bar{y}_{ui} = 1 \\ \mathcal{L}_{CE}(u,i), & \text{otherwise,} \end{cases}$$

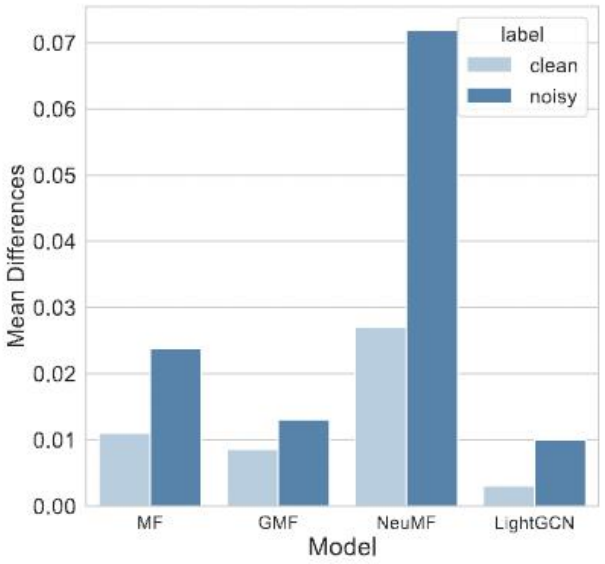
Wang, Wenjie, et al. "Denoising implicit feedback for recommendation." Proceedings of the 14th ACM international conference on web search and data mining. 2021.

4.2.2 Data Adaptive Correction

Example2: DeCA finds that clean samples produce similar predictions across different models. By jointly training two recommendation models, it differentiates clean samples from noisy ones based on prediction disagreements.



(a) Different models.

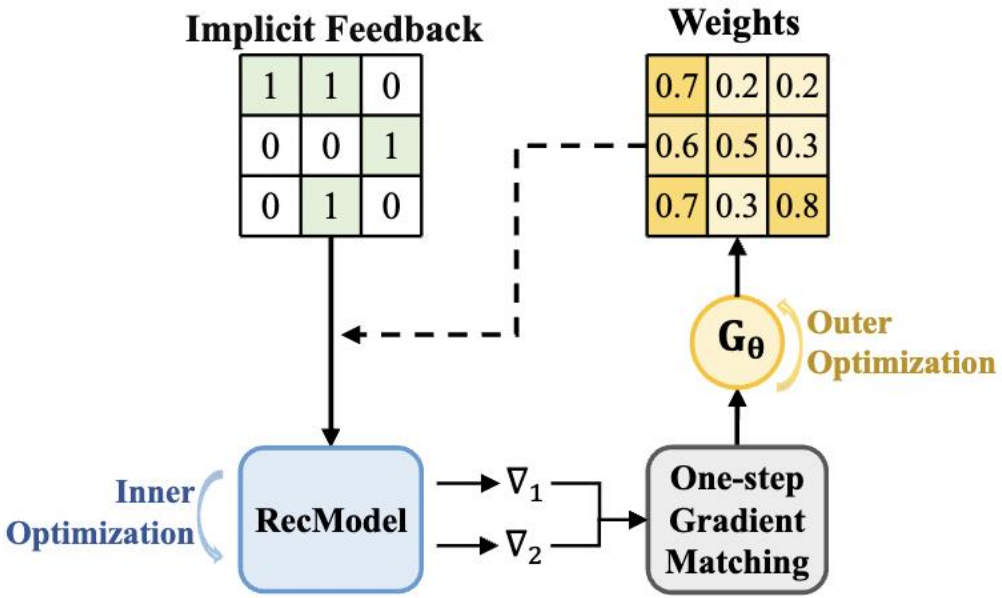


(b) Different random seeds.

Wang, Yu, et al. "Learning robust recommenders through cross-model agreement." Proceedings of the ACM Web Conference 2022. 2022.

4.2.2 Data Adaptive Correction

Example3: BOD finds that samples yielding consistent losses across different loss functions are likely to be clean.



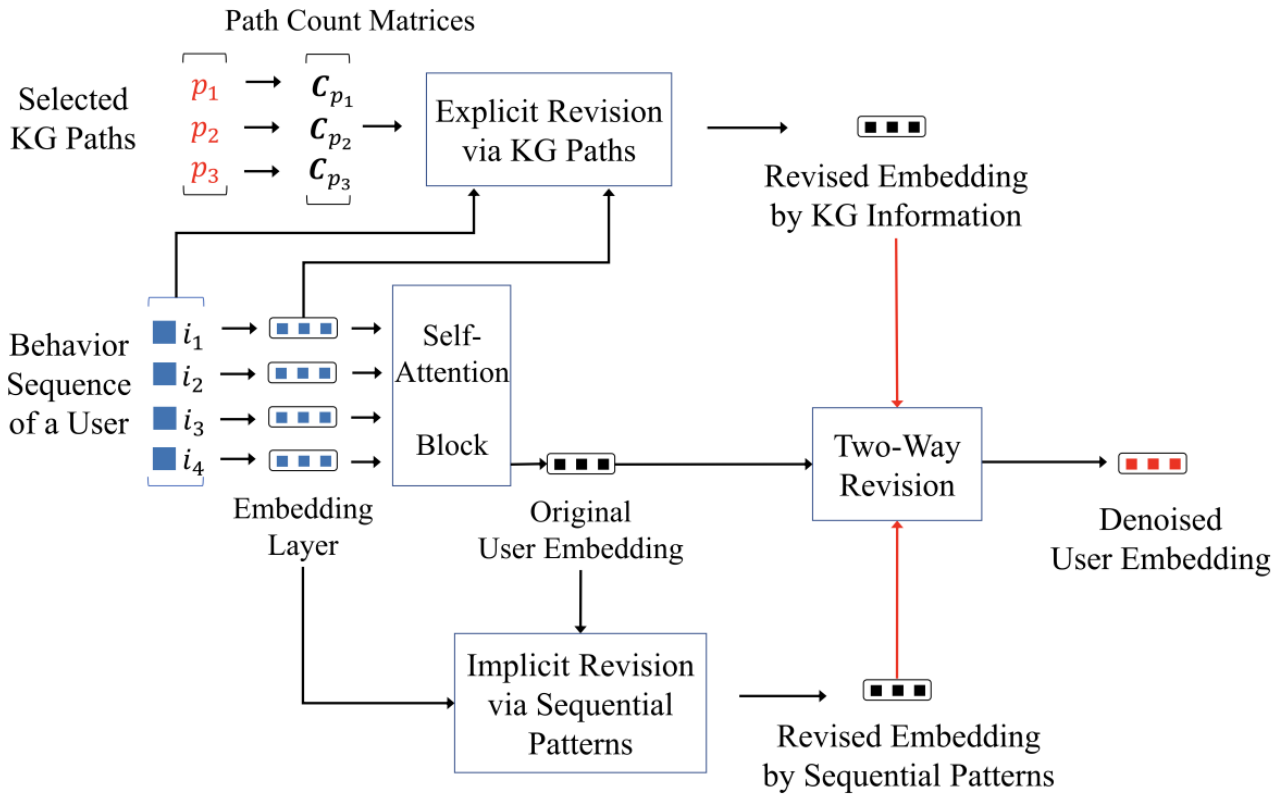
Wang, Zongwei, et al. "Efficient bi-level optimization for recommendation denoising." *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2023.

4.2.3 Data Information Augmentation

Data Information Augmentation: Introduce additional external information or data sources to enrich and correct existing data, thereby reducing the noise's impact on the recommendation system.

4.2.3 Data Information Augmentation

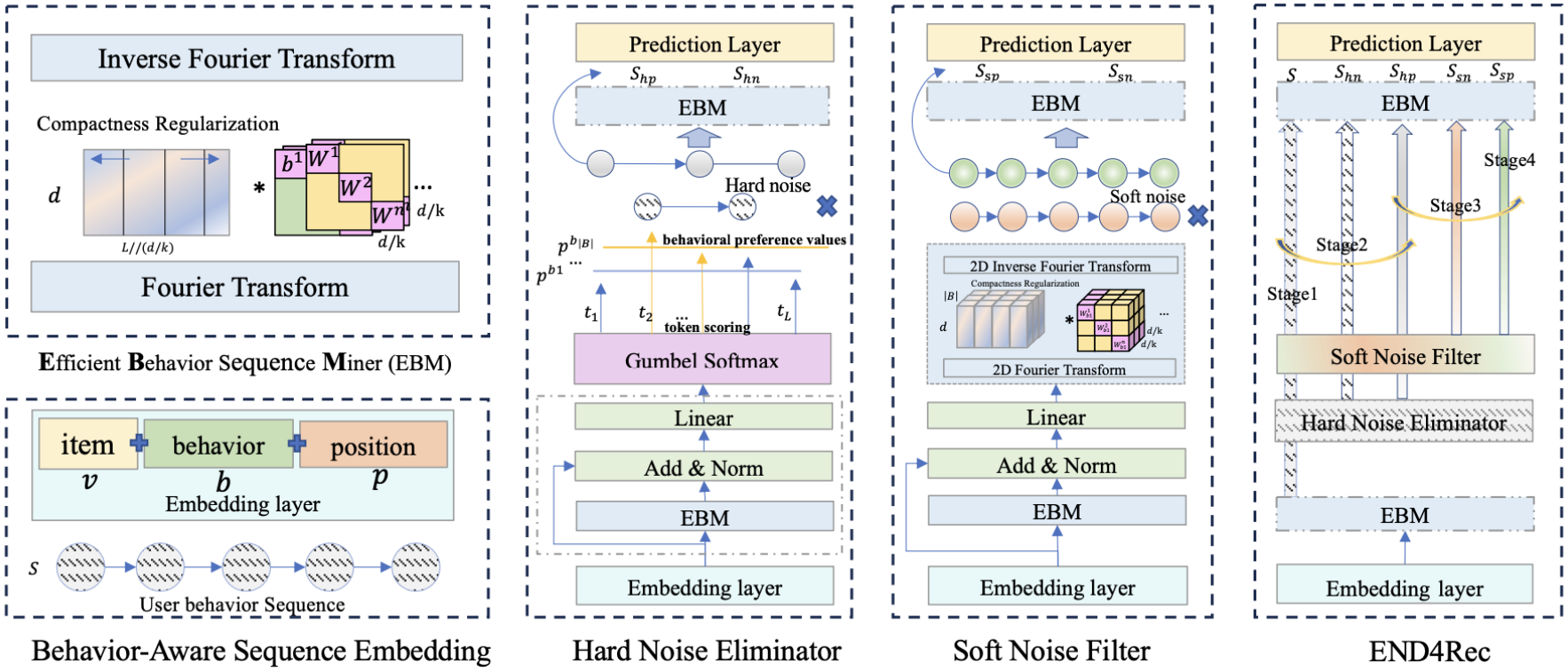
Example1: KGDPL mitigates the negative effects of the noise behaviors contained in historical sequences by using the knowledge path information.



Liu, Hongzhi, Yao Zhu, and Zhonghai Wu. "Knowledge Graph-Based Behavior Denoising and Preference Learning for Sequential Recommendation." IEEE Transactions on Knowledge and Data Engineering (2023).

4.2.3 Data Information Augmentation

Example2: END4Rec designs hard and soft denoising modules for different noise types and fully explore the relationship between behaviours and noise.



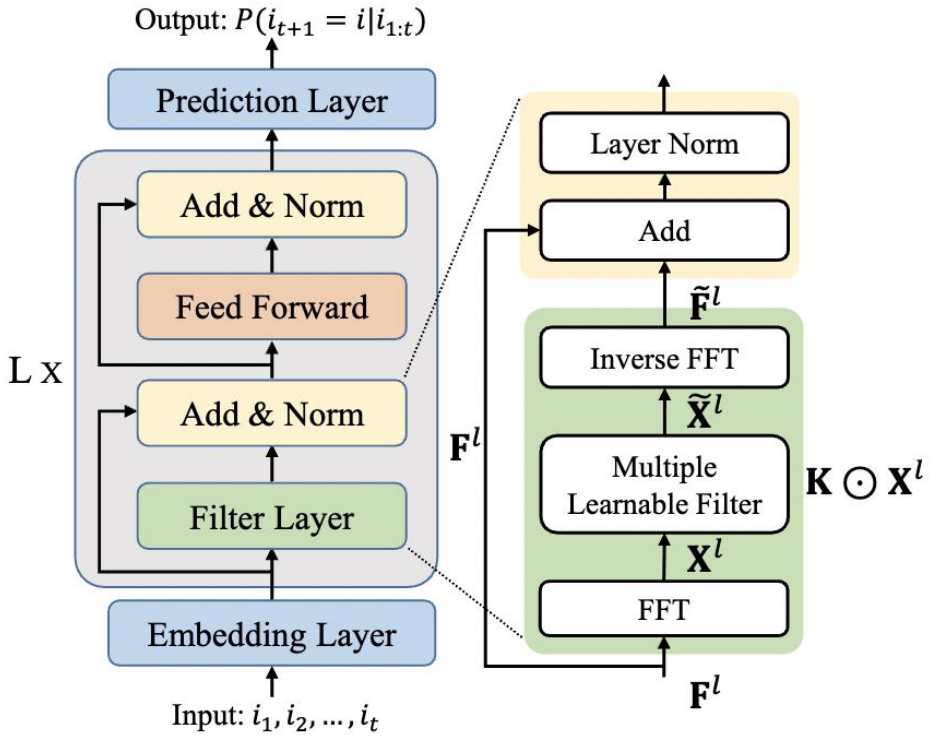
Han, Yongqiang, et al. "Efficient Noise-Decoupling for Multi-Behavior Sequential Recommendation." Proceedings of the ACM on Web Conference 2024. 2024.

4.2.4 Model Enhancement

Model Enhancement: Improve the structure or training method of the recommendation model itself to enhance its robustness to noise. This can be achieved by introducing a more complex model architecture or adopting specialized training techniques.

4.2.4 Model Enhancement

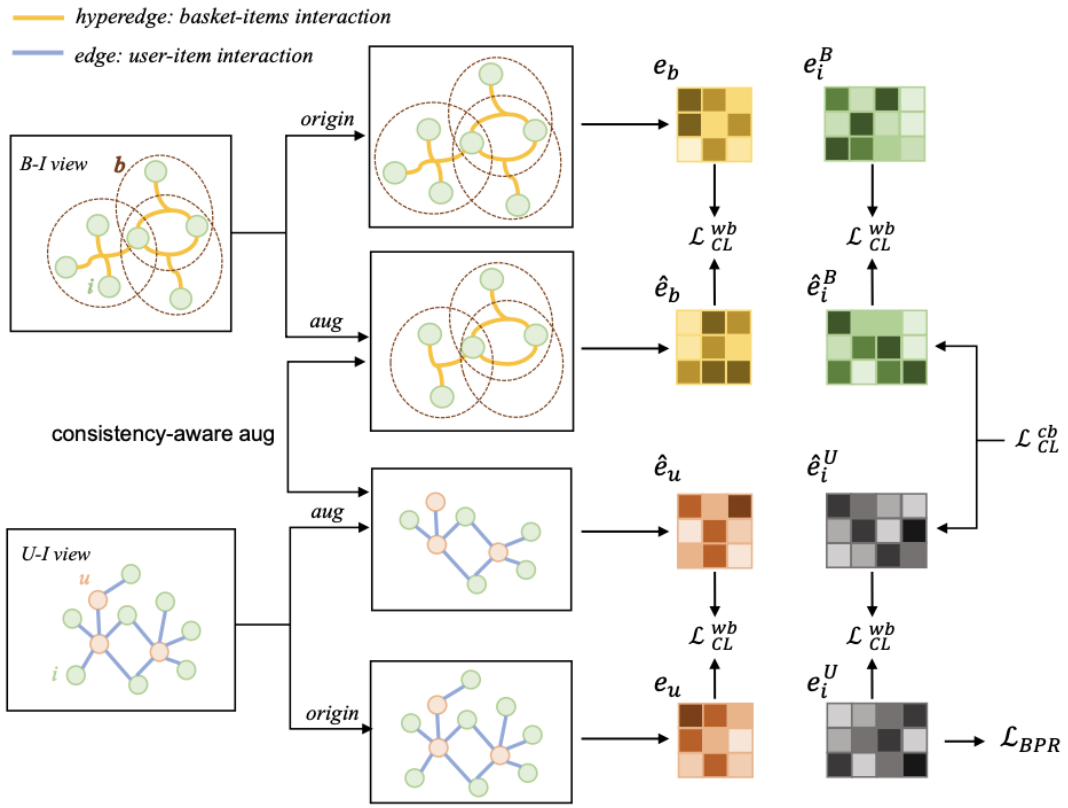
Example1: FMLP-Rec integrates simple filtering algorithms (e.g., Band-Stop Filter) with an all-MLP architecture



Zhou, Kun, et al. "Filter-enhanced MLP is all you need for sequential recommendation." Proceedings of the ACM web conference 2022. 2022.

4.2.4 Model Enhancement

Example2: BNCL designs cross-behavior contrastive learning to suppress the noise during the fusion of diverse behaviors



He, Xinrui, Tianxin Wei, and Jingrui He. "Robust Basket Recommendation via Noise-tolerated Graph Contrastive Learning." Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 2023.

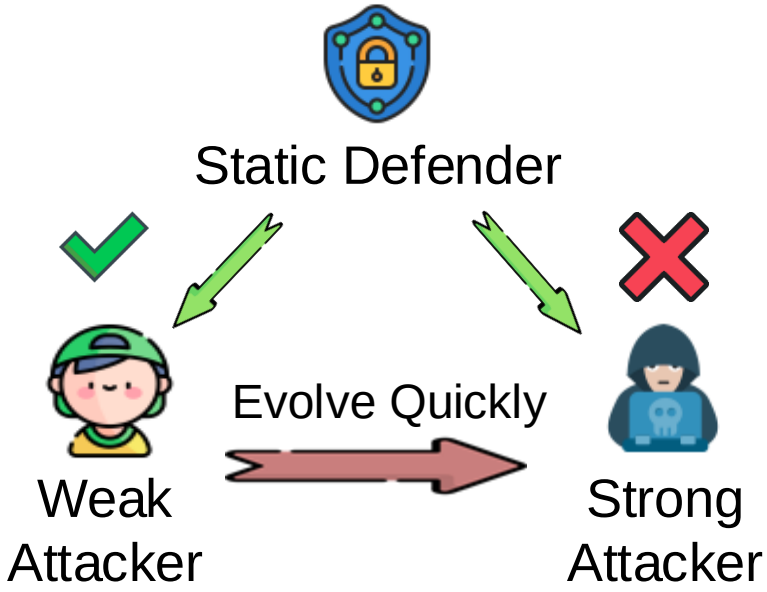
Outlines

1. Introduction
2. Securing Data Integrity
3. Preserving Data Privacy
4. Managing Data Noise
5. Limitations and Opportunities
6. Toolkit

5. Limitations and Opportunity

Lagging of Data Security Defense

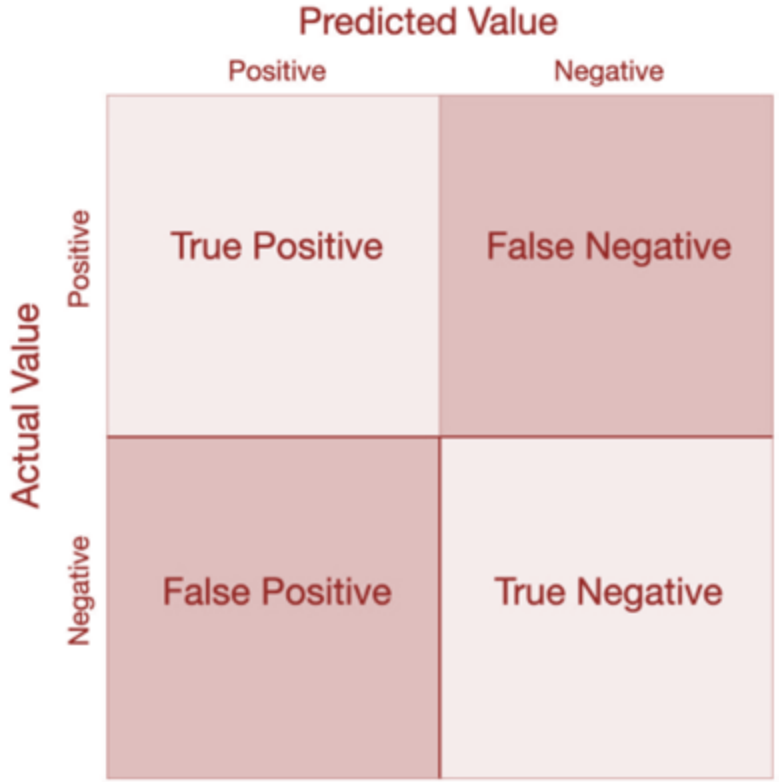
Current defense mechanisms (e.g., anomaly detection) exhibit limited adaptability to evolving attack vectors (e.g., dynamic gradient manipulation, cross-platform collusion), as they primarily rely on pre-defined attack signatures rather than real-time data behavior monitoring.



5. Limitations and Opportunity

Data Quality Degradation via Over-Aggressive Sanitization

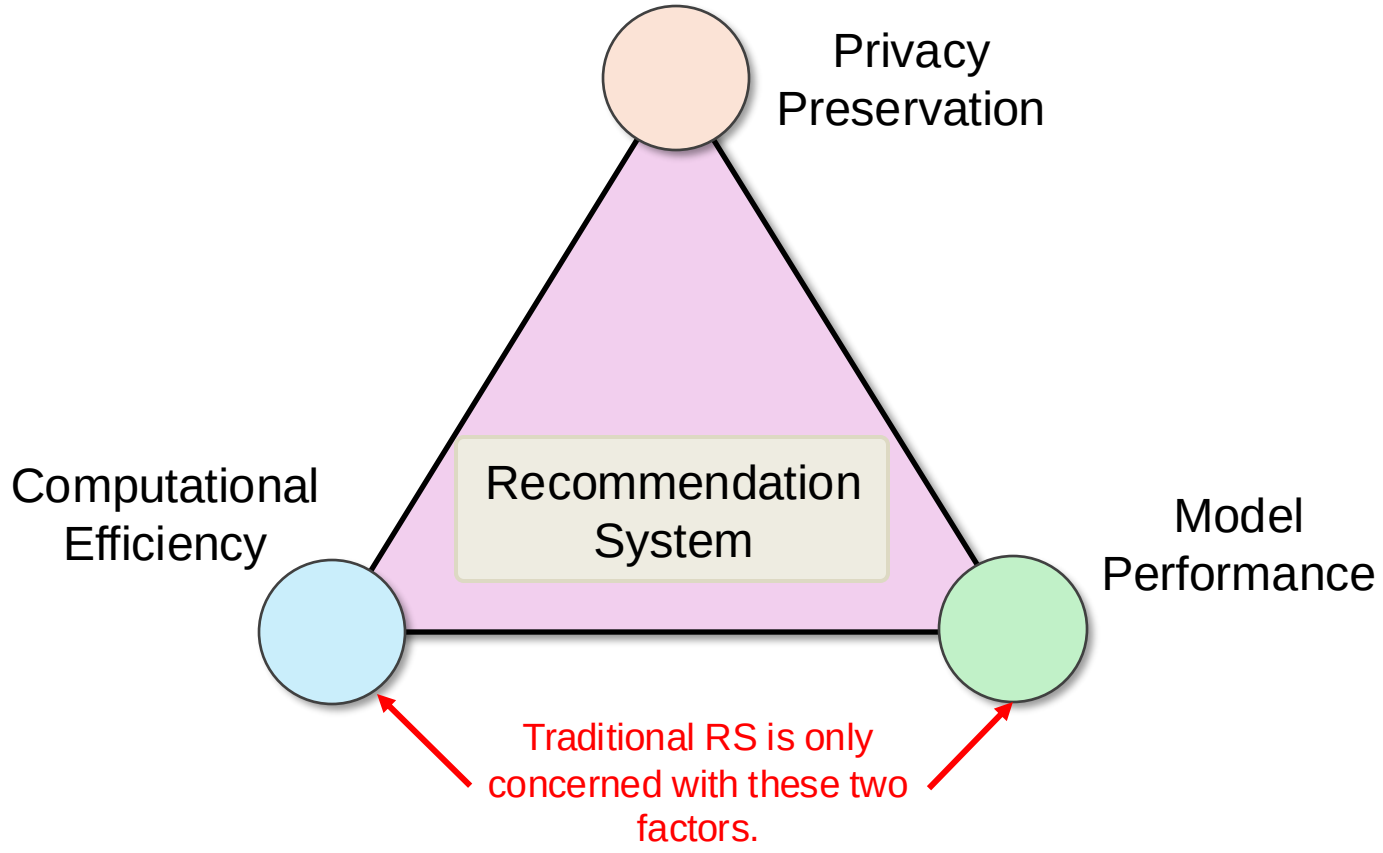
Data cleaning and verification technologies may mistakenly delete high-value data (such as sparse but important user behaviors), resulting in reduced recommendation accuracy.



5. Limitations and Opportunity

The Trade-Off between Data Protection and Recommendation Performance

Protection methods degrade the semantic richness of user interaction data, leading to homogenized recommendations and reduced personalization.



5. Limitations and Opportunity

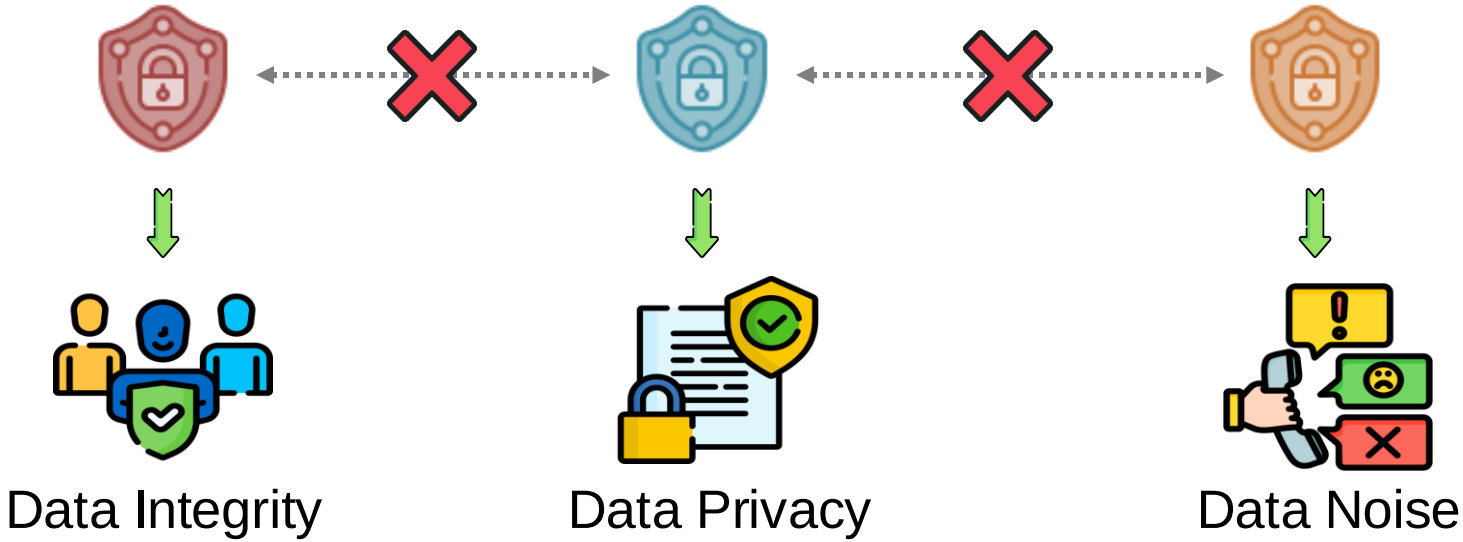
Difficulty in Distinguishing Natural Noise and Malicious Noise

Natural noise (such as user mis-clicks, device failures) and maliciously injected noise (such as adversarial samples, false comments) may be highly similar in behavior patterns, and existing methods are difficult to accurately distinguish.

5. Limitations and Opportunity

Integrity, Privacy, and Noise Issues Are Treated in Isolation

Most studies deal with security attacks (such as poisoning), privacy leaks (such as attribute reasoning), and noise issues in isolation, without considering the correlation between the three.



5. Limitations and Opportunity

Dynamic Data Monitoring and Adaptation

Opportunity 1: Light-Weight and Real-Time Data Trustworthiness Scoring

Design and implement real-time streaming frameworks for evaluating data authenticity by leveraging multimodal indicators, including temporal interaction patterns and cross-platform consistency metrics.

Opportunity 2 : Malicious Data Tracing

Investigate causal inference/watermarking methodologies to differentiate the underlying factors contributing to data anomalies.

5. Limitations and Opportunity

Multiple Data Goal Collaborative Defense

Opportunity: All-in-one Unified Data Representation Learning

Develop adversarial learning frameworks to construct feature representations that simultaneously preserve privacy-sensitive attributes and maintain robustness against adversarial attacks through encoded interaction patterns.

5. Limitations and Opportunity

LLM-Driven Data Curation and Protection

Opportunity 1: Semantic-Aware Data Sanitization

Utilize large language models as contextual filtering mechanisms for detecting sophisticated adversarial content through semantic pattern recognition (e.g., identifying semantically inconsistent reviews that conceal promotional intent).

Opportunity 2: Synthetic Data Augmentation

Employ LLM distillation techniques to synthesize privacy-preserving interaction traces while maintaining behavioral diversity characteristics.

Opportunity 3: Autonomous Data Governance Agents

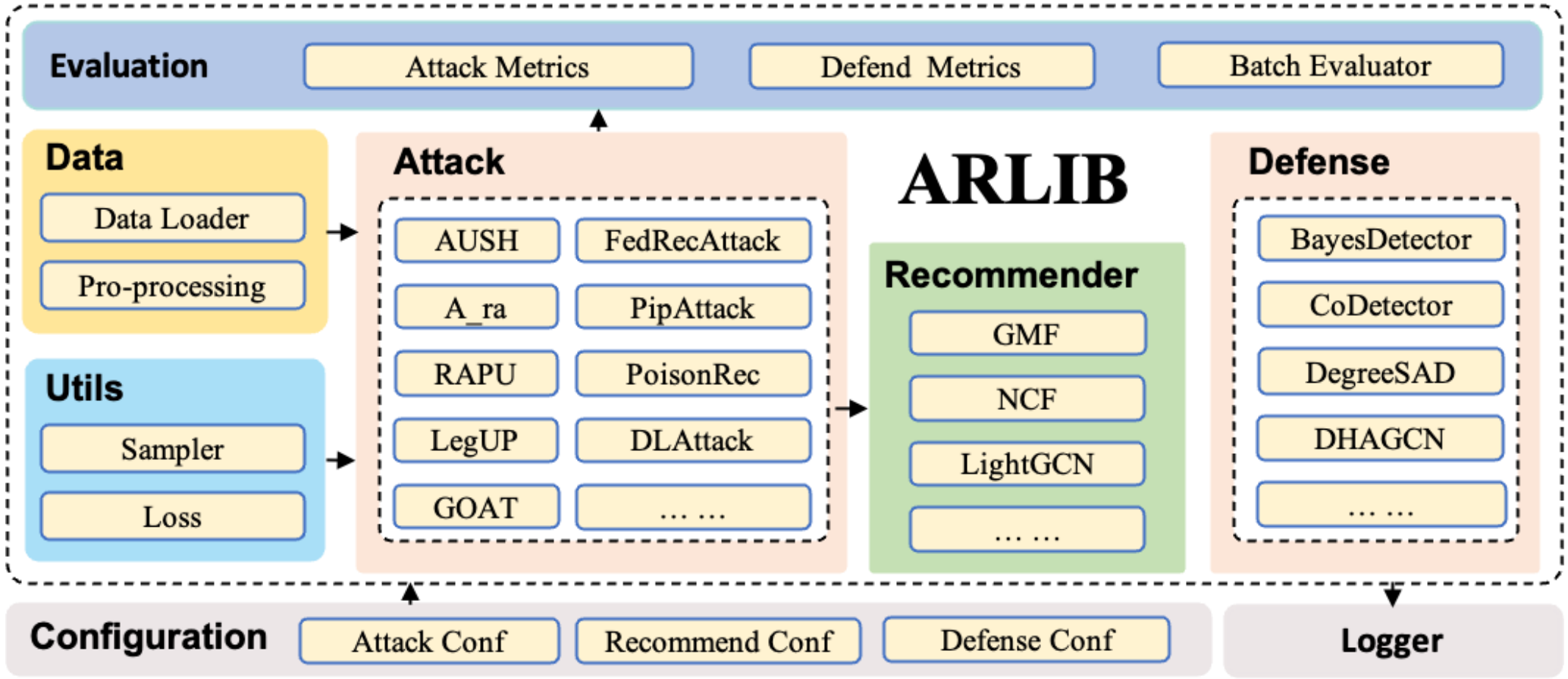
Develop LLM-powered monitoring agents that dynamically adjust data ingestion policies based on real-time trust metrics.

Outlines

1. Introduction
2. Securing Data Integrity
3. Preserving Data Privacy
4. Managing Data Noise
5. Limitations and Opportunities
6. Toolkit

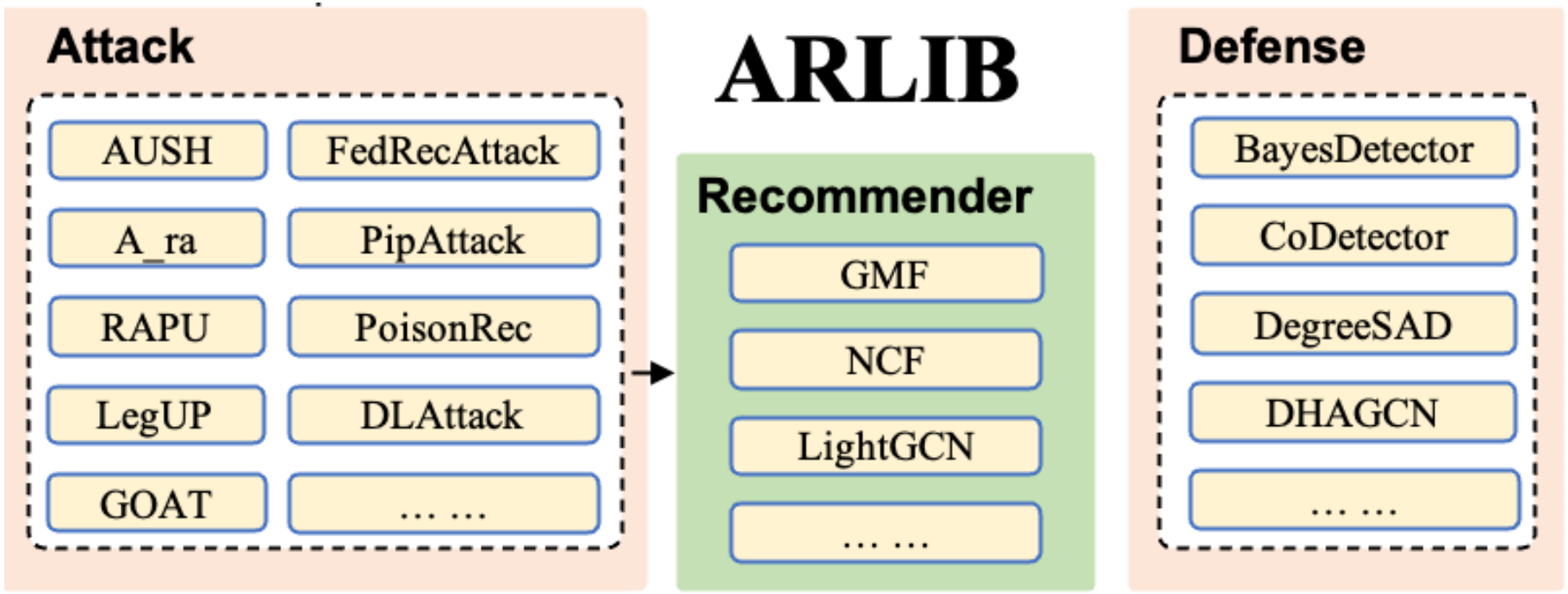
6. Toolkit-ARLib

An open-source framework for conducting data poisoning attacks and corresponding defense on recommendation systems, designed to assist researchers and practitioners.



6. Toolkit-ARLib

Implement **11** recommendation algorithms, **15** attack methods, and **10** detection methods.



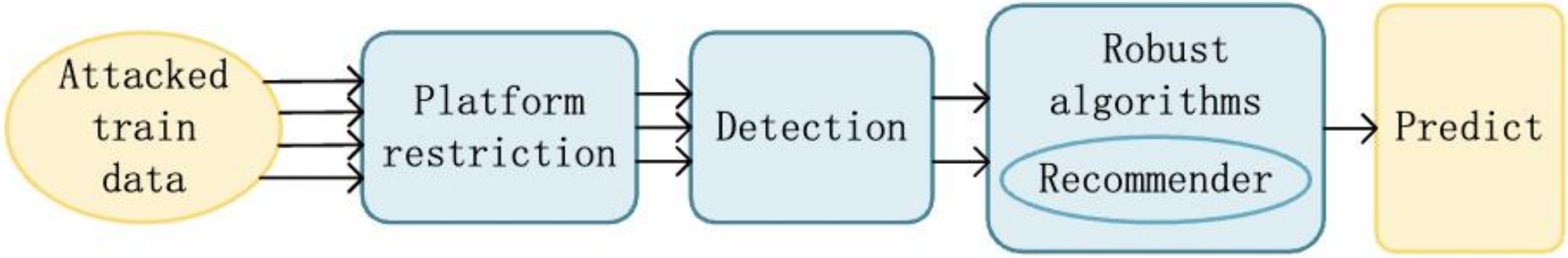
6. Toolkit-ARLib

Features:

- Fast Implementation
- Modularized Architecture

Usages:

- Comprehensive Assessment
- Highly Scalability
- Poisonous Data Simulation



6. Dataset

Dataset	#Users	#Itmes	#Interactions	Sparsity	Average action per user	Average action per item	Timestamp	Social data
Last.fm[10]	1,892	17,632	92,834	99.72%	49.07	5.27	✓	✓
Pinterest[11]	55,187	9,916	1,500,809	99.73%	27.19	151.35		
Gowalla[12]	107,092	1,280,969	6,442,892	99.99%	60.16	5.03	✓	✓
Adressa 2M Compact[13]	15,514	923	2,717,915	81.02%	175.19	2944.65	✓	

Implicit feedback dataset

Dataset	#Users	#Itmes	#Ratings	Rating range	Sparsity	Average action per user	Average action per item	Timestamp	Social data
MovieLens-100K[1]	943	1,682	100,000	1-5	93.70%	106.04	59.45	✓	
MovieLens-1M[1]	6,040	3,706	1,000,209	1-5	95.53%	165.60	269.89	✓	
MovieLens-10M[1]	69,878	10,681	10,000,054	1-5	98.66%	143.11	936.25	✓	
MovieLens-20M[1]	138,493	27,278	20,000,263	0.5-5.0	99.47%	144.41	733.20	✓	
Amazon Review 2014[2]	20,980,000	9,350,000	82,830,000	1-5	99.99%	3.95	8.86	✓	
Yelp[3]	1,987,897	150,346	6,990,280	1-5	99.99%	3.52	46.49	✓	✓
Netflix[4]	480,189	17,770	100,480,507	1-5	98.82%	209.25	5654.50	✓	
Epinions[5]	22,164	296,277	922,267	1-5	99.99%	41.61	3.11	✓	✓
FilmTrust[6]	1,508	2,071	35,497	0.5-4.0	98.86%	23.54	17.14		✓
Ciao[7]	10,877	103,935	269,197	1-5	99.98%	24.75	2.59	✓	✓
CiaoDVD[8]	17,615	16,121	72,665	1-5	99.97%	4.13	4.51	✓	✓
Book-Crossing[9]	105,283	340,556	1,149,780	0-10	99.99%	10.92	3.38		

Explicit feedback dataset

Cheng, Lei, et al. "Towards Robust Recommendation: A Review and an Adversarial Robustness Evaluation Library." *arXiv preprint arXiv:2404.17844* (2024).

重庆大学
CHONGQING UNIVERSITY

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

wsdm 2025

Q&A