Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 1460234, 11 pages
https://doi.org/10.1155/2019/1460234

Research Article

A Differential Privacy Framework for Collaborative Filtering

Jing Yang[1], **Xiaoye Li**[1,2], **Zhenlong Sun**[1,2] and **Jianpei Zhang**[1]

[1]*College of Computer Science and Technology, Harbin Engineering University, Harbin, Heilongjiang 150001, China*
[2]*College of Computer and Control Engineering, Qiqihar University, Qiqihar, Heilongjiang 161006, China*

Correspondence should be addressed to Xiaoye Li; xiaoyeli@hrbeu.edu.cn

Received 18 October 2018; Revised 23 November 2018; Accepted 18 December 2018; Published 9 January 2019

Academic Editor: A. M. Bastos Pereira

Focusing on the privacy issues in recommender systems, we propose a framework containing two perturbation methods for differentially private collaborative filtering to prevent the threat of inference attacks against users. To conceal individual ratings and provide valuable predictions, we consider some representative algorithms to calculate the predicted scores and provide specific solutions for adding Laplace noise. The DPI (Differentially Private Input) method perturbs the original ratings, which can be followed by any recommendation algorithms. By contrast, the DPM (Differentially Private Manner) method is based on the original ratings, which perturbs the measurements during implementation of the algorithms and releases the predicted scores. The experimental results showed that both methods can provide valuable prediction results while guaranteeing DP, which suggests it is a feasible solution and can be competent to make private recommendations.

1. Introduction

In the Internet age, users are constantly troubled by information overload, since they cannot get really useful parts of large amounts of information. As a promising solution, recommender systems with personalized technologies have been widely used to enhance user experience in various online services. The typical case is that Netflix has been working on recommending movies which are best suitable for the users' taste. Collaborative filtering (CF for short) is one of the most dominant techniques used in recommender systems. The basic idea is to predict user preference based on preferences of other similar users. The methods are generally divided into two classes, the memory-based methods and the model-based methods [1]. However, users' rating data collected for recommendation is a potential source of leaking privacy for inferring users' sensitive information [2]. Calandrino et al. [3] developed the algorithms to demonstrate several inference attacks from continual recommendations with auxiliary information. In this work, we focus on the privacy issues in recommender systems and seek feasible solutions based on differential privacy (DP for short) [4], which is recognized as a promising technology for the privacy framework.

Zhu et al. [5] proposed a private neighbor collaborative filtering algorithm consisting of two major steps. Firstly, a redesigned exponential mechanism is used to privately select neighbors with higher quality to enhance the performance of the recommendations. The involved recommendation-aware sensitivity is a new sensitivity based on the notion of local sensitivity. Then, the original ratings of the selected neighbors are perturbed by adding Laplace noise. Zhu et al. [6] designed two differentially private algorithms with sampling, named DP-IR and DP-UR for item and user based recommendation, respectively. Both algorithms use the exponential mechanism with a carefully designed quality function. Jorgensen et al. [7] proposed a privacy preserving framework for personalized social recommendations. There are two distinct graphs in the model settings, an unweighted preference graph and an insensitive social graph. The users are clustered according to natural community structure of the social network, which significantly reduces the amount of noise required to guarantee DP. However, relationships in social networks are sometimes considered sensitive information.

Friedman et al. [8] proposed a generic framework and evaluated several ways of differentially private matrix factorization for recommender systems. The specific methods are input perturbation, stochastic gradient perturbation and ALS

with output perturbation. Through comparison and analysis, the input perturbation performs best in the recommendation results. Mcsherry et al. [9] adapted several leading algorithms in the Netflix Prize competition to the framework of DP. Concretely, the Laplace noise is incorporated into various global effects and covariance matrix of user rating vectors based on item-item similarities. Given these noisy measurements, several algorithms (the k-Nearest Neighbor method [10] and the standard SVD-based prediction mechanism) are employed to make private recommendations directly. Liu et al. [11] proposed a hybrid approach for privacy preserving recommender system to hide users' private data and prevent privacy inference. The users' original data are disguised through randomized perturbation (RP for short). Similar to literature [9], covariance matrix and some averages are masked with a particular amount of noise again to achieve DP. Then, some existing algorithms can run directly on the published noisy measurements.

Differently from the works in literature [5, 6], we choose to calculate the predicted scores by using all users' ratings, not the recommended list, which can make full use of data information for the estimation of noise error. In literature [6], the theoretical results are presented in detail on both privacy and accuracy of the proposed method, however, the experimental results are lacking. By comparison, more experimental results are provided in this work to demonstrate the relationship between privacy and accuracy. In this work, the similarity is calculated by row vectors in the rating matrices. That is, the calculation is based on user-user similarities, not item-item similarities as in literature [9]. This mainly takes into account the recommendation based on users with the similar preferences. Furthermore, the experimental results of this paper showed that the DPI method performs better than the DPM method, which is consistent with the conclusion in literature [8].

In this paper, we propose a differential privacy framework for collaborative filtering, which includes three existing algorithms to calculate the predicted scores and adopts two methods of adding Laplace noise to conceal individual ratings and provide valuable prediction results. The rest of the paper is organized as follows. Section 2 introduces the background knowledge. A detailed description of the framework is presented in Section 3. Section 4 reports the experimental evaluations. Finally, Section 5 concludes the study and provides further research directions.

2. Background

2.1. Differential Privacy. DP offers a mathematical definition of privacy and a provable privacy guarantee for each record in the dataset. Intuitively, the output of the computation should not reveal too much information about any record in the dataset. The probability of the output is insensitive to small input changes, whether one record is in the dataset or not. DP is presented in a series of papers [12–16], mainly used in data publishing [17–19] and data mining [20–22].

Definition 1 (ε-DP [4]). A randomized computation K satisfies ε-DP if, for any neighboring datasets A and B differing on at most one record, and for all subsets of possible outputs $S \subseteq Range(K)$

$$\Pr[K(A) \in S] \leq \exp(\varepsilon) \times \Pr[K(B) \in S] \tag{1}$$

where ε is the privacy budget to make the trade-off between privacy and accuracy. The value of ε is generally set to a small positive value. The smaller it is, the higher privacy and lower accuracy it provides and vice versa.

Specifically, the neighboring datasets contain the same ratings except for one in this context. The rating r_u^i in A that a user u assigns to an item i is different from r'^i_u in B.

The noise mechanism is suitable for perturbing the numerical outputs, which is one of the common ways to achieve DP. The amount of noise required is dependent on global sensitivity of the function. There are important combination properties in differentially private algorithms. Formally, the relevant definitions and propositions are described as follows.

Definition 2 (global sensitivity [4]). For a function $f: D \longrightarrow R^d$, the global sensitivity of f is

$$\triangle f = \max_{D,D'} \left\| f(D) - f\left(D'\right) \right\|_1 \tag{2}$$

where R^d is a real vector of d dimensions and D and D' are neighboring datasets. The global sensitivity denotes the maximum extent that a single record could affect the output results.

Definition 3 (Laplace mechanism [23]). For a function $f: D \longrightarrow R^d$, the randomized algorithm M satisfies ε-DP if

$$M(D) = f(D) + \langle X_1, X_2, \ldots, X_d \rangle \tag{3}$$

where $X_i \sim Lap(\Delta f/\varepsilon), i = 1, 2, \ldots, d$, which are *i.i.d.* random variables sampled from the Laplace distribution with mean 0 and scale parameter $\Delta f/\varepsilon$.

Proof (see [24]). Suppose $f(\mathrm{D}) = \langle a_1, a_2, \ldots, a_d \rangle$. For any output $\mathrm{t} = \langle \mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t}_\mathrm{d} \rangle$,

$$\begin{aligned}
&\Pr[\mathrm{M}(\mathrm{D}) = \mathrm{t}] = \Pr\left[f(D) + \langle X_1, X_2, \ldots, X_d \rangle = \mathrm{t}\right] \\
&= \Pr\left[\langle a_1, a_2, \ldots, a_d \rangle + \langle X_1, X_2, \ldots, X_d \rangle \right. \\
&\quad \left. = \langle \mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t}_\mathrm{d} \rangle\right] \\
&= \Pr\left[\left(X_1 = \mathrm{t}_1 - a_1\right) \bigwedge \left(X_2 = \mathrm{t}_2 - a_2\right)\right] \bigwedge \\
&\quad \ldots \bigwedge \left(X_d = \mathrm{t}_\mathrm{d} - a_d\right)\Big]
\end{aligned}$$

$$= \prod_{i=1}^{d} \Pr\left[X_i = t_i - a_i\right]$$
$$= \prod_{i=1}^{d} \frac{\varepsilon}{2\Delta f} \exp\left(\frac{-\varepsilon \left|t_i - a_i\right|}{\Delta f}\right)$$
$$= \left(\frac{\varepsilon}{2\Delta f}\right)^{d} \exp\left(\frac{-\varepsilon \sum_{i=1}^{d} \left|t_i - a_i\right|}{\Delta f}\right)$$
$$= \left(\frac{\varepsilon}{2\Delta f}\right)^{d} \exp\left(\frac{-\varepsilon \left\|t - f(D)\right\|_1}{\Delta f}\right) \tag{4}$$

Similarly, $\Pr[M(D') = t] = (\varepsilon/2\Delta f)^d \exp(-\varepsilon\|t - f(D')\|_1/\Delta f)$. Thus,

$$\frac{\Pr[M(D) = t]}{\Pr[M(D') = t]} = \frac{\exp\left(-\varepsilon \left\|t - f(D)\right\|_1 / \Delta f\right)}{\exp\left(-\varepsilon \left\|t - f(D')\right\|_1 / \Delta f\right)}$$
$$= \exp\left(\frac{\varepsilon\left(\left\|t - f(D')\right\|_1 - \left\|t - f(D)\right\|_1\right)}{\Delta f}\right) \tag{5}$$
$$\leq \exp\left(\frac{\varepsilon\left(\left\|f(D) - f(D')\right\|_1\right)}{\Delta f}\right) \leq \exp(\varepsilon)$$

□

Proposition 4 (sequential composition [25]). *Let each algorithm A_i provide ε_i-DP. The combination algorithm A (A_1 (D), A_2 (D),…, A_k (D)) over the dataset D provides $\sum_{i=1}^{k} \varepsilon_i$- DP.*

Proof (see [24]). For any output $t = (t_1, t_2, \ldots, t_k)$,

$$\Pr[A(D) = t] = \Pr\left[A_1(D) = t_1\right]$$
$$\cdot \Pr\left[A_2(D) = t_2\right] \ldots \Pr\left[A_k(D) = t_k\right]$$
$$\leq e^{\varepsilon_1} \Pr\left[A_1(D') = t_1\right]$$
$$\cdot e^{\varepsilon_2} \Pr\left[A_2(D') = t_2\right] \ldots e^{\varepsilon_k} \Pr\left[A_k(D') = t_k\right] \tag{6}$$
$$= e^{\varepsilon_1 + \varepsilon_2 + \cdots + \varepsilon_k} \Pr\left[A(D') = t\right]$$

□

Proposition 5 (parallel composition [25]). *Let each algorithm A_i provide ε_i-DP. The combination algorithm A (A_1 (D_1), A_2 (D_2),…, A_k (D_k)) over the disjoint subsets of dataset D provides* $\max_{i \in [1,\ldots,k]} \varepsilon_i$*-DP.*

Proof (see [24]). Without loss of generality, assume that D_j differs one element from D_j'; other subsets are exactly the same. For any output $t = (t_1, t_2, \ldots, t_k)$,

$$\Pr[A(D) = t] = \Pr\left[\left(A_1(D_1) = t_1\right) \bigwedge \left(A_2(D_2) = t_2\right) \bigwedge \ldots \bigwedge \left(A_k(D_k) = t_k\right)\right]$$
$$= \Pr\left[A_j(D_j) = t_j\right] \prod_{i \neq j} \Pr\left[A_i(D_i) = t_i\right]$$
$$\leq e^{\varepsilon_j} \Pr\left[A_j(D_j') = t_j\right] \prod_{i \neq j} \Pr\left[A_i(D_i') = t_i\right] \tag{7}$$
$$\leq e^{\max_{i \in [1,\ldots,k]} \varepsilon_i} \Pr\left[A(D') = t\right]$$

In privacy preserving computations, the measurements need to be allocated a reasonable privacy budget ε_i based on the combination properties. □

2.2. Collaborative Filtering. The CF system usually presents a sorted list of predicted items to the active user. The performance can be evaluated by the expected utility of the items in the recommended list. Alternatively, the system may provide numeric scores directly for the predicted items. The performance can be measured by a normed distance between the predicted scores and the actual preference values (i.e., the original ratings). There are two common ways of calculating the similarity, Pearson correlation coefficient (Pcc) and Cosine-based similarity (Cos) [26].

(1) Pcc is a linear correlation coefficient ρ_{XY}, which is used to measure the correlation between two random variables X and Y. The range of values is between -1 and 1, and the larger the absolute value is, the stronger the correlation is. When X is linearly dependent with Y, the value is 1 (namely, positive linear correlation) or -1 (namely, negative linear correlation). It is defined as the ratio of covariance and standard deviation between X and Y, where E is the expected value.

$$\rho_{XY} = \frac{\mathrm{Cov}(X, Y)}{\sqrt{D(X)}\sqrt{D(Y)}} = \frac{E((X - EX)(Y - EY))}{\sqrt{D(X)}\sqrt{D(Y)}} \tag{8}$$

(2) Cos evaluates the similarity by calculating the angle cosine of two vectors X and Y, which pays more attention to the difference in directions between the vectors. The range of values is between -1 and 1, and the value of 0 denotes that two vectors are orthogonal. When the directions of two vectors coincide, the angle cosine takes the maximum value of 1, and vice versa. The vector similarity is defined as follows:

$$sim(X, Y) = \cos\theta = \frac{\vec{x} \bullet \vec{y}}{\|x\| \bullet \|y\|} = \frac{\sum_1^n (X_i \times Y_i)}{\sqrt{\sum_1^n X_i^2} \times \sqrt{\sum_1^n Y_i^2}} \tag{9}$$

3. The Proposed Method

DP is essentially a property that the system should maintain, rather than a specific way of calculating. Therefore, the framework designed includes different perturbation methods to carry out predictions in a differentially private manner. The

Input: the rating matrix, the privacy budget ε, the parameter k
Output: the predicted scores
1. $\varepsilon = \varepsilon_1 + \varepsilon_2$
2. simMat = similarity (activeMatTrain, otherMatTrain)
3. simMat = simMat + Lap $(2/\varepsilon_1)$
4. otherAvgVec = mean (otherMat^T) + Lap $(\Delta r/(k * \varepsilon_2))$
5. otherMatPred = otherMatPred - otherMatAvg
6. for each active user j
7. activeAvg = mean (activeMatTrain (j)) + Lap $(\Delta r/(k * \varepsilon_2))$
// The spdiags function transforms the similarity vector of user j into a diagonal matrix.
8. activeMatPred (j) = sum (spdiags (simMat (j))∗otherMatPred) + activeAvg
9. end for
10. return activeMatPred

Algorithm 1: The perturbed Pcc (or Cos) algorithm.

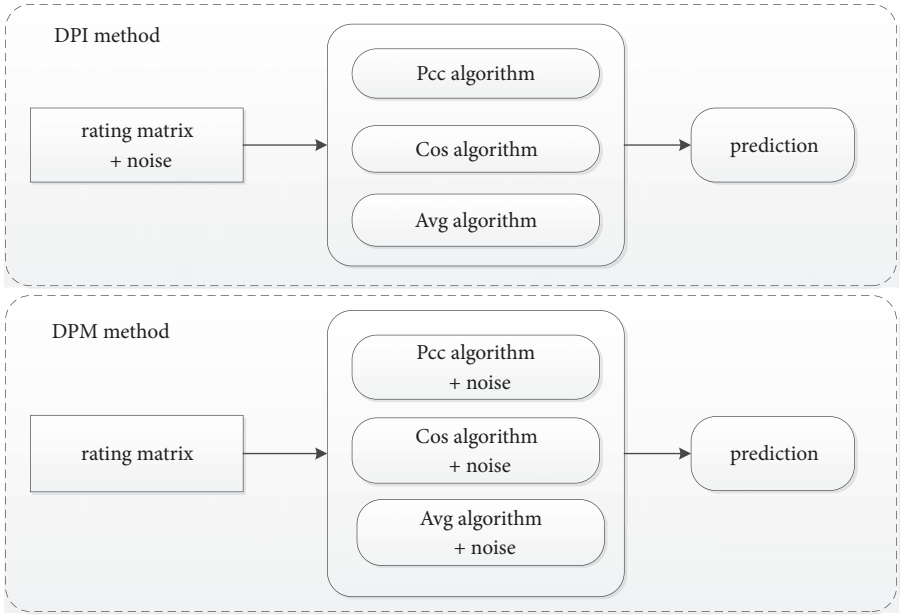

Figure 1: The framework for private predictions.

simple technique of noise addition is fully fit for predicting the scores while protecting original ratings without leakage.

As shown in Figure 1, the framework includes three different algorithms from the previous research [1]. The calculation methods are similar in the algorithms Pcc and Cos except for different measurements of similarities between users. For a detailed description, see Algorithm 1. The Avg algorithm makes use of the average of original ratings to predict the scores. According to the locations of adding noise, the perturbation methods are divided into two forms, respectively. In the DPI method, the noise is added to each entry in the rating matrix to mask the original data. In the DPM method, the noise is added to the various measurements of the algorithms based on the original rating matrix.

3.1. DPI Method. In this method, the original ratings are perturbed with Laplace noise before performing the prediction algorithms. The input perturbation, a relatively simple and efficient strategy, can be followed by any recommendation algorithms.

The magnitude of noise added to the rating matrix is according to the global sensitivity of the ratings. The range of ratings is $\Delta r = r_{max} - r_{min}$, which dictates the maximum amount of changes in the ratings. According to Definition 3, the rating matrix will be perturbed by adding noise Lap$(\Delta r/\varepsilon)$. As the postprocessing, the noisy ratings are clamped to $[r_{min}, r_{max}]$ using the following to limit the influence of some excessive noise.

$$r = \begin{cases} r_{min,} & r < r_{min} \\ r, & r_{min} \le r \le r_{max} \\ r_{max,} & r > r_{max} \end{cases} \tag{10}$$

These algorithms take the noisy matrix as inputs to carry out predictions without accessing the original matrix.

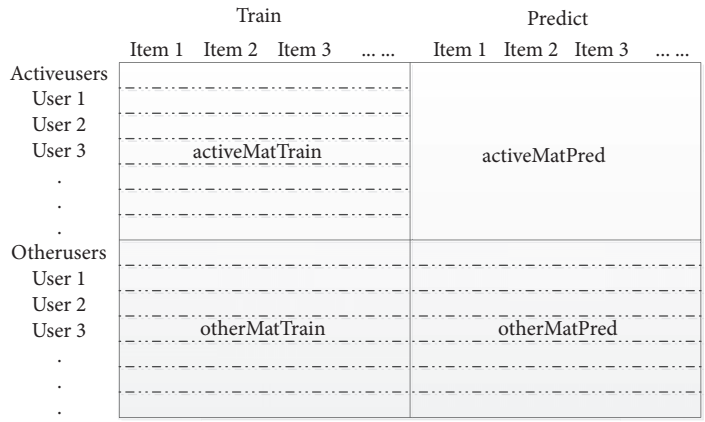

Figure 2: The partitioning of the rating matrix.

Therefore, the whole process of the DPI method can guarantee ε-DP.

3.2. DPM Method. In this method, privacy protection needs to be guaranteed during the implementation of the algorithms based on the original rating matrix. Note that, the users generally assign ratings to part of the items in the dataset. The algorithms only consider nonzero ratings in the matrix. The constraint is that each user has rated at least k items, which is an important parameter in Algorithm 1. Otherwise, the system will not make predictive recommendations for the user.

3.2.1. Description of Algorithm. As shown in Figure 2, the rating matrix is partitioned into four blocks in Algorithm 1. The users are randomly divided into active users and other users. Meanwhile, a part of items is for training, and the rest are for predicting. The task of prediction is to score nonzero ratings in the activeMatPred.

Algorithm 1 provides a detailed description of adding Laplace noise to the Pcc (or Cos) algorithm.

Line 1 divides the privacy budget ε into two parts (i.e., ε_1 and ε_2) to be consumed in a sequential way. Line 2 calculates the similarity matrix simMat with activeMatTrain and otherMatTrain based on the measurements Pcc (or Cos). Line 3 perturbs simMat by adding noise Lap $(2/\varepsilon_1)$ to each entry in the matrix. Line 4 calculates the averages of the ratings otherAvgVec adding noise Lap $(\Delta r/(k * \varepsilon_2))$ for other users. Line 5 generates the deviation matrix otherMatPred by subtracting otherMatAvg, where each nonzero entry is set to the noisy average. Note that, the coordinates of the nonzero values are one-to-one in otherMatPred and otherMatAvg. Line 7 calculates the average of the ratings activeAvg adding noise Lap $(\Delta r/(k * \varepsilon_2))$ for each active user j. Line 8 generates the predicted scores activeMatPred (j) by adding activeAvg to the predicted deviations calculated with simMat (j) and otherMatPred. Line 10 returns the matrix activeMatPred, where nonzero values are assigned to the predicted scores.

In the Avg algorithm, the users in the rating matrix are no longer divided and the items are still divided into two parts, training and predicting. In other words, the rating matrix is vertically divided into two parts, not four blocks as seen in Figure 2. The average of each user's ratings in the training block is used directly as a predicted score for each nonzero item in the predicting block. The intuitive impression is that the ratings assigned by each user are relatively stable. The average of the ratings is the unique measurement that needs to be perturbed by adding Laplace noise. Therefore, the privacy budget ε no longer needs to be divided, and the magnitude of noise added is Lap $(\Delta r/(k * \varepsilon))$. The description of the algorithm is omitted here, as it is relatively simple and easy to implement.

3.2.2. Analysis of Privacy

Theorem 6. *The global sensitivity of the average of the ratings is $\Delta r/k$.*

Proof. The global sensitivity is

$$\begin{aligned} \Delta avg &= \Delta\left(\frac{r_1 + r_2 + \cdots + r_n}{n}\right) \\ &= \frac{\Delta\left(r_1 + r_2 + \cdots + r_n\right)}{n} = \frac{\Delta r}{n} \leq \frac{\Delta r}{k} \end{aligned} \tag{11}$$

where n is assumed to be constant and $n \geq k$ and $\Delta r = r_{max} - r_{min}$. For simplicity, $\Delta r/k$ is used as a specified upper limit of the sensitivity. Therefore, the magnitude of noise added to the average of the ratings is Lap $(\Delta r/(k * \varepsilon))$. □

Theorem 7. *The proposed Algorithm 1 guarantees ε-DP.*

Proof. (1) The value of Pcc (or Cos) ranges from -1 to 1, and the maximum amount of changes is 2. According to Definition 3, the magnitude of noise added to line 2 is Lap $(2/\varepsilon_1)$. (2) According to Theorem 6 and Proposition 5, the magnitude of noise added to line 4 and line 7 is Lap $(\Delta r/(k * \varepsilon_2))$. (3) The measurements to calculate activeMatPred in line 8 are completely perturbed by Laplace noise. According to Proposition 4, the proposed Algorithm 1 guarantees ε-DP. □

Theorem 8. *The perturbed Avg algorithm guarantees ε-DP.*

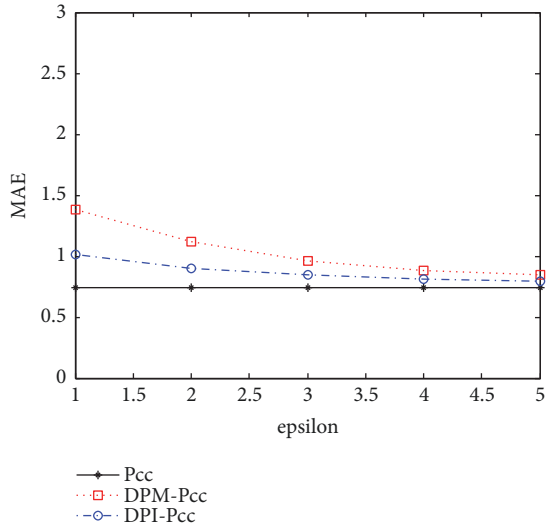

Figure 3: MAE of DP-Pcc on the ml-latest-small dataset.

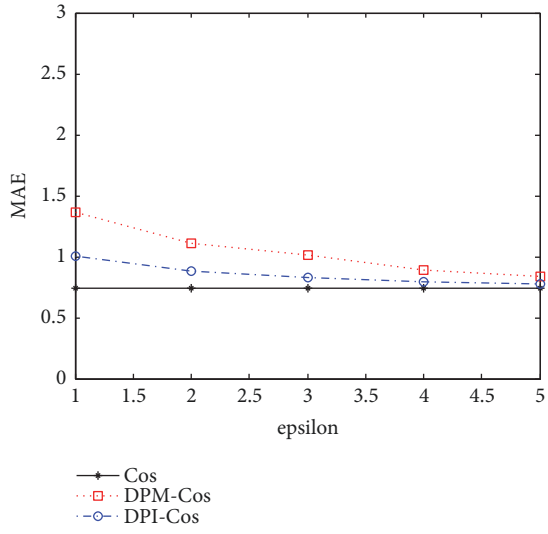

Figure 4: MAE of DP-Cos on the ml-latest-small dataset.

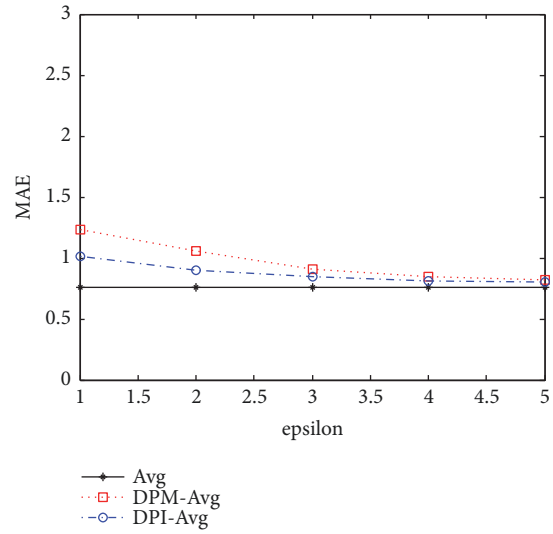

Figure 5: MAE of DP-Avg on the ml-latest-small dataset.

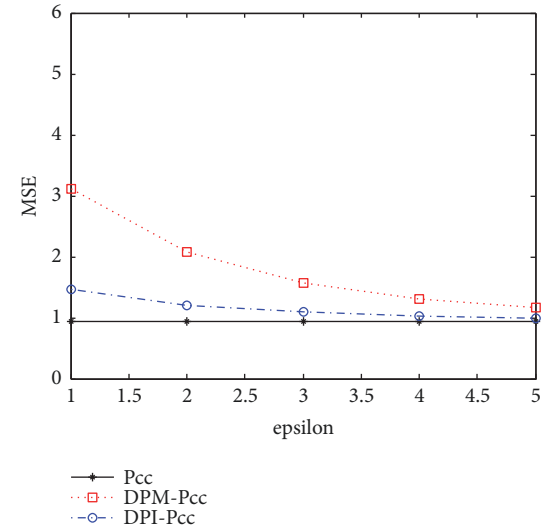

Figure 6: MSE of DP-Pcc on the ml-latest-small dataset.

Proof. As the predicted score, the average of the ratings is directly perturbed by adding Laplace noise. According to Theorem 6, the magnitude of noise added to the average of the ratings is Lap $(\Delta r/(k * \varepsilon))$, and the perturbed Avg algorithm guarantees ε-DP. □

4. Experiments

In this section, we first conduct an experiment to provide an instance of the Laplace noise, as shown in Table 1. The first column contains original counts that need to be perturbed, the second column contains the generated random data, the others contain the generated noise corresponding to different ε. The last row calculates sum of squared error (SSE) of the noise in each column. In the vertical direction, when r is around 0.5, the generated noise is minimal, and the further r goes, the higher the noise. In the horizontal direction, the noise decreases with the increase of ε, so does SSE of the noise, and the level of privacy is naturally lower.

Then, we conduct experiments on three new datasets [27], ml-20m, ml-latest-small and ml-latest, which are collected and made available rating datasets from the MovieLens website by GroupLens Research. Respectively, we randomly selected 700 users from ml-latest-small, 1000 users from ml-20m, 2000 users, and 5000 users from ml-latest. Note that some users will be removed in the preprocessing if they have rated less than 3 items, that is, $k = 3$ in the experiments. The actual size of original and experimental datasets is shown in Table 2. The ratings in the datasets are $\{0.5, 1.0, 1.5, 2.0, 2.5, \ldots, 5.0\}$, respectively, that is, the global sensitivity $\Delta r = 4.5$. The privacy budget ε is set to $\{1, 2, 3, 4, 5\}$, respectively. In Algorithm 1, we set $\varepsilon_1 : \varepsilon_2 = 0.2 : 0.8$. Meanwhile, the selected datasets are divided equally in both horizontal and vertical directions. The evaluation

Table 1: The noise of Laplace mechanism.

counts	random r	ε =0.1	ε =0.5	ε =ln 2	ε =1	ε =ln 3
44	0.8633	12.9718	2.5944	1.8714	1.2972	1.1807
12	0.344	-3.7392	-0.7478	-0.5395	-0.3739	-0.3404
38	0.4569	-0.9018	-0.1804	-0.1301	-0.0902	-0.0821
76	0.2381	-7.4196	-1.4839	-1.0704	-0.742	-0.6754
92	0.7712	7.816	1.5632	1.1276	0.7816	0.7114
SSE of the noise		299.203	11.9681	6.2275	2.992	2.479

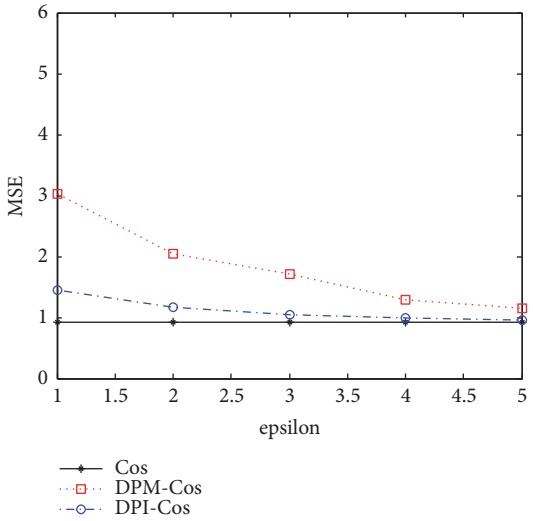

Figure 7: MSE of DP-Cos on the ml-latest-small dataset.

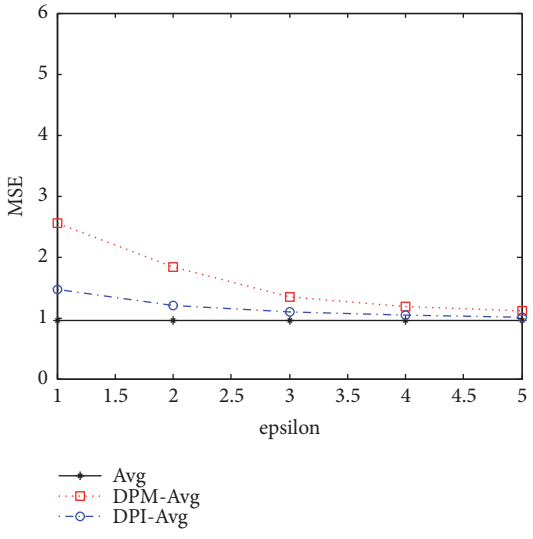

Figure 8: MSE of DP-Avg on the ml-latest-small dataset.

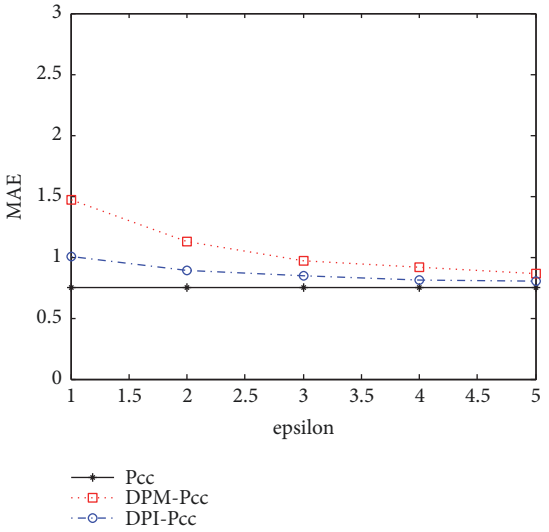

Figure 9: MAE of DP-Pcc on the ml-20m (1000) dataset.

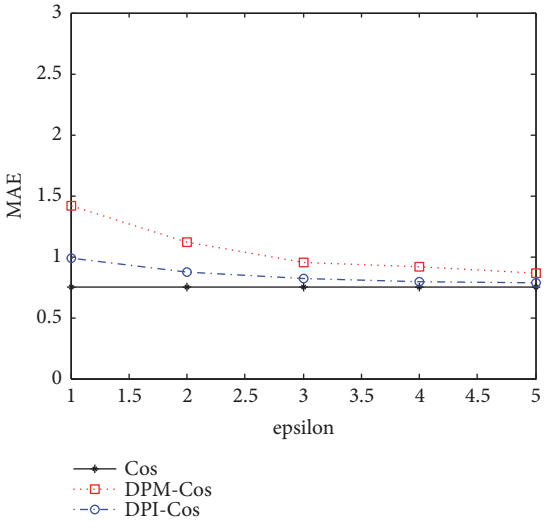

Figure 10: MAE of DP-Cos on the ml-20m (1000) dataset.

measurements are mean absolute error (MAE) and mean square error (MSE), respectively. $MAE = mean|predval - trueval|$ and $MSE = mean(predval - trueval)^2$, where *predval* is the predicted score and *trueval* is the original rating.

As shown in Figures 3–14, three algorithms run on two smaller datasets in two perturbation methods. In summary, the DPI method performs better than the DPM method especially when evaluating MSE. MAE of nonprivate method is around 0.75 and MSE of nonprivate method is around 1. When $\varepsilon = 1$, MAE of DPM is slightly less than 1.5 and MAE of DPI is around 1. Thus, the error ratio between two DP methods is probably less than 1.5. When $\varepsilon = 1$, MSE of DPM

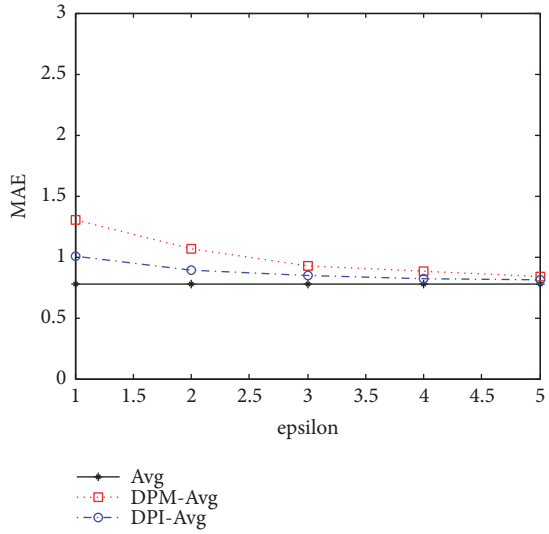

Figure 11: MAE of DP-Avg on the ml-20m (1000) dataset.

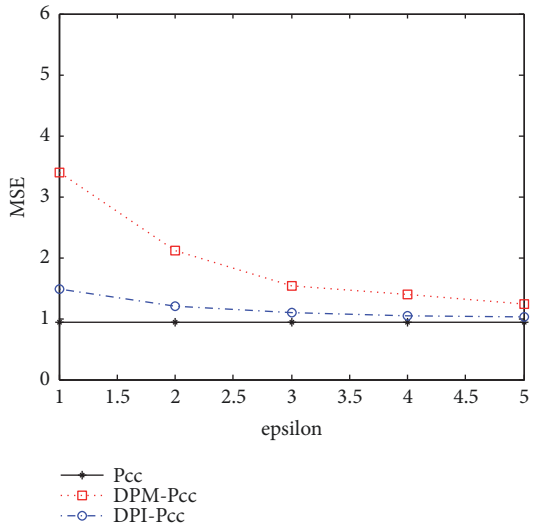

Figure 12: MSE of DP-Pcc on the ml-20m (1000) dataset.

Table 2: The size of datasets.

Datasets	User × item (original)	User × item (experimental)
ml-latest-small	700 × 9000	671 × 4801
ml-20m(1000)	138,000 × 27,000	1000 × 5870
ml-latest(2000)	270,000 × 45,000	1907 × 6144
ml-latest(5000)	270,000 × 45,000	4773 × 8972

is around 3 and MSE of DPI is around 1.5. Thus, the error ratio between two DP methods is approximately 2. When ε =2, the error curves of the DP methods decrease greatly. When ε ≥3, these error curves keep going down steadily. When weaker privacy guarantee is acceptable, that is ε = 5, the prediction accuracy of the DP methods can get close to that of the nonprivate method. The errors decrease with the increase of ε, which is consistent with the analysis of noise data in Table 1.

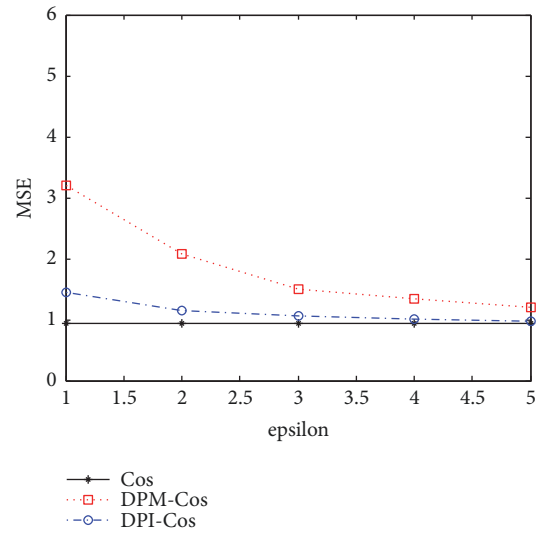

Figure 13: MSE of DP-Cos on the ml-20m (1000) dataset.

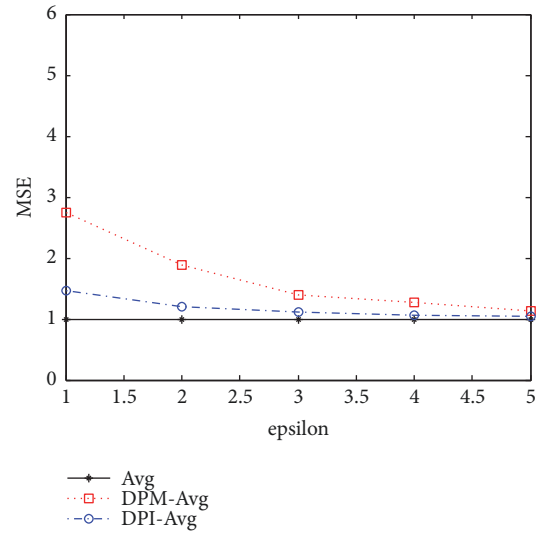

Figure 14: MSE of DP-Avg on the ml-20m (1000) dataset.

The experimental results of three algorithms are similar, and DPM-Avg has a slightly better effect than other DPM algorithms. This is mainly due to the noise error caused by perturbing the similarity matrix in Algorithm 1. Sequentially, the algorithms Pcc and Avg continue to run on two larger datasets, and the experimental results are shown in Figures 15–22. When ε = 1, MAE of DPI is slightly greater than 1 and MSE of DPI is close to 2. Thus, the error of DPI gets a little larger at this point. Beyond that, the trend of all error curves remains almost unchanged. Although four datasets of different sizes are tested, the number of selected users has little influence on the prediction errors. In conclusion, both perturbation methods enable better scalability and obtain consistent experimental results for larger datasets. The proposed framework is feasible and competent to make predictive recommendations while guaranteeing DP.

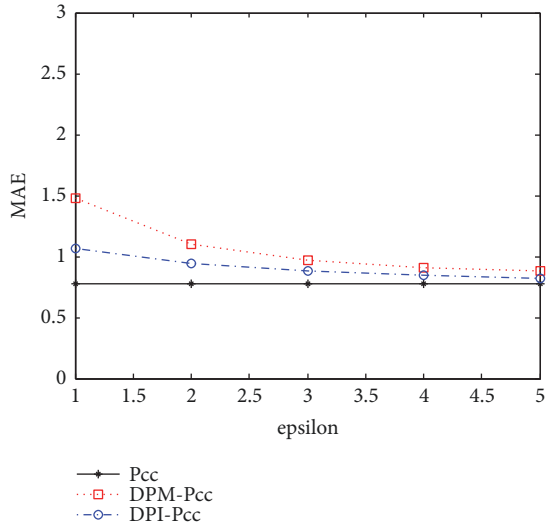

Figure 15: MAE of DP-Pcc on the ml-latest (2000) dataset.

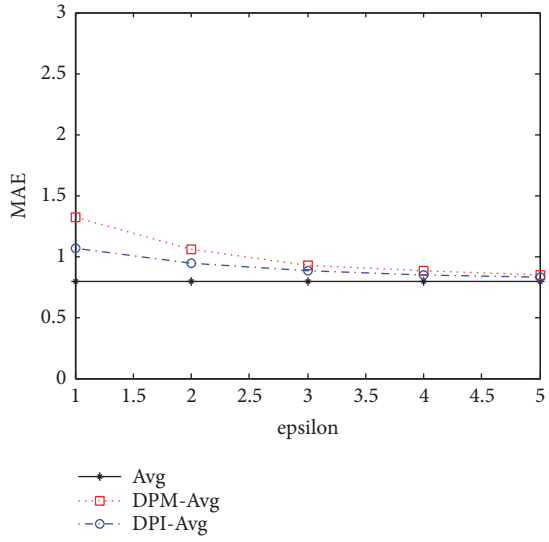

Figure 16: MAE of DP-Avg on the ml-latest (2000) dataset.

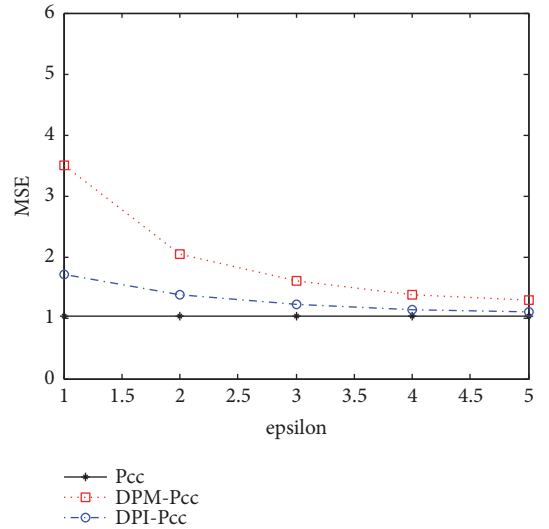

Figure 17: MSE of DP-Pcc on the ml-latest (2000) dataset.

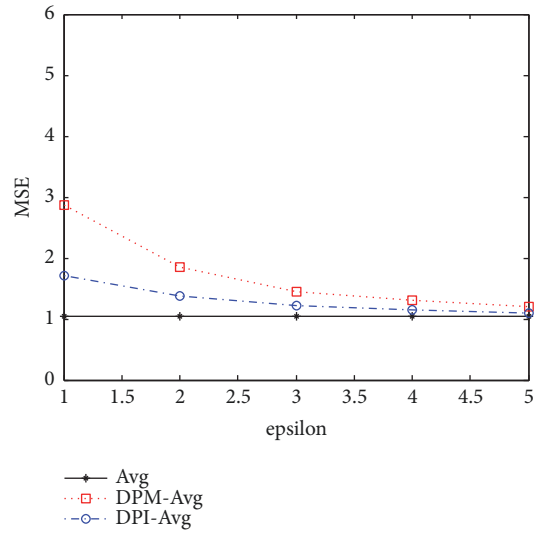

Figure 18: MSE of DP-Avg on the ml-latest (2000) dataset.

5. Conclusion

In this paper, we addressed the problem of differentially private collaborative filtering based on existing algorithms. The proposed framework includes two perturbation methods by adding Laplace noise. The solution is actually a way of dealing with data, which may be applicable to other more advanced CF algorithms. However, the potential problem is that privacy budget ε may be larger, which may make the promised privacy protection ineffective. The experimental results showed that the number of selected users has little influence on the prediction errors. We can consider analyzing the problem from different perspectives. On the one hand, it seems enough to take a small amount of data to calculate such predictive scores. And on the other, the amount of noise required could be tolerable when the number of users grows. Therefore, the proposed framework is feasible and competent to make differentially private recommendations.

For commercial applications, it is worthy to study more valuable recommender system for social services. We will explore how to make predictive recommendations combining three degrees of influence, which is a strong connection that can trigger users' behaviors in the social networks. To address privacy concerns, we consider employing local differential privacy [28], a stronger variant, to eliminate users' worries in data collections. It dates back to randomized response technique (RRT for short) [29], which can provide plausible deniability for the individual ratings. The collected datasets then could be used without extra privacy protection. Finally, the performance of new recommender system requires a lot of experiments for further verification.

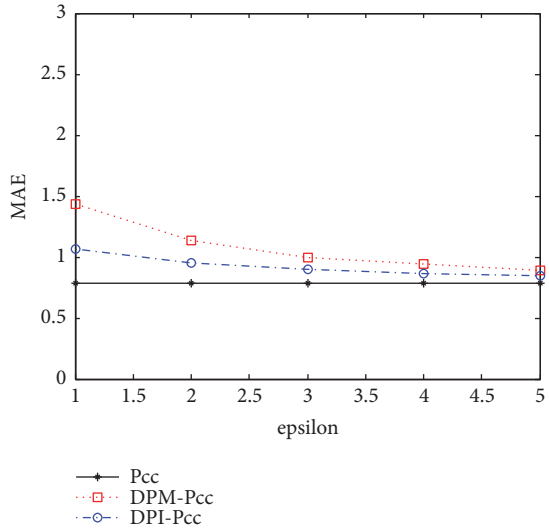

Figure 19: MAE of DP-Pcc on the ml-latest (5000) dataset.

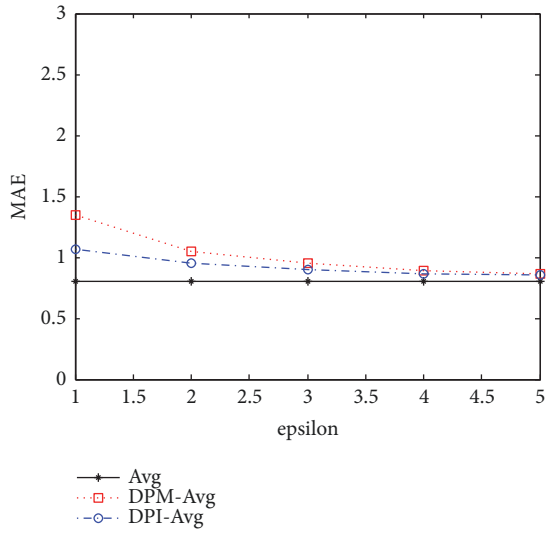

Figure 20: MAE of DP-Avg on the ml-latest (5000) dataset.

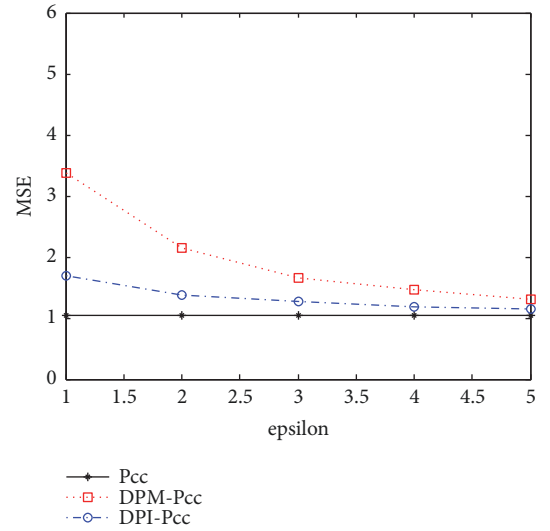

Figure 21: MSE of DP-Pcc on the ml-latest (5000) dataset.

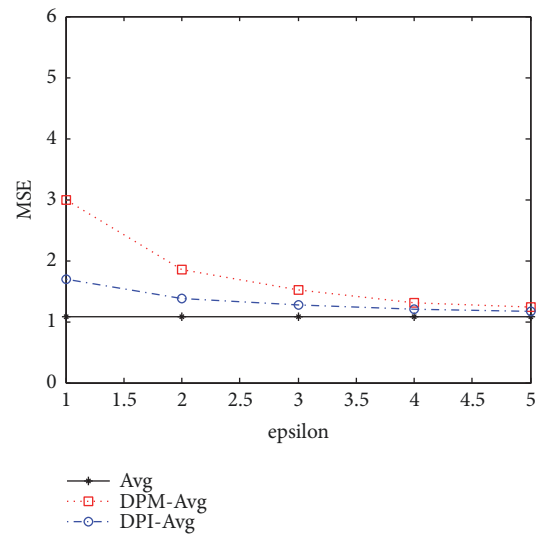

Figure 22: MSE of DP-Avg on the ml-latest (5000) dataset.

Data Availability

The data used to support the findings of this study can be accessed from https://grouplens.org/datasets/movielens/ without any restrictions.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was partially supported by National Natural Science Foundation of China (Nos. 61672179, 61370083, and 61402126), Natural Science Foundation of Heilongjiang Province (No. F2015030), Youth Science Fund of Heilongjiang Province (Nos. QC2016083, QC2017079), Postdoctoral Fellowship of Heilongjiang Province (No. LBH - Z14071), and the Fundamental Research Funds in Heilongjiang Provincial Universities (Nos. 135109245, 135109314).

References

[1] S. Breese John, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pp. 43–52, 1998.

[2] A. J. Jeckmans, M. Beye, Z. Erkin, P. Hartel, R. L. Lagendijk, and Q. Tang, "Privacy in Recommender Systems," in *Social Media Retrieval*, Computer Communications and Networks, pp. 263–281, Springer London, London, 2013.

[3] J. A. Calandrino, A. Kilzer, A. Narayanan, E. W. Felten, and V. Shmatikov, ""You might also like:" Privacy risks of collaborative

filtering," in *Proceedings of the 2011 IEEE Symposium on Security and Privacy, SP 2011*, pp. 231–246, USA, May 2011.

[4] C. Dwork, "Differential privacy," in *Automata, Languages and Programming*, vol. 4052 of *Lecture Notes in Computer Science*, pp. 1–12, Springer, Berlin, Germany, 2006.

[5] T. Zhu, Y. Ren, W. Zhou, J. Rong, and P. Xiong, "An effective privacy preserving algorithm for neighborhood-based collaborative filtering," *Future Generation Computer Systems*, vol. 36, pp. 142–155, 2014.

[6] X. Zhu and Y. Sun, "Differential privacy for collaborative filtering recommender algorithm," in *Proceedings of the 2nd ACM International Workshop on Security and Privacy Analytics (IWSPA '16)*, pp. 9–16, New Orleans, La, USA, March 2016.

[7] Z. Jorgensen and T. Yu, "A privacy-preserving framework for personalized, social recommendations," in *Proceedings of the 17th International Conference on Extending Database Technology, EDBT 2014*, pp. 571–582, Greece, March 2014.

[8] A. Friedman, S. Berkovsky, and M. A. Kaafar, "A differential privacy framework for matrix factorization recommender systems," *User Modeling and User-Adapted Interaction*, vol. 26, no. 5, pp. 425–458, 2016.

[9] F. McSherry and I. Mironov, "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 (KDD '09)*, pp. 627–635, Paris, France, July 2009.

[10] R. M. Bell and Y. Koren, "Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights," in *Proceedings of the Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pp. 43–52, Omaha, NE, USA, October 2007.

[11] X. Liu, A. Liu, X. Zhang et al., "When Differential Privacy Meets Randomized Perturbation: A Hybrid Approach for Privacy-Preserving Recommender System," in *Database Systems for Advanced Applications*, vol. 10177 of *Lecture Notes in Computer Science*, pp. 576–591, Springer International Publishing, Berlin, Germany, 2017.

[12] C. Dwork, "Differential privacy: a survey of results," *International Conference on Theory and Applications of MODELS of Computation*, vol. 4978, pp. 1–19, 2008.

[13] C. Dwork, "The differential privacy frontier," in *Proceedings of Theory of Cryptography Conference*, pp. 496–502, Springer, Berlin Heidelberg, 2009.

[14] C. Dwork and J. Lei, "Differential privacy and robust statistics," in *ACM Symposium on Theory of Computing*, pp. 371–380, ACM, 2009.

[15] C. Dwork, "Differential privacy in new settings," pp. 174–183, SIAM, Philadelphia, PA.

[16] C. Dwork, "The Promise of Differential Privacy: A Tutorial on Algorithmic Techniques," in *Annual Symposium on Foundations of Computer Science*, vol. 4052 of *Lecture Notes in Computer Science*, pp. 1–12, Springer, Berlin, Germany, 2011.

[17] L. Qi, "Differential privacy data publishing method based on cell merging," *IEEE, International Conference on Networking, Sensing and Control*, pp. 778–782, 2017.

[18] L. Shen, P. Su, X. Lu, X. Wang, Y. Liu, and H. Ouyang, "A toll data publishing method using encryption and differential privacy preservation technology," in *Proceedings of the 2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pp. 1586–1594, Chengdu, December 2017.

[19] L. Xiaoye, "Differential Privacy for Edge Weights in Social Networks," *Security &; Communication Networks*, vol. 4, pp. 1–10, 2017.

[20] L. Zhang, Y. Liu, R. Wang, X. Fu, and Q. Lin, "Efficient privacy-preserving classification construction model with differential privacy technology," *Journal of Systems Engineering and Electronics*, vol. 28, no. 1, Article ID 7870511, pp. 170–178, 2017.

[21] J. Ren, J. Xiong, Z. Yao, R. Ma, and M. Lin, "DPLK-Means: A Novel Differential Privacy K-Means Mechanism," in *Proceedings of the 2017 IEEE Second International Conference on Data Science in Cyberspace (DSC)*, pp. 133–139, Shenzhen, China, June 2017.

[22] C. Xiang, "A Two-Phase Algorithm for Differentially Private Frequent Subgraph Mining," *IEEE Transactions on Knowledge & Data Engineering*, p. 99, 2018.

[23] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4–7, 2006. Proceedings*, vol. 3876 of *Lecture Notes in Computer Science*, pp. 265–284, Springer, Berlin, Germany, 2006.

[24] N. Li, M. Lyu, D. Su, and W. Yang, "Differential Privacy: From Theory to Practice," *Synthesis Lectures on Information Security, Privacy, and Trust*, vol. 8, no. 4, pp. 1–138, 2016.

[25] F. McSherry, "Privacy integrated queries: an extensible platform for privacy-preserving data analysis," *Communications of the ACM*, vol. 53, no. 9, pp. 89–97, 2010.

[26] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.

[27] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 5, no. 4, pp. 1–19, 2016.

[28] P. Kairouz, S. Oh, and P. Viswanath, "Extremal mechanisms for local differential privacy," in *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 2879–2887, MIT Press, 2014.

[29] S. L. Warner, "Randomized response: a survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–66, 1965.