Optimal Accuracy-Privacy Trade-off of Inference as Service

Yulu Jin and Lifeng Lai

***Abstract*—In this paper, we propose a general framework to provide a desirable trade-off between inference accuracy and privacy protection in the inference as service scenario (IAS). Instead of sending data directly to the server, the user will preprocess the data through a privacy-preserving mapping, which will increase privacy protection but reduce inference accuracy. To properly address the trade-off between privacy protection and inference accuracy, we formulate an optimization problem to find the optimal privacy-preserving mapping. Even though the problem is non-convex in general, we characterize nice structures of the problem and develop an iterative algorithm to find the desired privacy-preserving mapping, with convergence analysis provided under certain assumptions. From numerical examples, we observe that the proposed method has better performance than gradient ascent method in the convergence speed, solution quality and algorithm stability.**

I. Introduction

The Internet of Things (IoT) is an emerging communication paradigm that aims at connecting different kinds of devices to the Internet [2]–[4]. Within the past decade, the number of IoT devices being introduced in the market has increased dramatically due to its low cost and convenience [5]. Sensors of IoT devices could generate contexts at a high velocity and the inference with the contexts becomes an essential component for IoT applications [6]. However, building inference systems is costly due to the overhead of maintaining contexts repositories, running inference algorithms and learning from the inference results for further applications of inference tasks. One of the emerging solutions to this problem is so-called inference-as-a-service (IAS) [7], [8]. In IAS, the devices will send data to a server in the cloud, who will make inference using sophisticated algorithms. However, the IAS paradigm brings privacy issues, as the devices will send their data to the cloud without knowing where these data is stored or what future purposes these data might serve. There are some interesting works that attempt to address this issue using Homomorphic Encryption (HE) technique [9]–[11]. Unfortunately, the complexity of HE-based solution is very high, and its privacy relies on the (unproved) assumption that certain mathematical problems are difficult to solve.

The goal of our paper is to address the fundamental trade-off between inference accuracy and privacy protection from information theory perspective. Instead of sending data directly to the server, the user will preprocess the data through a privacy-preserving mapping. This privacy-preserving mapping has two opposing effects. On one hand, it will prevent the server from observing the data directly and hence enhance the privacy protection. On the other hand, this might reduce the inference accuracy. To properly address the trade-off between these two competing goals, we formulate an optimization problem to find the optimal privacy-preserving mapping. As the inference accuracy is directly related to the mutual information between parameters of interest and post-mapping data, we use mutual information to measure the inference accuracy. However, determining the privacy measure is tricky, as there are many existing information leakage measures [12], each of which is useful for certain specific scenarios. Hence, in our problem formulation, instead of using a specific privacy leakage measure, we propose a general framework that is applicable for different privacy metrics. The proposed framework is defined by a continuous function f with certain properties. Different choices of f lead to different privacy measures. For example, if f is chosen to be $-\log$ function, the proposed privacy leakage metric is the same as mutual information, a widely used information leakage measure. Moreover, we introduce a parameter β to represent the relative weight between these two measures. Thus, the trade-off problem between privacy and accuracy can be solved through a maximization problem where the objective function is composed of a weighted sum of accuracy and privacy terms.

To solve the maximization problem, if we optimize over the space of the privacy-preserving mapping directly, the formulated problem is a complicated non-convex problem with multiple constraints. Through various transformations and variable augmentations, we transform the optimization problem into a form that has three dominating arguments with certain nice concavity properties. In particular, if any two arguments are fixed, the problem is concave in the remaining argument. We then exploit this structure and design an algorithm with two nested loops to solve the optimization problem for general f by iterating between those three dominating arguments until reaching convergence. For the outer loop, we solve the optimization on the first dominating argument, for which we have a closed-form update formula. For the inner loop, using certain concavity properties of the objective function on the other two dominating arguments, we apply the Alternating Direction Method of Multipliers (ADMM) methods to solve the non-convex problem efficiently. Compared with solving the optimization problem using gradient ascent in the space of the privacy-preserving mapping directly, the

Y. Jin and L. Lai are with the Department of Electrical and Computer Engineering, University of California, Davis, CA. Email: {yuljin,lflai}@ucdavis.edu. The work of Y. Jin and L. Lai was supported by the National Science Foundation under Grants CCF-1717943, ECCS-1711468, CNS-1824553, CCF-1908258 and ECCS-2000415. This paper has been presented in part in the 2021 IEEE International Conference on Acoustics, Speech and Signal Processing [1].

proposed method does not need parameter tuning, converges much faster and finds solutions that have much better qualities. To further illustrate the proposed framework and algorithm, we also provide several examples by specializing f to particular function choices and provide numerical results.

Moreover, we provide the convergence analysis of the proposed method. Since there are two nested loops in the proposed method, we first prove the convergence of the inner loop, which is the convergence proof of the ADMM process. Although there are many existing convergence proofs for typical ADMM, most of them focus on separable problems only. In our case, the considered optimization problem has non-separable structure. Inspired by recent research works about convergence analysis of ADMM with non-separable objective functions [13]–[15], we provide two proofs with different assumptions on f. Based on the convergence proof of the ADMM procedure, we further prove that the function value is non-decreasing between two iterations of the outer-loop. Then with a guarantee that the objective function is upper-bounded, the proposed algorithm is shown to converge.

There exist many other privacy-preserving techniques that are based on perturbations of data, which provide privacy guarantees at the expense of a loss of accuracy [16]–[20]. k-anonymity is proposed by Samarati and Sweeney [16], which requires that each record is indistinguishable from at least k–1 other records within the dataset. Differential privacy works by adding a pre-determined amount of randomness into a computation performed on a data set [17]. For example, a local randomization approach is proposed in [20] to solve the privacy concern in distributed machine learning whose privacy-preserving property is measured by local differential privacy, and ADMM is used as a parallel computing approach. These concepts and techniques are very useful for the privacy protection of data analysis through a dataset or database, which is different from the setup considered in this paper. Moreover, various minimax formulations and algorithms have also been proposed to defend against inference attacks in different scenarios [21]–[23]. Bertran et al. [21] proposed an optimization problem where the terms in the objective function were defined in terms of mutual information, showed the performance bound for the optimization problem and learned the sanitization transform in a data-driven fashion using an adversarial approach with Deep Neural Networks (DNNs). Under their formulation, they analyzed a trade-off between utility loss and attribute obfuscation under the constraint of the attribute obfuscation $I(A;Z) \leq k$. Feutry et al. [22] measured the utility and privacy by expected risks, formulated the utility-privacy trade-off as a min-diff-max optimization problem and proposed a learning-based and task-dependent approach to solving this problem, while only deterministic mechanisms are considered. To address this issue, a privacy-preserving adversarial network was proposed in [23] by employing adversarially-trained neural networks to implement randomized mechanisms and to perform a variational approximation of mutual information privacy. Different from them, we propose a more general framework of privacy protection and avoid the reliance on DNNs to derive the privacy-preserving mapping.

This work has been partially presented in [1]. Compared with [1], the algorithms in this paper are significantly improved with theoretical convergence guarantee. In particular, while the algorithm in our conference paper [1] converges in various numerical examples, there is no convergence proof. It is in fact difficult to provide theoretical convergence guarantee for the algorithm in [1]. In this work, by designing an iterative algorithm with two nested loops involving ADMM procedure, we can provide convergence guarantee both theoretically and numerically.

The remainder of the paper is organized as follows. In Section II, we introduce the problem formulation. In Section III, we present the proposed algorithm and provide the convergence analysis. In Section IV, we present numerical results. Finally, we offer concluding remarks in Section V.

II. Problem formulation

Consider an inference problem, in which one would like to infer the parameter $S \in \mathcal{S}$ of data $Y \in \mathcal{Y}$, in which $\mathcal{Y}$ has a finite alphabet. In the inference as service scenario, one would send Y to the server who will determine the parameter S using its sophisticated models and powerful computing capabilities. However, directly sending data Y to the server brings the privacy issue, as now the server knows Y perfectly. To reduce the privacy leakage, instead of sending Y directly, one can employ a privacy-preserving mapping to transform data Y to $U \in \mathcal{U}$ and send U to the server. Here, $\mathcal{U}$ also has a finite alphabet and is allowed to be different from $\mathcal{Y}$. Without loss of generality, we will employ a randomized privacy-preserving mapping and use $p(u|y)$ to denote the probability that data $Y = y$ will be mapped to $U = u$ and the whole mapping is denoted as $P_{U|Y}$. Furthermore, we use P_S to denote the prior distribution of S and $P_{Y|S}$ to denote the conditional distribution Y given S, while the lower-case letter p is used to denote the component-wise probability (e.g., $p(s), p(y), p(y|s)$ will be used in the sequel).

To measure the inference accuracy, note that the distributional difference between P_S and $P_{S|U}$ characterizes the information about S contained in U. Since the inference at the server side is solely based on U, such information determines the inference accuracy. As $I(S;U)$ is the averaged Kullback–Leibler (KL) divergence between P_S and $P_{S|U}$, we use it to measure the inference accuracy. We would like to make $I(S;U)$ as large as possible, which means that we would like to retain as much information about the parameter of interest S in U as possible so that the server can make a more accurate inference.

To measure the privacy leakage, instead of choosing one particular privacy metric, we intend to investigate a general form $\mathbb{E}_{Y,U}[d(y,u)]$ that is applicable for different privacy metrics. Here, $d(y,u) = f(\frac{p(y)}{p(y|u)})$ and f is a continuous function defined on $(0, +\infty)$. We note that $\mathbb{E}_{Y,U}[d(y,u)] = \mathbb{E}_{Y,U}[f(\frac{p(y)}{p(y|u)})]$ measures the distributional distance between P_Y and $P_{Y|U}$, where P_Y is the prior distribution of Y and

$P_{Y|U}$ is the posterior distribution of Y after observing U. Hence, the smaller the distance, the less information U can provide about Y and the better the privacy protection. Note that $\frac{p(y)}{p(y|u)} = \frac{p(u)}{p(u|y)}$. Hence we will also use $\frac{p(u)}{p(u|y)}$ as the argument to f in the sequel. Since $p(u|y)$ shows in the denominator, we assume that $\epsilon \leq p(u|y) \leq 1, \forall y, u$, where $\epsilon > 0$.

To balance the inference accuracy and privacy protection, we propose to find the optimal privacy-preserving mapping $P_{U|Y}$ by solving the following optimization problem

$$\max_{P_{U|Y}} \quad \mathcal{F}[P_{U|Y}] \triangleq I(S;U) - \beta \mathbb{E}_{Y,U}\left[f\left(\frac{p(y)}{p(y|u)}\right)\right], \quad (1)$$

$$\text{s.t.} \quad p(u|y) \geq \epsilon, \forall y, u,$$

$$\sum_u p(u|y) = 1, \forall y. \quad (2)$$

Here, $\beta \in (0, \infty)$ is a trade-off parameter that indicates the relative importance of maximizing $I(S;U)$ (i.e., maximizing inference accuracy) and minimizing the distance $\mathbb{E}_{Y,U}[d(y,u)]$ between P_Y and $P_{Y|U}$ (i.e., maximizing the privacy).

For the privacy measure function f, we assume that

(a) $f(\cdot)$ is a strictly convex function;

(b) $f^{'}(t)$ is l_f- Lipschitz continuous of t.

Here we provide some comments about these assumptions. (a) guarantees certain convexity of the problem. In particular, under (a), the sub-problems are shown to be convex, which ensures the feasibility and simplification of the proposed method. (b) is needed to ensure the convergence of the proposed method. These assumptions are fairly weak. As will be discussed in Section IV, most of the widely used distance measures satisfy these assumptions.

The proposed framework in (1) is very general. Different choices of f will lead to different privacy measures. For example, if we choose f to be $-\log(\cdot)$, then we have

$$\mathbb{E}_{Y,U}[d(y,u)] = -\sum_{y,u} p(y)p(u|y)\log\left(\frac{p(u)}{p(u|y)}\right)$$
$$= \sum_y p(y) D_{KL}[P_{U|y} \parallel P_U] = I[U;Y],$$

in which $D_{KL}(\cdot \parallel \cdot)$ is the KL divergence. As the result, choosing f to be the $-\log$ function means we will use mutual information between U and Y to measure information leakage, a very common choice in information theory study. More examples will be provided in Section IV.

III. Algorithms and Convergence Proof

In this section, we discuss how to solve the optimization problem defined in (1) for general f. As the objective function is a complicated non-convex function of $P_{U|Y}$, we only expect to find a local maximal point. One natural approach to solving (1) is to apply the gradient ascent (GA) algorithm. However, GA faces several challenges such as proper step size, computation complexity, convergence speed and the quality of the local optimal point found etc. To overcome these challenges, we propose a new algorithm that transforms the maximization over single argument to an alternative maximization problem over multiple arguments and then employs ideas from ADMM to solve the transformed problem.

A. Algorithm

We first have the following lemma that are useful for transforming the objective function.

Lemma 1:

$$I(S;U) = I(S;Y) - \sum_{u,y} p(y)p(u|y) D_{KL}[P_{S|y} \parallel P_{S|u}].$$

Proof: Please refer to Appendix A. ∎

By Lemma 1, the objective function defined in (1) can be written as

$$\mathcal{F}[P_{U|Y}, P_U, P_{S|U}] = I(S;Y) - \beta \mathbb{E}_{Y,U}[d(y,u)] - \sum_{u,y} p(y)p(u|y) D_{KL}[P_{S|y} \parallel P_{S|u}].$$

Note that $I(S;Y)$, $p(y)$ and $p(s|y)$ are fixed, hence the cost function can be viewed as a function of three arguments $P_{U|Y}$, P_U and $P_{S|U}$. For consistency, we require the following equations to be satisfied simultaneously

$$p(u) = \sum_y p(u|y)p(y), \forall u, \quad (3)$$

$$p(s|u) = \frac{\sum_y p(u|y)p(s,y)}{p(u)}, \forall u, \forall s. \quad (4)$$

By (4), we further require that $p(u) > 0, \forall u$. As the result, we can reformulate (1) as the following alternative optimization problem

$$\max_{P_{S|U}} \max_{P_U} \max_{P_{U|Y}} \quad \mathcal{F}[P_{U|Y}, P_U, P_{S|U}]. \quad (5)$$

$$\text{s.t.} \quad p(u|y) \geq \epsilon, \forall y, \forall u, \ \sum_u p(u|y) = 1, \forall y,$$

$$p(u) > 0, \forall u, \ \sum_u p(u) = 1,$$

$$p(u) = \sum_y p(u|y)p(y), \forall u,$$

$$p(s|u) \geq 0, \forall u, \forall s, \ \sum_s p(s|u) = 1, \forall u,$$

$$p(s|u) = \frac{\sum_y p(u|y)p(s,y)}{p(u)}, \forall u, \forall s.$$

The following lemma illustrates the nice property of the alternative formulation (5): the alternative optimization problem is convex in each argument given the other two arguments.

Lemma 2: Suppose that $f(\cdot)$ is a strictly convex function. Then for given $P_U, P_{S|U}$, $\mathcal{F}[P_{U|Y}, P_U, P_{S|U}]$ is concave in each $P_{U|y_i}, \forall y_i \in \mathcal{Y}$. Similarly, for given $P_{U|Y}, P_{S|U}$, $\mathcal{F}[P_{U|Y}, P_U, P_{S|U}]$ is concave in P_U. For given $P_{U|Y}, P_U$, $\mathcal{F}[P_{U|Y}, P_U, P_{S|U}]$ is concave in $P_{S|U}$.

Proof: Please refer to Appendix B. ∎

Using this lemma, a natural approach to solving (5) is to alternately iterate between $P_{U|Y}$, P_U and $P_{S|U}$ until reaching convergence. In particular, for a given P_U and $P_{U|Y}$, we

first update $P_{S|U}$ by solving the maximization on $P_{S|U}$ and derive an analytical result as a function of P_U and $P_{U|Y}$. Then, for the derived $P_{S|U}$, we update P_U and $P_{U|Y}$ by solving the maximization on P_U and $P_{U|Y}$. We iterate this process until a convergence condition is satisfied. Because of the convexity property in Lemma 2, each subproblem can be solved efficiently. In the following, we provide details for each iteration. The convergence proof of the proposed algorithm will be presented in Section III-B.

1) Updating $P_{S|U}$: For the $P_{S|U}$ subproblem, the maximization problem is

$$\begin{aligned} \max_{P_{S|U}} \quad & \mathcal{F}[P_{S|U}|P_{U|Y}, P_U], \\ \text{s.t.} \quad & p(s|u) \geq 0, \forall u, \forall s, && (6) \\ & \sum_s p(s|u) = 1, \forall u, && (7) \\ & p(s|u) = \frac{\sum_y p(u|y)p(s,y)}{p(u)}, \forall u, \forall s. && (8) \end{aligned}$$

We first ignore (6), (8) and solve the optimization problem subject to (7) only. Since P_U and $P_{U|Y}$ are given, it is a convex optimization problem and the solution can be easily derived as

$$p(s|u) = \frac{\sum_y p(u|y)p(s,y)}{p(u)} \geq 0, \tag{9}$$

which not only guarantees the non-negativity condition in (6), but also satisfies the constraint in (8) exactly and preserves the consistency of different arguments.

2) Updating $P_{U|Y}$ and P_U: Now, for a given $P_{S|U}$, we discuss how to update $P_{U|Y}$ and P_U by solving

$$\begin{aligned} \max_{P_{U|Y}} \max_{P_U} \quad & \mathcal{F}[P_{U|Y}, P_U|P_{S|U}], && (10) \\ \text{s.t.} \quad & p(u|y) \geq \epsilon, \forall y, \forall u, \ \sum_u p(u|y) = 1, \forall y, && (11) \\ & p(u) > 0, \forall u, \ \sum_u p(u) = 1, && (12) \\ & \delta(u) = p(u) - \sum_y p(u|y)p(y) = 0, \forall u, && (13) \end{aligned}$$

where (13) corresponds to the consistency requirement (3). Moreover, note that each row in the matrix $P_{U|Y}$ is independent and we further show that the objective function in (10) can be written as the sum of $|\mathcal{Y}|$ terms, each of which depends only on one row of $P_{U|Y}$.

$$\begin{aligned} \mathcal{F}[P_{U|Y}, P_U|P_{S|U}] &= -\beta \sum_{i=1}^{|\mathcal{Y}|} \left[p(y_i) \sum_u p(u|y_i) d\left(\frac{p(u)}{p(u|y_i)}\right) \right] \\ &\quad - \sum_{i=1}^{|\mathcal{Y}|} \left[p(y_i) \sum_u p(u|y_i) D_{KL}[P_{S|y_i} \parallel P_{S|u}] \right] + I(S;Y) \\ &= \sum_{i=1}^{|\mathcal{Y}|} \mathcal{F}_i' \left[P_{U|Y}, P_U|P_{S|U}\right] + I(S;Y), \end{aligned} \tag{14}$$

where

$$\begin{aligned} \mathcal{F}_i' \left[P_{U|Y}, P_U|P_{S|U}\right] = p(y_i) \Bigg[& -\beta \sum_u p(u|y_i) f\left(\frac{p(u)}{p(u|y_i)}\right) \\ & - \sum_u p(u|y_i) D_{KL}[P_{S|y_i} \parallel P_{S|u}] \Bigg]. \end{aligned} \tag{15}$$

Thus, the optimization on $P_{U|Y}$ can be divided into $|\mathcal{Y}|$-problems, each of which corresponds to one row in $P_{U|Y}$.

As the result, although (10) is a non-convex problem in $(P_{U|Y}, P_U)$ jonitly, it is a convex problem of one argument given the others, as shown in Lemma 2. This motivates us to apply the ADMM approach to solve the problem.

The augmented Lagrangian for the above problem is

$$\begin{aligned} & \mathcal{L}[P_{U|Y}, P_U, P_{S|U}; \Lambda] \\ & = \mathcal{F}[P_{U|Y}, P_U|P_{S|U}] + \sum_u \lambda(u)\delta(u) - \frac{\rho}{2} \sum_u \delta^2(u), \end{aligned} \tag{16}$$

where Λ is a vector of size $|\mathcal{U}|$ and each component is denoted as $\lambda(u)$. Since $P_{S|U}$ is given, we will omit it from the expression of $\mathcal{L}$.

In the ADMM approach, there are updates of $P_{U|Y}$, P_U and Λ respectively. Exploiting the structure in (14), we can solve (10) using the following iterative procedure

$$P_{U|y_i}^{t+1} = \arg\max_{P_{U|y_i}} \mathcal{L}[P_{U|y_i}, P_{U|Y^{(i-)}}^{t+1}, P_{U|Y^{(i+)}}^{t}, P_U^t; \Lambda^t], \tag{17}$$

$$P_U^{t+1} = \arg\max_{P_U} \mathcal{L}[P_{U|Y}^{t+1}, P_U; \Lambda^t], \tag{18}$$

$$\Lambda^{t+1} = \Lambda^t + \rho(P_U^{t+1} - (P_{U|Y}^{t+1})^T P_Y), \tag{19}$$

$$\begin{aligned} \text{or} \quad \lambda^{t+1}(u) &= \lambda^t(u) + \rho[p^{t+1}(u) - \sum_y p^{t+1}(u|y)p(y)] \\ &= \lambda^t(u) + \rho\delta^{t+1}(u), \end{aligned}$$

where $P_{U|Y^{(i-)}}$ denotes all rows before the i-th row in the matrix $P_{U|Y}$ and $P_{U|Y^{(i+)}}$ denotes all rows after the i-th row.

For $P_{U|y_i}$, the optimization problem is

$$\begin{aligned} \max_{P_{U|y_i}} \quad & \mathcal{L}[P_{U|y_i}, P_{U|Y^{(i-)}}^{t+1}, P_{U|Y^{(i+)}}^{t}, P_U^t; \Lambda^t], && (20) \\ \text{s.t.} \quad & p(u|y_i) \geq \epsilon, \forall u, \sum_u p(u|y_i) = 1. \end{aligned}$$

We have the following lemma regarding the objective function in (20). The proof follows similar steps as that of the proof of Lemma 2, and hence is omitted.

Lemma 3: The objective function in (20) is a strictly concave function.

Hence, each sub-problem is a convex optimization problem with $|\mathcal{U}|$ inequality constraints, and one equality constraint. In practice, under a specified $f(\cdot)$, the sub-problem can be solved numerically.

The sub-problem with respect to P_U is

$$\begin{aligned} \max_{P_U} \quad & \mathcal{L}[P_{U|Y}^{t+1}, P_U; \Lambda^t], && (21) \\ \text{subject to} \quad & p(u) > 0, \forall u, \ \sum_u p(u) = 1. \end{aligned}$$

Following similar steps of Lemma 2, we can prove the following lemma.

Lemma 4: The objective function in (21) is a strictly concave function.

Although there are two constraints in this sub-problem, we ignore them first and in the convergence proof we will show that for the optimal solution point, these two constraints are naturally satisfied. We represent the solution to the unconstrained problem as $P_U^{t+1} = \arg\max_{P_U} \mathcal{L}[P_{U|Y}^{t+1}, P_U; \Lambda^t]$.

After solving two sub-problems on $P_{U|Y}$ and P_U respectively, we update the value of Λ.

In summary, we employ two nested loops to find the optimal privacy-preserving mapping. In the outer loop, we update $P_{S|U}$ by (9). In the inner loop, we update $P_{U|Y}$ and P_U for a given $P_{S|U}$ by methods of ADMM, i.e. going through the process of (17), (18), (19). We will use (j) to denote the j-th outer iteration and use $(j), t$ to denote the arguments at the t-th inner iteration of the j-th outer iteration. The algorithm is summarized in Algorithm 1. To quantify the matrix differences, we use the Frobenius norm [24], where for an $m \times n$ matrix $\boldsymbol{A}$, $\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{i,j}|^2}$. To quantify the vector differences, we use the ℓ_1 norm, where for vector $\boldsymbol{b} = (b_1, b_2, \cdots, b_n)$, $\|\boldsymbol{b}\|_{\ell_1} = \sum_{i=1}^n |b_i|$.

Algorithm 1 Design the optimal privacy-preserving mapping

Input:

Prior distribution P_S and conditional distribution $P_{Y|S}$.

Trade-off parameter β.

Converge parameter η, η_p, η_d.

Output:

A mapping $P_{U|Y}$ from $Y \in \mathcal{Y}$ to $U \in \mathcal{U}$.

Initialization:

Randomly initiate $P_{U|Y}$ and calculate $P_U, P_{S|U}$ by (3) and (4).

$j = 1$.
while $\left\| P_{S|U}^{(j)} - P_{S|U}^{(j-1)} \right\|_F > \eta$ **do**
  $P_U^{(j),1} = P_U^{(j-1)}$.
  $P_{U|Y}^{(j),1} = P_{U|Y}^{(j-1)}$.
  $t = 1$.
  while $t = 1$ or $\left\| P_U^{(j),t} - P_U^{(j),t-1} \right\|_{\ell_1} > \eta_p$ **do**
    Update $P_{U|y_i}$ by solving (20).
    Update P_U by solving (21).
    Update Λ by (19).
    $t = t + 1$.
  Update $P_{S|U}^{(i)}$ by (9).
  $j = j + 1$.
return $P_{U|Y}$

B. Convergence Analysis

In this section, we provide the convergence proof for Algorithm 1. In order to guarantee the existence of the solution to the optimization problem, the object function is required to be upper-bounded. The following lemma shows that the objective function is upper-bounded.

Lemma 5: For a continuous function $f(\cdot)$, $\mathcal{F}[P_{U|Y}, P_U, P_{S|U}]$ is bounded from above.

Proof: Please refer to Appendix C. ■

According to Algorithm 1, there are two loops in the proposed method. To show the convergence of the algorithm, we look at the outer loop first. Since we already have that the objective function $\mathcal{F}$ is upper-bounded, as long as we prove that the value of $\mathcal{F}$ is non-decreasing between two iterations of the outer loop, the algorithm will converge to a local optima. There are two steps in the outer loop, updating $P_{S|U}$ by (9) and updating $P_{U|Y}$ and P_U by applying ADMM. For the update of $P_{S|U}$, since the optimization with respect to $P_{S|U}$ is a convex optimization problem and has a closed-form solution as the update function, the objective function $\mathcal{F}$ is non-decreasing in this step. Then we have to show that the function value is non-decreasing after the ADMM procedure. In order to derive this result, we will prove a stronger result. In particular, in the following we prove that the ADMM procedure converges to the optimal point of the maximization problem on $P_{U|Y}$ and P_U defined in (10).

To prove the convergence of the proposed ADMM procedure, we first note that the typical ADMM convergence proofs focus on separable problems [25]. When there is a non-separable structure, the convergence of ADMM is not guaranteed and has to be handled differently. In our case, since the problem has a non-separable structure, we need to develop a new proof.

To make the presentation clear, in the following, we show the convergence in the case of $|\mathcal{Y}| = 2$ and the proof can be easily generalized to the case when $\mathcal{Y}$ has a finite alphabet. For $|\mathcal{Y}| = 2$, the optimization problem can be rewritten as

$$
\begin{aligned}
\max \quad & -\Bigg[p(y_1) \sum_u p\left(u|y_1\right) D_{KL}[P_{S|y_1} \parallel P_{S|u}] \\
& + p(y_2) \sum_u p\left(u|y_2\right) D_{KL}[P_{S|y_2} \parallel P_{S|u}]\Bigg] \\
& - \beta \sum_u \left[p\left(u|y_1\right) p(y_1) f\left(\frac{p(u)}{p(u|y_1)}\right) \right. \\
& \left. + p(u|y_2) p(y_2) f\left(\frac{p(u)}{p(u|y_2)}\right)\right], \\
\text{s. t} \quad & p(u|y_i) \geq \epsilon, \forall u, \sum_u p(u|y_i) = 1, i = 1, 2, \\
& p(u) > 0, \forall u, \sum_u p(u) = 1, \\
& -p(u|y_1)p(y_1) - p(u|y_2)p(y_2) + p(u) = 0, \forall u,
\end{aligned}
$$

in which the last constraint can also be written in the vector form $-p(y_1)P_{U|y_1} - p(y_2)P_{U|y_2} + P_U = \mathbf{0}$.

For presentation convenience, we denote

$$h_i(P_{U|y_i}) = -p(y_i)\sum_u p(u|y_i) D_{KL}[P_{S|y_i} \parallel P_{S|u}], \quad i = 1, 2,$$

$$g(P_{U|y_1}, P_{U|y_2}, P_U) = -\beta \sum_u \left[p(u|y_1)p(y_1) f\left(\frac{p(u)}{p(u|y_1)}\right) + p(u|y_2)p(y_2) f\left(\frac{p(u)}{p(u|y_2)}\right)\right].$$

Thus, the objective function is

$$h_1(P_{U|y_1}) + h_2(P_{U|y_2}) + g(P_{U|y_1}, P_{U|y_2}, P_U).$$

Recall that the augmented Lagrangian is

$$\begin{aligned} & \mathcal{L}[P_{U|Y}, P_U, P_{S|U}; \Lambda] \\ = \;& \mathcal{F}[P_{U|Y}, P_U | P_{S|U}] + \sum_u \lambda(u)\delta(u) - \frac{\rho}{2}\sum_u \delta(u)^2 \\ = \;& h_1(P_{U|y_1}) + h_2(P_{U|y_2}) + g(P_{U|y_1}, P_{U|y_2}, P_U) \\ & + \sum_u \lambda(u)\delta(u) - \frac{\rho}{2}\sum_u \delta(u)^2. \end{aligned}$$

To prove the convergence of ADMM, we first provide some useful results.

Lemma 6:

$$\begin{aligned} & \mathcal{L}[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^{t+1}] \geq \mathcal{L}[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^t] \\ \geq \;& \mathcal{L}[P^{t+1}_{U|Y}, P^t_U; \Lambda^t] \geq \mathcal{L}[P^t_{U|Y}, P^t_U, \Lambda^t]. \end{aligned}$$

Proof: Please refer to Appendix D. ∎

Lemma 7: $\mathcal{L}[P_{U|Y}, P_U; \Lambda]$ is upper-bounded.

Proof: Please refer to Appendix E. ∎

Lemma 8:

$$\begin{aligned} \left\|\Lambda^{t+1} - \Lambda^t\right\|_{\ell_1} \leq l_g \Bigg(& \left\|P^{t+1}_{U|y_1} - P^t_{U|y_1}\right\|_{\ell_1} + \left\|P^{t+1}_{U|y_2} \right. \\ & \left. - P^t_{U|y_2}\right\|_{\ell_1} + \left\|P^{t+1}_U - P^t_U\right\|_{\ell_1}\Bigg), \end{aligned} \tag{22}$$

with $l_g = \frac{2\beta l_f}{\epsilon^2}$.

Proof: Please refer to Appendix F. ∎

Using these lemmas, we now prove that $\mathcal{L}$ converges with constraints (11), (12) and (13) satisfied. In the ADMM procedure, we have sub-problems on $P_{U|Y}, P_U$ and the update of multiplier Λ. In the sequel, we will analyze each sub-problem respectively and show the convergence in the last step.

1) $P_{U|Y}$ update: We consider the optimization on $P_{U|y_1}$ and $P_{U|y_2}$ separately. Recall that

$$P^{t+1}_{U|y_i} = \arg\max_{P_{U|y_i}} \mathcal{L}[P_{U|y_i}, P^{t+1}_{U|Y^{(i-)}}, P^t_{U|Y^{(i+)}}, P^t_U; \Lambda^t].$$

Then there exists non-negative $l^t_{p_1}$ and $l^t_{p_2}$ such that

$$\begin{aligned} & \left|\mathcal{L}\left[P^{t+1}_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right] - \mathcal{L}\left[P^t_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right]\right| \\ \geq \;& \left|l^t_{p_1}\right| \cdot \left\|P^{t+1}_{U|y_1} - P^t_{U|y_1}\right\|_{\ell_1}, \end{aligned} \tag{23}$$

and

$$\begin{aligned} & \left|\mathcal{L}\left[P^{t+1}_{U|y_1}, P^{t+1}_{U|y_2}, P^t_U; \Lambda^t\right] - \mathcal{L}\left[P^{t+1}_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right]\right| \\ \geq \;& \left|l^t_{p_2}\right| \cdot \left\|P^{t+1}_{U|y_2} - P^t_{U|y_2}\right\|_{\ell_1}. \end{aligned} \tag{24}$$

Note that when

$$\mathcal{L}\left[P^{t+1}_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right] - \mathcal{L}\left[P^t_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right] > 0,$$

we have $l^t_{p_1} > 0$.

When

$$\mathcal{L}\left[P^{t+1}_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right] - \mathcal{L}\left[P^t_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right] = 0,$$

we have

$$\left\|P^{t+1}_{U|y_1} - P^t_{U|y_1}\right\|_{\ell_1} = 0$$

since $\mathcal{L}\left[P_{U|y_1}, P^t_{U|y_2}, P^t_U; \Lambda^t\right]$ is strictly concave in $P_{U|y_1}$. Thus, $l^t_{p_1}$ can be an arbitrary value and we take $l^t_{p_1} = 1$.

Hence, we have $l^t_{p_1} > 0, \forall t$ and similarly, $l^t_{p_2} > 0, \forall t$.

2) P_U update: Recall that

$$P^{t+1}_U = \arg\max_{P_U} \mathcal{L}[P^{t+1}_{U|Y}, P_U; \Lambda^t].$$

Then there exists non-negative l^t_u such that

$$\begin{aligned} & \left|\mathcal{L}\left[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^t\right] - \mathcal{L}\left[P^{t+1}_{U|Y}, P^t_U; \Lambda^t\right]\right| \\ \geq \;& \left|l^t_u\right| \cdot \left\|P^{t+1}_U - P^t_U\right\|_{\ell_1}. \end{aligned} \tag{25}$$

Note that when

$$\mathcal{L}\left[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^t\right] - \mathcal{L}\left[P^{t+1}_{U|Y}, P^t_U; \Lambda^t\right] > 0,$$

we have $l^t_u > 0$. On the other hand, when

$$\mathcal{L}\left[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^t\right] - \mathcal{L}\left[P^{t+1}_{U|Y}, P^t_U; \Lambda^t\right] = 0,$$

we have

$$\left\|P^{t+1}_U - P^t_U\right\|_{\ell_1} = 0$$

since $\mathcal{L}\left[P^{t+1}_{U|y_1}, P^{t+1}_{U|y_2}, P_U; \Lambda^t\right]$ is strictly concave in P_U. Then l^t_u can be an arbitrary value and we take $l^t_u = 1$.

Hence, by combining these two cases, we have $l^t_u > 0, \forall t$.

3) Λ update. We have

$$\begin{aligned} & \mathcal{L}\left[P^{t+1}_{U|y_1}, P^{t+1}_{U|y_2}, P^{t+1}_U; \Lambda^{t+1}\right] \\ -\mathcal{L}\left[P^{t+1}_{U|y_1}, P^{t+1}_{U|y_2}, P^{t+1}_U; \Lambda^t\right] = & \frac{1}{\rho}\left\|\Lambda^{t+1} - \Lambda^t\right\|^2_{\ell_1} \geq 0. \end{aligned} \tag{26}$$

With these supporting results, we now analyze the convergence of the ADMM method. By Lemmas 6 and 7, $\mathcal{L}[P_{U|Y}, P_U; \Lambda]$ is non-decreasing and upper-bounded and thus

will converge to a local optima after some iterations. Thus, there exists t_0, such that

$$\begin{aligned}
\infty \quad > \quad & \sum_{t=t_0}^{\infty} \Big| \mathcal{L}\left[P_{U|y_1}^t, P_{U|y_2}^t, P_U^t; \Lambda^t\right] \\
& \qquad -\mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^{t+1}, P_U^{t+1}; \Lambda^{t+1}\right] \Big| \\
\overset{(a)}{=} \quad & \sum_{t=t_0}^{\infty} \Big\{ \mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^t, P_U^t; \Lambda^t\right] \\
& -\mathcal{L}\left[P_{U|y_1}^t, P_{U|y_2}^t, P_U^t; \Lambda^t\right] \\
& +\mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^{t+1}, P_U^t; \Lambda^t\right] \\
& -\mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^t, P_U^t; \Lambda^t\right] \\
& +\mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^{t+1}, P_U^{t+1}; \Lambda^t\right] \\
& -\mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^{t+1}, P_U^t; \Lambda^t\right] \\
& +\mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^{t+1}, P_U^{t+1}; \Lambda^{t+1}\right] \\
& -\mathcal{L}\left[P_{U|y_1}^{t+1}, P_{U|y_2}^{t+1}, P_U^{t+1}; \Lambda^t\right] \Big\} \\
\overset{(b)}{\geq} \quad & \min_{t_0 \leq t \leq \infty} \{l_{p_1}^t\} \sum_{t=t_0}^{\infty} \left\| P_{U|y_1}^{t+1} - P_{U|y_1}^t \right\|_{\ell_1} \\
& + \min_{t_0 \leq t \leq \infty} \{l_{p_2}^t\} \sum_{t=t_0}^{\infty} \left\| P_{U|y_2}^{t+1} - P_{U|y_2}^t \right\|_{\ell_1} \\
& + \min_{t_0 \leq t \leq \infty} \{l_u^t\} \sum_{t=t_0}^{\infty} \left\| P_U^{t+1} - P_U^t \right\|_{\ell_1},
\end{aligned}$$

where (a) is from Lemma 6 and (b) is from (23), (24), (25) and (26). Since $l_{p_1}^t, l_{p_2}^t, l_u^t > 0$, as $t \to \infty$, we have $\left\| P_{U|y_1}^{t+1} - P_{U|y_1}^t \right\|_{\ell_1} \to 0$, $\left\| P_{U|y_2}^{t+1} - P_{U|y_2}^t \right\|_{\ell_1} \to 0$, and $\left\| P_U^{t+1} - P_U^t \right\|_{\ell_1} \to 0$. By Lemma 8, we have $\left\| \Lambda^{t+1} - \Lambda^t \right\|_{\ell_1} \to 0$, which implies

$$P_U^{t+1} - p(y_1) P_{U|y_1}^{t+1} - p(y_2) P_{U|y_2}^{t+1} \to 0. \tag{27}$$

Then we check all the constraints in the optimization problem on $P_{U|Y}, P_U$ defined in (10).

- (11) is satisfied in solving the sub-problem of $P(U|y_i)$;
- (13) is satisfied as shown in (27);
- (12) is established once (11) and (13) are satisfied. As $\delta(u) \to 0$, we have $p(u) = \sum_y p(u|y)p(y) > 0$ based on the fact that $p(u|y) \geq \epsilon > 0, \forall y, \forall u$. Besides, as $\delta(u) \to 0$, we also have

$$\begin{aligned}
\sum_u p(u) &= \sum_u [\delta(u) + \sum_y p(u|y)p(y)] \\
&= \sum_u \sum_y p(u|y)p(y) \\
&= \sum_y p(y) \sum_u p(u|y) = \sum_y p(y) = 1.
\end{aligned}$$

Therefore, the ADMM procedure is shown to converge with all constraints satisfied, which indicates that we have arrived at a stationary point. As a result, the value of $\mathcal{F}$ is non-decreasing after the ADMM procedure and thus is also non-decreasing between two iterations of the outer loop, which shows that the algorithm will converge to a local optima.

C. Stronger Convergence for f with More Assumptions

In Section III-B, for the convergence analysis of ADMM, the value of $l_{p_1}^t, l_{p_2}^t, l_u^t$ are not specified and vary from t. In this subsection, we propose another ADMM procedure with Bregman distance and make stronger assumptions on f to provide a stronger convergence analysis.

First we introduce the definition of Bregman distance. Let $\phi : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable and strictly convex function. Denote $\nabla\phi(y)$ as the gradient of ϕ on y. Then the Bregman distance induced by ϕ is defined as

$$\Delta_\phi(x, y) = \phi(x) - \phi(y) - < \nabla\phi(y), x - y >, \tag{28}$$

where ϕ is called the kernel function or distance-generating function. From the property of Bregman distance, we have that $\Delta_\phi(x, y)$ is convex in x for fixed y [26]. The Bregman distance plays an important role in iterative algorithms. In particular, Bregman divergences are used to replace the quadratic penalty term in the standard ADMM (see $\delta^2(u)$ in (16)). Then we can choose a suitable Bregman divergence so that the sub-problems can be solved more efficiently [26].

To solve the optimization problem in (10), for notation simplicity, we denote $x_1 : P_{U|y_1}$, $x_2 : P_{U|y_2}$, and $v : P_U$.

Recall the definition of $h_1(\cdot), h_2(\cdot), g(\cdot)$ in Section III-B. We propose an algorithm starting with $\left(x_1^0, x_2^0, v^0\right)$ and Λ^0. Suppose that $\varphi_1, \varphi_2, \phi$ are differentiable and strictly convex functions. Then with the given iteration point $w^k = \left(x_1^k, x_2^k, v^k, \Lambda^k\right)$, the new iteration point $w^{k+1} = \left(x_1^{k+1}, x_2^{k+1}, v^{k+1}, \Lambda^{k+1}\right)$ is given as:

$$\begin{aligned}
x_1^{k+1} &= \arg\max \Big\{ h_1(x_1) + \left(x_1 - x_1^k\right)^T \nabla_{x_1} g\left(x_1^k, x_2^k, v^k\right) \\
&\quad - \frac{\rho}{2} \left\| p(y_1)x_1 + p(y_2)x_2^k - v^k - \frac{\Lambda^k}{\rho} \right\|^2 - \Delta_{\varphi_1}\left(x_1, x_1^k\right) \Big\}, \\
x_2^{k+1} &= \arg\max \Big\{ h_2(x_2) + \left(x_2 - x_2^k\right)^T \nabla_{x_2} g\left(x_1^k, x_2^k, v^k\right) \\
&\quad - \frac{\rho}{2} \left\| p(y_1)x_1^{k+1} + p(y_2)x_2 - v^k - \frac{\Lambda^k}{\rho} \right\|^2 - \Delta_{\varphi_2}\left(x_2, x_2^k\right) \Big\}, \\
v^{k+1} &= \arg\max \Big\{ g\left(x_1^{k+1}, x_2^{k+1}, v\right) \\
&\quad - \frac{\rho}{2} \left\| p(y_1)x_1^{k+1} + p(y_2)x_2^{k+1} - v - \frac{\Lambda^k}{\rho} \right\|^2 - \Delta_\phi\left(v, v^k\right) \Big\}, \\
\Lambda^{k+1} &= \Lambda^k + \rho\left(-p(y_1)x_1^{k+1} - p(y_2)x_2^{k+1} + v^{k+1}\right),
\end{aligned} \tag{29}$$

where $\Delta_{\varphi_1}\left(x_1, x_1^k\right), \Delta_{\varphi_2}\left(x_2, x_2^k\right)$, and $\Delta_\phi\left(v, v^k\right)$ are the Bregman distances associated with φ_1, φ_2, and ϕ respectively. Here, φ_1, φ_2, and ϕ should be properly chosen with respect to different $f(\cdot)$ adopted in the privacy measure. Moreover, in this subsection, $\|\cdot\|$ is used to denote the Euclidean norm.

To guarantee that the algorithm converges, we assume that

1) ∇g is l_g-Lipschitz continuous;

2) $\nabla\varphi_1, \nabla\varphi_2, \nabla\phi$ are Lipshitz continuous with the modulus $l_{\varphi_1}, l_{\varphi_2}, l_\phi$, respectively;
3) $\varphi_1, \varphi_2, \phi$ are strongly convex with the modulus $\delta_{\varphi_1}, \delta_{\varphi_2}, \delta_\phi$, and $\delta_{\varphi_1}, \delta_{\phi_2} > l_g$.

Then we have

Lemma 9:

$$\begin{aligned}&\left\|\Lambda^{k+1}-\Lambda^k\right\|^2\\ &\le 3l_g^2\left(\left\|x_1^{k+1}-x_1^k\right\|^2+\left\|x_2^{k+1}-x_2^k\right\|^2\right)\\ &\quad+3\left(l_g^2+l_\phi^2\right)\left\|v^{k+1}-v^k\right\|^2+3l_\phi^2\left\|v^k-v^{k-1}\right\|^2,\end{aligned}\tag{30}$$

which indicates the relationship between the dual variable Λ and the primal variables.

Proof: Please refer to Appendix F. ∎

By considering the updates of 3 primal variables, we have

Lemma 10:

$$\begin{aligned}&\mathcal{L}\left(w^{k+1}\right)-\mathcal{L}\left(w^k\right)\ge\frac{\delta_{\varphi_1}-l_g}{2}\left\|x_1^{k+1}-x_1^k\right\|^2\\ &\quad+\frac{\delta_{\varphi_2}-l_g}{2}\left\|x_2^{k+1}-x_2^k\right\|^2+\frac{\delta_\phi}{2}\left\|v^{k+1}-v^k\right\|^2.\end{aligned}$$

Proof: Please refer to Appendix H. ∎

By assumption 3), since $\delta_{\varphi_1}, \delta_{\phi_2} > l_g$, following similar analysis in Section III-B, we have that $\left\|x_1^{k+1}-x_1^k\right\|^2 \to 0$, $\left\|x_2^{k+1}-x_2^k\right\|^2 \to 0$, and $\left\|v^{k+1}-v^k\right\|^2 \to 0$. Then by Lemma 9, we have $\left\|\Lambda^{k+1}-\Lambda^k\right\|^2 \to 0$, and the ADMM procedure proposed in this subsection converges with specified modulus. Thus, when replacing the ADMM procedure in Section III-A with this ADMM procedure with Bregman distance, Algorithm 1 also converges.

IV. Examples and Numerical Results

In this section, we first give examples of different choices of f and then provide numerical results with specific f to show the performance of the proposed method.

A. Examples of f

We now provide examples of f, each of which leads to a well-known and widely used divergence measure.

In the first example, we consider $f(t) = -\log(t)$. As shown in Section II, if $f(t) = -\log(t)$, the privacy measure is then the mutual information. For the algorithm proposed in this chapter, we check whether all the assumptions are satisfied. Since $\epsilon \le p(u|y) \le 1$, we have $\epsilon \le \frac{p(u)}{p(u|y)} \le \frac{1}{\epsilon}$. Then we first have that $-\log(\cdot)$ is strictly convex on $\left[\epsilon, \frac{1}{\epsilon}\right]$. Secondly, we have that $f'(t) = -\frac{1}{t}$ is Lipschitz continuous since it is everywhere differentiable on $\left[\epsilon, \frac{1}{\epsilon}\right]$ and the absolute value of the derivative is bounded above by $\frac{1}{\epsilon^2}$.

In the second example, we consider the following strictly convex function

$$f(t) = t\log\frac{2t}{t+1} + \log\frac{2}{t+1}.$$

This choice leads to the Jensen-Shannon divergence [27]:

$$\mathbb{E}_{Y,U}[d(y,u)] = \sum_y p(y) JS[P_{U|y}, P_U],$$

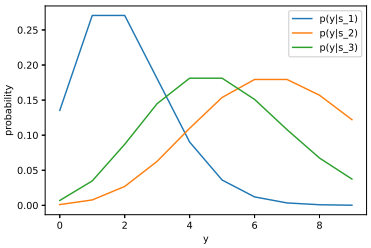

Fig. 1. Conditional distribution $p(y|s)$

in which

$$\begin{aligned}JS[P_{U|y}, P_U] = D_{KL}\left[P_{U|y} \parallel \frac{P_{U|y}+P_U}{2}\right]\\ +D_{KL}\left[P_U \parallel \frac{P_{U|y}+P_U}{2}\right].\end{aligned}$$

To check the assumption (b), we have

$$f'(t) = \log\frac{2t}{t+1}, f''(t) = \frac{1}{t(t+1)} \le \frac{1}{\epsilon(\epsilon+1)},$$

and thus it is Lipschitz continuous.

In the third example, consider the strictly convex function $f(t) = (1-t)^2/(2t+2)$, which leads to the Le Cam divergence [28] as the privacy measure,

$$\mathbb{E}_{Y,U}[d(y,u)] = \sum_y p(y) LC[P_{U|y} \parallel P_U],$$

in which

$$LC[P_{U|y} \parallel P_U] = \frac{1}{2}\sum_u \frac{[p(u|y)-p(u)]^2}{p(u|y)+p(u)}.$$

For this choice of f, again, the assumption (b) is satisfied.

In the fourth example, we consider the following function $f(t) = (1-\sqrt{t})^2$, which corresponds to the squared Hellinger distance [29]. It is easy to check that the assumptions are satisfied.

B. Numerical Results

In this subsection, we provide numerical examples to show that our methods converge much faster than GA, and the local maxima found by our methods has much better quality than the one found by GA. Moreover, we explore how the weight parameter β and the alphabet size of $\mathcal{U}$ affects the privacy protection.

In the first example, we set the prior distribution $P_S = \{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\}$ and let $|\mathcal{Y}| = 10, |\mathcal{U}| = 11$. The conditional distributions $P_{Y|S}$ under each s are shown in Fig. 1. Under this setup, we will perform both Algorithm 1 and GA to find the optimal transition mapping $P_{U|Y}$ that maximizes the functional defined in (1). Suppose that the trade-off parameter $\beta = 2$ and Jensen-Shannon divergence is used as the privacy metric. The initial mapping $P_{U|Y}$ is obtained by selecting random numbers conforming to uniform distribution and normalizing them.

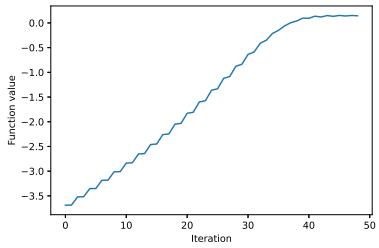

Fig. 2. Function value v.s. iteration (Algorithm 1, inner loop)

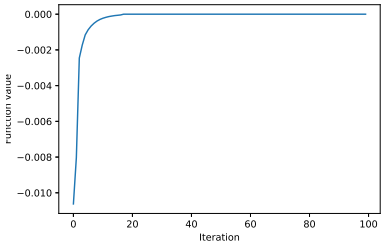

Fig. 3. Function value v.s. iteration (Algorithm 1, outer loop)

For the convergence speed, since there are two nested loops in Algorithm 1, we first explore the relationship between the function value $\mathcal{F}$ and the inner iteration, which is shown in Fig. 2, and then investigate the relationship between $\mathcal{F}$ and the outer iteration, which is illustrated in Fig 3. We notice that for the first outer iteration, by applying the ADMM method, the function value is increasing and converges during the inner loop. Moreover, for the outer loop, the function value is also increasing and converges as the iterative process progresses. For comparison purposes, we also plot the corresponding figure for GA in Fig. 4. From these figures, we can see that Algorithm 1 converges within 20 iterations. On the other hand, for gradient ascent algorithm, even for a pretty small step size 0.0001, the function value fails to keep increasing, which indicates that the step size is too large. Then for a smaller step size 0.00005, the function value converges as shown in Fig. 5. However, the value of the objective function found by GA is smaller than the value found by Algorithm 1.

For the relationship between β and the privacy protection, after random initialization, we run Algorithm 1 and

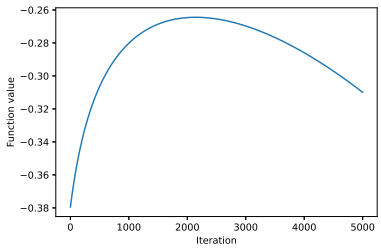

Fig. 4. Function value v.s. iteration (GA)

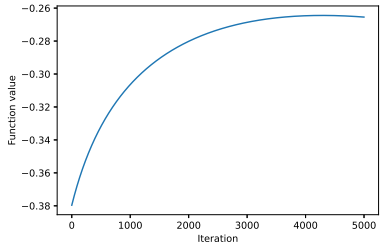

Fig. 5. Function value v.s. iteration (GA)

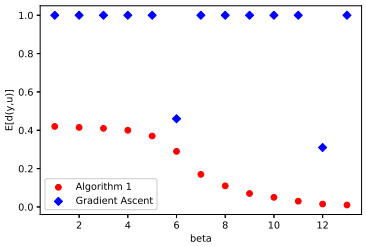

Fig. 6. β v.s. privacy protection (Algorithm 1 and GA)

GA until they terminate. The stopping criterion is either $||P_{U|Y}^{t+1} - P_{U|Y}^{t}||_F < 10^{-5}$ (convergence case) or a maximum number of iterations is reached (divergence case). We repeat this procedure 100 times for each β. Recall that the smaller the term $\mathbb{E}[d(y,u)]$, the better the privacy protection. In particular, we set $\mathbb{E}[d(y,u)]$ to be 1 for divergence cases since the maximum $\mathbb{E}[d(y,u)]$ under the converge scenario is smaller than 1. As shown in Fig. 6, we notice that $\mathbb{E}[d(y,u)]$ decreases as β increases for our proposed method while it is non-decreasing for GA. By setting the maximum number of iterations to be 3000, GA diverges under many choices of β. Even for the scenarios where GA converges, compared with Algorithm 1, the privacy protection obtained by GA is weaker. Therefore, the privacy-preserving mapping designed by GA could hardly guarantee the protection of privacy.

To explore other privacy measures, we now set f as $f(t) = (1-t)^2/(2t+2)$, which corresponds to the Le Cam divergence as discussed in Section IV-A. We again compare Algorithm 1 and GA. The results are shown in Table I. From the table, we can see that the maximum function value found by our method is greater than the one found by GA. Moreover, since the objective function is quite complex in $P_{U|Y}$, it is hard to find a proper step size for GA algorithm while there is no need to do so in our algorithm.

To compare different privacy measures, we set the trade-off parameter $\beta = 8$, which indicates that the privacy term is dominant in the objective function. As shown in Fig. 7, although the function values under JS-divergence and LC-divergence are different, the convergence speed and convergence curve are almost the same, which shows that the proposed algorithm can converge in a similar manner under different metrics. However, the optimal privacy-preserving mapping $P_{U|Y}$ found

Methods	Convergent value
Algorithm 1	-6.697e-14
Gradient ascent($\alpha = 0.05$)	-0.251
Gradient ascent($\alpha = 0.07$)	-0.245
Gradient ascent($\alpha = 0.1$)	-0.317
Gradient ascent($\alpha = 0.15$)	-0.235
Gradient ascent($\alpha = 0.2$)	Diverge

TABLE I
CONVERGENT VALUE OF ALGORITHM 1 AND GA

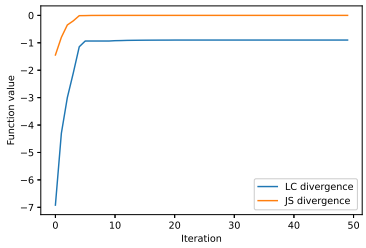

Fig. 7. Convergence process for JS and LC divergences (Algorithm 1)

by those two privacy measures are different. Therefore, in practical applications, an appropriate task-oriented privacy measure needs to be chosen.

Finally, we explore the relationship between $|\mathcal{U}|$ and the privacy protection. Note that in the proposed method, the alphabet sizes of $\mathcal{Y}$ and $\mathcal{U}$ are not necessarily equal. Thus, for $|\mathcal{Y}| = 10$, we explore how $|\mathcal{U}|$ affects the convergent function value. From Fig. 8, it is shown that although the function value is increasing as $|\mathcal{U}|$ increases, the alphabet size $|\mathcal{U}|$ has limited effects on the function value when $|\mathcal{U}| \geq 7$, which indicates that a large alphabet size of $\mathcal{U}$ is not necessary to derive a satisfactory privacy-preserving mapping. By setting $|\mathcal{Y}|$ to different values, we notice that when $\frac{|\mathcal{U}|}{|\mathcal{Y}|} \geq 0.8$, the convergent function value is relatively large.

V. CONCLUSION

We have proposed a general framework to design privacy-preserving mapping to achieve privacy-accuracy trade-off in the IAS scenarios. We have formulated optimization problems to find the optimal mapping. We have discussed the structure of the formulated problems and designed an iterative method to solve these complicated optimization problems. We have also proved the convergence of the proposed method under certain assumptions. Moreover, we have provided numerical results showing that this method has better performance than GA in the convergence speed, solution quality and algorithm stability.

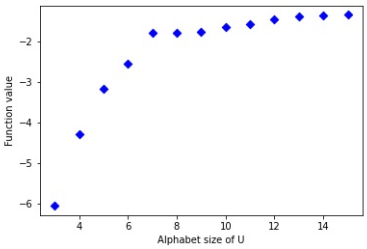

Fig. 8. Function value v.s. Alphabet size of $\mathcal{U}$ (Algorithm 1)

APPENDIX A
PROOF OF LEMMA 1

$$
\begin{aligned}
& I(S;U) + \sum_{u,y} p(y)p(u|y) D_{KL}[P_{S|y} \parallel P_{S|u}] \\
= & \sum_{s,u,y} p(s,u,y) \log \frac{p(s|u)}{p(s)} \\
& + \sum_{s,u,y} p(y)p(u|y)p(s|y) \log \frac{p(s|y)}{p(s|u)} \\
\overset{(a)}{=} & \sum_{s,u,y} p(s,u,y) \left[\log \frac{p(s|u)}{p(s)} + \log \frac{p(s|y)}{p(s|u)} \right] \\
= & \sum_{s,y} p(s,y) \log \frac{p(s|y)}{p(s)} = I(S;Y),
\end{aligned}
$$

where (a) uses the fact that $S \to Y \to U$ is a Markov chain since given Y, S and U are independent.

APPENDIX B
PROOF OF LEMMA 2

First, prove that $\mathcal{F}[P_{U|Y}]$ is concave with respect to $P_{S|U}$. By applying Lemma 1, (1) can be written in the following form,

$$
\begin{aligned}
\mathcal{F}[P_{U|Y}] \quad = \quad & I(S;Y) - \beta \mathbb{E}_{Y,U}[d(y,u)] \\
& - \sum_{u,y} p(y)p(u|y) D_{KL}[P_{S|y} \parallel P_{S|u}]. \qquad (31)
\end{aligned}
$$

Note that $I(S;Y)$ is a constant under our setup. Given $P_{U|Y}$ and P_U, $\mathbb{E}_{Y,U}[d(y,u)]$ is independent of $P_{S|U}$. Moreover, $P_{S|u}$ and $P_{S|u'}$ are two independent vectors. For given u and y, we have

$$
D_{KL}[P_{S|y} \parallel P_{S|u}] = \sum_{s} p(s|y) \log \frac{p(s|y)}{p(s|u)}. \qquad (32)
$$

Since $a \log(x)$ is concave in x, (32) is convex in $P_{S|u}$ and $\mathcal{F}[P_{U|Y}]$ is concave with respect to $P_{S|U}$.

Second, we prove that $\mathcal{F}[P_{U|Y}]$ is concave w.r.t P_U when f is strictly convex. Note that P_U only shows up in $\mathbb{E}_{Y,U}[d(y,u)]$ and since f is strictly convex, taking the sum doesn't change the concavity and $\mathcal{F}[P_{U|Y}]$ is also concave in P_U.

Third, we consider $P_{U|Y}$. There are $|\mathcal{Y}|$ conditional distributions in the mapping $P_{U|Y}$, where $P_{U|y}$ and $P_{U|y'}$ are independent when $y \neq y'$. Then we consider a particular row $P_{U|y}$ and prove the concavity. The Hessian matrix of $\mathcal{F}$ with respect to $P_{U|y}$ is

$$
\boldsymbol{H}_{\mathcal{F}} = \begin{bmatrix}
\frac{\partial^2 \mathcal{F}[p(u|y)]}{\partial p(u_1|y)^2} & \cdots & \frac{\partial^2 \mathcal{F}[P_{U|Y}]}{\partial p(u_1|y) \partial p(u_{|\mathcal{U}|}|y)} \\
\cdots & \cdots & \cdots \\
\frac{\partial^2 \mathcal{F}[p(u|y)]}{\partial p(u_{|\mathcal{U}|}|y) \partial p(u_1|y)} & \cdots & \frac{\partial^2 \mathcal{F}[p(u|y)]}{\partial p(u_{|\mathcal{U}|}|y)^2}
\end{bmatrix}.
$$

Then we calculate each element in $\boldsymbol{H}_{\mathcal{F}}$. Assume that $i \neq j$. Taking derivative based on the form in (31), we have

$$\frac{\partial^2 \mathcal{F}[P_{U|Y}]}{\partial p(u_i|y)^2} = -\beta \frac{\partial^2 \mathbb{E}_{Y,U}[d(y,u)]}{\partial p(u_i|y)^2},$$
$$\frac{\partial^2 \mathcal{F}[P_{U|Y}]}{\partial p(u_i|y)\partial p(u_j|y)} = -\beta \frac{\partial^2 \mathbb{E}_{Y,U}[d(y,u)]}{\partial p(u_i|y)\partial p(u_j|y)},$$

in which

$$\frac{\partial^2 \mathbb{E}_{Y,U}[d(y,u)]}{\partial p(u_i|y)^2} = p(y)\left[f'(t)\frac{-p(u_i)}{p(u_i|y)^2} - f'(t)\frac{-p(u_i)}{p(u_i|y)^2} - tf''(t)\frac{-p(u_i)}{p(u_i|y)^2}\right] = p(y)f''(t)\frac{t^2}{p(u_i|y)} > 0,$$
$$\frac{\partial^2 \mathbb{E}_{Y,U}[d(y,u)]}{\partial p(u_i|y)\partial p(u_j|y)} \overset{(a)}{=} 0,$$

where $t = \frac{p(u_i)}{p(u_i|y)}$ and (a) is due to the fact that t is independent of $p(u_j|y)$ when $i \neq j$ and P_U is given. Then we have $\frac{\partial^2 \mathcal{F}[P_{U|Y}]}{\partial p(u_i|y)^2} < 0$ and $\frac{\partial^2 \mathcal{F}[P_{U|Y}]}{\partial p(u_i|y)\partial p(u_j|y)} = 0$. Thus, the Hessian matrix $\boldsymbol{H}_{\mathcal{F}}$ is a diagonal matrix with negative entries, which indicates that the objective function $\mathcal{F}$ is concave in $P_{U|y_i}$ and the lemma is proved.

Appendix C
Proof of Lemma 5

First, note that $I(S;U) \leq H(S)$, which is bounded. Thus, $\mathcal{F}[P_{U|Y}]$ is upper bounded if $\mathbb{E}_{Y,U}[d(y,u)]$ is bounded from above. Let $t(y,u) = \frac{p(u)}{p(u|y)}$. We have that

$$\begin{aligned}\mathbb{E}_{Y,U}[d(y,u)] &= \sum_{y,u} p(y)p(u|y)f(t(y,u)) \\ &= \sum_{y,u} p(y)p(u)\frac{f(t(y,u))}{t(y,u)},\end{aligned}$$

where $p(y)p(u) \leq 1$. Since $\epsilon \leq p(u|y) \leq 1$, we have that $t(y,u) \in \left[\epsilon, \frac{1}{\epsilon}\right], \forall y,u$. Since f is continuous, it's natural to have $\frac{f(t(y,u))}{t(y,u)} < +\infty$. Then $\mathbb{E}_{Y,U}[d(y,u)]$ is bounded from above.

Appendix D
Proof of Lemma 6

By (17) and (18), we have

$$\mathcal{L}[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^t] \geq \mathcal{L}[P^{t+1}_{U|Y}, P^{t}_U; \Lambda^t] \geq \mathcal{L}[P^{t}_{U|Y}, P^{t}_U, \Lambda^t].$$

By (19) and the expression of $\mathcal{L}$, we have

$$\begin{aligned}&\mathcal{L}[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^{t+1}] \\ =\ &\mathcal{L}[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^{t}] + \sum_u (\lambda^{t+1}(u) - \lambda^t(u))\delta^{t+1}(u) \\ =\ &\mathcal{L}[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^{t}] + \sum_u \rho(\delta^{t+1}(u))^2 \qquad (33) \\ \geq\ &\mathcal{L}[P^{t+1}_{U|Y}, P^{t+1}_U; \Lambda^{t}],\end{aligned}$$

and the lemma is proved.

Appendix E
Proof of Lemma 7

Since $\sum_y p(u|y)p(y) \leq 1$, we have $|\delta(u)| \leq 2$ and thus $\sum_u \lambda(u)\delta(u) < \infty, \frac{\rho}{2}\sum_u \delta(u)^2 < \infty$ due to the fact that $\mathcal{U}$ is a finite set. Moreover, by Lemma 5, $\mathcal{F}[P_{U|Y}, P_U, P_{S|U}]$ is bounded from above for any $P_{S|U}$. Hence, for a particular $P_{S|U}$, we have

$$\begin{aligned}&\mathcal{L}[P_{U|Y}, P_U, P_{S|U}; \Lambda] \\ =\ &\mathcal{F}[P_{U|Y}, P_U|P_{S|U}] + \sum_u \lambda(u)\delta(u) - \frac{\rho}{2}\sum_u \delta(u)^2 < \infty.\end{aligned}$$

Appendix F
Proof of Lemma 8

By the optimality of P_U, we have

$$\begin{cases} 0 = \nabla_{P_U} g\left(P^{t+1}_{U|y_1}, P^{t+1}_{U|y_2}, P^{t+1}_U\right) \\ \qquad + \rho\left(-p(y_1)P^{t+1}_{U|y_1} - p(y_2)P^{t+1}_{U|y_2} + P^{t+1}_U\right) + \Lambda^t, \\ \Lambda^{t+1} = \Lambda^t + \rho\left(p(y_1)P^{t+1}_{U|y_1} - p(y_2)P^{t+1}_{U|y_2} + P^{t+1}_U\right), \end{cases}$$

which implies $0 = \nabla_{P_U} g\left(P^{t+1}_{U|y_1}, P^{t+1}_{U|y_2}, P^{t+1}_U\right) + \Lambda^{t+1}$. Then we have

$$\begin{aligned}&\left\|\Lambda^{t+1} - \Lambda^t\right\|_{\ell_1} \\ =\ &\sum_{u\in\mathcal{U}} \left|\frac{\partial g}{\partial p^{t+1}(u)} - \frac{\partial g}{\partial p^t(u)}\right|, \qquad (34)\end{aligned}$$

where

$$\begin{aligned}\frac{\partial g}{\partial p^t(u)} = -\beta\Bigg[&p(y_1)f'\left(\frac{p^t(u)}{p^t(u|y_1)}\right) \\ &+p(y_2)f'\left(\frac{p^t(u)}{p^t(u|y_2)}\right)\Bigg].\end{aligned}$$

Given that $f'(t)$ is l_f- Lipschitz continuous of t, we further have that for any given u

$$\begin{aligned}&\left|\frac{\partial g}{\partial p^{t+1}(u)} - \frac{\partial g}{\partial p^t(u)}\right| \\ =\ &\Bigg|-\beta p(y_1)\left[f'\left(\frac{p^t(u)}{p^t(u|y_1)}\right) - f'\left(\frac{p^{t+1}(u)}{p^{t+1}(u|y_1)}\right)\right] \\ &-\beta p(y_2)\left[f'\left(\frac{p^t(u)}{p^t(u|y_2)}\right) - f'(\frac{p^{t+1}(u)}{p^{t+1}(u|y_2)})\right]\Bigg| \\ \leq\ &\beta l_f\left[p(y_1)\left|\frac{p^t(u)}{p^t(u|y_1)} - \frac{p^{t+1}(u)}{p^{t+1}(u|y_1)}\right|\right. \\ &\left.+p(y_2)\left|\frac{p^t(u)}{p^t(u|y_2)} - \frac{p^{t+1}(u)}{p^{t+1}(u|y_2)}\right|\right], \qquad (35)\end{aligned}$$

where

$$\frac{p^t(u)}{p^t(u|y_1)} - \frac{p^{t+1}(u)}{p^{t+1}(u|y_1)} = \frac{p^t(u)p^{t+1}(u|y_1) - p^{t+1}(u)p^t(u|y_1)}{p^t(u|y_1)p^{t+1}(u|y_1)}.$$

Using the assumption that $\frac{1}{p(u|y)} \leq \frac{1}{\epsilon} < \infty$, we have

$$\begin{aligned}&\frac{p^t(u)}{p^t(u|y_1)} - \frac{p^{t+1}(u)}{p^{t+1}(u|y_1)} \\ \leq\ &\left(\frac{1}{\epsilon}\right)^2 \left|p^t(u)p^{t+1}(u|y_1) - p^{t+1}(u)p^t(u|y_1)\right|.\end{aligned}$$

To further bound $\left|p^t(u)p^{t+1}(u|y_1)-p^{t+1}(u)p^t(u|y_1)\right|$, we have

$$\begin{aligned}&\left|\frac{p^t(u)p^{t+1}(u|y_1)-p^{t+1}(u)p^t(u|y_1)}{p^t(u)-p^{t+1}(u)}\right|\\ =\;& p^{t+1}(u|y_1)+p^{t+1}(u)\frac{|p^{t+1}(u|y_1)-p^t(u|y_1)|}{|p^t(u)-p^{t+1}(u)|}\\ \leq\;& 1+\frac{|p^{t+1}(u|y_1)-p^t(u|y_1)|}{|p^t(u)-p^{t+1}(u)|},\end{aligned}$$

and

$$\begin{aligned}&\left|\frac{p^t(u)p^{t+1}(u|y_1)-p^{t+1}(u)p^t(u|y_1)}{p^{t+1}(u|y_1)-p^t(u|y_1)}\right|\\ =\;& p^t(u)+p^t(u|y_1)\frac{|p^t(u)-p^{t+1}(u)|}{|p^{t+1}(u|y_1)-p^t(u|y_1)|}\\ \leq\;& 1+\frac{|p^t(u)-p^{t+1}(u)|}{|p^{t+1}(u|y_1)-p^t(u|y_1)|}.\end{aligned}$$

Moreover,

$$\begin{aligned}\min\quad&\left\{\frac{|p^{t+1}(u|y_1)-p^t(u|y_1)|}{|p^t(u)-p^{t+1}(u)|},\right.\\ &\left.\frac{|p^t(u)-p^{t+1}(u)|}{|p^{t+1}(u|y_1)-p^t(u|y_1)|}\right\}\leq 1.\end{aligned}$$

Then we have

$$\begin{aligned}&|p^t(u)p^{t+1}(u|y_1)-p^{t+1}(u)p^t(u|y_1)|\\ \leq\;& 2|p^t(u)-p^{t+1}(u)|+2|p^{t+1}(u|y_1)-p^t(u|y_1)|,\end{aligned}$$

and thus

$$\begin{aligned}&\left|\frac{p^t(u)}{p^t(u|y_1)}-\frac{p^{t+1}(u)}{p^{t+1}(u|y_1)}\right|\\ \leq\;& \frac{2}{\epsilon^2}\left[|p^t(u)-p^{t+1}(u)|+|p^{t+1}(u|y_1)-p^t(u|y_1)|\right].\end{aligned}\tag{36}$$

Similarly, for $\left(\frac{p^t(u)}{p^t(u|y_2)}-\frac{p^{t+1}(u)}{p^{t+1}(u|y_2)}\right)$, we have

$$\begin{aligned}&\left|\frac{p^t(u)}{p^t(u|y_2)}-\frac{p^{t+1}(u)}{p^{t+1}(u|y_2)}\right|\\ \leq\;& \frac{2}{\epsilon^2}\left[|p^t(u)-p^{t+1}(u)|+|p^{t+1}(u|y_2)-p^t(u|y_2)|\right].\end{aligned}\tag{37}$$

Plugging (36) and (37) into (35) and (34), we have

$$\begin{aligned}&\left\|\Lambda^{t+1}-\Lambda^t\right\|_{\ell_1}\\ =\;&\sum_{u\in\mathcal{U}}\left|\frac{\partial g}{\partial p^{t+1}(u)}-\frac{\partial g}{\partial p^t(u)}\right|\\ \leq\;&\frac{2\beta l_f}{\epsilon^2}\sum_{u\in\mathcal{U}}\big[|p^t(u)-p^{t+1}(u)|\\ &\qquad+p(y_1)|p^{t+1}(u|y_1)-p^t(u|y_1)|\\ &\qquad+p(y_2)|p^{t+1}(u|y_2)-p^t(u|y_2)|\big]\\ \leq\;& l_g\left(\left\|P^{t+1}_{U|y_1}-P^t_{U|y_1}\right\|_{\ell_1}+\left\|P^{t+1}_{U|y_2}-P^t_{U|y_2}\right\|_{\ell_1}\right.\\ &\left.+\left\|P^{t+1}_U-P^t_U\right\|_{\ell_1}\right),\end{aligned}$$

where $l_g=\frac{2\beta l_f}{\epsilon^2}$.

Appendix G
Proof of Lemma 9

The optimality condition of v-subproblem yields

$$\begin{aligned}0=&\nabla_v g\left(x_1^{k+1},x_2^{k+1},v^{k+1}\right)-\Lambda^k+\rho\left(p(y_1)x_1^{k+1}\right.\\ &\left.+p(y_2)x_2^{k+1}-v^{k+1}\right)+\nabla\phi\left(v^{k+1}\right)-\nabla\phi\left(v^k\right).\end{aligned}$$

As $\Lambda^{k+1}=\Lambda^k+\rho\left(-p(y_1)x_1^{k+1}-p(y_2)x_2^{k+1}+v^{k+1}\right)$, we have

$$\Lambda^{k+1}=\nabla_v g\left(x_1^{k+1},x_2^{k+1},v^{k+1}\right)+\nabla\phi\left(v^{k+1}\right)-\nabla\phi\left(v^k\right).$$

Thus,

$$\begin{aligned}&\left\|\Lambda^{k+1}-\Lambda^k\right\|^2\\ \leq&\left(\left\|\nabla_v g\left(x_1^{k+1},x_2^{k+1},v^{k+1}\right)-\nabla_v g\left(x_1^k,x_2^k,v^k\right)\right\|\right.\\ &\left.+\left\|\nabla\phi\left(v^{k-1}\right)-\nabla\phi\left(v^k\right)\right\|+\left\|\nabla\phi\left(v^{k-1}\right)-\nabla\phi\left(v^k\right)\right\|\right)^2\\ \leq&3\left(\left\|\nabla_v g\left(x_1^{k+1},x_2^{k+1},v^{k+1}\right)-\nabla_v g\left(x_1^k,x_2^k,v^k\right)\right\|^2\right.\\ &\left.+\left\|\nabla\phi\left(v^{k-1}\right)-\nabla\phi\left(v^k\right)\right\|^2+\left\|\nabla\phi\left(v^{k-1}\right)-\nabla\phi\left(v^k\right)\right\|^2\right)\\ \leq&3l_g^2\left(\left\|x_1^{k+1}-x_1^k\right\|^2+\left\|x_2^{k+1}-x_2^k\right\|^2\right)\\ &+3\left(l_g^2+l_\phi^2\right)\left\|v^{k+1}-v^k\right\|^2+3l_\phi^2\left\|v^k-v^{k-1}\right\|^2.\end{aligned}$$

Appendix H
Proof of Lemma 10

From the update of x_1, we have

$$\begin{aligned}&h_1\left(x_1^{k+1}\right)+\left\langle x_1^{k+1}-x_1^k,\nabla_{x_1}g\left(u^k\right)\right\rangle\\ &\quad+\left\langle\Lambda^k,p(y_1)x_1^{k+1}+p(y_2)x_2^k-v^k\right\rangle\\ &\quad-\frac{\rho}{2}\left\|p(y_1)x_1^{k+1}+p(y_2)x_2^k-v^k\right\|^2-\Delta_{\varphi_1}\left(x_1^{k+1},x_1^k\right)\\ &\geq h_1\left(x_1^k\right)+\left\langle\Lambda^k,r_k\right\rangle-\frac{\rho}{2}\left\|r_k\right\|^2,\end{aligned}$$

where $r_k=p(y_1)x_1^k+p(y_2)x_2^k-v^k$.

From the update of x_2, we have

$$\begin{aligned}&h_2\left(x_2^{k+1}\right)+\left\langle x_2^{k+1}-x_2^k,\nabla_{x_2}g\left(u^k\right)\right\rangle\\ &\quad+\left\langle\Lambda^k,p(y_1)x_1^{k+1}+p(y_2)x_2^{k+1}-v^k\right\rangle\\ &\quad-\frac{\rho}{2}\left\|p(y_1)x_1^{k+1}+p(y_2)x_2^{k+1}-v^k\right\|^2-\Delta_{\varphi_2}\left(x_2^{k+1},x_2^k\right)\\ &\geq h_2\left(x_2^k\right)+\left\langle\Lambda^k,p(y_1)x_1^{k+1}+p(y_2)x_2^k-v^k\right\rangle\\ &\quad-\frac{\rho}{2}\left\|p(y_1)x_1^{k+1}+p(y_2)x_2^k-v^k\right\|^2.\end{aligned}$$

From the update of v, we have

$$\begin{aligned}&g\left(u^{k+1}\right)+\left\langle\Lambda^k,r_{k+1}\right\rangle-\frac{\rho}{2}\left\|r_{k+1}\right\|^2-\Delta_\phi\left(v^{k+1},v^k\right)\\ &\geq g\left(x_1^{k+1},x_2^{k+1},v^k\right)-\frac{\rho}{2}\left\|p(y_1)x_1^{k+1}+p(y_2)x_2^{k+1}-v^k\right\|^2\\ &\quad+\left\langle\Lambda^k,p(y_1)x_1^{k+1}+p(y_2)x_2^{k+1}-v^k\right\rangle.\end{aligned}$$

Adding up the above three inequalities, we have

$$
\begin{aligned}
&h_1\left(x_1^{k+1}\right)+h_2\left(x_2^{k+1}\right)+g\left(u^{k+1}\right)\\
&\qquad+\left\langle\Lambda^k, r_{k+1}\right\rangle-\frac{\rho}{2}\left\|r_{k+1}\right\|^2\\
\geq\ & h_1\left(x_1^k\right)+h_2\left(x_2^k\right)+g\left(x_1^{k+1}, x_2^{k+1}, v^k\right)+\left\langle\Lambda^k, r_k\right\rangle\\
&-\left[\left\langle x_1^{k+1}-x_1^k, \nabla_{x_1} g\left(u^k\right)\right\rangle+\left\langle x_2^{k+1}-x_2^k, \nabla_{x_2} g\left(u^k\right)\right\rangle\right]\\
&+\Delta_{\varphi_1}\left(x_1^{k+1}, x_1^k\right)+\Delta_{\varphi_2}\left(x_2^{k+1}, x_2^k\right)+\Delta_\phi\left(v^{k+1}, v^k\right)\\
&-\frac{\rho}{2}\left\|r_k\right\|^2\\
\geq\ & \mathcal{L}\left(w^k\right)+g\left(x_1^{k+1}, x_2^{k+1}, v^k\right)-g\left(u^k\right)\\
&-\left\langle\left(x_1^{k+1}-x_1^k, x_2^{k+1}-x_2^k, 0\right), \nabla g\left(u^k\right)\right\rangle\\
&+\Delta_{\varphi_1}\left(x_1^{k+1}, x_1^k\right)+\Delta_{\varphi_2}\left(x_2^{k+1}, x_2^k\right)+\Delta_\phi\left(v^{k+1}, v^k\right)\\
\overset{(a)}{\geq}\ & \mathcal{L}\left(w^k\right)-\frac{l_g}{2}\left[\left\|x_1^{k+1}-x_1^k\right\|^2+\left\|x_2^{k+1}-x_2^k\right\|^2\right]\\
&+\frac{\delta_{\varphi_1}}{2}\left\|x_1^{k+1}-x_1^k\right\|^2+\frac{\delta_{\varphi_2}}{2}\left\|x_2^{k+1}-x_2^k\right\|^2\\
&+\frac{\delta_\phi}{2}\left\|v^{k+1}-v^k\right\|^2,
\end{aligned}
$$

where (a) follows from the assumption 3) and the fact from [30] that if $h:\mathbb{R}^n\longrightarrow\mathbb{R}$ is a continuous differentiable function where gradient ∇h is Lipschitz continuous with the modulus $l_h>0$, then for any $x,y\in\mathbb{R}^n$, we have

$$
|h(y)-h(x)-<\nabla h(x), y-x>|\leq\frac{l_h}{2}\|y-x\|^2,
$$

and we apply this result on g here.

Thus, we have

$$
\begin{aligned}
&\mathcal{L}\left(w^{k+1}\right)-\mathcal{L}\left(w^k\right)\\
\geq\ & \frac{\delta_{\varphi_1}-l_g}{2}\left\|x_1^{k+1}-x_1^k\right\|^2\\
&+\frac{\delta_{\varphi_2}-l_g}{2}\left\|x_2^{k+1}-x_2^k\right\|^2+\frac{\delta_\phi}{2}\left\|v^{k+1}-v^k\right\|^2.
\end{aligned}
$$

REFERENCES

[1] Y. Jin and L. Lai, "Privacy-accuracy trade-off of inference as service," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, (Toronto, Canada), pp. 2645–2649, Jun. 2021.

[2] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "IoT: Internet of threats? A survey of practical security vulnerabilities in real IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8182–8201, May. 2019.

[3] A. Ahrabian, S. Kolozali, S. Enshaeifar, C. Cheong-Took, and P. Barnaghi, "Data analysis as a web service: A case study using IoT sensor data," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, (New Orleans, LA), pp. 6000–6004, Mar. 2017.

[4] M. Sun, W. P. Tay, and X. He, "Toward information privacy for the Internet of things: A nonparametric learning approach," *IEEE Transactions on Signal Processing*, vol. 66, no. 7, pp. 1734–1747, Jan. 2018.

[5] J. Wurm, K. Hoang, O. Arias, A. Sadeghi, and Y. Jin, "Security analysis on consumer and industrial IoT devices," in *Proc. Asia and South Pacific Design Automation Conference*, (Macao, China), pp. 519–524, Jan. 2016.

[6] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the Internet of things: A survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414–454, 2013.

[7] C. Zhang, M. Yu, W. Wang, and F. Yan, "Mark: Exploiting cloud services for cost-effective, slo-aware machine learning inference serving," in *Proc. Annual Technical Conference*, (Renton, WA), pp. 1049–1062, Jul. 2019.

[8] A. Gujarati, S. Elnikety, Y. He, K. McKinley, and B. Brandenburg, "Swayam: Distributed autoscaling to meet SLAs of machine learning inference services with resource efficiency," in *Proc. ACM/IFIP/USENIX Middleware Conference*, (Las Vegas, NV), pp. 109–120, Dec. 2017.

[9] M. Tebaa, S. El Hajji, and A. El Ghazi, "Homomorphic encryption applied to the cloud computing security," in *Proc. World Congress on Engineering*, vol. 1, (London, U.K), pp. 4–6, Jul. 2012.

[10] F. Boemer, A. Costache, R. Cammarota, and C. Wierzynski, "nGraph-HE2: A high-throughput framework for neural network inference on encrypted data," in *Proc. ACM Workshop on Encrypted Computing & Applied Homomorphic Cryptography*, (London, UK), pp. 45–56, Nov. 2019.

[11] C. Gentry and D. Boneh, *A fully homomorphic encryption scheme*, vol. 20. Stanford university, 2009.

[12] I. Issa, A. Wagner, and S. Kamath, "An operational approach to information leakage," *IEEE Transactions on Information Theory*, vol. 66, no. 3, pp. 1625–1657, Mar. 2019.

[13] Y. Wang, W. Yin, and J. Zeng, "Global convergence of ADMM in nonconvex nonsmooth optimization," *Journal of Scientific Computing*, vol. 78, no. 1, pp. 29–63, Feb. 2019.

[14] M. Chao, Z. Deng, and J. Jian, "Convergence of linear Bregman ADMM for nonconvex and nonsmooth problems with nonseparable structure," *Complexity*, Feb. 2020.

[15] C. Chen, M. Li, X. Liu, and Y. Ye, "Extended ADMM and BCD for nonseparable convex minimization models with quadratic coupling terms: convergence analysis and insights," *Mathematical Programming*, vol. 173, no. 1-2, pp. 37–77, Mar. 2019.

[16] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," 1998.

[17] C. Dwork, "Differential privacy: A survey of results," in *Proc. International Conference on Theory and Applications of Models of Computation*, (Xi'an, China), pp. 1–19, Apr. 2008.

[18] F. Calmon and N. Fawaz, "Privacy against statistical inference," in *Proc. Annual Allerton Conference on Communication, Control, and Computing*, (Monticello, IL), pp. 1401–1408, Oct. 2012.

[19] C. Glackin, G. Chollet, N. Dugan, N. Cannings, J. Wall, S. Tahir, I. G. Ray, and M. Rajarajan, "Privacy preserving encrypted phonetic search of speech data," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, (New Orleans, LA), pp. 6414–6418, Mar. 2017.

[20] X. Wang, H. Ishii, L. Du, P. Cheng, and J. Chen, "Privacy-preserving distributed machine learning via local randomization and ADMM perturbation," *IEEE Transactions on Signal Processing*, vol. 68, pp. 4226–4241, Jul. 2020.

[21] B. Martin, M. Natalia, P. Afroditi, Q. Qiang, R. Miguel, R. Galen, and S. Guillermo, "Adversarially learned representations for information obfuscation and inference," in *Proc. International Conference on Machine Learning*, (Long Beach, CA), pp. 614–623, Jun. 2019.

[22] J. Hamm, "Minimax filter: Learning to preserve privacy from inference attacks," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 4704–4734, 2017.

[23] A. Tripathy, Y. Wang, and P. Ishwar, "Privacy-preserving adversarial networks," in *Proc. Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, (Monticello, IL), pp. 495–505, Sep. 2019.

[24] A. Böttcher and D. Wenzel, "The Frobenius norm and the commutator," *Linear Algebra and Its Applications*, vol. 429, no. 8-9, pp. 1864–1885, May 2008.

[25] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[26] F. Wang, W. Cao, and Z. Xu, "Convergence of multi-block bregman ADMM for nonconvex composite problems," *Science China Information Sciences*, vol. 61, no. 12, pp. 1–12, Dec. 2018.

[27] B. Fuglede and F. Topsoe, "Jensen-Shannon divergence and Hilbert space embedding," in *Proc. IEEE International Symposium on Information Theory*, (Parma, Italy), p. 31, Oct. 2004.

[28] B. Yu, "Assouad, Fano, and Le Cam," in *Festschrift for Lucien Le Cam*, pp. 423–435, Springer, 1997.

[29] L. Le Cam and G. L. Yang, *Asymptotics in statistics: some basic concepts*. Springer Science & Business Media, 2012.

[30] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*, vol. 87. Springer Science & Business Media, 2003.