

## Multiple Strategies Differential Privacy on Sparse Tensor Factorization for Network Traffic Analysis in 5G

Wang, Jin; Han, Hui ; Li, Hao; He, Shiming ; Sharma, Pradip Kumar ; Chen, Lydia

**DOI**

[10.1109/TII.2021.3082576](https://doi.org/10.1109/TII.2021.3082576)

**Publication date**

2021

**Document Version**

Accepted author manuscript

**Published in**

IEEE Transactions on Industrial Informatics

**Citation (APA)**

Wang, J., Han, H., Li, H., He, S., Sharma, P. K., & Chen, L. (2021). Multiple Strategies Differential Privacy on Sparse Tensor Factorization for Network Traffic Analysis in 5G. *IEEE Transactions on Industrial Informatics*, 18(3), 1939 - 1948. Article 9439054. <https://doi.org/10.1109/TII.2021.3082576>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

# Multiple Strategies Differential Privacy on Sparse Tensor Factorization for Network Traffic Analysis in 5G

Jin Wang, *Senior Member, IEEE*, Hui Han, Hao Li, Shiming He, Pradip Kumar Sharma, *Senior Member, IEEE*, Lydia Chen, *Senior Member, IEEE*

**Abstract**—Due to high capacity and fast transmission speed, 5G plays a key role in modern electronic infrastructure. Meanwhile, Sparse Tensor Factorization (STF) is a useful tool for dimension reduction to analyze High-Order, High-Dimension, and Sparse Tensor (HOHDST) data which is transmitted on 5G Internet-of-things (IoT). Hence, HOHDST data relies on STF to obtain complete data and discover rules for real-time and accurate analysis. From another view of computation and data security, the current STF solution seeks to improve the computational efficiency but neglects privacy security of the IoT data, e.g., data analysis for network traffic monitor system. To overcome these problems, this paper proposes a Multiple-strategies Differential Privacy framework on STF (MDPSTF) for HOHDST network traffic data analysis. MDPSTF comprises three Differential Privacy (DP) mechanisms, i.e.,  $\epsilon$ -DP, Concentrated DP (CDP), and Local DP (LDP). Furthermore, the theoretical proof of privacy bound is presented. Hence, MDPSTF can provide general data protection for HOHDST network traffic data with high-security promise. We conduct experiments on two real network traffic datasets (*Abilene* and *GEANT*). The experimental results show that MDPSTF has high universality on the various degrees of privacy protection demands and high recovery accuracy for the HOHDST network traffic data.

**Index Terms**—Differential Privacy Framework; Sparse Tensor Factorization; Multiple-strategies Privacy Protection; Network Traffic Analysis.

## I. INTRODUCTION

With the advent of 5G technology, high speed and low latency 5G networks bring in huger transmission capacity than 4G, which can support the emerging application technologies relying on IoT, e.g., virtual reality, augmented reality, and wearable devices linked with mobile phones. Meanwhile, thousands of connected devices on the 5G IoT network will generate a tremendous amount of data, and the network traffic data which can capture the moving data across a network, usually be used to prevent the network jam and paralysis [1]. Hence, 5G IoT relies heavily on accurate and real-time network traffic analysis to maintain a steady, fluent,

and high-speed network environment [2]. Due to distributed deployment of sensor devices, network traffic data has the form of temporal-spatial characteristic, and, own to downtime and some other crash problems for some devices linked by 5G IoT, the network traffic data presents the form of High-Order, High-Dimension, and Sparse Tensor (HOHDST) [3]. For HOHDST data, Sparse Tensor Factorization (STF) can draw the low-rank feature of each order from HOHDST data simultaneously. Thus, STF plays a key role in the analysis of network traffic data [4], which can obtain complete data to measure more valuable information. Practitioners focus on mining the STF algorithm to conduct an accurate recovery for HOHDST network traffic data, while neglecting the data privacy in the transmission and analysis process.

Network traffic data involves the information about the location coordinate and network flow, and attackers can infer the individual private information. Meanwhile, there are three layers that consider how to address the network security problems, i.e., physical network security, technical network security, and administrative network security. It is designed to prevent unauthorized personnel from obtaining physical access to the network, viruses and other malicious software to manipulate the network, and protect the data.

But none of them can deal with the hidden danger that keeps high-security promise for the traffic data on data-level. Differential Privacy (DP) is a strong privacy protection mechanism in data-level which can guarantee that anyone cannot make inference about the individual's private information by adding a kind of noise; Meanwhile, DP can ensure the availability of data [5]. DP can also provide mathematical provable privacy protection against common privacy attacks, i.e., linkage and reconstruction attacks [6].

More recently, due to the appetency to provide a privacy guarantee [7], DP naturally becomes a preferred tool for data privacy protection in the training process of Machine Learning (ML) models [8–10]. DP can provide privacy protection for the classic dimension reduction model, i.e., Matrix Factorization (MF) [11, 12] and the application fields have been extended to recommender systems and social networks [13, 14]. However, the DP for MF model (DPMF) can only handle the privacy protection for two-order matrix data and the DPMF cannot provide the same privacy security promise for third or higher-order tensor data; Meanwhile, the current DPMF models can apply  $\epsilon$ -DP and Local DP (LDP) individually, but don't have a general privacy protection framework.

Jin Wang, Hui Han, and Shiming He are with School of Computer and Communication Engineering, ChangSha University Of Science and Technology. Hao Li and Lydia Chen are with Faculty of Electrical Engineering, Mathematics & Computer Science, Delft University of Technology. Pradip Kumar Sharma is with Department of Computing Science, University of Aberdeen.

Corresponding author: Hao Li.

E-mail: jinwang@csust.edu.cn, h\_han@stu.csust.edu.cn, H.Li-9@tudelft.nl (lihao123@hnu.edu.cn), smhe\_cs@csust.edu.cn, pradip.sharma@abdn.ac.uk, lydiachen@ieee.org.

There are several works that explore the DP mechanism for Tensor Factorization (TF). Wang et al. proposed a DP framework for the tensor level method and the tensor data is symmetric and dense one [15]. However, this method only considers the  $\epsilon$ -DP mechanism, and the memory overhead is not scalable. For distributed, large-scale, symmetric and dense tensor data, Imtiaz et al, presented a distributed DP framework for orthogonal TF and the factorization process for the low-rank matrix of each order involves the Singular Value Decomposition (SVD) for a huge symmetric matrix. Because the privacy proof on dense data doesn't apply to the asymmetric and HOHDST data generated in real-world applications, the above DP framework cannot provide a security promise, e.g., network traffic data and medical health data [16].

For HOHDST data, researchers focus on how to improve the computational efficiency and prediction accuracy for missing values, especially in traffic network data. Li et al., proposed a high performance computation framework on GPU for Sparse MF (SMF) [17]. Li et al., presented Stochastic Gradient Descent (SGD) based algorithm for STF to reduce the computational complexity [18] and in the face of accurate HOHDST network traffic data recovery, there have been many accurate processing technologies. Xie et al., widely explored the low-rank structure of two-order matrix and high-order tensor generated from network traffic data and proposed an accurate sparse matrix and tensor recovery framework based on SMF and STF, respectively [19]. The above methods don't solve the problem of privacy protection for HOHDST data, just improve the computational efficiency and recovery accuracy.

Recently, in order to overcome the above limitations and meet the requirements of real-time and accurate recovery of HOHDST network traffic data, Ma et al., proposed a framework of Canonical Polyadic (CP) factorization under Concentrated DP (CDP) protection for electronic health records. This method can compute the distributed STF which can impute local missing diagnosis information and avoid direct data sharing, meanwhile, this model doesn't leak the local patient diagnosis information [20]. Nie et al., presented an STF analysis framework for IoT data generated from cloud and edge under  $\epsilon$ -DP protection [21]. However, those methods don't solve the general privacy protection problem for HOHDST network traffic data, which means that those methods cannot ensemble various DP, i.e.,  $\epsilon$ -DP, CDP, and LDP, and provide a general privacy protection framework for HOHDST network traffic data. To overcome the above limitations and meet the requirements of accurate recovery of HOHDST network traffic data under restrict and mathematically provable privacy protection, the following challenges should be solved:

- 1) Different DP has different system framework requirements on the STF.
- 2) LDP applications that do not apply to third-order tensors.
- 3) Achieve an optimal trade-off between the degree of privacy protection and the precision of data recovery.

To handle these problems, A HOHDST network traffic data recovery framework under Multiple-strategies Differential Privacy protection on STF (MDPSTF) is proposed. The HOHDST network traffic data recovery relies on the

CP factorization, the factorization process of CP meets the requirements of real-time analysis in the recovery process of HOHDST network traffic data and only involves the low-rank matrix of each order. The main contributions of this work are summarized as follows:

- 1) This is the first work to realize an ensemble MDPSTF framework. The framework MDPSTF combines three DP mechanisms, i.e.,  $\epsilon$ -DP, CDP, and LDP, which can recover the HOHDST network traffic data and provide privacy protection.
- 2) Laplace mechanism and Gaussian mechanism add corresponding noise to the third-party trusted server respectively and realize the privacy protection following DP and zCDP definition.
- 3) The LDP protects the data collection source, that is, adding noise to the whole tensor data on the user's device, which can protect the data from the attack of the untrusted third-party server.
- 4) The theoretical proof of privacy under various mechanisms is given. Combined with the experimental results, the data recovery accuracy of each mechanism under different privacy degrees is analyzed.

The rest of this paper is organized as follows: the problem formulation and preliminaries are presented in Section II. Section III introduces the framework of MDPSTF for the recovery problem of HOHDST network traffic data under general privacy protection. The experimental results are given in Section IV. Section V concludes this paper.

## II. PROBLEM FORMULATION AND PRELIMINARIES

### A. Sparse Tensor Factorization (STF)

The main notation, including scalars, vectors, matrices, and tensors, as well as the slice format of tensors are listed in Table I. In the following sections, for simplicity, the STF refers to the CP factorization.

TABLE I: Definition of symbols.

Symbol	Definition
$I_n$	The size of row in the $n$ th factor matrix;
$R$	The rank of CP factorization;
$\mathcal{X}$	$N$ -order tensor $\in \mathbb{R}^{I_1 \times \dots \times I_n \times \dots \times I_N}$ ;
$\mathbf{A}^{(n)}$	The $n$ -th factor matrix $\in \mathbb{R}^{I_n \times R}$ ;
$x_{(i_1, \dots, i_n, \dots, i_N)}$	The $(i_1, \dots, i_n, \dots, i_N)$ -th entry of $\mathcal{X}$ ;
$\mathbf{a}_{i_n}^{(n)}$	$i_n$ -th row vector $\in \mathbb{R}^R$ of $\mathbf{A}^{(n)}$ ;
$\mathbf{a}_{:,j_n}^{(n)}$	$j_n$ -th column vector $\in \mathbb{R}^{I_n}$ of $\mathbf{A}^{(n)}$ ;
$a_{i_n, j_n}^{(n)}$	The $(i_n, j_n)$ -th element of $\mathbf{A}^{(n)}$ ;
$\ \bullet\ _2$	$L_2$ norm;
$\Omega$	Index $(i_1, \dots, i_n, \dots, i_N)$ of a tensor;
$\mathbf{X}_{:,i_n,:}$	The $i_n$ -th lateral slice matrix $\in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N}$ of tensor $\mathcal{X}$ ;
$\circ$	Outer production of vectors;

**Definition 1** (Tensor Approximation). *Given a  $N$ -order tensor  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_n \times \dots \times I_N}$ , the low-rank tensor approximation problem can be formalized as  $\mathcal{X} = \tilde{\mathcal{X}} + \zeta$ , where  $\tilde{\mathcal{X}}$  is the low-rank tensor and  $\zeta$  is noise data. The optimization problem can be formalized as  $\arg \min_{\tilde{\mathcal{X}}} \|\mathcal{X} - \tilde{\mathcal{X}}\|_2^2$ .*

In this paper, we only consider the accurate recovery and privacy protection problem of HOHDST network traffic data. Hence, besides tensor approximation, the definition of sparse tensor factorization should be presented.

**Definition 2** (Sparse Tensor Factorization (STF)). *The approximation tensor can be obtained by sum of outer product rank-1 tensors  $\tilde{\mathcal{X}} = \sum_{r=1}^R \lambda_r \tilde{a}_{:,r}^{(1)} \circ \dots \circ \tilde{a}_{:,r}^{(n)} \circ \dots \circ \tilde{a}_{:,r}^{(N)}$ . The constant  $\lambda_r$ ,  $r \in \{1, \dots, R\}$  can be omitted. Factor matrices  $\mathbf{A}^{(n)}$ ,  $n \in \{1, \dots, N\}$  are obtained by following the sparsity pattern of the sparse tensor  $\mathcal{X}$ .*

**Definition 3** ( $\mu$  Strongly-convex). *For any  $x_1, x_2 \in \mathbb{R}^r$ , if there is a constant  $\mu > 0$ ,  $f(x_1) \geq f(x_2) + \nabla f(x_2)(x_1 - x_2)^T + \frac{1}{2}\mu\|x_1 - x_2\|_2^2$ , then the continuously differentiable function  $f(x)$  satisfies the  $\mu$  strongly-convexity.*

**Definition 4** ( $L$ -Lipschitz Continuity). *Suppose there is a continuously differentiable and  $L$ -smooth function  $f(x)$ ,  $x \in \mathbb{R}^r$ . Take any two gradient values  $\nabla f(x_1), \nabla f(x_2) \in \mathbb{R}^r$  of the two variables  $x_1, x_2 \in \mathbb{R}^r$ , and if these two gradient values satisfy  $\|\nabla f(x_1) - \nabla f(x_2)\|_2 \leq L\|x_1 - x_2\|_2$ , then the gradient  $\nabla f(x)$  is  $L$ -Lipschitz continuous for any  $x \in \mathbb{R}^r$ .*

**Definition 5** (Stochastic Gradient Descent (SGD)). *The optimization loss function  $f(w)$  is  $\mu$  Strongly-convex and  $L$ -Lipschitz Continuity as:*

$$\arg \min_{w \in \mathbb{R}^R} f(w) = \underbrace{L \left( w \middle| y_i, x_i, w \right)}_{\text{Loss Function}} + \underbrace{\lambda_w R(w)}_{\text{Regularization}} \quad (1)$$

where  $y_i \in \mathbb{R}^1$ ,  $x_i \in \mathbb{R}^R$ ,  $i \in \{1, \dots, N\}$ ,  $w \in \mathbb{R}^R$  and  $L(w|y_i, x_i, w) + \lambda_w R(w) = \sum_{i=1}^N L_i(w|y_i, x_i, w) + \lambda_w R_i(w)$ . The original optimization model needs gradient which should select all the samples  $\{x_i|i \in \{1, \dots, N\}\}$  from the dataset  $\Omega$  and the Gradient Descent (GD) is presented as  $w \leftarrow w - \gamma \frac{\partial f_{\Omega}(w)}{\partial w}$ , where  $\frac{\partial f_{\Omega}(w)}{\partial w} = \frac{1}{N} \sum_{i=1}^N \frac{\partial (L_i(w) + \lambda_w R_i(w))}{\partial w}$ . An  $M$  entries set  $\Psi$  is randomly selected from the set  $\Omega$ , and the SGD [22] is presented as  $w \leftarrow w - \gamma \frac{\partial f_{\Psi}(w)}{\partial w}$ , where  $\frac{\partial f_{\Psi}(w)}{\partial w} = \frac{1}{M} \sum_{i \in \Psi} \frac{\partial (L_i(w) + \lambda_w R_i(w))}{\partial w}$  and  $\gamma$  is the learning rate.

In our work, the HOHDST network traffic data can be formalized as a third-order tensor  $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$  [23, 24]. Thus, the value  $N$  in this work is set as 3 and we substitute  $\mathbf{A}^n$ ,  $n \in \{1, 2, 3\}$  as  $\{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$ , respectively. The objective function  $f(\mathbf{A}, \mathbf{B}, \mathbf{C})$  is represented as:

$$f(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \underbrace{\frac{1}{2} \sum_{(i,j,k) \in \Omega} \left( x_{ijk} - \sum_{r=1}^R a_{ir} b_{jr} c_{kr} \right)^2}_{\text{Loss Function}} + \underbrace{\frac{1}{2} \lambda_1 \|\mathbf{A}\|_2^2 + \frac{1}{2} \lambda_2 \|\mathbf{B}\|_2^2 + \frac{1}{2} \lambda_3 \|\mathbf{C}\|_2^2}_{\text{Regularization}} \quad (2)$$

## B. Differential Privacy (DP)

Some preliminary knowledge about DP mechanism, i.e.,  $\epsilon$ -DP, CDP, and LDP [7–14, 20, 21], are presented.

**Definition 6** (Neighbor Datasets). *Supposed that for any two datasets on the database  $\mathcal{D}$ , deemed as  $\tilde{\mathcal{D}}$  and  $\hat{\mathcal{D}}$ , and the two datasets have the same structure. Entry record difference is denoted as  $\tilde{\mathcal{D}} \triangle \hat{\mathcal{D}}$ , and  $|\tilde{\mathcal{D}} \triangle \hat{\mathcal{D}}|$  means the number of entries record. If  $|\tilde{\mathcal{D}} \triangle \hat{\mathcal{D}}| = 1$ , the two datasets  $\tilde{\mathcal{D}}$  and  $\hat{\mathcal{D}}$  are called neighbor datasets.*

**Definition 7** ( $\epsilon$ -Differential Privacy ( $\epsilon$ -DP)). *Suppose any two neighbor datasets  $\tilde{\mathcal{D}}, \hat{\mathcal{D}} \in \mathcal{D}$ , and the output set is  $S \subset \mathbb{R}$ . If a random mechanism  $f: \mathcal{D} \rightarrow \mathbb{R}$  satisfies  $\Pr[f(\tilde{\mathcal{D}}) \in S] \leq e^\epsilon \Pr[f(\hat{\mathcal{D}}) \in S] + \delta$ . Then  $f$  satisfies  $(\epsilon - \delta)$ -DP. The  $\epsilon$  here is called the privacy.  $\delta$  can relax and guarantee on a very small probability. The larger the privacy, the higher privacy protection, but the worse the data availability [8–13].*

**Definition 8** (Global Sensitivity). *Suppose there is a query function  $f: \mathcal{D} \rightarrow \mathbb{R}^d$ . If any pair of neighbor datasets  $\tilde{\mathcal{D}}$  and  $\hat{\mathcal{D}}$ , their global sensitivity is given by  $GS_f(\mathcal{D}) = \max \|f(\tilde{\mathcal{D}}) - f(\hat{\mathcal{D}})\|_1$ , where  $\|f(\tilde{\mathcal{D}}) - f(\hat{\mathcal{D}})\|_1$  is the Manhattan distance between  $f(\tilde{\mathcal{D}})$  and  $f(\hat{\mathcal{D}})$ .  $L_1$  norm is available. Global sensitivity has nothing to do with datasets, only with query results [8, 13].*

**Definition 9** (Laplace Mechanism). *The Laplace mechanism derives  $\epsilon$ -DP is often used in numerical output functions. It is basically to add a noise tensor of the same size as the original tensor data, where the noise element conforms to the Laplace distribution. Defined a function  $f: \mathcal{D} \rightarrow \mathbb{R}^d$ . The probability density function of tensor Laplace distribution:  $f(x_{i_1, i_2, \dots, i_N}) = \frac{\exp(-\frac{|x_{i_1, i_2, \dots, i_N}|}{\lambda})}{2\lambda}$ , when  $\lambda$  is the noise parameter [21].*

**Definition 10** (zero-Concentrated Differential Privacy (zCDP)). *Supposed that there is a random mechanism  $M$ , If any two neighbor databases  $\tilde{\mathcal{D}}$  and  $\hat{\mathcal{D}}$ , and  $\tilde{\mathcal{D}}$  differs from  $\hat{\mathcal{D}}$  by at most one entry. If the  $\alpha$ -Rényi divergence between the distributions of  $M(\tilde{\mathcal{D}})$  and  $M(\hat{\mathcal{D}})$  of these two databases with respect to the distribution satisfies  $D_{\alpha}(M(\tilde{\mathcal{D}})||M(\hat{\mathcal{D}})) \triangleq \frac{1}{\alpha-1} \log(\mathbb{E}[e^{(\alpha-1)Z}]) \leq \xi + \rho\alpha$ , where  $Z$  is the privacy loss random variable denoted as  $\text{PrivLoss}(M(\tilde{\mathcal{D}})||M(\hat{\mathcal{D}}))$ . Then the random mechanism  $M$  is  $\rho$ -zCDP in  $\alpha \in (1, \infty)$ . Define the privacy loss random variable  $Z$  between  $\tilde{\mathcal{D}}$  and  $\hat{\mathcal{D}}$ .  $Z$  is distributed according to  $f(\tilde{\mathcal{D}})$ , where the function  $f: \tilde{\mathcal{D}} \rightarrow \mathbb{R}$  by  $f(d) = \log \frac{\mathbb{P}(\tilde{\mathcal{D}}=d)}{\mathbb{P}(\hat{\mathcal{D}}=d)}$  [20].*

**Definition 11** (Local Differential Privacy (LDP)). *Given a privacy mechanism  $M$  and its domain and range are defined as  $\text{Dom}(M)$  and  $\text{Ran}(M)$ , respectively. If the mechanism  $M$  satisfied the following inequality on any two records  $x$  and  $x'$  ( $x, x' \subset \text{Dom}(M)$ ) that obtained the same output  $x^*$  ( $x^* \subseteq \text{Ran}(M)$ ), then  $M$  satisfies the  $\epsilon$ -LDP as  $\Pr[M(x) = x^*] \leq e^\epsilon \Pr[M(x') = x^*]$ . The Randomized Response (RR) is the primary disturbance mechanism of LDP. This mechanism mainly consists of two steps: perturbation statistics and correction [7, 14].*

Current DP mechanisms have been applied successfully in dense TF communities. However, due to the lack of data recovery and general privacy protection strategies, those solutions

cannot make privacy protection for HOHDST network traffic data. In section III, the general privacy protection framework MDPSTF for HOHDST network traffic data is presented.

Both DP and CDP work in a supposedly trusted third-party server. The difference lies in that DP accepts the entire tensor dataset transmitted by the user, while CDP accepts matrix data from each user and aggregates it into the tensor dataset. The next step: decomposing the tensor dataset through CP factorization, and three factor matrices can be obtained. As for the factor matrix  $\mathbf{A}$ , it contains a large amount of user information. So Laplace mechanism and Gaussian mechanism are respectively used to add noise in this paper. LDP is based on the assumption that the third-party server is not trusted. In order to protect the data at the source, the randomized response mechanism is used to add noise on each user before the server aggregates the data.

### III. MULTIPLE STRATEGIES DIFFERENTIAL PRIVACY FOR SPARSE TENSOR FACTORIZATION (MDPSTF)

The research background of this paper is the data recovery of HOHDST network traffic data generated from 5G network. Due to equipment failure, signal missing and other crash problems, the network traffic data is commonly sparse and the transmission process of the HOHDST network traffic data cannot leak privacy information. Therefore, in this work, we conduct the MDPSTF framework to make data recovery under privacy protection for HOHDST network traffic data.

As Fig. 1 shows that MDPSTF can protect the HOHDST network traffic data from two views: 1) the obtained factor matrices and 2) the original HOHDST data. DP is used to ensure the privacy security of users and to achieve a balance between recovery accuracy and privacy protection. To consider the accuracy of the data recovery, the HOHDST network traffic data is firstly modeled as a tensor, i.e., a third-order tensor  $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ . The tensor  $\mathcal{X}$  is generated from a dynamic 5G network which comprises user and location coordinate. By combining the low-rank representation of CP factorization and its factor matrix with various DP strategies, a general framework MDPSTF is proposed, which can recover the HOHDST network traffic data under privacy protection.

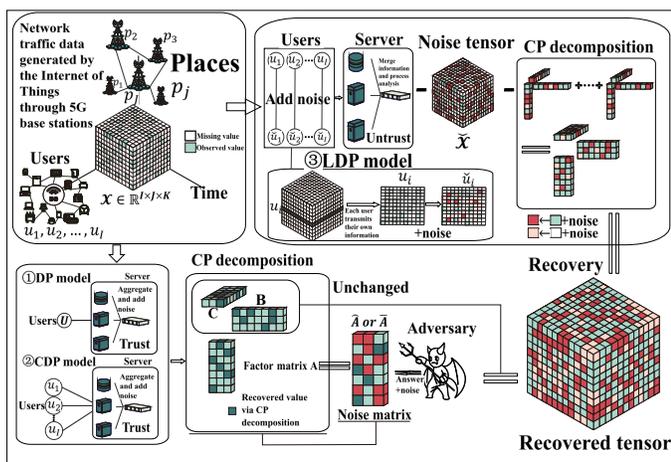


Fig. 1: Framework of MDPSTF

The privacy strategies can be divided into two categories: 1) after CP factorization, the servers will aggregate information and add noise processing for  $\epsilon$ -DP and CDP; 2) In LDP, the server will aggregate the information and CP factorization after users add noise information. In Fig. 1,  $u_i$  denotes the information matrix of the  $i$ -th user, and  $\tilde{u}_i$  represents the noise information matrix of the  $i$ -th user. Table II records the variables which are used in MDPSTF.

TABLE II: Definition of symbols.

Symbol	Definition
$\tilde{\mathcal{X}}$	The recovered tensor by CP factorization;
$\hat{\mathcal{X}}$	The recovered tensor after $\epsilon$ -DP;
$\tilde{\mathcal{X}}$	The recovered tensor after CDP;
$\tilde{\mathcal{X}}^*$	The recovered tensor after LDP;
$\alpha$	The learning rate of SGD;
$\rho$	Privacy budget;
$\eta$	The noise matrix $\in \mathbb{R}^{I \times R}$ ;
$\mu$	The mean of a Gaussian distribution;
$\sigma$	The variance of a Gaussian distribution;
$L$	The Lipschitz Constant;
$u$	A Bernoulli random variable;
$R$	Rank of CP factorization (CP-ranks);
$\mathcal{X}^n$	Each of these elements represents information about a user.

#### A. $\epsilon$ -DP Mechanism

$\hat{\mathbf{A}}$  is the noise factor matrix which means adding the noise  $\hat{\mathbf{A}} = \mathbf{A} + \eta$ , and then the disturbed objective function is  $f(\hat{\mathbf{A}}, \mathbf{B}, \mathbf{C}) = f(\mathbf{A} + \eta, \mathbf{B}, \mathbf{C})$ .

**Theorem 1.** Let  $\mathcal{X}$  be the range of user network traffic data values. If each noise value  $\eta$  is independent and randomly selected from the density function of Laplace distribution, where  $\Delta = \mathcal{X}_{max} - \mathcal{X}_{min}$ , the derived factor matrix  $\hat{\mathbf{A}}$  is deduced to satisfy the  $\epsilon$ -DP.

*Proof.* We assume that there is only one record difference between the two HOHDST tensor  $\left\{ \tilde{\mathcal{X}} = \{ \tilde{x}_{(1,1,1)}, \dots, \tilde{x}_{(i,j,k)}, \dots, \tilde{x}_{(I,J,K)} \}, \hat{\mathcal{X}} = \{ \hat{x}_{(1,1,1)}, \dots, \hat{x}_{(i,\hat{j},\hat{k})}, \dots, \hat{x}_{(I,J,K)} \} \right\}$ ,  $(i, j, k)$  and  $(i, \hat{j}, \hat{k}) \subset \Omega$ .  $N$  and  $\hat{N}$  are respectively expressed as the noise matrix of  $\mathcal{X}$  and  $\hat{\mathcal{X}}$ . We observe that  $f(\hat{\mathbf{A}}, \mathbf{B}, \mathbf{C})$  are differentiable anywhere. After obtaining the gradient, we set the equality  $\frac{\partial f(\mathbf{A}, \mathbf{B}, \mathbf{C})}{\partial a_{i,r}} = \frac{\partial f(\hat{\mathbf{A}}, \mathbf{B}, \mathbf{C})}{\partial \hat{a}_{i,r}}$ ,  $\{a_{i,r}, \hat{a}_{i,r}\}$  are  $(i, r)$ -element of the factor matrix  $\{\mathbf{A}, \hat{\mathbf{A}}\}$ , respectively. The above equation is expanded as:  $\eta_{i,r} - \sum_{(i,j,k) \in \Omega} (\tilde{x}_{(i,j,k)} - \sum_{r=1}^R a_{i,r} b_{j,r} c_{k,r}) (-b_{j,r} c_{k,r}) = \hat{\eta}_{i,r} - \sum_{(i,j,k) \in \Omega} (\hat{x}_{(i,j,k)} - \sum_{r=1}^R a_{i,r} b_{j,r} c_{k,r}) (-b_{j,r} c_{k,r})$ . If  $(i, j, k) \neq (i, \hat{j}, \hat{k})$ ,  $\eta_{i,r} - \hat{\eta}_{i,r} = 0$ . Else  $(i, j, k) = (i, \hat{j}, \hat{k})$ ,  $\eta_{i,r} - \hat{\eta}_{i,r} = (\sum_{r=1}^R a_{i,r} b_{j,r}^2 c_{k,r}^2) (\tilde{x}_{(i,j,k)} - \hat{x}_{(i,j,k)})$ . Then, the global sensitivity is defined as  $GS(a_{p,q}) = (\mathcal{X}_{max} - \mathcal{X}_{min}) \times (\max_{(i,j,k)} \|\sum_{l=p+1}^I \sum_{l=1}^{p-1} \sum_{r=q+1}^R \sum_{r=1}^{q-1} a_{l,r} b_{j,r}^2 c_{k,r}^2\|_F)$ . Because the factor matrices  $\{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$  are randomly sampled from the uniform distribution  $[0, 1]$ , so  $\max_{(i,j,k)} \|\sum_{l=p+1}^I \sum_{l=1}^{p-1} \sum_{r=q+1}^R \sum_{r=1}^{q-1} a_{l,r} b_{j,r}^2 c_{k,r}^2\|_F \leq 1$ . Hence, we can infer  $GS(a_{i,r}) \leq \Delta$ , then,  $\|\eta_{i,r} - \hat{\eta}_{i,r}\|_F \leq GS(a_{i,r}) \leq \Delta$ . The density function defined as:  $p(\eta_{i,r})$

$$\begin{aligned} &\propto e^{-\frac{\epsilon \|\eta_{i,r}\|}{2\Delta}}. \text{ Finally, } \frac{\Pr[\tilde{\mathcal{X}}]}{\Pr[\hat{\mathcal{X}}]} = \frac{\prod_{i \in \{1, \dots, I\}} p(\eta_{i,r})}{\prod_{i \in \{1, \dots, I\}} p(\hat{\eta}_{i,r})} = \\ &e^{-\frac{(\sum_{i \in \{1, \dots, I\}} \|\eta_{i,r}\| - \sum_{i \in \{1, \dots, I\}} \|\hat{\eta}_{i,r}\|)}{2\Delta}} = e^{-\frac{(\|\eta_{i,r}\| - \|\hat{\eta}_{i,r}\|)}{2\Delta}} \leq \\ &e^{-\epsilon}. \quad \square \end{aligned}$$

### B. CDP Mechanism

The privacy degree of CDP is tighter than that of  $\epsilon$ -DP, which provides a more explicit analysis for many calculations that retain privacy. Dwork and Rothblum came up with this CDP [25], Bun and Steinke propose an alternative formulation of CDP called "zero-CDP (zCDP)" [26]. The realization mechanism of zCDP is the same as the Gaussian mechanism of  $(\epsilon - \delta)$ -DP. The general Gaussian distribution as follows  $p(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(x-\mu)^2}{2\sigma^2})$ , where  $\mu$  represents the mean and  $\sigma$  represents the variance.

**Theorem 2** (Gaussian Mechanism). *Supposed that there is a random algorithm  $M$ . Let  $\epsilon \in (0, 1)$  be a arbitrary variable. For  $c^2 > 2\ln(\frac{1.25}{\delta})$ , the Gaussian mechanism with the parameter  $\sigma \geq c \Delta_2(M)/\epsilon$ , and adding noise scaled to  $\mathcal{N}(0, \sigma^2)$  to each component of algorithm  $M$  output, is  $(\epsilon - \delta)$ -DP.*

The following propositions will be used together in the proof of zCDP[25]:

**Proposition 1.** *The relation between Gaussian mechanism and zCDP. If for all  $x, x' \in \mathcal{X}^n$  differing in a single entry, and define a function  $q: \mathcal{X}^n \rightarrow \mathbb{R}$  be a sensitivity- $\Delta$ . We have  $|q(x) - q(x')| \leq \Delta$ . Suppose there have a Gaussian mechanism  $M: \mathcal{X}^n \rightarrow \mathbb{R}$  releases a sample from  $\mathcal{N}(q(x), \sigma^2)$  on the input  $x$ , then  $M$  satisfies  $(\frac{\Delta^2}{2\sigma^2}) - zCDP$ .  $(\frac{\Delta^2}{2\sigma^2})$  can also be written as  $\rho$ , namely  $\sigma = \frac{\Delta}{\sqrt{2\rho}}$ .*

**Proposition 2.** *The transformation between DP and zCDP. A randomized mechanism  $M$  is a  $\rho - zCDP$ . If any  $\delta$  with  $\epsilon = \rho + 2\sqrt{\rho \ln(\frac{1}{\delta})}$ , then  $M$  is a  $(\epsilon - \delta)$ -DP. On the contrary, the mechanism  $M$  satisfies  $(\epsilon - \delta)$ -DP. If  $\rho \approx \frac{\epsilon^2}{4\ln(\frac{1}{\delta})}$ , it suffices to satisfy  $\rho - zCDP$ .*

The theory of zCDP can be proved by the combination of the above two key propositions[20, 25, 26].

**Proposition 3** (Serial Composition). *Let  $M: \mathcal{D}^n \rightarrow \mathcal{Y}$  and  $M': \mathcal{D}^n \rightarrow \mathcal{Z}$  are any two random algorithms. If  $M$  is  $\rho - zCDP$  and  $M'$  is  $\rho' - zCDP$ . In addition to define a new random algorithm  $M'': \mathcal{D}^n \rightarrow \mathcal{Y} \times \mathcal{Z}$  by  $M'' = (M, M')$ . Then  $M''$  is  $(\rho + \rho') - zCDP$ .*

**Proposition 4** (Parallel Composition). *Supposed that a mechanism  $A$  consist of a sequence of  $T$  adaptive mechanisms,  $\{A_1, \dots, A_T\}$ , where each  $A_t: \prod_{j=1}^{iter-1} \mathcal{O}_j \times \mathcal{D}_t \rightarrow \mathcal{O}_{iter}$  and  $A_t$  satisfies  $\rho_t - zCDP$ . Let  $\{\mathcal{D}_1, \dots, \mathcal{D}_T\}$  be a randomized partition of the input  $\mathcal{D}$ . The mechanism  $A(\mathcal{D}) = (A_1(\mathcal{D})_1, \dots, A_T(\mathcal{D})_T)$  satisfies  $\frac{1}{T} \sum_{t=1}^T \rho_t - zCDP$ .*

**Theorem 3.** *Set a random mechanism as  $M$ , and we assume that the added Gaussian noise parameter setting conforms to Theorem 2, then the random mechanism satisfies  $(\epsilon - \delta)$ -DP. Let's assume that the total privacy budget each iteration input*

*of a factor matrix is  $\rho_a = \frac{\epsilon^2}{4T\ln(\frac{1}{\delta})}$ , where  $T$  is the total number of achieve converges.*

*Proof.* First, Theorem 2 shows that the random mechanism  $M$  satisfies  $(\epsilon - \delta)$ -DP.

$$f(\mathbf{A}^{[t]}) = \frac{1}{2} \sum_{(i,j,k) \in \Omega} \left( x_{(i,j,k)} - \bar{x}_{(i,j,k)} \right)^2 + \frac{\lambda_1}{2} \|\mathbf{A}^{[t]}\|_2^2, \quad (3)$$

where  $\bar{x}_{(i,j,k)} = \sum_{r=1}^R a_{i,r} b_{j,r} c_{k,r}$ , and the  $L_2$ -sensitivity of this problem is  $\Delta_2 = \|2TL\lambda_1\|$  ( $L$  is the Lipschitz Constant),  $\Delta_2 = 2TL\lambda_1 = 2T \|\sum_{j,k:(i,j,k)} \sum_{r=1}^R (b_{jr}^2 c_{kr}^2)\|_2 \lambda_1$ . The base zCDP parameter is  $\rho_a$ . According to the Proposition 3,  $\mathbf{A}$  costs  $T\rho_a$  in total. According to the Proposition 4, then the total privacy budget cost of all  $N$  users is  $\frac{\sum_{n=1}^N T\rho_n}{N} = \frac{T(\rho_1 + \rho_2 + \dots + \rho_N)}{N} = T\rho_a$ . Finally, according to Proposition 1 and 2, the random mechanism  $M$  satisfies  $(\epsilon - \delta)$ -DP that can be converted to zCDP by setting as  $\rho = T\rho_a$ , where  $\rho_a = \frac{\epsilon^2}{4T\ln(\frac{1}{\delta})}$ .  $\square$

Algorithm 1 consists of three modules. The module ① is the original network traffic data tensor  $\mathcal{X}$  decomposed by CP factorization to obtain three factor matrices, namely  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$ . Then, the corresponding noise is added by module ② or module ③. Module ② is to add noise matrix with the same size as factor matrix  $\mathbf{A}$  under DP strategy based on Laplace mechanism. After updating the elements in factor matrix  $\mathbf{A}$ , Algorithm 1 returns a recovery tensor  $\hat{\mathcal{X}}$ . Module ③ add a noise matrix of the same size as factor matrix  $\mathbf{A}$  based on Gaussian mechanism under zCDP strategy, and Algorithm 1 returns a recovery tensor  $\mathcal{X}$  after updating the elements in factor matrix  $\mathbf{A}$ .

### C. LDP Mechanism

Because the HOHDST network traffic data presents the tensor form, the DP communities try to extend the DP mechanism from MF to STF. So far, only DP and zCDP have been applied to TF just for the privacy protection on dense and symmetric tensor. Thus, the application domain is limited. The theoretic base of both approaches is the assumption that the third-party servers are trustworthy and cannot provide LDP protection mechanism, LDP can address this assumption well. We want to combine the STF for HOHDST network traffic data recovery and privacy protection with LDP. Fig. ?? illustrates that LDP is used to collect the individual user information and used for data publishing or querying.

LDP inherits the combined characteristics of the zCDP and can extend it furthermore. LDP mainly uses the randomized response mechanism to the input noise disturbance. Thus, LDP can resist the privacy attack from the source from the third-party data collector, which is assumed to be untrusted. CDP has serial and parallel propositions (Propositions 3 and 4). The serial combination can allocate privacy budget under different iteration times of the algorithm. The parallel combination ensures that private datasets on disjoint confidential subsets satisfy differential privacy respectively. From the definition, the DP is defined on the neighboring datasets, while the LDP is specified on the two records. However, the form of a privacy

**Algorithm 1:** DPSTF( $\mathcal{X}, R, \varepsilon, \lambda, \rho$ )

**Input:** A third-order tensor  $\mathcal{X}$ , CP-ranks  $R$ , privacy  $\varepsilon$  and parameters  $\lambda, \rho$ .  
**Output:** ① Factor matrices  $\mathbf{A} \in \mathbb{R}^{I \times R}$ ,  $\mathbf{B} \in \mathbb{R}^{J \times R}$ , and  $\mathbf{C} \in \mathbb{R}^{K \times R}$ , ② Noise factor matrix  $\hat{\mathbf{A}} \in \mathbb{R}^{I \times R}$  and recovered tensor  $\hat{\mathcal{X}}$ , ③ Noise factor matrix  $\bar{\mathbf{A}} \in \mathbb{R}^{I \times R}$  and recovered tensor  $\bar{\mathcal{X}}$ .

---

- 1 ① Initialize  $\mathbf{A} \in \mathbb{R}^{I \times R}$ ,  $\mathbf{B} \in \mathbb{R}^{J \times R}$ ,  $\mathbf{C} \in \mathbb{R}^{K \times R}$ ;
- 2 While convergence threshold is not reached or  $t < iter_{max}$  do;
- 3  $t = t + 1$ ;
- 4 Randomly sample element  $x_{ijk}$  from  $\mathcal{X}$ ;
- 5  $x_{ijk}^* = a_r \circ b_r \circ c_r$ ;
- 6 **for**  $(i, j, k) \in \Omega$  **do**
- 7     Compute the gradient  $\frac{\partial f}{\partial a_{i,r}}, \frac{\partial f}{\partial b_{j,r}}, \frac{\partial f}{\partial c_{k,r}}$ ;
- 8     Update  $\tilde{x}_{i,j,k} \leftarrow \sum_{r=1}^R a_{i,r} b_{j,r} c_{k,r}$ ;
- 9 **end**;
- 10 If the convergence condition meets or reaches the maximum number of iterative steps, then outputs the factor matrices  $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ;

---

- 12 ② **for**  $i = 1, \dots, I; r = 1, \dots, R$  **do**
- 13     Generate noise that corresponds to the Laplace distribution  $f(a_{ir}) = \frac{\exp(-\frac{|a_{ir}|}{\lambda})}{2\lambda}$ ;
- 14     Update  $\hat{a}_{ir} \leftarrow a_{ir} + f(a_{ir})$ ;
- 15 **end**;
- 16  $\hat{\mathcal{X}} = \sum_{r=1}^R \hat{a}_{i,r} \circ b_{j,r} \circ c_{k,r}$ ;
- 17 **Return**  $\hat{\mathcal{X}}$  and  $\hat{\mathbf{A}}$ ;

---

- 19 ③ **for**  $i = \{1, \dots, I\}, r = \{1, \dots, R\}$  **do**
- 20     Update  $\bar{\mathbf{A}} \leftarrow \mathbf{A} + \text{Gaussian noise matrix } \mathcal{N}(0, \frac{\Delta^2}{2\rho})$ ;
- 21 **end**;
- 22  $\bar{\mathcal{X}} = \sum_{r=1}^R \bar{a}_r \circ b_r \circ c_r$ ;
- 23 **Return**  $\bar{\mathcal{X}}$  and  $\bar{\mathbf{A}}$ ;

---

guarantee does not change. Therefore, the LDP also continues the serial composition of DP.

**Theorem 4** (Serial Composition on the LDP). *Supposed that a method consists of  $m$  independent random functions  $\mathcal{M} = \{\mathcal{M}_1, \dots, \mathcal{M}_m, \dots, \mathcal{M}_M\}$ , and each function  $\mathcal{M}_m$  satisfies  $\varepsilon_m$ -LDP. Then the method  $\mathcal{M}$  satisfies  $\sum_{m=1}^M \varepsilon_m$ -LDP[27].*

**Lemma 1.** *Let  $u_i$  and  $u'_i$  be any two non-private information matrix as inputs.  $u$  and  $u'$  is the Bernoulli variable generated by the given input  $u_i$  and  $u'_i$ , respectively.  $\tilde{u}_i$  is the output of the algorithm. Then, each submission of the user's information matrix satisfies  $\varepsilon/I$ -LDP. So the noisy information matrix submitted by all the users satisfies the  $\varepsilon$ -LDP.*

*Proof.* Sample a Bernoulli variable  $u$  such that  $\Pr[u = 1] = \frac{u_{j,k}(e^\varepsilon - 1) + e^\varepsilon + 1}{2e^\varepsilon + 2}$ . Assume that  $\frac{\Pr[\tilde{u}_i | u_i]}{\Pr[u' = 1 | u'_i]} \leq \frac{\max_{u_i} \Pr[u = 1 | u_i]}{\min_{u'_i} \Pr[u' = 1 | u'_i]} = \frac{\max_{u_i} ((u_i)_{j,k}(e^{\varepsilon/I} - 1) + e^{\varepsilon/I} + 1)}{\min_{u'_i} ((u'_i)_{j,k}(e^{\varepsilon/I} - 1) + e^{\varepsilon/I} + 1)} = \frac{2e^{\varepsilon/I}}{e^{\varepsilon/I} + 1} \leq e^{\varepsilon/I}$ . It can prove that the noise matrix of each user to submit satisfies  $\varepsilon/I$ -LDP. By Theorem 4, the total noise matrix meets  $\varepsilon$ -LDP.  $\square$

Algorithm 2 adds noise to the user information matrix  $u_i, i = 1, \dots, I$  through the Laplace mechanism or Gaussian mechanism. This disturbs the user's network traffic data at the source and avoids the attack of untrusted third-party servers. The server receives the disturbing user information matrix  $\tilde{u}_i, i = 1, \dots, I$ , aggregates them into a third-order noise tensor  $\tilde{\mathcal{X}}$ , and decomposes it through CP factorization and recovers it into  $\hat{\mathcal{X}}^*$ .

IV. EXPERIMENTAL RESULTS AND ANALYSIS

The experiments are carried on the public traffic trace data *Abilene* [23] and the pan-European research backbone network *GÉANT* [24] to evaluate the performance of MDPSTF framework. *Abilene* network consists of 12 nodes; Thus it will generate 144 original-source pairs. Thus, *Abilene* network contained 144 users, 288 locations and 168 time points. So *Abilene* contains a network traffic tensor data with a size of  $\mathbb{R}^{144 \times 288 \times 168}$ , and the *GÉANT* is a large-scale, symmetric, and dense tensor. It records monitoring data vary in 112 days and containing a network traffic tensor data with a size of  $\mathbb{R}^{272 \times 96 \times 112}$ . Mean Squared Error(MSE), Root Mean Squared Error (RMSE), and Fitness error are defined as:

**Definition 12** (Mean Squared Error (MSE)). *The formula for MSE is  $MSE(\mathcal{X}) = \frac{\sum_{(i,j,k) \in \Gamma} (x_{ijk} - \tilde{x}_{(i,j,k)})^2}{|\Gamma|}$ ; where  $\tilde{x}_{(i,j,k)} = \sum_{r=1}^R a_{i,r} b_{j,r} c_{k,r}$  and  $\Gamma$  is the test set.*

**Definition 13** (Root Mean Squared Error (RMSE)). *The formula for RMSE is  $RMSE(\mathcal{X}) = \sqrt{\frac{\sum_{(i,j,k) \in \Gamma} (x_{ijk} - \tilde{x}_{(i,j,k)})^2}{|\Gamma|}}$  where  $\tilde{x}_{(i,j,k)} = \sum_{r=1}^R a_{i,r} b_{j,r} c_{k,r}$  and  $\Gamma$  is the test set.*

**Definition 14** (Fitness Error). *Fitness error can measuring the recovery error of entries in the tensor[28]. The formula is defined as  $\frac{\sqrt{\sum_{(i,j,k) \in \Omega} (x_{ijk} - \tilde{x}_{(i,j,k)})^2}}{\sqrt{\sum_{(i,j,k) \in \Omega} x_{ijk}^2}}$ .*

First, the performance of CP factorization is tested on the two datasets, i.e., *Abilene*, and *GÉANT*. To investigate the

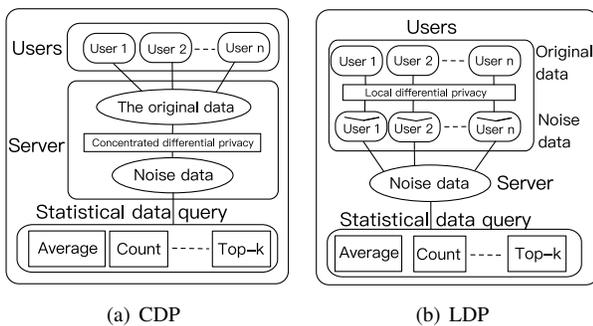


Fig. 2: Data processing framework for CDP and LDP

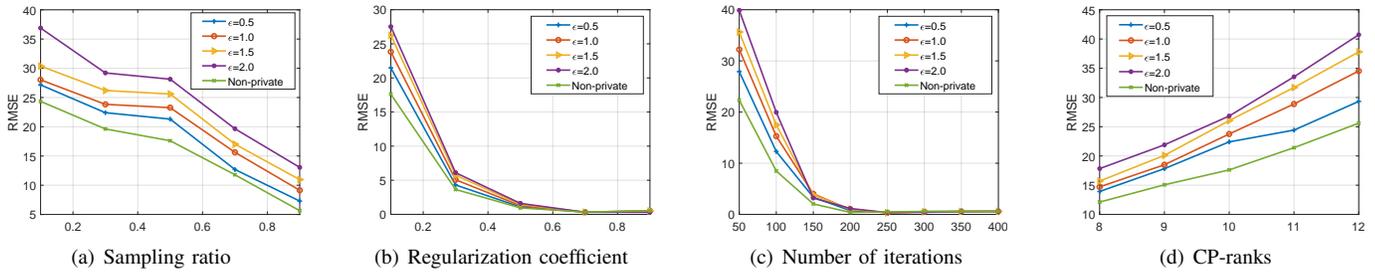


Fig. 3: Various parameter settings for DP in *Abilene* dataset

**Algorithm 2:** LDPSTF( $\mathcal{X}, R, \varepsilon, \lambda, \rho$ )

**Input:** A third-order tensor  $\mathcal{X}$ , CP-ranks  $R$ , privacy  $\varepsilon$  and parameters  $\lambda, \rho$ .

**Output:** Factor matrices  $\mathbf{A} \in \mathbb{R}^{I \times R}$ ,  $\mathbf{B} \in \mathbb{R}^{J \times R}$ , and  $\mathbf{C} \in \mathbb{R}^{K \times R}$ , noise user information matrix  $\check{u}_i \in \mathbb{R}^{J \times K}, i = 1, \dots, I$  and recovered tensor  $\check{\mathcal{X}}^*$ .

```

1 for  $i = 1, \dots, I$  do
2   Initialize  $(\check{u}_i)_{j,k} \in \{0\}^{J \times K}$ ;
3   Randomly generate a probability  $rnd_i$ ;
4    $p = \frac{e^\varepsilon}{e^\varepsilon + n + 1}$ ;
5   if  $rnd_i < p$ : The user transmits  $u_i$  directly;
6   then
7     Generate noise by the Laplace or Gaussian distribution:
8     for  $j = \{1, \dots, J\} \in \Omega, k = \{1, \dots, K\} \in \Omega$  do
9        $f(u_{j,k}) = \frac{\exp(-\frac{|u_{j,k}|}{\lambda})}{2\lambda}$  or  $p(u_{j,k}|\varepsilon) = \frac{1}{\sqrt{2\pi\varepsilon}} e^{-\frac{u_{j,k}^2}{2\varepsilon}}$ ;
10      Update  $\check{u}_{j,k} \leftarrow u_{j,k} + f(u_{j,k})$  or  $p(u_{j,k}|\varepsilon)$ ;
11   end;
12   Aggregate all user information matrix into a third-order noise tensor  $\check{\mathcal{X}} \in \mathbb{R}^{I \times J \times K}$ ;
13   Initialize  $\mathbf{A} \in \mathbb{R}^{I \times R}, \mathbf{B} \in \mathbb{R}^{J \times R}, \mathbf{C} \in \mathbb{R}^{K \times R}$ ;
14   While convergence threshold is not reached or  $t < iter_{max}$  do;
15   t=t+1;
16   Randomly sample element  $\check{x}_{ijk}$  from  $\check{\mathcal{X}}$ ;
17   for  $(i, j, k) \in \Omega$  do
18     Compute the gradient  $\frac{\partial f}{\partial \check{a}_{i,r}}, \frac{\partial f}{\partial \check{b}_{j,r}}, \frac{\partial f}{\partial \check{c}_{k,r}}$ ;
19     Update  $\check{x}_{i,j,k} \leftarrow \sum_{r=1}^R \check{a}_{i,r} \check{b}_{j,r} \check{c}_{k,r}$ ;
20   end;
21   If the convergence condition meets or reaches the maximum number of iterative steps, then outputs the factor matrices  $\check{\mathbf{A}}, \check{\mathbf{B}}, \check{\mathbf{C}}$ ;
22    $\check{\mathcal{X}}^* = \sum_{r=1}^R \check{a}_r \circ \check{b}_r \circ \check{c}_r$  for  $\check{\mathbf{A}}, \check{\mathbf{B}}, \check{\mathbf{C}}$ ;
23   Return  $\check{\mathcal{X}}^*$ .

```

impact of the privacy, this value is set to between 0 and 2, while the other parameters are set to default values (we choose  $\delta = 10^{-4}$ , sampling ratio=50%, CP-ranks=10, regularization coefficient=0.1 and number of iterations=200).

We compare the difference between our methods and other methods in Table III.

TABLE III: Describe of models.

Model	Describe
CP-SGD	CP factorization recovery based on SGD algorithm;
CP-DP	Laplace mechanism of DP based on CP factorization with SGD algorithm;
CP-zCDP	Gaussian mechanism of zCDP based on CP factorization with SGD algorithm;
DPFacT[20]	DPFacT is a distributed tensor factorization method, which enhances differential privacy;
CP-ALS[29]	CP factorization recovery based on Alternating Least Squares optimization algorithm;

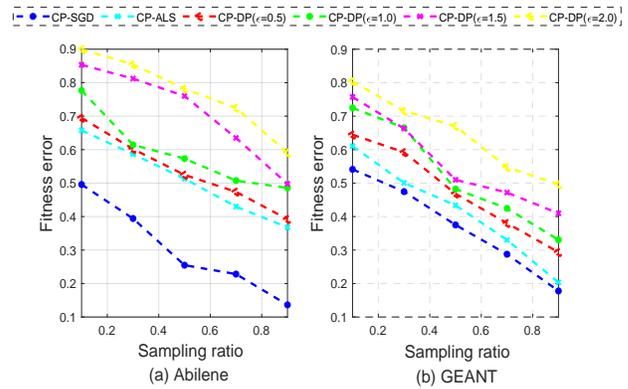


Fig. 4: The fitness error on (a) *Abilene* (b) *GEANT*

We study the effects of sampling ratio, CP-ranks, regularization coefficient and the number of iterations under five different privacy degree schemes. Firstly, with the increase of sampling ratio, regularization coefficient, and the number of iterations in Fig. 3 (a)-(c), the RMSE of the five privacy schemes also decreases. And basically keep the order  $RMSE(\varepsilon = 0.5) < RMSE(\varepsilon = 1.0) < RMSE(\varepsilon = 1.5) < RMSE(\varepsilon = 2.0)$ . In Fig. 3 (d), the RMSE of the five privacy schemes increases with the increase of the CP-ranks.

Fig. 4 shows the performance in terms of fitness error and comparison with others algorithms. With the increase of the sampling ratio thus sample data, the fitness error decrease and thus better recovery performance is obtained; Meanwhile, with

the gradual increase of privacy and the reduction of noise value, the fitness error decreases gradually.

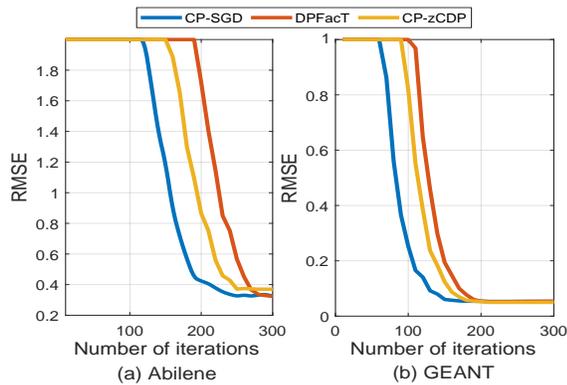


Fig. 5: RMSE on (a) *Abilene* (b) *GEANT*

In Fig. 5, RMSE grows as the number of algorithm iterations increases. By Theorem 3, the total privacy budget is  $(3.035, 10^{-4}), (2.575, 10^{-4})$  under the  $(\epsilon - \delta) - DP$  for *Abilene* and *GEANT* dataset, respectively when DP-zCDP converges.

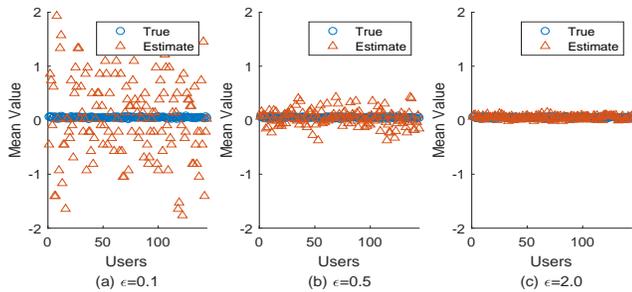


Fig. 6: Mean value estimation under different privacy budget in *Abilene* dataset

As can be seen from Fig. 6 (a), we used mean estimates to observe the impact of privacy. When the privacy is small, the statistical result of the estimated mean value has greatly deviated from the true value. According to Fig. 6 (b) and Fig. 6 (c), when the privacy is large, the statistical result of the estimated mean value is very close to the true value. Because the privacy  $\epsilon$  directly affects the probability of getting a true answer under the randomized response mechanism. As shown in Fig. 7, the larger  $\epsilon$  is, the higher the probability  $p$  of a true answer.

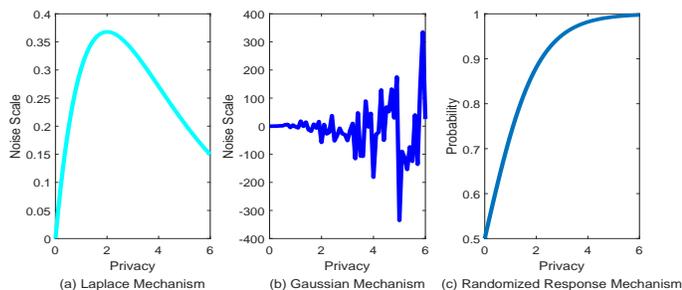


Fig. 7: Functional relation between  $\epsilon$  and Noise scale or  $p$

As the privacy increases, the noise generated by the Laplace mechanism increases first and then decreases gradually in Fig. 7 (a). In Fig. 7 (b), the noise generated by the Gaussian mechanism gradually increases. In addition, it can be seen that when the given privacy is larger, the probability of a true answer is greater in Fig. 7 (c). And also shows that how various privacy can affect the trade-off between data availability and privacy protection. We need to set the privacy parameters appropriately according to the actual different applications.

## V. CONCLUSION

Secure use of network traffic data in the future of 5G networks to provide user privacy is a new concern. Many effective tensor factorization method is proposed, but previous efforts to recover the network traffic data tensor have focused on increasing the computational rate issue. However, there is no privacy protection method for the disclosure of user information in the data center. So the framework for network traffic tensor data privacy protection (MDPSTF) is proposed. The multiple-strategies differential privacy is used for network traffic tensor data, and the experiment is carried out on two real datasets (*Abilene* and *GEANT*). The experimental results show that various privacy budgets have different effects on RMSE and other evaluation indexes. In general, the framework can achieve privacy protection according to different privacy budgets and maintain data availability to a certain extent.

## REFERENCES

- [1] J. Liu, F. Liu, and N. Ansari, "Monitoring and analyzing big traffic data of a large-scale cellular network with hadoop," *IEEE network*, vol. 28, no. 4, pp. 32–39, 2014.
- [2] J. M. Batalla, E. Andrukiewicz, G. P. Gomez, P. Sapienza, C. X. Mavromoustakis, G. Mastorakis, J. Zurek, and M. Imran, "Security risk assessment for 5g networks: National perspective," *IEEE Wireless Communications*, vol. 27, no. 4, pp. 16–22, 2020.
- [3] H. Zhou, D. Zhang, K. Xie, and Y. Chen, "Spatio-temporal tensor completion for imputing missing internet traffic data," in *2015 IEEE 34th international performance computing and communications conference (ipccc)*. IEEE, 2015, pp. 1–7.
- [4] H. Xiao, J. Gao, D. S. Turaga, L. H. Vu, and A. Biem, "Temporal multi-view inconsistency detection for network traffic analysis," in *Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 455–465.
- [5] J. Xiong, J. Ren, L. Chen, Z. Yao, M. Lin, D. Wu, and B. Niu, "Enhancing privacy and availability for data clustering in intelligent electrical service of iot," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1530–1540, 2018.
- [6] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211–407, 2014.
- [7] M. Gong, Y. Xie, K. Pan, K. Feng, and A. K. Qin, "A survey on differentially private machine learning," *IEEE*

- Computational Intelligence Magazine*, vol. 15, pp. 49–64, 2020.
- [8] Y. Wang, Y.-X. Wang, and A. Singh, “Differentially private subspace clustering,” *Advances in Neural Information Processing Systems*, vol. 28, pp. 1000–1008, 2015.
- [9] D. Su, J. Cao, N. Li, E. Bertino, and H. Jin, “Differentially private k-means clustering,” in *Proceedings of the sixth ACM conference on data and application security and privacy*, 2016, pp. 26–37.
- [10] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, “Deep learning with differential privacy,” in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 308–318.
- [11] J. Hua, C. Xia, and S. Zhong, “Differentially private matrix factorization,” in *Proceedings of the 24th International Conference on Artificial Intelligence*. AAAI Press, 2015, p. 1763–1770.
- [12] J. Upadhyay, “The price of privacy for low-rank factorization,” in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 4180–4191.
- [13] H. Shin, S. Kim, J. Shin, and X. Xiao, “Privacy enhanced matrix factorization for recommendation with local differential privacy,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1770–1782, 2018.
- [14] B. Ermiş and A. T. Cemgil, “Data sharing via differentially private coupled matrix factorization,” *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 14, no. 3, pp. 1–27, 2020.
- [15] Y. Wang and A. Anandkumar, “Online and differentially-private tensor decomposition,” in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016, pp. 3539–3547.
- [16] H. Imtiaz and A. D. Sarwate, “Distributed differentially private algorithms for matrix and tensor factorization,” *IEEE journal of selected topics in signal processing*, vol. 12, no. 6, pp. 1449–1464, 2018.
- [17] H. Li, K. G. Li, J. An, and K. G. Li, “An online and scalable model for generalized sparse non-negative matrix factorization in industrial applications on multi-gpu,” *IEEE Transactions on Industrial Informatics*, 2019.
- [18] H. Li, Z. Li, K. Li, J. S. Rellermeyer, L. Y. Chen, and K. Li, “Sgd\_tucker: A novel stochastic optimization strategy for scalable parallel sparse tucker decomposition,” *IEEE Transactions on Parallel and Distributed Systems*, 2020.
- [19] K. Xie, L. Wang, X. Wang, G. Xie, and J. Wen, “Low cost and high accuracy data gathering in wsns with matrix completion,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 7, pp. 1595–1608, 2017.
- [20] J. Ma, Q. Zhang, J. Lou, J. C. Ho, L. Xiong, and X. Jiang, “Privacy-preserving tensor factorization for collaborative health data analysis,” in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 1291–1300.
- [21] X. Nie, L. T. Yang, J. Feng, and S. Zhang, “Differentially private tensor train decomposition in edge-cloud computing for sdn-based internet of things,” *IEEE Internet of Things Journal*, 2019.
- [22] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.
- [23] “The abilene observatory data collections, accessed: Jul. 20, 2004. [online], available: <http://abilene.internet2.edu/observatory/data-collections.html>.”
- [24] S. Uhlig, B. Quoitin, J. Lepropre, and S. Balon, “Providing public intradomain traffic matrices to the research community,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, pp. 83–86, 2006.
- [25] C. Dwork and G. N. Rothblum, “Concentrated differential privacy,” 2016.
- [26] M. Bun and T. Steinke, “Concentrated differential privacy: Simplifications, extensions, and lower bounds,” in *In Theory of Cryptography Conference*. Springer, 2016, pp. 4–15.
- [27] D. Kortenkamp, T. Milam, R. Simmons, and J. L. Fernandez, “Collecting and analyzing data from distributed control programs,” *Electronic Notes in Theoretical Computer Science*, vol. 55, no. 2, pp. 236–254, 2001.
- [28] K. Xie, L. Wang, X. Wang, G. Xie, J. Wen, and G. Zhang, “Accurate recovery of internet traffic data: A tensor completion approach,” in *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 2016, pp. 1–9.
- [29] C. Battaglino, G. Ballard, and T. G. Kolda, “A practical randomized cp tensor decomposition,” *SIAM Journal on Matrix Analysis and Applications*, 2017.



**Jin Wang** received the M.S. degree from Nanjing University of Posts and Telecommunications, China in 2005. He received Ph.D. degree from Kyung Hee University Korea in 2010. Now, he is a professor at Changsha University of Science and Technology. He has published more than 400 international journal and conference papers. His research interests mainly include wireless ad hoc and sensor network, network performance analysis and optimization etc. He is a senior member of the IEEE and a Fellow of IET.



**Hui Han** received the B.S. degree in Mathematics and Applied Mathematics from Changsha University, Changsha, China, in 2019. She is currently working toward the M.S. degree in soft engineering with Changsha University of Science and Technology, Changsha, China. Her research interests include tensor decomposition and differential privacy.



**Hao Li** is currently working toward the Ph.D. degree at Hunan University, China and he is a visiting Ph.D. student in TU Delft, 2019-2021. His research interests are mainly in large-scale sparse matrix and tensor factorization, recommender systems, machine learning, and parallel and distributed computing. He has published several journal and conference papers in IEEE-TPDS, InforSci, IEEE-TII, ACM-TDS, ACM CIKM, and IEEE ISPA. He also serves reviewer of the top-tier conferences and journals, e.g., HPCC, IJCAI, WWW, Neurocomputing, IEEE

Access, JPDC, InforSCI, IEEE TNSM, IoT, ACM TKDD and IEEE TDSC.



**Lydia Chen** is an Associate Professor in the Department of Computer Science at the Delft University of Technology. She received Ph.D. from the Pennsylvania State University and B.A. from National Taiwan University in 2006, and 2002, respectively. Her research interests center around dependability management, resource allocation and privacy enhancement for large scale data processing systems and services. More specifically, her work focuses on developing stochastic and machine learning models and applying these techniques to application

domains, such as datacenters and AI systems. She has published more than 80 papers in journals, e.g., IEEE Transactions on Distributed Systems, IEEE Transactions on Service Computing, and conference proceedings, e.g., INFOCOM, Sigmetrics, DSN, and Eurosys. She was a co-recipient of the best paper awards at CCgrid'15 and eEnergy'15. She received TU Delft Professor fellowship in 2018. She was program co-chair for Middleware Industry Track 2017 and IEEE ICAC 2019 and track vice-chair for ICDCS 2018. She has served on the editorial boards of IEEE Transactions on Network and Service Management, IEEE Transactions on Service Computing, IEEE Transactions on Dependable and Secure Computing and IEEE Transactions on Parallel and Distributed Systems. She is a senior IEEE member.



**Shiming He** received the B.S. degree in information security, and the Ph.D. degree in computer science and technology from Hunan University, China, in 2006 and 2013, respectively. She is currently an Associated Professor with the School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, China. Her research interests include machine learning, data analysis, and anomaly detection.



**Dr. Pradip Kumar Sharma** is an Assistant Professor of Cybersecurity in the Department of Computing Science at the University of Aberdeen, UK. He received his Ph.D. in CSE (August 2019) from the Seoul National University of Science and Technology, South Korea. He also worked as a Postdoctoral Research Fellow in the Department of Multimedia Engineering at the Dongguk University, South Korea. He was a Software Engineer at MAQ Software, India, and involved on variety of projects, proficient in building largescale complex data warehouses,

OLAP models, and reporting solutions that meet business objectives and align IT with business. He has published many technical research papers in leading journals from IEEE, Elsevier, Springer, MDPI, etc. Some of his research findings are published in the most cited journals. He has been an expert reviewer for IEEE Transactions, Elsevier, Springer, and MDPI journals and magazines. He is listed in the world's Top 2 Scientists for citation impact during the calendar year 2019 by Stanford University. Also, he received a top 1 reviewer in computer science by Publons Peer Review Awards 2018 and 2019, Clarivate Analytics. He has also been invited to serve as the technical programme committee member and chair in several reputed international conferences such as IEEE DASC 2021, IEEE CNCC 2021, CSA 2022, IEEE ICC2019, IEEE MENACOMM'19, 3ICT 2019, etc. Currently, he is Associate Editor of Peer-to-Peer Networking and Applications (PPNA), Human-centric Computing and Information Sciences (HCIS), Electronics (MDPI), and Journal of Information Processing Systems (JIPS) journals. He has been serving as a Guest Editor for international journals of certain publishers such as IEEE, Elsevier, Springer, MDPI, JIPS, etc. His current research interests are focused on the areas of Cybersecurity, Blockchain, Edge computing, SDN, and IoT security