Fuzzy Systems and Data Mining VII
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA210171

Rényi Differential Privacy Protection Algorithm for SVD Recommendation Model

Shaoquan CHEN [a], Jianping CAI [a] and Lan SUN [a,1]
[a] *College of Computer and Data Science,Fuzhou University,Fuzhou 350108, China*

Abstract. With the widely use of recommendation systems in various mobile applications, privacy leakage has been a longstanding threat, for which many researchers have come up with a great number of methods that achieve the protective effect to a certain extent. However, the protection scope of these methods is limited, especially in the protection of original data. To address this issue, we propose a data perturbation based Rényi differential privacy algorithm to protect the SVD recommendation model. This paper uses the data perturbation method to perturb the original training dataset in the data preprocessing stage, then leverages the perturbed data to train the SVD model, and the unperturbed data is used as a test set to verify the accuracy of the model. Compared with the objective perturbation, gradient perturbation, and output perturbation, the data perturbation can protect a broader range and realize the corresponding functions of the other three perturbed methods by using the post-processing property of differential privacy. Experimental results show that the proposed method can effectively protect user privacy, improve the effectiveness of data, and generate better recommendation results without seriously affecting the accuracy of the model.

Keywords. Rényi differential privacy, data perturbation, SVD recommendation model, recommendation system, differential privacy

1. Introduction

In recent years, with the rapid development of Internet technology and the advent of the era of big data, a large amount of data will be generated every day. The recommendation system uses a large amount of user data to mine the consumption needs of potential users and recommend the items they need. In the fierce market competition environment, whether to use the recommendation algorithm to accurately grasp the needs of users and provide them with good services and then obtain more potential customers has become a major challenge for enterprises.

As an important and popular recommendation strategy, collaborative filtering [1] technology predicts the evaluation of various commodities by collecting a large number of users' evaluation data and then provides personalized recommendation services to users according to the predicted score. Compared with other recommendation strate-

[1]Corresponding Author: Lan Sun,College of Computer and Data Science,Fuzhou University,Fuzhou 350108, China; E-mail: lsun@fzu.edu.cn.

gies, collaborative filtering technology can achieve more accurate recommendations by using only a small amount of user evaluation data. It is relatively simple to realize, and there is no need to understand the correlation between items. It can not only generate relatively novel recommendation results, but also recommend more complex items to users. Therefore, this technology is also widely used by well-known enterprises such as Amazon.

However, when collecting a large number of user evaluation data, we will face a bigger problem: leaking the sensitive personal information of potential customers. Because some evaluation data will contain users' privacy preferences, we will inevitably leak some sensitive data when using user evaluation data for the recommendation. In a real scene, not only the original dataset may leak some personal information, but also some parameters may indirectly leak user-sensitive data in the training process. How to effectively deal with the effectiveness of the model and data protection has become a research hotspot in recent years.

In recent years, a great number of researchers have proposed many methods for privacy protection, such as k-anonymity [2], t-closeness [3], l-diversity [4] and so on. However, these methods have some limitations in background knowledge and attack assumptions, and can't provide sufficient privacy guarantee. Moreover, there is a lack of effective and strict proof to explain their privacy protection level.

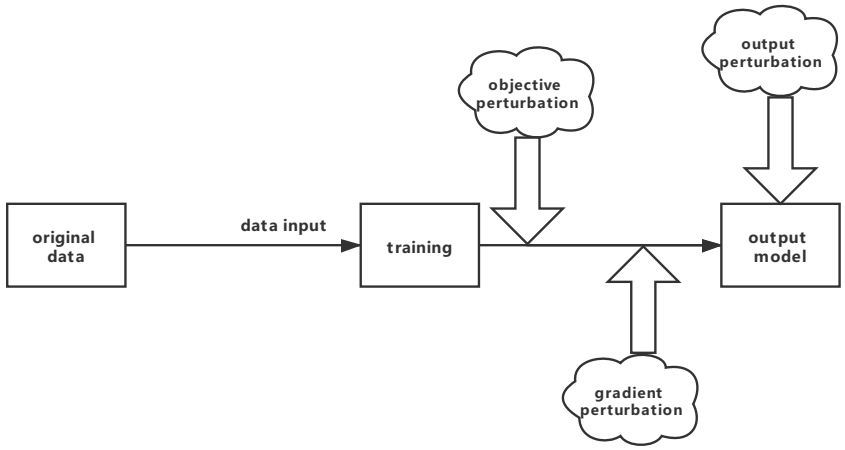
In 2006, Dwork et al. [5] proposed a differential privacy mechanism, which has attracted extensive attention. In recent years, this mechanism has been widely used in many fields for privacy protection. The mechanism fully considers the availability of data and the privacy and security of users. By adding random noise to perturb the published data, whether a single record is in the dataset has little impact on the calculation results. The commonly used perturbation methods mainly include objective perturbation, gradient perturbation, and output perturbation.

McSherry et al. [6] combined differential privacy with a recommendation system to realize corresponding privacy protection. They mainly started with the item covariance matrix and randomly added noise to the item covariance matrix to perturb the recommendation results. Hua et al. [7] mainly started from the perspective of objective perturbation and added random noise to the objective function to realize the corresponding privacy protection. Xian Zhengzheng et al. [8] proposed SVD ++ privacy protection recommendation models based on gradient perturbation, objective perturbation, and output perturbation. Yan Shen et al. [9] proposed a collaborative filtering method for differential privacy protection with social awareness. Zheng Jian et al. [10] used differential privacy and added random noise to a matrix decomposition recommendation model based on fusion label similarity, which still has high recommendation accuracy under the condition of ensuring user privacy. Friedman A et al. [11] proposed a method of using differential privacy in each step of matrix decomposition, Wang et al. [12] proposed a privacy protection model with objective perturbation by adding differential privacy noise to the bias matrix decomposition recommendation model. Kamalika Chaudhuri et al. [13] proposed differential privacy based on the exponential mechanism to add random noise to the PCA process, then the processed matrix is applied to the SVD model to realize the corresponding privacy protection.

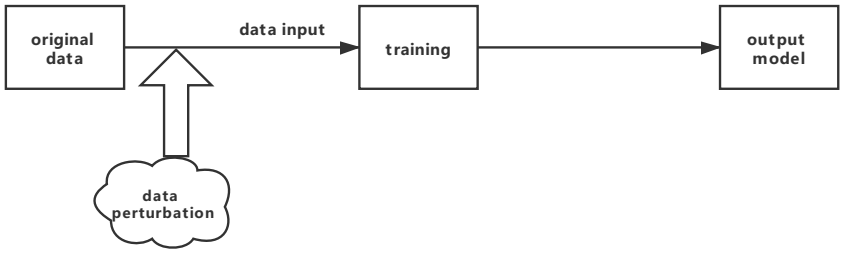
However, these privacy protection methods described above are based on three technologies: objective perturbation, output perturbation, and gradient perturbation to protect

the data. They do not protect the original data, and the original data is also likely to be attacked by attackers, so it is also very important to protect the original dataset.

To solve the above problems, this paper uses the data perturbation [14] method proposed in recent years, which makes the final model achieve (ε, δ)- differential privacy by adding random noise to the original training dataset. The difference between the data perturbation method used in this paper and the three perturbation methods mentioned earlier is shown in Figure 1.



(a) three common perturbed methods



(b) data perturbation

Figure 1. different perturbation methods

From Figure 1, we can see that the data perturbation method that we use can protect the final model with the help of the post-processing property of differential privacy.

Considering privacy protection and the final recommendation effect of the model, this paper adopts the SVD model based on the optimization algorithm of block coordinate descent and proposes a collaborative filtering algorithm based on data perturbation Rényi differential privacy protection, called data perturbation based Rényi differential privacy(DBRDP). The main contributions of this paper are as follows:

- In the process of data preprocessing, we add random Gaussian noise to the original training dataset and use the perturbed training dataset to train the SVD model so that the final model not only obtains the corresponding privacy protection but also has high model accuracy.

- We propose the DBRDP algorithm to protect the privacy of the BCDSVD model. In the process of data preprocessing, the original dataset is protected by adding the random Gaussian noise processed by Rényi differential privacy. Because of the post-processing property of differential privacy, the characteristic matrix is also protected, and the final model can be protected and still have high recommendation accuracy.
- We use the real datasets filmtrust, movielens100K and movielens1M in the experiment. The results show that the DBRDP algorithm has a better recommendation effect under the same privacy budget.

The rest of this paper is organized as follows: Section 2 introduces the relevant background knowledge, Section 3 introduces the process of the DBRDP algorithm, Section 4 shows the relevant experimental results and analyzes the experimental results, Section 5 concludes the work of the full text.

2. Background

In this section, we will introduce some related concepts of the BCDSVD model, differential privacy definition, and some related contents of Rényi differential privacy(RDP).

2.1. Block Coordinate Descent

Block coordinate descent (BCD) [15] algorithm is a method to optimize the subset of variables at the same time and decomposes the original problem into multiple subproblems. The core idea of this method is to divide the variables x to be optimized into p blocks, which are recorded as follows:$x = [x_1, x_2, \ldots, x_p]$. In each iteration, the original problem optimizes the ith block, and the other blocks remain unchanged, then the optimization expression of each subblock is:

$$x_i^{t+1} \in \arg\min_{x_i} f(x_1^{t+1}, x_2^{t+1}, ..., x_{i-1}^{t+1}, x_i^t, ..., x_p^t) \tag{1}$$

The pseudocode of the block coordinate descent algorithm is shown in algorithm 1:

Algorithm 1 Block Coordinate Descent Method

1:Initialize $x^0 \leftarrow \left[{x_1}^0, {x_2}^0, ..., {x_p}^0\right], t \leftarrow 0$;
2: **while** stopping criteria have not been satisfied **do**
3: **for** each $i \in \{1, 2, ..., p\}$ **do**
4: $x_i^{t+1} \leftarrow \arg\min_{x_i} f(x_1^{t+1}, x_2^{t+1}, ..., x_{i-1}^{t+1}, x_i^t, ..., x_p^t)$;
5: end **for**
6: $t \leftarrow t + 1$;
7: end **while**
8: **return** x^t ;

Compared with other optimization algorithms, the BCD algorithm has the advantages of fast speed, simple implementation, and strong stability. It will also be affected

by some properties of the objective function, such as the fast convergence that BCD can achieve $O(1/k)$ when the objective function is block strong convex. Because of its low iterative cost, the block coordinate descent method is also widely used in large-scale numerical optimization.

2.2. BCDSVD Recommendation Model

SVD recommendation model obtains the user characteristic matrix $U \in R^{d\times n}$ and the corresponding item characteristic matrix $M \in R^{d\times m}$ by singular value decomposition of the user scoring matrix $S \in R^{n\times m}$, and calculates the corresponding predicted score matrix $P \in R^{n\times m}$. SVD model is a common model in collaborative filtering recommendation algorithm. n represents the number of users, m represents the number of scoring items, d represents the dimension of a feature vector. Based on the SVD model, Paterek A [16] improves the model, and introduces the concept of bias vector $\mathbf{0}$ to learn the preference factors of users and items, and puts forward the BiasSVD model to further improve the recommendation accuracy of the model. The expression of the BiasSVD model is:

$$P = U^T M + b_n 1_m{}^T + 1_n b_m{}^T + c \tag{2}$$

In Eq.(2), $b_n \in R^{n\times 1}$ represents the user bias vector, $b_m \in R^{m\times 1}$ represents the item bias vector, c represents the mean value of all scoring records, and $\mathbf{1}$ represents a column vectors with all value of 1.

Compared with the traditional SVD recommendation model, BCDSVD [17] recommendation model uses the optimization algorithm of block coordinate descent in the solution process and solves the problems of capacity matrix inversion and sparse matrix solution. It has also been further improved in terms of speed and model accuracy, which is better than the traditional SVD model. BCDSVD model also has good expansibility. It can be used to solve the SVD recommendation model with implicit feedback. The high efficiency and good expansibility of the BCDSVD model make it have good usability and research value.

2.3. Differential Privacy

This subsection mainly introduces some differential privacy mathematical formulas and some common basic concepts and definitions.

There is only one data difference between two datasets $D, D' \in D^n$. We call such dataset sibling dataset. Based on sibling dataset, some common differential privacy concepts are defined as follows:

Definition 1 ((ε, δ) - Differential Privacy) [5]. A random function $M : D^n \to R^m$ satisfies (ε, δ)- differential privacy if the following conditions satisfy:

$$\Pr[M(D) \in S] \leq e^{\varepsilon} \Pr\left[M(D') \in S\right] + \delta \tag{3}$$

In Eq. (3), $S \in range(M)$. ε is the privacy budget, and its value range is $\varepsilon \geq 0$. The smaller the value of the privacy budget, the higher the degree of protection. The value range of δ is $0 \leq \delta \leq 1$, and when δ is 0, it is called pure differential privacy, and when $\delta > 0$, it is called approximate differential privacy.

Definition 2 (Gaussian Mechanism) [18]. Given the function $f : D^m \to R$, for any $D \in D^m$, the definition of Gaussian mechanism G is:

$$G(f(D)) = f(D) + v \tag{4}$$

In Eq. (4), v follows the Gaussian distribution $N(0, \sigma^2 I)$ with mean value of 0 and variance $\sigma \geq \frac{\sqrt{2\log(1.25/\delta)}\Delta_2 f}{\varepsilon}$, where f is L_2 - sensitivity is defined as

$$\Delta_2 f \triangleq \max_{D,D'} \| f(D) - f(D') \|_2 \tag{5}$$

Definition 3 (Rényi Divergence) [19]. For the two probability distributions P and Q which are defined on R, when $\alpha > 1$, the definition of Rényi divergence is as follows:

$$D_\alpha(P \| Q) \triangleq \frac{1}{\alpha - 1} \log E_{x \sim Q}(\frac{P(x)}{Q(x)})^\alpha \tag{6}$$

In Eq. (6), $P(x)$ and $Q(x)$ are the density at x. The relationship between Rényi divergence and differential privacy can obtain directly when $\alpha = \infty$. A random mechanism M satisfies ε-differential privacy if and only if it is distributed between two adjacent input datasets D and D', and satisfies Eq. (7):

$$D_\infty(M(D) \| M(D')) \leq \varepsilon \tag{7}$$

Definition 4 ((α, ε) - RDP) [19]. Given parameters $\alpha > 1$, privacy budget $\varepsilon \geq 0$. If a random mechanism $M : D \to R$ satisfies Eq.(8) for any sibling dataset:

$$D_\alpha(M(D) \| M(D')) \leq \varepsilon \tag{8}$$

we call the random mechanism satisfied (α, ε) - RDP. RDP is a relaxation of the concept of differential privacy. Using Rényi divergence to measure the difference between two sibling datasets can provide stronger privacy protection than (ε, δ)- differential privacy.

Definition 5 (RDP to (ε, δ)-Differential Privacy) [19]. Given a function M which satisfies (α, ε) - RDP, it will also satisfy $(\varepsilon(\delta), \delta)$- differential privacy, where

$$\varepsilon(\delta) \geq \varepsilon + \frac{\log(\frac{1}{\delta})}{\alpha - 1} \tag{9}$$

3. Algorithm

In this section, before introducing our DBRDP algorithm, we first introduce the data perturbation based Gaussian noise(DBGDP) and then propose an improved algorithm DBRDP for the data perturbation based Gaussian noise algorithm.

3.1. Data Perturbation Based Gaussian Noise

In this paper, when we mention the data perturbation, we will turn $z = (x, y)$ into $z' = (x + noise, y)$, which *noise* represents the random noise that we add, z represents the original data, and z' represents the perturbed data. In the process of algorithm processing, we mainly add random noise to the original training dataset at the data preprocessing stage, the random noise follows the Gaussian distribution $N(0, \sigma^2 I)$, where the mean value is 0, and the variance is $\sigma = \frac{d+1}{n\varepsilon}\sqrt{2\log\left(\frac{d^2+d}{\delta 2\sqrt{2\pi}}\right)} + \frac{1}{\sqrt{\varepsilon}n}$ [13], where d represents the dimension of query data, n represents the amount of query data, and ε represents the privacy budget. Then we apply the perturbed data to the training process of the BCDSVD recommendation model. Due to the post-processing property of differential privacy, it is not difficult to see that when using the processed perturbed data to process the block data by the BCD method, block data will also be protected. Moreover, when applying BCD to the SVD recommendation model, the output of the final model is also protected.

Compared with objective perturbation, gradient perturbation, and output perturbation, data perturbation has a broader protection range. It can even be said that the data perturbation can indirectly realize the output perturbation, objective perturbation, and gradient perturbation because of the post-processing property of differential privacy. In subsequent processing, it will also have varying degrees of impact on the other three corresponding perturbations. Moreover, it can protect data input, which can better prevent the attack of attackers. The pseudocode of data perturbation based Gaussian noise(DBGDP) algorithm is shown in algorithm 2.

Algorithm 2 Data Perturbation Based Gaussian Noise(DBGDP)

1: **Inputs**: Dataset D, privacy budget ε , privacy bias δ,variance σ ;
2: **Output**: perturbed dataset $\mathrm{D}_{privacy}$;
3: for all data instances $z_i (i \in \{1, 2, ..., n\})$,transform them to $z' = z_i + b$
$b \sim N(0, \sigma^2 I)$;
4: $\mathrm{D}_{privacy} = \{z'\}$;
5: **return** $\mathrm{D}_{privacy}$;

In this paper, we use the perturbed training dataset obtained by algorithm 2 to train the BCDSVD recommendation model. We assume that the predicted score matrix $\mathbf{P}$ is a function of characteristic parameter λ (column vector). For this problem, the loss function we use is the mean square error. We use the minimization function of the loss function $E(\lambda)$ [17] to solve the subsequent problem. The expression is:

$$E(\lambda) = \frac{1}{2} trace\left((\mathbf{J} * (\mathbf{V} - \mathbf{P}))^T (\mathbf{J} * (\mathbf{V} - \mathbf{P}))\right) + reg(\lambda) \tag{10}$$

In Eq. (10), $\mathbf{J}$ is the matrix to mark whether the user has scored the corresponding items, * represents the Hardmard product operator, $reg(\lambda)$ is a regularization function, which can be expressed as $reg(\lambda) = \frac{1}{2}\lambda^T diag(\kappa)\lambda$, κ is the vector composed of various characteristic parameters, and the corresponding gradient of the regularization function is $diag(\kappa)\lambda$. Following this, we can get the gradient expression of the loss function $E(\lambda)$ [17]:

$$\frac{\partial E(\lambda)}{\partial \lambda} = \sum_{i,j} (e_i{}^T (\mathbf{J} * (\mathbf{P} - \mathbf{V})) e_j \times \frac{\partial (e_i{}^T P e_j)}{\partial \lambda}) + \frac{\partial reg(\lambda)}{\partial \lambda} \tag{11}$$

In Eq. (11), e_i represents the basic column vector in which the ith element is 1 and the other elements are 0. Eq. (11) transforms the gradient solution of the loss function $E(\lambda)$ into the gradient solution of each predicted score.

Based on the related concepts and the input perturbed dataset, we can solve the corresponding gradients of the user characteristic matrix $\mathbf{U}$ and item characteristic matrix $\mathbf{M}$ under the BCDSVD model. Due to the post-processing property of differential privacy, the corresponding gradient perturbation will also be generated when using the perturbed data to solve the corresponding gradient. Therefore, the gradient expression corresponding to the user characteristic matrix is:

$$\frac{\partial E(\mathbf{U})}{\partial \mathbf{U}} = \mathbf{M}(\mathbf{J} * (\mathbf{P} - \mathbf{V}))^T + k_u \mathbf{U} + v \tag{12}$$

The item characteristic matrix is:

$$\frac{\partial E(\mathbf{M})}{\partial \mathbf{M}} = \mathbf{U}(\mathbf{J} * (\mathbf{P} - \mathbf{V}))^T + k_q \mathbf{M} + v \tag{13}$$

v is the noise added indirectly in the gradient processing process due to the use of the perturbed original dataset to train the model. There is no noise added to the gradient again. The random noise v follows the distribution of $v \sim N(0, \sigma^2 I)$. It is difficult to directly use Eqs. (12) and (13) to solve the user characteristic matrix and item characteristic matrix. Therefore, in the process of solving the model, we adopt the block idea to divide the original characteristic matrix by column. The subblocks S can express as:

$$\mathbf{S} = \mathbf{V} - b_u 1_q{}^T - 1_u b_q{}^T - c \tag{14}$$

After block processing, the corresponding gradient solution formula can transform into:

$$\frac{\partial E(\mathbf{u}_i)}{\partial \mathbf{u}_i} = (\mathbf{M} diag(\mathbf{e}_i{}^T \mathbf{J}) \mathbf{Q}^T + k_u \mathbf{I}) \mathbf{u}_i - \mathbf{Q}(\mathbf{J} * \mathbf{W})^T \mathbf{e}_i + v \tag{15}$$

$$\frac{\partial E(\mathbf{m}_i)}{\partial \mathbf{m}_i} = (\mathbf{U} diag(\mathbf{J}\mathbf{e}_i) \mathbf{U}^T + k_q \mathbf{I}) \mathbf{m}_i - \mathbf{U}(\mathbf{J} * \mathbf{V})^T \mathbf{e}_i + v \tag{16}$$

In addition, the gradient value after block processing is $\mathbf{0}$. After transposition, the general expression of user feature subblock $\mathbf{u}_i$ is:

$$\mathbf{u}_i = (\mathbf{M} diag(\mathbf{e}_i{}^T \mathbf{J}) \mathbf{Q}^T + k_u \mathbf{I})^{-1} \mathbf{Q}(\mathbf{J} * \mathbf{W})^T \mathbf{e}_i - (\mathbf{M} diag(\mathbf{e}_i{}^T \mathbf{J}) \mathbf{Q}^T + k_u \mathbf{I})^{-1} v \tag{17}$$

The item feature subblock $\mathbf{m}_i$ is:

$$\mathbf{m}_i = (\mathbf{U}diag(\mathbf{J}\mathbf{e}_i)\mathbf{U}^T + k_q\mathbf{I})^{-1}\mathbf{U}(\mathbf{J} * \mathbf{V})^T\mathbf{e}_i - (\mathbf{U}diag(\mathbf{J}\mathbf{e}_i)\mathbf{U}^T + k_q\mathbf{I})^{-1}v \quad (18)$$

Based on Eqs. (17) and (18), we can get the final user feature matrix $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_u]$ and item feature matrix $\mathbf{M} = [\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_q]$.

3.2. Data Perturbation Based Rényi Differential Privacy

When the privacy budget is small, the effect of some datasets in training the DBGDP-BCDSVD model is not very good, and the running efficiency is also worse than that of the non-noisy BCDSVD model. In recent years, Rényi differential privacy(RDP) proposed by Mironov I has been gradually applied. Under the same privacy budget, by using RDP, we can get better results than the general differential privacy. Therefore, this subsection uses Rényi differential privacy to control the random noise added to the original training dataset. With the help of the relationship between Rényi differential privacy and (ε, δ) - differential privacy, under the same privacy budget ε, using the relationship between (α, ε) - RDP and (ε, δ) - differential privacy, and inequality $\varepsilon(\delta) \geq \varepsilon + \frac{\log(\frac{1}{\delta})}{\alpha - 1}$. Under the premise of ensuring (ε, δ) - differential privacy, our model can get better performance. Under the same privacy protection, the method of random perturbation using Rényi differential privacy can improve the performance of our BCDSVD recommendation model and obtain a better recommendation effect. The specific flow chart of DBRDP-BCDSVD is shown in Figure 2.

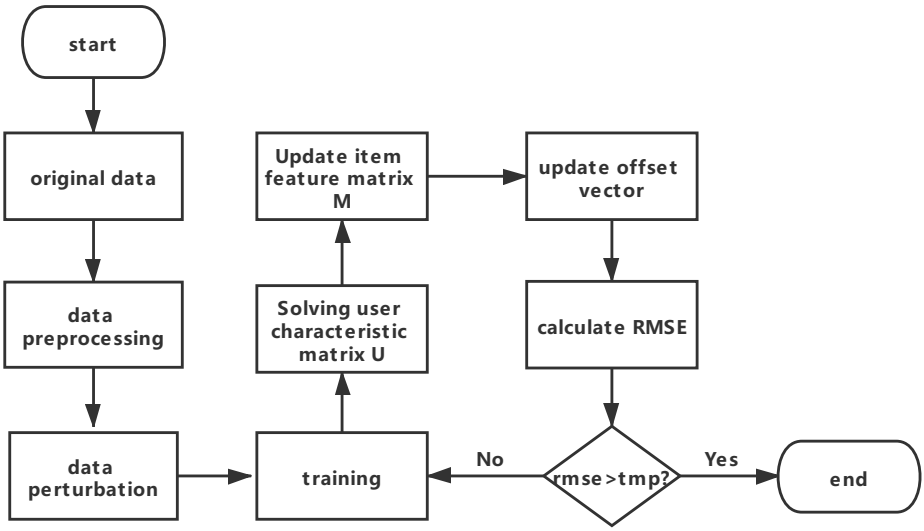


Figure 2. the flow chart of DBRDP-BCDSVD.

To prevent the algorithm from iterating continuously and affecting the efficiency, we set the iteration termination condition in the process of model training: when the model evaluation index rmse meets $rmse > tmp$, we terminate the algorithm, where tmp is the difference between the RMSE value of the previous round of training and a preset error threshold. The pseudocode of data perturbation based Rényi Differential Privacy(DBRDP) algorithm is shown in algorithm 3.

Algorithm 3 Data Perturbation Based Rényi Differential Privacy(DBRDP)

1: **Inputs**: Dataset D, privacy budget ε , privacy bias δ,variance σ,Rényi parameter α;
2: **Output**: perturbed dataset $\mathrm{D}_{privacy}$;
3: for all data instances $z_i(i \in \{1,2,...,n\})$,transform them to $z' = z_i + b$;
$b \sim N(0, \sigma^2 I)$;
4: $\mathrm{D}_{privacy} = \{z'\}$;
5: **return** $\mathrm{D}_{privacy}$;

When perturbed the original training dataset, it is not difficult to see that it is satisfied (α, ε) - RDP in the process of perturbation. Due to the post-processing property of differential privacy, when the BCDSVD model is trained with the perturbed dataset, the model will also be protected. After privacy processing is performed on the most original training dataset, we do not need to perform output perturbation, objective perturbation, and gradient perturbation on the model in the process of model training iterations. Due to using the perturbed training dataset, the user characteristic matrix will also be protected. The protected user characteristic matrix will also be used for subsequent model parameter updating, such as item characteristic matrix and offset vector. After privacy processing of the original dataset, other important data, such as user characteristic matrix and item characteristic matrix, will also be protected. Therefore, using the perturbed training dataset to train the BCDSVD model, its overall training process will be satisfied (α, ε) - RDP.

4. Experimental Results And Analysis

This section mainly implements the two algorithms proposed in Section 3 and analyzes the corresponding experimental results.

4.1. Experimental Datasets

This paper mainly adopts the movie scoring datasets filmtrust, movielens100K, and movielens1M, which are commonly used in the recommendation system. The size of filmtrust is [35494,3], the size of movielens100K is [100000,3], and the size of movielens1M is [1000209,3]. The dimension (i.e. the number of columns) of these three datasets is 3, and the third column of data is the user's rating data of items. Filmtrust is the data captured from the whole filmtrust website in 2011. Movielens is the data collected and rated by GroupLens research on its film website.

4.2. Some Main Parameters in Experiment

The main parameters that we use in the experiment are shown in Table 1.

Table 1. Experimental Parameters

Variable	*Description*	*Default Value*
ε	Privacy Budget	-
δ	Privacy Bias	1e-5
d	Dataset Dimension	3
n	Dataset Size	-
α	Rényi Parameter	1000

The default value of these parameters is set by relevant reference or by the structure of the dataset used in this experiment.

4.3. Evaluation

The recommendation system often takes some errors as performance evaluation index, such as RMSE, MSE, MAE, etc. In the process of the experiment, we use the root mean square error (RMSE) as the corresponding evaluation index, and its expression is:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{19}$$

Where n is the total number of samples, y_i is the real value of the ith sample, and $\hat{y}_i$ is the predicted value of the ith sample. The evaluation standard of root mean square error (RMSE) is: the smaller the value of root mean square error, the better performance of the system.

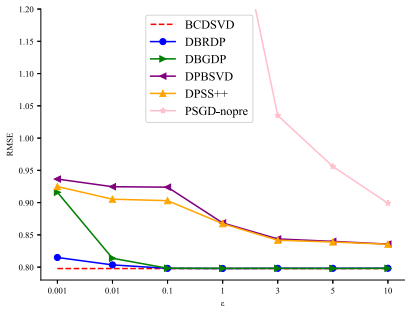
4.4. Experimental Results And Analysis

This paper takes the BCDSVD algorithm without privacy protection as the benchmark, the SVD algorithm PSGD based on gradient perturbation PSGD-nopre proposed by Berlioz A et al, and the DPBSVD algorithm based on gradient perturbation proposed by Paterek A, and the DPSS ++ algorithm based on gradient perturbation proposed by Xian Zhengzheng et al are compared with the data perturbation based Gaussian noise(DBGDP) and the data perturbation based Rényi differential privacy(DBRDP) proposed in this paper. In the process of comparative experiment, we conduct 10 experiments on a privacy budget value and calculate the average value of the 10 experiments as the evaluation index to compare the advantages and disadvantages of various algorithms. To facilitate viewing, we summarize all the experiments in Table 2.

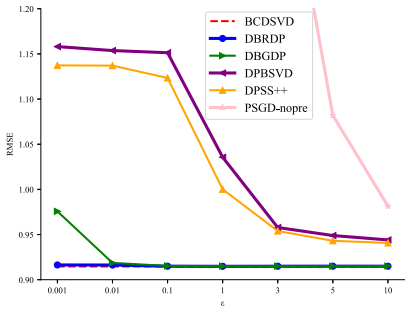
Table 2. Experimental Algorithms

Algorithms	**Description**
BCDSVD	Non-privacy protection
PSGD-nopre	SVD algorithm based on gradient perturbation, without preprocessing
DPSS ++	SVD ++ algorithm based on gradient perturbation
DPBSVD	BiasSVD algorithm based on gradient perturbation
DBGDP	BCDSVD algorithm based on data perturbation,and adds Gaussian random noise
DBRDP	BCDSVD algorithm based on data perturbation, and satisfies (α, ε) - RDP

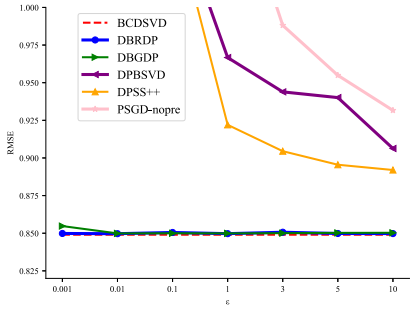
The algorithms listed in Table 2 are trained with filmtrust, movielens100K and movielens1M. We take root mean square error (RMSE) as the experimental evaluation index of various algorithms. The experimental results are shown in Figure 3.



(a) filmtrust



(b) movielens100K



(c) movielens1M

Figure 3. The RMSE of different datasets

From Figure 3, we can see that under different datasets, with the value of privacy budget ε increasing, the RMSE of different algorithms will roughly show a downward trend. Because the larger ε is, the smaller the random noise adds. Besides, the corre-

sponding degree of protection will also decrease, and the recommendation effect of the algorithm will be better.

From Figure 3, we can also see that different models will also have a certain impact on the evaluation indicators. For example, the effect of DPSS ++ is better than DPBSVD. This is because the SVD ++ recommendation model used by DPSS ++ has better performance and higher accuracy than the BiasSVD model used by DPBSVD. The reason is that the SVD ++ model increases the performance of implicit feedback. In this way, the accuracy of the model will be improved to a certain extent and the recommendation performance will be better. From Figure 3, we can see that the algorithms proposed in this paper have a better recommendation effect than several comparison algorithms, and with the increase of privacy budget, The changing trend of RMSE is also relatively stable and the convergence speed is relatively fast. Compared with the comparative experiment, the recommendation effect of the DBRDP algorithm in the three datasets is better and has been improved to some extent, and is closer to the benchmark experiment in this paper than other comparative experiments.

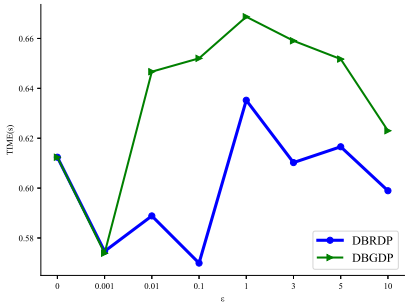
Figure 3 shows that the algorithms proposed in this paper can provide privacy protection for the model on different privacy budgets, and the evaluation index RMSE is smaller than the previously mentioned methods under the same privacy budget, which can provide privacy protection for the model and maintain the recommendation performance of the model.

In addition, to verify whether the proposed method will affect the performance of BCDSVD, this paper also compares the running time of DBGDP-BCDSVD, DBRDP-BCDSVD, and non-noisy BCDSVD models(where $\varepsilon = 0$) to verify whether different privacy budgets will affect the performance of the BCDSVD model. We still repeat the experiment 10 times under the same privacy budget and take the average time of these 10 experiments as the measurement index for comparison. The experimental results are shown in Figure 4.

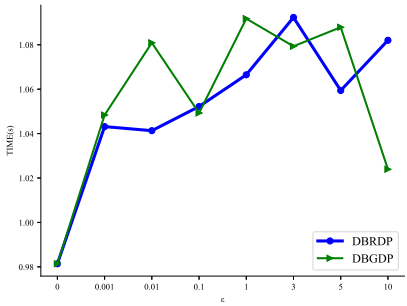
Figure 4 shows that although the algorithm of adding privacy has a certain impact on the running performance of the BCDSVD model under different privacy budgets, it is within an acceptable range. Although in some datasets, the running time of the method proposed in this paper is larger and more unstable than the non-noisy method, the recommended performance of the algorithm proposed in this paper is better, and it can quickly converge to the situation without noise. Therefore, we believe that on the premise of not seriously affecting the recommendation effect of the model, the running speed of the model will be affected and fluctuate under different privacy budgets, but we think such fluctuations are acceptable.

5. Conclusion

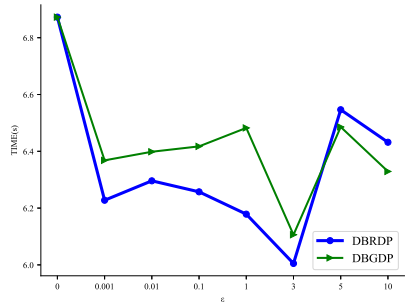
We design two algorithms by using the BCDSVD recommendation model and related differential privacy technology. The first one is data perturbation based Gaussian noise(DBGDP), and the other is data perturbation based Rényi differential privacy(DBRDP). Through corresponding theoretical and experimental analysis, we can conclude that the method proposed in this paper can better protect the model without seriously affecting the model recommendation effect, and can quickly converge to the non-noisy recommendation model. It can not only provide a better recommendation ef-



(a) filmtrust



(b) movielens100K



(c) movielens1M

Figure 4. The running time of different datasets

fect, but also provide privacy protection for the model. The usability and practicability of the model are better.

The main research work in the future is whether we can use the relevant technology of data perturbation to add dynamically allocated random noise to the dataset and apply it to model training without seriously affecting the performance of the model, and obtain better privacy protection.

References

[1] Shi Y, Larson M, Hanjalic A. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges[J]. ACM Computing Surveys (CSUR), 2014, 47(1): 1-45.

[2] Sweeney L. k-anonymity: A model for protecting privacy[J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2002, 10(05): 557-570.

[3] Li N, Li T, Venkatasubramanian S. t-closeness: Privacy beyond k-anonymity and l-diversity. 2007 IEEE 23rd International Conference on Data Engineering. IEEE, 2007: 106-115.

[4] Machanavajjhala A, Kifer D, Gehrke J, et al. l-diversity: Privacy beyond k-anonymity[J]. ACM Transactions on Knowledge Discovery from Data (TKDD), 2007, 1(1): 3-es.

[5] Dwork C. Differential Privacy. International Colloquium on Automata, Languages and Programming. Springer, Berlin, Heidelberg, 2006: 1-12.

[6] McSherry F, Mironov I. Differentially private recommender systems: Building privacy into the netflix prize contenders. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. 2009: 627-636.

[7] Hua J, Xia C, Zhong S. Differentially private matrix factorization. International Conference on Artificial Intelligence. 2015: 1763–1770.

[8] Xian Zhengzheng et al. Collaborative filtering via SVD ++ with differential privacy.Control and Decision.2019.

[9] Yan Shen et al. DynaEgo:Privacy-reserving collaborative filtering recommender system based on social-Aware differential privacy.Proc of the Int Conf on Information and Communications Security. Binlin:Springer,2016:347-357.

[10] Zheng et al. Differential privacy matrix factorization recommendation algorithm fusing tag similarity. Applicatiion Research of Computers. 2020.

[11] Friedman A,Berkovsky S,Kaafar M A. A differential privacy framework for matrix factorization recommender systems[J].User Modeling and User-Adapted Interaction,2016,26(5):1-34.

[12] Wang J, Wang A. An Improved Collaborative Filtering Recommendation Algorithm Based on Differential Privacy. 2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS). IEEE, 2020: 310-315.

[13] K.Chaudhuri,A.Sarwate,andK.Sinha.Near-optimal differentially private principal components. In Advances in Neural Information Processing Systems,pages 989–997,2012.

[14] Kang et al, "Differentially Private ERM Based on Data Perturbation " . (2020). https: // arxiv.org/abs/2002.08578.

[15] Xu Y, Yin W. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM J Imag Sci, 2013, 6: 1758–1789

[16] Paterek A. Improving regularized singular value decomposition for collaborative filtering. Proceedings of KDD cup and workshop. 2007, 2007: 5-8.

[17] Cai et al. Efficient solution of the SVD recommendation model with implicit feedback.SciSinInform,2020,50:1544–1558.

[18] Martin Abadi,Andy Chu,Ian Goodfellow,H Brendan McMahan,Ilya Mironov,Kunal Talwar,and Li Zhang.2016.Deep learning with differential privacy.In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security.308–318.

[19] Mironov I . Rényi Differential Privacy. 2017 IEEE 30th Computer Security Foundations Symposium (CSF). IEEE, 2017.