Studying the Impact of Data Disclosure Mechanism in Recommender Systems via Simulation

ZIQIAN CHEN, Alibaba Group, China
FEI SUN, Alibaba Group, China
YIFAN TANG, Alibaba Group, China
HAOKUN CHEN, Alibaba Group, China
JINYANG GAO, Alibaba Group, China
BOLIN DING, Alibaba Group, United States

Recently, privacy issues in web services that rely on users' personal data have raised great attention. Despite that recent regulations force companies to offer choices for each user to opt-in or opt-out of data disclosure, real-world applications usually only provide an "all or nothing" binary option for users to either disclose all their data or preserve all data with the cost of no personalized service.

In this paper, we argue that such a binary mechanism is not optimal for both consumers and platforms. To study how different privacy mechanisms affect users' decisions on information disclosure and how users' decisions affect the platform's revenue, we propose a privacy aware recommendation framework that gives users fine control over their data. In this new framework, users can proactively control which data to disclose based on the trade-off between anticipated privacy risks and potential utilities. Then we study the impact of different data disclosure mechanisms via simulation with reinforcement learning due to the high cost of real-world experiments. The results show that the platform mechanisms with finer split granularity and more unrestrained disclosure strategy can bring better results for both consumers and platforms than the "all or nothing" mechanism adopted by most real-world applications.

CCS Concepts: • **Information systems → Recommender systems**; • **Security and privacy → Privacy protections**.

Additional Key Words and Phrases: Recommender System; Privacy; GDPR

ACM Reference Format:
Ziqian Chen, Fei Sun, Yifan Tang, Haokun Chen, Jinyang Gao, and Bolin Ding. 2022. Studying the Impact of Data Disclosure Mechanism in Recommender Systems via Simulation. *ACM Transactions on Information Systems* 37, 4, Article 111 (August 2022), 26 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Recommender systems play an essential role on today's web service platforms, e.g., e-commerce [46, 77] and social media [14, 81], since they can reduce users' cognitive load by automatically offering personalized services that match their interests and needs [10]. While the recommender systems

Authors' addresses: Ziqian Chen, eric.czq@alibaba-inc.com, Damo Academy, Alibaba Group, Hangzhou, Zhejiang, 311121, China; Fei Sun, ofey.sf@alibaba-inc.com, Damo Academy, Alibaba Group, Beijing, 100102, China; Yifan Tang, yifan.tang95@gmail.com, Luohan Academy, Alibaba Group, Hangzhou, Zhejiang, 311121, China; Haokun Chen, hankel.chk@alibaba-inc.com, Damo Academy, Alibaba Group, Hangzhou, Zhejiang, 311121, China; Jinyang Gao, jinyang.gjy@alibaba-inc.com, Damo Academy, Alibaba Group, Hangzhou, Zhejiang, 311121, China; Bolin Ding, bolin.ding@alibaba-inc.com, Damo Academy, Alibaba Group, Seattle, WA, 98004, United States.

1046-8188/2022/8-ART111 $15.00
https://doi.org/XXXXXXX.XXXXXXX

greatly facilitate the distribution and acquisition of information, they also bring critical privacy concerns due to unsolicited gathering users' demographical and behavioral data [67, 84]. Several regulations have been proposed recently to better protect personal data, e.g., General Data Protection Regulation (GDPR) in the European Union and the California Privacy Rights Act (CPRA) in the United States.

On the one hand, various privacy-preserving methods have been proposed to protect users' data from leakage or abuse, like federated learning [45, 52, 55, 58] and differential privacy (DP) [2, 19, 50, 66]. On the other hand, to comply with laws like GDPR and CCPA, most platforms now provide users the choice to opt-in (under GDPR) or opt-out (under CCPA) of data disclosure. Previous studies have shown that giving users control over their data closure can reduce their privacy concerns [9, 83]. However, in practice, these platforms mostly only provide a very coarse-grained option for each user, named "all or nothing" binary mechanism in this paper, i.e., disclosing all data or none at all. Usually, if the users choose to not disclose their data, they will either not be able to continue using the applications [1] or not be able to enjoy the precisely personalized services.

This raises a question, *is such an "all or nothing" binary mechanism the optimal choice?* Obviously, some privacy sensitive users might choose not to disclose their data. In this case, these users cannot enjoy the benefits of personalized services. At the same time, the platform revenues from these privacy sensitive users will decrease due to the disappointed personalized services and the platform also loses their data to train a better model. It seems that such a strict mechanism might not be a satisfying choice for both parties in the ecology. Therefore, we wonder whether there exists a better mechanism to take both the gains of the users and the revenues of the platform into account?

This paper aims to study how different privacy mechanisms affect users' decisions on information disclosure and how their decisions affect the recommendation model's performance and the platform's revenue. For this purpose, we propose a privacy aware recommendation framework under *privacy calculus theory* [15, 41]. Under this new setting, users need to calculate the trade-off between the anticipated privacy risks and the potential utilities, then *proactively control which data to disclose.* In this way, all users' dispersed privacy preferences are fully accommodated.

For service providers, they naturally want to entice users to disclose as much data as possible. For end users, they want to figure out how to enjoy the benefits of personalized services with minimal privacy risks. Formally speaking, under this privacy aware recommendation task setting, we aim to study *what will happen if the platforms give users fine-grained control over their personal data.* More specifically, we investigate questions including:

i) How do different platform mechanisms affect users' decisions in information disclosure? Is the "all or nothing" binary mechanism the best choice for the platform?

ii) How do different recommendation models affect users' decisions in information disclosure? Can a platform attract users to disclose more data by optimizing the model to provide better services?

To answer these questions, we first formulate our idea in formal settings. Following current researches in economics [44, 70], we model the privacy cost as a linear summation of the user's disclosed personal data, meaning that the user loses control over such disclosed data, which also fits a fundamental notion in privacy calculus, i.e., the control over the data. Then recommendation performance (e.g., NDCG) is employed as the potential utility from users' disclosed data. To formally define user privacy decisions, we formulate the platform mechanisms using two components, i.e., data split rule and data disclosure choice space, which define the choices a user can take. Based on

[1] Some applications will refuse to serve the users who refuse to disclose data by turning off the software when users clicks the refusal button, so as to force users to approve their data disclosure disclaimer and collect their data.

these simplified settings, we now can conduct experiments with different platform mechanisms or recommendation models to find answers to the above questions.

However, there is one big challenge for directly realizing our idea in real-world applications. Direct deployment of the proposed framework in real-world applications might seriously harm the end users' experiences and the revenues of platforms. To address this challenge, inspired by the success of simulation studies in recommender systems [26, 39, 47, 79], we propose to use simulations to study the effects of the proposed framework. Specifically, we propose a reinforcement learning method to simulate users' privacy decision making on two benchmark datasets with three representative recommendation models and three user types (i.e. different privacy sensitivity). The experimental results show that the platform mechanism with finer split granularity and more unconstrained disclosure strategy can bring better results for both end users and platforms than "all or nothing" binary mechanism adopted by most platforms. In addition to mechanism design, the results also point out that optimizing model is another option for the platform to collect more data while protecting user privacy.

Our main contributions can be summarized as following:

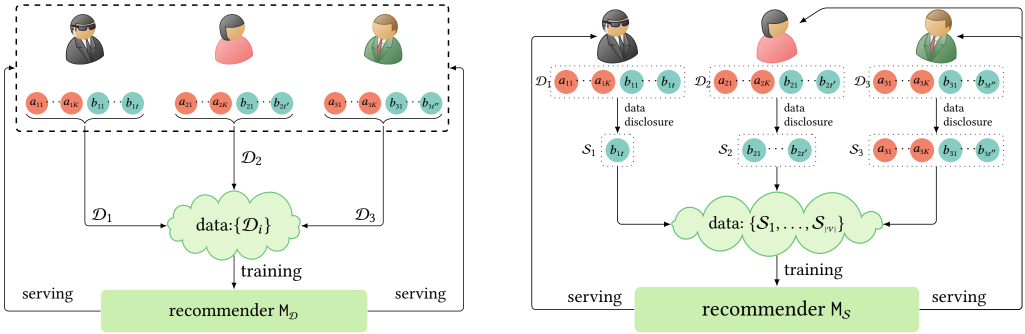
- We study an important and new problem in recommender systems and privacy protection, the effects of platform mechanisms on users' privacy decision.
- We propose a privacy aware recommendation framework that gives users control over their personal data. To the best of our knowledge, this is the first work to give users fine-grained control over implicit feedback data in recommendation.
- We formulate the process of users' privacy decision making and the platform's data disclosure mechanisms using mathematics language. Then we instantiate the platform's mechanisms with one data split rule and three data disclosure strategies that we proposed.
- We propose a reinforcement learning method to simulate users' privacy decision making. The extensive simulations are conducted on two benchmark datasets with three representative recommendation models.
- The extensive experimental results show the effectiveness of our proposed framework in protecting users' privacy. The results also shed some light on data disclosure mechanism design and model optimization.

2 FRAMEWORK FORMULATION

2.1 Overview

To study how different platform mechanisms affect users' decisions in information disclosure, we first need to incorporate the users' data disclosure decisions into the recommendation process. Thus, we propose a privacy aware recommendation framework where users can freely choose which data to disclose with the recommender system. As illustrated in Fig. 1, the critical difference between our framework and traditional recommendation is that the platform can only use the sub-data disclosed by the users. For example, the user on the left in Fig. 1b can choose to hide his sensitive demographic attributes (e.g., age, gender, and education) and only discloses the last behavior to the service provider.

To enjoy the benefits of personalized services, users need to disclose their data to the recommender system to better model them. Intuitively, more data the recommender system gets, better results the users can get. However, disclosing data to the platform will increase users' privacy concerns, e.g., data abusing [49] and privacy leakage [84]. Thus, under the privacy aware setting, users need



(a) Traditional recommender systems.

(b) Privacy aware recommender systems.

Fig. 1. Illustrative examples for two different recommender system frameworks.

to make information disclosure decisions based on the trade-off between anticipated privacy risks and potential utilities. This idea can date back to *Privacy Calculus Theory* [15, 41].

Before going into details, we first define the entire personal data $\mathcal{D}_i$ of user $i \in \mathcal{V}$'s as:

$$\mathcal{D}_i = \{\mathcal{D}_{i,a}, \mathcal{D}_{i,b}\} = \{\{a_{i1}, \dots, a_{iK}\}, \{b_{i1}, \dots, b_{it_i}\}\}, \tag{1}$$

where $\mathcal{D}_{i,a}$=$\{a_{i1}, \dots, a_{iK}\}$ denotes user i's all profile attributes, a_{ik} denotes the k-th profile attribute for user i, and K is the number of profile attributes. $\mathcal{D}_{i,b}$=$\{b_{i1}, \dots, b_{it_i}\}$ denotes user i's behaviors, b_{ij} is the j-th behavior of user i, and t_i is the last behavior timestamp. $\mathcal{V}$ is the set that includes all users in the platform.

A rational user is only willing to disclose data when she feels that she gains more from the platform than she loses in data disclosure. Formally speaking, supposing user i with whole data $\mathcal{D}_i$ currently discloses data $\mathcal{S}_i \subset \mathcal{D}_i$, now she tries to get a better recommendation results via disclosing more data $\mathcal{S}'_i \subset \mathcal{D}_i$ where $|\mathcal{S}'_i| > |\mathcal{S}_i|$, only if

$$\mathsf{U}_i(\mathcal{S}'_i) - \mathsf{U}_i(\mathcal{S}_i) > \lambda_i\big(\mathsf{C}_i(\mathcal{S}'_i) - \mathsf{C}_i(\mathcal{S}_i)\big), \tag{2}$$

where $\mathsf{U}_i(x)$ denotes the utility that user i can get from the platform with disclosed data x, function $\mathsf{C}_i(x)$ measures the privacy cost paid by the user i when she discloses the data x to the platform, and λ_i is the sensitive weight measuring how much user i cares about her privacy. Apparently, compared to privacy insensitive users (i.e., small λ_i), the platform needs to provide more performance improvements to attract privacy sensitive users (i.e., large λ_i) to disclose their data. More details can be found in Section 3.4.2.

2.1.1 ***User Objective.***

Unlike traditional task settings where users can only passively accept recommendation results (i.e., without tools to optimize their objectives), in our framework, a rational user i tends to maximize her utility $\mathsf{U}_i(\mathcal{S}_i)$ while minimize the privacy risk $\mathsf{C}_i(\mathcal{S}_i)$ by control the disclosed data $\mathcal{S}_i$. The objective function for a specific user i can be formalized as the following:

$$\mathsf{R}_i(\mathcal{S}_i) = -\lambda_i \mathsf{C}_i(\mathcal{S}_i) + \mathsf{U}_i(\mathcal{S}_i). \tag{3}$$

The linear combination for user objective function follows the initial idea from *Privacy Calculus Theory* [15, 41] where both the recommendation performances from the platform and potential privacy cost are considered. This formulation is also compatible with privacy related research in economics [18, 31, 44]. They studied the micro-foundation on a user's intrinsic and instrumental preferences from disclosing personal information. In our formulation, user's privacy cost $\mathsf{C}_i(\mathcal{S}_i)$

corresponds to intrinsic value for personal data (i.e., protecting the data from being obtained by others), while recommendation utility $\mathsf{U}_i(\mathcal{S}_i)$ corresponds to the instrumental value for personal data.

2.1.2 *Platform Objective.*

In the proposed framework, the goal of a platform is still to maximize its revenue (e.g., purchases, clicks, or watching time) by improving the users' recommendation utility (e.g., providing more accurate results). Thus, we define its objective as the summation of all users' recommendation utilities in Eq. (3), where $\mathcal{V}$ denotes all users in the platform:

$$\mathsf{R_p} = \sum_{i \in \mathcal{V}} \mathsf{U}_i(\mathcal{S}_i). \tag{4}$$

Considering the utility also depends on the recommendation model, the utility function $\mathsf{U}_i(x)$ can be further defined as:

$$\mathsf{U}_i(\mathcal{S}_i) = \mathsf{U}(\mathcal{S}_i) = \mathsf{U}'(\mathcal{S}_i, \mathsf{M}_\mathcal{S}), \tag{5}$$

where $\mathsf{M}_\mathcal{S} : \mathcal{S}_i \longrightarrow l_i$ (l_i is recommendation results) is a recommendation model trained using all users' disclosed data $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{|\mathcal{V}|}\}$ and U' represents detailed recommendation utility function. Here, without loss of generality, we assume that all users share the same utility function. We will explore the personalized utility function in the future work.

2.1.3 *Recommendation Utility Function.*

As shown in Eq. (3) and Eq. (4), the recommendation utility U occurs in the objective functions of both end users and the platform. Here, we use the users' satisfaction with the results produced by the recommendation model to measure its utility. It is worth noting that user satisfaction is still an open problem in recommender systems. Here, we simply quantify it by the user's interactions with the recommendation results, e.g., clicks, watches, and reads. Based on such feedbacks, we can calculate different quantitative metrics as the utility in our framework, e.g., hit ratio and normalized discounted cumulative gain (NDCG) [29]. In this paper, we choose NDCG as the utility function U for all users because of its widespread use [22, 33, 68].

In traditional recommendation task, the platform can optimize this objective by only optimizing the model $\mathsf{M}_\mathcal{S}$ since $\mathcal{S}_i = \mathcal{D}_i$ is a fixed, i.e., all users disclose their whole data. However, this premise is broken in our proposed framework, where the user's disclosed data $\mathcal{S}_i$ is varying. Thus, in our new framework, platforms also seek to attract users to share more data in other ways besides optimizing models, such as platform mechanism design.

2.2 Platform Mechanism

As mentioned before, the disclosed data $\mathcal{S}_i$ lives at the heart of the framework. Ideally, user i can freely choose any data $\mathcal{S}_i$ to disclose with the platform, e.g., choosing any profile attribute a or behavior data b as shown in Fig. 1b. However, in practice, such a degree of freedom is difficult to achieve for two reasons. On the one hand, from the perspective of human-computer interaction, too fine granularity of disclosure choice (e.g., single behavior) can adversely hurt user experience [82]. On the other hand, although the privacy regulations ensure users the right to determine the use of their data, they do not stipulate how the service providers implement this function.

In practice, the platform usually designs some data disclosure mechanisms to provide the end users with several convenient options. Here, we formulate the platform mechanism $\mathrm{G} =< \delta, \Pi >$ using two components, data split rule δ and disclosure choice spaces Π. The data split rule δ is regarded as a function that reorganizes the original user data $\mathcal{D}_i$ using different granularity, and Π denotes the space of all possible choices the platform provides to the user. We illustrate a toy example in Fig. 2.

2.2.1 ***Data Split Rule.*** Since user data usually consists of two different data types (as in Eq. (1)), we defined δ as:

$$\delta(x) = \{\delta_a(\mathcal{D}_{i,a}), \delta_b(\mathcal{D}_{i,b})\},$$

where δ_a and δ_b have similar forms that split the original data into several pieces according to the corresponding granularity and rules:

$$\{x_1, x_2, \ldots, x_n\} \xrightarrow[\delta_a]{\delta_b} \{x'_1, x'_2, \ldots, x'_m\},$$

where $m \leq n$ and x'_j is the candidate units for data disclosure. According to the segmentation rules, x'_j can be several consecutive data points like $\{x_1, x_2, x_3\}$ or discontinuous random data like $\{x_5, x_{22}\}$.

δ_a aims to reorganize the user's profile attributes. The common approach is to keep original granularity (i.e., user can freely disclose any subset of attributes) or take all attributes as a whole (i.e., disclose all attributes or not). Formally, it can be instantiated as:

$$\begin{aligned} \delta_a(\mathcal{D}_{i,a}) &= \{a_{i1}, \ldots, a_{iK}\} \\ \text{or} \quad \delta_a(\mathcal{D}_{i,a}) &= \{\{a_{i1}, \ldots, a_{iK}\}\}. \end{aligned}$$

Similarly, δ_b aims to transfer a user's original behavior data (e.g., thousands of clicks or more views) to few data disclosure options. For example, "percentage split" with 10% granularity divides a user's behavior sequence into 10 equal length subsequences, while "daily split" divides the user's behaviors by day. Take "percentage split" with 10% granularity as an example, it can be instantiated as:

$$\delta_b(\mathcal{D}_{i,b}) = \{\mathcal{S}_{i,b1}, \mathcal{S}_{i,b2}, \ldots, \mathcal{S}_{i,b10}\},$$

where $\mathcal{S}_{i,bj} = \{b_{i,\lfloor 0.1 t_i * (j-1) \rfloor + 1}, \ldots, b_{i,\lfloor 0.1 t_i * j \rfloor}\}$ is the j-th candidate option of behavior data for user to disclosed.

2.2.2 ***Data Disclosure Choice Space*** Π. Assuming the platform has transferred user i's original data $\mathcal{D}_i$ to $\delta(\mathcal{D}_i) = \{\mathcal{S}_{i,a1}, \ldots, \mathcal{S}_{i,an}, \mathcal{S}_{i,b1}, \ldots, \mathcal{S}_{i,bm}\}$, the platform can define data disclosure choice space Π on these $m + n$ candidates as:

$$\begin{aligned} &\Pi = \{\Pi_1, \Pi_2, \ldots, \Pi_N\}, \\ &\Pi_j \sim [o_1, \cdots, o_k, \cdots, o_{n+m}], \quad o_k \in \{0, 1\}, \end{aligned}$$

where $o_k = 1$ denotes disclosing the k-th data in $\delta(\mathcal{D}_i)$, while $o_k = 0$ means not; Π_j is $j-$th data disclosure option that users can take; N is the number of possible choices the platform provides to users. For example, a full 0 vector $\Pi_j = [0, 0, \ldots, 0]$ denotes that users can choose it to do not disclose any data. More detailed instantiations can be found in Section 3.3.

2.2.3 ***Platform Mechanism Design.*** With the mechanism G =< δ, Π >, we can formally define the disclosed data $\mathcal{S}_i$ from user i. Assuming user i's original data $\mathcal{D}_i$ has been spited into candidates $\delta(\mathcal{D}_i) = \{\mathcal{S}_{i,1}, \ldots, \mathcal{S}_{i,m}\}^2$. Then, $\mathcal{S}_i$ can be defined as the union of candidates in $\delta(\mathcal{D}_i)$ selected by a specific choice $\alpha_i = \Pi_j \in \Pi$:

$$\mathcal{S}_i = \alpha_i \otimes \delta(\mathcal{D}_i) = \left\{ \bigcup_{\substack{o_k = 1 \\ o_k \in \Pi_j}} \mathcal{S}_{i,k} \right\}, \tag{6}$$

where δ is the platform data split rule and action α_i is sampled from user i's privacy disclosure policy π_i, which decides the data to be disclosed. The operator $\otimes$ denotes the aggregation of the selected spilt data based on his choice α_i. Fig. 2 illustrates a tiny example of the data disclosure process (i.e., generation process of $\mathcal{S}_i$) of a user with three profile attributes and four behaviors.

[2] Here, we simplify the subscripts for easy description

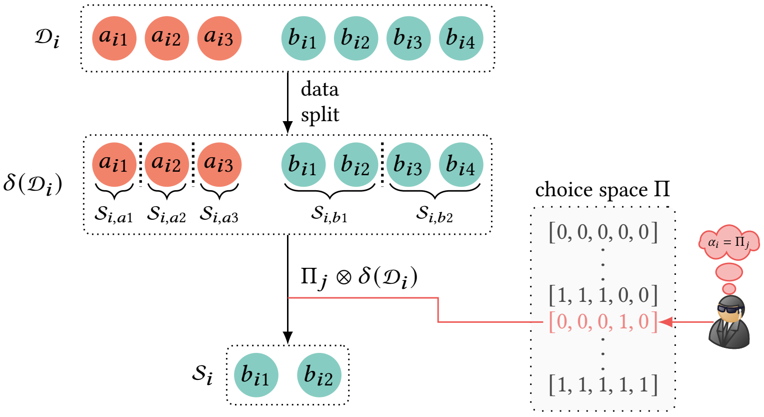

$\mathcal{D}_i$
a_{i1} a_{i2} a_{i3} b_{i1} b_{i2} b_{i3} b_{i4}
data split
$\delta(\mathcal{D}_i)$
$\mathcal{S}_{i,a1}$ $\mathcal{S}_{i,a2}$ $\mathcal{S}_{i,a3}$ $\mathcal{S}_{i,b1}$ $\mathcal{S}_{i,b2}$
choice space Π
[0, 0, 0, 0, 0]
[1, 1, 1, 0, 0]
[0, 0, 0, 1, 0]
[1, 1, 1, 1, 1]
$\alpha_i = \Pi_j$
$\Pi_j \otimes \delta(\mathcal{D}_i)$
$\mathcal{S}_i$
b_{i1} b_{i2}

Fig. 2. An illustrative example for platform mechanism. The platform firstly split the user's data $\mathcal{D}_i$ (three profile attributes and four behaviors) using the rule δ (keeping independent for attributes; percentage split with 50% granularity for behaviors). Then it provides choices Π for the user to choose to produce the final disclosed data $\mathcal{S}_i$.

With formal definition of $\mathcal{S}_i$, we can re-write the platform utility R_p in Eq. (4) using the platform mechanism G and model $\mathsf{M}_\mathcal{S}$ as below,

$$\mathsf{R}_\mathrm{p}|_{\mathrm{G}=<\delta,\Pi>} = \sum_{i \in \mathcal{V}} \mathsf{U}'_i(\mathcal{S}_i, \mathsf{M}_\mathcal{S}) = \sum_{i \in \mathcal{V}} \mathsf{U}'_i(\alpha_i \otimes \delta(\mathcal{D}_i), \mathsf{M}_\mathcal{S}). \tag{7}$$

One may figure out some possible optimal solutions towards the platform's best mechanisms. However, the optimal platform mechanism design is another complex topic, usually considered from the view of game theory, and is out of scopes of this work. Here, we take the first step, studying the data disclosure decision of users and platform revenues under several common mechanisms.

2.3 Relationship with Privacy Preservation

In this subsection, we will discuss the relationship between our work and privacy preservation, and clarify the position of our paper.

First, as claimed in the introduction, our work does not aim to propose a new method to directly protect the users' privacy data from privacy attacks. The main motivation of this paper is to study the effectiveness of existing privacy mechanisms deployed in the platforms and further explore how different privacy mechanisms affect users' privacy decisions. For this purpose, our proposed framework is model-agnostic. The model $\mathsf{M}_\mathcal{S}$ used here can be an ordinary recommendation model (e.g., NCF [22] or GRU4Rec [25]) or privacy-preserving recommendation models [8]. For deploying a privacy-preserving model in our framework, we only need to modify the privacy cost function (C_i), taking into account the influence of protecting privacy, i.e., less privacy cost than a normal model when disclosing the same amount of data. In this paper, for the convenience of analysis and considering that privacy-preserving technologies are not widely used in real-world applications, we only studied the proposed framework in ordinary recommendation models. We leave the exploration of the impact of privacy-preserving technologies on users' privacy cost functions for future work.

Second, our proposed framework provides users with the power to proactively trade off their privacy cost and recommendation utility. From the final result point of view, users can optimize their data disclosure decision to discard those data that are not helpful for their recommendation results. In this way, the proposed framework gives users the tool to achieve data minimization [3, 53] by themselves, rather than waiting for the platform to implement the data minimization algorithms. From this perspective, our proposed framework can be seen as implicitly protecting user privacy. Even if a privacy attack occurs, only part of users' disclosed data will be leaked. Besides, giving users control over the recommendation process has also been found be effective in reducing their privacy concerns [9, 83].

3 SIMULATION

3.1 motivation

The most efficient way to figure out the answers to the questions we posed in the introduction is to deploy the proposed framework on a real-world platform and analyze how users adopt different and complex privacy policies to optimize their rewards. However, direct deployment of these strategies and investments is currently impractical due to the following reasons.

Firstly, the most important reason is that such an online experiment may lead to the decline of the recommendation performances as well as the user experience, which harms the platform's revenue. In the real world, nearly all the companies determine their platform mechanism driven by interest, and the revenues of the platforms are highly correlated with the recommendation performances. Therefore, it's nearly impossible to persuade any platform to directly deploy proposed strategies and mechanism online without other benefits.

Secondly, the experiments are built upon several simplifications, mentioned in Section 3.2, which poses challenges towards recommendation model training process. For example, we assume when a user adjusts his data disclosure policy, the recommendation system will forget his un-disclosed data. To facilitate such challenges, model unlearning or other privacy-preserving technologies are imposed. However, in real-world applications, very few the e-commercial platforms have deployed these privacy-preserving technologies during the deep recommendation model training and evaluation processing. As a result, we may still fail to guarantee the assumptions and simulation methodology becomes a substitution.

In summary, inspired by the success of simulation study on dynamic interactive problems in real-world applications [26, 39, 47, 79], we employ the simulation to study the effects of the proposed framework and the possible game between users and the platform.

3.2 Simplified Assumptions

To simplify the simulation process for easier analysis, we make some necessary assumptions to simplify the problem.

Assumption 1 (Static Assumption). *User i optimizes her/his policy on the fixed data $\mathcal{D}_i$ which is not affected by user policy π_i.*

Here static means the user data $\mathcal{D}_i$ is fixed during the simulation, but the disclosed data $\mathcal{S}_i$ produced by different user policies is dynamic. It is also the most common setting for recommendation task in research papers [22, 25, 33, 61, 68]. In the simulation, we train the recommendation system $\mathtt{M}_{\mathcal{S}}$ on the collected dynamic data $\mathcal{S}$ and validate the recommendation efficiency on a fixed test set. In real-world applications, the data $\mathcal{D}_i$, which contains the behavior data from the interaction with the recommender $\mathtt{M}_{\mathcal{S}}$, is also dynamically changing with the user's policy π_i. It is beyond the scope of this paper and we leave it as the future work.

Assumption 2 (Immediate Assumption). *The recommendation model $\mathtt{M}_{\mathcal{S}}$ can only use the data $\mathcal{S}_i$ currently disclosed by each user i.*

The motivation of this assumption is that an untrusted platform can leverage user *i*' all data $\mathcal{D}_i$ if it can use the data disclosed in previous actions. Without this constraint, the privacy right discussed in this paper is meaningless. To achieve this, the platform can retrain the model from scratch with new data $\mathcal{S}_i'$ or quick unlearn the data in $\mathcal{S}_i$ then finetune with data $\mathcal{S}_i'$ [4, 5, 7].

However, the Assumption 2 also raises a new challenge that the asynchronous changes of user policy bring intractable computation costs for the platform since each time the user changes the disclosed data, the platform needs to update the model. Here, we make an assumption for simplifying the simulation, assuming all users realize that the platform will cyclically (e.g., once a day) collect their privacy decisions and update recommender systems.

Assumption 3 (Cyclical Assumption). *Platform cyclically collects user privacy choices, and then the platform updates the model using all newly disclosed data.*

In summary, for easy analysis in simulations, we introduce these assumptions to ignore the time and dynamic effects in this feedback system, just like the traditional recommendation task formulation.

3.3 Platform Mechanism Simulation

In order to validate the effect of the platform mechanism, we adopt several mechanisms during simulation. For easy comparison, we utilize one mechanism at each experiment.

3.3.1 ***Data Split Rule.*** In our simulation, we do not split the profile attribution and the user can determine whether to share all of their attributes. For behavior data, we apply "percentage split" as δ_b with different split granularity p (e.g., 1/3) to split the behavior sequence into $1/p$ parts. One obvious advantage of "percentage split" is that it can normalize the size of user action space and decrease the inconvenience of the interaction between the user and the platform.

3.3.2 ***Data Disclosure Strategy.*** As the platforms have certain flexibility to implement different data disclosure strategies, we discuss three representative disclosure strategies used in our study for behavior data in this subsection. These strategies determine the data disclosure action space Π the user can choose. For profile attributes, we found that all users tend not to disclose them in the experiments since these features are negligible for improving recommendation utility in the presence of behavior data. Similar result that user profile features contribute very marginal to the recommendation results in the case of strong user behavior modeling on public benchmark datasets has also been reported in other works [33, 68]. Thus, in the following study, we mainly focus on modeling only behavior data.

The "*separate*" rule gives the users the control to freely disclose any split personal data. For this rule, the size of user i's the action space is exponentially expended on the size of the spilt data set $|\delta_b(\mathcal{D}_{i,b})|$, denoted as $2^{|\delta_b(\mathcal{D}_{i,b})|}$. However, too many choices might make it difficult for users to make better data disclosure decisions.

Another data disclosure strategy named "*oldest continuous*" provides users the choices to disclose continuous behavior data from the beginning time, such as selecting "the oldest 33% data". In this strategy, to disclose newer behavior data $\mathcal{S}_{i,bj}$, users must also disclose all behavior data before it. Take an already split data $\delta_b(\mathcal{D}_{i,b}) = \{\mathcal{S}_{i,b1},\mathcal{S}_{i,b2},\mathcal{S}_{i,b3}\}$ as an example, the action space provided by oldest continuous strategy is $\Pi = \{[0,0,0],[1,0,0],[1,1,0],[1,1,1]\}$, and its corresponding disclosed data is $\{\varnothing, \{\mathcal{S}_{i,b1}\}, \{\mathcal{S}_{i,b1}, \mathcal{S}_{i,b2}\}, \{\mathcal{S}_{i,b1},\mathcal{S}_{i,b2}, \mathcal{S}_{i,b3}\}\}$. "*Latest continuous*" mechanism is similar to "oldest continuous", with the only difference in the opposite direction. The size of these two mechanisms' action spaces is $|\delta_b(\mathcal{D}_{i,b})|$.

3.4 User Policy Simulation

In this subsection, we introduce the simulation of user policy in our proposed framework. As defined in Eq. (6), the disclosed data $\mathcal{S}_i$ is result of the platform mechanism G and user's disclosure policy π_i. Meanwhile, in Eq. (5), the recommendation utility $\mathsf{U}_i(\mathcal{S}_i) = \mathsf{U}'(\mathcal{S}_i, \mathsf{M}_{\mathcal{S}})$ is also determined by the recommendation model $\mathsf{M}_{\mathcal{S}}$, which is built upon the all users' disclosed data $\mathcal{S}$. The reward of user i may be varied even when i keeps the disclosed data $\mathcal{S}_i$ unchanged since other users might change their disclosed data and the recommender system is changed. Thus, the expectation rewards are considered rather than immediate value defined in Eq. (3) and we assume all the users are rational and seek for the optimal privacy disclosure action α_i^* to the optimal expected reward $E[\mathsf{R}_i|\alpha_i]$ as his policy, i.e.,

$$\begin{aligned}\alpha_i^* &= \mathop{\arg\max}_{\alpha_i \in \Pi} E[\mathsf{R}_i|\alpha_i] = \mathop{\arg\max}_{\mathcal{S}_i \in [\Pi \otimes \delta(\mathcal{D}_i)]} E[\mathsf{R}_i(\mathcal{S}_i)] \\ &= \mathop{\arg\max}_{\alpha_i \in \Pi} E\Big[-\lambda_i \mathsf{C}_i\big(\alpha_i \otimes \delta(\mathcal{D}_i)\big) + \mathsf{U}_i\big(\alpha_i \otimes \delta(\mathcal{D}_i)\big)\Big].\end{aligned} \tag{8}$$

As mentioned before, recommendation utility U_i has been discussed in Section 2.1.2. To study this objective, we need to define the privacy cost function C_i and sensitive weight λ_i.

3.4.1 *Privacy Cost Function*.

We simulate every user with the same cost function C and leave the diversity of user privacy sensitivity to the parameter λ_i. Following current experiment specifications in the economics literature [44, 70], we model the privacy cost function as a linear summation[3] of disclosed personal data.

According to the comprehensive survey on privacy value definition [11], people will measure the value of their privacy into the intrinsic value of privacy and the instrumental value of privacy. The intrinsic loss indicates the sake of protecting their intrinsic private data, which measures the valuation on the intrinsic properties such as the education or the income levels. In this work, we denote the intrinsic loss towards the privacy cost on amount of the sharing user profile attributes. The instrumental value of privacy indicates how the transaction efficiency would be affected by sharing user data, especially the data generated in the applications. In this work we denote the privacy cost towards the percentage of shard user historical behavior data. Therefore, the privacy cost function is described below,

$$\mathsf{C}_i(\mathcal{S}_i) = \beta_i * |\mathcal{S}_{i,a}| + \frac{|\mathcal{S}_{i,b}|}{|\mathcal{D}_{i,b}|} \tag{9}$$

where the first term indicates the intrinsic loss and the second term indicates the instrumental loss. If user does not tend to disclose profile attribute, such privacy cost function can be simplified to the following format with the instrumental value alone. As mentioned in Section 3.3, user tends not to disclose profile attributes $\mathcal{D}_{i,a}$ due to no gains in our experiments, so we only consider behavior data here, i.e.,

$$\mathsf{C}_i(\mathcal{S}_i) = \mathsf{C}(\mathcal{S}_i) = \frac{|\mathcal{S}_{i,b}|}{|\mathcal{D}_{i,b}|}, \tag{10}$$

where the $|x|$ is the number of elements in x. Here, the percentage based measurement regards different amount of users' data equally.

This reduced form specification is not unrealistic as it captures the substitution effect among personal data and incorporates the idea of constant marginal privacy cost. One might argue for a higher order functional to capture richer implications. However, there is little experimental evidence that the higher order form for privacy cost exists and how the functional form looks like.

[3] See the Eq. 2 in [44] and the dis-utility from disclosure in the econometric specification session in [70].

3.4.2 ***Privacy Sensitive Weight.*** For user i who disclosed all her/his data (i.e., $\mathcal{S}_i = \mathcal{D}_i$), her/his privacy cost compared to not sharing any data (i.e., $\mathcal{S}_i = \emptyset$) is

$$\mathsf{C}(\mathcal{D}_i) - \mathsf{C}(\emptyset). \tag{11}$$

Meanwhile, her/his anticipated recommendation utility compared to not sharing any data is:

$$\mathsf{U}(\mathcal{D}_i) - \mathsf{U}(\emptyset). \tag{12}$$

We assume all users have accessed to the recommendation utility $\mathsf{U}(\mathcal{D}_i) = \mathsf{U}'(\mathcal{D}_i, \mathsf{M}_{\mathcal{D}})$ computed on all the data $\mathcal{D}_i$ and the recommendation utility without their data $\mathsf{U}(\emptyset)$ before they can take data disclosing actions, which can be regard as a prior knowledge, like the experiences before the platform adopted our framework. With Eq. (11) and Eq. (12), we define the privacy sensitive weight λ_i as:

$$\lambda_i = w_i * \frac{\mathsf{U}(\mathcal{D}_i) - \mathsf{U}(\emptyset)}{\mathsf{C}(\mathcal{D}_i) - \mathsf{C}(\emptyset)}, \tag{13}$$

where w_i indicates the diversity of user types towards privacy sensitivity. The users with $w_i > 1$ is privacy sensitive users, as they will not be willing to disclose the corresponding data $\mathcal{D}_i$ if they only get $\mathsf{U}(\mathcal{D}_i)$ as before. While users with $w_i < 1$ are just the opposite. Therefore, the user's privacy sensitive weight is pre-computed, and the $\mathsf{U}(\mathcal{D}_i)$ can be regarded as the benchmark expectation of the platform. The formulation of the privacy sensitive weight λ_i also meets the idea from [44], where the heterogeneity from users' social demographic variety should also be explicitly characterized.

3.4.3 ***Simulation Algorithm.*** As users behave rationally to find the optimal strategy with a trade-off of exploration and exploitation, it just meets the idea of the reinforcement learning algorithm. Therefore, we model each user as a unique agent and apply a multi-agent reinforcement learning method to simulate user possible policy adaptation. The recommender system is regarded as the environment to provide feedback, which is built upon the disclosed user data. All agents' policies are optimized simultaneously by determining their actions, i.e., the disclosed data $\mathcal{S}^t$ at simulation epoch t, which is used to train the recommendation model $\mathsf{M}_{\mathcal{S}^t}$. As mentioned before, users tend to find an optimal action over possible action space Π to maximize his expected reward, which is determined by all agents in this dynamic MARL environment.

We assume each user (agent) realizes this situation that the immediate reward is the result of all agents, but no communication or observation among agents is permitted. Then, each agent is concerned about her/his own utility and regards the environment as a dynamic system that is partially correlated to herself/himself. Now, it is simplified to a Multi-Armed Bandit problem [34].

However, the challenge of the exploration and explication problem also exists in our simulation. To address it, we adopt a simple but efficient method, Epsilon Greedy [69] algorithm, to simulate user's policy π_i as following,

$$\alpha_i^{t+1} = \begin{cases} \alpha_i \sim \mathsf{P}_i^t, & \text{with possibility } \epsilon \\ \arg\max_{\alpha_i} Q_i^t(\alpha_i), & \text{with possibility } 1-\epsilon \end{cases} \tag{14}$$

where $Q_i^t(\alpha)$ is the user i's estimation value at simulation epoch t on action α, and P_i^t denotes a random sample policy. To conduct an efficient policy exploration, we sample a less explored action with a higher possibility as following,

$$\mathsf{P}_i^t(\alpha) = \frac{1/(N_i^{t-1}(\alpha)+1)}{\sum_{x\in\Pi} 1/(N_i^{t-1}(x)+1)}, \tag{15}$$

where $N_i^{t-1}(\alpha)$ represents the total number of action α was taken by user i from start to the last simulation epoch $t-1$. In convenience, we adopt the approximated expected estimation results and update it with the residual between the estimation $Q_i^{t-1}(\alpha_i^{t-1})$ and immediate reward R_i^{t-1} when she/he takes action α_i^{t-1} as following.

$$Q_i^t(\alpha)=\begin{cases} Q_i^{t-1}(\alpha), & \text{if } \alpha_i^{t-1}\neq\alpha \\ Q_i^{t-1}(\alpha)+\frac{1}{N_i^t(\alpha)}\left(\mathsf{R}_i^{t-1}(\alpha\otimes\delta(\mathcal{D}_i))-Q_i^{t-1}(\alpha)\right), & \text{if } \alpha_i^{t-1}=\alpha \end{cases}$$

where R_i^{t-1} is user i-th immediate objective at simulation epoch $t-1$, computed by Eq. (3). $Q_i^0(\alpha)$ is the user i's initial expected reward if she/he takes action α. which is initialized to 0 as users have no prior about their behaviors on the new dynamic environment.

In our simulation, we set initial $\epsilon = 0.5$ for all agents and decay a half during the MARL training processing. The detailed decay epoch is co-related to the size of possible action space Π. Here, we define it as $\epsilon = 0.5^{t/(3*|\Pi|)}$, where t is the epoch during the reinforcement learning training processing.

3.5 Discussion

To figure out how the platform mechanism affects users' behavior, we turn to the simulation built upon several simplified assumptions. One fundamental assumption is the hypothesis of rational man, where users will seek their optimal policies to maximize their objectives. However, in the real-world scenarios, human behaviors are also affected by psychological factors, which should also be modeled in future work. One detailed example is that some users may feel exhausted digging out all the potential privacy choices with the provided platform mechanism. In our simulation, we assume there remains no mental cost when a user adjusts his policy. However, in the reality, some users may refuse to change their policy frequently, especially in complex user interaction applications. For such situation, a convenient user interface (UI) could be a potential solution to mitigate users' fatigues. Another important factor is that users may adjust their trust level towards the platform during their exploration. One detailed example is that if the platform or even the recommender system [86] is easy to be attacked or the platform will abuse their disclosed data to other applications, they may re-consider their privacy sensitivity. Though some works have discussed the utilization of trusted platform or the privacy-preserved recommendation model, the possible effects on user psychological factors might be tackled by a dynamic modeling on the user privacy sensitive weights, which is out of the scope of this work. We simplify the influences of the psychological factors in this work and leave the exploration of psychological effects in mechanism designs and UI designs for future works.

4 EXPERIMENTS

4.1 Research Question

- **RQ1**: How do different platform mechanisms affect the recommendation performance and the data disclosure decisions of users with different privacy sensitivity?
- **RQ2**: What is the role of recommendation model in this framework? Can a more accurate model attract users to disclose more data?
- **RQ3**: How does user population composition affects the user behavior in this framework?

4.2 Experiment Setup

4.2.1 Dataset. We conduct our experiments on two real-world representative datasets which vary in domains and sparsity:

Table 1. Statistical details of the evaluation datasets.

Dataset	#User	#Item	#Interaction	Density
ML-100k	637	1278	90,554	11.12%
Yelp	8338	35,476	760,635	0.26%

- **Movielens-100k**[4] (ML-100k) [20]: This is a popular benchmark dataset for evaluating recommendation algorithms. Here, we use the version that includes 100k user ratings.
- **Yelp**[5]. This is a popular and continuously updated dataset for business recommendation. The version we download for the experiments in this paper is Feb. 16 2021, i.e., all review records are written before Feb. 16 2021.

Since we focus on recommendation based on implicit feedback, we follow the common practice to convert the numeric rating or a review into implicit feedback of 1 (i.e., indicating the user interacted with the item). After that, we build the behavior sequence for each user by grouping and sorting their behaviors according to the timestamps. To properly simulate the information disclosure decision making process, we filter out the user with less than 40 interactions and items with less than 5 interactions. For efficiency reasons, we further subsample the users in Yelp, resulting in a dataset with 8338 users. The statistics of the processed datasets are summarized in Table 1.

4.2.2 Simulation Setup. For each user, we hold out the last item of the behavior data as the test data to compute recommendation utility [22, 33, 68]. The rest of the behavior data is used for training simulation, treating the last interaction data in disclosed data as validation data and the remaining disclosed data for training data.

For recommendation utility evaluation, we adopt the widely used *leave-one-out evaluation* [22, 33, 68] protocol with NDCG@100 computed on the whole item set as the metric. In particular, for Yelp, we compute the sampled metric with 1000 negative samples since the large item candidate set makes the results on Yelp are too small to simulate stably. These sampled results are consistent with the scores on the whole candidate set [40].

For privacy risk function, we use disclosed data percentage measurement according to Section 3.4.1, which is weighted by the sensitive weight λ_i defined in Eq. (13). To study the data disclosure decision making for users with different privacy sensitivity, we randomly divide users into three groups (each with $1/3$ users) with different privacy sensitive levels by adjusting the w_i,

- $w_i = 0$: non-sensitive user who does not care privacy at all.
- $w_i = 1$: normal user who weights privacy risk and recommendation results in a relatively normal way.
- $w_i = 10$: sensitive user who concerns more about privacy than recommendation utility.

To acquire the sensitive weight λ_i, the benchmark recommendation result $\mathsf{U}(\mathcal{D}_i)$ is computed based on GRU model with the whole dataset. The assumption here is that the non-privacy aware framework that the user used before was based on GRU4Rec. As this work focus on the effects on the user disclose choices, we also assume that there exists no privacy leak among data transmission period and users only access to their private recommendation results and the platform recommendation models are well protected with privacy guarantee. It is worth noting that we group users into three categories here just for the convenience of analyses and discussions in subsequent experiments. In fact, according to Eq. (13), users in each category still have different privacy sensitivity λ_i.

[4] https://grouplens.org/datasets/movielens/100k/

[5] https://www.yelp.com/dataset

Table 2. The averaged recommendation results (%) on three different recommendation models when all users disclose history behaviors with or without profile attributes.

Dataset	Model	with profile	without profile
ML-100k	NCF	12.69	12.65
	GRU	19.89	19.87
	BiSA	19.90	20.06

In the RL training, we model each user as an agent following the setup, and train 400 epochs with Epsilon Greedy algorithm. In each simulation epoch, the recommendation model is trained from scratch as discussed in Assumption 2, i.e., the platform can only use the data that the user disclosed in the current simulation epoch. The simulation epoch is enlarged to 3000 epochs in "separate" data disclosure strategy with $p = 1/8$ due to the slow convergence.

4.2.3 Recommendation Model. To study the role of different models in users' data disclosure decision making, we conduct the simulations on different models, including two state of the art sequential recommendation models and one CF model.

- **GRU4Rec** [25]: It uses GRU with ranking based loss to model user sequences for session based recommendation.
- **BiSA** (**Bi**directional **S**elf-**A**ttention) [33, 68]: It uses a self-attention architecture to capture users' sequential behaviors and achieves state-of-the-art performance on sequential recommendation. It usually can obtain better results than GRU4Rec with more powerful architecture.
- **NCF** [22]: NCF models user–item interactions with a multi layer perceptron. It is included as a weaker model since it is not designed to capture the sequential information in user behaviors.

We implement these models using PyTorch[6]. The hyper-parameters are carefully tuned using a grid search to achieve optimal performances. After tuning, the embedding size and hidden size is set to 128 for all the models, the dropout ratio is set to 0.2, the learning rate equals to 1e-3 for the models except BiSA with a learning rate 3e-4, and the number of negative samples is set to 16. All models are trained with adam optimizer [35] with early stop.

4.3 Study 0: Impact of User Profile Attribution

Before conduct our experiment, we validate whether user feature perform a essential part of our latter experiments with Movielens-100k for example. In Movielen, we include all provided 4 user profile attributes: sexual, age, job, and zip code. We firstly conduct experiments on the situation where all users submit their historical data with and without user profile with three mentioned models in the following tables From the Table 2 we observe that the user profile attributes introduce minor recommendation improvements on all three models if the recommendation systems have already absorbed enough user information from their historical data. In other words, when the platforms have already collected enough information for the recommender systems, users have minor incentives to disclose their user profiles attributes. we secondly conduct experiments to simulate the circumstances where all users are able to determine whether to disclose their history behavior data along with their profile attributes. In the following experiments, the guarantee

[6] The source code will be released after the review phase.

Table 3. The percentage(%) of the users who are willing to disclose behavior data or the profile data under different split granularity after convergence.

Data Composition	p	Disclose behavior percentage	Disclose profile percentage
With Profile	1	42.39	6.00
	1/2	52.90	6.28
	1/4	60.91	6.28
Without Profile	1	43.80	-
	1/2	52.74	-
	1/4	56.67	-

that the recommender system collect sufficient user information may not hold, where we conduct simulations with the "continuous" data spilt rule on GRU models under 3 different granularity. In simplify, we set the $\beta_i = 1/|\mathcal{D}_{i,b}|$ according the Eq. (9). As we import the user profile attribution, the action spaces of agents are enlarged, which impose the convergence challenges. Therefore, we restrict the users whose disclosure decision have at least 0.01 improvements on their utility than not to disclose. The converged results are displayed on the Table 3 where we record the users who are willing to disclose their historical behavior data or their profile attributes. From Table 3, we can conduct two conclusions: First, barely users are stick to disclose their profile attribution among different data split granularity; Second, the user disclosure policies towards their historical data are similar no matter their profile attribution is introduced or not. The converged number of the users who are willing to disclose their historical are similar from Table 3. Regarding the slight influences on the user profile attributes, the introduce of the user profile attributes will directly enlarge the user possible action space, which impose more challenge on the multi-agent reinforcement learning processing and may require much more training cost.

As a results, we only conduct experiments on user historical data and assume all users are refuse to disclose their user profile attribution on the latter experiments.

4.4 Study 1: Impact of Platform Mechanism

We firstly conduct experiments on platform mechanisms specifying various data split granularity and data disclosure strategies with a widely-used sequential recommendation model GRU4Rec. We begin by answering which mechanism is preferred by users with different privacy sensitivity (types, hereafter).

4.4.1 Split Granularity p. We first validate how the split granularity affects users on the three aforementioned data disclosure strategies. The recommendation results and the data disclosure percentage on different user types are reported in Table 4 where all results are averaged on the last 20 epochs after convergence. From the results, it can be observed that:

(1) Comparing the NDCG performances among different settings, we can derive a negative answer for the question in the introduction considering "all or nothing" binary choice (i.e., p=1) performs worst on all disclosure strategies for all datasets. Even looking at the detailed results for user groups with different privacy sensitivity, p=1 still performs very poorly, if not the worst.

(2) Comparing the results within different user groups, a prominent and expected result is that users who care about privacy are only willing to disclose very little data, especially

Table 4. Results on different data sets with different platform mechanisms. All the results are averaged on the last 20 epochs. “dis.%” denotes average percentage of disclosed data , “NDCG” means NDCG@100 (%).

		ML-100k							
strategy	p	non-sensitive		normal		sensitive		all	
		NDCG	dis.%	NDCG	dis.%	NDCG	dis.%	NDCG	dis.%
all	—	21.74	100	19.45	100	17.46	100	19.45	100
latest continuous	1	17.07	100	10.07	21.80	6.72	11.75	11.30	43.37
	1/2	17.54	100	16.71	25.11	7.38	8.88	13.89	43.57
	1/4	**18.19**	100	18.54	18.39	6.99	5.75	14.58	40.19
	1/8	17.31	100	19.04	17.83	8.23	7.08	14.87	40.43
	1/16	17.65	100	19.26	16.22	12.01	8.82	16.31	40.45
oldest continuous	1	17.07	100	10.07	21.80	6.72	11.75	11.30	43.37
	1/2	17.40	100	14.76	25.99	8.07	11.98	13.41	44.91
	1/4	17.48	100	16.10	**28.67**	8.63	12.96	14.07	46.17
	1/8	17.44	100	18.08	22.63	10.35	15.41	15.28	44.78
	1/16	17.55	100	21.55	26.85	12.88	**17.03**	17.33	**46.89**
separate	1	17.07	100	10.07	21.80	6.72	11.75	11.30	43.37
	1/2	17.91	100	19.35	28.57	8.06	11.05	15.11	45.50
	1/4	17.38	100	22.26	23.88	9.44	11.85	16.36	44.12
	1/8	18.08	100	**27.63**	25.65	**14.93**	17.00	**20.21**	46.46
		Yelp							
latest continuous	1	26.59	100	11.36	32.58	3.13	6.11	13.70	46.70
	1/2	**27.23**	100	24.38	41.45	3.48	5.49	18.37	49.43
	1/4	26.85	100	26.72	27.25	3.93	5.07	19.17	44.60
	1/8	26.28	100	27.00	19.49	7.44	5.49	20.24	42.18
	1/16	26.09	100	25.82	17.22	**15.62**	6.08	22.33	41.10
oldest continuous	1	26.59	100	11.36	32.58	3.13	6.11	13.70	46.70
	1/2	26.47	100	19.34	37.27	3.15	5.84	16.32	48.16
	1/4	26.16	100	22.25	30.91	3.60	5.42	17.34	45.92
	1/8	26.02	100	23.76	26.31	5.49	5.64	18.42	44.49
	1/16	25.99	100	24.77	24.68	11.06	6.46	20.61	44.21
separate	1	26.59	100	11.36	32.58	3.13	6.11	13.70	46.70
	1/2	27.00	100	26.86	**42.91**	3.79	4.71	19.22	**49.65**
	1/4	26.48	100	**29.97**	27.73	4.56	4.64	20.34	44.61
	1/8	26.60	100	29.14	27.97	13.97	**15.71**	**23.23**	48.34

privacy sensitive users. Besides, normal users can obtain comparable recommendation results (even better on ML-100k) to non-sensitive users with much less data. On the one hand, this indicates that our proposed framework can effectively protect users’ privacy. On the other hand, proactively controlling the disclosed data also allows users to improve their recommendation results by themselves.

Table 5. The percentage (%) of users who disclosed data, which is averaged on the last 20 epochs.

Dataset	Strategy	#p=1	#p=1/2	#p=1/4	#p=1/8	#p=1/16
ML-100k	latest	43.80	52.74	56.67	61.02	70.63
	oldest	43.80	50.39	51.81	59.65	64.65
	separate	43.80	56.54	66.27	79.48	-
Yelp	latest	45.90	62.03	66.34	70.11	79.64
	oldest	45.90	56.73	61.37	64.94	73.33
	separate	45.90	63.68	68.26	80.82	-

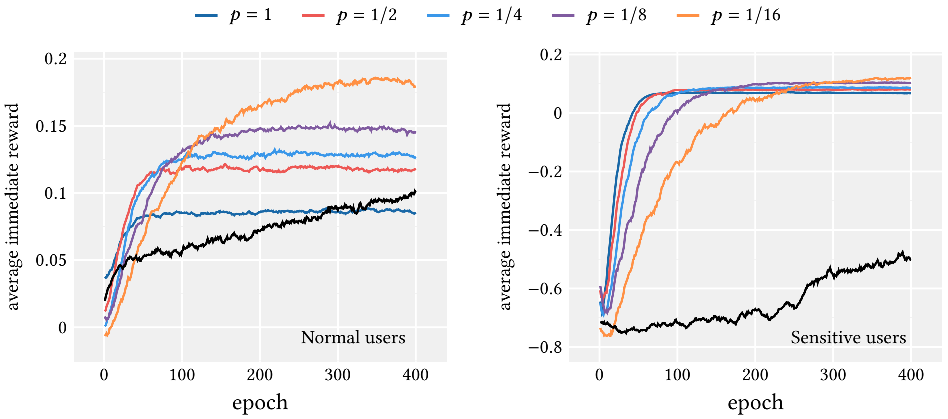


Fig. 3. Simulation process using “oldest continuous” strategy with different granularity p on ML-100k. The results are smoothed using exponential moving average with smoothing factor 0.9 for clearer visualization. The black line denotes the “separate” with p=1/8 on ML-100k truncated at 400 epochs.

(3) For platform, finer split granularity can usually bring better performances in all three mechanisms for all datasets. Unexpectedly, these superior performances are not always obtained through more disclosed data. For example, under “latest continuous” rule, the overall recommendation performances for p=1/16 (16.31% on ML-100k) are much better than p=1/2 (13.89% on ML-100k) with less training data (40.45% vs. 43.57% on ML-100k).

To figure out the reason for this phenomenon, we analyzed the distribution of user’ data disclosure. We reported the percentage of users who disclosed data in Table 5. The results show that more users turn to disclose data since finer granularity allows users to disclose a small amount of data for certain recommendation utilities. Conversely, more users suffer from poor recommendations as they refuse to disclose data under coarse-grained granularity.

4.4.2 Data Disclosure Strategy. We study how data disclosure strategy affects users’ decisions using three strategies with different degrees of freedom. The results are also reported in Table 4.

It is easy to see that the flexible “separate” strategy is superior to other mechanisms within the same granularity. The “separate” strategy achieves better overall recommendation results with similar or even less disclosed data. One possible reason is that it enables users to freely disclose the data that benefits their recommendations. In this way, users will discard those data that are

Table 6. Results on different recommendation models with different platform mechanism. “dis.%” denotes average user data disclosure percentage, “NDCG” means NDCG@100 (%). All the results are averaged on the last 20 epochs.

		ML-100k							
strategy	model	non-sensitive		normal		sensitive		all	
		NDCG	dis.%	NDCG	dis.%	NDCG	dis.%	NDCG	dis.%
latest continuous p=1/16	NCF	10.73	100	15.69	13.52	9.87	9.78	12.10	39.82
	GRU4Rec	17.65	100	19.26	16.22	12.01	8.82	16.31	40.45
	BiSA	18.04	100	21.88	15.61	**14.55**	12.80	18.16	40.44
oldest continuous p=1/16	NCF	11.60	100	14.08	14.88	10.11	7.80	11.93	39.61
	GRU4Rec	17.55	100	21.55	26.85	12.88	17.03	17.33	46.89
	BiSA	**18.05**	100	**24.52**	**34.72**	13.35	**17.73**	**18.64**	**49.85**
separate p=1/4	NCF	11.37	100	16.32	17.57	9.06	8.37	12.25	40.76
	GRU4Rec	17.38	100	22.26	23.88	9.44	11.85	16.36	44.12
	BiSA	17.53	100	24.33	26.40	9.64	12.42	17.17	45.42
		Yelp							
latest continuous p=1/16	NCF	23.18	100	25.57	16.90	13.60	5.85	20.79	41.41
	GRU4Rec	26.09	100	25.82	17.22	15.62	6.08	22.33	41.10
	BiSA	25.35	100	26.34	12.85	**19.48**	7.51	**23.72**	40.38
oldest continuous p=1/16	NCF	23.21	100	21.13	15.97	10.22	4.56	18.19	40.71
	GRU4Rec	25.99	100	24.77	24.68	11.06	6.46	20.61	44.21
	BiSA	25.16	100	**31.53**	26.53	14.07	**8.66**	23.59	**45.18**
separate p=1/4	NCF	23.81	100	27.25	24.60	5.73	3.32	18.92	43.15
	GRU4Rec	**26.48**	100	29.97	**27.73**	4.56	4.64	20.34	44.61
	BiSA	26.30	100	30.47	26.32	6.64	5.02	21.14	44.28

not helpful for their recommendations, which is equivalent to data optimization by users. It also explains why “separate” with p=1/8 outperforms “all” in ML-100K. These results are consistent with the research in data minimization [3, 12, 75].

For the other two strategies, i.e., “latest continuous” and “oldest continuous” , the results show they perform not very consistently in different datasets. This could be caused by the characteristics of different datasets. It reminds us to design data disclosure mechanisms carefully according to the characteristics of data we deal with in real-world applications.

In summary, the platform mechanism affects both the possibility of a user to disclose and the volume of her/his disclosed data. Normally, a finer granularity and a more free data disclosure strategy can improve recommendation results for users and better protect users’ privacy at the same time. However, the price is that a large action space leads to slow convergence on the optimal user policy. As shown in Fig. 3, “separate” with p=1/8 (128 possible options) converges much slower than other mechanisms. It indicates users might be hard to find their best policies under these fine-grained mechanisms. Thus, we constraint users’ possible choice number to 16 for all mechanisms for fair comparisons in the latter experiments.

Table 7. Results for different user group compositions on ML-100K. All the results are averaged on the last 20 epochs.

NDCG@100 (%)				
Strategy	#100% Non-sen	#100%normal	#100 % sensitive	#1:1:1
latest $p = 1/16$	19.45	16.64	10.50	16.31
oldest $p = 1/16$	19.45	18.76	11.87	17.33
separate $p = 1/4$	19.45	19.28	9.27	16.36
statistics of data disclosure (% of disclosed data (avg. # user))				
Strategy	#100% Non-sen	#100% normal	#100 % sensitive	#1:1:1
latest $p = 1/16$	100 (637)	15.70 (429.7)	11.66 (215.5)	70.96 (449.9)
oldest $p = 1/16$	100 (637)	24.55 (378.0)	15.86 (201.3)	66.88 (411.9)
separate $p = 1/4$	100 (637)	25.36 (462.1)	15.11 (163.4)	68.13 (422.2)

4.5 Study 2: Impact of Recommender

This study answers whether a better model (BiSA) or a worse model (NCF) will attract users to disclose more data or not. Table 6 reports the results on three different data disclosure strategies with optimal granularity $p = 1/16$ except the $p = 1/4$ on "separate" for keeping the same number of data disclosing choices. It can be observed that:

(1) The results show that a more powerful model BiSA can attract sensitive users to disclose more data by improving the recommendation results for all platform mechanisms on all datasets.

(2) Though a better model usually can incentive users to disclose data, the total volume of data disclosed by normal users is not always increased. One reason is that marginal recommendation utility by disclosing more data may decrease on a better model considering the model already predicted precisely based on the disclosed data. This phenomenon is prominent on the BiSA in "latest continuous" considering a better sequential model may rely less on older behavior data [33, 68].

In summary, the Table 6 results suggest that the platform may pay more attention towards mechanism optimization while always stick to a better recommender system.

4.6 Ablation Study

Here, we study the impacts of compositions of user groups and the privacy sensitive hyper-parameter w_i. Due to page limitation, we only report the results based on GRU4Rec on ML-100k.

4.6.1 *User Group Compositions.*

In this subsection, we adjust the user group composition where each user in the dataset is non-sensitive/normal/sensitive. The average percentage of disclosed data and recommendation results are reported in Table 7. The platform can get higher revenues when users are less concerned about their privacy risks. Moreover, user group compositions play a more critical role in the platform's revenue than the data disclosure mechanisms. This encourages the platform to take more actions on privacy protection to prevent users from becoming sensitive.

4.6.2 *Privacy Sensitive.*

We report the effects of the hyper-parameter w_i in Fig. 4. The results show that all mechanisms perform quite stable on both recommendation results and data disclosure with

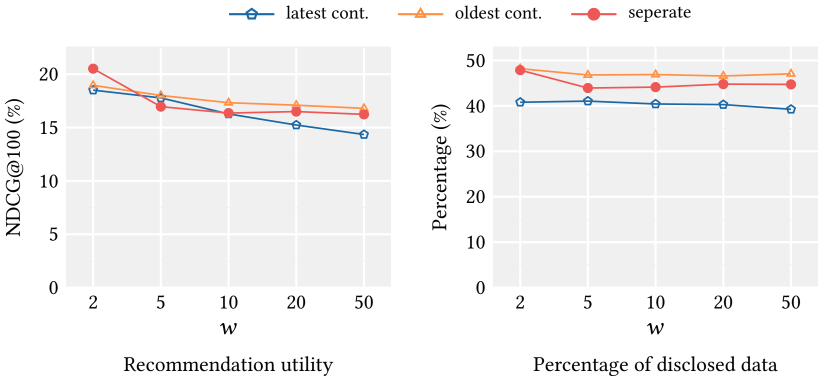


Fig. 4. Averaged results of different w on ML-100K.

different hyper-parameter w, especially when $w > 5$. The reason for these results is that a user will barely disclose their data unless she/he observes a significant improvement on the recommendation results when her/his privacy sensitivity is very high (e.g., $w > 5$). This also demonstrates that the conclusions of our previous experiments are stable.

4.7 Summarized Insights

In this subsection, we summarize several mentioned insights as below.

(1) We derive a negative attitude towards the "all or nothing" binary mechanism due to its worst performances compared to other proposed mechanisms.

(2) The platform mechanism affects both the possibility of a user to disclose and the volume of her/his disclosed data. A finer granularity mechanism will normally attract more users to disclose while the volume of her/his disclosed data and the recommendation performances are not always monotonically increasing.

(3) Though a better model usually can incentive users to disclose data, the total volume of data disclosed by normal users is not always increased due to the marginal effects, which suggests the platform may pay more attention towards mechanism optimization based on a specific recommender system rather than always stick to the recommender system optimization.

5 RELATED WORK

In this section, we will review previous works which are highly related to ours in the three fields, i.e., recommender systems, privacy research in recommender systems, and simulation.

5.1 Recommendation Systems

Recommender systems play an essential role in today's web service platforms, e.g., e-commerce [46, 77] and social media [14, 81], since they provide a personalized and convenient tool for every user to alleviate the information overload problem or explore serendipity things. Besides the attention of industry, recommender systems have also become the most active direction in information retrieval research [59, 76, 85].

Early works on recommender systems mainly model the users' interests statically as collaborative filtering (CF) task with implicit feedback. Early representative works include item-base CF algorithms [46, 63] and matrix factorization (MF) [38, 54] Recently, deep learning has also revolutionized collaborative filtering. One line of research seeks to improve the CF models with the representation learned from auxiliary information, e.g., text [72] and images [74] using deep learning models. While more mainstream way is to take the place of conventional CF models with more powerful neural models, like neural collaborative filtering (NCF) [23] and graph neural network based recommendation models [21, 81].

In recent years, sequential recommendation has become another mainstream task in recommender systems since it can better capture users' dynamic interests from their historical behaviors [59]. Sequential recommendation has also experienced the development process from traditional markov chain based models [61, 64] to neural sequential models, e.g., GRU4Rec [24, 25] and self-attention models [33, 68]. Considering that sequential recommendation has become the mainstream in real-world applications [42, 48], we study the proposed task with sequential models in this paper.

5.2 Privacy in Recommender Systems

The research about privacy concerns in recommender systems can be classified into two categories: privacy-preserving recommendation modeling and decision making in privacy.

Privacy-preserving recommendation modeling mainly aims to protect user's sensitive information from being leaked by designing specific models. An emerging paradigm is to use federated learning to train recommender systems without uploading users' data to the central server [45, 55, 58, 73]. Federated learning dramatically enhances user privacy since user data never leaves their devices. However, recent works have shown that federated learning can unintentionally leak information through gradients [43, 87] and is also vulnerable to attacks like membership inference attacks [51, 56]. To address such issues, differential privacy [17], a powerful mathematic framework for privacy, has been employed to guarantee user privacy in the procedure of recommender systems [2, 19, 50, 66]. The basic idea of this paradigm is to add random noise into the recommender system to prevent information leakage. As a promising framework, one limitation of differential privacy is that it usually decreases performance [16].

Decision making in privacy from other disciplines, e.g., economic [44], management sciences [15], and human–computer interaction [36, 37], mainly focus on studying the problem like where privacy concerns come from and how to mitigate them. They mainly study the procedure of user's decision making about information disclosure using the *privacy calculus theory* [15, 41], which views privacy as an economic commodity. It is to say that the user decides to disclose his/her information by weighing the anticipated risks of disclosing personal information against the perceived utility. Numerous works have studied the factors that influenced the user's decision using questionnaires or mock-up applications [9, 36, 37, 82]. Multiple studies highlight that "control" is a key factor in decision making about privacy, and providing control over the recommendation process to users can reduce their privacy concerns [9, 83]. Going a step further, in this paper, we give users not only control over whether or not to disclose data, but also control over which data to disclose. Then we investigate the consequences caused by this novel setting, including how users make choices and how different platform mechanisms and recommendation models perform.

Another close work to ours is [78] that studies a recommendation task where a small set of "public" users who disclose all their ratings (large amount) and a large set of "private" users refuse to disclose their data. Our work differs from [78] in the following aspects: i) Most importantly, as explained in the introduction and last paragraph, our goal is not the performances of the recommender systems; ii) We provide users with more fine-grained control over their data; iii) our task is built on implicit feedback, which is the mainstream of the real-world applications.

5.3 Simulation

Recent years have witnessed the wide applications of the simulation techniques among various scenarios, e.g., recommendation [6, 28, 47, 80], autopilot [57], traffic scheduling [1, 13] and robotic [60]. The primary reason to utilize simulation is that straightforwardly conducting experiments in the real world may remain too expensive [65] and risky [57]. Besides, the solutions derived from simulations can be transferred to solve the real-world problems [65, 71].

In the research areas of recommender systems, it is of great significance to utilize the carefully designed simulation environments to efficiently evaluate recommendation policy [65] or draw insightful conclusions for specific studies such as societal impact analysis [6]. Ie et al. [27] builds upon a simulation environment for slate-based recommender systems, which facilitates the recommendation policy evaluation. Shi et al. [65] proposes to utilize the historical user behavior data to train the simulator and verifies that the policies trained in the simulator achieve superior online performance. A series of works [6, 47, 80] utilize simulation environments to get in touch with user society impacts over recommender systems such as fairness and societal biases.

There also exist a series of works to simulate user decisions to maximize their personalized utilities [30, 32, 62]. Typically, the user is modeled as a rational agent whose policy can be learned following a trial-and-error schema such as RL-based algorithms [32, 34]. In this work, each user is modeled as a rational agent to optimize his (her) unique data disclosure policy under a designed platform mechanism. The efficiency of recommender systems is also evaluated within such a simulation environment.

6 CONCLUSIONS AND FUTURE WORK

This paper proposes a privacy aware recommendation framework based on privacy calculus theory to study what will happen if the platform gives users control over their data. To avoid the great cost in online experiments, we propose to use reinforcement learning to simulate the users' privacy decision making under different platform mechanisms and recommendation models on public benchmark datasets. The results show a well-designed data disclosure mechanism can perform much better than the popular "all or nothing" binary mechanism. Our work provides some insights to improve current rough solutions in privacy protection regulations, e.g., opt-in under GDPR and opt-out under CCPA.

This paper only takes the first step in studying users' privacy decision making under different platform mechanisms, and several directions remain to be explored. First, a more complex and accurate privacy cost function can help us better understand users' privacy decision making. In this work we have modeled different users with their individual privacy sensitivity weights, and one may modify the privacy cost function on the effects of the users' trust towards the platform in the future. Second, more sophisticated platform mechanisms are also worth exploring. Recent mechanism design works also turn to the perspectives of deep neural network based mechanism designs, which can be explored with our proposed framework. Last but not least, deploying online experiments and analyzing users' decisions in real-world can facilitate further researches.

REFERENCES

[1] Monireh Abdoos. 2020. A Cooperative Multiagent System for Traffic Signal Control Using Game Theory and Reinforcement Learning. *IEEE Intelligent Transportation Systems Magazine* 13, 4 (2020), 6–16.

[2] Arnaud Berlioz, Arik Friedman, Mohamed Ali Kaafar, Roksana Boreli, and Shlomo Berkovsky. 2015. Applying Differential Privacy to Matrix Factorization. In *Proceedings of the 9th ACM Conference on Recommender Systems* (Vienna, Austria). Association for Computing Machinery, New York, NY, USA, 107–114.

[3] Asia J. Biega, Peter Potash, Hal Daumé, Fernando Diaz, and Michèle Finck. 2020. *Operationalizing the Legal Principle of Data Minimization for Personalization.* Association for Computing Machinery, New York, NY, USA, 399–408.

[4] Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2021. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP).* IEEE, 141–159.

[5] Yinzhi Cao and Junfeng Yang. 2015. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy.* 463–480.

[6] Allison JB Chaney, Brandon M Stewart, and Barbara E Engelhardt. 2018. How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In *Proceedings of the 12th ACM Conference on Recommender Systems.* 224–232.

[7] Chong Chen, Fei Sun, Min Zhang, and Bolin Ding. 2022. Recommendation Unlearning. arXiv:2201.06820 [cs.IR]

[8] Chaochao Chen, Jun Zhou, Bingzhe Wu, Wenjing Fang, Li Wang, Yuan Qi, and Xiaolin Zheng. 2020. Practical Privacy Preserving POI Recommendation. *ACM Transactions on Intelligent Systems and Technology* 11, 5, Article 52 (jul 2020), 20 pages. https://doi.org/10.1145/3394138

[9] Tsai-Wei Chen and S. Shyam Sundar. 2018. *This App Would Like to Use Your Current Location to Better Serve You: Importance of User Assent and System Transparency in Personalized Mobile Services.* Association for Computing Machinery, New York, NY, USA, 1–13.

[10] David N Chin. 2007. Information filtering, expertise and cognitive load. In *International Conference on Foundations of Augmented Cognition.* Springer, 75–83.

[11] W. Jason Choi and Kinshuk Jerath. 2022. Privacy and Consumer Empowerment in Online Advertising. *Foundations and Trends® in Marketing* 15, 3 (2022), 153–212. https://doi.org/10.1561/1700000053

[12] Richard Chow, Hongxia Jin, Bart Knijnenburg, and Gokay Saldamli. 2013. Differential Data Analysis for Recommender Systems. In *Proceedings of the 7th ACM Conference on Recommender Systems* (Hong Kong, China). Association for Computing Machinery, New York, NY, USA, 323–326.

[13] Tianshu Chu, Jie Wang, Lara Codecà, and Zhaojian Li. 2019. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intelligent Transportation Systems* 21, 3 (2019), 1086–1095.

[14] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems* (Boston, Massachusetts, USA). Association for Computing Machinery, New York, NY, USA, 191–198.

[15] Mary J. Culnan and Pamela K. Armstrong. 1999. Information Privacy Concerns, Procedural Fairness, and Impersonal Trust: An Empirical Investigation. *Organization Science* 10, 1 (1999), 104–115.

[16] Josep Domingo-Ferrer, David Sánchez, and Alberto Blanco-Justicia. 2021. The Limits of Differential Privacy (and Its Misuse in Data Release and Machine Learning). *Commun. ACM* 64, 7 (jun 2021), 33–35.

[17] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* 9, 3–4 (aug 2014), 211–407.

[18] Joseph Farrell. 2012. Can privacy be just another good. *J. on Telecomm. & High Tech. L.* 10 (2012), 251.

[19] Chen Gao, Chao Huang, Dongsheng Lin, Depeng Jin, and Yong Li. 2020. *DPLCF: Differentially Private Local Collaborative Filtering.* Association for Computing Machinery, New York, NY, USA, 961–970.

[20] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (dec 2015), 19 pages. https://doi.org/10.1145/2827872

[21] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. 2020. *LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation.* Association for Computing Machinery, New York, NY, USA, 639–648.

[22] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web.* 173–182.

[23] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 173–182.

[24] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM international conference on information and knowledge management* (Torino, Italy). 843–852.

[25] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *Proceedings of International Conference on Learning Representations*.

[26] Eugene Ie, Chih-Wei Hsu, Martin Mladenov, Vihan Jain, Sanmit Narvekar, Jing Wang, Rui Wu, and Craig Boutilier. 2019. RecSim: A Configurable Simulation Platform for Recommender Systems. *CoRR* abs/1909.04847 (2019).

[27] Eugene Ie, Vihan Jain, Jing Wang, Sanmit Narvekar, Ritesh Agarwal, Rui Wu, Heng-Tze Cheng, Morgane Lustman, Vince Gatto, Paul Covington, et al. 2019. Reinforcement learning for slate-based recommender systems: A tractable decomposition and practical methodology. *arXiv preprint arXiv:1905.12767* (2019).

[28] Dietmar Jannach, Lukas Lerche, Iman Kamehkhosh, and Michael Jugovac. 2015. What recommenders recommend: an analysis of recommendation biases and possible countermeasures. *User Modeling and User-Adapted Interaction* 25, 5 (2015), 427–491.

[29] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446.

[30] Chunxiao Jiang, Linling Kuang, Zhu Han, Yong Ren, and Lajos Hanzo. 2017. Information credibility modeling in cooperative networks: Equilibrium and mechanism design. *IEEE Journal on Selected Areas in Communications* 35, 2 (2017), 432–448.

[31] Ginger Zhe Jin and Andrew Stivers. 2017. Protecting consumers in privacy and data Security: A perspective of information economics. *Available at SSRN 3006172* (2017).

[32] Johan Källström and Fredrik Heintz. 2019. Tunable dynamics in agent-based simulation using multi-objective reinforcement learning. In *Adaptive and Learning Agents Workshop (ALA-19) at AAMAS, Montreal, Canada, May 13-14, 2019*. 1–7.

[33] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 197–206.

[34] Michael N Katehakis and Arthur F Veinott Jr. 1987. The multi-armed bandit problem: decomposition and computation. *Mathematics of Operations Research* 12, 2 (1987), 262–268.

[35] Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of International Conference on Learning Representations*.

[36] Bart P. Knijnenburg, Svetlin Bostandjiev, John O'Donovan, and Alfred Kobsa. 2012. Inspectability and Control in Social Recommenders. In *Proceedings of the Sixth ACM Conference on Recommender Systems* (Dublin, Ireland). Association for Computing Machinery, New York, NY, USA, 43–50.

[37] Bart P. Knijnenburg and Alfred Kobsa. 2013. Making Decisions about Privacy: Information Disclosure in Context-Aware Recommender Systems. *ACM Trans. Interact. Intell. Syst.* 3, 3, Article 20 (oct 2013), 23 pages.

[38] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 08 (aug 2009), 30–37. https://doi.org/10.1109/MC.2009.263

[39] Karl Krauth, Sarah Dean, Alex Zhao, Wenshuo Guo, Mihaela Curmei, Benjamin Recht, and Michael I Jordan. 2020. Do Offline Metrics Predict Online Performance in Recommender Systems? *arXiv preprint arXiv:2011.07931* (2020).

[40] Walid Krichene and Steffen Rendle. 2020. On Sampled Metrics for Item Recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. Association for Computing Machinery, New York, NY, USA, 1748–1757.

[41] Robert S. Laufer and Maxine Wolfe. 1977. Privacy as a Concept and a Social Issue: A Multidimensional Developmental Theory. *Journal of Social Issues* 33, 3 (1977), 22–42.

[42] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-Interest Network with Dynamic Routing for Recommendation at Tmall. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (Beijing, China). Association for Computing Machinery, New York, NY, USA, 2615–2623.

[43] Wenqi Li, Fausto Milletarì, Daguang Xu, Nicola Rieke, Jonny Hancox, Wentao Zhu, Maximilian Baust, Yan Cheng, Sébastien Ourselin, M Jorge Cardoso, et al. 2019. Privacy-preserving federated brain tumour segmentation. In *International workshop on machine learning in medical imaging*. Springer, 133–141.

[44] Tesary Lin. 2019. Valuing intrinsic and instrumental preferences for privacy. *Available at SSRN* 3406412 (2019).

[45] Yujie Lin, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Dongxiao Yu, Jun Ma, Maarten de Rijke, and Xiuzhen Cheng. 2020. Meta Matrix Factorization for Federated Rating Predictions. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, New York, NY, USA, 981–990.

[46] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.Com Recommendations: Item-to-Item Collaborative Filtering. *IEEE Internet Computing* 7, 1 (jan 2003), 76–80.

[47] Eli Lucherini, Matthew Sun, Amy Winecoff, and Arvind Narayanan. 2021. T-RECS: A simulation tool to study the societal impact of recommender systems. *arXiv preprint arXiv:2107.08959* (2021).

[48] Fuyu Lv, Taiwei Jin, Changlong Yu, Fei Sun, Quan Lin, Keping Yang, and Wilfred Ng. 2019. SDM: Sequential deep matching model for online large-scale recommender system. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2635–2643.

[49] Jonathan R. Mayer and John C. Mitchell. 2012. Third-Party Web Tracking: Policy and Technology. In *2012 IEEE Symposium on Security and Privacy*. 413–427.

[50] Frank McSherry and Ilya Mironov. 2009. Differentially Private Recommender Systems: Building Privacy into the Netflix Prize Contenders. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Paris, France). Association for Computing Machinery, New York, NY, USA, 627–636.

[51] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 691–706.

[52] Lorenzo Minto, Moritz Haller, Benjamin Livshits, and Hamed Haddadi. 2021. Stronger Privacy for Federated Collaborative Filtering With Implicit Feedback. In *Fifteenth ACM Conference on Recommender Systems*. Association for Computing Machinery, New York, NY, USA, 342–350.

[53] Fatemehsadat Mireshghallah, Mohammadkazem Taram, Ali Jalali, Ahmed Taha Taha Elthakeb, Dean Tullsen, and Hadi Esmaeilzadeh. 2021. Not All Features Are Equal: Discovering Essential Features for Preserving Prediction Privacy. In *Proceedings of the Web Conference 2021* (Ljubljana, Slovenia). Association for Computing Machinery, New York, NY, USA, 669–680. https://doi.org/10.1145/3442381.3449965

[54] Andriy Mnih and Russ R Salakhutdinov. 2008. Probabilistic Matrix Factorization. In *Advances in Neural Information Processing Systems*, Vol. 20. Curran Associates, Inc., 1257–1264.

[55] Khalil Muhammad, Qinqin Wang, Diarmuid O'Reilly-Morgan, Elias Tragos, Barry Smyth, Neil Hurley, James Geraci, and Aonghus Lawlor. 2020. FedFast: Going Beyond Average for Faster Training of Federated Recommender Systems. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Virtual Event, CA, USA). Association for Computing Machinery, New York, NY, USA, 1234–1242.

[56] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 739–753.

[57] Błażej Osiński, Adam Jakubowski, Paweł Zięcina, Piotr Miłoś, Christopher Galias, Silviu Homoceanu, and Henryk Michalewski. 2020. Simulation-based reinforcement learning for real-world autonomous driving. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 6411–6418.

[58] Tao Qi, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2020. Privacy-Preserving News Recommendation Model Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, Online, 1423–1432.

[59] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. 2018. Sequence-Aware Recommender Systems. *ACM Comput. Surv.* 51, 4, Article 66 (jul 2018), 36 pages.

[60] Kanishka Rao, Chris Harris, Alex Irpan, Sergey Levine, Julian Ibarz, and Mohi Khansari. 2020. Rl-cyclegan: Reinforcement learning aware simulation-to-real. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11157–11166.

[61] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing Personalized Markov Chains for Next-Basket Recommendation. In *Proceedings of the 19th International Conference on World Wide Web* (Raleigh, North Carolina, USA). Association for Computing Machinery, New York, NY, USA, 811–820.

[62] Pedram Samadi, Hamed Mohsenian-Rad, Robert Schober, and Vincent WS Wong. 2012. Advanced demand side management for the future smart grid using mechanism design. *IEEE Transactions on Smart Grid* 3, 3 (2012), 1170–1180.

[63] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-Based Collaborative Filtering Recommendation Algorithms. In *Proceedings of the 10th International Conference on World Wide Web* (Hong Kong, Hong Kong). Association for Computing Machinery, New York, NY, USA, 285–295.

[64] Guy Shani, David Heckerman, and Ronen I. Brafman. 2005. An MDP-Based Recommender System. *Journal of Machine Learning Research* 6, 43 (2005), 1265–1295. http://jmlr.org/papers/v6/shani05a.html

[65] Jing-Cheng Shi, Yang Yu, Qing Da, Shi-Yong Chen, and An-Xiang Zeng. 2019. Virtual-Taobao: Virtualizing Real-World Online Retail Environment for Reinforcement Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 4902–4909.

[66] Hyejin Shin, Sungwook Kim, Junbum Shin, and Xiaokui Xiao. 2018. Privacy enhanced matrix factorization for recommendation with local differential privacy. *IEEE Transactions on Knowledge and Data Engineering* 30, 9 (2018), 1770–1782.

[67] H. Jeff Smith, Sandra J. Milberg, and Sandra J. Burke. 1996. Information Privacy: Measuring Individuals' Concerns about Organizational Practices. *MIS Quarterly* 20, 2 (1996), 167–196.

[68] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (Beijing, China). Association for Computing Machinery, New York, NY, USA, 1441–1450.

[69] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction.* MIT press.

[70] Huan Tang. 2019. The value of privacy: Evidence from online borrowers. *Working Paper, HEC Paris* (2019).

[71] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. 2017. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS).* IEEE, 23–30.

[72] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative Deep Learning for Recommender Systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Sydney, NSW, Australia). Association for Computing Machinery, New York, NY, USA, 1235–1244.

[73] Qinyong Wang, Hongzhi Yin, Tong Chen, Junliang Yu, Alexander Zhou, and Xiangliang Zhang. 2021. Fast-adapting and Privacy-preserving Federated Recommender System. *The VLDB Journal* (2021).

[74] Suhang Wang, Yilin Wang, Jiliang Tang, Kai Shu, Suhas Ranganath, and Huan Liu. 2017. What Your Images Reveal: Exploiting Visual Contents for Point-of-Interest Recommendation. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 391–400.

[75] Hongyi Wen, Longqi Yang, Michael Sobolev, and Deborah Estrin. 2018. Exploring Recommendations under User-Controlled Data Filtering. Association for Computing Machinery, New York, NY, USA, 72–76.

[76] Shiwen Wu, Wentao Zhang, Fei Sun, and Bin Cui. 2020. Graph Neural Networks in Recommender Systems: A Survey. *CoRR* abs/2011.02260 (2020).

[77] Xu Xie, Fei Sun, Xiaoyong Yang, Zhao Yang, Jinyang Gao, Wenwu Ou, and Bin Cui. 2021. Explore User Neighborhood for Real-time E-commerce Recommendation. In *2021 IEEE 37th International Conference on Data Engineering (ICDE).* IEEE, 2464–2475.

[78] Yu Xin and Tommi Jaakkola. 2014. Controlling privacy in recommender systems. In *Advances in Neural Information Processing Systems*, Vol. 27. Curran Associates, Inc., 2618–2626.

[79] Sirui Yao, Yoni Halpern, Nithum Thain, Xuezhi Wang, Kang Lee, Flavien Prost, Ed H Chi, Jilin Chen, and Alex Beutel. 2021. Measuring Recommender System Effects with Simulated Users. *arXiv preprint arXiv:2101.04526* (2021).

[80] Sirui Yao and Bert Huang. 2017. Beyond parity: Fairness objectives for collaborative filtering. *arXiv preprint arXiv:1705.08804* (2017).

[81] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph Convolutional Neural Networks for Web-Scale Recommender Systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (London, United Kingdom). Association for Computing Machinery, New York, NY, USA, 974–983.

[82] Bo Zhang and S. Shyam Sundar. 2019. Proactive vs. reactive personalization: Can customization of privacy enhance user experience? *International Journal of Human-Computer Studies* 128 (2019), 86–99.

[83] Bo Zhang, Na Wang, and Hongxia Jin. 2014. Privacy Concerns in Online Recommender Systems: Influences of Control and User Data Input. In *10th Symposium On Usable Privacy and Security.* 159–173.

[84] Minxing Zhang, Zhaochun Ren, Zihan Wang, Pengjie Ren, Zhunmin Chen, Pengfei Hu, and Yang Zhang. 2021. Membership Inference Attacks Against Recommender Systems. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security* (Virtual Event, Republic of Korea). Association for Computing Machinery, New York, NY, USA, 864–879.

[85] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. Deep Learning Based Recommender System: A Survey and New Perspectives. *ACM Comput. Surv.* 52, 1, Article 5 (feb 2019), 38 pages.

[86] Shijie Zhang, Hongzhi Yin, Tong Chen, Zi Huang, Quoc Viet Hung Nguyen, and Lizhen Cui. 2022. Pipattack: Poisoning federated recommender systems for manipulating item promotion. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining.* 1415–1423.

[87] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep Leakage from Gradients. In *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc.