Privacy-Preserving Synthetic Data Generation for Recommendation Systems

Fan Liu
National University of Singapore
liufancs@gmail.com

Zhiyong Cheng
Qilu University of Technology
(Shandong Academy of Sciences)
jason.zy.cheng@gmail.com

Huilin Chen
Tianjin University of Technology
ClownClumsy@outlook.com

Yinwei Wei
National University of Singapore
weiyinwei@hotmail.com

Liqiang Nie
Shandong University
nieliqiang@gmail.com

Mohan Kankanhalli
National University of Singapore
mohan@comp.nus.edu.sg

ABSTRACT

Recommendation systems make predictions chiefly based on users' historical interaction data (*e.g.*, items previously *clicked* or *purchased*). There is a risk of privacy leakage when collecting the users' behavior data for building the recommendation model. However, existing privacy-preserving solutions are designed for tackling the privacy issue only during the model training [32] and results collection [40] phases. The problem of privacy leakage still exists when directly sharing the private user interaction data with organizations or releasing them to the public. To address this problem, in this paper, we present a User Privacy Controllable Synthetic Data Generation model (short for UPC-SDG), which generates synthetic interaction data for users based on their privacy preferences. The generation model aims to provide certain privacy guarantees while maximizing the utility of the generated synthetic data at both data level and item level. Specifically, at the data level, we design a selection module that selects those items that contribute less to a user's preferences from the user's interaction data. At the item level, a synthetic data generation module is proposed to generate a synthetic item corresponding to the selected item based on the user's preferences. Furthermore, we also present a privacy-utility trade-off strategy to balance the privacy and utility of the synthetic data. Extensive experiments and ablation studies have been conducted on three publicly accessible datasets to justify our method, demonstrating its effectiveness in generating synthetic data under users' privacy preferences.

CCS CONCEPTS

• **Information systems → Personalization**; **Recommender systems**; **Collaborative filtering**.

KEYWORDS

Recommendation, Privacy-Preserving, Privacy Preference, Synthetic Data Generation

1 INTRODUCTION

Recommendation systems, particularly collaborative filtering (CF) systems [17, 20, 28–30, 34], play an important role in various online platforms (e.g., Amazon[1], eBay[2], TikTok[3]). It makes personalized recommendations of products or services to users to help them navigate through an overwhelming amount of information. Most existing CF-based recommendation methods learn the users' and items' representations by exploiting the information from the users' historical interaction data, such as the behavioral information, purchase history, recommendation feedback, etc. As the recommendation methods are heavily data-driven, the more data it uses, the better is the obtained recommendation [20]. However, such users' data is often highly private for numerous personal, social, and financial reasons. The risks concerning violating individuals' privacy present a significant impediment to storing or using the users' private data in an insecure environment. For instance, the user's private data may be shared with various organizations or stored in the public cloud.

In the past few years, privacy issues in recommendation systems have been attracting more attention. Many emerging solutions have been developed by applying privacy-preserving techniques (e.g., Federated learning [32] and Differential privacy [12, 40]) in recommendation systems. For example, Yang et al. [32] proposed a federated learning-based approach, which collaboratively trains recommendation models with users' private data maintained locally at each party (mobile device or organization). The user's private data is stored locally by each party in their approach. Only the intermediate results (e.g., parameter updates) are used to communicate with other parties. Gao et al. [12] adopted a differentially private protection mechanism to help obfuscate the real interactions before reporting them to the server in the training process. Wang et al. [40] applied the differential privacy concept to prevent

---

[1] https://www.amazon.com
[2] https://www.eBay.com
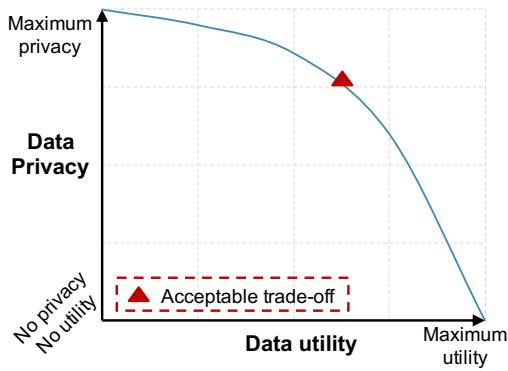[3] https://www.tiktok.com

Figure 1: The trade-off between data privacy and data utility.

user history information recovery through the recommendation results. In general, the existing solutions are presented based on the decentralized framework. Besides, they are designed for the privacy issue in the model training [12, 32] and result collection [40] phases.

Despite their effectiveness, we argue that the existing solutions still suffer from the following three limitations: (1) **Communication and computation cost**. Data transfers and local computing in decentralized methods make it hard to apply these methods in real-world recommendation scenarios; (2) **Data sharing or releasing risks**. There is a considerable risk of privacy leakage while explicitly sharing data with other organizations or storing the raw data in public (e.g., the public cloud); (3) **Users have different privacy preferences for different recommendation scenarios**. For example, users would pay more attention to their medical and financial information privacy protection than for grocery items purchase records. Existing solutions neglect users' privacy preferences while building recommendation system.

The use of privacy-preserving synthetic data provides an alternative method to tackle the privacy issue in the recommendation. The key idea of privacy-preserving synthetic data is to create synthetic data from generative models while preserving as many properties of the original data as possible. As shown in Fig. 1, there is a trade-off between data privacy and data utility. That is, it is mathematically impossible to maintain the full data value without privacy leakage risks. The effectiveness of creating privacy-preserving synthetic data has been demonstrated in some cases. For example, Barak et al. [2] constructed the desired data by adding noise to the Fourier coefficients so that the original data could not be easily re-identified. Cunningham et al. [9] generated synthetic location data from real locations to protect each individual's existence and true location in the original dataset. By introducing privacy-preserving synthetic data in recommendation, the centralized recommendation models can explicitly utilize synthetic data because the sensitive information has been removed from the original data. Besides, the synthetic data can also be released to the public or shared with organizations depending upon users' privacy requirements and consent.

Motivated by the above discussion, in this paper, we present a User Privacy Controllable Synthetic Data Generation model (UPC-SDG for short), which generates privacy-preserving synthetic data from the original users' interaction data that incorporates users'

privacy preferences. The proposed UPC-SDG model can provide a privacy guarantee for the original data at both data level and item level. To be more specific, at the data level, we design a selection module to select the items used to generate the synthetic item. To maximize the utility of synthetic data, we employ an attention mechanism to estimate the contribution of each purchased item to the user's preference. The items that contribute less to user preference are considered to be the selected items. At the item level, a synthetic item generation module is proposed to generate the synthetic item using the selected item while considering the user's preferences. To maximize the utility of synthetic items while providing necessary privacy protection for the original items, we present a privacy-utility trade-off strategy to optimize the item generation process. Extensive experiments on three real-world datasets have been carried out to demonstrate the effectiveness of our model. We have released the code and relevant parameter settings to facilitate repeatability as well as further research [4].

In summary, the main contributions of this work are summarized as follows:

- We highlight the limitations of existing recommendation systems from the privacy perspective. Motivated by this, we propose a novel UPC-SDG model, which generates privacy-preserving synthetic data from the original data under users' privacy preferences.
- At the data level, we propose a selection module to select the items which contribute less to the user's preference. At the item level, a synthetic item generation module is developed to create the corresponding synthetic item.
- To maximize the utility of the synthetic item while providing necessary privacy protection for the original item, we present a privacy-utility trade-off strategy to optimize the synthetic item generation process.
- We have conducted extensive experiments on three real-world datasets to verify our model and perform comprehensive ablation studies to verify the key assumptions of our model. Experimental results demonstrate the effectiveness of our approach.

2 RELATED WORK

2.1 Collaborative filtering

Collaborative filtering (CF) [7, 8, 14, 20–22, 25–28, 31, 34, 35, 43] has long been recognized as an effective approach for building recommendation systems over the past decades. In model-based CF models, users and items are represented as dense vectors (i.e., embeddings) in the same latent space. An interaction function is used to predict a user's preference for an item based on the learned vectors. Take the Matrix Factorization (MF) approach for example, the user and item embeddings are learned by minimizing the error of re-constructing the user-item interaction matrix, and the dot product is used as the interaction function for prediction. Since this simple idea has achieved great success, many variants of MF have been developed [22, 25, 34, 35]. The advent of deep learning has accelerated the development of model-based CF techniques. Due to the powerful capability of deep learning, it has been widely

---

[4]https://github.com/HuilinChenJN/UPC_SDG.

applied in recommendation to learn better user and item embeddings [6, 7, 46] or model more complicated interactions between users and items [20]. For example, NeuMF [20] model employs the deep neural network to model the complex interactions between users and items; A^3NCF [6] learns the user attention from reviews by using the deep learning technique. More recently, Graph Convolution Network (GCN) techniques have also been applied in recommendation and achieved great success [28, 29, 39, 43–45]. The advantage of GCN-based recommendation models is attributed to their capability of explicitly modeling high-order proximity between users and items. For example, NGCF [43] exploits high-order proximity by propagating embeddings on the user-item interaction graph; IMP-GCN [28] learns user and item embeddings by performing high-order graph convolution inside sub-graphs constructed by leveraging users' interests.

All of the methods mentioned above learn the representations of users and items heavily relying on the historical interaction data collected from users in various recommendation scenarios. And their performance suffers when the interactions are insufficient in quantity or when users refuse to share their data due to the sensitive information leakage concern. The use of privacy-preserving synthetic data provides an alternative to combat this problem in recommendation systems.

2.2 Privacy-Preserving Synthetic Data

Open release and free exchange of data would be beneficial for research and development but is not always feasible for sensitive data with privacy implications such as clinical and genomics data.

Solutions have been proposed in two broad categories to address this privacy challenge. In the first category, the data anonymization-based approaches [2, 36] try to use various definitions to sanitize data so that it cannot be easily re-identified. For example, Barak et al. [2] constructed the desired data from the original data via adding noise to the Fourier coefficients. Although these approaches have some important use cases, they are not usually based on rigorous privacy definitions that can withstand various types of re-identification attacks. In the second category, synthetic data generation approaches have been proposed to generate realistic synthetic data using rigorous differential privacy definitions [1, 3, 9]. To maximize the utility of the data, the distribution of the generated synthetic data should be as close as possible to that of the original dataset. Still, it should not contain synthetic examples that are too close to real data instances, as the privacy of the original dataset could be compromised. In particular, Acs et al. [1] first clustered the original datasets into k clusters with private kernel k-means. Afterward, the generative neural network is adopted to produce synthetic data for each cluster. Bindschaedler et al. [3] introduced the idea of plausible deniability instead of adding noise to the generative model directly. A privacy threshold ensures plausible deniability in releasing synthetic data. Here, an adversary cannot tell whether a particular input belongs to the original data by observing synthetic records. Cunningham et al. [9] presented two methods with high practical utility for generating synthetic location data from real locations, both of which protect each individual's existence and true location in the original dataset.

Although these approaches have been shown to work in some cases, there is still a challenge in generating privacy-preserving synthetic data in recommendation scenarios. In this work, we present a privacy-preserving synthetic data generation model for recommendation.

2.3 Generative Models in Recommendation

Generative models have led to rapid development of the deep learning areas, including image [49] and video [10] generation. Numerous studies have also demonstrated the potential of generative techniques to improve recommendation systems by tackling these systems' challenges.

Despite existing recommendation models' pervasiveness and strong performance, recommendation systems still suffer from two main problems: data noise and data sparsity. As an extrinsic problem [13], data noise stems from the casual, malicious, and uninformative feedback in the training data collected from users in different recommendation scenarios. For example, users might purchase products out of their interests. And, this data noise is also collected and leveraged into the training procedure of the recommendation model. Besides, the randomly selected negative samples would also mislead the recommendation models during the training. All in all, the recommendation results could be manipulated by injecting the noisy data into the RS. Several solutions have been proposed for the data noise problem [4, 11, 18, 37, 41, 47]. For instance, He et al. [18] verified the effectiveness of adding adversarial perturbations into a recommendation system, and they proposed an adversarial personalized ranking model to enhance model generalizability. Compared to the data noise problem, data sparsity is an intrinsic problem [13]. It occurs because each user generally only consumes a small proportion of the available items. Existing CF models typically rely on the historical interactive information between users and items when capturing users' interests. When a vast quantity of the data is missing, the CF models commonly fail to capture users' preferences and often provide inaccurate recommendations. To alleviate the data sparsity problem, numerous generative models have been developed to improve the performance of recommendation systems by augmenting them with missing interactive information [5, 33, 38, 42, 48]. For example, Chae et al. [5] generated the purchase vectors of users rather than item IDs; Wang et al. [42] generated interactions for different recommendation tasks under different auxiliary information conditions.

In this work, we focus on the privacy issue in recommendation systems. We present a user privacy controllable synthetic data generation model, which aims to remove sensitive information based on the users' privacy preferences while maximizing the utility of the original data.

3 THE PROPOSED MODEL

3.1 Preliminaries

*3.1.1 **Problem Setting and Notation.*** Before describing our model, we would like to introduce the problem setting first. Given a dataset containing the real user-item interaction data, we aim to generate the synthetic interaction data under users' privacy

preferences [5]. For a user set $\mathcal{U}$ and an item set $\mathcal{I}$, we use $\mathcal{R}^{N_u \times N_i}$ to denote the user-item interaction matrix, in which a nonzero entry $r_{u,i} \in \mathcal{R}$ indicates that user $u \in \mathcal{U}$ has interacted with item $i \in \mathcal{I}$ before; otherwise, the entry is zero. $\mathcal{I}_u$ denotes the set of items that user u has interacted with. Note that the interactions can be either implicit (e.g., click) or explicit (e.g., rating). N_u and N_i are the numbers of users and items, respectively. In our setting, each user and item is assigned a unique ID represented by a trait vector. Let $\mathbf{p_u} \in \mathbb{R}^d$ and $\mathbf{q_i} \in \mathbb{R}^d$ denote the feature vector of the user u and a historical item i, respectively. d is the dimensionality of the latent space. Therefore, for the historical data $\mathcal{I}_u$ of user u, our goal is to generate the synthetic data $\mathcal{V}_u$, in which the items generated by our model are used to replace a certain percentage of the original items. Note that the synthetic users' interaction data contains few or no sensitive information.

To evaluate the utility of the synthetic data, several recommendation models are employed to model users' preferences over the synthetic dataset.

3.1.2 Privacy Definitions. In this work, we provide both data level and item level privacy protection for the original data by using the following two parameters.

Definition 1 (Replacement Ratio). The replacement ratio k is defined to evaluate the privacy guarantee for the original data at the data level. It is certain that if more original items are replaced by synthetic items, the synthetic data could provide higher level privacy guarantees for the original data. For the given original data set $\mathcal{I}_u$ and synthetic data set $\mathcal{V}_u$, the replacement ratio k is defined as:

$$k = \frac{|\mathcal{I}_u \wedge \mathcal{V}_u|}{|\mathcal{I}_u|}. \tag{1}$$

Definition 2 (Sensitivity). The sensitivity γ is defined to describe the privacy guarantee for the original item at the item level. It is obtained by calculating the relative similarity between the original and synthetic data. Given the original data $i \in \mathcal{I}_u$ and the synthetic data $v \in \mathcal{V}_u$, the relative similarity between the original item i and the synthetic item v is calculated by:

$$f_{sim}\left(\boldsymbol{q}_i, \boldsymbol{q}_v\right) = \frac{\boldsymbol{q}_i^T \boldsymbol{q}_v - \min(\boldsymbol{q}_i)}{\boldsymbol{q}_i^T \boldsymbol{q}_i - \min(\boldsymbol{q}_i)}, \tag{2}$$

where $\min(\cdot)$ denotes the cosine similarity between item i and the item which is the most insensitive one in the item set $\mathcal{I}$. In our definition, the synthetic item can meet user's privacy requirement in terms of the sensitivity if $f_{sim}(\boldsymbol{q}_i, \boldsymbol{q}_v) \le \gamma$.

Intuitively, different users have different privacy preferences. Therefore, both $k \in (0, 1)$ and $\gamma \in (0, 1)$ are utilized as parameters which can be adjusted to change the security level of the synthetic data in our proposed model, and the two parameters are controlled by users based on their personal privacy preferences.

3.2 Model Overview

In this section, we present an overview of the proposed User Privacy Controllable Synthetic Data Generation (short for UPC-SDG) model, which aims to generate privacy-preserving synthetic data based on the users' privacy preferences.

It should be noted that UPC-SDG is designed for providing the privacy guarantee to the original data while maximizing the utility of the synthetic data from both data level and item level perspectives. At the data level, we present a selection module, it needs to select the items which will be replaced by the generated items from the original data. The number of items that need to be replaced by the generated synthetic items depends on the user's privacy requirements, which can be adjusted by setting the replacement ratio k. To maximize the utility of the synthetic data at the data level, we present an attention mechanism to estimate all items' contributions to the user's preference. The items that contribute less to the user's preference are selected for replacement by the generated synthetic items. At the item level, a generation module is developed to generate the item from all items based on the users' preferences, the users' privacy preferences and the characteristics of the selected items. The generation module aims to generate the items that are appealing to a user while meeting the user's privacy requirements in terms of sensitivity. In other words, the users' privacy preferences are controllable and could be adjusted by setting the sensitivity γ. In addition, for balancing the privacy and utility between generated items and original items, a privacy-utility trade-off strategy is deployed. The strategy can provide the privacy guarantee of the original item while maximizing the utility of the synthetic item.

3.3 Synthetic Data Generation

The proposed UPC-SGD model aims to provide privacy guarantees while maximizing the utility for the original user interaction data at both the data level and the item level. Next, we detail the model for privacy-preserving synthetic data generation.

3.3.1 Data level. For a user u, her privacy is protected if the non-sensitive items replace the sensitive items in the original data. In this work, the number of items that need to be replaced is determined using the replacement ratio k. The more items are replaced, the less data leakage risk there is. To maximize the utility of the synthetic data at the data level, we design a selection module to find the items which contribute less to the user's preference than the other items in the original data.

Selection Module. It has shown that historically interacted items have different contributions to model users' preferences [8, 19]. Therefore, by employing the representations of items that a user u purchased, the user's preferences can also be presented as follows:

$$t_u = \frac{1}{|\mathcal{I}_u|} \sum_{i \in \mathcal{I}_u} a_{ui} q_i, \tag{3}$$

where $a_{ui} \in A_u$ is a trainable parameter that denotes the attention weight of item i in contributing to user u's preference; A_u represents the weight set of the purchased items by the user u. Formally, the attention weight a_{ui} is estimated by an attention mechanism which is described as following:

$$v_{ui} = \mathbf{h}^T ReLU(\mathbf{W}_1([\mathbf{p}_u : \mathbf{q}_i]) + \mathbf{b}_1), \tag{4}$$

$$a_{ui} = \frac{\exp(v_{ui})}{[\sum_{j' \in \mathcal{I}_u} \exp(v_{ui'})]^\beta}, \tag{5}$$

where W_1 and b_1 are respectively the weight matrix and bias vector that project the input onto a hidden layer, and $\mathbf{h}^T$ is the vector that projects the output of the hidden layer to an output attention weight v_{ui}. β is the smoothing exponent. It is employed as a hyperparameter which is set in $[0, 1]$. We use the Rectified Linear Unit (ReLU) as the activation function of the hidden layer.

As both p_u and t_u can capture user's preferences, we employ p_u as the supervision information to learn the attention weight a_{ui}. To be specific, We define the following L_2 regularizer:

$$\mathcal{L}_D = \sum_{u \in \mathcal{U}} \| f(t_u, \theta) - p_u \|^2, \qquad (6)$$

where we learn a transformation function $f(\cdot)$ that projects t_u into the common representation space with p_u. θ denotes the parameters of function $f(\cdot)$. Note that we employ a multi-layer perceptron (MLP) with dropout as our transformation function $f(\cdot)$ for its superior representational capacity and ease of training. The parameters of the MLP are trainable. During the training, we minimize $\mathcal{L}_D$ to estimate the contributions of each item to the user preference. In this way, the attention weights of all purchased items for user u are obtained. With the obtained attention weights, the items that contribute less to the user's preference are selected as the replaced item:

$$I_u^r = \{ i \mid i \in I_u, a_{ui} \in A_u^k \}, \qquad (7)$$

where A_u^k denotes the k percent of the items, the weights of which are smaller than the others.

*3.3.2   **Item level**.* With the selected items I_u^r, the synthetic items for the user u can be generated. One of the targets of generating the synthetic item is that the generated item can also attract user u' attention. In other words, to maximize the utility of the synthetic item, we present a generation module that generates the item considering the users' preferences, the users' privacy preferences and the characteristics of the selected items.

Generation Module. For generating the synthetic item, we first concatenate the user vector p_u, the selected item vector q_i ($i \in I_u^r$) and privacy parameter γ as an input, then project the input into a latent space:

$$R_{u,i} = W_2 [p_u; q_i; \gamma_u] + b_2, \qquad (8)$$

where $R_{u,i}$ is the latent feature of the output. W_2 and b_2 are respectively the weight matrix and bias vector. Let $E_I \in \mathbb{R}^{N_i \times d}$ represents the embeddings of all items, we calculate the similarity between the latent feature $R_{u,i}$ and all item embeddings:

$$h_{u,i} = R_{u,i} E_I^T. \qquad (9)$$

The output $h_{u,i}$ whose elements represent the similarity of the latent feature $R_{u,i}$ with each item embedding, which is going to be learned during training. Finally, we estimate the probability distribution over all candidate items. It is formulated as:

$$y_{u,i} \sim softmax(h_{u,i}), \qquad (10)$$

where $y_{u,i}$ represents the probability distribution over all possible items. Intuitively, the item with the highest possibility should be sampled as the generated synthetic item for the selected item. However, there is a crucial problem with sampling that the discrete sampling procedure is non-differentiable. In other words, the training of the generation module is not end-to-end, resulting in an inferior

performance. The recently proposed Gumbel-Softmax [15, 23] can approximate categorical samples by applying a differentiable procedure. Thus, Gumbel-Max trick is proposed to re-parameterize the sampling operator as below:

$$y_{u,i} = one\_hot(argmax_{1 \le k \le N_i}(h_{u,i}^k + g_{u,i}^k)), \qquad (11)$$

where $h_{u,i}^k \in h_{u,i}$ and $g_{u,i}^k \in g_{u,i}$. The $g_{u,i}^k$ is an independent variable and follow a Gumbel distribution $g_{u,i}^k = -\log(-\log(\mu))$ whith $\mu \sim Uniform(0, 1)$. However, the "one_hot" and "argmax" operations are still non-differentiable. To deal with this problem, we further approximate them by softmax, which yields

$$\tilde{y_{u,i}} = softmax(\frac{1}{\tau}(h_{u,i} + g_{u,i})). \qquad (12)$$

Note that $\tau > 0$ is a tunable parameter called temperature to control the softness. As $\tilde{y_{u,i}}$ is differentiable with respect to $h_{u,i}$, it can be used instead of $y_{u,i}$.

After the scores $y_{u,i}$ for candidate items are obtained, we sample the item not consumed by the user u with the Gumbel-Softmax as shown in Eq. 12. Finally, the generated item is delivered to the loss function to help optimize the generation process. If the generated synthetic item has properties similar to that of the selected item, there is still the risk of privacy leakage. Hence, for providing privacy guarantees while maximizing the utility of the synthetic item, we propose a privacy-utility trade-off strategy at the item level.

Privacy-utility Trade-off Strategy. To provide privacy guarantees at the item level, we employ a privacy regularizer to constrain the differences of relative similarity between the selected original item and the generated synthetic item. The privacy regularizer is defined as:

$$\mathcal{L}_s = \sum_{(u,i,v)} [f_{sim}(q_i, q_v) - \gamma_u]_+. \qquad (13)$$

where $[z]_+ = max(z, 0)$ denotes the standard hinge loss. For a user u, the sensitivity γ_u is adopted as a safety margin that the similarity between the selected original item and the generated synthetic item is tolerated within a certain range.

Inspired by the recommendation approach in [20], we assume that a user u would give the synthetic item a higher score if she prefers this item. To maximize the utility of synthetic items, we aim to generate the items (i.e., v) that can attract the user's attention. To achieve the goal, we construct a user-item pair of $\{u, v\}$ with a generated interaction between user u and item v, and define the following utility regularizer:

$$\mathcal{L}_g = \sum_{(\mathbf{u},\mathbf{v})} -\ln \sigma(p_u^T q_v). \qquad (14)$$

The final loss function of the privacy-utility trade-off strategy is formulated as:

$$\mathcal{L}_I = \lambda_s \mathcal{L}_s + \lambda_g \mathcal{L}_g, \qquad (15)$$

where λ_s and λ_g are hyperparameters that control the weights of the privacy regularizer and utility regularizer, respectively. The loss function of the generation model can be interpreted as the model trying to produce such items that the user prefers. Finally, the proposed privacy-utility trade-off strategy finely balances the privacy and utility of the synthetic item at the item level.

3.4 Optimization

In this work, we focus on generating privacy-preserving synthetic data for the Top-N recommendation. With the above consideration of the selection and generation module, the final objective function for optimizing the parameters of UPC-SGD is:

$$\mathcal{L} = \mathcal{L}_D + \mathcal{L}_I. \tag{16}$$

The mini-batch Adam [24] optimizer is used for optimizing the generation model. And the model parameters are updated by using the gradients of the loss function.

4 EXPERIMENT

To evaluate the effectiveness of our proposed UPC-SDG model, we perform extensive experiments on three publicly real-world datasets. Through the experiments, we attempt to answer the following research questions.

- **RQ1:** How is the existing recommendation models' performance on the synthetic datasets generated under different privacy settings?
- **RQ2:** Does the synthetic data remove the sensitive information from the original data?
- **RQ3:** How do different components of our model affect the performance?

Next, we introduce the experimental setup and then sequentially report and analyze the experimental results to answer the above questions.

4.1 Experimental Setup

4.1.1 Datasets. To evaluate the effectiveness of UPC-SDG, we conducted experiments for the generation task on three public real-world datasets: *Office*, *Clothing*, and *Gowalla*.

The first two datasets are from the public Amazon review dataset [6], widely used for recommendation evaluation in previous studies [27] and adopted for experiments in this work. The third dataset is a check-in dataset collected from Gowalla [43], where users share their locations by checking in. Following the settings in previous studies [27, 43], we pre-process all the datasets to keep only the user and items with at least 10 interactions. The basic statistics of the three datasets are shown in Table 1. We can see that the three datasets are of different sizes and sparsity. For example, the *Office* dataset is denser, and the *Gowalla* dataset is larger than other datasets. The diversity of datasets helps analyze the performance of our method in different situations.

In this work, we focus on the privacy-preserving synthetic data generation task, which aims to generate interaction data for each user under the user's privacy preference. Besides, we verify the utility of the generated data based on the recommendation task. Thus, for each dataset, we randomly split each dataset into training, validation, and testing sets with a ratio of 80:10:10 for each user. The observed user-item interactions in the original three datasets were treated as the input feature in the generation task.

For the recommendation task, the experiments are conducted over the generated datasets. The observed user-item interactions

Table 1: Basic statistics of the experimental datasets.

Dataset	#user	#item	#interactions	sparsity
Office	4,874	2,405	52,957	99.55%
Clothing	18,209	17,317	150,889	99.95%
Gowalla	29,858	40,981	1,027,370	99.91%

in the generated datasets were treated as positive instances. In addition, for the recommendation methods which adopt the pairwise learning strategy, we randomly sample a negative instance that the user did not consume before, to pair with each positive instance.

4.1.2 Evaluation Metrics. We evaluate the utility of our generated interaction data on the top-n recommendation task. For each user in the test set, we treat all the items that the user did not interact with as negative items. Three widely used metrics for the recommendation are used in our evaluation: *Precision* (PR), *Recall* and *Normalized Discounted Cumulative Gain* (NDCG) [16]. For each metric, the performance is computed based on the top 20 results. Notice that the reported results are the average values across all the testing users.

4.1.3 Recommendation Models. This work aims to generate privacy-preserving synthetic data for users in recommendation systems. Therefore, we explore the utility of the generated interaction data with the following recommendation models, such as Random [8], BPRMF [34], NeuMF [20] and LightGCN [17].

- **Random** recommends items randomly to users.
- **BPRMF** [34] is a classic MF-based method for top-n recommendation. It only utilizes the user-item interaction data and uses the Bayesian personalized ranking loss to learn user and item embeddings.
- **NeuMF** [20] is a neural collaborative filtering method, which employs multiple hidden layers above the element-wise to capture the non-linear interactions between users and items.
- **LightGCN** [17] is a simplified GCN-based recommendation model, which explicitly encodes the collaborative signal of high-order neighbors by performing embedding propagation in the user-item bipartite graph.

In the experiments on recommendation tasks, for all the recommendation models, we use the code that their authors released for evaluation. And, we put a lot of effort to tune the hyperparameters of these models and reported their best performance.

4.1.4 Experimental Settings. Our proposed UPC-SDG model is implemented with PyTorch [7] and its key parameters are carefully tuned. The pre-trained user and item embeddings [8] are adopted as user and item features in our model, and the embedding size is fixed to 64. We optimized our method with Adam [24] and the learning rate is searched in the range of $\{1e^{-1}, 1e^{-2}, \cdots, 1e^{-5}\}$. Besides, to verify the controllability of the selection module and the generation module of UPC-SDG, we generate various synthetic data for each user with different privacy settings by adjusting

Table 2: Performance comparison of recommendation models over the original and generated datasets. Specifically, we use the proposed UPC-SDG model to separately generate two synthetic datasets for each original dataset under two privacy settings. Such as $Office^-$ is generated with $k = 0.8$ and $\gamma = 0.1$; and $Office^+$ is generated with $k = 0.2$ and $\gamma = 0.9$. In addition, "-" indicates that the recommendation model has the same performance over the generated datasets as the original datasets.

Models	Random			BRPMF			NeuMF			LightGCN		
Metrics	PR	Recall	NDCG	PR	Recall	NDCG	PR	Recall	NDCG	PR	Recall	NDCG
Office	0.0021	0.0118	0.0054	0.0215	0.1254	0.0740	0.0126	0.0605	0.0710	0.0223	0.1323	0.0752
Clothing	0.0001	0.0019	0.0006	0.0039	0.0526	0.0243	0.0120	0.0668	0.0303	0.0043	0.0692	0.0312
Gowalla	0.0005	0.0012	0.0008	0.0475	0.1565	0.1277	0.0285	0.1017	0.1548	0.0523	0.1823	0.1555
$Office^-$	-	-	-	0.0032	0.0176	0.0080	0.0058	0.0316	0.0374	0.0046	0.0235	0.0125
$Clothing^-$	-	-	-	0.0001	0.0023	0.0026	0.0007	0.0109	0.0044	0.0020	0.0345	0.0148
$Gowalla^-$	-	-	-	0.0035	0.0121	0.0073	0.0030	0.0088	0.0213	0.0047	0.0148	0.0100
$Office^+$	-	-	-	0.0166	0.0878	0.0506	0.0121	0.0580	0.0702	0.0148	0.0839	0.0496
$Clothing^+$	-	-	-	0.0025	0.0422	0.0187	0.0110	0.0553	0.0245	0.0038	0.0639	0.0281
$Gowalla^+$	-	-	-	0.0305	0.1051	0.0740	0.0185	0.0736	0.0870	0.0353	0.1257	0.8120

the replacement ratio $k \in \{0.2, 0, 4, 0, 6, 0.8\}$ and the sensitivity $\gamma \in \{0.1, 0, 3, 0, 5, 0, 7, 0.9\}$. For balancing the effect of the privacy regularizer and utility regularizer, we set $\lambda_s = 3$ and $\lambda_g = 1$ for *Office* and *Gowalla*, and set $\lambda_s = 5$ and $\lambda_g = 7$ for *Clothing* in generation task.

For the recommendation task, the embeddings size is set to 64 for all recommendation models. We used the default mini-batch size of 2048 for all datasets. The L_2 regularization coefficient is searched in the range of $\{1e^{-1}, 1e^{-2}, \cdots, 1e^{-6}\}$. All the recommendation models have also been carefully tuned for a fair comparison. In addition, the early stopping and validation strategies are kept the same as those in LightGCN [17].

4.2 Utility Analysis (RQ1)

*4.2.1 **Performance Comparison**.* The results of existing recommendation models over the original datasets (*Office*,*Clothing*,*Gowalla*) and the datasets generated based on the three datasets under different privacy settings are reported in Table 2 in terms of PR@20, Recall@20 and NDCG@20. Specifically, we use the proposed UPC-SDG model to generate two synthetic datasets separately based on each dataset under two privacy settings. For example, $Office^-$ is generated with $k = 0.8$, $\gamma = 0.1$; and $Office^+$ is generated with $k = 0.2$, $\gamma = 0.9$. Similarly, the synthetic datasets $Clothing^-$, $Clothing^+$, $Gowalla^-$ and $Gowalla^-$ are obtained.

We firstly focused on the performance of the recommendation models over the original datasets. Specifically, the performance of Random is poor as it randomly recommends the items to users. Benefiting from the technique of matrix factorization, BPRMF can achieve better performance than the random method. The reason is that the user and item representations are learned by exploiting the interaction data. NeuMF obtains better performance over BPRMF, because it utilizes neural networks to model user-item interactions. LightGCN outperforms both BPRMF and NeuMF by a large margin as it explicitly leverages the high-order connectivities between users and items.

The second and third blocks are the performance of the recommendation models over the generated datasets [9]. $Office^-$,$Clothing^-$

and $Gowalla^-$ are generated under the higher privacy level ($k = 0.8$, $\gamma = 0.1$). As we can see, the recommendation models can still model users' preferences over these datasets. On the contrary, $Office^+$, $Clothing^+$ and $Gowalla^+$ are generated under the lower privacy level ($k = 0.2$, $\gamma = 0.9$). The recommendation models can achieve comparable performance with them over the original datasets. The experimental results demonstrate the utility of the generated datasets.

*4.2.2 **Effect of Privacy Preference**.* To investigate the utility of the generated datasets under different privacy preferences, we generated various datasets based on the *Office*, *Clothing* and *Gowalla* under different privacy settings. Fig. 2 shows the performance of the BPRMF over various generated datasets. From the results, we obtained some interesting observations:

- The results indicate that with the increase of replacement ratio k and the decrease of sensitivity γ, the performance of the recommendation model decreases as expected. Moreover, the generated dataset can still be used to model users' preferences, even when the replacement ratio k is set as 0.8 and the sensitivity γ is set as 0.1. This also demonstrates the effectiveness of our model in retaining the utility of synthetic data. Note that the performance of BPRMF drops when γ is set as 0.9. The reason might be that the generated item is very similar to the items in the original user interaction data. Thus, the diversity of the items in each user's interaction data is reduced, which leads to sub-optimal performance.
- The best trade-off point is different for users in different recommendation scenarios. From the perspective of Sensitive γ, the best trade-off points are 0.5, 0.7, 0.5 on *Office*, *Clothing*, and *Gowalla*, respectively [10]. In other words, users need to give up more privacy in exchange for recommendation performance on *Clothing*.
- The sensitivity γ has different impact on different datasets. From the results, the impact of sensitivity adjustment on *Gowalla* is lower than in the other datasets. That might be

---

[9]Random has similar performance over both the original and generated datasets.

[10]This depends on whether the recommendation performance will degrade rapidly compared with the recommendation performance over the generated datasets of other settings.
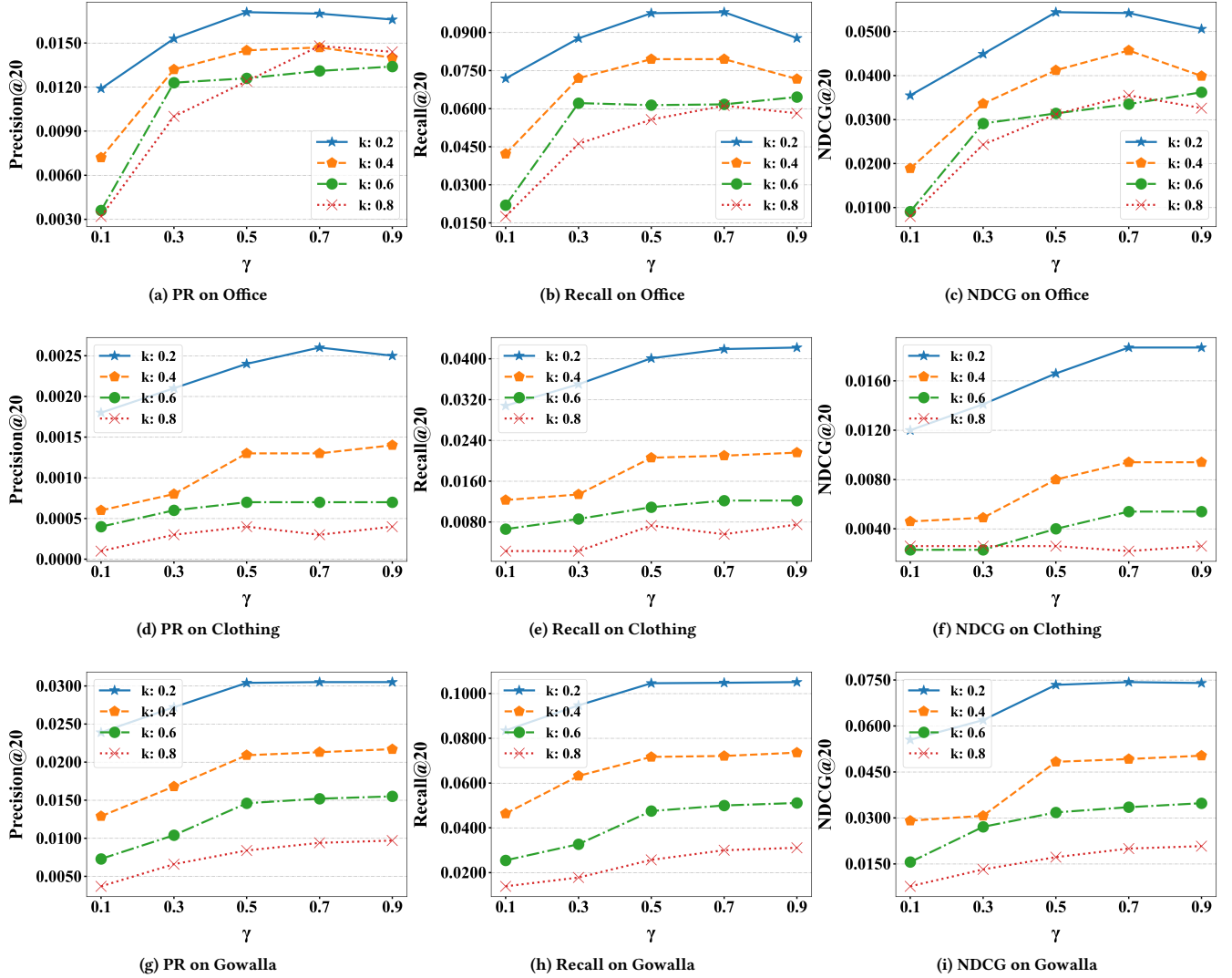
Figure 2: Performance comparison of BPRMF over the generated datasets of different privacy settings. k and γ represent replacement ratio and sensitivity, respectively.

because the designed generation model introduces numerous parameters, and the model is difficult to train on smaller datasets.

- The replacement ratio k has different impact on different datasets. From the results, the impact of replacement adjustment on *Office* is lower than in the other datasets. The reason is that *Office* is denser than other datasets. With rich interactions, users' preferences could still be modeled well even if some of the interaction data are replaced by synthetic data.

Overall, the proposed UPC-SDG model can generate privacy-preserving synthetic data for recommendation systems. And it can provide privacy guarantees for the original private data while maximizing the utility of the synthetic data.

4.3 Privacy Analysis (RQ2)

4.3.1 ***Case Study***. Employing the privacy-utility trade-off strategy, we can provide a privacy guarantee for the original item while maximizing the utility of the generated item. Towards this end, we selected one user and the user's interaction data both from *Clothing* and the generated dataset based on *Clothing* under specific privacy settings ($k = 0.2$).

Figure 3 shows the visualization of the original and generated data of the user. The original and generated user data are listed in rows. In contrast, the items in the original and generated user data are listed in columns. The items in the first two columns are the selected items in the original data and generated items in the generated user data. More specifically, the first row represents the

Sensitivity	1	2	3	4	5	6
Original Data						
0.9						
0.7						
0.5						
0.3						
0.1						

Figure 3: Examples of privacy-preserving synthetic data on different sensitivities. The items that need to be replaced in the original and generated synthetic data under different sensitives are circled in the red box.

user's original data from the Clothing dataset. From the perspective of the original items, we can find that the user likes jewelry a lot.

Our UPC-SDG model generates synthetic items by considering the user's preferences, the users' privacy preferences and the selected item's characteristics. From the items of the first two columns, it can be found that the generated synthetic items have more similar characteristics to the original item with increasing sensitivity. For example, when sensitivity γ is set as 0.9 and 0.7, the generated items belong to the same category as that of the original items. With the decrease in sensitivity, the difference between the original and generated items becomes greater. Other than the differences between the original and generated items, we can also find that the generated item keeps up with the user's preferences. For instance, when γ is set as 0.5, the model is inclined to generate the accessory items in line with the user's preference. Finally, when γ is set as 0.3 and 0.1, very different items are generated.

Overall, we perform visualization on the items to compare the original and generated user data, which demonstrates the effectiveness of our proposed model. In other words, the proposed model can remove the sensitive information from the user data by considering their privacy preference while maximizing the utility of the generated data.

*4.3.2 **Visualization of Similarity**.* For further analysis, we visualized the average similarity between the generated items and the original items for all users under different sensitivity settings. As shown in Figure 4, the x-axis represents the sensitivity that is preset by users before data generation, and the y-axis represents the average similarity calculated between the original items and
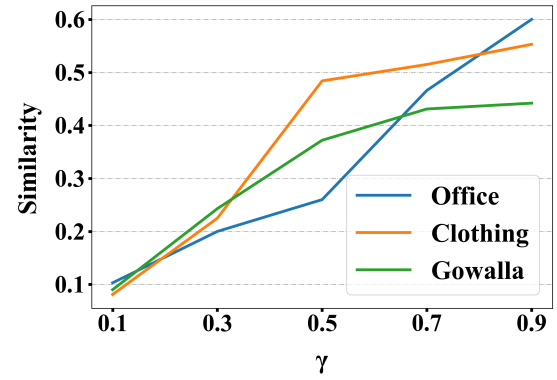
Figure 4: Visualization of the similarity between the generated synthetic items and original items. γ denotes the sensitivity.

the generated items. From the figure, the sensitivity and similarity are positively correlated over the three datasets. The results demonstrate that the proposed model can provide privacy guarantees for users' interaction data, and the privacy level is controllable.

4.4 Ablation Study (RQ3)

In this section, we examined the contributions of different components in our model to the final performance by comparing UPC-SDG with the following variants:

Table 3: Performance of BPRMF over the datasets generated by our proposed model and its variants. The datasets are generated based on *Office* under $\gamma = 0.9$ and $k_i \in \{0.2, 0.4, 0.6, 0.8\}$, for $i = 1, 2, 3, 4$.

Dataset	Office$_{k1}$			Office$_{k2}$			Office$_{k3}$			Office$_{k4}$		
Metrics	PR	Recall	NDCG	PR	Recall	NDCG	PR	Recall	NDCG	PR	Recall	NDCG
UPC-SDG$_s$	0.0028	0.0466	0.0215	0.0022	0.0356	0.0160	0.0016	0.0261	0.0118	0.0008	0.0143	0.0056
UPC-SDG$_r$	0.0030	0.0495	0.0227	0.0020	0.0325	0.0151	0.0014	0.0221	0.0098	0.0002	0.0027	0.0012
UPC-SDG$_f$	0.0030	0.0497	0.0229	0.0021	0.0338	0.0154	0.0016	0.0252	0.0109	0.0007	0.0105	0.0047
UPC-SDG	**0.0166**	**0.0878**	**0.0506**	**0.0140**	**0.0717**	**0.0399**	**0.0134**	**0.0646**	**0.0362**	**0.0048**	**0.0582**	**0.0326**

- **UPC-SDG$_s$** is a variant of our model which adopts a random strategy to replace the selection module.
- **UPC-SDG$_r$** is a variant of our method which adopts a random strategy to replace the generation module.
- **UPC-SDG$_f$** is a variant of our method which generates items by fixing the similarity between the selected item and the generated item.

The results of BPRMF over the datasets generated by the three variants and our model are reported in Table 3, in which the best results are highlighted in bold. We conducted ablation experiments over the generated dataset based on *Office* under different replacement ratios and the same sensitivity[11]. UPC-SDG$_f$ outperforms UPC-SDG$_r$ over all datasets, because the similarity between the generated item and the selected item is set as 0.9. Due to the close similarity, recommendation models can learn better user representation by exploiting the generated items. It can be seen that our model consistently surpasses all variants across all metrics, which shows the reasonable design of our proposed model.

5 CONCLUSION

In this paper, we present a User Privacy Controllable Synthetic Data Generation (UPC-SDG) model, which learns the characteristics of the original data and creates synthetic data based on users' privacy preferences for recommendation. Existing solutions are designed for addressing privacy issues in the training and results collection phase of recommendation systems. Users can still suffer from the problem of privacy leakage if their behavior data is either shared with organizations or are released publicly. Therefore, the proposed UPC-SDG model aims to provide certain guarantees about privacy while maximizing the utility of synthetic data from both data level and item level perspectives. In light of this, at the data level, we design a selection module to select the items which contribute less to a user's preferences as the selected items need to be replaced. At the item level, a synthetic data generation module is proposed to generate the synthetic items using the selected items by considering the user's preferences. Moreover, we also present a privacy-utility trade-off strategy to balance privacy and utility. Extensive experiments have been conducted on three real-world datasets from Amazon and Gowalla. The experimental results demonstrate the effectiveness of our model. The ablation studies further validate the importance of the various components of our model.

REFERENCES

[1] Gergely Acs, Luca Melis, Claude Castelluccia, and Emiliano De Cristofaro. 2017. Differentially Private Mixture of Generative Neural Networks. In *ICDM*. 715–720.
[2] Boaz Barak, Kamalika Chaudhuri, Cynthia Dwork, Satyen Kale, Frank McSherry, and Kunal Talwar. 2007. Privacy, Accuracy, and Consistency Too: A Holistic Solution to Contingency Table Release. In *PODS*. ACM, 273–282.
[3] Vincent Bindschaedler, Reza Shokri, and Carl A. Gunter. 2017. Plausible Deniability for Privacy-Preserving Data Synthesis. *Proc. VLDB Endow.* (2017), 481–492.
[4] Xiaoyan Cai, Junwei Han, and Libin Yang. 2018. Generative Adversarial Network Based Heterogeneous Bibliographic Network Representation for Personalized Citation Recommendation. In *AAAI*. AAAI Press, 5747–5754.
[5] Dong-Kyu Chae, Jin-Soo Kang, Sang-Wook Kim, and Jung-Tae Lee. 2018. CFGAN: A Generic Collaborative Filtering Framework Based on Generative Adversarial Networks. In *CIKM*. ACM, 137–146.
[6] Zhiyong Cheng, Ying Ding, Xiangnan He, Lei Zhu, Xuemeng Song, and Mohan S Kankanhalli. 2018. A3NCF: An Adaptive Aspect Attention Model for Rating Prediction.. In *IJCAI*. Morgan Kaufmann, 3748–3754.
[7] Zhiyong Cheng, Ying Ding, Lei Zhu, and Kankanhalli Mohan. 2018. Aspect-aware latent factor model: Rating prediction with ratings and reviews. In *WWW*. IW3C2, 639–648.
[8] Zhiyong Cheng, Fan Liu, Shenghan Mei, Yangyang Guo, Lei Zhu, and Liqiang Nie. 2022. Feature-Level Attentive ICF for Recommendation. *TOIS* 40, 4, 1–24.
[9] Teddy Cunningham, Graham Cormode, and Hakan Ferhatosmanoglu. 2021. Privacy-Preserving Synthetic Location Data in the Real World. In *SSTD*. ACM, 23–33.
[10] Hehe Fan and Mohan Kankanhalli. 2021. Motion = Video - Content: Towards Unsupervised Learning of Motion Representation from Videos. In *ACM Multimedia Asia*. ACM, 1–7.
[11] Wenqi Fan, Tyler Derr, Yao Ma, Jianping Wang, Jiliang Tang, and Qing Li. 2019. Deep Adversarial Social Recommendation. In *IJCAI*. AAAI Press, 1351–1357.
[12] Chen Gao, Chao Huang, Dongsheng Lin, Depeng Jin, and Yong Li. 2020. DPLCF: Differentially Private Local Collaborative Filtering. In *SIGIR*. ACM, 961–970.
[13] Min Gao, Junwei Zhang, Junliang Yu, Jundong Li, Junhao Wen, and Qingyu Xiong. 2021. Recommender systems based on generative adversarial networks: A problem-driven perspective. *INS* 546 (2021), 1166–1185.
[14] Yangyang Guo, Zhiyong Cheng, Jiazheng Jing, Yanpeng Lin, Liqiang Nie, and Meng Wang. 2020. Enhancing Factorization Machines with Generalized Metric Learning. *TKDE* (2020), 1–1.
[15] Serhii Havrylov and Ivan Titov. 2017. Emergence of Language with Multi-Agent Games: Learning to Communicate with Sequences of Symbols. In *NIPS*. CAI., 2146–2156.
[16] Xiangnan He, Tao Chen, Min-Yen Kan, and Xiao Chen. 2015. TriRank: Review aware Explainable Recommendation by Modeling Aspects. In *CIKM*. ACM, 1661–1670.
[17] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR*. ACM, 639–648.

---

[11]We omitted the results on *Clothing* and *Gowalla* because they show exactly the same trend.

[18] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. 2018. Adversarial personalized ranking for recommendation. In *SIGIR*. ACM, 355–364.
[19] Xiangnan He, Zhankui He, Jingkuan Song, Zhenguang Liu, Yu-Gang Jiang, and Tat-Seng Chua. 2018. NAIS: Neural attentive item similarity model for recommendation. *TKDE* 30 (2018), 2354–2366.
[20] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *WWW*. ACM, 173–182.
[21] Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. 2017. Collaborative metric learning. In *WWW*. IW3C2, 193–201.
[22] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *ICDM*. IEEE, 263–272.
[23] Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical Reparameterization with Gumbel-Softmax. (2017).
[24] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.
[25] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *SIGKDD*. ACM, 426–434.
[26] Yehuda Koren. 2010. Factor in the neighbors: Scalable and accurate collaborative filtering. *TKDD* 4 (2010), 1–24.
[27] Fan Liu, Zhiyong Cheng, Changchang Sun, Yinglong Wang, Liqiang Nie, and Mohan S. Kankanhalli. 2019. User Diverse Preference Modeling by Multimodal Attentive Metric Learning. In *MM*. ACM, 1526–1534.
[28] Fan Liu, Zhiyong Cheng, Lei Zhu, Zan Gao, and Liqiang Nie. 2021. Interest-aware Message-Passing GCN for Recommendation. In *WWW*. IW3C2, 1296–1305.
[29] Fan Liu, Zhiyong Cheng, Lei Zhu, Chenghao Liu, and Liqiang Nie. 2020. A2-GCN: An Attribute-aware Attentive GCN Model for Recommendation. In *TKDE*. 1–1.
[30] Han Liu, Yangyang Guo, Jianhua Yin, Zan Gao, and Liqiang Nie. 2022. Review polarity-wise recommender. *IEEE TNNLS* (2022), 1–1.
[31] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. 2008. One-class collaborative filtering. In *ICDM*. IEEE, 502–511.
[32] Yang Qiang. 2019. Federated Recommendation Systems. In *ICBD*. 1–1.
[33] Dimitrios Rafailidis and Fabio Crestani. 2019. Adversarial Training for Review-Based Recommendations. In *SIGIR*. ACM, 1057–1060.
[34] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*. AUAI, 452–461.
[35] Ruslan Salakhutdinov and Andriy Mnih. 2007. Probabilistic Matrix Factorization. In *NIPS*. CAI, 1257–1264.
[36] Latanya Sweeney. 2002. K-Anonymity: A Model for Protecting Privacy. *UFKS* 10, 5 (2002), 557–570.
[37] Yuzhen Tong, Yadan Luo, Zheng Zhang, Shazia Sadiq, and Peng Cui. 2019. Collaborative Generative Adversarial Network for Recommendation Systems. In *ICDE Workshops*. 161–168.
[38] Thanh Tran, Renee Sweeney, and Kyumin Lee. 2019. Adversarial Mahalanobis Distance-Based Attentive Song Recommender for Automatic Playlist Continuation. In *SIGIR*. ACM, 245–254.
[39] Rianne van den Berg, Thomas N. Kipf, and Max Welling. 2018. Graph Convolutional Matrix Completion. In *SIGKDD: Deep Learning Day*. ACM.
[40] Jun Wang and Qiang Tang. 2017. Differentially Private Neighborhood-Based Recommender Systems. In *SEC*. SIP, 459–473.
[41] Qinyong Wang, Hongzhi Yin, Zhiting Hu, Defu Lian, Hao Wang, and Zi Huang. 2018. Neural Memory Streaming Recommender Networks with Adversarial Training. In *SIGKDD*. ACM, 2467–2475.
[42] Qinyong Wang, Hongzhi Yin, Hao Wang, Quoc Viet Hung Nguyen, Zi Huang, and Lizhen Cui. 2019. Enhancing Collaborative Filtering with Generative Augmentation. In *SIGKDD*. ACM, 548–556.
[43] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *SIGIR*. ACM, 165–174.
[44] Yinwei Wei, Zhiyong Cheng, Xuzheng Yu, Zhou Zhao, Lei Zhu, and Liqiang Nie. 2019. Personalized Hashtag Recommendation for Micro-videos. In *MM*. ACM, 1446–1454.
[45] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, and Tat-Seng Chua. 2020. Graph-Refined Convolutional Network for Multimedia Recommendation with Implicit Feedback. In *MM*. ACM, 3541–3549.
[46] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. 2017. Deep Matrix Factorization Models for Recommender Systems.. In *IJCAI*. AAAI Press, 3203–3209.
[47] Feng Yuan, Lina Yao, and Boualem Benatallah. 2019. Adversarial Collaborative Auto-encoder for Top-N Recommendation. In *IJCNN*. 1–8.
[48] Wei Zhao, Benyou Wang, Jianbo Ye, Yongqiang Gao, Min Yang, and Xiaojun Chen. 2018. PLASTIC: Prioritize Long and Short-term Information in Top-n Recommendation using Adversarial Training. In *IJCAI*. AAAI Press, 3676–3682.
[49] Na Zheng, Xuemeng Song, Zhaozheng Chen, Linmei Hu, Da Cao, and Liqiang Nie. 2019. Virtually Trying on New Clothing with Arbitrary Poses. In *MM*. ACM, 266–274.