Hindawi
Security and Communication Networks
Volume 2020, Article ID 6611463, 10 pages
https://doi.org/10.1155/2020/6611463

*Research Article*

# A Movie Recommendation System Based on Differential Privacy Protection

**Min Li**,[1] **Yingming Zeng**,[2] **Yue Guo**,[1] **and Yun Guo**[1]

[1]*College of Cyber Science, Nankai University, Tianjin 300017, China*
[2]*Hangtian INC, Beijing 100140, China*

Correspondence should be addressed to Yun Guo; guoyun@nankai.edu.cn

Received 19 October 2020; Revised 19 November 2020; Accepted 3 December 2020; Published 16 December 2020

Academic Editor: Hao Peng



In the past decades, the ever-increasing popularity of the Internet has led to an explosive growth of information, which has consequently led to the emergence of recommendation systems. A series of cloud-based encryption measures have been adopted in the current recommendation systems to protect users' privacy. However, there are still many other privacy attacks on the local devices. Therefore, this paper studies the encryption interference of applying a differential privacy protection scheme on the data in the user's local devices under the assumption of an untrusted server. A dynamic privacy budget allocation method is proposed based on a localized differential privacy protection scheme while taking the specific application scene of movie recommendation into consideration. What is more, an improved user-based collaborative filtering algorithm, which adopts a matrix-based similarity calculation method instead of the traditional vector-based method when computing the user similarity, is proposed. Finally, it was proved by experimental results that the differential privacy-based movie recommendation system (DP-MRE) proposed in this paper could not only protect the privacy of users but also ensure the accuracy of recommendations.

## 1. Introduction

With the development of information technology, tons of data are piled up on the Internet and users have many ways to access these data. For the users, what they spend most of their time on is no longer where to get information, but to find out what they are really interested in among numerous information. Therefore, the recommendation system came into being as an inevitable product of this era of big data. However, a key factor that usually influences the performance of recommendation systems is whether the amount of user data is enough or not and that may lead to a high risk of privacy leakage. In 2013, LG Corporation was charged for illegal collection of user data via smart TVs, which reflects the increasing awareness of privacy protection among users. What is more, IoT devices such as WiFi fingerprint which are frequently used in our daily life are also facing many kinds of security attacks [1, 2]. However, most of the existing recommendation systems [3–6] are developed based on the assumption of trusted servers. In most commonly used collaborative filtering algorithms, a trusted server collects all user data and makes user behavior analysis to give out personalized recommendations.

The application of differential privacy protection scheme in recommendation systems was first proposed by McSherry et al. [7, 8]. In their scheme, the server is responsible for encrypting user data, and random noise is added to each step of aggregation in the recommendation system. In such privacy protection schemes, only the circumstances that user data were published to a third-party from a trusted server were considered. However, other circumstances, such that when user data are uploaded from the local device to the cloud, attackers may eavesdrop on the transmission channel and launch a Man-in-the-Middle (MITM) attack or the attackers may directly hack into the cloud server and get access to sensitive user data, are not taken into consideration. Therefore, we reach our research question that how to apply differential privacy protection on users' local data under the basic assumption of an untrusted server. In this paper, existing differential privacy protection schemes and

commonly used recommendation algorithms are reviewed, and the application of localized differential privacy protection scheme in recommendation systems to solve the security issue in recommendation algorithms is investigated. The main contributions of this work are summarized as follows:

(i) A privacy budget allocation scheme that can dynamically allocate privacy budget is proposed based on the localized differential privacy protection. In this allocation scheme, users' behaviors such as movie watching records are allocated to the nodes in the privacy prefix tree with equal probability. After that, Laplace noise is added according to the privacy budget allocated to each node. This scheme could avoid the extreme circumstances of unevenly distributed privacy budget and added noise. In the meantime, this allocation scheme could also ensure the security of users' private data, as well as guaranteeing the accuracy of recommendation results by recording the combinatorial sequences of user behavior.

(ii) The traditional user-based collaborative filtering recommendation algorithm is improved by taking the specific application scene of movie recommendation into consideration. During the process of calculating user similarity to find out a similar group of the target users, a matrix-based method is proposed to replace the traditional vector-based method. More specifically, after the privacy prefix tree is generated, we construct a user-interest matrix $E$ according to users' movie watching records and the characteristics of combinatorial sequences, then apply the user-based collaborative filtering recommendation algorithm with matrix $E$ to calculate the similarity between users and find out the similar group of the target user, and finally give out the recommendation results.

## 2. Theoretical Basis

*2.1. Differential Privacy.* In 2006, Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith introduced the concept of differential privacy [8–12], which assumes that the attackers are able to access all information except the target information and makes it hard for attackers to access users' privacy via difference calculation. For the calculation result of the dataset, whether a single record is in the dataset or not has a negligible impact on the result. The basic definitions and properties of differential privacy involved in this paper are as follows.

Assume that datasets $D_1$ and $D_2$ have the same property structure, the symmetric difference between them is denoted as $D_1 \Delta D_2$, and the number of records in $D_1 \Delta D_2$ is denoted as $|D_1 \Delta D_2|$. If $|D_1 \Delta D_2| = 1$, we say that $D_1$ and $D_2$ are adjacent datasets.

*Definition 1* ($\varepsilon$-differential privacy). Let $\varepsilon$ be a positive real number and $M$ be a random algorithm that takes a dataset as input. Let $\mathrm{M}(x)$ denote the result obtained from a query of random algorithm $M$ and $R$ be a subset of $M(x)$. The algorithm $M$ is said to provide $\varepsilon$-differential privacy if, for all adjacent dataset pairs of $D_1$ and $D_2$ that differ on a single element and all subsets $R$ of $M(x)$, the following equation is satisfied:

$$\Pr[M(D_1) \in R] \leq e^{\varepsilon} \times \Pr[M(D_2) \in R]. \quad (1)$$

*Definition 2* (global sensitivity). For query function $f: D \longrightarrow R^d$, where $D$ is a dataset and $R^d$ is a $d$-dimensional vector of real numbers representing the query result, the global sensitivity of $f$ over all adjacent dataset pairs of $D_1$ and $D_2$ is described by

$$\mathrm{GS}_{f(D)} = \max \| f(D_1) - f(D_2) \|. \quad (2)$$

Global sensitivity describes the maximum range of changes when a query function is performed on a pair of adjacent datasets. It has nothing to do with the dataset, but it is only determined by the query function itself. The global sensitivity of the counting query is 1.

*Property 1* (Sequential composition). Assume that there are $n$ independent algorithms $M_1, M_2, \ldots, M_N$ whose privacy guarantees are $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n$, respectively. Then for the same dataset $D$, the composite algorithm $M(M_1(D), M_2(D), \ldots, M_n(D))$ is $(\sum_{i=1}^{n} \varepsilon_i)$-differentially private.

*Definition 3* (The Laplace mechanism). The Laplace mechanism adds Laplace noise to the original query outputs to realize $\varepsilon$-differential privacy. The noise is from Laplace distribution Lap $(\sigma)$ that can be expressed by the following probability density function with mean value 0 and scale parameter:

$$p(x) = \frac{1}{2\sigma} \exp\left( -\frac{|x|}{\sigma} \right). \quad (3)$$

*2.2. Privacy Prefix Tree.* The movie recommendation system based on differential privacy protection that we proposed in this paper combines users' movie watching records with the characteristic structure of prefix tree [13] to construct a privacy prefix tree (DP-tree), which can be considered as an improved prefix tree, and its structure is shown in Figure 1.

In Figure 1, Prior is a pointer point to the parent node; Value stores the value; Num is the number of times that this value shows; Depth is the depth of value; Child[$i$] is an array of pointers that point to the child nodes, and EndNum stores the number of times that the current node is the end of each path. The genres of all the movies that a user has watched are recorded and abstracted to a privacy prefix tree with a root node denoted as *Root*, in which each node represents a genre. In the privacy prefix tree, each branch is actually a sequence of the combination of different tags that represent different movie genres, and each sequence is started with node *Root*. The identical subsequences are merged and the

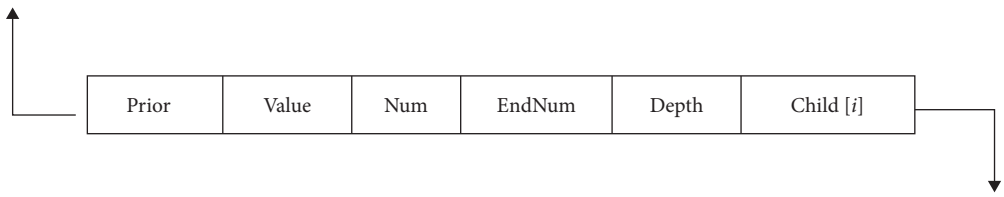


Figure 1: Data structure of a privacy prefix tree.

number of times that the subsequence shows is accumulated. Finally, the frequency that each genre of movie is watching as well as the frequency that each movie genre sequence shows is also recorded.

Based on the construction of the privacy prefix tree, the movie recommendation system proposed in this paper decomposes the record of user behavior, allocates privacy budget dynamically for the privacy prefix tree, and adds Laplace noise that satisfies the Laplace distribution. After that, a user-interest matrix $E$ is constructed according to the appearance frequency of different movie genres and the movie genre sequences that we get from the privacy prefix tree. Finally, matrix similarity is calculated to find out the similar user group of the target user, and a user-based collaborative filtering algorithm is adopted to give out a recommendation of movies.

## 3. Design of the Differential Privacy Protection Scheme

*3.1. Principal Steps in Differential Privacy Protection Scheme.* There are two principal steps when designing a differential privacy protection scheme: firstly, select appropriate privacy budget parameters and allocate a proper privacy budget for the protected data; secondly, add some noise interference to the protected data.

For the noise addition of the counting query, the Laplace mechanism is adopted to add interference to the privacy data, and the size of noise is closely related to the result of privacy budget allocation. More precisely, the privacy budget $\varepsilon$ is inversely proportional to the size of the added noise. Therefore, the privacy budget $\varepsilon$ not only determines the level of differential privacy protection but also influences the addition of noise interference; that is why $\varepsilon$ is the core parameter in differential privacy protection scheme. In this paper, we will mainly focus on how to allocate the privacy budget appropriately.

For the movie recommendation system based on differential privacy protection, firstly, a privacy prefix tree movie genre is constructed according to users' watching history. Movie genres and sequences that appear more frequently in the privacy prefix tree are more likely to arouse users' interest, and they also have a higher possibility of being attacked. In order to prevent the privacy budget from being exhausted, we usually allocate more privacy budgets for the data that are commonly used. However, the traditional privacy budget allocation method which evenly allocates the privacy budget to each node or each layer of the privacy prefix tree will lead to unreasonable addition of noise interference. What is more, limited privacy budget allocation for commonly used data may lead to quick exhaustion of the total budget, which will undermine the protection of users' privacy. Therefore, the problem of how to allocate the privacy budget reasonably is worth further investigation. In this paper, we proposed a scheme based on prefix tree allocation that can allocate the privacy budget $\varepsilon$ dynamically and reasonably according to the frequency of data use.

*3.2. Prefix Tree Privacy Budget Allocation Scheme.* The film recommendation system based on differential privacy introduced in this paper is based on the tree structure for data protection and encryption. Figure 2 shows the structure of user information based on the prefix tree structure; the genres are extracted as a movie feature and a privacy prefix tree is constructed based on the prefix tree structure. Specifically speaking, the genres (types) are extracted from users' watching records and stored in sequences in the substructure of a tree, where each path represents a certain combination of movie types and then records the showing frequency of each child node as well as the frequency of them appearing as leaf nodes. In order to reasonably allocate the privacy budget, we assign the privacy budget for each node in the privacy prefix tree proportionally. In particular, the root node $R$ is abstract and does not represent a real movie type, so it will not consume any privacy budget. All other nodes in subtrees need to be assigned a privacy budget.

Instead of storing the movie type directly in the prefix tree, the corresponding letter representation of the movie type is stored, as shown in Table 1.

Table 2 shows the data stored in each node in the privacy prefix tree structure shown in Figure 2.

As shown in Figure 3 and Table 2, the first path represents that the times (counts) of user watching movies that are tagged with $a$ are 10, $b$ is 5, and the end number is 2, which means that the user has watched 3 movies that are depicted by sequence $<a, b>$, and similarly, we can tell that he or she has also watched 2 movies that are depicted by sequence $<a, b, c>$.

As shown in Figure 3, assuming that the total privacy budget of the tree is $\varepsilon$, start with the first level of this tree; the frequencies of movie types $a$, $d2$, and $f$ are 10, 6, and 4, respectively. Therefore, the total privacy budget allocation proportion of the subtree with node $a$ as its root node should be $(10/20)\varepsilon$; thus, the dashed box shown in Figure 3 should totally be assigned $0.5\varepsilon$ privacy budget. Similarly, $\varepsilon_b = (0.5 * 0.5 * 0.6)/2\varepsilon$. When a movie type appears in different sequences, the privacy budget of it equals the total

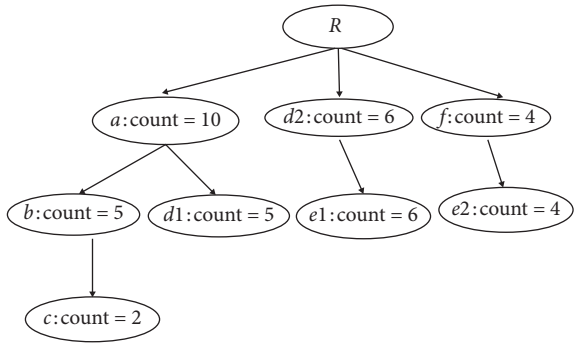


Figure 2: User information diagram based on the prefix tree structure.

Table 1: Mapping table of movie genres to tags.

| Genre | Love | Suspense | Action | Comedy | Plot | Tragedy |
|---|---|---|---|---|---|---|
| Tag | *a* | *b* | *c* | *d* | *e* | *f* |

Table 2: DP-tree data structure.

| Prior | Value | Num | EndNum | Depth | Child[*i*] |
|---|---|---|---|---|---|
| — | *R* | — | — | 0 | [*a*, *d*2, f] |
| *R* | *a* | 10 | 0 | 1 | [*b*, *d*1] |
| *A* | *b* | 5 | 3 | 2 | [*c*] |
| *B* | *c* | 2 | 2 | 3 | — |
| *A* | *d*1 | 5 | 5 | 2 | — |
| *R* | *D*2 | 6 | 0 | 1 | [*e*1] |
| *d*2 | *E*1 | 6 | 6 | 2 | — |
| *F* | *E*2 | 4 | 4 | 2 | — |
| *R* | *f* | 4 | 0 | 1 | [*e*2] |

sum of the allocated privacy budget in each sequence. For example, the privacy budget of movie type $d$ is $\varepsilon_d = \varepsilon_{d1} + \varepsilon_{d2} = 0.125\varepsilon + 0.15\varepsilon = 0.275\varepsilon$. According to Property 1, the sequential composition property of the differential privacy protection, it can be concluded that

$$\varepsilon = \varepsilon_a + \varepsilon_b + \cdots + \varepsilon_f. \quad (4)$$

It can be seen that, compared with other privacy budget allocation methods [14–18], the method of allocating privacy budget is based on the value of each node in the prefix tree, instead of just allocating uniformly according to the level structure. This allocation method can allocate the privacy budget reasonably and dynamically in the case that big differences exist among structures of the subtrees, and it also eliminates the requirement of artificially adjusting the value of privacy budget allocation.

*3.3. Prefix Tree Privacy Budget Allocation Algorithm.* The privacy budget allocation algorithm based on the prefix tree is shown as follows. *TMovie* stores the result of privacy budget allocation of movie type nodes; DP-tree movie type node $v$ and its privacy $\varepsilon_v$ are stored as $<v, \varepsilon_v>$ in the queue set *TQueue*; *Pv* is the statistical frequency of the current node $v$ being watched by users; *GetTop (LinkQueue Q*, *string r*, and *float e)* represents the dequeue function of header element (Algorithm 1).

In the above algorithm, the *TMovie* and *TQueue* sets are initialized to be empty after inputting the privacy budget $\varepsilon$, and the prefixed prefix tree and root node $R$ are constructed. Then, add the current node and its privacy budget to *TQueue* (when $R$ is not the root), and compare the weight of the current node with its parent node. If their weights are equal, assign half of the current privacy budget for both of them. Otherwise, compare the current node with its brother nodes and assign half of parent nodes' privacy budget to them according to their weight ratio. Repeat this process for each child node of the current node.

## 4. Design of DP-MRE

*4.1. Overall Framework of DP-MRE.* Figure 4 is the overall frame diagram of the movie recommendation system based on differential privacy protection, where the overall system is composed of five components. Firstly, users' private data are collected on their local devices, and then a prefix tree is constructed based on the collected data to dynamically allocate the privacy budget. Next, noise interference that obeys Laplace distribution is added, and then the users' data after being interfered with as well as public data are used together as the input of recommendation system and finally give out movie recommendations. The detailed meaning of each component in Figure 4 is as follows:

Public data refer to the public information related to users' private data from internal or external resources. We chose the MovieLens 1M dataset, which contains 100 million ratings from 6,000 users on nearly 4,000 movies. This dataset will be used as an experimental dataset and test dataset for experimental verification in this paper.

User data refer to the historical data of users' behavior collected from their personal devices. In this paper, we used the historical records of movies watched by users, such as the frequency of a user watching a certain type of movie, as well as users' ratings on these movies. What is more, this part of data is not interfered with.

Privacy quantification refers to the process that constructs the privacy prefix tree according to users' behavior records and allocates privacy budget according to the appearing times and frequencies of each node in the privacy prefix tree that we proposed in this paper.

Data perturbation refers to the process that adds noise which obeys Laplace distribution to each node in the privacy prefix tree according to its privacy budget, in order to interfere with the original data to ensure the security of users' private data while preserving the effectiveness of data. In other words, the interfered data should satisfy two necessary conditions: being secure enough to protect users' privacy and being effective enough to give out accurate recommendation in the subsequent recommendation stage.

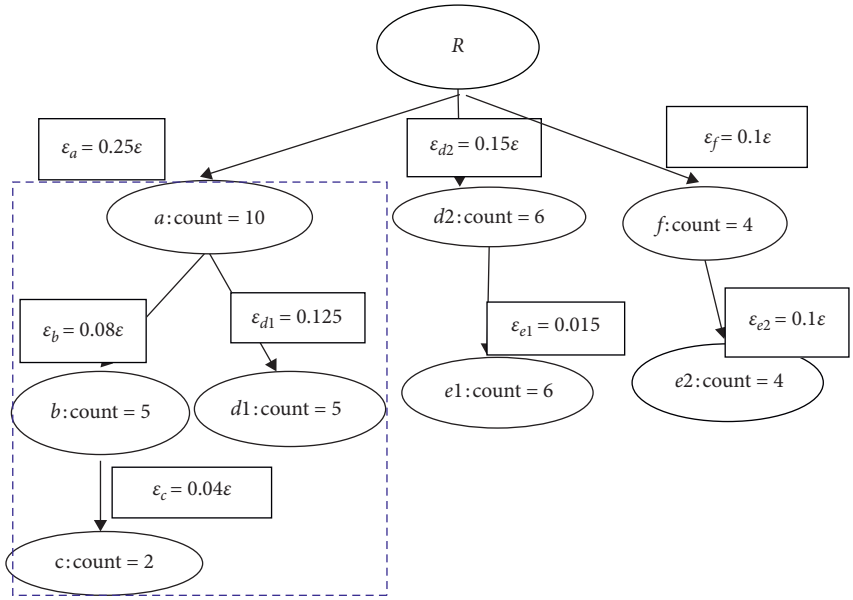


Figure 3: Privacy budget allocation scheme based on the prefix tree.

Input: Privacy budget $s$, prefix tree DP-Tree, root node R
Output: Privacy budget allocation results set TMovie
(1) Initialize set TMovie and TQueue to 0
(2) If ($R$ = = ' ')
(3) $\varepsilon_R = 0$
(4) R ⟶ child(R)
(5) Else
(6) Add the current node <$R$,$\varepsilon_R$> to TQueue
(7) While TQueue ≠ NULL Do
(8) GetTop(TQueue,R $\varepsilon_R$)
(9) IF R∈ TMovie Then
(10) $\varepsilon_R$⟵ privacy budget for node $R$ in TMovie
(11) TMovie ← <R,$\varepsilon_R + \varepsilon_{P_R}$>
(12) Else
(13) TMovie ← <R,$\varepsilon_{P_R}$>
(14) End If
(15) If ($P_R = P_{R-\text{parent}}$)
(16) $\varepsilon \leftarrow \varepsilon/2$
(17) Else
(18) $\varepsilon \leftarrow (\varepsilon - \varepsilon_{P_R})/2$
(19) For $v$ (child node of the current node)
(20) $P_v$← frequency of watching movies with tag v
(21) Append <$v$, $\varepsilon_{P_v}$> to TQueue
(22) End For
(23) End while

Algorithm 1: Privacy budget allocation algorithm.

Recommendation refers to the final stage of our DP-MRE system design, in which an untrusted third-party server obtains the data after perturbation, that is, after adding Laplace noise, and then uses these data to build a user-interest matrix according to user's preference on different types of movie. Next, similarity calculation based on the multidimensional matrix is performed to find out similar user groups, and a user-based collaborative filtering algorithm is adopted to give out a final recommendation for users.

*4.2. User-Based Collaborative Filtering Algorithm.* The user-based collaborative filtering recommendation algorithm [19–21] is usually composed of two parts: (1) to calculate the user similarity; (2) to recommend the interested contents of similar user groups to the target user.

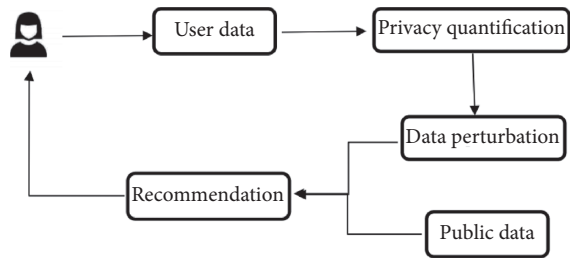


Figure 4: Frame diagram of the movie recommendation system based on differential privacy protection.

This paper extends the traditional method of computing vector-based similarity to matrix-based similarity and further combines the watching frequency of movie types as well as the combinatorial sequence of movie types. The specific method is to construct an N ∗ N user-interest matrix $E$ with movie type as both horizontal and vertical quantities. For example, assume that $a{\sim}n$ represent movie types and the user-interest matrix $E$ is constructed as follows:

$$E = \begin{pmatrix} p_{aa} & q_{ab} & \cdots & q_{an} \\ q_{ba} & p_{bb} & \cdots & q_{bn} \\ \vdots & \vdots & \ddots & \vdots \\ q_{na} & q_{nb} & \cdots & p_{nn} \end{pmatrix}, \tag{5}$$

where the diagonal of the matrix, that is, the set $P = \{p_{aa}, p_{bb}, \ldots, p_{mn}\}$, represents users' rating scores on movie type $a \sim n$; other quantities $q_{mn}$ represent users' rating score on certain movie type sequences. For example, $p_{aa} = 3, p_{bb} = 2, q_{ab} = 2$ indicate that the user has an interest score of 3 for type $a$ movies, 2 for type $b$ movies, and 2 for $<a,b>$ sequence.

As shown in Figure 3, the privacy prefix tree is constructed from user $A$'s movie watching record. According to the values of each node in the prefix tree and the sequence relationship between movie types in Figure 3, the user-interest matrix of user $A$ can be constructed as follows:

$$E = \begin{pmatrix} 10 & 5 & 2 & 5 & - & - \\ - & 5 & 2 & - & - & - \\ - & - & 2 & - & - & - \\ - & - & - & 11 & 6 & - \\ - & - & - & - & 10 & - \\ - & - & - & - & 4 & 4 \end{pmatrix}. \tag{6}$$

After constructing the user-interest matrix, the similarity between users can be obtained via matrix similarity calculation. In this paper, the correlation coefficient is used to evaluate the similarity of two matrices. The correlation coefficient is an indicator used to measure the statistical relationship between two variables, and it is a ratio, which can also be regarded as a special form of covariance after the standardization that eliminates the impact of the variation of amplitude. The correlation coefficient could be either positive or negative, which represents the direction of correlation between two variables but does not change the degree of similarity. In other words, the degree of similarity between two variables is reflected by the absolute value of the correlation coefficient. The correlation coefficient used in this paper is calculated as follows:

$$r = \frac{\sum_m \sum_n (A_{mn} - \overline{A})(B_{mn} - \overline{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \overline{A})^2\right)\left(\sum_m \sum_n (B_{mn} - \overline{B})^2\right)}}, \tag{7}$$

where $\overline{A} = \text{mean}(A), \overline{B} = \text{mean}(B)$, matrix $A$ and $B$ are two matrices with the same size, $\overline{A}$ and $\overline{B}$ represent the mean value matrix of $A$ and $B$, respectively, and $r$ denotes the correlation coefficient which ranges in [1, +1]. It indicates that matrices $A$ and $B$ share high similarity when the absolute value of $r$ is close to 1, and when $r$ is close to 0, it indicates that matrices A and B are less similar.

The similarity of the rating scores on movie type *(a~c)* between *UserA* and *UserB~UserE* is calculated using the method described above in this section, and the results are as shown in Table 3.

According to the calculation results based on matrix similarity in Table 3, the similarity between *UserA* and *UserE* is the highest. However, if we change to use the Pearson correlation coefficient to evaluate the similarity between users, although the structure of the user-interest matrix of *UserA* and *UserE* shares the highest similarity, the common rating items of *UserA* and *UserC* will lead to the calculation result of the similarity between *UserA* and *UserC* being exactly 1, which is not consistent with the real situation. However, in the matrix-based similarity calculation method we proposed, the similarity between *UserA* and *UserE* is a little higher than that between *UserA* and *UserD*. Therefore, both the absolute value and the quantity structure of the matrices are taken into account in the method we proposed based on matrix similarity.

What is more, if we change to use the Euclidean distance to evaluate the similarity between users, if there are no common rating items between two users, the similarity it gives out would be relatively low even if the structure and value are highly similar to each other. For example, in Table 3, the similarity between *UserA* and *UserB*, *UserC*, and *UserE* is all relatively low. When calculating the similarity of matrices, we can easily notice that actually *UserA* and *UserE* have high similarity, and their similarities to *UserB* and *UserC* are also higher than the results given by Euclidean distance calculation.

Assuming that the number of users who need personalized recommendation is $u$ and the similar group of the target user is $K$, use $S(u,K)$ to denote the process of selecting items that user $u$ interested in from similar group $K$, denote the interest rating score of user $v$ to item $j$ as $r_{vj}$ and similarity between the interest of user $u$ and user $v$ as $w_{uv}$, and denote the user group who are interested in item $j$ as $N(j)$. Then, the interest rating score of user $u$ to item $j$ should be given by equation (8):

$$p(u, j) = \sum_{v \in S(u,K) \cap N(j)} w_{uv} \times r. \tag{8}$$

Table 3: User similarity calculated based on matrix similarity.

| User | User-interest matrix | Similarity to *UserA* (absolute value) |
|---|---|---|
| *UserA* | $\begin{pmatrix} 5.0 & 3.0 & 0 \\ 0 & 3.0 & 0 \\ 0 & 0 & 2.5 \end{pmatrix}$ | 1.000 |
| *UserB* | $\begin{pmatrix} 2.0 & 0 & 0 \\ 0 & 2.5 & 0 \\ 0 & 2.5 & 5.0 \end{pmatrix}$ | 0.546 |
| *UserC* | $\begin{pmatrix} 2.5 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ | 0.329 |
| *UserD* | $\begin{pmatrix} 5.0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 3.0 \end{pmatrix}$ | 0.875 |
| *UserE* | $\begin{pmatrix} 4.0 & 2.0 & 0 \\ 0 & 3.0 & 0 \\ 0 & 0 & 2.0 \end{pmatrix}$ | 0.977 |

After calculating $p\ (u,j)$, compare the value $p\ (u,j)$ between different users. If two users have a similar value of $p\ (u, j)$, it indicates that they share a similar interest in a particular item. Then, the recommendation results could be given out by sorting the values from largest to smallest and selecting the highest-ranked items.

*4.3. Analysis of Privacy Security.* In this section, the privacy security of the DP-MRE algorithm proposed in this paper is analyzed based on differential privacy protection. Let D1 and D2 be the adjacent dataset (i.e., d $(D_1, D_2) = 1$), $f(D_i)$ denotes the category set of users' private data, C denotes the size of the public movie set, $j$ denotes the users' private data, and z(j) is the size of the Laplace noise added to movie type $j$. From the definition of differential privacy, we can know that, for arbitrary $r = (r_1, \ldots, r_c) \in \text{Range}(\text{DP} - \text{MRE})$, if the algorithm DP - MRE satisfies

$$\Pr[\text{DP} - \text{MRE}(D_1) = r] \le e^{\varepsilon} \times \Pr[\text{DP} - \text{MRE}(D_2) = r], \tag{9}$$

or if the algorithm DP-MRE satisfies

$$\frac{\Pr[\text{DP} - \text{MRE}(D_1) = r]}{\Pr[\text{DP} - \text{MRE}(D_2) = r]} \le e^{\varepsilon}, \tag{10}$$

then we can conclude that the algorithm DP-MRE satisfies the $\varepsilon$-differential privacy protection.

According to the differential privacy protection proposed in this paper, the differential privacy protection is carried out on users' local private devices, so the privacy protection analysis only focuses on the steps of privacy budget allocation and noise addition, while there is no privacy leakage problem in the user similarity calculation and recommendation steps. Therefore, privacy security analysis can be performed in the privacy budget allocation and noise addition steps as follows:

$$\begin{aligned} \frac{\Pr[\text{DP} - \text{MRE}(D_1) = r]}{\Pr[\text{DP} - \text{MRE}(D_2) = r]} &= \prod_{j \in C} \frac{\Pr[\text{DP} - \text{MRE}(D_1)(j) = r(j)]}{\Pr[\text{DP} - \text{MRE}(D_2)(j) = r(j)]} \\ &\ge \exp\left( - \sum_{j \in C} \frac{1}{z(j)} \left| f_j(D_1) - f_j(D_2) \right| \right) \\ &\ge \exp\left( - \max_{d(D_1, D_2) = 1} \sum_{j \in C} \frac{1}{z(j)} \left| f_j(D_1) \right. \right. \\ &\quad \left. \left. - f_j(D_2) \right| \right) \ge e^{-\varepsilon}. \end{aligned} \tag{11}$$

In the first step, according to the sequential composition property of difference privacy, the noise is added to each category set independently; thus, the difference in privacy remains unchanged. Furthermore, the second step can be derived from the added Laplace noise and triangle inequality. Therefore, we have proved that the DP-MRE algorithm satisfies Inequality 11.

## 5. Experimental Results and Analysis

*5.1. Privacy Budget Allocation.* The key point in the application of differential privacy protection algorithm is to preserve users' privacy as well as the effectiveness of data in the meantime. On one hand, users' privacy is ensured by the differential privacy protection mechanism, which is realized by adding the noise satisfying Laplace distribution to users' personal data. On the other hand, the effectiveness means the property of data that it can still be analyzed and processed after being protected by a differential privacy scheme, and the analysis results should be relatively accurate. At the same time, to allocate the privacy budget reasonably should also be taken into consideration when designing a differential privacy protection scheme.

In order to evaluate the effectiveness of the prefix tree privacy budget allocation method proposed in this paper, the query error of each node in the tree structure is analyzed, and it is compared with the traditional allocation method which allocates the privacy budget uniformly or proportionally according to arithmetic or geometric series. Mean square error is adopted to evaluate the query error. Assume that the accurate value of a set of data is given by $(a_1, a_2, \ldots, a_n)$ and the approximate value is given by $(a_1', a_2', \ldots, a_n')$. Then, the mean square error (MSE) is given by equation (12).

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (a_i' - a_i)^2. \tag{12}$$

MovieLens 1M dataset, which contains 6,000 user ratings on nearly 4,000 films, was used and we designed a query for the Movies dataset and repeated the query $n$ times $(n = 10, 20, \ldots, 1000)$ to obtain the mean square error value generated by these $n$ queries. In order to get a more accurate

result and to avoid the extreme circumstance that the randomness of noise may lead to, the calculation of mean square error is repeated $d$ rounds ($d = 100$); for each round, the mean square error is denoted as MS ($i = 1, 2, \ldots, d$). Thus, we could get the average of the mean square error $\overline{\text{MSE}}$. The greater value of $\overline{\text{MSE}}$ reflects the larger noise and correspondingly infers a lower accuracy of query results. The calculation method of ($i = 1, 2, \ldots, d$) and $\overline{\text{MSE}}$ is shown in equation (11) and (12).

$$\begin{aligned} \text{MSE} &= \frac{1}{n}\sum_{j=1}^{n}\left(y_j - x_j\right)^2, \\ \overline{\text{MSE}} &= \frac{1}{d}\sum_{i=1}^{d}\text{MSE}. \end{aligned} \tag{13}$$

Denote the query defined on Movies dataset as $f$, $x_j$ is the result of the $j$-th query on $f$, and $y_j$ is the corresponding noise result.

As can be seen from Figure 5, under repeated attacks, the errors generated by all privacy budget allocation schemes are increasing. The traditional allocation method which evenly allocates privacy budget to each layer generates the largest error, which indicates that this method produces the largest error in the case of uneven distribution of tree structure. In the cases when the number of queries is relatively small, the error between privacy budget allocation schemes based on arithmetic difference and arithmetic ratio is not much different from that based on the prefix tree structure. However, with the increase of the number of queries, the noise error generated under the privacy budget allocation based on the prefix tree is lower than other methods. The results indicate that when the number of queries is relatively small, all privacy budget allocation schemes produce relatively similar errors, except the scheme that evenly allocates privacy budget based on layers. However, when the number of queries is large enough, the privacy budget allocation scheme based on the prefix tree performs better than all the other schemes.

*5.2. Performance of DP-MRE.* In order to reflect the impact of differential privacy on the recommendation quality of the recommendation system (DP-MRE) in this paper, we use precision and recall to evaluate the performance of the recommendation system. Precision and recall are two indicators that are commonly used to evaluate the efficiency and quality of information retrieval systems with chaotic data. Both of these two indicators range from 0 to 1. The closer their value is to 1, the higher the quality of the system is, in other words, the higher the accuracy of the results given out by the information retrieval system is. Precision is defined according to the prediction results, which indicates how many of the samples whose predictions are positive are truly positive, whereas recall is defined according to our original samples, which indicates how many positive samples are predicted correctly as positive. The definition of precision and recall in a recommendation system is shown as follows:

$$\begin{aligned} \text{precision} &= \frac{\#\text{ of effective recommended sets}}{\#\text{of total recommended sets}}, \\ \text{recall} &= \frac{\#\text{ of effective recommended sets}}{\#\text{ of total tested sets}}. \end{aligned} \tag{14}$$

In order to objectively analyze the feasibility and effectiveness in the film recommendation system of DP-MRE algorithm based on differential privacy protection proposed in this paper, it is compared with the S-DPDP algorithm based on differential privacy protection proposed by Shen et al. We set the difference privacy parameter $\varepsilon$ as an independent variable, took different values for the privacy budget parameter in the experiment, and controlled a single variable to compare multiple recommendation algorithms. In addition, in order to more intuitively reflect the impact of privacy protection on the overall recommendation algorithm, this paper also added the data recommendation algorithm *Baseline* without privacy protection scheme into comparison. Therefore, two algorithms with differential privacy protection scheme, S-DPDP and DP-MRE algorithm, as well as an algorithm without privacy protection are taken into comparison.

Figure 6 shows the impact of differential privacy protection on the precision of the recommendation system. From the experimental results, we could see that, for the recommendation system without privacy protection, the precision of the user-based collaborative filtering recommendation system is about 0.53, and differential privacy protection algorithms DP-MRE and S-DPDP indeed cause a certain degree of loss in recommendation precision. When the differential privacy parameter $\varepsilon$ is close to 1, the precision of DP-MRE and S-DPDP algorithm recommended is about 0.51. With the increase of the privacy parameter $\varepsilon$, the precision of DP-MRE and S-DPDP algorithms gradually increases to that of *Baseline* algorithm. Compared with S-DPDP, DP-MRE has a smaller loss of precision, since DP-MRE allocates the privacy budget according to the DP-tree structure, which maintains the type combination sequence and frequency characteristics of the movies watched by users and distributes the Laplace noise reasonably, therefore reducing the loss of recommended quality caused by noise addition. However, S-DPDP adopted an iterative algorithm to add noise, which blurs the similarity between users. Therefore, from the perspective of recommendation quality loss, DP-MRE performs better than S-DPDP algorithm, whereas DP-MRE has a higher time complexity in the privacy budget allocation process, which affects the overall system efficiency.

Figure 7 shows the impact of differential privacy protection on the recall rate of the recommendation systems. From the experimental results, we could see that, for the recommendation system without privacy protection, the user-based collaborative filtering recommendation system has a recall rate of around 0.51, and differential privacy protection algorithms DP-MRE and S-DPDP also cause a certain degree of recommendation quality loss. However, with the increase of the privacy parameter $\varepsilon$, the recall rate gradually increases to that of *Baseline* algorithm. In the

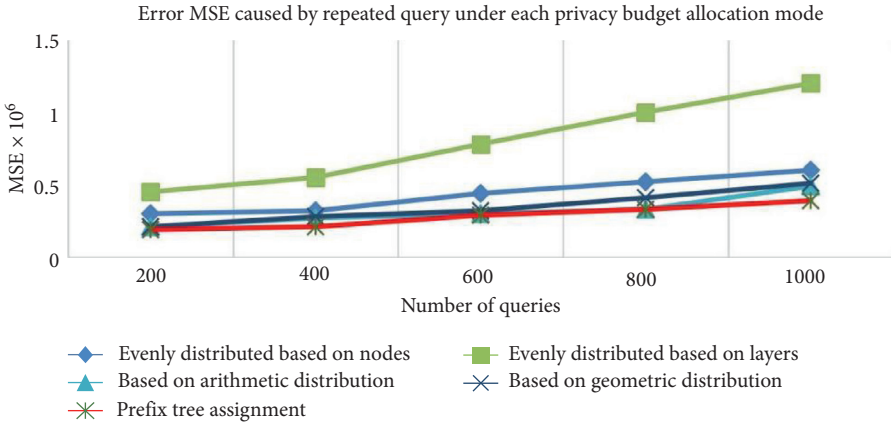


Figure 5: Error MSE caused by repeated query under each privacy budget allocation scheme.

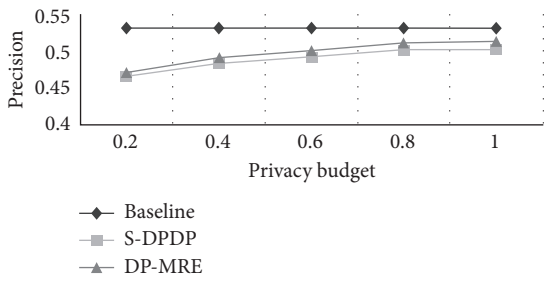


Figure 6: Impact of differential privacy protection on the precision of recommendation systems.

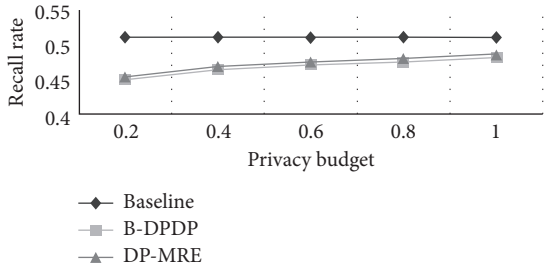


Figure 7: Impact of differential privacy protection on the recall rate of recommendation systems.

recommendation results, higher precision and recall rate indicate a higher recommendation system. According to the experimental results, the recall rate of DP-MRE is very similar to that of S-DPDP; especially when the dataset is relatively large, the recall rate of these two recommendation algorithms is basically the same, whereas the recall rate of DP-MRE is slightly higher than that of S-DPDP algorithm.

## 6. Conclusion

In this paper, we mainly introduced how to apply the differential privacy protection scheme in a movie recommendation system to protect users' privacy during the recommendation process, while in the meantime, ensuring the recommendation performance will not suffer too much loss. In conclusion, the scheme proposed in this paper firstly adds noise to local sensitive data in a dynamic manner to ensure users' privacy, then sends the data with added noise to the server for similarity calculation, and finally gives out movie recommendation via user-based collaborative filtering algorithm. The experimental results have shown that this scheme could achieve a considerable balance in the trade-off between preserving users' privacy and ensuring the performance of recommendation system. A meaningful attempt of combining differential privacy and recommendation algorithm has been made in our research. However, there are still a lot of open issues that are worth to be investigated in both fields of differential privacy and recommendation algorithms [22]. What is more, the application of differential privacy in recommendation algorithms other than user-based collaborative filtering algorithm will be further studied in our future research.

## Data Availability

All data are owned by third parties. The dataset used in this paper is the MovieLens 1M from https://grouplens.org/datasets/movielens/1 m/.

## Disclosure

An earlier version of this paper was presented at the International Symposium on Security and Privacy in Social Networks and Big Data (Social Sec 2020).

## Conflicts of Interest

The authors declare that they do not have any commercial or associative interest that represents conflicts of interest in connection with the work submitted.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 61672300) and the Industrial Internet Innovation and Development Project of the Ministry of Industry and Information Technology of PRC (Grant no. TC190H3WM).

## References


[1] Z. Yang and K. Järvinen, "Towards modeling privacy in WiFi fingerprinting indoor localization and its application," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 10, no. 1, pp. 4–22, 2019.

[2] J. Jung, H.-J. Kim, S.-J. Cho, S. Han, and K. Suh, "Efficient android malware detection using API rank and machine learning," *Journal of Internet Services and Information Security*, vol. 9, no. 1, pp. 48–59, 2019.

[3] K. Chaudhuri, A. Sarwate, and K. Sinha, "Near-optimal differentially private principal comp-onents," in *Proceedings of the Conference on Neural Information Processing Systems*, pp. 989–997, Toronto, Canada, December 2012.

[4] K. Chaudhuri and S. A. Vinterbo, "A stability-based validation procedure for differentilly private machine learning," in *Proceedings of the Conference on Neural Information Processing Systems*, pp. 2652–2660, Lake Tahoe, ND, USA, December 2013.

[5] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: asurvey of recent developments," *ACM Computing Surveys*, vol. 42, no. 4, p. 14, 2010.

[6] A. Guha Thakurta and A. Smith, "Optimal algorithms for private online learning in full-information and bandit settings," in *Proceedings of the Conference on Neural Information Processing Systems*, pp. 2733–2741, Lake Tahoe, ND, USA, December 2013.

[7] M. Hardt, K. Ligett, and F. Mcsherry, "A simple and practical algorithm for differentially private data release," in *Proceedings of the Conference on Neural Information Processing Systems*, pp. 2339–2347, Toronto, Canada, December 2012.

[8] F. McSherry and I. Mironov, "Differentially private recommender systems: building privacy into the net," in *Proceedings of the Knowledge Discovery and Data Mining*, pp. 627–636, Bangkok, Thailand, July 2009.

[9] Q. Ye, X. Meng, M. Zhu et al., "A review of localized differential privacy," *Journal of Software*, vol. 29, no. 7, pp. 159–183, 2018.

[10] C. Dwork, K. Kenthapadi, F. Mcsherry et al., "Our data, ourselves: privacy via distributed noise generation, advances in cryptology-EUROCRYPT 2006," in *Proceedings of the 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, St. Petersburg, Russia, June 2006.

[11] C. Dwork, "Calibrating noise to sensitivity in private data analysis," *Lecture Notes in Computer Science*, vol. 3876, no. 8, pp. 265–284, 2012.

[12] C. Dwork, F. Mcsherry, and K. Talwar, "The price of privacy and the limits of lp decoding, acm symposium on theory of computing," *Association for Computing Machinery*, vol. 23, 2007.

[13] S. Vágvölgyi, "Descendants of a recognizable tree language for prefix constrained linear monadic term rewriting with position cutting strategy," *Theoretical Computer Science*, vol. 732, pp. 60–72, 2018.

[14] M. Hay, V. Rastogi, G. Miklau et al., "Boosting the accuracy of differentially private histograms through consistency," *Proceedings of the VLDB Endowment*, vol. 29, pp. 1021–1032, 2009.

[15] R. Chen, B. C. M. Fung, and B. C. Desai, "Differentially private transit data publication:a case study on the Montreal transportation system," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 213–221, Beijing China, August 2012.

[16] T. Shang, Z. Zheng, W. Shu et al., "Algorithm of big data decision tree based on isometric privacy budget allocation," *Engineering Science and Technology*, vol. 51, no. 2, pp. 134–140, 2019.

[17] X. Wang, H. Han, Z. Zhang, Q. Yu, and X. Zheng, "Budget allocation method for tree index data differential privacy," *Computer Application*, vol. 38, no. 7, pp. 1960–1966, 2008.

[18] D. Hu and Z. Liao, "Differential privacy location privacy protection method for m- fork average tree," *Journal of Small and Micro Computer Systems*, vol. 40, no. 3, pp. 76–82, 2019.

[19] J. Paul Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM*, vol. 35, no. 3, 1997.

[20] J. J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Kno-wledge-Based Systems*, vol. 46, 2013.

[21] J. Chen, X. Wang, S. Zhao, F. Qian, and Y. Zhang, "Deep attention user- based collaborative filtering for recommendation," *Neurocomputing*, vol. 383, 2020.

[22] L. J. Helsloot, G. Tillem, and Z. Erkin, "BAdASS: preserving privacy in behavioural advertising with applied secret sharing," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 10, no. 1, pp. 23–41, 2019.