Multi-Modal Recommendation Unlearning for Legal, Licensing, and Modality Constraints

Yash Sinha[1], **Murari Mandal**[2], **Mohan Kankanhalli**[1]

[1]School of Computing, National University of Singapore
[2]RespAI Lab, School of Computer Engineering, KIIT Bhubaneswar, India
yashsinha@comp.nus.edu.sg, murari.mandalfcs@kiit.ac.in, mohan@comp.nus.edu.sg

Abstract

User data spread across multiple modalities has popularized multi-modal recommender systems (MMRS). They recommend diverse content such as products, social media posts, TikTok reels, etc., based on a user-item interaction graph. With rising data privacy demands, recent methods propose unlearning private user data from uni-modal recommender systems (RS). However, methods for unlearning item data related to outdated user preferences, revoked licenses, and legally requested removals are still largely unexplored.

Previous RS unlearning methods are unsuitable for MMRS due to the incompatibility of their matrix-based representation with the multi-modal user-item interaction graph. Moreover, their data partitioning step degrades performance on each shard due to poor data heterogeneity and requires costly performance aggregation across shards.

This paper introduces MMRecUn, the first approach known to us for unlearning in MMRS and unlearning item data. Given a trained RS model, MMRecUn employs a novel Reverse Bayesian Personalized Ranking (BPR) objective to enable the model to forget marked data. The reverse BPR attenuates the impact of user-item interactions within the forget set, while the forward BPR reinforces the significance of user-item interactions within the retain set. Our experiments demonstrate that MMRecUn outperforms baseline methods across various unlearning requests when evaluated on benchmark MMRS datasets. MMRecUn achieves recall performance improvements of up to 49.85% compared to baseline methods and is up to 1.3× faster than the Gold model, which is trained on retain set from scratch. MMRecUn offers significant advantages, including superiority in removing target interactions, preserving retained interactions, and zero overhead costs compared to previous methods.

Code — https://github.com/MachineUnlearn/MMRecUN
Extended version — https://arxiv.org/abs/2405.15328

1 Introduction

Recommender systems (RS) (Smith and Linden 2017; He et al. 2020) leverage techniques like collaborative filtering, content-based filtering, and neural networks to recommend diverse content such as movies, music, products, news articles, and more, thereby boosting user engagement and satisfaction. As user data increasingly spans multiple modalities (Covington, Adams, and Sargin 2016), multi-modal recommender systems (MMRS) are gaining traction (Wu et al. 2022; Zhou et al. 2023). Recent advancements (Yu et al. 2023; Liu et al. 2023b; Wei et al. 2020, 2019; Xu et al. 2018) have focused on capturing user preferences through both behavior data and diverse multi-modal item information. With growing concerns over data privacy, regulations like GDPR (Voigt and Von dem Bussche 2017) emphasize the importance of data protection and the "right to be forgotten." While significant progress has been made in unlearning user data to enhance privacy (Chen et al. 2022; Li et al. 2023a), the unlearning of item data remains largely unexplored. This paper introduces MMRecUn, which, to the best of our knowledge, is the first attempt to address unlearning in MMRS and unlearning item data.

Motivation. Recent needs for dynamic recommendations.

Complex content licensing agreements: Universal Music Group's (UMG) recent decision to pull its library from TikTok (Johnson 2024) silenced millions of user-created videos, necessitating TikTok's recommendations to adapt to the absence of UMG content. Similarly, the 2020 lawsuit (Deahl 2018) against Spotify by Wixen Music Publishing over songwriter compensation and the 2016 dispute (Levine 2018) between YouTube and Warner Music Group, which led to temporary content removal, highlight potential conflicts between existing models and evolving contractual obligations. These scenarios highlight the need for a dynamic RS that can adapt swiftly.

Legal compliance: The Algorithmic Accountability Act (Gursoy, Kennedy, and Kakadiaris 2022) emphasizes the responsibility of companies to evaluate their machine learning systems for bias and discrimination. Current RS often reinforce user preferences, perpetuating stereotypes and creating filter bubbles and echo chambers (Chen et al. 2023; Lin et al. 2021). This highlights the need for an RS that balances personalization with fairness and diversity.

Evolving user interests in different modalities: Imagine a user who usually posts about fitness and healthy living. Over time, she develops a new passion for travel photography, sharing photos and stories from her trips to exotic locations. Although her images and hashtags now reflect her love for travel, the RS still prioritizes fitness-related content. By analyzing both text and images, the system can adapt to her new interests, offering more relevant recommendations and enhancing her browsing experience.

Selective unlearning based on modalities: When Universal Music Group (UMG) removed its library from TikTok, the RS had to adapt to the absence of UMG audio content while still considering other modalities like videos. In MMRS, both the graph structure and feature embeddings are closely linked. Unlearning interactions in one modality, such as audio, can affect the entire RS, requiring careful adjustments to maintain accurate and effective recommendations.

Aiding selective transfer: Data privacy concerns and strict protection policies (Liu et al. 2023c) challenge cross-domain RS (Liu et al. 2024). While sharing user-item data can be beneficial, it risks negative transfer, such as using horror movie ratings to recommend comedies. Unlearning can remove irrelevant data from the source domain before transfer, enhancing recommendation accuracy.

Dynamic MMRS that handle both user and item data can effectively tackle these multifaceted challenges. By enabling systems to dynamically update their models and remove outdated or irrelevant content, unlearning methods can significantly enhance user privacy, ensure legal compliance, adapt to changing content licenses, and evolve with user preferences. These systems can also combat recommendation bias and data poisoning. Moreover, they can reduce the GPU-hours carbon footprint by minimizing the need for repeated, resource-intensive retraining from scratch.

Background and Related Work. *Given these benefits, one might ask: Can unlearning methods for uni-modal systems be adapted for the unique challenges of multi-modal systems?* Previous unlearning methods (Nguyen et al. 2022) face the following challenges: ❶ MMRS integrate diverse user-item data, including images, text, and behavior, into a unified convolutional graph. This is incompatible with the matrices, latent factors, feature representations, and temporal sequences used in matrix factorization (Liu et al. 2023a; Xu et al. 2022; Liu et al. 2022; Zhang et al. 2023a), collaborative filtering (Li et al. 2024; Schelter, Ariannezhad, and de Rijke 2023), and sequential RS (Ye and Lu 2023), respectively. ❷ For methods that do use graph-based representations, integrating diverse modalities is challenging. Because structure and feature embeddings are tightly integrated, unlearning in one modality has cascading effects on the other (Cheng and Amiri 2023). ❸ *Approximate unlearning methods* (You et al. 2024; Zhang et al. 2023b; Li et al. 2023b) involve expensive operations, such as inverting Hessian matrix or the Fisher Information Matrix. These computations are costly and impractical for MMRS having a large number of feature dimensions. ❹ *Exact unlearning methods* (Chen et al. 2022; Li et al. 2023a) adapt SISA (Bourtoule et al. 2021) to split the dataset into shards. This disrupts the graph structure and degrades performance within each shard due to limited data and poor data heterogeneity (Koch and Soll 2023) by $10-30\%$ (You et al. 2024). The aggregation step, to preserve data structure and aggregate performance across shards, introduces significant overhead costs, impacting both training and inference, increasing proportionally with the number of shards (Ramezani et al. 2021). In worst case, when data that needs to be forgotten come from multiple shards, efficiency falls to the level of retraining from scratch. ❺ While unlearning might appear efficient by retraining only specific shards, the process of setting up and aggregating these shards introduces additional, unaccounted overhead. If the data is initially partitioned by user dimensions, it cannot be re-split for item unlearning. Consequently, simultaneous unlearning of both user and item data becomes impossible.

Level	Qty	Catering to	Example
Interaction	Single	Privacy	Remove a watched movie
User Pref.	Many	Evolving preferences	Disinterest in Apple products
Biased Item	Many	Bias elimination	Amazon tackling fake reviews
Account	All	Privacy laws	User deletes account
License	All	Licensing agreements	UMG removing library from TikTok

Table 1: Unlearning request types in MMRS, addressing privacy, preferences, bias elimination, and legal compliance.

These constraints underscore the need for methods specifically designed for unlearning in MMRS and for unlearning item data. The challenges and previous works, including (Cheng and Amiri 2023; Yuan et al. 2023; Li et al. 2023c; Chen et al. 2024; Ganhör et al. 2022; Xin et al. 2024; Wang et al. 2024; Sinha, Mandal, and Kankanhalli 2023; Tarun et al. 2023b; Chundawat et al. 2023b; Tarun et al. 2023a; Chundawat et al. 2023a), provide valuable insights into this domain.

Contributions. They are summarized as follows:

1. MMRecUn is the first approach known to us for unlearning in MMRS and unlearning item data.
2. Our work addresses various unlearning requests, including single interaction removal, user preference adjustments, bias elimination, account deletion, and item removal, as shown in Table 1.
3. We define three properties for measuring unlearning in MMRS and introduce *item-centric metrics* alongside traditional user-centric metrics. We also propose BPR divergence as a robust alternative to KL divergence for comparing recommendation scores.
4. The experiments demonstrate that MMRecUn outperforms the baselines in various unlearning scenarios: user, item and user-item (simultaneous) unlearning. It achieves recall performance improvements of up to $\mathbf{49.85\%}$ compared to the baseline methods. It is up to $\mathbf{1.3}\times$ faster than the Gold model, which is trained on retain data from scratch. Moreover, MMRecUn offers enhanced efficiency, superior performance in removing target elements, preservation of performance for retained elements, and minimal overhead costs.

2 Preliminaries

MMRS. Let $\mathcal{U}=\{u\}$ denote the set of users and $\mathcal{I}=\{i\}$ denote the set of items. The embedding matrix for users is denoted as $\mathbf{E}_u \in \mathbb{R}^{d\times|\mathcal{U}|}$, where d is the embedding dimension. Similarly, the embedding matrices for each item modality are represented as $\mathbf{E}_{i,m} \in \mathbb{R}^{d_m\times|\mathcal{I}|}$, where d_m is the dimension of the features, $m \in \mathcal{M}$ denotes the modality, and $\mathcal{M} = \{v, t\}$, visual and textual, is the set of modalities considered. The historical behavior data of users is represented by matrix $\mathcal{Y} \in \{0,1\}^{|\mathcal{U}|\times|\mathcal{I}|}$, where each entry $y_{u,i}$ indicates whether user u interacted with item i. This data can be interpreted as a sparse behavior graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{\mathcal{U} \cup \mathcal{I}\}$ denotes the set of nodes and $\mathcal{E} = \{(u,i) \mid u \in \mathcal{U}, i \in \mathcal{I}, y_{u,i} = 1\}$ denotes the set of edges. Let the model be denoted by $M(\cdot, \varphi)$ with parameters φ which aims to encode collaborative signals latent in the interaction matrix $\mathcal{Y}$. The objective of MMRS is to accurately predict users' preferences by ranking items for each user based on predicted preference scores $\hat{y}_{u,i}$.

Unlearning. Let $\mathcal{D} = \{(u,i), y_{u,i}\}^{|\mathcal{E}|}, (u,i) \in \mathcal{E}$ represent a dataset of user-item interactions, split for training, validation and testing as $\mathcal{D}_T$, $\mathcal{D}_v$, and $\mathcal{D}_t$, respectively. The aim is to forget a set of data points, represented by $\mathcal{D}_f = \mathcal{E}_f \subseteq \mathcal{E}_T$, while retaining another set of data points, represented by $\mathcal{D}_r = \mathcal{E}_r$. It holds that $\mathcal{D}_r \bigcup \mathcal{D}_f = \mathcal{D}_T$ and $\mathcal{D}_r \bigcap \mathcal{D}_f = \emptyset$. If a node, i.e., a user or an item is marked for forgetting, then all interactions involving that user or item are also marked for forgetting. $(u,i) \in \mathcal{E}_f \implies \{(u', i') \mid u' = u \vee i' = i\} \subseteq \mathcal{E}_f$, where (u,i) represents the interaction between user u and item i, and (u', i') represents any interaction involving the same user or item. Given an input x, the model's output is $M(x, \varphi)$. For a machine learning algorithm A, it generates model parameters as $\varphi = A(\mathcal{D}_T)$. A *gold model* is trained from scratch only on the retain set $\mathcal{D}_r$, denoted by $\varphi_r = A(\mathcal{D}_r)$. An unlearning algorithm U utilizes all or a subset of $\mathcal{D}_r$ and $\mathcal{D}_f$, as well as the original model φ to generate an unlearned model φ_u. Hence, $\varphi_u = U(\varphi, \mathcal{D}_r, \mathcal{D}_f)$.

Problem Formulation. Given a sparse behavior graph $\mathcal{G}$ and a model $M(\cdot, \varphi)$, devise an unlearning algorithm U, that unlearns the forget set $\mathcal{D}_f$ to obtain an unlearned model $M(\cdot, \varphi_u)$ with updated parameters such that φ_u closely approximates the performance of the gold model:

$$\mathcal{P}(M(x, \varphi_u) = y) \approx \mathcal{P}(M(x, \varphi_r) = y), \quad \forall x \in \mathcal{D} \tag{1}$$

where $\mathcal{P}(X)$ is the distribution of random variable X.

Properties. Let $\forall x \in \mathcal{D}$, $f(\mathcal{D}, \epsilon)$ be defined as, $f(\mathcal{D}, \epsilon) : \mathcal{P}(M(x, \varphi_u) = y) - \mathcal{P}(M(x, \varphi_r) = y) \leq \epsilon$. For close performance approximation, the unlearned model must possess:

Unlearning Specificity $f(D_f, \epsilon_f)$. The unlearning algorithm U should effectively remove the influence of entities in $\mathcal{D}_f$, with φ_u aligning closely with the gold model's probability distribution. A high ϵ_f indicates unsuccessful unlearning, while a low ϵ_f may signal a *Streisand* effect, making the forget set more noticeable. Ideally, ϵ_f should approach zero.

Retention Fidelity $f(D_t, \epsilon_t)$. Equally important is preserving performance of the model. So the probability distribution on test set should align. A high value of ϵ_t indicates that the model's utility is compromised, as it becomes excessively tailored to the specific traits of the retain set. On the other hand, if ϵ_t is low, it indicates that the unlearning process has inadvertently led to the loss of characteristics of the retain set. So, ϵ_t should tend to zero.

Unlearning Generalizability $f(D_v, \epsilon_v)$. The unlearning algorithm U must ensure that the process does not introduce biases or distortions that impair the model's generalization. This is evaluated by comparing scores on unseen data, confirming that unlearning preserves the model's performance on data excluded from training and unlearning. A high ϵ_v indicates that unlearning may have overly tailored the model to the retain set, reducing generalization. Conversely, a low ϵ_v suggests that unlearning effectively retained the retain set's characteristics while removing the forget set's influence.

Unlearning Efficiency. The unlearning process should be faster than retraining the model from scratch. It should be efficient in terms of time and computational resources.

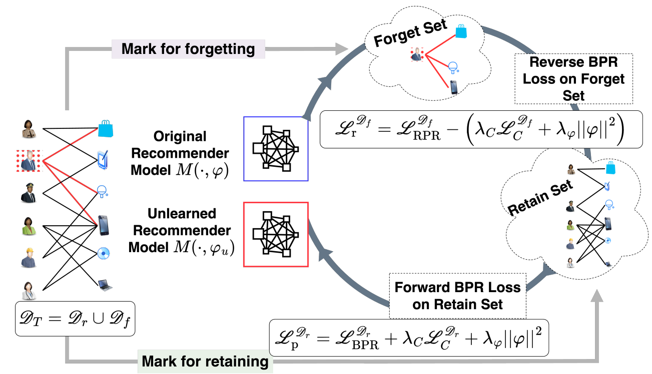


Figure 1: The proposed MMRecUn method illustrated. Adapts MGCN's architecture to unlearn multi-modal data while balancing retention fidelity and unlearning specificity.

3 Proposed MMRecUn Method

Traditionally, there are two stages for $M(\cdot, \varphi)$. In training, M encodes collaborative signals inherent in the user-item interaction matrix $\mathcal{Y}$. The optimization process minimizes the BPR loss (Rendle et al. 2012):

$$\min \mathcal{L}_{\text{BPR}}^{\mathcal{D}} = \sum_{(u,i) \in \mathcal{D}} \sum_{\substack{i \in \mathcal{Y}_u^+ \\ j \in \mathcal{Y}_u^-}} -\ln \sigma(\hat{y}_{u,i} - \hat{y}_{u,j}) \tag{2}$$

When the RS receives an unlearning request, it must first nullify the interaction data in the interaction matrix by setting $y_{u,i} = 0$ for all $y_{u,i} \in \mathcal{Y}_u^+$. Then, it would typically retrain the model $M(\cdot, \varphi)$ using the updated interaction matrix $\mathcal{Y}$. However, retraining is computationally expensive and impractical for frequent unlearning requests. Therefore, we propose unlearning the trained model $M(\cdot, \varphi)$ using MMRecUn. The overall process is illustrated in Fig. 1.

Reverse BPR Objective. To achieve the objectives of unlearning in MMRS, MMRecUn employs a reverse objective inspired by the concept of amnesiac unlearning (Graves, Nagisetty, and Ganesh 2021). This approach suggests selectively undoing the learning steps associated with a forget set $\mathcal{D}_f$, essentially removing the parameter updates related to forgotten data points and thus minimizing the predicted probability to a sufficiently small value. This works well for traditional machine learning tasks like image classification. Extending it to our context of MMRS, the reverse objective of user u for her marked interactions is: minimize the

Model	Valid				Test				Forget			
	Recall	**Prec**	**NDCG**	**MAP**	**Recall**	**Prec**	**NDCG**	**MAP**	**Recall**	**Prec**	**NDCG**	**MAP**
Dataset						Baby						
MGCN	0.0928	0.0049	0.0419	0.0274	0.0941	0.0052	0.0411	0.0255	0.5923	0.1430	0.4705	0.3003
GOLD	0.0929	0.0049	0.0413	0.0266	0.0944	0.0052	0.0410	0.0253	0.0105	0.0032	0.0072	0.0024
AMUN	0.0614	0.0033	0.0269	0.0170	0.0618	0.0034	0.0275	0.0174	0.0101	0.0023	0.0065	0.0025
MMRECUN	0.0889	0.0047	0.0406	0.0266	0.0870	0.0048	0.0384	0.0241	0.0105	0.0023	0.0087	0.0043
Dataset						Sports						
MGCN	0.1054	0.0056	0.0473	0.0306	0.1074	0.0060	0.0474	0.0295	0.4624	0.1091	0.3382	0.1924
GOLD	0.1059	0.0056	0.0476	0.0309	0.1076	0.0060	0.0481	0.0305	0.0048	0.0015	0.0034	0.0012
AMUN	0.0507	0.0027	0.0220	0.0138	0.0537	0.0030	0.0235	0.0145	0.0028	0.0009	0.0020	0.0007
MMRECUN	0.1035	0.0055	0.0469	0.0307	0.1042	0.0058	0.0467	0.0296	0.0049	0.0011	0.0035	0.0014
Dataset						Clothing						
MGCN	0.0899	0.0046	0.0400	0.0260	0.0898	0.0047	0.0406	0.0266	0.8057	0.1785	0.6443	0.4721
GOLD	0.0895	0.0045	0.0394	0.0254	0.0891	0.0046	0.0409	0.0271	0.0048	0.0011	0.0031	0.0012
AMUN	0.0405	0.0021	0.0176	0.0112	0.0415	0.0022	0.0180	0.0114	0.0052	0.0011	0.0039	0.0018
MMRECUN	0.0716	0.0036	0.0318	0.0206	0.0737	0.0038	0.0330	0.0214	0.0053	0.0011	0.0046	0.0024

Table 2: Unlearning 5% of **users**. MMRECUN matches GOLD across validation, test, and forget sets on varied datasets outperforming AMUN by 49.85%% in unlearning generalizability, by 46.93% in retention fidelity and high unlearning specificity.

Model / Metric	Set / K	Valid Recall	Valid Prec	Test Recall	Test Prec	Forget Recall	Forget Prec
Dataset				Baby			
MGCN		0.9481 ± 0.0005	0.0452 ± 0.0004	0.9073 ± 0.0003	0.0465 ± 0.0005	0.8070 ± 0.0005	0.0082 ± 0.0003
GOLD	500	0.9481 ± 0.0006	0.0446 ± 0.0003	0.9073 ± 0.0004	0.0460 ± 0.0006	0.5103 ± 0.0003	0.0014 ± 0.0001
AMUN		0.9271 ± 0.0004	0.0054 ± 0.0006	0.8943 ± 0.0007	0.0055 ± 0.0006	0.5639 ± 0.0005	0.0019 ± 0.0000
MMRECUN		0.9481 ± 0.0004	0.0104 ± 0.0005	0.9073 ± 0.0003	0.0113 ± 0.0005	0.5086 ± 0.0005	0.0010 ± 0.0001
Dataset				Sports			
MGCN		0.9180 ± 0.0009	0.0108 ± 0.0010	0.8621 ± 0.0005	0.0099 ± 0.0006	0.7573 ± 0.0008	0.0016 ± 0.0001
GOLD	1000	0.9180 ± 0.0006	0.0099 ± 0.0008	0.8621 ± 0.0007	0.0096 ± 0.0008	0.6469 ± 0.0006	0.0013 ± 0.0001
AMUN		0.9433 ± 0.0005	0.0045 ± 0.0007	0.9013 ± 0.0008	0.0046 ± 0.0006	0.6471 ± 0.0007	0.0017 ± 0.0008
MMRECUN		0.9373 ± 0.0007	0.0046 ± 0.0008	0.9001 ± 0.0005	0.0047 ± 0.0001	0.6693 ± 0.0008	0.0024 ± 0.0006
Dataset				Clothing			
MGCN		0.9857 ± 0.0010	0.0079 ± 0.0012	0.9690 ± 0.0009	0.0060 ± 0.0001	0.8840 ± 0.0009	0.0063 ± 0.0001
GOLD	1500	0.9857 ± 0.0011	0.0416 ± 0.0001	0.9690 ± 0.0011	0.0423 ± 0.0006	0.7134 ± 0.0009	0.0020 ± 0.0001
AMUN		0.9721 ± 0.0012	0.0034 ± 0.0001	0.9516 ± 0.0008	0.0030 ± 0.0001	0.7605 ± 0.0009	0.0018 ± 0.0001
MMRECUN		0.9761 ± 0.0008	0.0034 ± 0.0001	0.9526 ± 0.0008	0.0029 ± 0.0001	0.7002 ± 0.0008	0.0016 ± 0.0000

Table 3: Unlearning 5% of **items**. MMRECUN matches GOLD across validation, test, and forget sets on varied datasets outperforming AMUN by 10.83% in unlearning specificity, and comparable retention fidelity, and unlearning generalizability.

predicted score of marked interactions relative to items with which she did not interact.

$$\min \mathcal{L}_{\text{RPR}}^{\mathcal{D}_f} = \sum_{(u,i)\in\mathcal{D}_f} \sum_{\substack{i\in\mathcal{Y}_u^+ \\ j\in\mathcal{Y}_u^-}} \ln \sigma(\hat{y}_{u,i} - \hat{y}_{u,j}) \tag{3}$$

However, updating the model with this reverse objective risks catastrophic forgetting of data in the retain set $\mathcal{D}_r$, possibly leading to inaccurate recommendations. Thus, it's essential to balance retention fidelity with unlearning specificity. MMRECUN addresses this by using the original BPR objective to preserve the retain set $\mathcal{D}_r$: $\mathcal{L}_{\text{BPR}}^{\mathcal{D}_r}$.

Unlearning Multi-Modal Data requires minimizing the predicted probability based on the discriminability of features. In addition to the collaborative signals from the user-item interaction matrix, $\mathcal{Y}$, $M(\cdot, \varphi)$ encodes item-item semantic correlations, into embeddings $\mathbf{E}_u$ for users and $\mathbf{E}_{i,m}$ for items in each modality $m \in \mathcal{M}$. A self-supervised auxiliary task maximizes mutual information between behavior features and fused multi-modal features, promoting the exploration of both, alongside L_2 regularization.

$$\mathcal{L}_C^{\mathcal{D}} = \sum_{u\in\mathcal{D}} -\log\left(\frac{\exp\left(\mathbf{e}_{u,\text{mul}} \cdot \bar{\mathbf{e}}_{u,\text{id}}/\tau\right)}{\sum_{v\in\mathcal{U}} \exp\left(\mathbf{e}_{v,\text{mul}} \cdot \bar{\mathbf{e}}_{v,\text{id}}/\tau\right)}\right) + \sum_{i\in\mathcal{D}} -\log\left(\frac{\exp\left(\mathbf{e}_{i,\text{mul}} \cdot \bar{\mathbf{e}}_{i,\text{id}}/\tau\right)}{\sum_{j\in\mathcal{I}} \exp\left(\mathbf{e}_{j,\text{mul}} \cdot \bar{\mathbf{e}}_{j,\text{id}}/\tau\right)}\right) \tag{4}$$

, where τ is temperature. For unlearning, we introduce a negated contrastive auxiliary loss and a regularization term to *reduce* the impact of learned item-item semantic correlations:

$$\mathcal{L}_{\text{r}}^{\mathcal{D}_f} = \mathcal{L}_{\text{RPR}}^{\mathcal{D}_f} - (\lambda_C \mathcal{L}_C^{\mathcal{D}_f} + \lambda_\varphi \|\varphi\|^2) \tag{5}$$

Model	Set	Valid				Test				Forget			
	@K	Recall	Precision	NDCG	MAP	Recall	Precision	NDCG	MAP	Recall	Precision	NDCG	MAP
Dataset						Baby							
MGCN	20	0.1529	0.0032	0.0533	0.0286	0.1574	0.0035	0.0538	0.0274	0.0048	0.0005	0.0030	0.0011
	3200	0.8447	0.0030	0.2263	0.0497	0.8180	0.0030	0.1452	0.0096	0.1792	0.0050	0.3857	0.1195
GOLD	20	0.1560	0.0033	0.0530	0.0277	0.1585	0.0035	0.0543	0.0279	0.0014	0.0002	0.0010	0.0004
	3200	0.8398	0.0030	0.2342	0.0553	0.8211	0.0029	0.1534	0.0120	0.2071	0.0049	0.3385	0.0843
AMUN	20	0.1606	0.0034	0.0546	0.0284	0.1628	0.0036	0.0557	0.0285	0.0014	0.0002	0.0007	0.0001
	3200	0.8417	0.0029	0.2020	0.0342	0.8119	0.0030	0.1490	0.0107	0.3416	0.0051	0.3676	0.1092
MMRECUN	20	0.1479	0.0031	0.0526	0.0288	0.1521	0.0034	0.0525	0.0271	0.0014	0.0002	0.0008	0.0002
	3200	0.8520	0.0029	0.3327	0.1426	0.8270	0.0028	0.1574	0.0134	0.2048	0.0051	0.7874	0.5434
Dataset						Sports							
MGCN	20	0.1681	0.0035	0.0604	0.0332	0.1739	0.0039	0.0612	0.0321	0.0030	0.0003	0.0018	0.0006
	7200	0.7392	0.0017	0.1882	0.0262	0.7154	0.0017	0.1941	0.0293	0.1034	0.0026	0.2308	0.0525
GOLD	20	0.1694	0.0036	0.0607	0.0334	0.1745	0.0039	0.0617	0.0326	0.0012	0.0001	0.0007	0.0002
	7200	0.7362	0.0015	0.1844	0.0243	0.7133	0.0017	0.2236	0.0475	0.1130	0.0026	0.2271	0.0499
AMUN	20	0.1716	0.0036	0.0608	0.0330	0.1752	0.0039	0.0612	0.0318	0.0012	0.0001	0.0006	0.0002
	7200	0.7291	0.0017	0.1821	0.0231	0.7191	0.0017	0.1957	0.0302	0.1843	0.0026	0.2236	0.0475
MMRECUN	20	0.1667	0.0035	0.0612	0.0346	0.1720	0.0038	0.0619	0.0334	0.0012	0.0001	0.0006	0.0002
	7200	0.7449	0.0015	0.2051	0.0356	0.7268	0.0016	0.2120	0.0399	0.1376	0.0024	0.2030	0.0343
Dataset						Clothing							
MGCN	20	0.1371	0.0028	0.0491	0.0272	0.1360	0.0028	0.0496	0.0278	0.0059	0.0005	0.0032	0.0010
	7800	0.8290	0.0012	0.0604	0.0003	0.8213	0.0011	0.1029	0.0018	0.0332	0.0017	0.4999	0.3332
GOLD	20	0.1390	0.0028	0.0497	0.0275	0.1355	0.0028	0.0492	0.0274	0.0014	0.0001	0.0007	0.0002
	7800	0.8197	0.0010	0.0388	0.0001	0.8144	0.0012	0.0696	0.0004	0.0502	0.0016	0.4999	0.3332
AMUN	20	0.1372	0.0028	0.0491	0.0272	0.1365	0.0028	0.0497	0.0278	0.0013	0.0001	0.0005	0.0001
	7800	0.8229	0.0012	0.0629	0.0003	0.8189	0.0012	0.0995	0.0015	0.1541	0.0019	0.4305	0.2499
MMRECUN	20	0.1374	0.0028	0.0496	0.0278	0.1366	0.0029	0.0501	0.0282	0.0014	0.0001	0.0008	0.0002
	7800	0.8293	0.0012	0.0502	0.0002	0.8126	0.0012	0.1054	0.0020	0.0757	0.0018	0.4305	0.2499

Table 4: Unlearning 5% of **users and items**. MMRECUN matches GOLD across validation, test, and forget sets outperforming AMUN by 41.32% in unlearning specificity and comparable unlearning generalizability and retention fidelity.

To mitigate the risk of catastrophic forgetting, MMRECUN employs the original training objective to *preserve* the knowledge within the retain set $\mathcal{D}_r$:

$$\mathcal{L}_{\text{p}}^{\mathcal{D}_r} = \mathcal{L}_{\text{BPR}}^{\mathcal{D}_r} + \lambda_C \mathcal{L}_C^{\mathcal{D}_r} + \lambda_\varphi \|\varphi\|^2 \tag{6}$$

However, this preservation introduces additional epochs of contrastive auxiliary loss and L_2 regularization, which may cause overfitting on the retain set $\mathcal{D}_r$. Thus, the loss in eq.5 also serves to counterbalance these effects by discouraging the model from maintaining invariance in these features. Finally, the loss function becomes

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{\text{p}}^{\mathcal{D}_r} + (1-\alpha) \cdot \mathcal{L}_{\text{r}}^{\mathcal{D}_f} \tag{7}$$

where α is a hyper-parameter that determines the relative importance of preservation and reduction.

Proposition 1 ***Bayesian Interpretation of* MMRECUN.** *The* MMRECUN *objective function in Eq. 3 can be interpreted through the Bayes theorem. Let the learning process be regarded as maximizing the posterior distribution estimated by φ, i.e., $\max P(\varphi|\mathcal{D})$, with a certain prior distribution of $g(\varphi)$. Maximizing the posterior distribution $\log P(\varphi|\mathcal{D}_r)$ is equivalent to the RPR objective in* MMRECUN*, which is to minimize the RPR objective with a regularizer. At the optimal point, $\mathcal{L}_{BPR} \approx 0$ and $\left\|\frac{\partial \mathcal{L}_{BPR}}{\partial \varphi_0}\right\|^2 \approx 0$. Then the approximation of optimal posterior distribution is*

$$\mathcal{L} \approx \frac{1}{2}(\varphi - \varphi_0)^T \frac{\partial^2 \mathcal{L}_{BPR}}{\partial \varphi_0^2}(\varphi - \varphi_0) \tag{8}$$

The Kullback-Leibler (KL) divergence (Kullback and Leibler 1951; Golatkar, Achille, and Soatto 2020; Tarun et al. 2023a) is difficult to implement in RS because the typical user-item interaction data are not inherently probabilistic but based on scores or ratings. Further, KL Divergence can be undefined when the target distribution has zero probability for an event that the input distribution considers likely, which can be problematic in sparse and high-dimensional recommendation datasets. Therefore, we *propose* BPR divergence, β as an alternative which directly measures the difference between the predicted scores of user-item interactions from different model states using mean squared error, which aligns well with the numerical nature of recommendation scores.

$$\beta(\varphi, \varphi') = \frac{1}{|\mathcal{Y}_u^+|} \sum_{i \in \mathcal{Y}_u^+} \Delta y_{u,i}^2 + \frac{1}{|\mathcal{Y}_u^-|} \sum_{j \in \mathcal{Y}_u^-} \Delta y_{u,j}^2 \tag{9}$$

where, $\Delta y_{u,i} = \hat{y}_{u,i} - \hat{y}_{u,i}^{'}$ and $\Delta y_{u,j} = \hat{y}_{u,j} - \hat{y}_{u,j}^{'}$.

Proposition 2 ***Information Bound of* MMRECUN.** *Let the posterior distribution before and after unlearning the forget set $\mathcal{D}_f$, be $P(\varphi|\mathcal{D}_T)$ and $P(\varphi|\mathcal{D}_r)$, [since $\mathcal{D}_T - \mathcal{D}_f = \mathcal{D}_r$], respectively. The convergence between can be expressed as $\beta(\varphi, \varphi') \leq \epsilon$ where n denote the number of epochs during unlearning. Since $\epsilon \propto 1/n$, we have $\epsilon = k \times \frac{1}{n}$, where k is a constant.*

As shown in Figure 2a, the BPR divergence between the gold and unlearned models on the forget set $\mathcal{D}_f$ decreases with more epochs, indicating that less information about $\mathcal{D}_f$ remains in the model over time.

Time Complexity for unlearning LightGCN with BPR and auxiliary contrastive loss using MMRecUn is $T = \mathcal{O}\left(\mathcal{M} \cdot d \cdot |\mathcal{E}| \cdot d_m + |\mathcal{U}| \cdot |\mathcal{I}| \cdot K + d^2 + |\mathcal{M}| \cdot d\right)$, where $\mathcal{M}$ is the number of modalities, d is the embedding dimension, $|\mathcal{E}|$ is the number of edges, d_m is the feature dimension for modality m, $|\mathcal{U}|$ is the number of users, $|\mathcal{I}|$ is the number of items, and K is the number of negative samples per user-item pair.

4 Experiments and Results

4.1 Experimental Setup

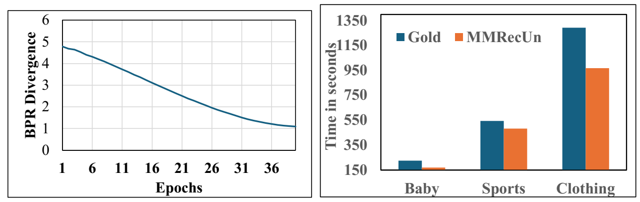
Datasets and baselines. We select MGCN (Yu et al. 2023) as our base model for several reasons: ❶ It is the state-of-the-art MMRS with BPR. ❷ MGCN represents BPR-based models well, making improvements likely to generalize. ❸ Unlike traditional methods like Collaborative Filtering and Matrix Factorization, it handles multi-modal features through graph convolutional representations. ❹ Other loss functions for uni-modal systems don't address multi-modal complexities, and those that do, perform weaker than MGCN. So, focusing on MGCN ensures our approach is relevant.

We use three distinct categories within the Amazon dataset (Hou et al. 2024) that are used to benchmark MMRS. The model trained on the retain set (named as the GOLD model) from scratch is used as baseline. We prepare another baseline by re-purposing the image classification unlearning AmUn (Graves, Nagisetty, and Ganesh 2021) for MMRS.

Previous RS unlearning methods using matrix factorization, collaborative filtering, or sequential systems operate on behavior matrices, making them incompatible with graph-based architectures. The code for approximate methods RRL, IFRU, and SCIF is unavailable. Exact methods, RECERASER and ULTRARE, require partitioning, which is challenging due to the need for specialized handling of each modality and ensuring coherence during aggregation, adding complexity outside the scope of this work.

Evaluation metrics To evaluate a given model $M(\cdot, \varphi)$, we need ❶ model predictions: ranked list of user-item pairs; ❷ the ground truth: user-item interactions $\mathcal{Y}$; and, ❸ K: number of the top recommendations to consider. To capture a variety of aspects of performance, we use ❶ predictive metrics, that reflect the "correctness", how well $M(\cdot, \varphi)$ finds relevant items such as Recall at K and Precision at K; and ❷ ranking metrics, that reflect ranking quality, how well $M(\cdot, \varphi)$ can sort items from more relevant to less relevant. NDCG considers both relevance and position of items in ranked list. MAP measures the average Precision across different Recall levels for a ranked list.

Settings and Hyper-parameters All experiments are performed on 4x NVIDIA RTX2080 (32GB). We use the identical settings as in the case of MGCN for training the original model $M(\cdot, \varphi)$. The data interaction history of each user is split $8:1:1$ for training, validation and testing. The training set is used to create retain set and forget set. We initialize the embedding with Xavier initialization of dimension 64, set the regularization coefficient $\lambda_\varphi = 10^{-4}$, and batch size $B = 2048$. For the self-supervised task, we set the



(a) BPR Divergence Over Epochs. Decreases with more epochs, indicating reduced retention of $\mathcal{D}_f$ information.

(b) MMRecUn reduces time by up to 1.3× compared to retraining from scratch, saving 25% of the time.

Figure 2: Information bound and efficiency analysis.

temperature to 0.2. For convergence, the early stopping and total epochs are fixed at 20 and 1000, respectively. We use Recall@20 on the validation data as the training-stopping indicator following (Zhang et al. 2022). We conducted experiments five times using different seed values and report the standard deviation values for item unlearning in Table 3. For the other types of unlearning, the standard deviation values were found to be negligible and are therefore omitted.

4.2 Results

We conduct experiments to assess the unlearning performance of MMRecUn in the four aspects. First, we compare recall, precision, NDCG and MAP metrics on *forget set*. Values closer to the GOLD model signify better *Unlearning Specificity*. Second, we compare the metrics on *test set*. Values closer to the GOLD model signify better *Retention Fidelity*. Third, we compare the metrics on *validation set*. Values closer to the GOLD model signify better *Unlearning Generalizability*. Finally, we compare the time duration required for unlearning to compare *efficiency*.

User-centric metrics measure the relevance from user's perspective i.e., whether the relevant items (the user likes or interacts with) are present in the top K recommendations. Users typically expect to see a small number of recommendations (e.g., 5 or 10), hence we take $K = 5, 10, 20, 50$ to assess how well the system can provide personalized and focused recommendations. Larger values of K, such as 20 or 50, are useful for evaluating the system's ability to cover a wider range of user preferences or measure performance in suggesting diverse options and catering to different user tastes. These metrics primarily focus on user satisfaction and do not capture the effectiveness of forgetting specific items. Hence, we *introduce* several *item-centric metrics*.

Item-centric metrics (Table 6) measure the relevance of the recommended items themselves i.e., how well the recommended items meet unlearning criteria, regardless of which users receive them. We take $K = 500, 1000, 1500$ to ensure that all items in the forget set $\mathcal{D}_f$ are considered for evaluating performance.

User Unlearning. Table 2 shows the results of unlearning 5% users. The GOLD model, which has never seen the forget set, falls in performance on this set as expected. However, it retains good performance on the test and validation sets. On the validation set, MMRecUn consistently achieves better

α	**0.001**	**0.003**	**0.01**	**0.03**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**
Valid	0.0779	0.0851	0.0882	0.0887	0.0893	0.0893	0.0913	0.0909	0.0897	0.0880	0.0878	0.0804	0.0775
Test	0.0748	0.0827	0.0848	0.0856	0.0871	0.0892	0.0920	0.0915	0.0910	0.0885	0.0885	0.0839	0.0809
Forget	0.0104	0.0103	0.0107	0.0102	0.0105	0.0103	0.0105	0.0106	0.0105	0.0107	0.0104	0.0107	0.0102

Table 5: Tuning α for balance. Recall@20 on validation, test, and forget sets while unlearning 5% of users in the Baby dataset. Lower α values optimize recall and convergence, while higher values improve forgetting at the expense of retain set recall.

Metric	**User-Centric**	**Item-Centric**
Recall@K	$\frac{\lvert\text{Relevant}_u \cap \text{Retrieved}_u\rvert}{\lvert\text{Relevant}_u\rvert}$	$\frac{\sum_{j=1}^{K} \text{rel}_{j,i}}{\lvert\text{Users}_i\rvert}$
Recall@K	$\frac{1}{\lvert\mathcal{U}\rvert} \sum_{u \in \mathcal{U}} \text{Recall}_u@K$	$\frac{1}{\lvert\mathcal{I}\rvert} \sum_{i \in \mathcal{I}} \text{Recall}_i@K$
Prec@K	$\frac{\lvert Rel_u \cap Rec_u\rvert}{\lvert Rec_u\rvert}$	$\text{Prec}_i@K = \frac{\sum_{j=1}^{K} \text{rel}_{j,i}}{K}$
Precision@K	$\frac{1}{\lvert\mathcal{U}\rvert} \sum_{u \in \mathcal{U}} \text{Prec}_u@K$	$\frac{1}{\lvert\mathcal{I}\rvert} \sum_{i \in \mathcal{I}} \text{Prec}_i@K$
DCG@K	$\sum_{i=1}^{K} \frac{2^{rel_{i,u}}-1}{\log_2(i+1)}$	$\sum_{j=1}^{K} \frac{2^{rel_{j,i}}-1}{\log_2(j+1)}$
IDCG@K	$\sum_{i=1}^{K} \frac{1}{\log_2(i+1)}$	$\sum_{j=1}^{K} \frac{1}{\log_2(j+1)}$
NDCG@K	$\frac{\text{DCG}_u@K}{\text{IDCG}_u@K}$	$\frac{\text{DCG}_i@K}{\text{IDCG}_i@K}$
NDCG@K	$\frac{1}{\lvert\mathcal{U}\rvert} \sum_{u \in \mathcal{U}} \text{NDCG}_u@K$	$\frac{1}{\lvert\mathcal{I}\rvert} \sum_{i \in \mathcal{I}} \text{NDCG}_i@K$
AP@N	$\sum_{k=1}^{N} \frac{P(k) \cdot \text{rel}_u(k)}{\min(m,N)}$	$\sum_{k=1}^{N} \frac{P(k) \cdot \text{rel}_i(k)}{\min(m,N)}$
MAP@N	$\frac{1}{\lvert\mathcal{U}\rvert} \sum_{u \in \mathcal{U}} \text{AP}_u@N$	$\frac{1}{\lvert\mathcal{I}\rvert} \sum_{i \in \mathcal{I}} \text{AP}_i@N$

Table 6: User-Centric vs. Item-Centric Metrics. Measures relevance from the user's perspective versus from the item's perspective, independent of users recommended to.

recall scores, outperforming AMUN by 29.6%, 49.85%, and 34.75% on the three datasets (baby, sports, and clothing, respectively), showcasing *best* unlearning generalizability. On the test set, MMRECUN maintains closest recall scores, surpassing AMUN by 26.69%, 46.93%, and 36.13%, demonstrating *best* retention fidelity. On the forget set, MMRECUN achieves better recall scores, outperforming AMUN by 3.8% and 43.75% on two datasets and slightly falling back by 2.08% on the clothing dataset, indicating *high* unlearning specificity. Similar trends follow for other metrics.

Item Unlearning Table 3 shows the results of unlearning 5% items. On the validation set, the recall scores differ by 2.21%, 0.65%, and 0.41% on the three datasets, showcasing *comparable* unlearning generalizability. Similarly, on the test set, the recall scores differ by 1.43%, 0.13%, and 0.10%, demonstrating *comparable* retention fidelity. However, MMRECUN achieves good recall scores, outperforming AMUN by 10.83% and 8.45% on two datasets and slightly falling back by 3.43% on the sports dataset, indicating *high* unlearning specificity.

User and Item Unlearning. Table 4 presents the results of unlearning both 5% users and 5% items. On *user-centric metrics*, the performance of AMUN and MMRECUN is comparable. The recall scores differ by 2.89%, 1.83% and 0% approximately across three datasets on the validation set, test set and forget set, respectively, showcasing *comparable* unlearning generalizability, retention fidelity and unlearning specificity. On *item-centric metrics*, MMRECUN achieves recall score up to 41.32% higher than AMUN on forget set, indicating *high* unlearning specificity. The recall scores differ by 2.14% and 1.83% approximately across three datasets on the validation and test sets, respectively, showcasing *comparable* unlearning generalizability and retention fidelity. We repeat the experiment 5 times using different seed values and report the standard deviation in Table 3. For the other types of unlearning, the standard deviation were found to be negligible and are therefore omitted.

Unlearning efficiency. On the baby dataset, MMRECUN takes 169.41 seconds in contrast to 225.73 seconds of GOLD model. Similarly, on the clothing dataset, MMRECUN takes 968.96 seconds in contrast to 1294.44 seconds of GOLD model. Thus, with MMRECUN, we experience up to 1.3× accelerated unlearning than retraining from scratch (Figure 2b). In other words, 25% time is saved.

Tuning hyper-parameter α. We assess how α balances adaptation and reform in unlearning, testing values from 0.001 to 0.9 (Table 5). Lower α values (0.1-0.2) effectively unlearn the forget set with minimal impact on retain set recall, and require fewer epochs for convergence, indicating an optimal balance. As α increases (0.3-0.9), the model emphasizes more on the reform loss, needing more epochs to forget, adversely affecting retain set recall. Extremely low α values (< 0.01) hinder effective forgetting, also increasing epochs and negatively impacting retain set. The optimal α depends on the specific task's forgetting severity and time constraints: higher α may ensure stronger forgetting with minor recall loss, while lower α enables faster convergence.

5 Conclusion

MMRECUN is the first framework for MMRS unlearning that handles complex multi-modal data and item unlearning. It caters to diverse unlearning requests, like user preference adjustments, item removal etc.; and scenarios: user, item and user-item simultaneous unlearning. We define three key properties to measure MMRS unlearning; introduce item-centric metrics with traditional user-centric ones; and propose BPR divergence as a robust alternative to KL divergence to compare recommendation scores. Our experiments demonstrate MMRECUN's superiority in retention fidelity, unlearning specificity, generalizability, and efficiency.

Acknowledgments

This research/project is supported by the National Research Foundation, Singapore under its Strategic Capability Research Centres Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

References

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.

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

Chen, C.; Zhang, Y.; Li, Y.; Meng, D.; Wang, J.; Zheng, X.; and Yin, J. 2024. Post-Training Attribute Unlearning in Recommender Systems. *arXiv preprint arXiv:2403.06737*.

Chen, J.; Dong, H.; Wang, X.; Feng, F.; Wang, M.; and He, X. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems*, 41(3): 1–39.

Cheng, J.; and Amiri, H. 2023. Multimodal Machine Unlearning. *arXiv preprint arXiv:2311.12047*.

Chundawat, V. S.; Tarun, A. K.; Mandal, M.; and Kankanhalli, M. 2023a. Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 7210–7217.

Chundawat, V. S.; Tarun, A. K.; Mandal, M.; and Kankanhalli, M. 2023b. Zero-shot machine unlearning. *IEEE Transactions on Information Forensics and Security*.

Covington, P.; Adams, J.; and Sargin, E. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*, 191–198.

Deahl, D. 2018. Spotify and Wixen settle the music publishing company's $1.6 billion lawsuit. *The Verge*.

Ganhör, C.; Penz, D.; Rekabsaz, N.; Lesota, O.; and Schedl, M. 2022. Unlearning protected user attributes in recommendations with adversarial training. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2142–2147.

Golatkar, A.; Achille, A.; and Soatto, S. 2020. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9304–9312.

Graves, L.; Nagisetty, V.; and Ganesh, V. 2021. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 11516–11524.

Gursoy, F.; Kennedy, R.; and Kakadiaris, I. 2022. A critical assessment of the algorithmic accountability act of 2022. *SSRN Electronic Journal*.

He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; and Wang, M. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 639–648. ACM.

Hou, Y.; Li, J.; He, Z.; Yan, A.; Chen, X.; and McAuley, J. 2024. Bridging Language and Items for Retrieval and Recommendation. *arXiv preprint arXiv:2403.03952*.

Johnson, A. 2024. Will tiktok videos be muted? here's what to know after licensing deal with Universal Music ends.

Koch, K.; and Soll, M. 2023. No Matter How You Slice It: Machine Unlearning with SISA Comes at the Expense of Minority Classes. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 622–637.

Kullback, S.; and Leibler, R. A. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1): 79–86.

Levine, R. 2018. Warner Music Group pulls music from YouTube. *Billboard*.

Li, Y.; Chen, C.; Zhang, Y.; Liu, W.; Lyu, L.; Zheng, X.; Meng, D.; and Wang, J. 2023a. UltraRE: Enhancing RecEraser for Recommendation Unlearning via Error Decomposition. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Li, Y.; Chen, C.; Zheng, X.; Liu, J.; and Wang, J. 2024. Making recommender systems forget: Learning and unlearning for erasable recommendation. *Knowledge-Based Systems*, 283: 111124.

Li, Y.; Chen, C.; Zheng, X.; Zhang, Y.; Gong, B.; Wang, J.; and Chen, L. 2023b. Selective and collaborative influence function for efficient recommendation unlearning. *Expert Systems with Applications*, 234: 121025.

Li, Y.; Chen, C.; Zheng, X.; Zhang, Y.; Han, Z.; Meng, D.; and Wang, J. 2023c. Making users indistinguishable: Attribute-wise unlearning in recommender systems. In *Proceedings of the 31st ACM International Conference on Multimedia*, 984–994.

Lin, C.; Liu, X.; Xv, G.; and Li, H. 2021. Mitigating sentiment bias for recommender systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 31–40.

Liu, J.; Li, D.; Gu, H.; Lu, T.; Wu, J.; Zhang, P.; Shang, L.; and Gu, N. 2023a. Recommendation Unlearning via Matrix Correction. *arXiv preprint arXiv:2307.15960*.

Liu, K.; Xue, F.; Guo, D.; Sun, P.; Qian, S.; and Hong, R. 2023b. Multimodal graph contrastive learning for multimedia-based recommendation. *IEEE Transactions on Multimedia*.

Liu, W.; Wan, J.; Wang, X.; Zhang, W.; Zhang, D.; and Li, H. 2022. Forgetting fast in recommender systems. *arXiv preprint arXiv:2208.06875*.

Liu, W.; Zheng, X.; Chen, C.; Hu, M.; Liao, X.; Wang, F.; Tan, Y.; Meng, D.; and Wang, J. 2023c. Differentially private sparse mapping for privacy-preserving cross domain recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia*, 6243–6252.

Liu, W.; Zheng, X.; Chen, C.; Xu, J.; Liao, X.; Wang, F.; Tan, Y.; and Ong, Y.-S. 2024. Reducing Item Discrepancy via Differentially Private Robust Embedding Alignment for Privacy-Preserving Cross Domain Recommendation. In *Forty-first International Conference on Machine Learning*.

Nguyen, T. T.; Huynh, T. T.; Nguyen, P. L.; Liew, A. W.-C.; Yin, H.; and Nguyen, Q. V. H. 2022. A survey of machine unlearning. *arXiv preprint arXiv:2209.02299*.

Ramezani, M.; Cong, W.; Mahdavi, M.; Kandemir, M. T.; and Sivasubramaniam, A. 2021. Learn locally, correct globally: A distributed algorithm for training graph neural networks. *arXiv preprint arXiv:2111.08202*.

Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*.

Schelter, S.; Ariannezhad, M.; and de Rijke, M. 2023. Forget me now: Fast and exact unlearning in neighborhood-based recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2011–2015.

Sinha, Y.; Mandal, M.; and Kankanhalli, M. 2023. Distill to Delete: Unlearning in Graph Networks with Knowledge Distillation. *arXiv preprint arXiv:2309.16173*.

Smith, B.; and Linden, G. 2017. Two decades of recommender systems at Amazon. com. *Ieee internet computing*, 21(3): 12–18.

Tarun, A. K.; Chundawat, V. S.; Mandal, M.; and Kankanhalli, M. 2023a. Deep regression unlearning. In *International Conference on Machine Learning*, 33921–33939. PMLR.

Tarun, A. K.; Chundawat, V. S.; Mandal, M.; and Kankanhalli, M. 2023b. Fast Yet Effective Machine Unlearning. *IEEE Transactions on Neural Networks and Learning Systems*.

Voigt, P.; and Von dem Bussche, A. 2017. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*.

Wang, H.; Lin, J.; Chen, B.; Yang, Y.; Tang, R.; Zhang, W.; and Yu, Y. 2024. Towards Efficient and Effective Unlearning of Large Language Models for Recommendation. *arXiv preprint arXiv:2403.03536*.

Wei, Y.; Wang, X.; Nie, L.; He, X.; and Chua, T.-S. 2020. Graph-refined convolutional network for multimedia recommendation with implicit feedback. In *Proceedings of the 28th ACM international conference on multimedia*, 3541–3549.

Wei, Y.; Wang, X.; Nie, L.; He, X.; Hong, R.; and Chua, T.-S. 2019. MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th ACM international conference on multimedia*, 1437–1445.

Wu, L.; He, X.; Wang, X.; Zhang, K.; and Wang, M. 2022. A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(5): 4425–4445.

Xin, X.; Yang, L.; Zhao, Z.; Ren, P.; Chen, Z.; Ma, J.; and Ren, Z. 2024. On the Effectiveness of Unlearning in Session-Based Recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 855–863.

Xu, M.; Sun, J.; Yang, X.; Yao, Y.; and Wang, C. 2022. Netflix and Forget: Fast Severance From Memorizing Training Data in Recommendations. In *NeurIPS ML Safety Workshop*.

Xu, Q.; Shen, F.; Liu, L.; and Shen, H. T. 2018. Graphcar: Content-aware multimedia recommendation with graph autoencoder. In *The 41st International ACM SIGIR conference on research & development in information retrieval*, 981–984.

Ye, S.; and Lu, J. 2023. Sequence Unlearning for Sequential Recommender Systems. In *Australasian Joint Conference on Artificial Intelligence*, 403–415. Springer.

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.

Yu, P.; Tan, Z.; Lu, G.; and Bao, B.-K. 2023. Multi-view graph convolutional network for multimedia recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia*, 6576–6585.

Yuan, W.; Yin, H.; Wu, F.; Zhang, S.; He, T.; and Wang, H. 2023. Federated unlearning for on-device recommendation. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 393–401.

Zhang, J.; Zhu, Y.; Liu, Q.; Zhang, M.; Wu, S.; and Wang, L. 2022. Latent structure mining with contrastive modality fusion for multimedia recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Zhang, S.; Lou, J.; Xiong, L.; Zhang, X.; and Liu, J. 2023a. Closed-form machine unlearning for matrix factorization. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 3278–3287.

Zhang, Y.; Hu, Z.; Bai, Y.; Feng, F.; Wu, J.; Wang, Q.; and He, X. 2023b. Recommendation unlearning via influence function. *arXiv preprint arXiv:2307.02147*.

Zhou, H.; Zhou, X.; Zeng, Z.; Zhang, L.; and Shen, Z. 2023. A comprehensive survey on multimodal recommender systems: Taxonomy, evaluation, and future directions. *arXiv preprint arXiv:2302.04473*.