VertiMRF: Differentially Private Vertical Federated Data Synthesis

Fangyuan Zhao
Xi'an Jiaotong University
zfy1454236335@stu.xjtu.edu.cn

Zitao Li
Alibaba Group
zitao.l@alibaba-inc.com

Xuebin Ren
Xi'an Jiaotong University
xuebinren@mail.xjtu.edu.cn

Bolin Ding
Alibaba Group
bolin.ding@alibaba-inc.com

Shusen Yang
Xi'an Jiaotong University
shusenyang@mail.xjtu.edu.cn

Yaliang Li
Alibaba Group
yaliang.li@alibaba-inc.com

ABSTRACT

Data synthesis is a promising solution to share data for various downstream analytic tasks without exposing raw data. However, without a theoretical privacy guarantee, a synthetic dataset would still leak some sensitive information. Differential privacy is thus widely adopted to safeguard data synthesis by strictly limiting the released information. This technique is advantageous yet presents significant challenges in the vertical federated setting, where data attributes are distributed among different data parties. The main challenge lies in maintaining privacy while efficiently and precisely reconstructing the correlation among cross-party attributes. In this paper, we propose a novel algorithm called VertiMRF, designed explicitly for generating synthetic data in the vertical setting and providing differential privacy protection for all information shared from data parties. We introduce techniques based on the Flajolet-Martin sketch (or frequency oracle) for encoding local data satisfying differential privacy and estimating cross-party marginals. We provide theoretical privacy and utility proof for encoding in this multi-attribute data. Collecting the locally generated private Markov Random Field (MRF) and the sketches, a central server can reconstruct a global MRF, maintaining the most useful information. Additionally, we introduce two techniques tailored for datasets with large attribute domain sizes, namely dimension reduction and consistency enforcement. These two techniques allow flexible and inconsistent binning strategies of local private MRF and the data sketching module, which can preserve information to the greatest extent. We conduct extensive experiments on four real-world datasets to evaluate the effectiveness of VertiMRF. End-to-end comparisons demonstrate the superiority of VertiMRF, and ablation studies validate the effectiveness of each component.

1 INTRODUCTION

With the increasing stringency of data privacy regulations such as the European General Data Protection Regulation (GDPR) [54] and the California Consumer Privacy Act [45], data privacy has become a significant concern for various data analysis tasks. Following this trend, data synthesis has emerged as a promising technique. For the tabular data domain, the synthesis algorithms aim to generate and release synthetic data that preserves the statistical characteristics of the original data, allowing for diverse data analysis tasks to be conducted without access to the original real data from individuals.

Coupled with *differential privacy* (DP) [13] techniques, the synthetic data can provide theoretical privacy guarantees for the membership information of individual records in the original datasets. Compared with other DP algorithms for specific analytic tasks, DP data synthesis can support an unlimited number of unrestricted downstream tasks without additional privacy loss other than the one occurring during data synthesis. However, it remains a challenge to ensure strict DP while generating synthetic data of high quality. A growing body of academic research has focused on improving the trade-off between privacy and utility of DP synthetic data, including GAN-based approaches [2, 5, 7, 14, 24, 64], game theory based approaches [15, 18, 53] and marginal based approaches [6, 38, 39, 63, 66]. Among these, marginal-based approaches have shown promising results and achieved the state-of-the-art performance. They generally follow the *"select-measure-generate"* paradigm. First, a set of low-dimensional marginals is selected, then these marginals are measured with the protection of DP, and a dataset is finally synthesized by aligning with the measured marginals. However, it is worth noting that these studies primarily focus on the centralized setting, assuming that the raw data has already been collected and processed by a trusted curator.

To realize the value of data at the furthest level, multiple data parties may want to cooperate on some tasks for more comprehensive and accurate information. If such cooperation is achieved without sharing data directly, the setting is generally called federated learning (FL) [25]. A relatively well-studied scenario in FL is that the data parties have data with the same set of attributes but from different groups of individuals. This scenario is called horizontal federated learning (HFL) because the local dataset can be obtained by splitting a virtual global dataset by individuals [40, 41, 57, 59]. Nevertheless, another attractive but challenging case is when multiple data parties have data from the same set of individuals but on different attributes [21, 33, 34, 36, 52, 60]. Symmetrically, this setting is called vertical federated learning (VFL) as the local datasets can be derived by dividing a global dataset by attributes. VFL techniques attract the attention of many medical or fintech companies because their model accuracy can be boosted by more comprehensive information brought by VFL. For example, Webank [56] combines users' basic profile data from internet companies with their behavioral data from car rental companies to enhance risk control measures. In this paper, we focus on data synthesis in the VFL setting as it has great potential in various aspects.

1) It facilitates the cross-party data analysis. Simply combining the synthetic data generated independently by the multiple parties loses the statistical property of cross-party attributes. However, when a VFL data synthesis algorithm that accurately captures the cross-party correlation is available, any downstream correlation analysis can be done efficiently once the synthetic data is ready. *2) It enables hyper-parameter selection.* VFL tasks often involve substantial costs for hyperparameter tuning among multi-parties, due to the strict limitations of cross-party data access. Releasing a synthetic dataset that preserves the statistical characteristics of the original data can help select optimal hyper-parameters before model training.

Despite the great potential, there are following *challenges* that hinder the practical applications.
C1: Information loss because of data modeling. The central DP synthesis algorithms construct dependency graphs and select the representative marginals to approximate joint distribution with methods of low computation cost and relatively high accuracy in assessing the inter-attribute statistics. However, each party only has access to his/her local attributes, and the server can only access the information shared by data parties. Consequently, it is unavoidable that some data structures will be adapted to encode and share the essential information of local datasets for building global dependency graphs. An inappropriate encoding strategy may introduce significant information loss, even without enforcing DP.
C2: Information loss because of DP. Since the attributes of a single record are spread among different parties in the vertical setting, the privacy budget is split among all parties to achieve sufficient protection. Furthermore, there are multiple cross-party attribute combinations to estimate, which would introduce more noise by the nature of DP algorithms. Therefore, the utility loss in the vertical setting would be more significant than in the centralized setting when the same level of DP is achieved.
C3: High communication cost. If a VFL algorithm relies on cryptography protocols (i.e., the inner-product protocol) to secure cross-party computation, it may incur huge communication overhead and can be infeasible when the data volume is large.

Although there are a few works on DP data synthesis under the vertical setting, they still have limitations related to the challenges above. DistDiffGen [43] is a two-party DP data synthesis framework. It falls short of handling C1 because it relies on a given taxonomy tree requiring strong prior knowledge and is tailored to classification tasks only. VertiGAN [23] adapts the DP-WGAN approach to vertical setting [23]. However, the GAN-based models are proven to be not suitable for synthesizing tabular data with DP, which indicates that C1 and C2 still hinder its practical application. DPLT [51] utilizes a latent tree model to capture the correlation among cross-party attributions. However, its application is limited by C1 and C3 because it is designed for datasets with binary attributes and suffers from the huge communication and computation costs incurred by the complicated cryptography protocol.

To handle the challenges, we propose VertiMRF for generating high-quality synthetic data with differential privacy guarantees in the VFL setting with multiple data parties and a semi-honest central server. Our key observation is that the central DP data synthesis can achieve great performance in terms of privacy-utility trade-off, and the cross-party statistic estimation is necessary but may unavoidably be less accurate. Thus, VertiMRF adapts, combines, and balances these two components. VertiMRF adapts PrivMRF [6] to capture and share differentially private intra-party attribute statistic information. We then design special protocols to let the data parties encode and the sever decode the cross-party attribute correlation information. With both intra-party and cross-party attribute correlation information, the server can reconstruct a global MRF for full-view data synthesis. Our key contributions assembled in VertiMRF are summarized as follows:

- As a building block of the algorithm, we apply the Flajolet-Martin (FM) sketch to estimate the cross-party multi-attribute marginals and utilize its inherent randomness to achieve provable differential privacy. In particular, we encode each local attribute using the same hash keys to generate FM sketches for its memberships. Relying on the support for the union operation and the inclusion-exclusion rule, the semi-honest central server can estimate the contingency histogram of each marginal simply with the sketches. We theoretically prove the privacy guarantee using the same hash key to all attributes and then give an error analysis. With the sketches, the server can estimate the cross-party correlations of any attribute combination without introducing additional privacy costs.
- We propose a communication efficient and differentially private vertical data synthesis framework VertiMRF. VertiMRF lets each data party privately encode the correlation of intra-party attributes with a Markov Random Field (MRF) and the membership information of each attribute values with FM sketch for later cross-party correlation estimation. With these two kinds of differentially private information, we design a sequence of algorithms to let the server build an MRF of the global view by connecting local MRF and refining parameters with decoded cross-party correlation information.
- We propose two important techniques to improve the performance of VertiMRF when dealing with datasets of large domain sizes. In particular, we first propose a heuristic dimension reduction technique to bin the attributes with large domain sizes while preserving the statistical information without violating DP. We also propose a consistency enforcement technique to reduce the overwhelming noises incurred by the inconsistent marginal estimations by local MRFs and FM sketches.
- We conduct empirical validation on four real-world datasets. The end-to-end comparison results demonstrate the superiority of our approach to the baseline algorithms. Furthermore, the impact and effectiveness of each component of our approach are validated by ablation studies.

2 PRELIMINARIES

2.1 Differential Privacy

Differential privacy (DP) is a rigorous privacy notion that quantifies the privacy loss of algorithms by analyzing the statistical difference between the algorithm outputs on neighboring datasets differing on only one record.

Definition 1 (Differential Privacy [13]). A randomized mechanism $\mathcal{M}$ satisfies (ϵ, δ)-differential privacy if for any neighboring datasets $D, D' \in \mathcal{D}$ that differ on only one record, their outputs fall in any $R \subset Range(\mathcal{M})$ with probability $Pr[\mathcal{M}(D) \subseteq R] \leq \exp(\epsilon) Pr[\mathcal{M}(D') \subseteq R] + \delta$.

DP is a popular privacy notion because the privacy loss is composable. Basically, with any two algorithms f and g which satisfy (ϵ_1, δ_1)-DP and (ϵ_2, δ_2)-DP respectively, the sequential use of $f \odot g$ on a dataset satisfies $(\epsilon_1 + \epsilon_2, \delta_1 + \delta_2)$-DP. However, such composition is not tight, which may incur huge algorithm utility loss with the overwhelming noises when a DP mechanism is applied repetitively. To mitigate this issue, R̀enyi Differential Privacy (RDP) is proposed to account for more accurate sequential privacy losses.

Definition 2 (Rényi Differential Privacy [42]). A randomized mechanism $\mathcal{M} : D \rightarrow R$ is said to be (λ, ϵ)-Rényi Differential Privacy, if for any two neighboring datasets D, D', it holds that

$$D_\lambda \left(\mathcal{M}(D) | \mathcal{M}(D')\right) \triangleq \frac{1}{\lambda} \log \mathbb{E}_R \left(\frac{\mathcal{M}(D) \subseteq R}{\mathcal{M}(D') \subseteq R} \right)^\lambda \leq \epsilon \qquad (1)$$

RDP can provide a tighter bound of DP when composing a large number of DP mechanisms.

Lemma 1 (DP Composition based on RDP). *Let f be the composition of n mechanisms that satisfies ϵ-DP, then for each $0 < \delta < 1$ with* $\log(1/\delta) \geq n\epsilon^2$, *$f$ satisfies $(4\epsilon\sqrt{2n \log 1/\delta}, \delta)$-DP.*

2.2 DP Flajolet-Martin Sketch

Flajolet-Martin (FM) Sketch is a probabilistic data structure for multi-set cardinality estimation with DP guarantee. It is constructed by hashing each element in a multi-set to an integer by a hash function $\mathcal{H}$ with a key ξ. The hashed integers are then independent geometric random variables with the parameter $\frac{\gamma}{1+\gamma}$ if ξ is sampled from a large set uniformly. Note that, the duplicated elements in the multi-set are mapped to the same integer. Thus, the cardinality k can be estimated as $k = (1 + \gamma)^\alpha$ where α denotes the maximum of the observed integer after hashing. It has been proved in [49] that $(1 + \gamma)^\alpha \in \left[\frac{k}{(1+\gamma)}, k(1 + \gamma)\right]$ with a reasonable probability. The estimation can be improved by repeating the procedure [49] multiple times with different hash functions and taking the $1/e$-th quantile of all the maxima as the final estimator.

The FM sketch-based cardinality estimation is widely used due to its appealing property that the sketch structure is mergeable. That is, given two different multi-set $\mathcal{X}_1$ and $\mathcal{X}_2$ and their corresponding FM sketches α_1 and α_2, then the cardinality of their union $\mathcal{X}_1 \cup \mathcal{X}_2$ can be simply estimated as $(1 + \gamma)^{\max(\alpha_1, \alpha_2)}$. Based on this and the inclusion-exclusion principle, i.e., $\mathcal{X}_1 \cap \mathcal{X}_2 = \overline{\overline{\mathcal{X}_2} \cup \overline{\mathcal{X}_2}}$, the cardinality of the intersection of two multi-sets can also be estimated.

Differentially private FM-sketch. As mentioned, estimating the cardinality of a multi-set using FM-sketch involves mapping distinct elements to geometric random variables and selecting the maximum value. However, this process may violate the privacy constraint as it requires the access to the raw data and the maximum value may reveal statistical information about the set. Fortunately, recent studies [11, 49] have demonstrated that FM-sketch can preserve DP under certain conditions.Specifically, if the multi-set contains at least $\frac{1}{e^\epsilon - 1}$ distinct elements and the maximum of the geometric random variables is lower bounded by $\lceil \log_{1+\gamma} \frac{1}{1-e^{-\epsilon}} \rceil$, where γ is parameter of the geometric distribution, then the process of selecting the maximum of these random variables ensures ϵ-DP. The privacy guarantee is formalized in lemma 2, and the DP FM-sketching algorithm is detailed in Algorithm 1.

Lemma 2. *Let $Y_1, \ldots, Y_{k+1}$ be independent random variables where each $Y_i \sim geometric(\frac{\gamma}{1+\gamma})$. Let $W_1 = \max\{Y_1, \ldots, Y_k, b\}$ and $W_2 = \max\{Y_1, \ldots, Y_{k+1}, b\}$. For any ϵ, if $k \geq \frac{1}{e^\epsilon - 1}$ and $b \geq \lceil \log_{1+\gamma} \frac{1}{1-e^{-\epsilon}} \rceil$, then it holds that $|\log \frac{Pr[W_1 = O]}{Pr[W_2 = O]}| \leq \epsilon, \forall O \in \mathbb{N}^+$.*

Algorithm 1 DPFM

```
Input: Multi-set $\mathcal{X} = \{x_1, ..., x_n\}$, domain $[u]$, distribution parameter $\gamma$, privacy budget $\epsilon'$, hash key $\xi \sim Uniform(R)$.
Output: DP FM-sketch $\alpha$ for $\mathcal{X}$.
  $k_p \leftarrow \lceil \frac{1}{e^{\epsilon'} - 1} \rceil$, $\alpha_{min} \leftarrow \lceil \log_{1+\gamma} \frac{1}{1-e^{-\epsilon'}} \rceil$
  $\alpha_p \leftarrow \max\{Y_1, ..., Y_{k_p}\}$ where $Y_i \sim Geometric(\frac{\gamma}{1+\gamma}), \forall i \leq k_p$
  $\alpha_{\mathcal{X}} \leftarrow \max\{\mathcal{H}_\xi(x_j)\}, \forall x_j \in \mathcal{X}$.
  return $\max\left\{\alpha_{\mathcal{X}}, \alpha_p, \alpha_{min}\right\}$
```

2.3 DP Data Synthesis

Let D be a set of data tuples $\{x^{(1)}, \ldots, x^{(n)}\}$. Each tuple consists of values of a set of attributes $\mathcal{A} = \{A^1, \ldots, A^d\}$. Each attribute $A^j, \forall j \in [d]$ has domain size u_j. Without loss of generality, we denote the domain of A^j as $[u_j] \triangleq \{1, \ldots, u_j\}$. With $M \subset \mathcal{A}$, $x_M^{(l)}$ denotes the values of tuple $x^{(l)}$ on an attribute set M. Let T_M be the counts of occurrences of all possible value tuples of attributes M in D. That is, T_M is a vector of length $\prod_{A^j \in M} u_j$ and each element is defined as

$$T_M[\mathbf{v}] = \sum_{l \in [n]} \mathbb{I}(x_M^{(l)} = \mathbf{v}), \quad \forall \mathbf{v} \in \prod_{A^j \in M} [u_j] \qquad (2)$$

T_M is referred as the *contingency histogram* of M.

Data synthesis focuses on generating a dataset $\hat{D}$ given D such that ideally $\forall M \subseteq \mathcal{A}, \hat{T}_M \approx T_M$. A key challenge of DP data synthesis is to circumvent the curse of dimensionality incurred by a large d. With the increase of d, the error of $T_{\mathcal{A}}$ grows exponentially, as DP noise has to be added to each count of the contingency histogram. To address this challenge, there have been works [6, 38, 39, 63, 66] that propose to utilize low-way marginal distributions to approximate the high-way distribution without losing much correlations among the attributes. Among these works, PrivMRF [6], utilizing *Markov Random Field* (MRF) to model the attribute correlations, shows the state-of-the-art performance. The basic idea of PrivMRF is to select an appropriate set of marginals to construct an MRF, which is then used to approximate the joint distribution of all attributes. In particular, PrivMRF consists of four phases:

- **Phases 1: Generate attribute graph.** PrivMRF starts by generating an attribute graph $\mathcal{G}$ through greedily linking up each attribute pair (A^i, A^j) in the descending order of noisy R-scores:

$$R(A^i, A^j) = \frac{n}{2} \left\| Pr[A^i, A^j] - Pr[A^i] \cdot Pr[A^j] \right\|_1 + \mathcal{N}(0, \sigma_R^2) \qquad (3)$$

After that, $\mathcal{G}$ is triangulated to ensure the domain size of the maximal clique not exceeding a threshold.

- **Phases 2: Choose candidate marginal set.** PrivMRF samples a set of candidate marginals $\mathcal{U}$ from the cliques of triangulated $\mathcal{G}$ and ensure each marginal $M \in \mathcal{U}$ is θ-useful. That is $\frac{n}{\prod_{A^i \in M} u_i} \leq \theta \cdot g$, where g denotes the expected absolute value of the noise to be injected into each count of T_M. θ-usefulness ensures that the average count in T_M is large enough to tolerate the noise.
- **Phases 3: Initialize the marginal set.** From $\mathcal{U}$, PrivMRF selects the most highly correlated marginal for each attribute to constitute an initialized marginal set $\mathcal{S}$, which is used to estimate the parameters Θ of the MRF. Θ is a real vector where each element corresponds to an entry in a contingency histogram $T_M, \forall M \in \mathcal{S}$.

The MRF models the distribution of arbitrary tuple x as:

$$Pr[x] \propto \prod_{M \in \mathcal{S}} \exp\left(\Theta_M[x_M]\right) \tag{4}$$

where Θ_M denotes the sub-vector of Θ corresponding to M, and $\Theta_M[x_M]$ is the element corresponding to x_M. Based on the estimated Θ, any marginal M' can be inferred by the MRF as

$$Pr[y] = \sum_{x, x_{M'} = y} Pr[x], \quad \forall y \in \prod_{A^j \in M'} u_j. \tag{5}$$

- **Phases 4: Refine the marginal set.** PrivMRF proceeds to refine the marginal set $\mathcal{S}$ by inserting marginals that cannot be accurately inferred by the MRF and iteratively refine the estimation of MRF.

3 DIFFERENTIALLY PRIVATE VERTICAL DATA SYNTHESIS

We provide the problem definition of DP data synthesis in the vertical setting and an overview of our solution in this section.

3.1 Problem Definition

We consider a system constituted by m data parties and an untrusted central server orchestrating the overall process. Each data party $\mathcal{P}_i, \forall i \in [m]$, possesses users' data $D_i = \{x^{(1)}_{\mathcal{A}_i}, \ldots, x^{(n)}_{\mathcal{A}_i}\}$ with a subset of attributes $\mathcal{A}_i \subset \mathcal{A}$. We assume that the user's data has been aligned across these m data parties by some record ID (e.g., social security number and phone number) with some private set intersection method [8, 12, 27]. That is, $x^{(l)}_{\mathcal{A}_i}$ and $x^{(l)}_{\mathcal{A}_j}$ are data tuples of a same individual l but on different attributes. The aligned data is a common setting with the vertical tasks [10, 19, 35, 62, 65]. Virtually speaking, there is a global dataset $D = (D_1 | \ldots | D_m)$ with attributes $\mathcal{A} = \cup_{i \in [m]} \mathcal{A}_i$ if all data parties' data can be merged.
Adversary model. We consider both the adversaries within and outside the system. By the adversary within the system, we consider the central server to be honest but curious, which would execute the protocol honestly but try his best to infer the private information of the input dataset. However, we assume that none of the data parties is interested in colluding with the central server because privacy regulations prevent data parties from doing so. We consider the adversary outside the system as all the third-party data analysts who aim to infer some private information of the input dataset from the synthetic dataset and the intermediate results carried out in the communication between data parties and the server.
Our goal. *Our work aims to generate a collection of synthetic data $\hat{D}$ with attributes $\mathcal{A}$, which follows the data distribution as the virtual global dataset D as closely as possible while protecting the privacy information.* Specifically, we employ DP to ensure a high probability, controlled by the privacy budget parameter ϵ, that no adversary can infer based on the synthesized dataset $\hat{D}$ whether any individual's data is used as the input of the data synthesis algorithm.

3.2 Overview of Our Solution

To address the problem defined above, we propose VertiMRF, a novel differentially private data synthesis approach. Figure 1 visualizes the workflow of VertiMRF shown in Algorithm 2, which can be divided into the following six phases:

Algorithm 2 VertiMRF

```
Input: The partitioned dataset $D = \{D_i, i \in [m]\}$, domain $([u_1] \times \ldots \times [u_d])$, maximal clique size $\tau$, total privacy budget $(\epsilon, \delta)$ is
    divided as $\epsilon_0 = \frac{\epsilon}{2m}$, $\delta_0 = \frac{\delta}{2m}$, $\epsilon_1 = \frac{\epsilon}{2}$, $\delta_1 = \frac{\delta}{2}$.
Output: Synthesized data $\hat{D}$.
1: Each local party $\mathcal{P}_i$:
     (a). constructs local MRF: $\{\mathtt{MRF}_i, \mathcal{G}_i, \mathcal{S}_i\} \leftarrow \mathtt{LocMRF}(D_i, \tau, \epsilon_0, \delta_0)$.
2: Each local party $\mathcal{P}_i$:
     (a). encodes local attributes: $\mathcal{M}_i \leftarrow \mathtt{LocEnc}(D_i, \mathcal{A}_i, \epsilon_1, \delta_1)$.
     (b). sends $\mathcal{M}_i$ and $\{\mathtt{MRF}_i, \mathcal{G}_i, \mathcal{S}_i\}$ to server.
3: Central server:
     (a). generates global graph: $\mathcal{G} \leftarrow \mathtt{GraphCom}(\{\mathcal{G}_i, \mathcal{M}_i | i \in [m]\})$.
4: Central server:
     (a). initializes marginal set: $\mathcal{S} \leftarrow \bigcup_{i=1}^{m} \{\mathcal{S}_i\}$.
     (b). initializes parameter $\Theta$ of the global MRF based on $\mathcal{S}$.
5: Central server:
     (a). selects cross-party marginals $\mathcal{S}^c$ from triangulated $\mathcal{G}$.
     (b). optimizes $\Theta$ based on $\mathcal{S}^c$.
6: Central server:
     (a). samples $\hat{D}$ based on the optimized global MRF.
```

- **Phase 1:** Each party $\mathcal{P}_i$ constructs a local Markov Random Field $\mathtt{MRF}_i$ to capture the correlation among local attributes $\mathcal{A}_i$. Besides, $\mathcal{P}_i$ preserves the inner results, including the local attribute graph $\mathcal{G}_i$ and the marginal set $\mathcal{S}_i$ (sub-procedure `LocMRF`).
- **Phase 2:** Each party $\mathcal{P}_i$ encodes local dataset with attributes $\mathcal{A}_i$ via differentially private FM sketch. Both the codes $\mathcal{M}_i$ and $\{\mathtt{MRF}_i, \mathcal{G}_i, \mathcal{S}_i\}$ are sent to the central server (sub-procedure `LocEnc`).
- **Phase 3:** The server generates a global attribute graph $\mathcal{G}$ by combining received disjoint local attribute graphs $\{\mathcal{G}_i | i \in [m]\}$. In the combining, server links up cross-party attribute pairs with higher R-scores estimated over the encoded attributes $\{M_i, i \in [m]\}$. The generated $\mathcal{G}$ is then triangulated (sub-procedure `GraphCom`).
- **Phase 4:** The server initializes a marginal set $\mathcal{S}$ by taking the union of the received local marginal set $\{\mathcal{S}_i | i \in [m]\}$. Based on $\mathcal{S}$, the parameter Θ of the global MRF is initialized with each contingency histogram $T_M, \forall M \in \mathcal{S}$ inferred from the received local `MRF`s (sub-procedure `InitMRF`).
- **Phase 5:** The server selects a set of cross-party marginals $\mathcal{S}^c$ from the cliques of $\mathcal{G}$. Based on the $\mathcal{S}^c$, Θ is further optimized. In the optimization, each contingency histogram $T_M, \forall M \in \mathcal{S}^c$ is estimated over the encoded attributes (sub-procedure `OptMRF`).
- **Phase 6:** The server generates synthetic data by sampling from the data distribution approximated by the global MRF.

In what follows, we show the solution for **Phase 1-2** in Section 4 which describes the DP information sharing approaches of each local party. Then we describe **Phase 3-6** in Section 5, presenting how to use the shared DP information to construct a global MRF.

4 DIFFERENTIALLY PRIVATE INFORMATION SHARING

Based on our security setting and DP's resistance to post-processing, the key to satisfying privacy protection is to ensure differential privacy guarantee for all the information shared from local parties, which are the outputs of `LocMRF` in **Phase 1** and `LocEnc` **Phase 2**

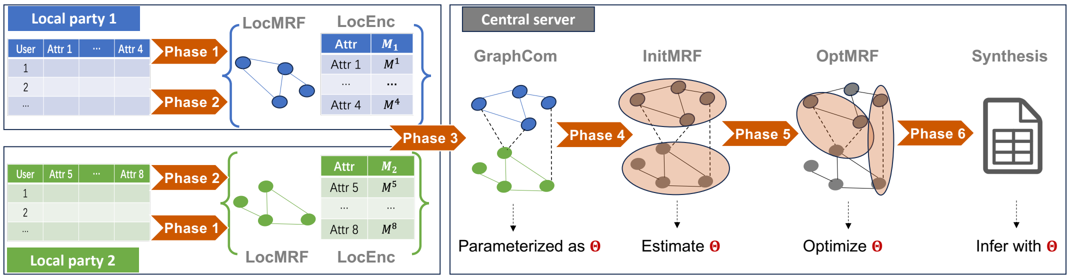

Figure 1: Workflow of VertiMRF

(in brackets Figure 1). Thus, we introduce the algorithms for LocMRF and LocEnc (together with its closely paired CarEst), providing bases of the following synthesis steps.

4.1 Local PrivMRF in Phase 1

Each local party $\mathcal{P}_i$ directly applies the PrivMRF approach to construct $\texttt{MRF}_i$. As shown in Section 2.3, there would be inner results generated when constructing $\texttt{MRF}_i$, including the attribute graph $\mathcal{G}_i$ and the refined marginal set $\mathcal{S}_i$. Apart from $\texttt{MRF}_i$, both $\mathcal{G}_i$ and $\mathcal{S}_i$ should also be preserved and sent to the central server. Notably, because the maximal clique size for the global MRF is always limited to control the complexity of the attribute graph, the maximal clique size of each local MRF should also be limited. The maximal local clique size for each $\texttt{MRF}_i$ is set as $\tau' \leq \frac{\tau}{m \cdot \bar{u}^2}$, with $\bar{u} = \frac{\sum_j u_j}{d}$ and τ is threshold of the clique size for global MRF. The constructed $\texttt{MRF}_i$ captures the correlations among the local attributes.

4.2 Frequency Oracle as a Baseline

Frequency oracle (FO) protocols provide DP protection by randomizing each user's data and allowing the frequency estimation of values in the original domain. In this case, we use the widely known FO protocol, called the Generalized Random Response technique (GRR) [55], to implement a baseline for LocEnc.

FO-based LocEnc. Each party $\mathcal{P}_i$ employs GRR to encode the local dataset. Specifically, for each local attribute $A^j \in \mathcal{A}_i$, a value $v_{(j)}$ is perturbed to an arbitrary $v'_{(j)} \in [u_j]$ with probabilities:

$$Pr\left[v'_{(j)} = v\right] = \begin{cases} \frac{e^{\epsilon'}}{e^{\epsilon'}+u_j-1}, & v = v_{(j)} \\ \frac{1}{e^{\epsilon'}+u_j-1}, & v \neq v_{(j)} \end{cases} \qquad (6)$$

Here, ϵ' denotes LDP level preserved by the GRR technique for each attribute. After applying GRR to each user's data value in the local dataset D_i, we obtain a perturbed version $\widetilde{D}_i$. The partition of $\widetilde{D}_i$ restricted to A^j, denoted as $\mathcal{M}^j$, is taken as the encoded attribute of A^j. Subsequently, the encoded local attributes $\mathcal{M}_i = \{\mathcal{M}^j, \forall A^j \in \mathcal{A}_i\}$ are reported to the central server.

FO-based CarEst. After receiving the reported encoded attributes $\mathcal{M} = \{\mathcal{M}^j | j \in [d]\}$, the central server can estimate the contingency histogram of any arbitrary marginal. For an l-way marginal $M = (A^1, \ldots, A^l)$, we obtain a noisy contingency histogram $\hat{T}_M$ by counting the occurrences of each value tuple $\mathbf{v} = (v_{(1)}, \ldots, v_{(l)}) \in \prod_{i=1}^{l}[u_i]$ from $\mathcal{M}$. However, relying solely on this estimation can introduce considerable bias. To mitigate this issue, a commonly employed technique is to utilize a transition probability matrix P to overcome the bias, which would then produce an unbiased estimate.

As shown in Equation (6), different attributes are encoded independently in LocEnc procedure. So each value tuple $\mathbf{v} = (v_{(1)}, \ldots, v_{(l)})$ can be encoded as any arbitrary $\mathbf{v}' = (v'_{(1)}, \ldots, v'_{(l)})$ with probability $Pr\left[\mathbf{v} \rightarrow \mathbf{v}'\right] = \prod_{A^i \in M} Pr\left[v_{(i)} \rightarrow v'_{(i)}\right]$. Since there are $\prod_{A^i \in M} u_i$ possible values for $\mathbf{v}$ in total, we can construct a $\left(\prod_{A^i \in M} u_i\right) \times \left(\prod_{A^i \in M} u_i\right)$-dimensional probability matrix P to establish the transition relationship between T_M and the noisy $\hat{T}_M$. That is $P \cdot T_M = \mathbb{E}[\hat{T}_M]$, where the expectation accounts for the randomness of GRR. Therefore, T_M can be estimated as $P^{-1} \cdot \hat{T}_M$, where the existence of P^{-1} is guaranteed by the positive definite property of the matrix. Furthermore, it can be shown that

$$\mathbb{E}[P^{-1} \cdot \hat{T}_M] = P^{-1} \cdot \mathbb{E}[\hat{T}_M] = P^{-1} \cdot P \cdot T_M = T_M.$$

This implies that $P^{-1} \cdot \hat{T}_M$ is an unbiased estimator of T_M.

Theorem 3 (**Privacy & Error Analysis**). *Given a marginal $M = (A^1, \ldots, A^l)$, if each attribute $A_i \in M$ is encoded with ϵ'-LDP following the rule shown in Equation (6), then the FO-based* LocEnc *preserves $\left(\min\left\{d\epsilon'/2, 2\epsilon'\sqrt{2d\log(1/\delta)}\right\}, \delta\right)$-DP, $\forall \delta < 1$, where $\delta = 0$ when the minimum taking $d\epsilon'/2$. The FO-based* CarEst *gives unbiased estimation for each count in T_M with variance $O(n/\epsilon'^{2l})$.*

Proof. The details are shown in the Appendix. □

4.3 Sketch-based LocEnc and CarEst

As explained in Section 2.2, FM sketch can be used to estimate the cardinality of a multi-set. And the estimation process can easily satisfy DP by incorporating phantom elements and bounding the maximum value of the hashed geometric variables. Building on this idea, we design our sketch-based LocEnc and CarEst.

Note that our sketch-based estimation is inspired by [34] that utilizes DP FM sketch to encode membership information of local clusters and estimate the cardinality of cross-party clusters' intersection. However, only one attribute is shared by a data party in the clustering task (i.e., which cluster an individual is clustered to), while each data party possesses data with multiple attributes in data synthesis. If we want the server to estimate any cross-party attribute combination with one communication round, using different hash keys for each combination is infeasible as each hash key introduces additional privacy loss. We solve the challenge by answering the following questions: whether encoding multi-dimensional memberships with-in the same party with the same key still provides privacy protection and whether this approach can provide satisfying privacy-utility trade-offs.

Sketch-based `LocEnc`. Each data party $\mathcal{P}_i$ encodes the membership information of local attribute A^j using an FM sketch. This information, denoted as $\left\{\mathbb{I}_{v_1^j}, \ldots, \mathbb{I}_{v_{u_j}^j}\right\}$, consists of u_j ID sets. Each set $\mathbb{I}_{v_i^j}$ contains the IDs of records x with $x_{(A^j)} = v_i^j$. An example illustrating the membership information of attributes is provided in *Example 1*. The sketch-based `LocEnc` involves two main procedures: the generation of hash keys and the generation of sketches.

Example 1: let D be a dataset containing records with attributes Gender, Age and Hobby as in the following table. Then, for attribute Gender, the membership information is $\{\mathbb{I}_{male}, \mathbb{I}_{female}\}$, where $\mathbb{I}_{male} = \{1\}$ and $\mathbb{I}_{female} = \{2, 3\}$.

Index	Gender	Age	Hobby
1	male	20-30	cook
2	female	20-30	basketball
3	female	10-20	cook

Due to the privacy concern, the hash keys should be collaboratively generated by the data parties and kept unknown to the central server. There are multiple secure multi-party computation (SMC) protocols can be applied to achieve this, such as the Diffie-Hellman protocol [28], which allows multi parties to negotiate a random number securely even if the central server is semi-honest [28].

Next, each party encodes the membership information of local attributes using `DPFM` (Algorithm 1) algorithm with the generated hash key ξ. Specifically, for the membership information $\{\mathbb{I}_{v_1^j}, \ldots, \mathbb{I}_{v_{u_j}^j}\}$ of attribute $A^j \in \mathcal{A}_i$, party $\mathcal{P}_i$ applies the `DPFM` algorithm to each $\mathbb{I}_{v_i^j}$ with a given privacy budget ϵ'. This generates a DP FM sketch tuple $(\alpha_{v_1^j}, \ldots, \alpha_{v_{u_j}^j})$ for $A^j \in \mathcal{A}_i$. Considering all local attributes, party $\mathcal{P}_i$ composes a tuple set $\{(\alpha_{v_1^j}, \ldots, \alpha_{v_{u_j}^j}) | A^j \in \mathcal{A}_i\}$. To enhance utility, this process is repeated t times, and party $\mathcal{P}_i$ sends t tuple sets $\left\{\mathcal{M}_i^{(h)} \triangleq \left\{\left(\alpha_{v_1^j}^{(h)}, \ldots, \alpha_{v_{u_j}^j}^{(h)}\right) \middle| A^j \in \mathcal{A}_i\right\} \middle| h \in [t]\right\}$ to the central server. The details of the sketch-based `LocEnc` method are presented in Algorithm 3.

Sketch-based `CarEst`. As mentioned in Section 2.2, the FM sketch enables us to estimate the cardinality of the intersection of multiple sets using the inclusion-exclusion principle. This property can be extended to the DP FM sketch, allowing the central server to estimate the contingency histogram of a marginal. The details of this estimation process are presented in Algorithm 4.

After receiving all the sketches from data parties, the central server aggregates them into t sets of sketch tuples $\left\{\mathcal{M}^{(h)} \triangleq \left\{\left(\alpha_{v_1^j}^{(h)}, \ldots, \alpha_{v_{u_j}^j}^{(h)}\right) | j \in [d]\right\} | h \in [t]\right\}$. For each l-way marginal $M = (A^1, \ldots, A^l)$, the estimation of the contingency histogram T_M involves estimating the cardinality of the intersection set $\bigcap_{i=1}^{l} \mathbb{I}_{v_{(i)}}$ for each $(v_{(1)}, \ldots, v_{(l)}) \in \prod_{i=1}^{l} [u_i]$. Here, $\mathbb{I}_{v_{(i)}}$ represents the membership information of attribute A^i with value $v_{(i)}$. Using the inclusion-exclusion principle (i.e., $\bigcap_{i=1}^{l} \mathbb{I}_{v_{(i)}} = \overline{\bigcup_{i=1}^{l} \overline{\mathbb{I}_{v_{(i)}}}}$), the cardinality of $\bigcap_{i=1}^{l} \mathbb{I}_{v_{(i)}}$ can be determined by calculating the cardinality of $\bigcup_{i=1}^{l} \overline{\mathbb{I}_{v_{(i)}}}$, where $\overline{\mathcal{X}}$ denotes the complementary set of $\mathcal{X}$. Thus, estimating the cardinality of an intersection is transformed into estimating the cardinality of the complementary set of a union. The basic approach to estimate $\left|\overline{\bigcup_{i=1}^{l} \mathbb{I}_{v_{(i)}}}\right|$ is as follows: first, estimate $\left|\bigcup_{i=1}^{l} \overline{\mathbb{I}_{v_{(i)}}}\right|$ using the mergeable property of sketches, and then subtract this estimate from a DP sanitized data number $\hat{n}$.

Algorithm 3 Sketch-based `LocEnc`

```
Input: $\mathcal{P}_i$'s local dataset $D_i$, attribute $A^j \in \mathcal{A}_i$, distribution parameter $\gamma$, privacy budget $(\epsilon, \delta)$, domain $([u_1] \times \ldots \times [u_l])$.
Output: $\mathcal{P}_i$'s sketch set $\mathcal{M}_i$.
Data parties generate $t$ hash keys $\{\xi_1, \ldots, \xi_t\}$ collaboratively.
$\epsilon' = \frac{\epsilon}{4\sqrt{td \log(1/\delta)}}$
for $h \in [t]$ do
    for each attribute $A^j \in \mathcal{A}_i$ do
        for $l \in [u_j]$ do
            $\alpha_{v_l^j}^{(h)} = \mathtt{DPFM}(D_i, u_j, \gamma, \epsilon', \xi_h)$
        end for
    end for
    $\mathcal{M}_i^{(h)} \leftarrow \left\{\left(\alpha_{v_1^j}^{(h)}, \ldots, \alpha_{v_{u_j}^j}^{(h)}\right) | A^j \in \mathcal{A}_i\right\}$
end for
return $\{\mathcal{M}_i^{(h)} | h \in [t]\}$
```

For each $\mathcal{M}^{(h)}$ among all t sketch sets, the sketch of $\overline{\mathbb{I}_{v_{(i)}}}$ can be estimated by $\max\left\{\alpha_{v_j^i}^{(h)} | j \in [u_i], v_l^i \neq v_{(i)}\right\}$. Here, $\alpha_{v_j^i}^{(h)}$ represents the sketch corresponding to attribute A^i with value v_j^i. Furthermore, the sketch of $\bigcup_{i=1}^{j} \overline{\mathbb{I}_{v_{(i)}}}$ can be estimated by $\max\left\{\max\left\{\alpha_{v_j^i}^{(h)} | j \in [u_i], v_j^i \neq v_{(i)}\right\} | A^i \in M\right\}$. After obtaining t estimates of the sketch of $\bigcup_{i=1}^{j} \overline{\mathbb{I}_{v_{(i)}}}$, a more stable and accurate estimate α can be obtained by taking the harmonic mean. Furthermore, since the above sketch estimation process involves max operations on $\sum_{i=1}^{l} (u_i - 1)$ sketches, each of which introduces k_p phantom elements as shown in Algorithm 1, there should be $\left(\sum_{i=1}^{l} (u_i - 1)\right) \cdot k_p$ phantom elements taken into account in total. By subtracting those phantom elements, $\left|\bigcup_{i=1}^{l} \overline{\mathbb{I}_{v_{(i)}}}\right|$ can be estimated by $(1+\gamma)^{\alpha} - \left(\sum_{i=1}^{l} (u_i - 1)\right) \cdot k_p$. Finally, the cardinality of $\bigcap_{i=1}^{l} \mathbb{I}_{v_{(i)}}$ can be obtained by subtracting the estimated $\left|\bigcup_{i=1}^{l} \overline{\mathbb{I}_{v_{(i)}}}\right|$ from a DP sanitized data number $\hat{n}$ and ensuring the non-negativity.

Notice that `CarEst` operates on sketches that are generated with privacy guarantees. Therefore, `CarEst` does not consume additional privacy budget due to the post-process property of DP.

THEOREM 4 (**PRIVACY ANALYSIS**). *Suppose the FM sketch $\alpha_{v_j^i}^{(h)}$ for value v_j^i of attribute A^i is generated with ϵ'-DP in the h-th run. Then, the sketch-based `LocEnc` method in Algorithm 3 guarantees $(4\epsilon'\sqrt{td\log(1/\delta)}, \delta)$-DP for all $\delta < 1$.*

Algorithm 4 Sketch-based `CarEst`

```
Input: Marginal $M = (A^1, \dots, A^l)$, domain $([u_1] \times \dots \times [u_l])$,
    sketch sets $\left\{\mathcal{M}^{(h)} \triangleq \left\{\left(\alpha^{(h)}_{v^j_1}, \dots, \alpha^{(h)}_{v^j_{u_j}}\right) | j \in [d]\right\} | h \in [t]\right\}$,
    privacy budget $(\epsilon, \delta)$, distribution parameter $\gamma$, noisy data number $\hat{n}$.
Output: Estimated contingency histogram $T_M$.
$\mathcal{T} \leftarrow \mathbf{0}^{t\times(u_1\times\ldots\times u_l)}, T_M \leftarrow \mathbf{0}^{(u_1\times\ldots\times u_l)}$
$\epsilon' = \frac{\epsilon}{4\sqrt{td\log(1/\delta)}}, k_p = \lceil\frac{1}{e^{\epsilon'-1}}\rceil$
for all $h \in [t]$, $\left(v_{(1)}, \dots, v_{(l)}\right) \in ([u_1] \times \dots \times [u_l])$ do
    $\mathcal{T}[h, (v_{(1)}, \dots, v_{(l)})] = \max\left\{\max\left\{\alpha^{(h)}_{v^i_l} | l \in [u_i], v^i_l \neq v_{(i)}\right\} | A^i \in M\right\}$
end for
for all $(v_{(1)}, \dots, v_{(l)}) \in (u_1 \times \dots \times u_l)$ do
    $\alpha = \mathtt{HarmonicMean}\left(\mathcal{T}[:, (v_{(1)}, \dots, v_{(l)})]\right)$
    $T_M[(v_{(1)}, \dots, v_{(l)})] = (1+\gamma)^\alpha - \sum_{i=1}^{l}(u_i - 1) \cdot k_p$
    $T_M[(v_{(1)}, \dots, v_{(l)})] = \max\left\{\hat{n} - T_M[(v_{(1)}, \dots, v_{(l)})], 0\right\}$
end for
return $T_M$
```

Theorem 5 (**Error Analysis**). *Let $M = \{A^1, \dots, A^l\}$ be an l-way marginal. Suppose $\hat{T}_M$ is the contingency histogram of M estimated using Algorithm 4 with privacy parameter (ϵ, δ) and distribution parameter γ. For each $\mathbf{v} \in \prod_{i=1}^{l}[u_i]$, the following inequality holds:*

$$\frac{|\hat{T}_M[\mathbf{v}] - T_M[\mathbf{v}]|}{T_M[\mathbf{v}]} \le \gamma \cdot (\frac{n}{T_M[\mathbf{v}]} - 1) + \frac{\hat{N} + C}{T_M[\mathbf{v}]}, \tag{7}$$

with a probability of at least $1 - \beta$. Here, $\hat{N}$ represents the Laplacian noise added to the data number n, and $C = O(\frac{\log^{1/2}(1/\delta)\log^{1/4}(1/\beta)}{\epsilon'})$.

Due to the space limitation, the proofs are shown in the Appendix. As shown in Theorem 5, the relative error tends to be larger when the proportion of $T_M[\mathbf{v}]$ in n decreases and when the count $T_M[\mathbf{v}]$ decreases. Meanwhile, a stronger privacy level, represented by a smaller value of ϵ, can indeed lead to a larger relative error, as indicated by the term C in the theorem.

4.4 Privacy and Communication Cost

Overall Privacy Analysis. As shown in the workflow of VertiMRF in Algorithm 2, LocMRF on all m local parties consumes $(m \cdot \frac{\epsilon}{2m}, m \cdot \frac{\delta}{2m})$-DP. As stated in Theorem 4, the remaining $(\frac{\epsilon}{2}, \frac{\delta}{2})$-DP is allocated for encoding the d attributes for t iterations in LocEnc. According to the sequential composition property of DP, we can conclude that VertiMRF implemented as in Algorithm 2 satisfies (ϵ, δ)-DP.

Communication cost. There is one communication round between each party $\mathcal{P}_i$ and the central server in VertiMRF. The communication includes encoded attributes $\mathcal{M}_i$ and the local MRF information $\{\mathsf{MRF}_i, \mathcal{S}_i, \mathcal{G}_i\}$. For sketch-based LocEnc, $\mathcal{M}_i$ contains $t\sum_{A^j \in \mathcal{A}_i} u_j$ sketches. For FO-based LocEnc, $\mathcal{M}_i$ contains a noisy version of local dataset. MRF_i is parameterized by a vector Θ with length $\sum_{M \in S_i}\prod_{A^j \in M} u_j$, controlled by the maximal clique size τ' for each local MRF. $\mathcal{G}_i$ is represented by a $(|\mathcal{A}_i| \times |\mathcal{A}_i|)$-dimensional adjacent matrix, with $|\mathcal{A}_i| < d$. The information in $\mathcal{S}_i$, which contains serveral attribute tuples, can be ignored in terms of communication costs. Considering a total of m parties, the communication cost of VertiMRF is $O(td\bar{u}) + O(d^2) + O(m\tau^{'})$ when using sketch-based LocEnc and $O(nd) + O(d^2) + O(m\tau^{'})$ when using FO-based. Here $\bar{u}$ represents the average domain size of attributes.

5 MRF GENERATION IN CENTRAL SERVER

After receiving local MRFs and encoded attributes $\mathcal{M}$ from all parties, the process of the central server can be divided into the following phases: generating the global attribute graph (**Phase 3**), initializing the marginal set thereby estimating the MRF parameter (**Phase 4**), refining the marginal set thereby optimizing the MRF parameter (**Phase 5**) and finally sampling the synthetic data (**Phase 6**).

5.1 `GraphCom` in Phase 3

Since the local attribute graphs are disjoint and each one accurately represents the correlation among a subset of attributes, a basic approach to creating a global attribute graph is to combine the disjoint graphs by linking up certain cross-party attribute pairs. However, there are two constraints (CSTR) that must be satisfied when selecting such cross-party attribute pairs, denoted as (A^i, A^j):

●CSTR1: (A^i, A^j) should exhibit strong correlation.

●CSTR2: The domain size of maximal cliques in the resulting attribute graph should not exceed a predefined threshold value τ.

To satisfy CSTR1, the central server estimates the R-score [6] $R(A^i, A^j)$ for each cross-party attribute pair (A^i, A^j) with CarEst approach introduced in Section 4 over the received encoded attributes $\mathcal{M}$: $R(A^i, A^j) \approx \frac{\hat{n}}{2}\left\|\frac{\hat{T}_{(A^i, A^j)}}{\hat{n}} - \frac{\hat{T}_{A^i}}{\hat{n}} \cdot \frac{\hat{T}_{A^i}}{\hat{n}}\right\|_1$, where $\hat{T}$ denotes the estimated contingency histogram. As explained in Section 2.3, attribute pairs with higher R-scores indicate stronger correlation. After the estimation, the server sorts all attribute pairs in descending order based on their estimated R-scores and greedily connects them in the global attribute graph.

For CSTR2, whenever a link between cross-party attributes is added to $\mathcal{G}$, the server checks the domain size of the maximal clique in the triangulated $\mathcal{G}$ to ensure it does not exceed τ. τ is always set empirically to strike the tradeoff between the model utility and computation complexity. A larger τ enables more flexible marginal selection but incur high computational efficiency. According to our observation, $[10^5, 5 \times 10^6]$ is an suitable range for τ. If CSTR2 is satisfied, the process of adding links continues. This process continues until it is no longer possible to satisfy CSTR2.

5.2 `InitMRF` in Phase 4

After generating the global attribute graph, the next step is to construct the global MRF. As shown in Section 2.3, the MRF construction process essentially is to learn a parameter vector Θ on a marginal set $\mathcal{S}$. In PrivMRF, $\mathcal{S}$ is first initialized by selecting the most highly correlated marginal for each attribute and then refined through adding marginals which cannot be accurately inferred by the global MRF. Meanwhile, Θ is initialized and optimized by reducing the error of inferring the marginals in $\mathcal{S}$ using a mirror descent algorithm [6]. The initialization of $\mathcal{S}$ serves to initialize the global MRF and select an reliable direction for the subsequent refinement of $\mathcal{S}$ and learning of Θ. we follow this process and initialize an $\mathcal{S}$ to initialize the global MRF.

However, unlike PrivMRF, the central server in our setting lacks access to the raw data, it is not practical to compute sufficiently accurate correlations among each attribute with multiple marginals to select the highly correlated ones. More severely, the true value of the contingency histograms are unavailable to compute the inferring error of Θ. Therefore, it is essential to select marginals that can be accurately estimated based on the DP shared information of local parties. With the observation that local MRFs can estimate their marginals with relative low error, we take the union of the local marginal sets as the initialized $\mathcal{S}$, that is $\mathcal{S} = \cup_{i=1}^{m} \mathcal{S}_i$ and take the local MRF inferred contingency histograms $\cup_{i=1}^{m} \{\mathtt{MRF}_i(M) | \forall M \in \mathcal{S}_i\}$ as the ground truth of contingency histograms, where $\mathtt{MRF}_i(M)$ refers to the inferred result of $\mathtt{MRF}_i$ on M. Based on this ground truth and the initialized $\mathcal{S}$, the server initializes the global MRF.

By observing Equation (4), we notice that an MRF encodes the correlation among multiple attributes by representing the marginals in the marginal set. Based on this observation, We can say that the inferred results of each $\mathtt{MRF}_i$ on the marginals in the marginal set $\mathcal{S}_i$ encapsulate all the "knowledge" encoded by the MRF. Therefore, such initialization can also be viewed as transferring the knowledge from the local MRFs to the global MRF.

5.3 OptMRF in Phase 5

After initialization, the encoded correlation in the local MRFs has already been transferred to the global MRF. However, the correlation among cross-party attributes has not been characterized by the global MRF. To address this, the central server further refines $\mathcal{S}$ by inserting cross-party marginals whose contingency histograms can be estimated using the CarEst approach over the encoded local attributes $\mathcal{M}$. To minimize noise, we mainly select the low-way cross-party marginals denoted as $\mathcal{S}^c$. Specifically, the average count in each cell of T_M of each marginal $M \in \mathcal{S}^c$ is controlled to be larger than a threshold d^c, as given by $\frac{\hat{n}}{\prod_{A^i \in M} |u_i|} \geq d^c$, where d^c controls the error of the estimation of CarEst, which ia also set empirically. As shown in Theorem 5, accurate estimation of CarEst becomes challenging with small average counts in the contingency histogram. It is worth noting that the optimization of Θ involves multiple rounds. In each round, the server randomly samples several cross-party marginals from $\mathcal{S}^c$ and optimize Θ mainly on the ones with significant inferring error, as measured by the L1 distance between the inferred histograms of the global MRF and the true values estimated using CarEst over the encoded attributes $\mathcal{M}$.

Once the global MRF is constructed, approximating the data distribution and sampling synthetic data becomes a straightforward task. For detailed information, please refer to [6].

6 IMPROVEMENTS

While **Phase 1 - 5** with details introduced in Section 4 and Section 5 can compose a complete algorithm for differentially private vertical data synthesis, we encounter a dilemma when optimizing the algorithm by tuning the granularities of the attributes. With a coarser granularity, the LocEnc-CarEst can have smaller relative errors, but the LocMRF and global MRF becomes inferior to its best performance with more fine-grained granularity. Thus, configuring different granularities for those parts can be an alternative improvement. However, there are two issues for this inconsistent granularity solution: 1) how to reduce dimension while keeping as much information as possible; 2) how to enforce consistency between the frequencies of different granularities.

6.1 Dimension reduction

As stated in Theorem 5, when the domain sizes of attributes increase, the estimated cross-party marginals by CarEst can deviate significantly. This deviation occurs because the expected number of data points in each cell of the contingency histogram decreases. To address this issue, we propose binning the attributes to reduce the domain sizes. The binned attributes are encoded using LocEnc and sent to the server (**Phase 2**). The encoded attributes are then used to calculate R-scores in the GraphCom procedure (**Phase 3**) and estimate the cross-party marginals to optimize the global MRF (**Phase 5**). However, since the global MRF is constructed based on the raw attributes without binning, the estimated contingency histogram cannot be used directly. To overcome this, we employ a histogram recovery technique to transform the estimated low-dimensional histograms of the binned attributes into the high-dimensional ones.

The basic idea is to approximate the high-dimensional distributions using low-dimensional ones. According to the joint distribution formula, when considering (A, B) as a marginal for estimation, where X and Y are the binned versions of A and B respectively, the high-dimensional marginal distribution can be estimated as $Pr[A, B] \approx \sum_{(X,Y)} Pr[X, Y] \cdot Pr[A|X] \cdot Pr[B|Y]$. Here, $Pr[X, Y]$ represents the low-dimensional distribution over the binned attributes, while $Pr[A|X]$ and $Pr[B|Y]$ are referred to as value distributions and are also low-dimensional. Figure 2 provides a visualization of our dimension reduction technique.

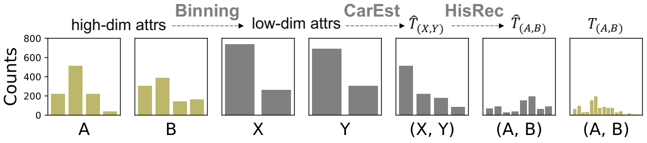

Figure 2: Instantiation of Dimension Reduction.

Attribute binning. Each party $\mathcal{P}_i$ applies equal-width binning to each local attribute $A^j \in \mathcal{A}_i$ before LocEnc. The number of bins b is specified by the server or negotiated by data parties. Equal-width binning is based solely on the domain size of each attribute and does not expose any statistical information from the raw data. However, to facilitate the subsequent histogram recovery, it is necessary to preserve the value distribution within each bin for each attribute during the binning process and send it to the server.

For an attribute A^j with a domain size of u_j, the values are allocated to b bins with equal width. Suppose the values $v_i^1, ..., v_i^k$ are allocated to the l-th bin, and their corresponding frequencies are $n_i^1, ..., n_i^k$. The distribution of the l-th bin of A^j is defined as:

$$U_{(j,l)} = \left[U_{(j,l)}^1, \ldots, U_{(j,l)}^k \right] \triangleq \left[\frac{n_i^1}{\sum_{h=1}^k n_i^h}, \ldots, \frac{n_i^k}{\sum_{h=1}^k n_i^h} \right].$$

Since the frequencies are obtained from the raw data, the resulting value distributions may expose sensitive statistical information.

To ensure privacy, we utilize the Laplacian mechanism to perturb the frequencies with a sensitivity of 1 and a privacy budget of ϵ'. The value distributions are then computed based on the noisy frequencies. Considering the sequential composition of DP, the total privacy cost of the overall binning procedure should be $d\epsilon'$.

Histogram recovery (HisRec). Let's assume the contingency histogram of the marginal (A^i, A^j) estimated by CarEst over the binned attributes is denoted as $\hat{T}^{(low)}$. In $\hat{T}^{(low)}$, $\hat{T}^{(low)}[(v'_{(i)}, v'_{(j)})]$ represents the number of data points falling in the $v'_{(i)}$-th bin of A^i and the $v'_{(j)}$-th bin of A^j. We can recover the high-dimensional histogram $\hat{T}^{(high)}$ by estimating the number of data points falling in each cell where $A^i = v_{(i)}$ and $A^j = v_{(j)}$ as:

$$\hat{T}^{(high)}\left[(v_{(i)}, v_{(j)})\right] = \hat{T}^{(low)}\left[(v'_{(i)}, v'_{(j)})\right] \cdot U^{h'_{(i)}}_{(i, v'_{(i)})} \cdot U^{h'_{(j)}}_{(j, v'_{(j)})}.$$

where $h'(i)$ and $h'(j)$ represent the value index of $v_{(i)}$ allocated to the $v'_{(i)}$-th bin of A^i and $v_{(j)}$ allocated to the $v'_{(j)}$-th bin of A^j. Although demonstrated with the two-way marginal case, this technique can be easily extended to higher-way marginal cases.

6.2 Consistency enforcement

As discussed in Section 5.3, estimating contingency histograms for intra-party and cross-party marginals in the marginal set $\mathcal{S}$ is vital for constructing the global MRF. Intra-party marginals are estimated using local MRFs, while cross-party marginals are estimated using CarEst. Nevertheless, variations in the sources of randomness can cause inconsistencies between the estimated histograms from local MRFs and CarEst for specific attribute sets.

Let's consider a two-way cross-party marginal, denoted as (A^i, A^j). The contingency histogram estimated by CarEst is denoted as $\hat{T}_{(A^i,A^j)}$, or simply $\hat{T}$. If we marginalize $\hat{T}$ to obtain $\hat{T}_{(A^i)}$ and $\hat{T}_{(A^j)}$, these results may exhibit inconsistencies with the histograms $\widetilde{T}_{(A^i)}$ and $\widetilde{T}_{(A^j)}$ inferred from local MRFs.

To address this inconsistency, we employ a two-step technique to ensure consistency among $\hat{T}, \widetilde{T}_{(A^i)}, \widetilde{T}_{(A^j)}$. Firstly, we transform all three contingency histograms into marginal distributions by normalizing them. For simplicity, we continue to use the notation T to represent the marginal distribution. The two-step technique operates on each attribute individually, taking A^i as an example.

Step 1: consistency. We begin by establishing agreement between $\widetilde{T}_{(A^i)}$ and $\hat{T}_{(A^i)}$ by taking their arithmetic mean: $\bar{T} = \frac{\widetilde{T}_{(A^i)} + \hat{T}_{(A^i)}}{2}$. We then update both $\hat{T}$ and $\widetilde{T}_{(A^i)}$ to be consistent with $\bar{T}$. Specifically, $\widetilde{T}_{(A^i)}$ is directly set to $\bar{T}$. As for $\hat{T}$, changing a cell in $\hat{T}_{(A^i)}$ would affect u_j cells in $\hat{T}$ (where u_j is a constant). To maintain the unchanged marginal distribution $\hat{T}_{(A^j)}$, we calculate the difference between $\hat{T}(A^i)$ and $\bar{T}$ for each cell when A^i takes a specific value $v_{(i)}$, and then distribute the difference equally among all u_j affected cells: $\hat{T}[(v_{(i)}, v_{(j)})] = \hat{T}[(v_{(i)}, v_{(j)})] + \frac{\bar{T}[v_{(i)}] - \hat{T}_{(A^i)}[v_{(i)}]}{u_j}$.

Step 2: normalization. After the consistency step, negative numbers may appear in the marginal distribution. To ensure non-negativity, we set all negative numbers to 0. However, this adjustment may cause the sum of the distribution $\hat{T}$ to exceed 1. To address this, we re-normalize $\hat{T}$. Nevertheless, this re-normalization may introduce inconsistency between $\hat{T}_{(A^i,A^j)}$ and $\widetilde{T}(A^i)$ again. To mitigate this issue, we repeat the consistency and normalization process for each attribute multiple times until the resulting inconsistency becomes negligible. Finally, the consistent marginal distribution is transformed into a contingency histogram by multiplying it with a DP-sanitized data number $\hat{n}$.

Table 1: Characteristics of Datasets

Dataset	Records	Attrs	Dom.	Dom. Size	Label
NLTCS	21574	16	2	$\approx 6 \times 10^4$	each
Adult	45222	15	2-42	$\approx 4 \times 10^{14}$	salary
BR2000	38000	13	2-21	$\approx 3 \times 10^9$	income
Fire	305119	15	2-46	$\approx 1 \times 10^{15}$	ALT unit

This consistency enforcement technique is employed in **Phase 5** and can be directly extended to higher-way marginal cases.

7 EXPERIMENTS

In this section, we first conduct the end-to-end comparisons of VertiMRF to baseline methods. Then, we validate each component by conducting the ablation experiments. The final experimental results demonstrate the superiority of VertiMRF.

7.1 Experiment settings

Datasets. We evaluate our algorithms on four datasets in Table 1.

- **Adult [3].** The data is sourced from the United States Census Bureau. It consists of 45,222 instances, each containing 15 attributes capturing demographic and socio-economic information, such as age, race, education, and income level.
- **NLTCS [37].** The data is collected from a study on health status of older adults. It includes 21,574 records and 16 attributes that describe demographic information and health conditions.
- **BR2000 [48].** The data originates from a population Census in Brazil and contains 38,000 records. It includes 13 attributes that provide information about demographic, and economic aspects.
- **Fire [47].** The data includes records of fire unit responses to calls in San Francisco. It consists of 305,119 records, with each record containing 15 attributes.

Metrics. We evaluate the performance based on two metrics.

- ***l*-way TVD.** We randomly sample 300 marginals with l attributes each from the synthetic data. For each marginal, we compute the total variation distance (TVD) between the raw and synthetic data. We calculate the average TVD for l values of 3, 4, 5 across all marginals and report the average measurement based on 10 iterations.
- **Misclassification rate.** We employ synthetic data to train SVM classifiers for predicting specific attributes based on all other attributes. The predicted attribute for each dataset is shown in the "label" row of Table 1. For NLTCS, each attribute serves as the label to predict, and the average result is reported. We use 80% of the raw data to generate synthetic data and train the classifier, while the remaining data is used as the test set to report the misclassification rate. We utilize 5-fold cross-validation and report the average.

Compared methods. We compare the following methods.

- **VertiGAN** [23] employs a partitioned GAN with a multi-output global generator and multiple local discriminators. To ensure privacy, local discriminators are trained with DP-SGD [1] using raw data. The global generator is updated by aggregating the local gradients. The privacy budget is fully utilized during the DP-SGD procedure of local discriminator.

- **Centralized** refers to PrivMRF [6] in the centralized setting.
- **VertiMRF-FO** & **VertiMRF-FM.** Both methods are based on our proposed VertiMRF framework, with one equipped with FO-based and the other with sketch-based LocEnc approach.

In addition, we encountered the DPLT approach [51], which utilizes a latent tree structure to capture attribute correlations. Despite our diligent efforts to replicate DPLT, we encountered significant computation overhead when calculating the marginal distribution of high-level latent attributes. We also faced ambiguity regarding data synthesis from the constructed tree. As a result, we have chosen not to include DPLT in our comparison.

Parameter setting. In our experiments, we use default values for VertiMRF-FM, setting the repetition number of the DP FM sketch to $t = 2000$ and $\delta = 1/n$. The network structure of VertiGAN follows the configuration described in the original paper [23] and the privacy is tracked with RDP. For all datasets except NLTCS, we set the binning number to $b = 4$. As for the privacy budget allocation, we allocate 40% to LocMRF, 40% to LocEnc (with 10% of the 40% used to generate a noisy data count $\hat{n}$), and the rest 20% for sanitizing the value distributions in the binning procedure.

7.2 End to End Comparisons

Comparison on l-way TVD. Figure 3 compares the methods based on the average TVD of the l-way marginals. As shown, VertiMRF-FM and VertiMRF-FO consistently produce smaller TVD than Verti-GAN regardless of privacy cost or dataset. This demonstrates the superiority of VertiMRF. Additionally, VertiMRF-FM outperforms VertiMRF-FO across all cases, indicating that the sketch-based LocEnc and CarEst can provide more accurate estimation of the cross-party marginals compared to the FO-based approaches. It is worth noting that VertiGAN consistently yields significantly larger TVD results. This can be attributed to the fact that GAN-based data synthesis methods are not well-suited for synthesizing tabular data, as discussed in previous studies [6, 39]. Furthermore, the advantages of Centralized over VertiMRF-FM become more prominent in datasets with larger domain sizes, such as adult, BR2000, and Fire. Although VertiMRF-FM performs closely to Centralized on NLTCS when ϵ is larger, the difference becomes more pronounced in the other three datasets due to their larger domain sizes. This aligns with our analysis in Theorem 5. A larger domain size leads to smaller average count in a contingency histogram, thereby deriving a larger estimation error of CarEst.

Comparison on SVM classification. Figure 4 presents the average misclassification rates of the SVM classifiers trained on the synthetic data. VertiMRF-FM consistently outperforms the other vertical DP methods. The misclassification rates of VertiGAN is as high as 40% even when $\epsilon = 3.2$, which is significantly larger than both VertiMRF-FM and VertiMRF-FO methods. Additionally, the advantages of the centralized algorithm over VertiMRF-FM are magnified as the domain size of the dataset increases. These findings align with the results shown in Figure 3, which illustrate the effectiveness of VertiMRF-FM and the negative impact of large domain sizes on the utility of synthetic data.

Impact of the number of parties. We examine the impact of the party number m on the utility of synthetic data. Figure 5 compares the TVD results obtained under different m settings on the NLTCS and Adult datasets with a privacy budget $\epsilon = 0.8$. In the experiments, $m = ALL$ indicates that the attributes are distributed to d parties, with each party having one distinct attribute. The results demonstrate that as m increases, the TVD results also increase. This is primarily because when attributes are partitioned to more parties, LocMRF with high precision contributes less to the global MRF. In such cases, the LocEnc and CarEst procedures dominate the errors.

Impact of privacy budget allocation. We examine the impact of privacy budget allocation on the utility of synthetic data. In this study, we fix the total privacy budget as $\epsilon = 0.8$ and compare the TVD results on the NLTCS dataset under different privacy budget ratio θ assigned to LocMRF. The remaining $(1 - \theta)$ proportion of the privacy budget is then fully allocated to LocEnc. Figure 6 illustrates that as θ increases, the TVD results initially decrease when $\theta \leq 0.2$, but then increase when $\theta > 0.2$. These results highlight the tradeoff between the impacts of LocEnc and LocMRF. LocMRF aids in estimating the intra-party marginals, while LocEnc aids in estimating the cross-party marginals. When θ is small, the error is dominated by inaccurate estimation of the intra-party marginals. Conversely, when θ is larger, the error is mainly caused by inaccurate estimation of the cross-party marginals.

Communication and computation cost. In Table 2, we compare the communication costs and computation time of the four methods on Adult dataset. As analyzed in Section 4.4, the communication overhead of VertiMRF-FM is expected to be smaller than that of VertiMRF-FO when $t\bar{u} < n$. Consistent with our analysis, we observe that the overhead of VertiMRF-FM is smaller than that of VertiMRF-FO when $t\bar{u} < n$ with $t = 2000$ but larger when $t\bar{u} > n$ with $t = 8000$. The communication in VertiGAN involves sending gradients of local generators to the server and broadcasting the updated model to the local parties. Therefore, the overall communication cost depends on the model size and the number of communication rounds.

In terms of computation time, when using the sketch-based LocEnc, each local party needs to perform tn hash mappings, whereas the FO-based LocEnc only requires $n|A_i|$ perturbations. Since $t >> |A_i|$, the FO-based LocEnc requires less computation time. The hash mappings can be accelerated by parallel computation since they run independently. By introducing 40 parallel threads, the computation time can be significantly reduced. On the server side, the computation time is nearly identical for both VertiMRF-FM and VertiMRF-FO. That's because apart from CarEst, both methods execute identical computations on the server side. Whether it is FO-based CarEst or sketch-based CarEst, the computation process solely involves simple calculations and does not significantly affect the computation time. In VertiGAN, each party generates fake data and computes model gradients, while the server aggregates and broadcasts the updated model. Therefore, the most time consumption occurs at the local party.

7.3 Ablation studies

Effect of LocMRF. Data parties generate local MRFs to help infer the intra-party marginals. However, simply using LocEnc and CarEst

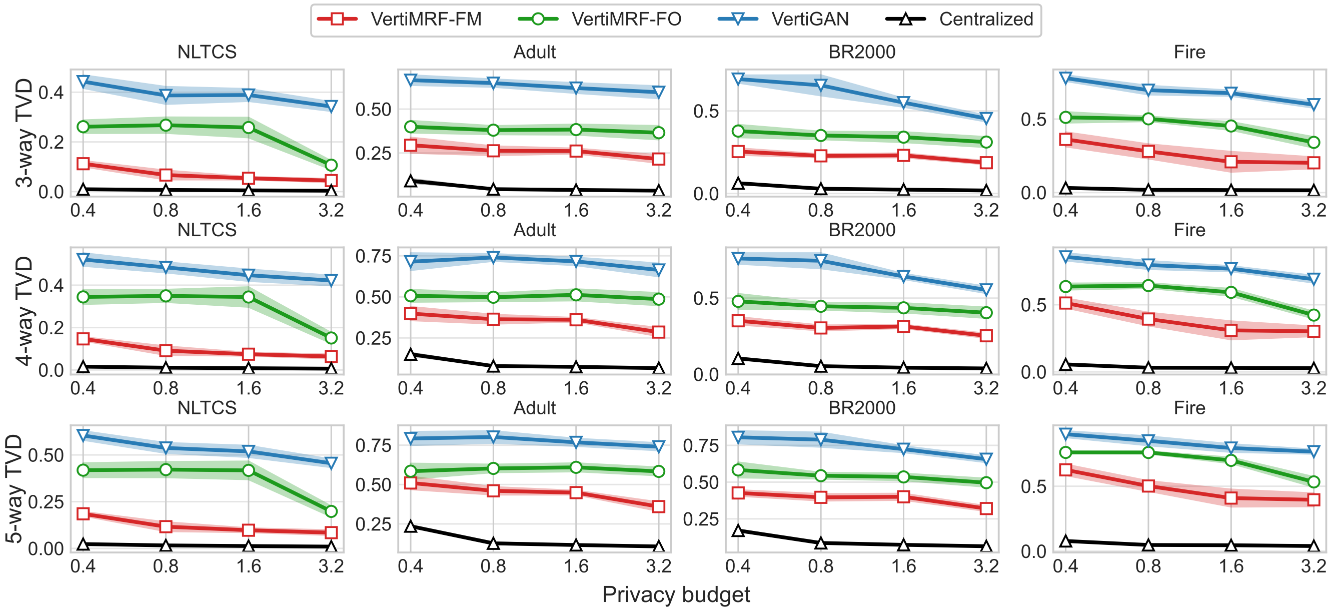

Figure 3: l-way TVD vs. privacy budget ϵ.

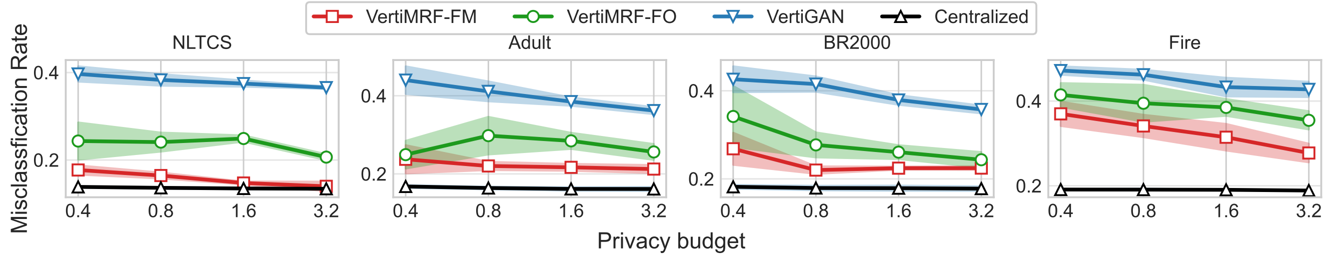

Figure 4: SVM misclassification rate vs. privacy budget ϵ.

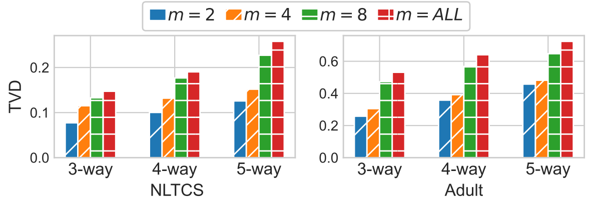

Figure 5: Impact of party number ($\epsilon = 0.8$).

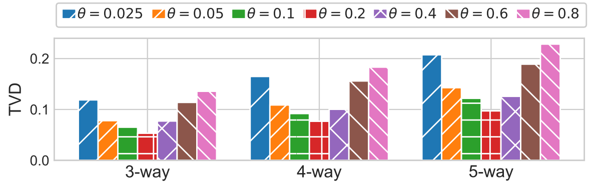

Figure 6: Impact of privacy budget allocation ($\epsilon = 0.8$).

Table 2: Communication cost and computation time

Dataset	methods	commu. cost	compu. time per party	compu. time server
Adult	VertiMRF-FM ($t = 2000$)	15 Mb	23.1 min	67 min
	VertiMRF-FM ($t = 2000$, *threads* = 40)	4.9 Mb	4.1 min	67 min
	VertiMRF-FM ($t = 8000$)	22 Mb	93 min	67 min
	VertiMRF-FO	18 Mb	2.5 min	67 min
	VertiGAN	112 Mb	8.3 min	10 s
	Centralized	-	-	16 min

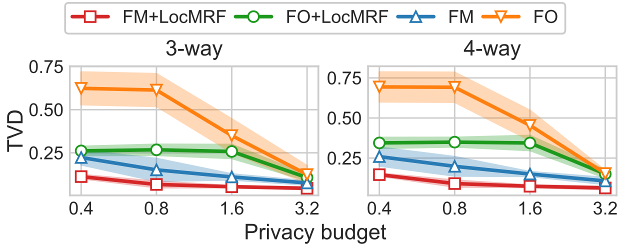

Figure 7: Effect of LocMRF on NLTCS.

can also achieve the marginal estimation. In Figure 7, we compare TVD on NLTCS when synthesizing data with and without using LocMRF respectively. Specifically, different LocEnc methods are considered in LocMRF, which are labeled as "FM+LocMRF" and "FO+LocMRF", respectively. As shown, for both "FM+LocMRF" and "FO+LocMRF", the TVD results are smaller than those when simply

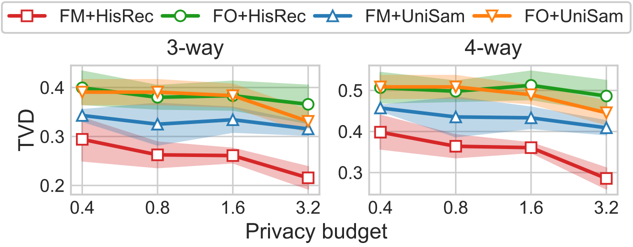

Figure 8: Effect of histogram recovery (HisRec) on Adult.

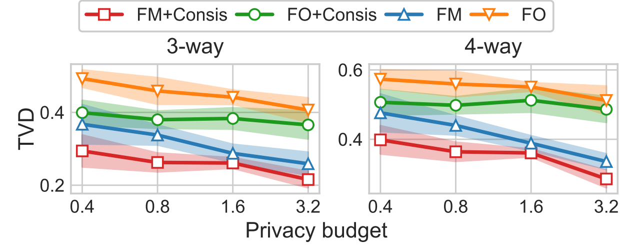

Figure 9: Effect of enforcing consistency on Adult.

using the sketch or FO-based LocEnc. This demonstrates the effectiveness of LocMRF. Furthermore, we also find that using LocMRF can dramatically reduce the variances of the generated TVD results. This is reasonable since LocMRF can capture the correlations among local attributes, thereby reducing the uncertainty when estimating the intra-party marginals.

Effect of histogram recovery. We demonstrate the effectiveness of our proposed histogram recovery (HisRec) by comparing it to the baseline that generates high-dimensional histogram via uniformly allocating the count in each cell of estimated low-dimensional histogram (denoted as UniSam) to the corresponding multiple cells of the high-dimensional histogram. For a fair comparison, the privacy budget for sanitizing the value distributions in HisRec is allocated to LocEnc in UniSam. As shown in Figure 8, FM+HisRec yields superior TVD results compared to FM+UniSam, which demonstrates the effectiveness of HisRec when being used in conjunction with FM sketch-based LocEnc and CarEst approaches. However, we also find that FO+HisRec performs closely to FO+UniSam when the privacy budget $\epsilon < 3.2$ and even worse when $\epsilon = 3.2$, that's mainly because the low-dimensional histogram estimated by FO-based CarEst is too noisy and the noise has dominated the estimation error. Then without prior knowledge of true distribution, uniform allocating shows an advantage as an impartial method.

Effect of consistency enforcement. As discussed in Section 6.2, we propose to improve marginal estimation by ensuring consistency among the histograms estimated by CarEst and local MRFs. In Figure 9, we compare TVD on the synthetic Adult dataset obtained with (denoted as +Consis) and without enforcing consistency. As shown, the TVD values under +Consis settings are consistently smaller than those without ensuring consistency, demonstrating the effectiveness of consistency enforcement. Furthermore, the TVD results become closer within a lower privacy regime (larger budget). This is due to the improved accuracy of both CarEst and local MRFs with the increased privacy budget, reducing the advantage of the consistency enforcement procedure. Therefore, enforcing consistency becomes more necessary in a higher privacy regime.

8 RELATED WORK

We review related work from the following three perspectives.

DP data synthesis. There have been plenty of approaches [4, 16, 29, 30, 32, 39, 46] to generate synthetic data with DP guarantee, which can be categorized into GAN-based [2, 5, 7, 14, 24, 64], game-based [15, 18, 53], and marginal-based approaches [6, 38, 39, 63, 66]. Among them, the marginal-based ones tend to perform best, aiming to approximate the joint distribution of high-dimensional data with multiple low-way marginals. Such an approximation can help to circumvent the curse of dimensionality, i.e., the exponentially exploded sizes of the contingency histogram with the increased attribute number. For example, PrivBayes [63] utilizes the Bayesian network to select low-way marginals to approximate a high-dimensional distribution. PrivMRF [6] applies a Markov Random Field to model the data distribution, which enables flexible selection of low-way marginals. Without learning a graph structure, PrivSyn [66] greedily searches numerous low-way marginals to represent and synthesize the original dataset directly. Despite high utility with DP guarantee, these approaches cannot be directly extended to the vertical federated setting.

Private vertical data synthesis. There are several works [22, 23, 43, 44, 51] on the private data synthesis under vertical setting. Among those works, some are based on the privacy model of k-anonymity [50], which has been proven to be vulnerable to various privacy attacks [26, 58]. A few works [23, 43, 51] explore DP data synthesis under a vertical setting. For instance, [43] proposes a two-party DP data synthesis framework relying on a given taxonomy tree, which is designed for classification tasks. [51] utilizes a latent tree model to capture the correlations among cross-party attributions. Besides, the DP-WGAN approach is also adapted to the vertical setting [23] to generate synthetic data. To the best of our knowledge, we are the first work adapt the marginal-based approach to the vertical setting. The empirical results have demonstrated the superiority.

Vertical data analysis with DP. Apart from data synthesis, there are also several works on the DP computing [17, 20] and DP machine learning [9, 34, 61, 62] under vertical setting. In particular, the work [17] applies DP to protect the loads of hash table for achieving malicious-secure two-party private set intersection. Another work [62] enables each data party to build a local feature extractor to output DP-sanitized feature embedding for realizing vertical deep learning with DP. A recent paper [34] achieves DP vertical k-means by leveraging the inherent randomness of FM-sketch to protect the membership information of data points.

9 CONCLUSION

We have presented VertiMRF, a novel differentially private algorithm to generate synthetic data in the vertical setting. In particular, we applied DP FM-sketch to encode the local data of each party and estimate cross-party marginals. Based on the shared sketches and local MRFs constructed by local parties, the central server can build an MRF to represent global correlations without access to the raw data and violation of DP. Additionally, we also provided two techniques tailored for datasets with large attribute domain sizes. Finally, we empirically validated VertiMRF by conducting end-to-end comparisons and ablation studies.

REFERENCES

[1] 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*, pages 308–318, 2016.

[2] N. C. Abay, Y. Zhou, M. Kantarcioglu, B. Thuraisingham, and L. Sweeney. Privacy preserving synthetic data release using deep learning. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2018, Dublin, Ireland, September 10–14, 2018, Proceedings, Part I 18*, pages 510–526. Springer, 2019.

[3] A. Asuncion and D. Newman. Uci machine learning repository, 2007.

[4] S. Aydore, W. Brown, M. Kearns, K. Kenthapadi, L. Melis, A. Roth, and A. A. Siva. Differentially private query release through adaptive projection. In *International Conference on Machine Learning*, pages 457–467. PMLR, 2021.

[5] B. K. Beaulieu-Jones, Z. S. Wu, C. Williams, R. Lee, S. P. Bhavnani, J. B. Byrd, and C. S. Greene. Privacy-preserving generative deep neural networks support clinical data sharing. *Circulation: Cardiovascular Quality and Outcomes*, 12(7):e005122, 2019.

[6] K. Cai, X. Lei, J. Wei, and X. Xiao. Data synthesis via differentially private markov random fields. *Proceedings of the VLDB Endowment*, 14(11):2190–2202, 2021.

[7] D. Chen, T. Orekondy, and M. Fritz. Gs-wgan: A gradient-sanitized approach for learning differentially private generators. *Advances in Neural Information Processing Systems*, 33:12673–12684, 2020.

[8] H. Chen, K. Laine, and P. Rindal. Fast private set intersection from homomorphic encryption. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1243–1255, 2017.

[9] T. Chen, X. Jin, Y. Sun, and W. Yin. Vafl: a method of vertical asynchronous federated learning. *arXiv preprint arXiv:2007.06081*, 2020.

[10] W. Chen, G. Ma, T. Fan, Y. Kang, Q. Xu, and Q. Yang. Secureboost+: A high performance gradient boosting tree framework for large scale vertical federated learning. *arXiv preprint arXiv:2110.10927*, 2021.

[11] C. Dickens, J. Thaler, and D. Ting. Order-invariant cardinality estimators are differentially private. *Advances in Neural Information Processing Systems*, 35:15204–15216, 2022.

[12] C. Dong, L. Chen, and Z. Wen. When private set intersection meets big data: an efficient and scalable protocol. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 789–800, 2013.

[13] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pages 265–284. Springer, 2006.

[14] L. Frigerio, A. S. de Oliveira, L. Gomez, and P. Duverger. Differentially private generative adversarial networks for time series, continuous, and discrete open data. In *ICT Systems Security and Privacy Protection: 34th IFIP TC 11 International Conference, SEC 2019, Lisbon, Portugal, June 25-27, 2019, Proceedings 34*, pages 151–164. Springer, 2019.

[15] M. Gaboardi, E. J. G. Arias, J. Hsu, A. Roth, and Z. S. Wu. Dual query: Practical private query release for high dimensional data. In *International Conference on Machine Learning*, pages 1170–1178. PMLR, 2014.

[16] C. Ge, S. Mohapatra, X. He, and I. F. Ilyas. Kamino: Constraint-aware differentially private data synthesis. *arXiv preprint arXiv:2012.15713*, 2020.

[17] A. Groce, P. Rindal, and M. Rosulek. Cheaper private set intersection via differentially private leakage. *Proceedings on Privacy Enhancing Technologies*, 2019(3), 2019.

[18] M. Hardt, K. Ligett, and F. McSherry. A simple and practical algorithm for differentially private data release. *Advances in neural information processing systems*, 25, 2012.

[19] S. Hardy, W. Henecka, H. Ivey-Law, R. Nock, G. Patrini, G. Smith, and B. Thorne. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677*, 2017.

[20] Y. He, X. Tan, J. Ni, L. T. Yang, and X. Deng. Differentially private set intersection for asymmetrical id alignment. *IEEE Transactions on Information Forensics and Security*, 17:3479–3494, 2022.

[21] Y. Hu, D. Niu, J. Yang, and S. Zhou. FDML: A collaborative machine learning framework for distributed features. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 2232–2240, New York, NY, USA, 2019. Association for Computing Machinery.

[22] W. Jiang and C. Clifton. A secure distributed framework for achieving k-anonymity. *The VLDB journal*, 15:316–333, 2006.

[23] X. Jiang, Y. Zhang, X. Zhou, and J. Grossklags. Distributed gan-based privacy-preserving publication of vertically-partitioned data. *Proceedings on Privacy Enhancing Technologies*, 2:236–250, 2023.

[24] J. Jordon, J. Yoon, and M. Van Der Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International conference on learning representations*, 2018.

[25] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[26] D. Kifer. Attacks on privacy and definetti's theorem. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 127–138, 2009.

[27] V. Kolesnikov, R. Kumaresan, M. Rosulek, and N. Trieu. Efficient batched oblivious prf with applications to private set intersection. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 818–829, 2016.

[28] H. Krawczyk. Hmqv: A high-performance secure diffie-hellman protocol. In *Annual international cryptology conference*, pages 546–566. Springer, 2005.

[29] H. Li, L. Xiong, and X. Jiang. Differentially private synthesization of multi-dimensional data using copula functions. In *Advances in database technology: proceedings. International conference on extending database technology*, volume 2014, page 475. NIH Public Access, 2014.

[30] H. Li, L. Xiong, L. Zhang, and X. Jiang. Dpsynthesizer: differentially private data synthesizer for privacy preserving data sharing. In *Proceedings of the VLDB Endowment International Conference on Very Large Data Bases*, volume 7, page 1677. NIH Public Access, 2014.

[31] N. Li, M. Lyu, D. Su, and W. Yang. *Differential privacy: From theory to practice*. Springer, 2017.

[32] N. Li, Z. Zhang, and T. Wang. Dpsyn: Experiences in the nist differential privacy data synthesis challenges. *arXiv preprint arXiv:2106.12949*, 2021.

[33] Z. Li, B. Ding, C. Zhang, N. Li, and J. Zhou. Federated matrix factorization with privacy guarantee. *Proc. VLDB Endow.*, 15(4):900–913, dec 2021.

[34] Z. Li, T. Wang, and N. Li. Differentially private vertical federated clustering. *Proceedings of the VLDB Endowment*, 16(6):1277–1290, 2023.

[35] Y. Liu, Y. Kang, X. Zhang, L. Li, Y. Cheng, T. Chen, M. Hong, and Q. Yang. A communication efficient collaborative learning framework for distributed features. *arXiv preprint arXiv:1912.11187*, 2019.

[36] Y. Liu, Y. Liu, Z. Liu, Y. Liang, C. Meng, J. Zhang, and Y. Zheng. Federated forest. *IEEE Transactions on Big Data*, (01):1–1, 2020.

[37] K. G. Manton. National long-term care survey: 1982, 1984, 1989, 1994, 1999, and 2004. *Inter-university Consortium for Political and Social Research*, 2010.

[38] R. McKenna, G. Miklau, and D. Sheldon. Winning the nist contest: A scalable and general approach to differentially private synthetic data. *arXiv preprint arXiv:2108.04978*, 2021.

[39] R. McKenna, D. Sheldon, and G. Miklau. Graphical-model based estimation and inference for differential privacy. In *International Conference on Machine Learning*, pages 4435–4444. PMLR, 2019.

[40] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In A. Singh and J. Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282, USA, 20–22 Apr 2017. PMLR.

[41] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang. Learning differentially private recurrent language models. In *International Conference on Learning Representations*. OpenReview.net, 2018.

[42] I. Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pages 263–275. IEEE, 2017.

[43] N. Mohammed, D. Alhadidi, B. C. Fung, and M. Debbabi. Secure two-party differentially private data release for vertically partitioned data. *IEEE transactions on dependable and secure computing*, 11(1):59–71, 2013.

[44] N. Mohammed, B. C. Fung, and M. Debbabi. Anonymity meets game theory: secure data integration with malicious participants. *The VLDB Journal*, 20:567–588, 2011.

[45] S. L. Pardau. The california consumer privacy act: Towards a european-style privacy regime in the united states. *J. Tech. L. & Pol'y*, 23:68, 2018.

[46] X. Ren, C.-M. Yu, W. Yu, S. Yang, X. Yang, J. A. McCann, and S. Y. Philip. Lopub: high-dimensional crowdsourced data publication with local differential privacy. *IEEE Transactions on Information Forensics and Security*, 13(9):2151–2166, 2018.

[47] D. Ridgeway, M. F. Theofanos, T. W. Manley, and C. Task. Challenge design and lessons learned from the 2018 differential privacy challenges. Technical report, Technical Report NIST Technical Note 2151, National Institute of Standards ..., 2021.

[48] S. Ruggles, K. Genadek, G. Ronald, G. Josiah, and M. Sobek. Ipums usa: Version 6.0. Technical report, Minneapolis: University of Minnesota, 2015.

[49] A. Smith, S. Song, and A. Guha Thakurta. The flajolet-martin sketch itself preserves differential privacy: Private counting with minimal space. *Advances in Neural Information Processing Systems*, 33:19561–19572, 2020.

[50] L. Sweeney. k-anonymity: A model for protecting privacy. *International journal of uncertainty, fuzziness and knowledge-based systems*, 10(05):557–570, 2002.

[51] P. Tang, X. Cheng, S. Su, R. Chen, and H. Shao. Differentially private publication of vertically partitioned data. *IEEE transactions on dependable and secure computing*, 18(2):780–795, 2019.

[52] J. Vaidya and C. Clifton. Privacy-preserving decision trees over vertically partitioned data. In *IFIP Annual Conference on Data and Applications Security and Privacy*, pages 139–152. Springer, 2005.

[53] G. Vietri, G. Tian, M. Bun, T. Steinke, and S. Wu. New oracle-efficient algorithms for private synthetic data release. In *International Conference on Machine Learning*, pages 9765–9774. PMLR, 2020.

[54] P. Voigt and A. Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 10(3152676):10–5555, 2017.

[55] T. Wang, B. Ding, J. Zhou, C. Hong, Z. Huang, N. Li, and S. Jha. Answering multi-dimensional analytical queries under local differential privacy. In *Proceedings of the 2019 International Conference on Management of Data*, pages 159–176, 2019.

[56] WeBank. Webank use case. https://www.fedai.org/cases/a-case-of-traffic-violations-insurance-using-federated-learning/, 2022.

[57] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15:3454–3469, 2020.

[58] R. C.-W. Wong, A. W.-C. Fu, K. Wang, and J. Pei. Minimality attack in privacy preserving data publishing. In *Proceedings of the 33rd international conference on Very large data bases*, pages 543–554, 2007.

[59] N. Wu, F. Farokhi, D. Smith, and M. A. Kaafar. The value of collaboration in convex machine learning with differential privacy. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 304–317, New York, NY, USA, 2020. IEEE.

[60] Y. Wu, S. Cai, X. Xiao, G. Chen, and B. C. Ooi. Privacy preserving vertical federated learning for tree-based models. *Proceedings of the VLDB Endowment*, 13(11):2090–2103, 2020.

[61] Y. Wu, S. Cai, X. Xiao, G. Chen, and B. C. Ooi. Privacy preserving vertical federated learning for tree-based models. *arXiv preprint arXiv:2008.06170*, 2020.

[62] C. Xie, P.-Y. Chen, C. Zhang, and B. Li. Improving privacy-preserving vertical federated learning by efficient communication with admm. *arXiv preprint arXiv:2207.10226*, 2022.

[63] J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, and X. Xiao. Privbayes: Private data release via bayesian networks. *ACM Transactions on Database Systems (TODS)*, 42(4):1–41, 2017.

[64] X. Zhang, S. Ji, and T. Wang. Differentially private releasing via deep generative model (technical report). *arXiv preprint arXiv:1801.01594*, 2018.

[65] Y. Zhang and H. Zhu. Additively homomorphical encryption based deep neural network for asymmetrically collaborative machine learning. *arXiv preprint arXiv:2007.06849*, 2020.

[66] Z. Zhang, T. Wang, N. Li, J. Honorio, M. Backes, S. He, J. Chen, and Y. Zhang. {PrivSyn}: Differentially private data synthesis. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 929–946, 2021.

A APPENDIX

A.1 proof of Theorem 3

Proof. First of all, we should notice that GRR achieves unbounded DP [31] which considers the neighboring dataset by replacing one record. It has been shown that any algorithm satisfying ϵ unbounded DP also satisfies 2ϵ bounded DP, where bounded DP considers neighboring datasets obtained by adding or removing a single record. In this paper, we consider bounded DP for consistency. Thus, given ϵ', each perturbation in Equation (6) should satisfy $\frac{\epsilon'}{2}$-DP. The privacy guarantee of the FO-based LocEnc procedure can be obtained by applying the sequential composition of DP, resulting in an overall privacy guarantee of $d\epsilon'/2$, where d represents the number of attributes in each record. However, Lemma 1 demonstrates that RDP provides an alternative bound for the composition of multiple DP algorithms, that is $(4\frac{\epsilon'}{2}\sqrt{2d\log(1/\delta)}, \delta)$-DP, where $0 < \delta < 1$ and $\log(1/\delta) \geq n(\frac{\epsilon'}{2})^2$. To obtain the tighter bound, we take the minimum between the two bounds, as stated in the theorem. The variance bound can be directly obtained from proposition 10 of [55]. □

A.2 Proof of Theorem 4

Proof. Let D and D' be neighboring datasets satisfying $D\nabla D' = X_{id} = \left\{v_{id}^1, ..., v_{id}^d\right\}$, where id denotes the record- index of X_{id}, v_{id}^j is the corresponding attribute value of A^j. Let f be the sketch-based LocEnc algorithm which maps t hash keys and input dataset to t set of sketch tuples

$$\left\{\mathcal{M}^{(h)} \triangleq \left\{\mathcal{M}_j^{(h)} \triangleq \left(\alpha_{v_1^j}^{(h)}, ..., \alpha_{v_{u_j}^j}^{(h)}\right) | j \in [d]\right\} | h \in [t]\right\}.$$

where $\alpha_{v_i^j}^{(h)}$ denotes the sketch for A^j taking value v_i^j generated by the hash key ξ_h.

We first calculate the privacy cost when applying a hash key ξ_h to the overall input dataset and returning sketch tuples $\mathcal{M}^{(h)}$. $\mathcal{M}^{(h)}$ has d sketch tuples and $\sum_{i=1}^d u_i$ sketches in total. Since X_{id} can only take one value v_{id}^j of each attribute A^j, then there should also be one sketch $\alpha_{v_{id}^j}^{(h)}$ in $\mathcal{M}_j^{(h)}$ may be different for D and D'. Therefore, according to the definition of RDP, it holds that

$$\exp\left[(\lambda - 1)D_\lambda\left(f(D, \xi_h)|f(D', \xi_h)\right)\right] \tag{8}$$

$$= \sum_{\mathcal{M}^{(h)}} Pr[\mathcal{M}^{(h)}]^\lambda Pr'[\mathcal{M}^{(h)}]^{1-\lambda} \tag{9}$$

$$= \sum_{\alpha_{v_{id}^1}^{(h)}=0}^{\infty} \cdots \sum_{\alpha_{v_{id}^d}^{(h)}=0}^{\infty} \{[Pr[\alpha_{v_{id}^1}^{(h)}] \prod_{1<i\leq d} Pr[\alpha_{v_{id}^i}^{(h)}|\{\alpha_{v_{id}^t}^{(h)}, t < i\}]]^\lambda \cdot \tag{10}$$

$$[Pr'[\alpha_{v_{id}^1}^{(h)}] \prod_{1<i\leq d} Pr'[\alpha_{v_{id}^i}^{(h)}|\{\alpha_{v_{id}^t}^{(h)}, t < i\}]]^{1-\lambda}\} \cdot \tag{11}$$

$$\underbrace{\sum_{\mathcal{M}_\neg^{(h)}} [Pr[\mathcal{M}_\neg^{(h)}|\vec{\alpha}]]^\lambda [Pr'[\mathcal{M}_\neg^{(h)}|\vec{\alpha}]]^{1-\lambda}}_{=1} \tag{12}$$

where the second equality follows the joint distribution formula, $\vec{\alpha} \triangleq \{\alpha_{v_{id}^i}^{(h)}, 1 \leq i \leq d\}$ and $\mathcal{M}_\neg^{(h)}$ denotes other sketches in $\mathcal{M}^{(h)}$ besides $\vec{\alpha}$.

Now consider term $Pr\left[\alpha_{v_{id}^i}^{(h)} | \left\{\alpha_{v_{id}^t}^{(h)}, t < i\right\}\right]$, there are two cases:

- $\forall t, \alpha_{v_{id}^i}^{(h)} \neq \alpha_{v_{id}^t}^{(h)}$. In such case, since $\mathcal{H}_\xi$ map distinct elements to independent variables and the k_p phantom elements are independently sampled, then $\alpha_{v_{id}^i}^{(h)}$ is independent of $\alpha_{v_{id}^t}^{(h)}, \forall t$. That indicates
$$Pr\left[\alpha_{v_{id}^i}^{(h)} | \left\{\alpha_{v_{id}^t}^{(h)}, t < i\right\}\right] = Pr\left[\alpha_{v_{id}^i}^{(h)}\right].$$
- $\exists t, s.t., \alpha_{v_{id}^i}^{(h)} = \alpha_{v_{id}^t}^{(h)}$. In such case, it should hold that $\alpha_{v_{id}^i}^{(h)} \geq \mathcal{H}_{\xi_h}(id)$. That indicates
$$Pr\left[\alpha_{v_{id}^i}^{(h)} | \left\{\alpha_{v_{id}^t}^{(h)}, t < i\right\}\right] = Pr'\left[\alpha_{v_{id}^i}^{(h)} | \left\{\alpha_{v_{id}^t}^{(h)}, t < i\right\}\right].$$

The left side of above Equation is the probability that $\alpha_{v_{id}^i}^{(h)}$ is the maximal among all elements in the set of hashed record ids and sampled geometric random variables on D. Since we have known that $\mathcal{H}_{\xi_h}(id)$ is not or not the only one maximal element in the set, then we can just consider other hashed ids and sampled variables. The ids are same for D and D' and each of the variables are i.i.d sampled from the same distribution, which can easily derive the equality of above equation.

W.l.o.g., we assume there are s terms $\{Pr\left[\alpha^{(h)}_{v^j_{id}} | \left\{\alpha^{(h)}_{v^t_{id}}, t < j\right\}\right], (d - s + 1) \leq j \leq d\}$ satisfying the second case. Then, Equation (8) can be bounded by:

$$\exp\left[(\lambda - 1)D_\lambda\left(f(D, \xi_h)|f(D', \xi_h)\right)\right] \tag{13}$$

$$= \sum_{\alpha^{(h)}_{v^1_{id}}=0}^{\infty} \cdots \sum_{\alpha^{(h)}_{v^{d-s}_{id}}=0}^{\infty} \left[\prod_{i=1}^{d-s} Pr[\alpha^{(h)}_{v^i_{id}}]\right]^{\lambda} \cdot \left[\prod_{i=1}^{d-s} Pr'[\alpha^{(h)}_{v^i_{id}}]\right]^{1-\lambda} \tag{14}$$

$$= \prod_{i=1}^{d-s} \underbrace{\left\{\sum_{\alpha^{(h)}_{v^i_{id}}=0}^{\infty} \left[Pr[\alpha^{(h)}_{v^i_{id}}]\right]^{\lambda}\left[Pr'[\alpha^{(h)}_{v^i_{id}}]\right]^{1-\lambda}\right\}}_{term(i)} \tag{15}$$

Lemma 2 demonstrates a statistical bound of $\epsilon^{'}$ under DP framework. According to the definition of RDP and the translation with DP, it holds that $term(i) \leq \exp\left[(\lambda - 1)(2\lambda(\epsilon')^2)\right]$. Then we can derive that

$$\exp\left[(\lambda - 1)D_\lambda\left(f(D, \xi_h)|f(D', \xi_h)\right)\right] \tag{16}$$

$$\leq \exp\left[(\lambda - 1)(2(d - s)\lambda(\epsilon')^2)\right] \tag{17}$$

$$\leq \exp\left[(\lambda - 1)(2d\lambda(\epsilon')^2)\right] \tag{18}$$

So far, we have proved that applying one hash key to map the overall input data satisfies $(\lambda, 2d\lambda(\epsilon')^2)$-RDP in a single run. Next, according to the sequential composition theorem of RDP [42], LocEnc algorithm involving t runs of the FM sketch generation process should satisfy $(\lambda, 2td\lambda(\epsilon')^2)$-RDP, which can be further translated to $(4\epsilon\sqrt{td\log(1/\delta)}, \delta)$-DP, $\forall\delta < 1$ if setting $\alpha \geq 2$. □

A.3 Proof of Theorem 5

Our proof is based on a lemma that bounds the error of the cardinality of a multi-set estimated by DP FM sketching algorithm shown in Algorithm 1.

Lemma 3. *Let k_{FM} be the estimated cardinality by Algorithm 1 with inputs $\gamma, \epsilon, \delta, \beta \in (0, 1)$, using $t = \frac{100\sqrt{\log(1/\beta)}}{\gamma^2}$ repeats, then for each multi-set $\mathcal{X} \subset u$, it holds that*

$$\frac{|\mathcal{X}|}{1+\gamma} - O \leq k_{FM} \leq (1+\gamma) \cdot |\mathcal{X}| + C \tag{19}$$

with probability at least $1 - \beta$, where $C = O(\frac{\log^{1/2}(1/\delta)\log^{1/4}(1/\beta)}{\epsilon})$.

Proof. We first bound each cardinality $\hat{T}_M[\mathbf{v}]$ estimated by FM sketch. As shown in the Algorithm 4, we compute each $\hat{T}_M[\mathbf{v}]$ using the inclusion-exclusion principle and the mergable property of sketch, that is $|A \cap B| = \hat{n} - |\bar{A} \cup \bar{B}|$, where $\hat{n}$ denotes the noisy data number sanitized by adding a Laplacian noise $\hat{N}$. Combining with lemma 3, we can derive that

$$\hat{n} - (n - T_M[\mathbf{v}]) \cdot (1+\gamma) - C \leq \hat{T}_M[\mathbf{v}] \leq \hat{n} - \frac{n - T_M[\mathbf{v}]}{1+\gamma} + C \tag{20}$$

$$-\gamma n + (1+\gamma)T_M[\mathbf{v}] + \hat{N} - C \leq \hat{T}_M[\mathbf{v}] \leq \frac{\gamma}{1+\gamma}n + \frac{T_M[\mathbf{v}]}{1+\gamma} + \hat{N} + C \tag{21}$$

By subtracting $T_M[\mathbf{v}]$ for both sides of Equation 21, we can obtain that:

$$-\gamma n + \gamma T_M[\mathbf{v}] + \hat{N} - C \leq \hat{T}_M[\mathbf{v}] - T_M[\mathbf{v}] \leq \frac{\gamma}{1+\gamma}n - \frac{\gamma T_M[\mathbf{v}]}{1+\gamma} + \hat{N} + C \tag{22}$$

By taking the absolute value for both sides and dividing them by $T_M[\mathbf{v}]$, we can derive that:

$$\frac{|\hat{T}_M[\mathbf{v}] - T_M[\mathbf{v}]|}{T_M[\mathbf{v}]} \leq \max\{\frac{\gamma}{1+\gamma}(\frac{n}{T_M[\mathbf{v}]} - 1) + \frac{\hat{N} + C}{T_M[\mathbf{v}]}, \tag{23}$$

$$\gamma(\frac{n}{T_M[\mathbf{v}]} - 1) - \frac{\hat{N} - C}{T_M[\mathbf{v}]}\} \tag{24}$$

$$\leq \gamma(\frac{n}{T_M[\mathbf{v}]} - 1) + \frac{\hat{N} + C}{T_M[\mathbf{v}]} \tag{25}$$

According to Lemma 3, the above bound holds with probability $1 - \beta$, and $C = O(\frac{\log^{1/2}(1/\delta)\log^{1/4}(1/\beta)}{\epsilon})$. □

Algorithm 5 Dimension reduction

Input: Partitioned dataset $D = \{D_i, i \in M\}$, domain $([u_1] \times ... \times [u_d])$, binning number b, privacy budget ϵ', noisy data number $\hat{n}$.

Output: Estimated cross-party contingency histograms.

```
for each party $\mathcal{P}_i$ do
    for each attribute $A^j \in \mathcal{A}_i$ do
        $A^j \leftarrow \mathtt{Bin}(A^j, b)$.
        for $l \in [b]$ do
            $k = \lfloor \frac{u_j}{b} \rfloor$ if $l < b$, else $k = u_j - \lfloor \frac{u_j}{b} \rfloor \times (l - 1)$
            $\widetilde{n}^h_i = n^h_i + Lap(0, \frac{d}{\epsilon}), \forall 1 \leq h \leq k$
            $U_{(j,l)} = \left[U^1_{(j,l)}, \ldots, U^k_{(j,l)}\right] = \left[\frac{\widetilde{n}^1_i}{\sum_{h=1}^{k} \widetilde{n}^h_i}, \ldots, \frac{\widetilde{n}^k_i}{\sum_{h=1}^{k} \widetilde{n}^h_i}\right]$
        end for
        $\mathcal{M}^j \leftarrow \mathtt{LocEnc}(A^j, aux, \epsilon_1, \delta_1, \hat{n})$
    end for
    Sends $\left\{\mathcal{M}^j, \left(U_{(j,l)}, l \in [b]\right) | A^j \in \mathcal{A}_i\right\}$ to server.
end for
for each cross-party marginal $(A^i, ..., A^j)$ do
    $\hat{T}^{(high)} \leftarrow \mathbf{0}^{(u_i \times ... \times u_j)}$
    $\hat{T}^{(low)} \leftarrow \mathtt{CarEst}((A^i, \ldots, A^j), aux, \epsilon_1, \delta_1)$.
    for $(v_{(i)}, \ldots, v_{(j)}) \in [u_i] \times \ldots \times [u_j]$ do
        $(v'_{(i)}, \ldots, v'_{(j)}) = (\lceil \frac{v_{(i)}}{\lfloor \frac{u_i}{b} \rfloor} \rceil, \ldots, \lceil \frac{v_{(j)}}{\lfloor \frac{u_j}{b} \rfloor} \rceil)$.
        $(h'_{(i)}, \ldots, h'_{(j)}) = (v_{(i)} \bmod \lfloor \frac{u_i}{b} \rfloor, \ldots, v_{(j)} \bmod \lfloor \frac{u_j}{b} \rfloor)$.
        $\hat{T}^{(high)}\left[(v_{(i)}, \ldots, v_{(j)})\right] = \hat{T}^{(low)}\left[(v'_{(i)}, \ldots, v'_{(j)})\right] \cdot \prod_{\cdot=i}^{j} U^{h'_{(\cdot)}}_{(\cdot, v'_{(\cdot)})}$
    end for
end for
```

A.4 Algorithm of Dimension reduction and consistency enforcement

Here we give the pseudo-codes of our proposed dimension reduction (Algorithm 5) and consistency enforcement techniques described in Section 6.

Algorithm 6 Consistency enforcement

```
Input: Partitioned dataset $D = \{D_i, i \in M\}$, domain $([u_1] \times ... \times [u_d])$, cross-party marginal $(A^i, \dots, A^j)$ estimated by CarEst as $\hat{T}$, local MRF estimated $\{\widetilde{T}_{(A^i)}, \dots, \widetilde{T}_{(A^j)}\}$, noisy data number $\hat{n}$.
Output: Consistent contingency histogram of $(A^i, \dots, A^j)$.
$\hat{T} \leftarrow \mathtt{Norm}\left(\hat{T}\right)$, $\{\widetilde{T}_{(A^i)}, \dots, \widetilde{T}_{(A^j)}\} \leftarrow \{\mathtt{Norm}(\widetilde{T}_{(A^i)}), \dots, \mathtt{Norm}(\widetilde{T}_{(A^j)})\}$.
while not consistent do
    for each attribute $A^l \in \{A^i, \dots, A^j\}$ do
        $\bar{T}_{(A^l)} = \frac{\widetilde{T}_{(A^l)} + \hat{T}_{(A^l)}}{2}$, where $\hat{T}_{(A^l)} = \sum_{\neg A^l} \hat{T}$
        $\widetilde{T}_{(A^l)} \leftarrow \bar{T}_{(A^l)}$
        for $(v_{(i)}, \dots, v_{(j)}) \in [u_i] \times \dots \times [u_j]$ do
            $\hat{T}[(v_{(i)}, \dots, v_{(j)})] = \hat{T}[(v_{(i)}, \dots, v_{(j)})] + \frac{\bar{T}_{(A^l)}[v_{(l)}] - \hat{T}_{(A^l)}[v_{(l)}]}{(u_i \times ... \times u_j)/u_l}$
        end for
        Ensure $\sum \hat{T}[(v_{(i)}, \dots, v_{(j)})] = 1$ and $\forall \hat{T}[(v_{(i)}, \dots, v_{(j)})] \geq 0$
    end for
end while
return $\hat{n} \cdot \hat{T}$
```