APPLYING DIFFERENTIAL PRIVACY TO TENSOR COMPLETION

Zheng Wei[1], *Zhengpin Li*[1], *Xiaojun Mao*[2] *and Jian Wang*[1]

[1]School of Data Science, Fudan University, China
[2]School of Mathematical Sciences, Shanghai Jiao Tong University, China

ABSTRACT

Tensor completion aims at filling the missing or unobserved entries based on partially observed tensors. However, utilization of the observed tensors often raises serious privacy concerns in many practical scenarios. To address this issue, we propose a solid and unified framework that contains several approaches for applying differential privacy to the two most widely used tensor decomposition methods: i) CANDECOMP/PARAFAC and ii) Tucker decompositions. For each approach, we establish a rigorous privacy guarantee and meanwhile evaluate the privacy-accuracy trade-off. Experiments on synthetic datasets demonstrate that our proposal achieves high accuracy for tensor completion while ensuring strong privacy protections.

***Index Terms*—** Tensor completion, differential privacy

1. INTRODUCTION

In machine learning knowledge, missing data is a prevalent issue, which can be caused by data collection, data corrosion, or other artificial reasons. As one of the most popular completion methods, low-rank matrix completion has received much attention in a wide range of applications, such as collaborative filtering [1], computer vision [2], and multi-class learning [3]. However, there are many genuine cases where data has more than two dimensions and are best represented as multi-way arrays, such as tensor. For instance, electronic health records (EHRs) [4], which reserve patients' clinical histories, consist of three parts: patients, diagnosis, and procedure. A more common scenario is that data contains the time dimension, such as traffic data of network [5], which can be viewed as a series traffic matrix presenting the volumes of traffic between original and destination pairs by unfolding as time intervals. Therefore, as a natural high-order extension of low-rank matrix completion, low-rank tensor completion is gaining more and more interest.

For completion methods, privacy-preserving is a key issue, which was firstly proposed in [6] and considered as a vital goal for mining the value of data while protecting its privacy. In recent years, this issue has attracted increasing attention in matrix and tensor completions as well as their applications. For example, users are required to offer their ratings to recommender service in recommendation scenarios, which often raises serious privacy concerns because of insidious attacks and unexpected inference on users' ratings or profiles [7]. The purpose of privacy-preserving tensor completion is to ensure high-level privacy of the observed data, while keeping completion performance as high as possible.

To the best of our knowledge, few studies systematically studied privacy-preserving tensor completion. In this work, we propose a solid and unified framework for two most widely used tensor decomposition methods: CANDECOMP/PARAFAC (CP) decomposition [8, 9, 10] and Tucker decomposition [11, 12, 13] to maintain privacy guarantees by utilizing differential privacy [14], the dominant standard for privacy protection. The framework contains several privacy-preserving computation ways: input perturbation, gradient perturbation, and output perturbation. They all result in the trade-off between the accuracy and privacy-preserving.

In this paper, we first propose a solid and unified framework for applying differential privacy to tensor completion. In addition, We provide complete algorithm procedures and theoretical analysis for each privacy-preserving approach in our framework. Experimental results on synthetic and real-world datasets demonstrate that the proposed approaches can yield high accuracy, while ensuring strong privacy protections.

2. PRELIMINARIES AND NOTATIONS

Definition 1. *The standard CP decomposition factorizes a tensor into a sum of component rank-one tensors. Given a tensor $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$, we have*

$$\mathcal{X} \approx \sum_{r=1}^{R} \mathbf{a}_{:r}^{(1)} \circ \cdots \circ \mathbf{a}_{:r}^{(n)} = [\![\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(n)}]\!],$$

where R denotes the rank of tensor and $\mathbf{A}^{(n)}$ is the n-mode factor matrix consisting of R columns representing R latent components which can be represented as $\mathbf{A}^{(n)} = [\mathbf{a}_{:1}^{(n)} \cdots \mathbf{a}_{:R}^{(n)}]$.

Definition 2. *The standard Tucker decomposition factorizes a tensor into a core tensor multiplied by a matrix along each*

This work is supported by National Natural Science Foundation of China (12001109, 92046021 and 61971146), Shanghai Municipal Science and Technology Major Project (2018SHZDZX01) and ZJlab, and Science and Technology Commission of Shanghai Municipality Grant (20dz1200600). Z. Wei and Z. Li contributed equally to this work. X. Mao and J. Wang are corresponding authors (E-mail: maoxj@sjtu.edu.cn; jian_wang@fudan.edu.cn).

978-1-6654-0540-9/22/$31.00 ©2022 IEEE

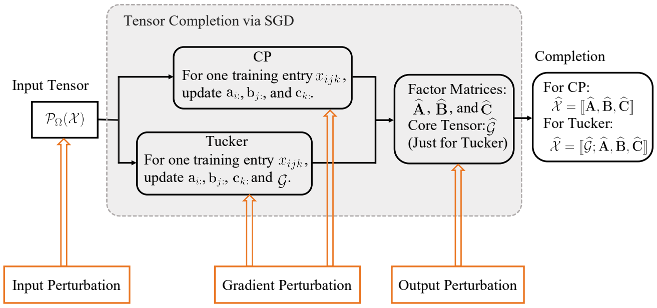

Tensor Completion via SGD
Input Tensor
$\mathcal{P}_\Omega(\mathcal{X})$
CP
For one training entry x_{ijk}, update $\mathbf{a}_{i:}, \mathbf{b}_{j:}$, and $\mathbf{c}_{k:}$.
Tucker
For one training entry x_{ijk}, update $\mathbf{a}_{i:}, \mathbf{b}_{j:}, \mathbf{c}_{k:}$ and $\mathcal{G}$.
Factor Matrices: $\widehat{\mathbf{A}}$, $\widehat{\mathbf{B}}$, and $\widehat{\mathbf{C}}$
Core Tensor: $\widehat{\mathcal{G}}$
(Just for Tucker)
Completion
For CP: $\widehat{\mathcal{X}} = [\![\widehat{\mathbf{A}}, \widehat{\mathbf{B}}, \widehat{\mathbf{C}}]\!]$
For Tucker: $\widehat{\mathcal{X}} = [\![\widehat{\mathcal{G}}; \widehat{\mathbf{A}}, \widehat{\mathbf{B}}, \widehat{\mathbf{C}}]\!]$
Input Perturbation
Gradient Perturbation
Output Perturbation

Fig. 1. Various perturbation approaches within tensor completion framework.

mode. For a tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, we can express it by

$$\mathcal{X} = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{t=1}^{T} g_{pqt} \mathbf{a}_{:p} \circ \mathbf{b}_{:q} \circ \mathbf{c}_{:t} = [\![\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$$

where $\mathcal{G} \in \mathbb{R}^{P \times Q \times T}$ and g_{pqt} indicate the core tensor and its element on coordinate (p, q, t) respectively, and $\mathbf{A} \in \mathbb{R}^{I \times P}$, $\mathbf{B} \in \mathbb{R}^{J \times Q}$ and $\mathbf{C} \in \mathbb{R}^{K \times T}$ denote the factor matrices.

Definition 3. *A (randomized) algorithm $\mathcal{A}$ whose outputs lie in a domain $\mathcal{S}$ is said to be ϵ-differentially private if for all subsets $S \subseteq \mathcal{S}$, for all datasets $\mathcal{D}$ and $\mathcal{D}'$ that differ in at most one entry, it holds that:*

$$\Pr(\mathcal{A}(\mathcal{D}) \in S) \leq e^{\epsilon} \Pr\left(\mathcal{A}\left(\mathcal{D}'\right) \in S\right). \tag{1}$$

Definition 4. *The L_p-sensitivity of a function $f : \mathcal{D}^n \to \mathbb{R}^d$ is the smallest number $\Delta_p(f)$ such that for all $\mathbf{x}, \mathbf{x}' \in \mathcal{D}^n$ which differ in a single entry,*

$$\left\|f(\mathbf{x}) - f\left(\mathbf{x}'\right)\right\|_p \leq \Delta_p(f), \tag{2}$$

where $\Delta_p(f)$ captures the magnitude by which a single individual's data can change the function f in the worst case, which provides an upper bound on how much we must perturb the input to preserve privacy.

3. DIFFERENTIAL PRIVACY TENSOR COMPLETION

In this section, we introduce the proposed framework for privacy-preserving tensor completion. We focus on the CP and Tucker decompositions with several privacy-preserving approaches via stochastic gradient descent (SGD) under the constraints of differential privacy. Considering the stages of tensor completion, we design input, gradient, and output perturbation approaches to maintain privacy, respectively. The overall framework is shown in Figure 1.

3.1. Problem Formulation

Hereafter, $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, which is generated by true tensor $\widetilde{\mathcal{X}}$ with unknown noise, represents the noisy incomplete tensor used to obtain estimated factor matrices and core tensor. We denote observation set by Ω which contains the indexes of available entries, and x_{ijk} is observed if and only if $(i, j, k) \in \Omega$. For convenience, we introduce the sampling operator $\mathcal{P}_\Omega$:

$$[\mathcal{P}_\Omega(\mathcal{X})]_{ijk} = \begin{cases} x_{ijk}, & (i, j, k) \in \Omega \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Denote three latent matrices derived from factorization by $\mathbf{A} \in \mathbb{R}^{n_1 \times d}$, $\mathbf{B} \in \mathbb{R}^{n_2 \times d}$, and $\mathbf{C} \in \mathbb{R}^{n_3 \times d}$ where d indicates the rank of $\widetilde{\mathcal{X}}$, and the CP based completion problem can be formulated as:

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \quad & f(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \|\mathcal{P}_\Omega(\mathcal{X} - [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!])\|_F^2 \\ & + \lambda(\|\mathbf{A}\|_F^2 + \|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2), \end{aligned} \tag{4}$$

where λ acts as a regularization parameter to control a tunable tradeoff between fitting errors and encouraging low-rank tensor. In terms of Tucker decomposition, we denote the core tensor by $\mathcal{G}$, and set the size of $\mathcal{G}$ to $d \times d \times d$ for simplicity. In a similar regularization manner for factor matrices, we impose a F-norm penalty to restrict the complexity of the core tensor. Thereby, we can reformulate the problem (4) as:

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathcal{G}} \; & f(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathcal{G}) = \|\mathcal{P}_\Omega(\mathcal{X} - [\![\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!])\|_F^2 \\ & + \lambda_o(\|\mathbf{A}\|_F^2 + \|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2) + \lambda_g \|\mathcal{G}\|_F^2, \end{aligned} \tag{5}$$

where λ_o and λ_g indicate regularization parameters for the factor matrices and the core tensor, respectively. The core tensor constitutes the main difference between these two decomposition methods. CP decomposition performs computationally more flexible in dealing with large-scale datasets, whereas Tucker decomposition is more general and effective because its core tensor can capture complex interactions among components that are not strictly trilinear [15]. Consequently, we can consider CP decomposition as a special case of Tucker decomposition where the cardinalities of the dimensions of latent matrices are equal and the off-diagonal elements of the core tensor are zero [16]. In the following parts, we provide theoretical analysis and algorithm procedures of the perturbation mechanisms based on Tucker decomposition.

3.2. Private Input Perturbation

In this approach, each entry of input tensor $\mathcal{X}$ is considered independent from the rest and perturbed by noise, which is bounded by L_1-sensitivity of $\mathcal{X}$. Suppose the entries of $\mathcal{X}$ are in the range of $[\mathcal{X}_{\max}, \mathcal{X}_{\min}]$, the L_1-sensitivity of the tensor is $\Delta_{\mathcal{X}}^{(I)} = \mathcal{X}_{\max} - \mathcal{X}_{\min}$, and noises are sampled from Laplace distribution denoted by $\mathrm{Lap}(\Delta_{\mathcal{X}}^{(I)}/\epsilon)$. This process is shown in Algorithm 1.

Theorem 1. *Algorithm 1 maintains ϵ-differential privacy.*

Proof of Theorem 1. The L_1-sensitivity of the input tensor is $\Delta_{\mathcal{X}}^{(I)} = \mathcal{X}_{\max} - \mathcal{X}_{\min}$. According to laplace mechanism [14], this algorithm maintains ϵ-differential privacy. □

Algorithm 1 Private Input Perturbation

```
Input: $\mathcal{X}$: noisy incomplete tensor, $\Omega$: indexes set of observations, $d$: rank of tensor, $\lambda_o$: regularization parameter for the factor matrices, $\lambda_g$: regularization parameter for the core tensor, $\epsilon$: privacy budget
Generate each entry of noise tensor $\mathcal{N}$ by $\mathrm{Lap}(\Delta_{\mathcal{X}}^{(I)}/\epsilon)$
Let $\mathcal{X}' = \{x_{ijk} + n_{ijk} | (i,j,k) \in \Omega\}$
Use $\mathcal{X}'$ as input to solve (5) via SGD and obtain estimated $\widehat{\mathbf{A}}, \widehat{\mathbf{B}}, \widehat{\mathbf{C}}$ and $\widehat{\mathcal{G}}$
Output: Estimated $\widehat{\mathbf{A}} \in \mathbb{R}^{n_1 \times d}$, $\widehat{\mathbf{B}} \in \mathbb{R}^{n_2 \times d}$, $\widehat{\mathbf{C}} \in \mathbb{R}^{n_3 \times d}$ and $\widehat{\mathcal{G}} \in \mathbb{R}^{d \times d \times d}$
```

3.3. Private Gradient Perturbation

The gradient perturbation maintains privacy by introducing noise in the SGD step [17]. In our gradient perturbation, we add noises to the computed gradients, and then utilize noisy gradients to update the corresponding rows of the factor matrices and the core tensor. For simplicity, we spend the all privacy budget on one single factor matrix $\mathbf{C}$. In each iteration, the gradient of $\mathbf{C}$ will be added by noise sampled from one exponential distribution. Before that, to be compatible with our theoretical assumption in Theorem 2, we clip the gradient l_2-norms of $\mathbf{C}$ to a constant m using $\mathbf{v} \leftarrow \mathbf{v}/\max(1, \|\mathbf{v}\|_2/m)$ [18]. The global sensitivity here is denoted by $\Delta_{\mathcal{X}}^{(G)}$. Algorithm 2 summarizes this process.

Algorithm 2 Private Gradient Perturbation

```
Input: $\mathcal{X}$: noisy incomplete tensor, $\Omega$: indexes set of observations, $d$: rank of tensor, $\lambda_o$: regularization parameter for the factor matrices, $\lambda_g$: regularization parameter for the core tensor, $n$: number of iterations, $\epsilon$: privacy budget, $\eta$: learning rate, $m$: clipping constant
Initialize random factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathcal{G}$
for $n$ iterations do
    for $x_{ijk} \in \mathcal{X}$ do
        $\mathbf{a}_{i:} \leftarrow \mathbf{a}_{i:} - \eta \nabla_{\mathbf{a}_{i:}} f$
        $\mathbf{b}_{j:} \leftarrow \mathbf{b}_{j:} - \eta \nabla_{\mathbf{b}_{j:}} f$
        $\nabla_{\mathbf{c}_{k:}} f \leftarrow \nabla_{\mathbf{c}_{k:}} f / \max(1, \|\nabla_{\mathbf{c}_{k:}} f\|_2/m)$
        Sample noise $\mathbf{n}_{i:}$ from $p(\mathbf{n}_{i:}) \propto \exp\left(-\frac{\varepsilon \|\mathbf{n}_{i:}\|}{\Delta_{\mathcal{X}}^{(G)}}\right)$
        $\mathbf{c}_{k:} \leftarrow \mathbf{c}_{k:} - \eta(\nabla_{\mathbf{c}_{k:}} f + \mathbf{n}_{i:})$
        $\mathcal{G} \leftarrow \mathcal{G} - \eta \nabla_{\mathcal{G}} f$
    end for
end for
Output: Estimated $\widehat{\mathbf{A}} \in \mathbb{R}^{n_1 \times d}$, $\widehat{\mathbf{B}} \in \mathbb{R}^{n_2 \times d}$, $\widehat{\mathbf{C}} \in \mathbb{R}^{n_3 \times d}$ and $\widehat{\mathcal{G}} \in \mathbb{R}^{d \times d \times d}$
```

Theorem 2. *Suppose that function f with regard to* $\mathbf{C}$ *in* (5) *is L-Lipschitz, Algorithm 2 maintains ϵ-differential privacy.*

Algorithm 3 Private Output Perturbation

```
Input: $\mathcal{X}$: noisy incomplete tensor, $\Omega$: indexes set of observations, $d$: rank of tensor, $\epsilon$: privacy budget
Solve (5) via SGD and obtain estimated $\widehat{\mathbf{A}}, \widehat{\mathbf{B}}, \widehat{\mathbf{C}}$ and $\widehat{\mathcal{G}}$
Sample noise matrix $\mathbf{N}$, whose rows are sampled from $\exp\left(-\frac{\epsilon \|\mathbf{n}_{i:}\|}{\Delta_{\mathcal{X}}^{(O)}}\right)$
$\widehat{\mathbf{C}} \leftarrow \widehat{\mathbf{C}} + \mathbf{N}$
Output: Estimated $\widehat{\mathbf{A}} \in \mathbb{R}^{n_1 \times d}$, $\widehat{\mathbf{B}} \in \mathbb{R}^{n_2 \times d}$, $\widehat{\mathbf{C}} \in \mathbb{R}^{n_3 \times d}$ and $\widehat{\mathcal{G}} \in \mathbb{R}^{d \times d \times d}$
```

Proof. Let $\mathcal{X}$ and $\mathcal{X}'$ be two tensors differing at only element x_{pqr} and x'_{pqr}. Let $\mathbf{N} = \{n_{ij}\}$ and $\mathbf{N}' = \{n'_{ij}\}$ be the noise matrices when training with $\mathcal{X}$ and $\mathcal{X}'$ respectively. According to the optimization formulation (5), it is obviously differentiable anywhere, which ensures the unique mapping from input to output. Denote $\mathbf{C}^*$ as the derived factor matrix minimizes both the optimization problems, and we have $\forall k \in \{1, 2, \cdots, n_3\}$, $\nabla_{\mathbf{c}_{k:}} f(\mathbf{c}^*_{k:}|\mathcal{X}) = \nabla_{\mathbf{c}_{k:}} f(\mathbf{c}^*_{k:}|\mathcal{X}')$. Thereby, given x_{ijk} and x'_{ijk}, we have:

$$\nabla_{\mathbf{c}_{k:}} f(\mathbf{c}^*_{k:}|\mathcal{X}) + \mathbf{n}_{k:} = \nabla_{\mathbf{c}_{k:}} f(\mathbf{c}^*_{k:}|\mathcal{X}') + \mathbf{n}'_{k:}. \tag{6}$$

Then we can derive that:

$$\|\mathbf{n}_{k:} - \mathbf{n}'_{k:}\| = \|\nabla_{\mathbf{c}_{k:}} f(\mathbf{c}^*_{k:}|\mathcal{X}) - \nabla_{\mathbf{c}_{k:}} f(\mathbf{c}^*_{k:}|\mathcal{X}')\| \leq 2L. \tag{7}$$

Denote $\Delta_{\mathcal{X}}^{(G)} = 2L$. For any pair of x_{pqg} and ${x'}_{pqg}$, we have:

$$\frac{\Pr[\mathbf{C} = \mathbf{C}^* \mid \mathcal{X}]}{\Pr[\mathbf{C} = \mathbf{C}^* \mid \mathcal{X}']} \leq \exp\left\{\frac{\epsilon(\|\mathbf{n}_{k:} - \mathbf{n}'_{k:}\|)}{\Delta_{\mathcal{X}}^{(G)}}\right\} \leq \exp(\epsilon). \tag{8}$$

Hence, the algorithm maintains ϵ-differential privacy for the whole process. □

3.4. Private Output Perturbation

The output perturbation achieves privacy protections by adding noise to the final model [14]. We can divide the privacy budget among all outputs in our approach, including the factor matrices and the core tensor. For simplicity, we only add noise to the estimated $\widehat{\mathbf{C}}$. After the updating process of SGD, noise vectors sampled by one exponential mechanism will be added to each row of $\widehat{\mathbf{C}}$. Define $\Delta_{\mathcal{X}}^{(O)} = 2\tau L\eta$ where τ, L, and η are the number of iterations, Lipschitz constant, and learning rate, respectively. This process is summarized in Algorithm 3.

Theorem 3. *Algorithm 3 maintains ϵ-differential privacy.*

Proof of Theorem 3. According to private convex permutation-based SGD [19], L_2-sensitivity is bounded by $2\tau L\eta$ denoted as $\Delta_{\mathcal{X}}^{(O)}$. By adding noises from exponential mechanism [14], it directly yields ϵ-differential privacy for this algorithm. □

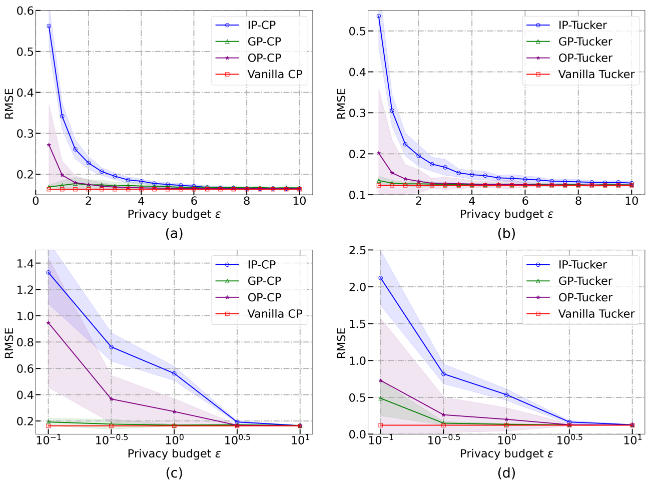
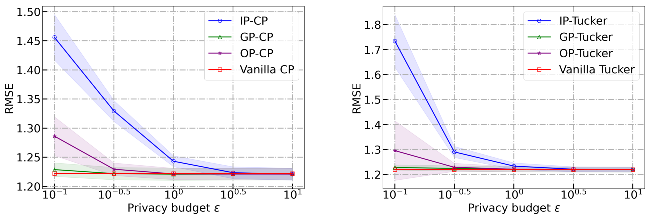


Fig. 2. Performance comparison of CP and Tucker decompositions. The left and right columns present the performance of CP decomposition and Tucker decomposition, respectively. The colored areas around curves reflects the standard deviations of RMSE, averaged over 50 runs.



(a) CP Decomposition

(b) Tucker Decomposition

Fig. 3. Comparison results on ML-100K through 10 runs.

4. EVALUATION

We evaluate our proposal on synthetic datasets where we set $\mathcal{X} \in \mathbb{R}^{20\times20\times20}$ with rank 3. We use different ways to generate target tensor for CP and Tucker decompositions. Following [20], we construct $\mathcal{X}$ for CP decomposition by $\mathcal{X} = [\![\widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}]\!] + \mathcal{N}$, where $\widetilde{\mathbf{A}} \in \mathbb{R}^{20\times3}$, $\widetilde{\mathbf{B}} \in \mathbb{R}^{20\times3}$ and $\widetilde{\mathbf{C}} \in \mathbb{R}^{20\times3}$ are from standard normal distribution, and $\mathcal{N}$ is a zero mean Gaussian noise tensor with signal-to-noise ratio (SNR) being one. In addition, all columns of factor matrices are normalized to unit length. For Tucker decomposition, we generate the factor matrices in a similar manner with columns orthogonal to each other. We draw the entries of the core tensor $\widetilde{\mathcal{G}} \in \mathbb{R}^{3\times3\times3}$ from standard normal distribution [15] and construct $\mathcal{X}$ via $\mathcal{X} = [\![\widetilde{\mathcal{G}}; \widetilde{\mathbf{A}}, \widetilde{\mathbf{B}}, \widetilde{\mathbf{C}}]\!] + \mathcal{N}$ where $\mathcal{N}$ has the same value as in CP decomposition. For visualization, we transform $\widetilde{\mathcal{X}}$ by min-max scaling before introducing noise tensor. For each experiment, we randomly split observations into 80% and 20% as train/test sets, and perform three perturbation approaches on two decomposition methods under appropriate parameters. For comparison, we use vanilla decomposition without perturbation as baselines. We evaluate the performance of tensor completion using Root Mean Square Error (RMSE), computed by $\mathrm{RMSE} = \sqrt{\sum_{\Omega}(\tilde{x}_{ijk} - \hat{x}_{ijk})^2/|\Omega|}$, where Ω denotes the set of indices of the test set.

Figure 2 shows the performance comparisons among several perturbation approaches under the same decomposition method. As expected, decomposition methods with perturbation approaches cannot outperform the baselines, and their RMSE increase with the privacy parameter ϵ shrinking. This can be explained by that keeping a higher level of privacy means introducing larger noises, which leads to lower accuracy. Overall, there is no significant difference in the trade-off of privacy-accuracy between CP decomposition and Tucker decomposition. Specifically, the performance of gradient perturbation (GP) is followed by output perturbation (OP) and input perturbation (IP) in terms of accuracy and stability. In Figure 2 (a) and (b), we observe that the curves of GP are very close to that of the baselines, which is caused by the experimental setting where we set $\Delta_{\mathcal{X}}^{(G)} = 2m$, and m here indicates the clipping constant. A small clipping constant means a small $\Delta_{\mathcal{X}}^{(G)}$, which can offset the impact of smaller ϵ. We observe the trade-off of gradient perturbation in Figure 2 (c) and (d), where privacy budgets are presented by exponential magnitude.

We also study the performance of our methods on MovieLens 100K [21] datasets, which consists of 943 users and 1682 movies with density 6.30%. We divide the timestamps into 212 values by day and unfold the original rating matrix to tensor by expanding timestamps as the third dimension. We utilize the canonical partition (ua.base/ua.test and ub.base/ub.test) for training and evaluation. To avoid the bias issue of data, we employ the bi-scaling procedure [22], which standardizes a matrix to have rows and columns of means zero and variances one, to matrices separated from tensor by timestamp before applying any perturbation methods. For parameters setting, we set λ to 0.01 in (4) and the learning rate to 0.005 in CP decomposition. Also, we set $\lambda_o = 0.01$, $\lambda_g = 0.001$ in (5) and the learning rate to 0.003 in Tucker decomposition. For both methods, we set the maximum number of iterations to 100. Figure 3 shows that three approaches have comparable performance to that on synthetic datasets, which validates the effectiveness of our proposal in practical scenarios.

5. CONCLUSION

In this paper, we have established a unified privacy-preserving framework for CP and Tucker decompositions. This framework contains three perturbation approaches to tackle the privacy issue in tensor completion via differential privacy. For each approach, we have provided the algorithm procedures and theoretical analyses. Experiments on synthetic datasets have verified the effectiveness of the framework. Particularly worth mentioning is that the gradient perturbation approach can achieve a stable and striking accuracy with small privacy budgets, indicating great potential for practical applications.

6. REFERENCES

[1] David Goldberg, David Nichols, Brian M Oki, and Douglas Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.

[2] Carlo Tomasi and Takeo Kanade, "Shape and motion from image streams under orthography: a factorization method," *International Journal of Computer Vision*, vol. 9, no. 2, pp. 137–154, 1992.

[3] Yonatan Amit, Michael Fink, Nathan Srebro, and Shimon Ullman, "Uncovering shared structures in multiclass classification," in *Proceedings of the 24th International Conference on Machine Learning*, 2007, pp. 17–24.

[4] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-Wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark, "Mimic-iii, a freely accessible critical care database," *Scientific Data*, vol. 3, no. 1, pp. 1–9, 2016.

[5] Yehuda Vardi, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 365–377, 1996.

[6] Rakesh Agrawal and Ramakrishnan Srikant, "Privacy-preserving data mining," in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, 2000, pp. 439–450.

[7] Frank McSherry and Ilya 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*, 2009, pp. 627–636.

[8] Frank L Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *Journal of Mathematics and Physics*, vol. 6, no. 1-4, pp. 164–189, 1927.

[9] J Douglas Carroll and Jih-Jie Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.

[10] Richard A Harshman et al., "Foundations of the parafac procedure: Models and conditions for an" explanatory" multimodal factor analysis," 1970.

[11] Ledyard R Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[12] Pieter M Kroonenberg and Jan De Leeuw, "Principal component analysis of three-mode data by means of alternating least squares algorithms," *Psychometrika*, vol. 45, no. 1, pp. 69–97, 1980.

[13] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle, "A multilinear singular value decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[14] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference*. Springer, 2006, pp. 265–284.

[15] Qingquan Song, Hancheng Ge, James Caverlee, and Xia Hu, "Tensor completion algorithms in big data analytics," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 13, no. 1, pp. 1–48, 2019.

[16] Tamara G Kolda and Brett W Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[17] Raef Bassily, Adam Smith, and Abhradeep Thakurta, "Private empirical risk minimization: Efficient algorithms and tight error bounds," in *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE, 2014, pp. 464–473.

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

[19] Xi Wu, Fengan Li, Arun Kumar, Kamalika Chaudhuri, Somesh Jha, and Jeffrey Naughton, "Bolt-on differential privacy for scalable stochastic gradient descent-based analytics," in *Proceedings of the 2017 ACM International Conference on Management of Data*, 2017, pp. 1307–1322.

[20] Evrim Acar, Daniel M Dunlavy, Tamara G Kolda, and Morten Mørup, "Scalable tensor factorizations for incomplete data," *Chemometrics and Intelligent Laboratory Systems*, vol. 106, no. 1, pp. 41–56, 2011.

[21] Maxwell F. Harper and Joseph A. Konstan, "The movielens datasets: History and context," *ACM Transactions on Unteractive Intelligent Systems*, vol. 5, no. 4, pp. 1–19, 2015.

[22] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *Journal of Machine Learning Research*, vol. 11, pp. 2287–2322, 2010.