Unbiased Rectification for Sequential Recommender Systems Under Fake Orders

Qiyu Qin, Yichen Li, Haozhao Wang, Cheng Wang, Rui Zhang, Ruixuan Li*

School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
{qiyu_qin, ycli0204}@hust.edu.cn, rayteam@yeah.net⟨www.ruizhang.info⟩

Abstract

Fake orders pose increasing threats to sequential recommender systems by misleading recommendation results through artificially manipulated interactions, including click farming, context-irrelevant substitutions, and sequential perturbations. Unlike injecting carefully designed fake users to influence recommendation performance, fake orders embedded within genuine user sequences aim to disrupt user preferences and mislead recommendation results, thereby manipulating exposure rates of specific items to gain competitive advantages. To protect users' authentic interest preferences and eliminate misleading information, this paper aims to perform precise and efficient rectification on compromised sequential recommender systems while avoiding the enormous computational and time costs of retraining existing models. Specifically, we identify that fake orders are not absolutely harmful—in certain cases, partial fake orders can even have a data augmentation effect. Based on this insight, we propose *Dual-view Identification* and *Targeted Rectification (DITaR)*, which primarily identifies harmful samples to achieve unbiased rectification of the system. The core idea of this method is to obtain differentiated representations from collaborative and semantic views for precise detection, and then filters detected suspicious fake orders to select truly harmful ones for targeted rectification with gradient ascent. This ensures that useful information in fake orders is not removed while preventing bias residue. Moreover, it maintains the original data volume and sequence structure, thus protecting system performance and trustworthiness to achieve optimal unbiased rectification. Extensive experiments on three datasets demonstrate that DITaR achieves superior performance compared to state-of-the-art methods in terms of recommendation quality, computational efficiency, and system robustness.

Code — https://github.com/QinWHang/DITaR

Introduction

Sequential recommender systems have become a core component of modern recommendation systems by modeling users' historical interaction sequences to predict their future preferences (Wang et al. 2019a; Fang et al. 2020; Li et al. 2023b). Unlike traditional collaborative filtering methods, sequential recommender systems capture the dynamic changes in user interests and temporal dependencies between items, playing important roles in e-commerce (Singer et al. 2022), social media (Chang et al. 2021), and streaming platforms (Shih, Han, and Tan 2025).

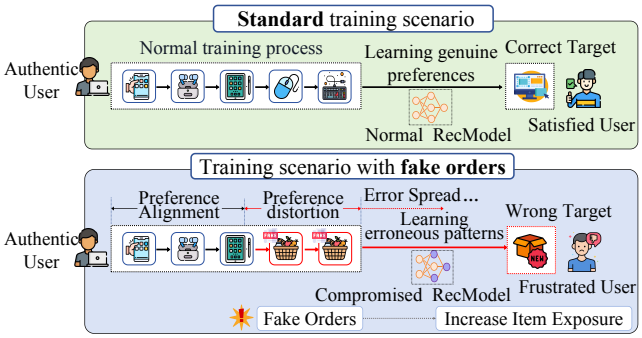


Figure 1: Comparison between standard sequential recommendation process and sequential recommendation process under fake orders. Fake orders alter recommendation results, leading to degraded user experience.

With the growing commercial value of sequential recommender systems, artificially manipulated interaction behaviors generated for specific commercial purposes deserve attention. These behaviors are strategically embedded in genuine user interaction sequences, including click farming, context-irrelevant semantic substitutions, and sequential perturbations, which we term *fake orders*. These covert fake orders steer users towards merchant-preferred items by disrupting the collaborative filtering patterns and semantic associations essential for accurate sequential modeling. This manipulation distorts authentic user preferences, introduces systematic bias, generates erroneous recommendations, and severely damages user trust in the system. Unlike previous adversarial approaches that inject carefully designed fake users to degrade overall system performance (Fan et al. 2021; Wu et al. 2021), fake orders exploit system trust in seemingly legitimate historical users to manipulate specific recommendation results. However, fake orders, by their artificial nature, struggle to maintain consistency across both collaborative and semantic dimensions simultaneously. This inherent limitation creates detectable cross-view discrepancies that enable their identification. Figure.1 illustrates how

*Ruixuan Li is the corresponding author.

fake orders infiltrate genuine user sequences, alter recommendation results, and degrade user experience.

Data rectification has emerged as a promising strategy to eliminate harmful data samples while enabling models to focus on meaningful information. In recommender systems, most rectification methods adopt user-wise or item-wise strategies (Li et al. 2024a). To improve precision, clustering-based approaches shard datasets and use attention-weighted aggregation for targeted retraining (Chen et al. 2022). For efficiency, influence function methods estimate data impact through gradient approximation (Zhang et al. 2024b; Feng et al. 2024). Model-specific strategies also exist, including hierarchical deletion operators for GNNs (Hao et al. 2024; Zhang 2024) and reverse learning that modifies objectives to forget manipulated data (You et al. 2024).

While existing rectification methods achieve promising results by removing noisy data through collaborative filtering approaches, ideal rectification should not simply remove all noisy data. Instead, it should gain deeper insights into the real impact of noisy data, removing truly harmful data while preserving beneficial information. Furthermore, when confronting the specific challenges posed by fake orders, existing methods face significant limitations. **C1:** Most existing methods are designed for collaborative filtering models and struggle with sequential systems where temporal dependencies and evolving user preferences create complex interdependencies that complicate efficient rectification. **C2:** How to achieve sample-wise rectification for fake orders in sequential models. Since fake orders can be injected at low cost without altering data volume, simple removal disrupts data integrity, requiring precise impact elimination while preserving surrounding valid interactions. **C3:** Building on the need for precise rectification, the challenge is how to avoid uniform fake order treatment, quantify their actual impact, and preserve beneficial information while removing harmful samples to achieve efficient unbiased rectification.

To address these challenges, we propose Dual-view Identification and Targeted Rectification (DITaR), an unbiased rectification framework for sequential recommender systems under fake orders. DITaR operates through two complementary stages: The identification phase derives differentiated representations from collaborative and semantic view and detects suspicious fake orders by analyzing cross-view representation inconsistencies and intrinsic attribute anomalies. The rectification phase employs influence function estimation to assess the actual impact of detected fake orders, filtering out truly harmful samples and applying targeted rectification with gradient ascent to achieve optimal unbiased performance.

Extensive experiments on three datasets demonstrate that our proposed DITaR significantly improves recommendation performance and computational efficiency compared to state-of-the-art methods while ensuring system robustness. Our contributions are as follows:

- We are the first to focus on the novel and covert scenario of fake orders embedded within genuine user sequences, which manipulate recommendation outcomes and undermine user trust, posing critical challenges for integrity.
- We propose a dual-view framework that exploits semantic-collaborative representation gaps for fake order identification, coupled with influence-guided filtering and gradient ascent for targeted rectification, achieving unbiased and efficient sample-level rectification.
- We conduct comprehensive experiments demonstrating that DITaR significantly outperforms state-of-the-art baselines across multiple evaluation metrics.

Related Works

Sequential Recommendation. Sequential recommendation systems aim to capture users' dynamic interests by modeling temporal dependencies and preference evolution within their interaction histories (Fan et al. 2022; Li et al. 2023a, 2025a). Recent attention-based models have brought significant advances: SASRec (Kang and McAuley 2018) leverages transformer self-attention for long-term dependencies, while BERT4Rec (Sun et al. 2019) employs bidirectional encoders for comprehensive interest modeling. To achieve more refined representations, recent work integrates multimodal features (Ye et al. 2025; Zhang et al. 2024c; Li et al. 2025b) and leverages large language models to transform item attributes into high-quality semantic embeddings (Harte et al. 2023; Liu et al. 2024; Zhang et al. 2025).

User History Manipulation. While sequential data's multi-layered complexity in collaborative patterns, semantic associations, and temporal dependencies enables precise modeling, it also creates vulnerabilities to manipulation. Recent studies have demonstrated that genuine user interaction histories can be manipulated through documented mechanisms: web injection techniques tamper with in-transit webpage content to force unintended user interactions (Zhang et al. 2019), while substitution-based approaches strategically replace vulnerable sequence elements with targeted items (Yue et al. 2022). Building on these feasibility demonstrations, this paper addresses three specific manipulation types embedded within genuine sequences, collectively termed fake orders.

Rectification Methods. Data rectification aims to eliminate harmful samples' negative impact without prohibitive retraining costs, crucial for maintaining system performance and robustness (Xu et al. 2024; Goel et al. 2024). This technique has seen wide application across domains, including noisy label correction in computer vision (Kodge et al. 2025; Li et al. 2025c), managing large language model lifecycles (Liu et al. 2025; Yao, Xu, and Liu 2024; Li et al. 2024b). Influence functions (Koh and Liang 2017) quantify individual samples' contributions to model performance (Warnecke et al. 2021; Zhang et al. 2024b). Other approaches include gradient-based methods like TracIN (Pruthi et al. 2020), Shapley-based methods (Kwon and Zou 2021; Xia et al. 2023), and clustering (Aytekin et al. 2018) or graph-based methods (Said et al. 2023; Zhang et al. 2024a) that identify anomalous samples. In this paper, we focus on rectification for sequential recommendation, identifying truly harmful items and applying targeted rectification.

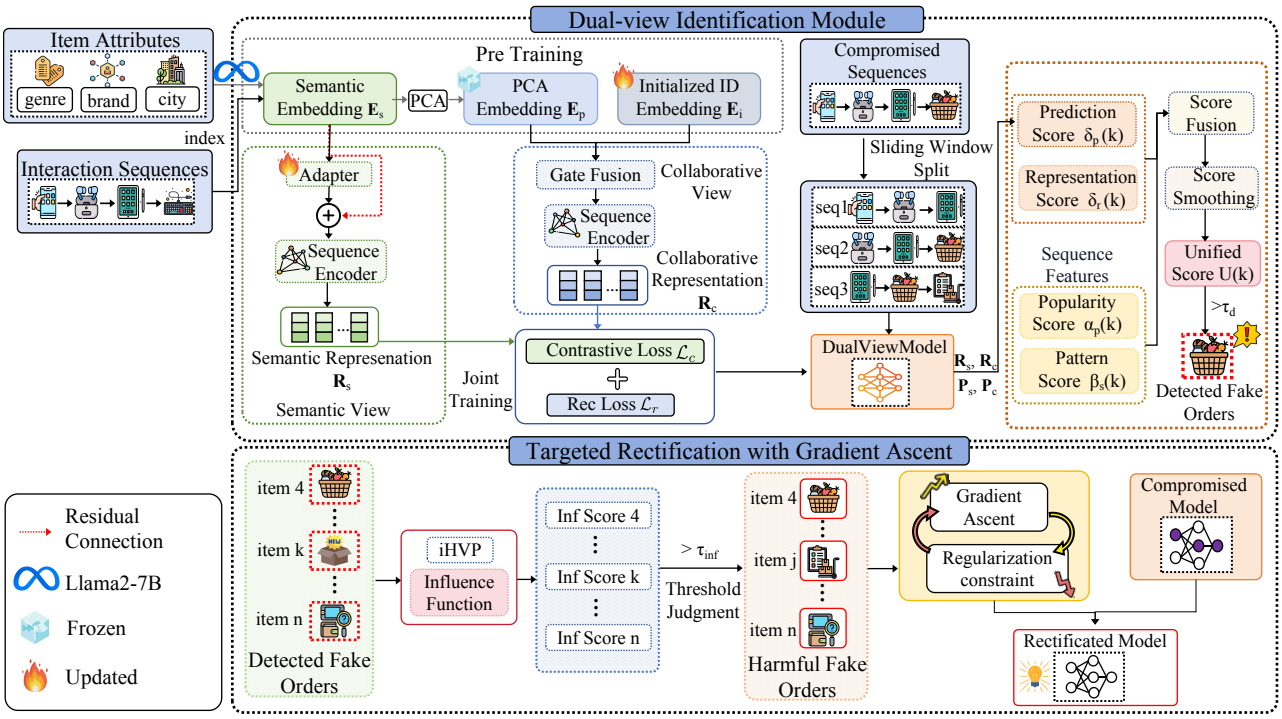


Figure 2: The framework of DITaR. DITaR first constructs collaborative and semantic view models to derive dual-view representations, which are combined with sequence features for fake order detection. Subsequently, influence function assesses the true impact of detected samples on model parameters, enabling selective gradient ascent rectification on harmful fake orders with appropriate regularization to achieve the final rectified model.

Methodology

This section formalizes the definitions of sequential recommendation and Hessian Matrix, then proposes the Dual-view Identification and Targeted Rectification (DITaR) framework. The framework consists of two core components: Dual-view Identification (DI) Module and Targeted Rectification (TaR) with Gradient Ascent. DI identifies fake orders by combining differentiated information from collaborative and semantic views. TaR filters items based on influence function and performs precise rectification on harmful fake orders. Figure.2 illustrates the overall DITaR framework.

Problem Formulation

Sequential Recommendation. Given a user's action sequence $\mathcal{U} = \{u_1, u_2, \ldots, u_m\}$ and item sequence $\mathcal{I} = \{i_1, i_2, \ldots, i_n\}$, sequential recommendation systems aim to predict the next item that user u is likely to interact with, based on the user's historical interaction sequence $S_u = \left[i^u_1, i^u_2, \ldots, i^u_{|S_u|}\right]$. The model parameters θ are trained by minimizing the loss function

$$\theta^* = \arg\min_{\theta} \sum_{u \in \mathcal{U}} \sum_{t=1}^{|S_u|-1} \mathcal{L}(f(S^{1:t}_u; \theta), i^u_{t+1}). \quad (1)$$

Hessian Matrix. Influence function requires computing the product of the inverse Hessian matrix and vector v: $H^{-1}_\theta v$. Given model θ, H_θ represents the Hessian matrix (Thacker 1989), i.e., the second-order derivative matrix of the loss function with respect to model parameters. It is defined as:

$$H_\theta = \nabla^2_\theta \sum_j \mathcal{L}(i_j, \theta). \quad (2)$$

For recommendation systems with large numbers of parameters, directly computing H^{-1}_θ is infeasible. With p as parameter dimension and J as recursion depth, we approximate $H^{-1}_\theta v$ through implicit Hessian-vector products (IHVP) (Koh and Liang 2017), reducing computational complexity from $O\left(p^2\right)$ to $O(p \cdot J)$.

Dual-view Identification Module

We consider complementary information from collaborative and semantic views. The collaborative view reflects statistical patterns of user-item interactions, while the semantic view explores intrinsic associations between items.Fake orders, constrained by their artificial generation mechanisms, struggle to maintain consistency across both dimensions simultaneously, creating detectable cross-view discrepancies that distinguish them from genuine interactions where both views naturally align. Through deep mining of dual-view information and analysis by a unified detection framework, DITaR accurately identifies different types of fake orders.

We learn separated collaborative filtering patterns and semantic association features through representation learning and contrastive decoupling. For the semantic view, item attributes and descriptive texts are organized into complete

prompts and fed into pre-trained LLaMA2-7B (Touvron et al. 2023) to extract semantic embedding $\mathbf{E}_s \in \mathbb{R}^d$, where d is LLaMA2's embedding dimension. To align features between language model and recommendation space, we employ a semantic adapter for feature transformation while preserving original semantics through residual connection. For the collaborative view, we apply Principal Component Analysis (PCA) to $\mathbf{E}_s$ to obtain dimension-reduced embedding $\mathbf{E}_p$, which is then fused with learnable ID embedding $\mathbf{E}_i$ through an adaptive gating mechanism. This mechanism balances collaborative patterns with item-specific personalization. Finally, to preserve distinct characteristics and prevent information leakage, separate sequential encoders f_s and f_c process each view independently to generate disentangled representations $\mathbf{R}_s$ and $\mathbf{R}_c$.

$$\begin{aligned} \mathbf{R}_s &= f_s\left[\text{Adapter}\left(\mathbf{E}_s\right) + \lambda_1 \cdot \left(w_1 \mathbf{E}_s\right)\right], \\ \mathbf{R}_c &= f_c\left[\mathbf{G} \odot \left(w_2 \mathbf{E}_p + b_1\right) + \left(1 - \mathbf{G}\right) \odot \mathbf{E}_i\right]. \end{aligned} \tag{3}$$

where Adapter$(\cdot)$ is a two-layer nonlinear network, λ_1 is the residual coefficient, $\mathbf{G} \in \mathbb{R}^{d_h}$ is the adaptive gating vector, $\odot$ denotes element-wise multiplication, and w_1, w_2, b_1 are learnable parameters. We optimize the model through a joint training objective that combines a recommendation loss with a contrastive loss. Specifically, we employ the InfoNCE loss as the contrastive learning objective to explicitly enforce the independence of the two views. The contrastive loss $\mathcal{L}_c$ is defined as:

$$\begin{aligned} \mathcal{L}_c &= -\frac{1}{2B} \sum_{i=1}^{B} \left[l_{\text{s}}(i) + l_{\text{c}}(i)\right], \\ \text{where } l_{\text{s}}(i) &= \log \frac{\exp\left(\hat{\mathbf{R}}_s(i) \cdot \hat{\mathbf{R}}_c(i)/\tau\right)}{\sum_{j=1}^{B} \exp\left(\hat{\mathbf{R}}_s(i) \cdot \hat{\mathbf{R}}_c(i)/\tau\right)}. \end{aligned} \tag{4}$$

where $\hat{\mathbf{R}} = \mathbf{R}/|\mathbf{R}|_2$ represents ℓ_2 normalization, τ is the temperature coefficient, and B denotes batch size. The final joint optimization objective $\mathcal{L}_t$ combines the recommendation loss $\mathcal{L}_r$ and contrastive loss $\mathcal{L}_c$:

$$\mathcal{L}_t = \alpha \mathcal{L}_r^s + (1 - \alpha)\mathcal{L}_r^c + \lambda_2 \mathcal{L}_c. \tag{5}$$

where α controls the weight between the two views and λ_2 controls the importance of contrastive learning for view independence. We set $\alpha = 0.5$ to balance dual-view contributions equally and $\lambda_2 = 0.1$ to provide moderate contrastive regularization. The joint optimization strategy ensures that dual views maintain feature independence while collaboratively completing recommendation tasks, thereby providing a more discriminative representational basis for detecting manipulated interactions.

Based on the trained dual-view model, we construct a unified analysis framework to identify fake orders. The core principle leverages cross-view information asymmetry: genuine interactions should maintain inherent consistency across collaborative and semantic views, while fake orders, due to their artificial manipulation nature, generate detectable behavioral inconsistency signals across different dimensions. Our detection strategy operates on the insight that fake orders manifest themselves through multiple complementary anomaly patterns. To capture these diverse signals comprehensively, we decompose the detection process into cross-view consistency analysis and intrinsic behavioral pattern analysis. For each interaction i_k in the sequence, we extract its dual-view representations $\mathbf{R}_s(k), \mathbf{R}_c(k)$ and prediction distributions $P_s(k), P_c(k)$ from the trained model. The first component focuses on cross-view inconsistencies that arise when fake orders disrupt the natural alignment between collaborative and semantic signals. We measure representation disagreement $\delta_r(k)$ using cosine similarity and prediction divergence $\delta_p(k)$ through Jensen-Shannon Divergence (Menéndez et al. 1997), as these metrics effectively capture the fundamental conflicts introduced by artificial manipulation. The second component analyzes intrinsic behavioral anomalies by examining popularity patterns and contextual disruptions. We quantify popularity anomaly $\alpha_p(k)$ through statistical z-score deviation and contextual disruption $\beta_s(k)$ via local sequential pattern consistency analysis. Importantly, these intrinsic features help distinguish fake orders from natural behavioral variations such as exploratory clicks: while both may show interest shifts, genuine variations are grounded in historical co-occurrence patterns, whereas fake orders lack such foundations and thus exhibit anomalies across both cross-view and intrinsic dimensions simultaneously. These complementary dimensions are integrated through an adaptive weighting mechanism that learns the relative importance of each signal for fake order detection. Specifically, the multi-dimensional anomaly feature vector $\mathbf{f}_k = [\delta_p(k), \delta_r(k), \alpha_p(k), \beta_s(k)]$ captures these four complementary signals. The unified anomaly score is computed as:

$$u(k) = \mathbf{w}^T \odot \mathcal{N}\left(\mathbf{f}_k\right). \tag{6}$$

where $\mathbf{w}$ is the learned weight vector, and $\mathcal{N}(\cdot)$ denotes score normalization. To enhance robustness against local noise and ensure stable detection performance, we apply temporal smoothing regularization to obtain the final decision score $U(k)$ for each position. An adaptive threshold τ_d is automatically tuned on a validation set to optimize detection performance. This framework achieves precise and robust identification of fake orders through systematic multi-dimensional signal analysis, providing a high-quality candidate set for subsequent influence-based rectification.

Targeted Rectification with Gradient Ascent

Given a compromised recommendation model trained on data containing fake orders, our goal is to rectify it efficiently without incurring the substantial cost of retraining, altering data volume, or causing system instability. To this end, we first leverage influence function to identify truly harmful fake orders, then perform a targeted gradient ascent to precisely neutralize their negative effects. This approach ensures system performance and robustness, achieving our goal of unbiased rectification. We begin with the insight that not all identified fake orders are detrimental to the model's performance. Therefore, for each suspicious interaction i_k in the detected fake orders set I_d, we quantify its true impact on the model's performance on a clean validation set using

influence function (Wu et al. 2023). The influence of an item i_k, denoted as $\mathrm{Inf}(i_k)$, can be expressed as:

$$\mathrm{Inf}\left(i_k\right) = -g_v^T H_\theta^{-1} \nabla_\theta \mathcal{L}\left(i_k, \theta\right). \tag{7}$$

where $\nabla_\theta \mathcal{L}\left(i_k, \theta\right)$ is the loss gradient of the suspicious sample, H_θ is the Hessian matrix, and g_v is the average gradient over the clean validation set. The influence function approximates how removing i_k would affect validation performance. Positive values indicate removing i_k decreases validation loss, thus i_k is harmful; negative values suggest beneficial augmentation effects.

$$I_h = \left\{ i_k \in I_\mathrm{d} : \mathrm{Inf}\left(i_k\right) > \tau_{\text{inf}} \right\}. \tag{8}$$

Typically, τ_{inf} is set to 0, ensuring that only interactions confirmed to be harmful are targeted for rectification, thereby fulfilling the objective of unbiased rectification. To precisely and efficiently remove the negative impact of the identified harmful fake orders I_h, we perform a targeted gradient ascent step. Let θ be the current model parameters. We compute the intermediate parameters θ_m as follows:

$$\theta_m = \theta_t + \eta_1 \sum_{i_k \in I_h} \nabla_\theta \mathcal{L}\left(i_k, \theta_t\right). \tag{9}$$

where η_1 is the rectification learning rate. Meanwhile, we apply regularization constraint by performing one-step gradient descent using clean dataset D_c after each update. With regularization learning rate $\eta_2 \ll \eta_1$, we ensure the model maintains modeling capability for normal recommendation tasks while rectifying harmful information.

$$\theta_{t+1} = \theta_m - \eta_2 \cdot \frac{1}{|D_c|} \sum_{j \in D_c} \nabla_\theta \mathcal{L}\left(j, \theta_m\right). \tag{10}$$

Through alternating optimization strategy and monitoring validation performance to prevent overfitting, we finally output the rectified model θ_r, achieving balance between system security and recommendation performance to accomplish unbiased rectification.

Experiment

In this section, we evaluate our proposed method using three datasets and various baselines. Our analysis assesses DITaR's rectification performance in terms of recommendation effectiveness, efficiency, and robustness. We then investigate the true impact of fake orders and perform ablation studies on DITaR's core detection and rectification modules to demonstrate the overall effectiveness of our approach from multiple perspectives.

Experiment Setup

Datasets. We conduct experiments on three datasets: MovieLens-1M (ML-1M) (Harper and Konstan 2015), Amazon-Beauty, and Yelp2018 (Wang et al. 2019b). Following (Liu et al. 2024), we apply standard preprocessing to filter out users and items with insufficient interactions. We design three fake order scenarios corresponding to the manipulation types introduced earlier: (1) Repetitive orders simulate click farming by repeating a selected item for k subsequent positions, disrupting collaborative filtering patterns; (2) Semantic orders replace target items with semantically irrelevant items, breaking semantic consistency; (3) Sequential orders alter the interaction order of two non-adjacent items within a specified window, corrupting temporal dependencies. The number of affected items is controlled by user ratio (proportion of affected users) and intensity (proportion of fake orders per affected user). Fake orders replace genuine interactions without changing sequence length, and we use mutually exclusive allocation to ensure only one type of fake order is applied per position, guaranteeing detection reliability by avoiding interference between different types. Following the dataset configuration of SASRec (Kang and McAuley 2018), we adopt a leave-one-out evaluation protocol where the last item in each user's interaction sequence constitutes the test set, while the second-to-last item forms the validation set. For evaluation, we randomly sample negative items that users have not interacted with, which are then paired with the ground truth positive item to compute ranking-based metrics. Dataset statistics and fake order configurations are presented in Table 1.

Dataset	#Users	#Items	#Interactions	#Fake Orders
ML-1M	6,040	3,416	999,611	140,258 (14.03%)
Amazon-Beauty	22,363	12,101	198,502	23,603 (11.89%)
Yelp2018	213,170	94,304	3,277,932	43,165 (13.18%)

Table 1: Dataset statistics and Fake Orders settings with user ratio=0.3, intensity =0.3.

Baselines & Recommendation Models. We select four rectification methods as baselines: Retrain, SISA (Bourtoule et al. 2021), RecEraser (Chen et al. 2022), and UltraRE (Li et al. 2023c). Retrain removes fake orders and retrains from scratch; SISA utilizes sharding aggregation by dividing data into shards and averaging predictions from all sub-models to obtain aggregated scores; RecEraser introduces clustering for diversified shard partitioning with attention-weighted aggregation; UltraRE extends RecEraser using Optimal Balanced Clustering (OBC) and simplifies aggregation logic. We set the number of shards to 10 following original implementations. These baselines, primarily designed for collaborative filtering, are adapted to sequential recommendation tasks. We implement these baselines and our method on classic sequential recommendation models including SASRec (Kang and McAuley 2018), GRU4Rec (Hidasi et al. 2015), and BERT4Rec (Sun et al. 2019).

Configurations. Fake orders are configured in two groups: (1) 30% user ratio with 30% intensity, and (2) 60% user ratio with 60% intensity, where each setting includes all three fake order types. For each baseline model, we configure according to original paper recommendations. For evaluation metrics, we use Hit Rate (HR)@k and NDCG@k with default k values of 10 and 20. To evaluate rectification efficiency and avoid differences caused by different training parameter designs, we use convergence epochs for assessment. For experimental robustness, we report average results from two independent runs with random seeds $\{42, 43\}$.

Dataset	Method	SASRec				GRU4Rec				Bert4Rec			
		N@10	H@10	N@20	H@20	N@10	H@10	N@20	H@20	N@10	H@10	N@20	H@20
ML-1M	Original	0.2449	0.4329	0.2908	0.6154	0.2440	0.4288	0.2876	0.6025	0.2790	0.5111	0.3306	0.7152
	Retrain	0.2406	0.4311	0.2829	0.5988	0.2365	0.4207	0.2806	0.5960	0.2666	0.4975	0.3154	0.6902
	SISA	0.1089	0.2185	0.1402	0.3443	0.0818	0.1523	0.1110	0.2682	0.2411	0.4404	0.2871	0.6242
	RecEraser	0.2035	0.3813	0.2483	0.5598	0.1803	0.3356	0.2235	0.5075	0.2355	0.4296	0.2786	0.6013
	UltraRE	0.2120	0.3732	0.2526	0.5348	0.1104	0.1907	0.1381	0.3012	0.2440	0.4434	0.2895	0.6238
	DITaR(ours)	**0.2470**	**0.4386**	**0.2893**	**0.6076**	**0.2448**	**0.4369**	**0.2854**	**0.5978**	**0.2763**	**0.5036**	**0.3267**	**0.7025**
Amazon-Beauty	Original	0.2036	0.3356	0.2344	0.4583	0.1809	0.3159	0.2140	0.4470	0.2029	0.3601	0.2361	0.4918
	Retrain	0.1971	0.3320	0.2268	0.4502	0.1735	0.3087	0.2054	0.4352	0.1912	0.3462	0.2243	0.4777
	SISA	0.1023	0.1999	0.1318	0.3173	0.1040	0.1989	0.1338	0.3168	0.1232	0.2179	0.1573	0.3527
	RecEraser	0.1817	0.3192	0.2140	0.4475	0.1630	0.2890	0.1937	0.4114	0.1779	0.3124	0.2082	0.4327
	UltraRE	0.1768	0.3113	0.2092	0.4396	0.1724	0.3050	0.2044	0.4318	0.1593	0.2969	0.1948	0.4380
	DITaR(ours)	**0.2032**	**0.3364**	**0.2331**	**0.4555**	**0.1725**	**0.3070**	**0.2060**	**0.4400**	**0.1971**	**0.3523**	**0.2300**	**0.3527**
Yelp2018	Original	0.3161	0.5155	0.3485	0.6433	0.3174	0.5187	0.3509	0.6510	0.4034	0.6453	0.4389	0.7851
	Retrain	0.3069	0.4932	0.3373	0.6130	0.2734	0.4547	0.3067	0.5870	0.3869	0.6225	0.4231	0.7650
	SISA	0.2009	0.3609	0.2349	0.4948	0.2070	0.3449	0.2369	0.4632	0.2968	0.4971	0.3333	0.6417
	RecEraser	0.2753	0.4634	0.3098	0.6000	0.2327	0.3974	0.2652	0.5259	0.3296	0.5415	0.3670	0.6891
	UltraRE	0.2795	0.4553	0.3105	0.5783	0.2881	0.4746	0.3224	0.6102	0.3515	0.5673	0.3861	0.7039
	DITaR(ours)	**0.3128**	**0.5003**	**0.3434**	**0.6215**	**0.3106**	**0.5072**	**0.3441**	**0.6397**	**0.3924**	**0.6309**	**0.4282**	**0.7720**

Table 2: Performance comparison of various methods for the top-k recommendation task. The best results are bold.

Performance Overall

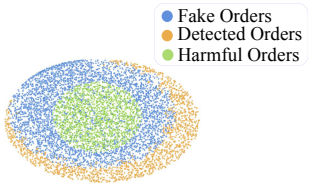
Rectification Results. We evaluate the effectiveness, robustness, and efficiency of DITaR. Table2 demonstrates the rectification performance of different methods across three datasets. DITaR consistently outperforms baseline methods and achieves performance comparable to retraining, occasionally even surpassing the original clean data performance. This indicates that DITaR successfully balances data preservation by removing harmful components while retaining beneficial information for high-quality rectification. The performance gap between retraining and the original method reveals the critical impact of data loss in recommender systems. As fake order injection frequency and intensity increase, this degradation becomes more severe, highlighting the limitations of simple data removal approaches. Existing sharding-based methods attempt to balance rectification efficiency and accuracy through partitioning and model aggregation. However, these approaches face inherent limitations. Sharding inevitably disrupts collaborative relationships between users, while aggregation introduces approximation errors that cannot fully recover the original collaborative signals. Moreover, these methods were primarily designed for traditional collaborative filtering models and struggle with the complex temporal dependencies in sequential data. Their uniform treatment of all detected samples overlooks the heterogeneous nature of fake orders, failing to distinguish between harmful and potentially beneficial instances.

In contrast, DITaR maintains performance levels close to the original clean data with minimal fluctuation, demonstrating strong resistance to fake order intrusions while preserving both system robustness and recommendation quality. The framework's consistent performance across different datasets and model architectures validates its effectiveness and demonstrates stability compared to baseline methods.

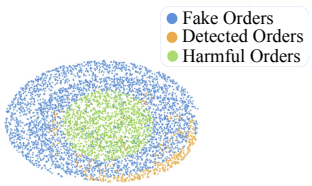
Dataset	Method	Model		
		SASRec	GRU4Rec	Bert4Rec
ML-1M	Retrain	140	100	130
	SISA	60	80	67
	RecEraser	52	55	50
	UltraRE	35	38	43
	DITaR(ours)	5	5	5
Amazon-Beauty	Retrain	175	110	125
	SISA	86	78	66
	RecEraser	43	50	51
	UltraRE	38	45	36
	DITaR(ours)	5	5	5
Yelp2018	Retrain	145	135	180
	SISA	119	81	120
	RecEraser	55	57	60
	UltraRE	43	55	50
	DITaR(ours)	5	5	5

Table 3: Convergence performance of different rectification methods under fake orders (user ratio = 0.3, intensity = 0.3).

Table 3 compares the computational efficiency of different methods using convergence epochs as a fair evaluation metric. For sharding-based methods, we report the rounded average convergence epochs of sub-models across different shards with evaluation performed every 5 epochs. DITaR demonstrates significantly fewer convergence epochs compared to baseline methods. While sharding-based approaches may achieve faster training on individual small models due to reduced data volume, their overall efficiency

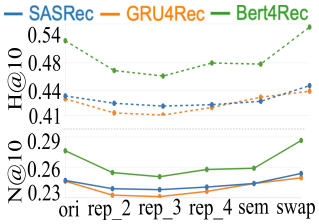


(a) Detection with SASRec on ML-1M.

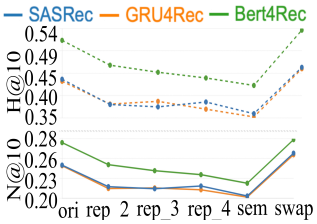


(b) Detection with Bert4Rec on Amazon-Beauty.

Figure 3: Fake order detection performance of two sequential recommendation models across different datasets.



(a) Fake Orders Effects on ML-1M with user ratio=0.3, intensity=0.3.



(b) Fake Orders Effects on ML-1M with user ratio=0.6, intensity=0.6.

Figure 4: Impact of fake order types on recommendation performance (ori: original data; rep_X: repetitive orders with length X; sem: semantic orders; swap: sequential orders).

decreases as the number of shards increases, creating a trade-off between training efficiency and accuracy. Moreover, more uniform sharding strategies lead to smoother convergence across sub-models, with similar convergence epochs among different shards. DITaR benefits from its targeted rectification paradigm, requiring only gradient ascent fine-tuning on pre-trained models rather than training from scratch, thus achieving superior computational efficiency.

Detection Effectiveness. Figure.3 illustrates the performance of our detection module. Through dual-view modeling and the unified detection framework, our method gains deep insights into the information disruptions caused by fake orders from sequences' collaborative patterns and semantic coherence. The results demonstrate that DITaR achieves high precision while maintaining strong recall, successfully capturing the majority of fake orders despite occasional false positives or missed detections. This provides a reliable candidate set for subsequent targeted rectification, establishing a solid foundation for precise fake order removal.

Real Impact of Fake Orders. To validate our core insight that "not all fake orders are harmful," we analyze the individual impact of different fake order types. Figure.4 reveals distinct effects across the three scenarios. Repetitive orders demonstrate performance fluctuations as repeat length increases, with moderate repetition potentially reinforcing certain interaction patterns. Semantic orders consistently degrade system performance due to their disruption of semantic coherence, showing pronounced negative effects. Most notably, sequential order swapping demonstrates a beneficial effect on system performance. This counterintuitive finding can be attributed to the fact that swapping non-adjacent items introduces controlled temporal noise that acts as implicit data augmentation, helping the model learn more robust sequential patterns and reducing overfitting to rigid temporal dependencies. This regularization effect enhances the model's generalization capability, particularly for handling naturally occurring temporal variations in user behavior. The impact magnitude correlates positively with both user ratio and intensity parameters, indicating that widespread and frequent fake order injections pose greater threats to system integrity and warrant heightened attention.

Ablation Study. As shown in Table 4, we evaluate the complementary effects of dual views and influence function filtering capabilities. Removing either view leads to decreased final rectification effectiveness, with collaborative view removal showing more significant impact. This demonstrates that collaborative filtering patterns and temporal dependencies remain the core of sequential modeling, while semantic information serves as crucial complementary features that enhance sequential representation learning. Additionally, to ensure high recall value, our detection module uses relatively lenient thresholds, detecting more candidate items than actual fake orders. Without influence function filtering, directly applying gradient ascent to all detected candidates would inadvertently harm beneficial or neutral items, leading to performance degradation. This highlights the indispensable role of influence function estimation in distinguishing harmful from beneficial fake orders, thereby achieving truly unbiased rectification. Nevertheless, dual-view detection without influence filtering still outperforms single-view approaches, further validating the superior detection capability of our dual-view framework.

	SASRec		GRU4Rec		Bert4Rec	
Method	N@10	H@10	N@10	H@10	N@10	H@10
DITaR	0.2032	0.3364	0.1725	0.3070	0.1971	0.3523
-w/o Semantic View	0.1865	0.3108	0.1566	0.2788	0.1802	0.3226
-w/o Collaborative View	0.1829	0.3061	0.1534	0.2736	0.1753	0.3153
-w/o Influence Function	0.1941	0.3240	0.1641	0.2924	0.1862	0.3338

Table 4: Ablation study results on Amazon-Beauty dataset across SASRec, GRU4Rec, and BERT4Rec models.

Conclusion

In this paper, we propose DITaR, an efficient unbiased rectification framework for sequential recommender systems under fake orders. Our approach exploits dual-view modeling to identify fake orders through cross-view discrepancies, then leverages influence function to selectively rectify genuinely harmful samples while preserving beneficial information. This enables precise sample-level rectification without retraining or altering data structures. Comprehensive experiments demonstrate that DITaR significantly outperforms the state-of-the-art methods across multiple evaluation metrics.

Acknowledgments

This work is supported by the National Key Research and Development Program of China under grant 2024YFC3307900; the National Natural Science Foundation of China under grants 62376103, 62302184 and 62436003; Major Science and Technology Project of Hubei Province under grant 2025BAB011 and 2024BAA008; and Hubei Science and Technology Talent Service Project under grant 2024DJC078.

References

Aytekin, C.; Ni, X.; Cricri, F.; and Aksu, E. 2018. Clustering and unsupervised anomaly detection with l 2 normalized deep auto-encoder representations. In *2018 International Joint Conference on Neural Networks (IJCNN)*, 1–6. IEEE.

Bourtoule, L.; Chandrasekaran, V.; Choquette-Choo, C. A.; Jia, H.; Travers, A.; Zhang, B.; Lie, D.; and Papernot, N. 2021. Machine unlearning. In *2021 IEEE symposium on security and privacy (SP)*, 141–159. IEEE.

Chang, J.; Gao, C.; Zheng, Y.; Hui, Y.; Niu, Y.; Song, Y.; Jin, D.; and Li, Y. 2021. Sequential recommendation with graph neural networks. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 378–387.

Chen, C.; Sun, F.; Zhang, M.; and Ding, B. 2022. Recommendation unlearning. In *Proceedings of the ACM web conference 2022*, 2768–2777.

Fan, W.; Derr, T.; Zhao, X.; Ma, Y.; Liu, H.; Wang, J.; Tang, J.; and Li, Q. 2021. Attacking black-box recommendations via copying cross-domain user profiles. In *2021 IEEE 37th international conference on data engineering (ICDE)*, 1583–1594. IEEE.

Fan, Z.; Liu, Z.; Wang, Y.; Wang, A.; Nazari, Z.; Zheng, L.; Peng, H.; and Yu, P. S. 2022. Sequential recommendation via stochastic self-attention. In *Proceedings of the ACM web conference 2022*, 2036–2047.

Fang, H.; Zhang, D.; Shu, Y.; and Guo, G. 2020. Deep learning for sequential recommendation: Algorithms, influential factors, and evaluations. *ACM Transactions on Information Systems (TOIS)*, 39(1): 1–42.

Feng, X.; Chen, C.; Li, Y.; and Lin, Z. 2024. Fine-grained pluggable gradient ascent for knowledge unlearning in language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 10141–10155.

Goel, S.; Prabhu, A.; Torr, P.; Kumaraguru, P.; and Sanyal, A. 2024. Corrective machine unlearning. *arXiv preprint arXiv:2402.14015*.

Hao, Y.; Zhuang, F.; Wang, D.; Liu, G.; Sheng, V. S.; and Zhao, P. 2024. A general strategy graph collaborative filtering for recommendation unlearning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 799–808.

Harper, F. M.; and Konstan, J. A. 2015. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4): 1–19.

Harte, J.; Zorgdrager, W.; Louridas, P.; Katsifodimos, A.; Jannach, D.; and Fragkoulis, M. 2023. Leveraging large language models for sequential recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, 1096–1102.

Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; and Tikk, D. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, 197–206. IEEE.

Kodge, S.; Ravikumar, D.; Saha, G.; and Roy, K. 2025. SAP: Corrective Machine Unlearning with Scaled Activation Projection for Label Noise Robustness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 17930–17937.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *International conference on machine learning*, 1885–1894. PMLR.

Kwon, Y.; and Zou, J. 2021. Beta shapley: a unified and noise-reduced data valuation framework for machine learning. *arXiv preprint arXiv:2110.14049*.

Li, J.; Wang, M.; Li, J.; Fu, J.; Shen, X.; Shang, J.; and McAuley, J. 2023a. Text is all you need: Learning language representations for sequential recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1258–1267.

Li, M.; Zhang, Z.; Zhao, X.; Wang, W.; Zhao, M.; Wu, R.; and Guo, R. 2023b. Automlp: Automated mlp for sequential recommendations. In *Proceedings of the ACM web conference 2023*, 1190–1198.

Li, Y.; Chen, C.; Zhang, Y.; Liu, W.; Lyu, L.; Zheng, X.; Meng, D.; and Wang, J. 2023c. Ultrare: Enhancing receraser for recommendation unlearning via error decomposition. *Advances in Neural Information Processing Systems*, 36: 12611–12625.

Li, Y.; Feng, X.; Chen, C.; and Yang, Q. 2024a. A survey on recommendation unlearning: Fundamentals, taxonomy, evaluation, and open questions. *arXiv preprint arXiv:2412.12836*.

Li, Y.; Qin, Q.; Zhu, G.; Xu, W.; Wang, H.; Li, Y.; Zhang, R.; and Li, R. 2025a. A Systematic Survey on Federated Sequential Recommendation. arXiv:2504.05313.

Li, Y.; Shan, Y.; Liu, Y.; Wang, H.; Wang, W.; Wang, Y.; and Li, R. 2025b. Personalized Federated Recommendation for Cold-Start Users via Adaptive Knowledge Fusion. In *Proceedings of the ACM on Web Conference 2025*, 2700–2709.

Li, Y.; Wang, H.; Xu, W.; Xiao, T.; Liu, H.; Tu, M.; Wang, Y.; Yang, X.; Zhang, R.; Yu, S.; Guo, S.; and Li, R. 2024b. Unleashing the Power of Continual Learning on Non-Centralized Devices: A Survey. arXiv:2412.13840.

Li, Y.; Wang, Y.; Wang, H.; Qi, Y.; Xiao, T.; and Li, R. 2025c. FedSSI: Rehearsal-Free Continual Federated Learning with Synergistic Synaptic Intelligence. In *Forty-second International Conference on Machine Learning*.

Liu, Q.; Wu, X.; Wang, Y.; Zhang, Z.; Tian, F.; Zheng, Y.; and Zhao, X. 2024. Llm-esr: Large language models enhancement for long-tailed sequential recommendation. *Advances in Neural Information Processing Systems*, 37: 26701–26727.

Liu, S.; Yao, Y.; Jia, J.; Casper, S.; Baracaldo, N.; Hase, P.; Yao, Y.; Liu, C. Y.; Xu, X.; Li, H.; et al. 2025. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, 1–14.

Menéndez, M. L.; Pardo, J. A.; Pardo, L.; and Pardo, M. d. C. 1997. The jensen-shannon divergence. *Journal of the Franklin Institute*, 334(2): 307–318.

Pruthi, G.; Liu, F.; Kale, S.; and Sundararajan, M. 2020. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33: 19920–19930.

Said, A.; Zhao, Y.; Derr, T.; Shabbir, M.; Abbas, W.; and Koutsoukos, X. 2023. A survey of graph unlearning. *arXiv preprint arXiv:2310.02164*.

Shih, K.; Han, Y.; and Tan, L. 2025. Recommendation system in advertising and streaming media: Unsupervised data enhancement sequence suggestions. *arXiv preprint arXiv:2504.08740*.

Singer, U.; Roitman, H.; Eshel, Y.; Nus, A.; Guy, I.; Levi, O.; Hasson, I.; and Kiperwasser, E. 2022. Sequential modeling with multiple attributes for watchlist recommendation in e-commerce. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, 937–946.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 1441–1450.

Thacker, W. C. 1989. The role of the Hessian matrix in fitting models to measurements. *Journal of Geophysical Research: Oceans*, 94(C5): 6177–6196.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, S.; Hu, L.; Wang, Y.; Cao, L.; Sheng, Q. Z.; and Orgun, M. 2019a. Sequential recommender systems: challenges, progress and prospects. *arXiv preprint arXiv:2001.04830*.

Wang, X.; He, X.; Wang, M.; Feng, F.; and Chua, T.-S. 2019b. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, 165–174.

Warnecke, A.; Pirch, L.; Wressnegger, C.; and Rieck, K. 2021. Machine unlearning of features and labels. *arXiv preprint arXiv:2108.11577*.

Wu, C.; Lian, D.; Ge, Y.; Zhu, Z.; and Chen, E. 2021. Triple adversarial learning for influence based poisoning attack in recommender systems. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 1830–1840.

Wu, C.; Lian, D.; Ge, Y.; Zhu, Z.; and Chen, E. 2023. Influence-driven data poisoning for robust recommender systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10): 11915–11931.

Xia, H.; Liu, J.; Lou, J.; Qin, Z.; Ren, K.; Cao, Y.; and Xiong, L. 2023. Equitable data valuation meets the right to be forgotten in model markets. *Proceedings of the VLDB Endowment*, 16(11).

Xu, J.; Wu, Z.; Wang, C.; and Jia, X. 2024. Machine unlearning: Solutions and challenges. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 8(3): 2150–2168.

Yao, Y.; Xu, X.; and Liu, Y. 2024. Large language model unlearning. *Advances in Neural Information Processing Systems*, 37: 105425–105475.

Ye, Y.; Zheng, Z.; Shen, Y.; Wang, T.; Zhang, H.; Zhu, P.; Yu, R.; Zhang, K.; and Xiong, H. 2025. Harnessing multimodal large language models for multimodal sequential recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 13069–13077.

You, X.; Xu, J.; Zhang, M.; Gao, Z.; and Yang, M. 2024. Rrl: Recommendation reverse learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 9296–9304.

Yue, Z.; Zeng, H.; Kou, Z.; Shang, L.; and Wang, D. 2022. Defending substitution-based profile pollution attacks on sequential recommenders. In *Proceedings of the 16th ACM Conference on Recommender Systems*, 59–70.

Zhang, C.; Long, G.; Zhou, T.; Zhang, Z.; Yan, P.; and Yang, B. 2024a. Gpfedrec: Graph-guided personalization for federated recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4131–4142.

Zhang, J. 2024. Graph unlearning with efficient partial retraining. In *Companion Proceedings of the ACM Web Conference 2024*, 1218–1221.

Zhang, Y.; Hu, Z.; Bai, Y.; Wu, J.; Wang, Q.; and Feng, F. 2024b. Recommendation unlearning via influence function. *ACM Transactions on Recommender Systems*, 3(2): 1–23.

Zhang, Y.; Xiao, J.; Hao, S.; Wang, H.; Zhu, S.; and Jajodia, S. 2019. Understanding the manipulation on recommender systems through web injection. *IEEE Transactions on Information Forensics and Security*, 15: 3807–3818.

Zhang, Z.; Liu, S.; Liu, Z.; Zhong, R.; Cai, Q.; Zhao, X.; Zhang, C.; Liu, Q.; and Jiang, P. 2025. Llm-powered user simulator for recommender system. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 13339–13347.

Zhang, Z.; Liu, S.; Yu, J.; Cai, Q.; Zhao, X.; Zhang, C.; Liu, Z.; Liu, Q.; Zhao, H.; Hu, L.; et al. 2024c. M3oe: Multi-domain multi-task mixture-of experts recommendation framework. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 893–902.