Joint Distribution Analysis for Set-Valued Data with Local Differential Privacy

Yaxuan Huang, Kaiping Xue, *Senior Member, IEEE*, Bin Zhu, David S.L. Wei, *Life Senior Member, IEEE*, Qibin Sun, *Fellow, IEEE*, Jun Lu

***Abstract*—Set-valued data are commonly used to represent subsets of a universal set and are frequently utilized in online services, such as online shopping preferences, website browsing records, and recently visited places. By collecting set-valued data from users, service providers can perform statistical analysis to obtain a joint distribution of service usage data and subsequently learn the association between different kinds of set-valued data to improve the quality of service. However, collecting set-valued data raises privacy concerns about the potential misuse of records to infer individuals' identities and preferences. Although some privacy-preserving aggregation mechanisms for set-valued data have been proposed, they have not yet achieved joint distribution analysis with high accuracy. In this paper, we propose a joint distribution analysis method for set-valued data with local differential privacy (LDP). We design a scalable perturbation mechanism under ϵ-LDP by limiting the range of users' responses in the collection process and cyclically shifting the set-valued data in an encoded uniform format, ensuring that the size of the universal set does not influence the accuracy of the results. Based on the perturbation method, we develop an analysis method to efficiently obtain association information between two sets. By performing specific bitwise operations on the perturbed data matrices, the computational overhead is linear with respect to the cardinality of the item set. In addition to theoretically analyzing the error bound and proving the security of our work, extensive experimental results on synthetic and real-world datasets demonstrate that our scheme achieves better utility than existing state-of-the-art approaches.**

***Index Terms*—local differential privacy, set-valued data, privacy preservation, joint distribution.**

I. Introduction

Set-valued data, which represent subsets of a universal set, play a pivotal role in online services [1]. Examples of set-valued data include records of online shopping, website browsing, food ordering, and recently visited places. These data can be used to improve the quality of service through big data analysis. In big data analysis, obtaining the joint distribution of specific combinations is often necessary. For instance, advertisements frequently show that users who have bought *A* often also like *B*.

However, precise set-valued data collection raises privacy concerns about the potential misuse of records to infer individuals' identities and preferences. For example, adversaries can infer income levels from online shopping records or deduce home addresses from recently visited places. Due to these privacy concerns, many countries and regions have enacted laws and regulations to protect citizens' right to privacy, such as the *GDPR* [2] in the European Union and the *UPDPA* [3] in the USA. Therefore, it is urgent to protect privacy while analyzing set-valued data.

Many privacy-preserving computing methods have been proposed recently to collect and analyze data while preserving privacy. Compared to other privacy-preserving computing technologies such as homomorphic encryption and secure multi-party computation (MPC), local differential privacy (LDP)-based methods offer higher computational efficiency and quantifiable privacy protection. Additionally, LDP-based methods preserve data privacy without the need for a trusted party, making them very practical in many real-world scenarios. Numerous LDP-based protocols have been applied in various fields, including IoT applications [4]–[6], edge computing [7], [8], social networks [9]–[11], data mining [12], [13], and machine learning [14]–[16]. In particular, LDP provides various estimation functions for different kinds of data, such as the simple average of numeric data, the frequency of categorical data, succinct histograms, and heavy hitters. Generally, data aggregation protocols with LDP preserve privacy even when users do not trust the aggregator. Users perturb the original data locally before uploading it to the aggregator, who then aggregates the collected data to reduce the impact of the perturbation.

Although LDP provides a practical way to preserve privacy in the data collection process, most studies in the field of set-valued data analysis still focus on item distribution estimation [17], [18] or heavy hitter detection [19], [20] for a single set. The joint distribution analysis mechanism has yet to be established between two sets of set-valued data. Moreover, these previous works cannot be directly applied to joint distribution analysis through simple adaptation due to three challenges that need to be addressed:

1) **Maintenance of relevance**: Perturbed set-valued data often lose the relevance between different sets, i.e., the same user's set-valued data from two different sets are perturbed independently, which results in low utility of the results. Maintaining the relationship between the two sets in the analysis of joint probability is a key issue. It is difficult to achieve probability calibration without destroying the relationship between the two sets.
2) **Heterogeneous sizes**: There can be many subsets of a universal set, meaning users' set-valued data may

Y. Huang, K. Xue, B. Zhu, Q. Sun and J. Lu are with the School of Cyber Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, China.

D. Wei is with the Department of Computer and Information Science, Fordham University, Bronx, NY 10458 USA.

J. Lu is also with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, Anhui 230027, China.

Corresponding Author: K. Xue (kpxue@ustc.edu.cn)

contain different items. Single-set analysis uses padding to solve the issue of inconsistent numbers of items. However, in the case of a two-set analysis, padding may change the relevance between the two sets and introduce additional error into the results. Therefore, new methods need to be developed to address the problem of heterogeneous sizes.

3) **Low accuracy**: Even with some trivial methods transforming a two-set analysis into a single-set analysis, the accuracy of the result may not be satisfactory. The accuracy of some single-set analysis methods decreases as the set cardinality increases. The trivial adaptation turns the problem domain into the Cartesian product of two sets, and the privacy budget is divided into more parts according to the sequential composition theorem of LDP, further decreasing accuracy.

To overcome these challenges, we propose a joint distribution analysis method for set-valued data with local differential privacy. Specifically, each user's set-valued data from two sets is encoded into a uniform bit string format and cyclically shifted using our perturbation mechanism. The server collects the perturbed set-valued data and obtains the association information of the two sets through our aggregation method. Subsequently, the server can efficiently derive calibrated joint distribution probabilities with high accuracy based on the privacy parameters and a calibration matrix.

To summarize, this paper makes the following contributions:

- Addressing the challenge of heterogeneous size: We designed a set-valued data perturbation mechanism that cyclically shifts encoded set-valued data, providing the same level of privacy preservation regardless of the number of items a user owns.
- Improving the accuracy of joint distribution estimation: We incorporated the idea of limiting the response range into the perturbation mechanism, ensuring that the cardinality of the set does not affect the accuracy of the results.
- Maintaining the relevance of set-valued data from two sets: We propose an aggregation method that transforms the rows and columns of the binary matrix, allowing the relationship between set-valued data from different sets to be associated for further calibration.
- Theoretical analysis and experimental validation: In addition to theoretically analyzing the error bound and proving the security of our work, extensive experimental results on synthetic and real-world datasets demonstrate that our scheme achieves better utility than state-of-the-art approaches.

The remainder of this paper is organized as follows. Section II introduces the related work, including existing works on LDP and set-valued data analysis. In Section III, we formally describe the definitions and theorems of LDP/set-valued data. The problem statement, including the system model, security assumptions, and our design goals, is given in Section IV. We present the details of our scheme for joint distribution estimation for set-valued data in Section V. Theoretical analysis and experimental results are provided in Sections VI and VII, respectively. Finally, Section VIII concludes this paper.

II. Related Work

Frequency estimation of categorical data. Numerous studies have focused on privacy-preserving statistical analysis using LDP. Among these, frequency estimation of categorical data, a primary statistical function, garnered significant attention. Kairouz *et al.* [21] proposed a classical frequency estimation method, k-RR, for categorical data, extending BRR [22] to categorical attributes with an arbitrary number of possible values. RAPPOR [23], presented by Erlingsson *et al.*, is well-known for longitudinal privacy-preserving data collections, where data is collected multiple times. RAPPOR employs Bloom filters to transform a sensitive string based on a set of hash functions, and then uses a two-step random response mechanism to preserve users' long-term privacy. To improve accuracy, Wang *et al.* [24] introduced the Optimized Local Hashing (OLH) protocol, which mitigates information loss between the hashing and randomization steps. Additionally, studies such as [25] and [26] focus on joint distributions of categorical data, which are similar to our work. Xue *et al.* [25] proposed JESS to learn joint distributions and used it to train a privacy-preserving Naïve Bayes classifier. Xu *et al.* [26] introduced the notion of user-level LDP to formalize and preserve users' privacy when their joint data tuples are released. However, due to differences in data types, leading to differences in privacy budget allocation and perturbation methods, joint distribution analysis methods for categorical data cannot be applied to set-valued data.

Heavy hitters of set-valued data. Compared to basic categorical data, set-valued data is more complex, and its estimation has been a focus of researchers in recent years. Most research on analyzing set-valued data was conducted in the area of heavy hitters identification (also known as frequent items mining), which is suitable for scenarios where only the frequencies of frequent items need to be calculated. LDPMiner [19] is the first work that provides heavy hitter estimation over set-valued data. Based on RAPPOR [23] and S-Hist [27], LDPMiner pads users' items to a uniform length, addressing the difficulty of heterogeneous sizes. Wang *et al.* [28] formally defined such padding-and-sample-based frequency oracles and proposed SVIM for finding frequent items in the set-valued LDP setting. To address challenges related to different item quantities among users and to improve utility, Zhu *et al.* [29] combined sampling and shuffling, designing a top-k frequent item estimation framework called EPS^2. To efficiently identify heavy hitters from set-valued data with a large domain, PemSet [20] only perturbs and reports prefixes of users' data, reducing computation cost. Due to different computational tasks, these works cannot be applied to frequency estimation of all items in set-valued data.

Frequency estimation of set-valued data. In the research on set-valued data, frequency estimation is more relevant to our work. In frequency estimation, the focus is on all items, rather than just the frequent items, as in heavy hitters.

To improve accuracy, Wang *et al.* proposed PrivSet [17], which privatizes items in set-valued data as a whole, and the Wheel mechanism [18], [30], maping set-valued data to numerical values. However, these works are limited to single-set estimation. To the best of our knowledge, no work has studied joint distribution estimation of set-valued data under LDP. Therefore, we propose to address the three specific challenges mentioned in Section I and achieve high-accuracy, privacy-preserving joint distribution estimation for set-valued data.

III. Preliminaries

A. Local Differential Privacy (LDP)

As a convincing privacy-preserving computing technique, differential privacy (DP) [31] has been proposed for more than 10 years. DP is independent of the adversaries' background knowledge and has excellent provable mathematical security. DP includes local differential privacy (LDP) and centralized differential privacy (CDP). The works of CDP [32]–[35] are based on the premise that there is a trusted centralized server for aggregation. However, in real life, this security assumption is often not met, and thus, LDP is favored. LDP protocols preserve privacy in scenarios where users do not trust the aggregator [36]–[40], and users perturb the original data locally before uploading it to the aggregator. The aggregator then aggregates the collected data to mitigate the impact of the perturbation.

Formally, ϵ-local differential privacy is defined on a randomized mechanism $\mathcal{M}$ and a privacy budget $\epsilon > 0$ as follows.

Definition 1 (*ϵ-Local Differential Privacy*) [41]. A randomized mechanism $\mathcal{M}$ satisfies ϵ-LDP if and only if for any pair of input values v and v' in the domain of $\mathcal{M}$, and for any possible output $y \in \mathcal{Y}$, the following condition holds:

$$\mathbb{P}[\mathcal{M}(v) = y] \leq \mathrm{e}^{\epsilon} \cdot \mathbb{P}[\mathcal{M}(v') = y],$$

where $\mathbb{P}[\cdot]$ denotes the probability and ϵ is the privacy budget. A smaller ϵ means stronger privacy protection and lower accuracy of aggregation results.

The idea of LDP is that any output y should be about as likely regardless of the individual's secret, while centralized differential privacy focuses more on ensuring that any output should be about as likely regardless of whether an individual's data is in the dataset. The degree of "regardless" is controlled by the privacy budget ϵ.

Similar to centralized differential privacy, LDP also has the composition theorem, which is widely used in the design of mechanisms. The composition theorem primarily guarantees the overall LDP for the combination of sequential algorithms that each satisfies LDP individually.

Theorem 1 (*Sequential Composition Theorem*) [41]. Suppose that a set of privacy mechanisms $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, ..., \mathcal{M}_m\}$ are sequentially performed on a dataset, and each $\mathcal{M}_i$ provides an ϵ_i-LDP guarantee. Then, $\mathcal{M}$ can provide $(\sum_{i=1}^{m} \epsilon_i)$-LDP.

Intuitively, when a set of randomized mechanisms is performed sequentially on a dataset, the final privacy guarantee is determined by the summation of total privacy budgets.

Binary Randomized Response (BRR) [21], [22] is a primary randomization method for achieving LDP for binary values. Its main idea is akin to flipping a biased coin: if tails, then respond truthfully; otherwise, respond falsely. Formally, the original value v is perturbed into v^* by

$$\mathbb{P}\left(\mathcal{M}(v) = v^*\right) = \begin{cases} \frac{\mathrm{e}^{\epsilon}}{\mathrm{e}^{\epsilon}+1}, & \text{if } v^* = v, \\ \frac{1}{\mathrm{e}^{\epsilon}+1}, & \text{if } v^* \neq v. \end{cases} \tag{1}$$

Since the coin is biased, the truth cannot be directly obtained from all perturbed answers. One more step is needed to calibrate the answer, which is also common in other LDP mechanisms. For BRR, the estimated true frequency $\hat{f}$ can be calibrated from directly collected frequency f as follows:

$$\hat{f} = \frac{f + p - 1}{2p - 1}. \tag{2}$$

Because BRR has the advantageous property of simple randomization satisfying LDP, many previous works [42]–[44] are conducted based on its method and achieve more complex functions.

B. Set-valued Data

The set-valued data describes subsets of a universal set. Formally, $\mathcal{U} = \{x_1, x_2, ..., x_c\}$ is the domain of items, and the set-valued data v_i of user i is denoted as a subset of $\mathcal{U}$, i.e., $v_i \subset \mathcal{U}$. Different users may have different numbers of items, so their set-valued data may be different subsets of the universal set. To exemplify this, Table I shows an example of a set-valued dataset of five users with the item domain $\mathcal{U} = \{hamburger, pizza, cola, frenchfries\}$ as food orders.

Table I: An Example of Set-Valued Dataset

Users' Data	Items of Set-Valued Data
$\boldsymbol{v}_1$	$\{pizza, cola\}$
$\boldsymbol{v}_2$	$\{hamburger, cola, frenchfries\}$
$\boldsymbol{v}_3$	$\{frenchfries\}$
$\boldsymbol{v}_4$	$\{hamburger, pizza, cola\}$
$\boldsymbol{v}_5$	$\{hamburger, cola\}$

The important notations frequently used throughout this paper are listed in Table II. The data aggregator needs to estimate some statistics of these set-valued data from all users. The set-valued data analysis in previous studies and this paper respectively focused on:

- **Single-set item distribution estimation**. The frequency of each item in all users' set-valued data is formally defined as:
$$f_j = \frac{\left|\{u_i | v_{i,j}^1 = 1\}\right|}{n},$$
where f_j is the proportion of users whose set-valued data contains item j. For example, this could represent the proportion of people who ordered *cola* in the previous example.
- **Two-set joint distribution estimation**. The joint distribution is defined as the frequency of every possible

combination of two items from two sets respectively, or formally we have:

$$f_{i,j} = \frac{\left|\{u_k|(v_{k,i}^1 = 1) \wedge (v_{k,j}^2 = 1)\}\right|}{n},$$

where $f_{i,j}$ is the proportion of users whose set-valued data contains item i (from the first set) and item j (from the second set). For example, if the food set is divided into a staple food set and a snack set, we need to obtain the proportion of people who ordered both *hamburger* and *french fries*. Furthermore, we can infer what items may best match a *hamburger* and help enterprises improve their marketing strategy.

Table II: The List of Notations

Notation	Description
$\mathcal{U}^1$,$\mathcal{U}^2$	Two sets of the item domain of user data
c	The cardinality of the item set
n	The total number of users
l	The parameter of cyclic shift function
u_i	The i-th user
ϵ	The privacy budget
$\boldsymbol{v}_i^1$,$\boldsymbol{v}_i^2$	The set-valued data (binary vector) of user i, from set 1 and set 2, respectively
$\boldsymbol{s}_i^1$, $\boldsymbol{s}_i^2$	The perturbed set-valued data (binary vector) of user i, from $\boldsymbol{v}_i^1$ and $\boldsymbol{v}_i^2$, respectively
$v_{i,j}^1$, $v_{i,j}^2$	The binary indicator of item j of user i's set-valued data, in $\boldsymbol{v}_i^1$ and $\boldsymbol{v}_i^2$, respectively
$s_{i,j}^1$, $s_{i,j}^2$	The perturbed binary indicator of item j of user i's set-valued data, in $\boldsymbol{s}_i^1$ and $\boldsymbol{s}_i^2$, respectively
S^1, S^2	The perturbed data matrix of each set
S^*	The aggregated perturbed data matrix
$\hat{F}$	The calibrated joint distribution frequency matrix
M	The calibration matrix
$f_{i,j}^*$, $\hat{f}_{i,j}$, $f_{i,j}$	The perturbed/estimated/true joint frequency of item i in the first set and item j in the second set

IV. Problem Statement

A. System Model

We consider a client-server architecture consisting of a server and a group of participating users. The server is the service provider of a certain application and is responsible for aggregating the perturbed set-valued data submitted by users and estimating the joint distribution. The participants are the application users.

In this article, we assume that there are n users (denoted as $u_1, u_2, \ldots, u_n$) who own set-valued data from two different universal sets $\mathcal{U}^1$ and $\mathcal{U}^2$. We assume that the application can execute a simple perturbation algorithm automatically on original usage data before it is uploaded to the server, and the server can only receive the perturbed set-valued data from the application. The server can perform some complex computing tasks to obtain the joint distribution of set-valued data.

B. Security Assumption

Application service providers always strictly establish service agreements and use licenses that have legal effects, so the server is assumed to be honest but curious. Specifically, the server processes the data according to the data aggregation protocol, but it is also interested in the users' privacy. We treat the users as honest participants who perturb the original data according to the protocol, and they are concerned about their data privacy. Additionally, we assume that there exist secure communication channels (i.e., the TLS/SSL protocol) between users and the server.

C. Design Goals

In this paper, we intend to devise a joint distribution analysis method for set-valued data with LDP. Our scheme aims to achieve the following design goals:

- *Privacy Preservation*. Users' input and output set-valued data strictly satisfy the definition of LDP. Moreover, the original set-valued data are well protected from disclosure to any other participants throughout the entire process.
- *Accuracy*. The expectation of the estimated joint probabilities converges to the true joint probability. Compared with adaptions of state-of-the-art works on set-valued data, the error bound of our scheme is optimal.
- *Scalability*. The size of the universal set does not influence the accuracy of results in our scheme, making it suitable for scenarios with large-scale item sets.

V. Proposed Scheme

A. Overview

In this section, we present a detailed joint distribution scheme for set-valued data estimation, comprising a set-valued data perturbation protocol for each user, a specialized aggregation method for perturbed set-valued data that preserves two-set relationships, and a calibration step for the server.

Similar to previous works based on frequency statistics with LDP, our set-valued data perturbation protocol also utilizes the randomized response method. Instead of randomly responding with items in set-valued data one by one, we limit the range of responses to improve aggregation accuracy. Specifically, users encode all chosen items into a uniform format bit string and respond with either the original string or a circularly shifted string, in accordance with the LDP definition.

The server then aggregates the perturbed set-valued data submitted by users to obtain the association information of two sets. By performing specific bit operations on users' set-valued data in the form of bit strings, the server can obtain aggregated information in linear time to the item set cardinality.

Since the aggregated results in the previous step are based on perturbed set-valued data, calibration is necessary. The server generates a frequency calibration matrix offline, meaning it can be generated without interaction with the user. The server then obtains the calibrated joint distribution through matrix division.

In the following three subsections, we detail our set-valued data randomization protocol, the aggregation method

that preserves two-set relationships, and the joint distribution estimation mechanism.

B. Data Randomization

As mentioned earlier, heterogeneous sizes pose a challenge in set-valued data joint distribution analysis. A straightforward approach is to encode users' choices into bit strings. Specifically, let $v_{i,j}$ denote whether user i has chosen item j from the set, where 1 indicates the user has chosen item j, and 0 indicates they have not. The simplest method to perturb data would be flipping each bit via BRR. However, we avoid this due to its low accuracy. Instead, we limit the range of responses to improve aggregation accuracy.

Additionally, $v_{i,j}^1$ and $v_{i,k}^2$ represent the binary indicators of items j and k of user i's set-valued data from sets 1 and 2, respectively. After encoding as described, user i has binary indicator vectors from the two sets: $\boldsymbol{v}_i^1$ and $\boldsymbol{v}_i^2$. These binary vectors serve as inputs to the set-valued data perturbation protocol, and the outputs of the protocol are $\boldsymbol{s}_i^1$ and $\boldsymbol{s}_i^2$, maintaining the same format as the input data. Algorithm 1 outlines the major steps of the protocol.

Algorithm 1: Set-Valued Data Perturbation Protocol

Input : the set-valued data of user i: $\{\boldsymbol{v}_i^1,\boldsymbol{v}_i^2\}$
the privacy budget: ϵ
Output: user i's perturbed set-valued data that satisfies ϵ-LDP: $\{\boldsymbol{s}_i^1,\boldsymbol{s}_i^2\}$
for $k \leftarrow 1$ *to* 2 **do**

$$\boldsymbol{s}_i^k = \begin{cases} \boldsymbol{v}_i^k, & \text{w.p. } p = \frac{\mathrm{e}^{\epsilon/2}}{\mathrm{e}^{\epsilon/2}+1}, \\ \mathrm{circshift}(\boldsymbol{v}_i^k, c-l), & \text{w.p. } q = \frac{1}{\mathrm{e}^{\epsilon/2}+1}. \end{cases} \tag{3}$$

end
Return $\{\boldsymbol{s}_i^1,\boldsymbol{s}_i^2\}$.

The main concept in Algorithm 1 is to limit the range of responses by responding with either the original set-valued data or a circularly shifted answer with probabilities given by Eq. 3. The probabilities in Eq. 3 are similar to those in BRR (i.e., Eq. 1), but with a privacy budget of $\epsilon/2$. This halving of the privacy budget is due to both sets of data undergoing the same but independent perturbations. According to Theorem 1, the joint frequency satisfies ϵ-LDP.

Specifically, in Algorithm 1, *cirshift* refers to the circular shift function [45], where the first parameter is the data to be shifted, and the second parameter is the number of bits to shift. In the process of randomized response, the data $\boldsymbol{s}_i^k$ to be uploaded by the user is either the original data $\boldsymbol{v}_i^k$ or $\mathrm{circshift}(\boldsymbol{v}_i^k, c-l)$. In this case, $\mathrm{circshift}(S^2, c+1-i, 2)$ refers to circularly shifting the matrix S^2 by $c+1-i$ positions along the second dimension. It is crucial that the direction of the shift, whether left or right, is consistent across all algorithms presented in this paper.

The shift parameter l is a constant across all users and serves as an intermediate parameter in our algorithm to control the shift bits during perturbation. Importantly, l is independent of the number of items in each user's sets, with the items belonging to full sets of size c. To ensure the algorithm's security, the server must specify parameter l under the following conditions:

- The shift parameter l must satisfy $0 < l < c$ and $l \nmid c$.
- Users' set-valued data from each set must neither include all items in the entire domain nor be empty.

These conditions ensure that the shifted set-valued data differs from the original, allowing the randomization algorithm to meet the definition of LDP. Specifically, $0 < l < c$ is essential for normalizing l. In cyclic shifting, moving a string of length c by l positions is equivalent to moving it by l plus any multiple of c. Normalizing l ensures consistent execution of the algorithm and maintains contextual uniformity. Besides, $l \nmid c$ is crucial for preserving user privacy, assuming the user's set-valued data conforms to formatting criterion. This condition ensures that the representation of the data before and after shifting differs, which guarantees that the algorithm's input and output comply with the definition of LDP.

C. Data Collection

To address the challenge of relevance maintenance indicated in Section I, we design an aggregation algorithm to collect set-valued data from two sets in linear time to the item sets cardinality c. Thanks to the development of efficient matrix and vector computing tools (e.g., MATLAB), our algorithm can obtain the association information of the two sets in linear time with respect to the item sets cardinality without involving complex nested loops.

Algorithm 2: Set-Valued Data Aggregation Mechanism

Input : the perturbed set-valued data of users: $\{\boldsymbol{s}_i^1,\boldsymbol{s}_i^2\}_{i=1}^n$
Output: the aggregated perturbed data matrix: S^*
$S^1 = [\boldsymbol{s}_1^1; \boldsymbol{s}_2^1; ...; \boldsymbol{s}_N^1]$;
$S^2 = [\boldsymbol{s}_1^2; \boldsymbol{s}_2^2; ...; \boldsymbol{s}_N^2]$;
$counter = [0]^{c\times c}$;
for $i \leftarrow 1$ *to* c **do**
  $counter(i,:) =$
  $\mathrm{sum}(S^1 \ \&\ \mathrm{circshift}(S^2, c+1-i, 2))$;
end
$S^* = [0]^{c\times c}$;
for $j \leftarrow 1$ *to* c **do**
  $S^*(:,j) = \mathrm{circshift}(counter(:,j), j-1)$;
end
Return S^*.

As demonstrated in Fig. 1, Algorithm 2 is primarily used to link the set-valued data of the two sets, which are perturbed independently, and to record the perturbed joint frequency. In the example shown in Fig. 1, five users' perturbed set-valued data are collected into S^1 and S^2. In the output matrix S^* of Algorithm 2, column numbers represent items in the first set, and row numbers represent items in the second set. The element at the corresponding coordinate is the number of users whose perturbed set-valued data items are marked as "1" in each set.

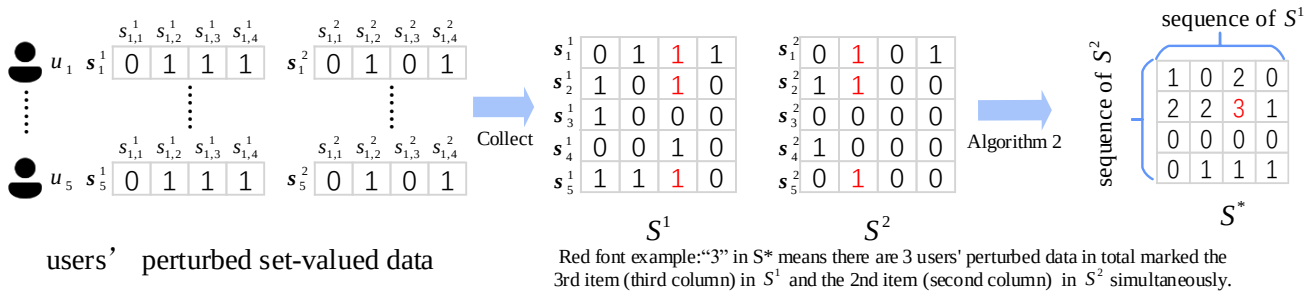


Fig. 1: An Example of 5 users' perturbed set-valued data with a set cardinality of 4.

The main idea of Algorithm 2 is that, with the convenience of binary computing, the information of the two matrices can be associated efficiently through cyclic shift and the Boolean AND operation. Specifically, we use *counter* to record the intermediate aggregated result. In the first loop body, "&" denotes the Boolean AND operator, and "$counter(i,:)$" represents the i-th row of $counter$. As a supplement to the *circshift* function described in the previous subsection, if the operand is a matrix, the function's third parameter is optional: "2" means to horizontally shift the columns of the matrix, and the default "1" means to shift rows vertically. In short, each row of $counter$ is the column sum of the Boolean AND results of S^1 and the cyclically shifted S^2. Finally, S^* can be obtained by cyclically shifting $counter$.

D. Joint Distribution Estimation

Since Algorithm 2 obtains the aggregated perturbed data, we also need to adjust the results to obtain the correct estimated joint frequency. Similar to the derivation of Eq. 2 from Eq. 1, according to Algorithm 1, we have:

$$f^*_{i,j} = \hat{f}_{i,j} \cdot p^2 + (\hat{f}_{i,j+l} + \hat{f}_{i+l,j}) \cdot p \cdot q + \hat{f}_{i+l,j+l} \cdot q^2. \quad (4)$$

It is worth noting that subscripts in Eq. 4 must be taken modulo c. For instance, if $i+l$ is larger than c, the actual subscript should be $i+l-c$. Moreover, the probabilities p and q can be learned from Eq. 3 during the process of data randomization.

The main idea of Algorithm 3 is to solve the recursive equations described in Eq. 4 using matrix division. Based on the concept of Eq. 4, Algorithm 3 first generates a calibration matrix M. Although the generation process of M appears complex, it can be completed prior to set-valued data collection. This calibration matrix is fixed and can be reused for the same privacy budget and set cardinality.

After generating M, in the penultimate line of Algorithm 3, the calibrated joint aggregated data can be derived by performing a matrix left division. The two operands of the left division are the calibration matrix M and the vectorized S^* ($S^*(:)$ denotes arranging the columns of the S^* matrix into a new column vector). Additionally, to obtain the calibrated joint distribution frequency matrix $\hat{F}$, we need to divide the result of the left division by the total number of users n.

Algorithm 3: Calibration Mechanism for Joint Distribution Probabilities

Input : the aggregated perturbed data matrix S^*
the privacy budget: ϵ
Output: the calibrated joint distribution frequency matrix $\hat{F}$

$M = [0]^{c^2 \times c^2}$;
// p, q are refered to Eq.3
for $i \leftarrow 1$ *to* c **do**
  for $j \leftarrow 1$ *to* c **do**
    $k = (i-1) \cdot c + j$;
    $M(k,k) = p \cdot p$;
    if $i + l \leq c$ **then**
      $M(k, (i-1+l) \cdot c + j) = p \cdot q$;
      if $j + l \leq c$ **then**
        $M(k, k+l) = p \cdot q$;
        $M(k, (i-1+l) \cdot c + j + l) = q \cdot q$;
      else
        $M(k, k+l-c) = p \cdot q$;
        $M(k, (i-1+l) \cdot c + j + l - c) = q \cdot q$;
      end
    else
      $M(k, (i-1+l-c) * c + j) = p \cdot q$;
      if $j + l \leq c$ **then**
        $M(k, k+l) = p \cdot q$;
        $M(k, (i-1+l-c) \cdot c + j + l) = q \cdot q$;
      else
        $M(k, k+l-c) = p \cdot q$;
        $M(k, (i-1+l-c) \cdot c + j + l - c) = q \cdot q$;
      end
    end
  end
end
$\hat{F} = M \backslash S^*(:) ./ n$;
Return $\hat{F}$.

VI. Theoretical Analysis

A. Privacy Analysis

Lemma 1: Our set-valued data perturbation protocol satisfies ϵ-LDP.

Proof. Let $\boldsymbol{s}_i^1$ be the perturbed set-valued data from the first set of canonical form outputted by Algorithm 1, whose elements consist of 0s and 1s (from the two points noted in

Subsection V-B, we know that the shifted data is not the same as the original one, and data elements cannot be all 0s or all 1s). The probability of observing $\boldsymbol{s}_i^1$, given its original input of protocol $\boldsymbol{v}_i^1$, is denoted as $\mathbb{P}(\boldsymbol{s}_i^1|\boldsymbol{v}_i^1)$. By observing $\boldsymbol{s}_i^1$, there are only two possible inputs: $\boldsymbol{s}_i^1$ and reverse shifted $\boldsymbol{s}_i^1$ (recall that in one case $\boldsymbol{s}_i^k = \text{circshift}(\boldsymbol{v}_i^k, c-l)$, so we have $\boldsymbol{v}_i^k = \text{circshift}(\boldsymbol{s}_i^k, l)$). According to the probability constraint of Eq. 3, the ratio of the two such conditional probabilities with distinct input data $\boldsymbol{v}_i^1\_1$ and $\boldsymbol{v}_i^1\_2$, is bounded by $\mathrm{e}^{\epsilon/2}$. Formally, we have:

$$\begin{aligned}\frac{\mathbb{P}(\boldsymbol{s}_i^1|\boldsymbol{v}_i^1\_1)}{\mathbb{P}(\boldsymbol{s}_i^1|\boldsymbol{v}_i^1\_2)} &\leq \frac{\mathbb{P}(\boldsymbol{s}_i^1|\boldsymbol{s}_i^1)}{\mathbb{P}\left(\boldsymbol{s}_i^1|\text{cirshift}(\boldsymbol{s}_i^1, l)\right)} \\ &= \left(\frac{\mathrm{e}^{\epsilon/2}}{\mathrm{e}^{\epsilon/2}+1}\right) \Big/ \left(\frac{1}{\mathrm{e}^{\epsilon/2}+1}\right) \\ &= \mathrm{e}^{\epsilon/2}.\end{aligned}$$

Thus, the perturbation on user i's set-valued data from the first set $\boldsymbol{v}_i^1$ satisfies $\epsilon/2$-LDP. Similarly, we can prove that the perturbation on $\boldsymbol{v}_i^2$ also satisfies $\epsilon/2$-LDP. According to the sequential composition theorem (Theorem 1), we conclude that our set-valued data perturbation protocol satisfies ϵ-LDP since $\epsilon = \epsilon/2 + \epsilon/2$, thereby achieving the design goal of *Privacy Preservation*. □

B. Utility Analysis

Theorem 2: The expectation of the estimated set-valued data joint frequency $\mathbb{E}(\hat{f}_{i,j})$ equals the actual frequency $f_{i,j}$, i.e., our estimation algorithm is unbiased.

Proof. According to Eq. 4, we have:

$$\begin{aligned}\mathbb{E}(f_{i,j}^*) = \mathbb{E}(\hat{f}_{i,j}) \cdot p^2 + \Big(\mathbb{E}(\hat{f}_{i,j+l}) + \mathbb{E}(\hat{f}_{i+l,j})\Big) \cdot p \cdot q \\ + \mathbb{E}(\hat{f}_{i+l,j+l}) \cdot q^2.\end{aligned} \tag{5}$$

Let $\mathcal{M}$ be the perturbation mechanism of Algorithm 1. We can obtain the following equation from the randomization process:

$$\mathbb{P}\left(\mathcal{M}(v_{i,j}^1, v_{i,j}^2) = s_{i,j}^1, s_{i,j}^2\right) = \begin{cases} p^2, & \text{if } v_{i,j}^1 = s_{i,j}^1, v_{i,j}^2 = s_{i,j}^2, \\ p \cdot q, & \text{if } v_{i,j+c-l}^1 = s_{i,j}^1, v_{i,j}^2 = s_{i,j}^2, \\ p \cdot q, & \text{if } v_{i,j}^1 = s_{i,j}^1, v_{i,j+c-l}^2 = s_{i,j}^2, \\ q^2, & \text{if } v_{i,j+c-l}^1 = s_{i,j}^1, v_{i,j+c-l}^2 = s_{i,j}^2. \end{cases}$$

Moreover, as mentioned in the previous subsection, by observing $\boldsymbol{s}_i^1$, there are only two possible inputs: $\boldsymbol{s}_i^1$ and reverse shifted $\boldsymbol{s}_i^1$ (in one case $\boldsymbol{s}_i^k = \text{circshift}(\boldsymbol{v}_i^k, c-l)$, so we have $\boldsymbol{v}_i^k = \text{circshift}(\boldsymbol{s}_i^k, l)$). The expectation of the perturbed joint frequency $\mathbb{E}(f_{i,j}^*)$ can be calculated by:

$$\mathbb{E}(f_{i,j}^*) = f_{i,j} \cdot p^2 + (f_{i,j+l} + f_{i+l,j}) \cdot p \cdot q + f_{i+l,j+l} \cdot q^2. \tag{6}$$

Combining Eq. 5 and Eq. 6, we can get a set of recursive linear simultaneous equations:

$$\begin{aligned}0 = \Big(\mathbb{E}(\hat{f}_{i,j}) - f_{i,j}\Big) \cdot p^2 + \Big(\mathbb{E}(\hat{f}_{i,j+l}) - f_{i,j+l}\Big) \cdot p \cdot q + \\ \Big(\mathbb{E}(\hat{f}_{i+l,j}) - f_{i+l,j}\Big) \cdot p \cdot q + \Big(\mathbb{E}(\hat{f}_{i+l,j+l}) - f_{i+l,j+l}\Big) \cdot q^2, \\ \text{for } i, j \in \{1, 2, ..., c\}.\end{aligned} \tag{7}$$

In Eq. 7, there are c^2 different equations and the same number of unknown expectations. According to the solvability condition of linear equations, Eq. 7 has an exclusive resolution:

$$\mathbb{E}(\hat{f}_{i,j}) = f_{i,j}, \text{ for } i, j \in \{1, 2, ..., c\}.$$

As can be seen, the expectation of the estimated joint frequency $\mathbb{E}(\hat{f}_{i,j})$ is equal to the actual frequency $f_{i,j}$, indicating that our scheme is unbiased. □

Theorem 3: Our joint distribution estimation mechanism has an error bound given by the following equation, which is independent of the set cardinality and the cyclic shift parameter.

$$\mathbb{E}(|\hat{f}_{i,j} - f_{i,j}|^2) \leq \frac{1}{n}\left(\frac{(\mathrm{e}^{\epsilon/2}+1)^4}{(\mathrm{e}^{\epsilon}+1)^2} - 1\right). \tag{8}$$

Proof. As it is proven in Theorem 2 that $\mathbb{E}(\hat{f}_{i,j}) = f_{i,j}$, we have $\mathbb{E}(|\hat{f}_{i,j} - f_{i,j}|^2) = \text{Var}(\hat{f}_{i,j})$. For convenience, we use variances to represent the mean square error. According to Eq. 4, we derive:

$$\begin{aligned}\text{Var}(f_{i,j}^*) = \text{Var}(\hat{f}_{i,j}) \cdot p^4 + \text{Var}(\hat{f}_{i+l,j+l}) \cdot q^4 \\ + \Big(\text{Var}(\hat{f}_{i,j+l}) + \text{Var}(\hat{f}_{i+l,j})\Big) \cdot p^2 \cdot q^2.\end{aligned}$$

Since Eq. 4 is recursive and symmetric, variances of the estimated frequency are all the same. The above equation can be written as:

$$\text{Var}(f_{i,j}^*) = \text{Var}(\hat{f}_{i,j})(p^4 + q^4 + 2p^2 \cdot q^2) = \text{Var}(\hat{f}_{i,j})(p^2 + q^2)^2. \tag{9}$$

Because $f_{i,j}^* = S^*(j,i)/n$, we know $\text{Var}(f_{i,j}^*) = \text{Var}(S^*(j,i))/n^2$. Besides, every element of S^* is the scaled summation of n independent random variables (w.p. $p^2, q^2, p \cdot q$) drawn from the Bernoulli distribution. Thus, we deduce:

$$\begin{aligned}\text{Var}\left(S^*(j,i)\right) =& n \cdot p^2 \cdot (1-p^2) \cdot f_{i,j} \\ &+ n \cdot q^2 \cdot (1-q^2) \cdot f_{i+l,j+l} \\ &+ n \cdot p \cdot q \cdot (1-p\cdot q) \cdot (f_{i,j+l} + f_{i+l,j}) \\ \leq& n \cdot (p^2 + q^2 + 2p \cdot q - p^4 - q^4 - 2p^2 \cdot q^2) \\ =& n \cdot \left(1 - (p^2+q^2)^2\right).\end{aligned}$$

Substituting the above inequality with Eq. 9, we get:

$$\text{Var}(\hat{f}_{i,j}) \leq \frac{1}{n}\left(\frac{1}{(p^2+q^2)^2} - 1\right) = \frac{1}{n}\left(\frac{(\mathrm{e}^{\epsilon/2}+1)^4}{(\mathrm{e}^{\epsilon}+1)^2} - 1\right).$$

Obviously, this error bound of the estimated set-valued joint frequency is not influenced by the set cardinality c and the cyclic shift parameter l, and thus we achieve the design goal of *Scalability*. □

C. Error Bounds Comparison

In this subsection, we compare the variance, or error bounds, of existing LDP-based methods with our proposed approach. As mentioned in Section I, to the best of our knowledge, there has been no prior work on privacy-preserving joint distribution analysis of set-valued data. Therefore, for a fair comparison, we adapt the existing frequency estimation and set-valued data analysis methods to align their functionality with our approach.

Table III presents the variances of these methods for their original functions and the adapted variances for joint distribution analysis of set-valued data. For simplicity, the variances in the table are represented using Big O notation. The specific variances of the compared methods can be found in their respective original papers, while the detailed variance and the proof of our approach are provided in Theorem 3. In the following paragraphs, we elaborate on the adaptation process of the compared methods and how we analyze the variance of these adapted methods.

Table III: Variance Comparison

Approaches	Original	(Adapted) Joint Distribution
BRR [22]	$O(\frac{1}{n\epsilon})$	$O(\frac{2c}{n\epsilon})$
k-RR [21]	$O(\frac{c}{n\epsilon})$	$O(\frac{mc^2}{n\epsilon})$[a]
UE [23], [24]	$O(\frac{1}{n\epsilon})$	$O(\frac{m}{n\epsilon})$[a]
Wheel [18]	$O(\frac{\sqrt{m}c}{n\epsilon^2})$[a]	$O(\frac{mc^2}{n\epsilon^2})$[a]
Our Work	$O(\frac{1}{n})$[b]	$O(\frac{1}{n})$[b]

[a] Assume that each user selects m pairs of items from two sets.

[b] Simplified. Refer to Theorem 3 for details.

BRR Adaptation: The Binary Randomized Response (BRR) is a basic method for binary categorical data frequency estimation. We adapt it for set-valued data joint distribution by encoding each item in the user's set-valued data as '1' or '0' and perturbing each bit according to the probabilities outlined in Eq. 1. As per the sequential composition theorem (Theorem 1), the privacy budget must be equally divided among each item in the combined domain (totaling $2c$ items). Consequently, the adapted variance is naturally amplified by a factor of $2c$ relative to the coefficient of ϵ.

UE and k-RR Adaptation: The Unary Encoding (UE) and k-Randomized Response (k-RR) methods are typically employed for frequency estimation in multi-category data, where users can select only one item from a set. We adapt these methods by considering the Cartesian product of the two original sets as a new set. Consequently, the joint distribution problem of the original two sets becomes a frequency estimation problem within the new set. Given that the new set contains c^2 elements, the original variance, which depends on the set's cardinality, requires the factor of c to be replaced with c^2. Furthermore, if users select m pairs of elements from the two sets, the privacy budget ϵ is divided into m parts, thereby dividing the coefficient of ϵ in the adapted variance by m.

Wheel Adaptation: PrivSet and Wheel are methods used for frequency estimation of set-valued data within a single set. Since PrivSet does not provide variance information in its paper and Wheel is its advanced version, we only compare our method with Wheel. Similar to the adaptations for UE and k-RR, we consider every possible pair of items from the original two sets as a new set, replacing the factor of c with c^2 in the original variance. However, unlike UE and k-RR, the coefficient of the privacy budget ϵ remains unchanged. This is because, in the Wheel adaptation, although the number of items changes, the object under consideration remains set-valued data. The adaptation serves as a pre-processing step and does not involve splitting the privacy budget in the LDP algorithm, still utilizing the entire ϵ.

In summary, comparing the error bounds of these related LDP-based methods confirms the achievement of our design goal of *Accuracy*.

VII. Experiments

In this section, we evaluate the actual performance of our proposed scheme through experimental results on both synthetic and real-world datasets. To the best of our knowledge, there is currently no work on privacy-preserving joint distribution analysis for set-valued data. To fairly compare with state-of-the-art LDP approaches for set-valued data that cannot estimate the joint distribution, we modified relevant works. Following the method described in Section VI-C, PrivSet [17] and the Wheel mechanism [18], [30] can be extended to estimate the joint distribution for set-valued data.

We describe the general settings of our experiments in VII-A. In subsection VII-B, we analyze the error of our scheme. Subsection VII-C examines whether the cyclic shift parameter l impacts accuracy. The impacts of the set cardinality and the number of users are evaluated in VII-D and VII-E, respectively.

A. General Settings

Parameter settings. For PrivSet [17], the output subset size k is set to the optimal size k^*. The set-valued data pre-processing method is similar to that described in the penultimate paragraph of Subsection VI-C. Since the number of items is expanded to c^2, without loss of generality, the padded size d is set to $c^2/5$. Specific parameter interpretations can be referred to in [17]. Applying the (c^2, d, ϵ, k^*)-PrivSet mechanism to pre-processed set-valued data ensures that the perturbation on every user's set-valued data satisfies ϵ-LDP. For the Wheel mechanism [18], [30], each possible pair (a total of c^2 pairs) of items from the two sets is mapped to an item in the original scheme, and every user consumes ϵ to perturb his/her data.

Datasets. We conduct experiments on both synthetic and real-world datasets. The synthetic datasets are generated by sampling set-valued data from the binomial distribution with probabilities of 0.3 and 0.6. We use two real-world datasets for performance evaluation, including online retail [46] and takeaway food orders [47], described as follows:

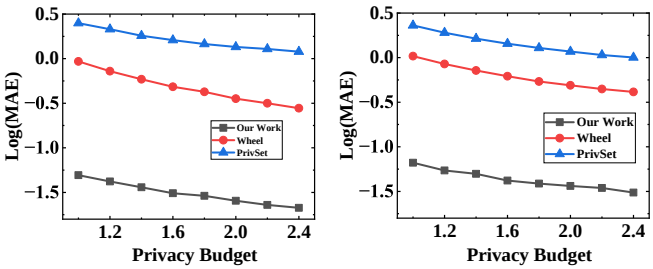
- **Online Retail**. This UCI dataset contains all the transactions for a UK-based, non-store online retailer between 2009 and 2011. We extract information on products, their categories, and customer IDs from the original dataset.
- **Takeaway Food Orders**. This dataset comprises takeaway food orders from two Indian restaurants in London, including over 10,000 food orders over three

years. We extract information about the ordered food, their types, and order IDs.

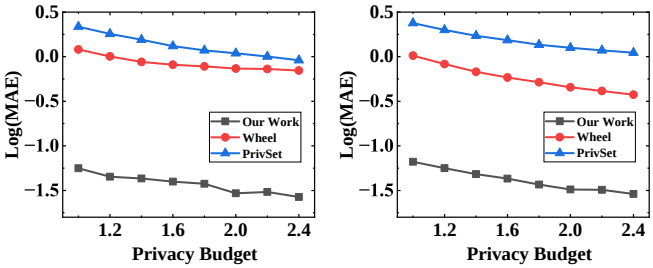
Measurement. To measure the accuracy of the estimated joint frequency of set-valued data, we follow previous works [18], [30] and use MAE (Mean Absolution Error) as a widely-used utility metric.

$$\text{MAE} = \frac{1}{c^2} \sum_{i \in \mathcal{U}^1, j \in \mathcal{U}^2} |f_{i,j} - \hat{f}_{i,j}|.$$

B. Error Analysis



(a) Synthetic, $B(n, 0.3)$ (b) Synthetic, $B(n, 0.6)$



(c) Online Retail (d) Takeaway Food Orders

Fig. 2: MAE (Mean Absolution Error) with $n = 200$, $c = 20$ and the privacy budget ranges from 0.2 to 2.4.

Effect of privacy budget. Fig. 2 illustrates the overall performance of our joint distribution analysis mechanism for set-valued data. It plots different errors with respect to various privacy budgets ϵ. Recall that, from the definition of LDP, ϵ determines the privacy protection level: a larger ϵ results in weaker privacy protection but higher utility. All the schemes in the figure follow this phenomenon: error decreases as ϵ increases. This observation is consistent with the trade-off between utility and privacy preservation.

Error comparison. From these figures, we observe that (1) our mechanism consistently achieves the lowest error (MAE) across all datasets. PrivSet and Wheel have similar performance, with Wheel performing slightly better than PrivSet as a new solution; (2) our mechanism is robust to large noise. Our method with small ϵ (< 1.6) even has less error than the other two methods with large ϵ (> 2.0).

C. Impact of Cyclic Shift Parameter

The parameter l is an intermediate parameter in our algorithm used to control the shift bits during perturbation. l is independent of the number of items in each user's sets, and the items belong to full sets of size c. As mentioned in Section V-B, to ensure privacy preservation, the shift parameter l must meet two specific requirements. Apart from privacy, we also conclude in Theorem 3 that the error is not related to the parameter l by proving the error bound. In this subsection, we demonstrate through experiments that parameter l does not affect the protocol's performance while meeting privacy-preservation requirements.

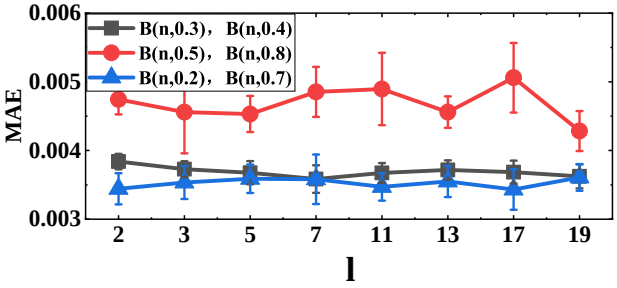
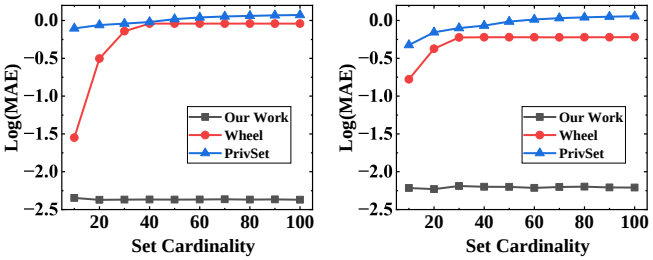


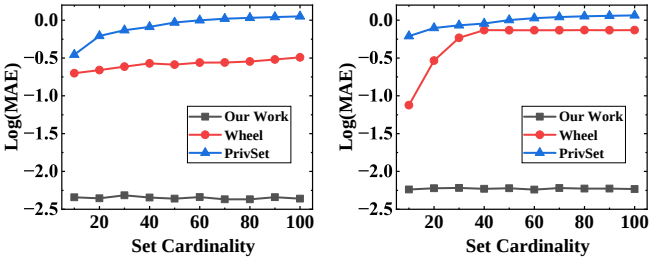
Fig. 3: MAE with $n = 5000$, $c = 23$, $\epsilon = 3$ and the shift parameter l ranges from 2 to 19.

The curves in Fig. 3 do not show a significant trend with the variation of l. It is worth mentioning that, compared to other figures, the vertical axis of Fig. 3 does not use a logarithmic scale, so even if the curves appear to fluctuate, the error fluctuation is actually minimal. To summarize, consistent with theoretical verification, the shift parameter l does not affect the overall performance of the protocol.

D. Impact of Set Cardinality



(a) Synthetic, $B(n, 0.3)$ (b) Synthetic, $B(n, 0.6)$



(c) Online Retail (d) Takeaway Food Orders

Fig. 4: MAE with $n = 3000$, $\epsilon = 3$ and the item set cardinality ranges from 10 to 100.

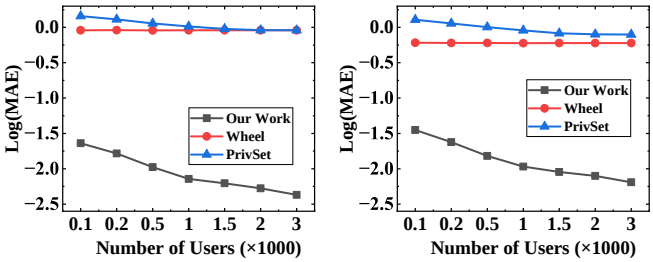
Effect of set cardinality. Fig. 4 quantifies the relationship between the set cardinality and the error. Regarding the challenging issue of low accuracy mentioned in Section I, the

results shown in this figure meet expectations: the accuracy of the adapted methods decreases as the set cardinality increases. Although the error growth of adapted PrivSet is not evident in the figure due to the logarithmic axis, the error does increase.

For any set cardinality, the error of our mechanism is significantly lower than that of the other two methods, verifying the design goal of *Accuracy* from another perspective. Under the MAE metric, adapted Wheel performs better than PrivSet, and the error growth of Wheel slows down as the set cardinality increases.

Independence between error and cardinality. These figures further demonstrate our design goal of *Scalability*. The curve of our estimation method is almost horizontal under MAE, proving that our scheme has good scalability and verifying the theoretical analysis in Section VI that the error of joint probability is independent of the set cardinality.

E. Impact of the Number of Users



(a) Synthetic, $B(n, 0.3)$ (b) Synthetic, $B(n, 0.6)$

(c) Online Retail (d) Takeaway Food Orders

Fig. 5: MAE with $c = 30$, $\epsilon = 3$ and the number of users ranges from 100 to 3000.

To examine accuracy from another perspective, we also analyze the impact of the number of participants on the error. Fig. 5 shows the error on different datasets as the number of users increases. Generally, LDP-based aggregation schemes can reduce average error by incorporating more data, consistent with the curves in the figure. The downward trend is evident in our work and PrivSet, while it is less evident in Wheel due to the logarithmic axis.

Effect of the number of users. Fig. 5 shows that our method achieves optimal accuracy as the number of users increases, and the performance on real-world datasets validates this in practice. Adapted Wheel is more accurate than PrivSet, but as the number of users (n) increases, the gap between the two decreases. When n reaches a certain level, the MAE of adapted PrivSet is almost equal to Wheel. The analysis of n also reveals that our scheme is very suitable for scenarios involving a large number of participants, such as big data analysis.

VIII. Conclusion

In this paper, we proposed a joint distribution analysis method for set-valued data with local differential privacy. The encoding of set-valued data effectively solves the problem of inconsistent item numbers among users. We designed a scalable perturbation mechanism under ϵ-LDP by limiting the range of users' responses in the collection process and cyclically shifting the encoded set-valued data, so that the size of the universal set does not influence result accuracy. To maintain the relationship between different sets and derive the joint distribution, we proposed an aggregation method and a calibration mechanism via matrix operations. Theoretical analysis and extensive experimental comparisons with state-of-the-art approaches on both synthetic and real-world datasets demonstrated the practicability of our scheme.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China (NSFC) under Grant No. 62372425, Anhui Provincial Key Research and Development Plan under Grant No. 2022a05020050, and Youth Innovation Promotion Association of Chinese Academy of Sciences (CAS) under Grant No. Y202093.

References

[1] H. Biao, H. Banghe, and T. Junqi, "Set-valued preference relation and its properties," in *2022 IEEE 8th International Conference on Cloud Computing and Intelligent Systems (CCIS)*, 2022, pp. 409–413.

[2] "General data protection regulation (GDPR)," https://gdpr-info.eu/, accessed: Mar., 2024.

[3] "Uniform personal data protection act (UPDPA)," https://www.uniformlaws.org/committees/community-home?CommunityKey=28443329-e343-4cbc-8c72-60b12fd18477, accessed: Mar., 2024.

[4] N. Bugshan, I. Khalil, N. Moustafa, and M. S. Rahman, "Privacy-preserving microservices in industrial internet-of-things-driven smart applications," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 2821–2831, 2023.

[5] M. Yang, I. Tjuawinata, K. Y. Lam, J. Zhao, and L. Sun, "Secure hot path crowdsourcing with local differential privacy under fog computing architecture," *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 2188–2201, 2022.

[6] B. Jiang, M. Li, and R. Tandon, "Local information privacy and its application to privacy-preserving data aggregation," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 3, pp. 1918–1935, 2022.

[7] B. Jiang, J. Li, H. Wang, and H. Song, "Privacy-preserving federated learning for industrial edge computing via hybrid differential privacy and adaptive compression," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1136–1144, 2023.

[8] X. Xu, Z. Fan, M. Trovati, and F. Palmieri, "Mlpkv: A local differential multi-layer private key-value data collection scheme for edge computing environments," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 1825–1838, 2023.

[9] H. Jiang, J. Pei, D. Yu, J. Yu, B. Gong, and X. Cheng, "Applications of differential privacy in social network analysis: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 108–127, 2023.

[10] M. Zhang, J. Zhou, G. Zhang, L. Cui, T. Gao, and S. Yu, "Apdp: Attribute-based personalized differential privacy data publishing scheme for social networks," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 2, pp. 922–933, 2023.

[11] X. Zheng, M. Guan, X. Jia, L. Guo, and Y. Luo, "A matrix factorization recommendation system-based local differential privacy for protecting users' sensitive data," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 3, pp. 1189–1198, 2023.

[12] X. Li, H. Yan, Z. Cheng, W. Sun, and H. Li, "Protecting regression models with personalized local differential privacy," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 960–974, 2023.

[13] Q. Ye, H. Hu, X. Meng, H. Zheng, K. Huang, C. Fang, and J. Shi, "Privkvm*: Revisiting key-value statistics estimation with local differential privacy," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 17–35, 2023.

[14] H. Zhang, Y. Xia, Y. Ren, J. Guan, and S. Zhou, "Differentially private nonlinear causal discovery from numerical data," in *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI)*. AAAI Press, 2023, pp. 12 321–12 328.

[15] Z. Xu, M. Collins, Y. Wang, L. Panait, S. Oh, S. Augenstein, T. Liu, F. Schroff, and H. B. McMahan, "Learning to generate image embeddings with user-level differential privacy," in *Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2023, pp. 7969–7980.

[16] W. Alghamdi, J. F. Gómez, S. Asoodeh, F. P. Calmon, O. Kosut, and L. Sankar, "The saddle-point method in differential privacy," in *Proceedings of the 39th International Conference on Machine Learning (ICML)*, vol. 202. PMLR, 2023, pp. 508–528.

[17] S. Wang, L. Huang, Y. Nie, P. Wang, H. Xu, and W. Yang, "PrivSet: Set-valued data analyses with locale differential privacy," in *Proceedings of the 2018 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2018, pp. 1088–1096.

[18] S. Wang, Y. Qian, J. Du, W. Yang, L. Huang, and H. Xu, "Set-valued data publication with local privacy: Tight error bounds and efficient mechanisms," *Proceedings of the VLDB Endowment*, vol. 13, no. 8, pp. 1234–1247, 2020.

[19] Z. Qin, Y. Yang, T. Yu, I. Khalil, X. Xiao, and K. Ren, "Heavy hitter estimation over set-valued data with local differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 192–203.

[20] Y. Zhu, Y. Cao, Q. Xue, Q. Wu, and Y. Zhang, "Heavy hitter identification over large-domain set-valued data with local differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 414–426, 2024.

[21] P. Kairouz, S. Oh, and P. Viswanath, "Extremal mechanisms for local differential privacy," in *Advances in Neural Information Processing Systems*, vol. 27. MIT Press, 2014.

[22] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[23] U. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 1054–1067.

[24] T. Wang, J. Blocki, N. Li, and S. Jha, "Locally differentially private protocols for frequency estimation," in *Proceedings of the 26th USENIX Security Symposium*. USENIX Association, 2017, pp. 729–745.

[25] Q. Xue, Y. Zhu, and J. Wang, "Joint distribution estimation and naïve bayes classification under local differential privacy," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 2053–2063, 2021.

[26] M. Xu, B. Ding, T. Wang, and J. Zhou, "Collecting and analyzing data jointly from multiple services under local differential privacy," *Proceedings of the VLDB Endowment*, vol. 13, no. 12, pp. 2760—-2772, 2020.

[27] R. Bassily and A. Smith, "Local, private, efficient protocols for succinct histograms," in *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*. ACM, 2015, pp. 127–135.

[28] T. Wang, N. Li, and S. Jha, "Locally differentially private frequent itemset mining," in *2018 IEEE Symposium on Security and Privacy (SP)*, 2018, pp. 127–143.

[29] L. Wang, Q. Ye, H. Hu, and X. Meng, "EPS^2: Privacy preserving set-valued data analysis in the shuffle model," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–14, 2023.

[30] S. Wang, Y. Li, Y. Zhong, K. Chen, X. Wang, Z. Zhou, F. Peng, Y. Qian, J. Du, and W. Yang, "Locally private set-valued data analyses: Distribution and heavy hitters estimation," *IEEE Transactions on Mobile Computing*, pp. 1–14, 2023.

[31] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography*. Springer Berlin Heidelberg, 2006, pp. 265–284.

[32] J. Wang, X. Zhang, Q. Zhang, M. Li, Y. Guo, Z. Feng, and M. Pan, "Data-driven spectrum trading with secondary users' differential privacy preservation," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 1, pp. 438–447, 2021.

[33] Z. Lu and H. Shen, "Differentially private kk-means clustering with convergence guarantee," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 4, pp. 1541–1552, 2021.

[34] J. Yang, L. Xiang, R. Chen, W. Li, and B. Li, "Differential privacy for tensor-valued queries," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 152–164, 2022.

[35] F. Farokhi, N. Wu, D. Smith, and M. A. Kaafar, "The cost of privacy in asynchronous differentially-private machine learning," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2118–2129, 2021.

[36] W. Lin, B. Li, and C. Wang, "Towards private learning on decentralized graphs with local differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2936–2946, 2022.

[37] L. Wang, D. Yang, X. Han, D. Zhang, and X. Ma, "Mobile crowdsourcing task allocation with differential-and-distortion geo-obfuscation," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 967–981, 2021.

[38] M. E. Gursoy, A. Tamersoy, S. Truex, W. Wei, and L. Liu, "Secure and utility-aware data collection with condensed local differential privacy," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2365–2378, 2021.

[39] M. E. Gursoy, L. Liu, K.-H. Chow, S. Truex, and W. Wei, "An adversarial approach to protocol analysis and selection in local differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1785–1799, 2022.

[40] X. Ren, C.-M. Yu, W. Yu, S. Yang, X. Yang, J. A. McCann, and P. S. Yu, "LoPub : High-dimensional crowdsourced data publication with local differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 9, pp. 2151–2166, 2018.

[41] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3—4, pp. 211–407, 2014.

[42] Q. Ye, H. Hu, X. Meng, and H. Zheng, "PrivKV: Key-value data collection with local differential privacy," in *Proceedings of the 2019 IEEE Symposium on Security and Privacy*. IEEE, 2019, pp. 317–331.

[43] J. Imola, T. Murakami, and K. Chaudhuri, "Locally differentially private analysis of graph statistics," in *Proceedings of the 30th USENIX Security Symposium*. USENIX Association, 2021, pp. 983–1000.

[44] ——, "Communication-Efficient triangle counting under local differential privacy," in *Proceedings of the 31st USENIX Security Symposium*. USENIX Association, 2022, pp. 537–554.

[45] "Function: circshift," https://www.mathworks.com/help/matlab/ref/circshift.html, accessed: May., 2024.

[46] "Online retail II data set," https://archive.ics.uci.edu/ml/datasets/Online+Retail+II, accessed: Mar., 2024.

[47] "Takeaway food orders," https://www.kaggle.com/datasets/henslersoftware/19560-indian-takeaway-orders, accessed: Mar., 2024.

Yaxuan Huang received her becholar's degree from the Department of Information Security, University of Science and Technology of China (USTC) in 2021. She is currently a graduate student in the School of Cyber Science and Technology, USTC. Her research interests include network security and privacy computing.

Kaiping Xue (M'09-SM'15) received his bachelor's degree from the Department of Information Security, University of Science and Technology of China (USTC), in 2003 and received his doctor's degree from the Department of Electronic Engineering and Information Science (EEIS), USTC, in 2007. From May 2012 to May 2013, he was a postdoctoral researcher with the Department of Electrical and Computer Engineering, University of Florida. Currently, he is a Professor in the School of Cyber Science and Technology, USTC. He is also the director of Network and Information Center, USTC. His research interests include next-generation Internet architecture design, transmission optimization and network security. His work won best paper awards in IEEE MSN 2017 and IEEE HotICN 2019, the Best Paper Honorable Mention in ACM CCS 2022, the Best Paper Runner-Up Award in IEEE MASS 2018, and the best track paper in MSN 2020. He serves on the Editorial Board of several journals, including the IEEE Transactions on Information Forensics and Security (TIFS), the IEEE Transactions on Dependable and Secure Computing (TDSC), the IEEE Transactions on Wireless Communications (TWC), and the IEEE Transactions on Network and Service Management (TNSM). He has also served as a (Lead) Guest Editor for many reputed journals/magazines, including IEEE Journal on Selected Areas in Communications (JSAC), IEEE Communications Magazine, and IEEE Network. He is an IET Fellow and an IEEE Senior Member.

Bin Zhu received his bachelor's degree in Information Security from the School of Cyber Science and Technology, University of Science and Technology of China (USTC) in 2019. He is currently working toward the Ph.D degree from the School of Cyber Science and Technology, USTC. His research interests include Network security and applied cryptography.

David S.L. Wei (SM'07) received his Ph.D. degree in Computer and Information Science from the University of Pennsylvania in 1991. From May 1993 to August 1997 he was on the Faculty of Computer Science and Engineering at the University of Aizu, Japan (as an Associate Professor and then a Professor). He has authored and co-authored more than 140 technical papers in various archival journals and conference proceedings. He is currently a Professor with the Computer and Information Science Department at Fordham University. He was a lead guest editor or a guest editor for several special issues in the IEEE Journal on Selected Areas in Communications, the IEEE Transactions on Cloud Computing, and the IEEE Transactions on Big Data. He also served as an Associate Editor of IEEE Transactions on Cloud Computing, 2014-2018, an editor of IEEE J-SAC for the Series on Network Softwarization & Enablers, 2018 – 2020, and an Associate Editor of Journal of Circuits, Systems and Computers, 2013-2018. Dr. Wei is the recipient of IEEE Region 1 Technological Innovation Award (Academic), 2020, for contributions to information security in wireless and satellite communications and cyber-physical systems. He is a member of ACM and AAAS, and is a life senior member of IEEE, IEEE Computer Society, and IEEE Communications Society. Currently, Dr. Wei focuses his research efforts on cloud and edge computing, cybersecurity, and quantum computing and communications.

Qibin Sun (F'11) received the Ph.D. degree from the Department of Electronic Engineering and Information Science (EEIS), University of Science and Technology of China (USTC), in 1997. He is currently a professor in the School of Cyber Science and Technology, USTC. His research interests include multimedia security, network intelligence and security, and so on. He has published more than 120 papers in international journals and conferences. He is a fellow of IEEE.

Jun Lu received his bachelor's degree from southeast university in 1985 and his master's degree from the Department of Electronic Engineering and Information Science (EEIS), University of Science and Technology of China (USTC), in 1988. Currently, he is a professor in the School of Cyber Science and Technology and the Department of EEIS, USTC. His research interests include theoretical research and system development in the field of integrated electronic information systems, network and information security. He is an Academician of the Chinese Academy of Engineering (CAE).