Research Article

Limiting Privacy Breaches in Average-Distance Query

Huihua Xia, Yan Xiong, Wenchao Huang , Zhaoyi Meng, and Fuyou Miao

School of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui, China

Correspondence should be addressed to Wenchao Huang; huangwc@ustc.edu.cn

Received 11 April 2020; Accepted 13 July 2020; Published 6 August 2020

Academic Editor: Benjamin Aziz

Copyright © 2020 Huihua Xia et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Querying average distances is useful for real-world applications such as business decision and medical diagnosis, as it can help a decision maker to better understand the users' data in a database. However, privacy has been an increasing concern. People are now suffering serious privacy leakage from various kinds of sources, especially service providers who provide insufficient protection on user's private data. In this paper, we discover a new type of attack in an average-distance query (AVGD query) with noisy results. The attack is general that it can be used to reveal private data of different dimensions. We theoretically analyze how different factors affect the accuracy of the attack and propose the privacy-preserving mechanism based on the analysis. We experiment on two real-life datasets to show the feasibility and severity of the attack. The results show that the severity of the attack is mainly influenced by the factors including the noise magnitude, the number of queries, and the number of users in each query. Also, we validate the correctness of our theoretical analysis by comparing with the experimental results and confirm the effectiveness of the privacy-preserving mechanism.

1. Introduction

Nowadays, a major concern in the modern society is the leakage of private information, e.g., health condition information and location information. Reports show that healthcare data breaches in 2018 resulted in the exposure of 13 million healthcare records [1]. Breaches in the US healthcare field cost \$6.2 billion each year [2]. Meanwhile, the disclosure of personal location data can cause serious issues, e.g., by analyzing the semantic information (e.g., hospital and church) of users' locations, sensitive information such as home address, health status, and religious faith may be revealed [3, 4]. On the other hand, users' data are valuable to be queried for public usage. Therefore, privacy-preserving mechanisms are often used during the query processing [5]. However, it is still questionable whether existing mechanisms are sufficient for privacy.

We focus on the average-distance query (AVGD query) [6–8], which serves as the basic component of several applications such as business decision and medical diagnosis [8]. Given a database containing data of users, an AVGD query takes a query point and a set of users as inputs and queries the average distance between the query point and the data of those

users. The query point is a value located in the same space as the users' data. To the best of our knowledge, privacy breaches in the AVGD query have not been studied yet, and the user privacy needs attention during such queries. In this paper, we study the privacy breaches in the AVGD query and propose a new kind of attack. As we find, an adversary can make an attack by selecting the user sets and querying on specific query points and reveal users' data by leveraging the query results.

Both the following examples of the AVGD query, which correspond to different dimensions of query points, respectively, are exposed to the attack. Example 1: a company plans to deploy a new branch, before which the branch's location is required to be carefully chosen according to several facts, e.g., traveling time. In this case, a location-based service [8] benefits the choosing process by outputting the average distance between local users and a location inputted by the company. Here, the query point is the coordinate of the location. Example 2: medical data can be clustered to predict the likelihood of diseases. The average distance between elements in a cluster and the cluster centroid and the average distance between elements in two clusters are the basic measurements in clustering approaches. We consider the following one-dimensional case: given the

medical data (e.g., blood pressure) of a group of patients and a query point (e.g., the cluster centroid), a hospital queries the average distance between the data of each patient and the query point. Here, the query point is the real-valued medical data, and the absolute value distance is considered as the distance measure. To perform the attack, an adversary selects two user sets, which differ in one target user. Then, the adversary chooses a set of query points and gets the results queried on these query points and the two user groups. The adversary can set up equations based on the correlations between the results and recover the data of the target user. Detailed process of the attack will be described in Section 3.

Nevertheless, most of the existing techniques on privacy-preserving query processing cannot prevent these kind of attacks. K -anonymity-based approaches [9–11] are suitable for publishing or querying on single user's record, while the AVGQ query returns aggregate results. Other methods like data transformation [12, 13] and homomorphic encryption [8, 14] avoid transmitting user data in the plain text. They prevent the exposure of the original user data during the process of the query, but they do not consider the information leakage from the query results.

Noise-based output perturbation [5] (e.g., differential privacy with Laplace noise [15]) is a possible method to prevent the attack. However, as we demonstrate in this paper, the efficiency of the noise may be weakened under the attack. Too less noise may not provide enough preservation, while too much noise destroys the utility of the results. Thus, the amount of noise should be quantified for balancing utility and privacy. We aim to decide the minimum amount of noise to meet the desired privacy requirement. Previous works [16–18] have studied attacks on queries with noisy outputs, but they all have focused on sum queries or questions based on sum queries. They showed that the lower bounds of noises (as a function of the number of queries) are needed to prevent violations of privacy. To the best of our knowledge, we are the first to investigate attacks on AVGQ queries. Instead of giving a lower bound directly, we formalize the process of the attack and obtain explicit expressions of the uncertainty of an adversary's estimation under different conditions, which serves as a guide on the amount of noises to be added. As we further find, the uncertainty is affected by several factors during the queries, e.g., the number of users in each query, the data value of the target user, and the number of queries. Finally, we propose the privacy-preserving mechanism based on the analysis.

We evaluate the proposed attack on two real-world datasets: a one-dimensional medical dataset and a two-dimensional location dataset. The results show the feasibility and severity of such attacks. We compare the experimental results with our theoretical analysis under different factors and show the correctness of the theoretical analysis, which guarantees the effectiveness of the corresponding privacy-preserving mechanism.

The contributions in this paper are as follows:

- (i) We discover a new kind of attack on the AVGQ query where the adversary can recover a target user's data through analysis of query results

- (ii) We perform a detailed theoretical analysis of the attack and propose the privacy-preserving mechanism based on the analysis
- (iii) We evaluate the attack on real-world datasets. We show the severity of the attack and the effectiveness of our theoretical analysis

The rest of this paper is organized as follows. Section 2 reviews the relevant work on the average-distance query and privacy-preserving query processing. Section 3 gives the formal definition of the AVGQ query and the detailed process of the proposed attack and describes the goal of this paper. Section 4 theoretically analyses the uncertainty of the adversary's estimation. Section 5 gives a privacy-preserving mechanism based on the theoretical analysis. Section 6 conducts the attack on real-world datasets and gives the experimental results. Section 7 concludes this work and discusses future directions.

2. Related Work

2.1. Average-Distance Query. The average distance serves as a basic metric in several real-life applications. Scellato et al. [6] introduce the average distance as one of the metrics for the spatial properties of social networks. Armenatzoglou et al. [7] use the average distance as a metric for ranking users in a geosocial network. Yang et al. [19] seek to find a group of attendees for an impromptu activity that the average spatial distance between each attendee and the rally point is minimized under a restriction of the social relationship. The average distance metric is also used for the problem of min-dist optimal location selection [8, 20, 21]: given a set of existing facilities and a set of users, a location for a new facility is found such that the average distance between each user to his/her nearest facility is minimized.

2.2. Privacy-Preserving Query Processing. Several approaches have been proposed to meet the purpose of the privacy-preserving query. K -anonymity-based approaches [9] mix the target user's data with $k - 1$ other users' data such that an adversary cannot distinguish the target user from these k users. L -diversity [10] and t -closeness [11] are proposed to enhance the definition of k -anonymity. The attack proposed in this paper leverages the outputs of AVGQ queries, which are aggregate information, while k -anonymity-based approaches preserve privacy when publishing or querying on single user's data.

Approaches of data transformation [12, 13] transform the user's data before sending queries to the server. The server answers queries based on the transformed data other than the original data. Some approaches use cryptography [22–24] like homomorphic encryption [8, 14] to encrypt the data of the users and enable queries on encrypted data. These approaches avoid transmitting the original user data during the process of the query, thus protecting the user privacy, but they do not consider the privacy leakage from the query results, which motivated the attack proposed in this paper.

Differential privacy [15] is proposed to protect the privacy of individuals while releasing aggregate outputs. For

two neighboring databases, differential privacy requires that query results on these two databases are indistinguishable by an exponent factor. One way to achieve differential privacy is to add controlled noise (e.g., Laplace noise [15]) on the query results. Applying differential privacy to the AVGD query is possible, but it is essentially a noise-based method, and the amount of noise should be carefully chosen under different factors, as shown in this paper.

Dinur and Nissim [16] derived lower bounds for the noise needed in sum queries to prevent violations of privacy, with the assumption of unbounded and bounded adversaries. Several works [17, 18, 25] further improved the study with more general databases and more efficient attacks. Blum et al. [26] proposed a practical framework to preserve the privacy where the number of queries is limited sublinear in the size of the database. All these works focused on questions based on sum queries.

3. Problem Formulation

In this section, we give the formal definition of the AVGD query. Then, we show the proposed attack on the AVGD query and describe the goal of this paper.

3.1. System Model of Average-Distance Query. The system model of the AVGD query is as follows. There is a server that provides a service of the AVGD query and a client that requests queries. The server has a database which contains records of users. We assume that all these records are real valued. To run an AVGD query, the client needs to specify a set of users in the database. One way to specify the users is to use a uniform identifier between the server and the client, e.g., the phone number. The client can also specify the users based on some attributes of the users, e.g., patients who meet specific symptoms in a healthcare database.

The AVGD query is formalized as follows.

Average-distance query: (1) the client requests a query by choosing a user set U and a query point p ; (2) the server answers the query by computing the average distance $\text{Avg}(U, p) = \sum_{u \in U} d(t_u, p) / |U|$. Here, t_u denotes the data of u and $d(t_u, p)$ is the distance between p and t_u . The expression of d depends on detailed applications. The implementation of d depends on detailed applications. For example, it can be the absolute distance in a one-dimensional case or the spatial distance when the data are two-dimensional location points.

3.2. Adversary Model. The client is considered to be “semi-honest.” The client will follow the protocol correctly to request queries. He/she will not break into the server system to get users’ information illegally. However, the client may get extra information by analyzing the query results. The server is considered to be honest and is aware of the exact data value of each user but has to hide users’ data from the client.

3.3. Attack on the Query. An adversary can make an attack as shown in Figure 1. He/she chooses two groups of users U_1 and U_2 that differ in one user (the differed user can be

targeted based on the attributes when the adversary knows enough partial information [18]). $U_1 = \{u_1, u_2, \dots, u_k\}$ and $U_2 = U_1 \cup \{u_0\}$. We assume that the adversary is not aware of the value of k . Then, the adversary chooses query points $p_1, p_2, \dots, p_n$. For each p_i , he/she gets the query results $a_{1i} = \text{Avg}(U_1, p_i)$ and $a_{2i} = \text{Avg}(U_2, p_i)$. The equation holds that $a_{2i} = (ka_{1i} + d(t_0, p_i)) / (k + 1)$, where t_0 denotes the data of u_0 . Then, the adversary can solve equation (1) to recover t_0 :

$$\begin{cases} a_{21} = \frac{a_{11}|U_1| + d(t_0, p_1)}{|U_1| + 1}, \\ \vdots \\ a_{2n} = \frac{a_{1n}|U_1| + d(t_0, p_n)}{|U_1| + 1}. \end{cases} \quad (1)$$

The goal of this paper is to quantify the effect caused by the attacks and design the corresponding noise-based protection schemes. Specifically, in the protection scheme, we assume that the server adds noises on the query results. We consider two kinds of noises: the multiplicative noise and the additive noise. The multiplicative noise is represented as a random variable z , and thus the noisy output is $\hat{a}_{\text{mul}} = (1 + z)a$. We assume that z obeys Gaussian distribution, which has been used for the statistical disclosure control [5]. The additive noise is represented as a random variable $\tilde{a}$, and the noisy output is $\hat{a}_{\text{add}} = a + \tilde{a}$. We assume that $\tilde{a}$ obeys Laplace distribution, which is one of the means to provide differential privacy [15]. Both kinds of noises are assumed to have a mean value of 0 to avoid bias [5]. Note that we aim to present the attack with different kinds of noises. Enabling a differential privacy mechanism for the AVGD query is out of the scope of this paper, and we do not require the additive noise with Laplace distribution to satisfy the demand of differential privacy. We analyze how the noises affect the estimation of the user’s data value recovered by the adversary and the utility of the query results.

4. Uncertainty Analysis

To protect users’ privacy, the server replies query results added with noises. This causes uncertainty in adversary’s estimation of the target user’s data. In this section, we investigate the factors that influence the precision of the adversary’s estimation. Specifically, we solve the problem in a general model under two kinds of noises, i.e., the multiplicative noise and the additive noise, and then refine the solutions in two practical cases.

4.1. The Multiplicative Noise. The adversary solves the following fitting problem to recover the data of the target user. Assume the adversary chooses two user sets U_1 and U_2 that $U_2 = U_1 \cup \{u_0\}$. For a query point p , the noisy results of the AVGD queries with U_1 and U_2 are $\hat{a}_{\text{mul}_1} = (1 + z_1)a_1$ and $\hat{a}_{\text{mul}_2} = (1 + z_2)a_2$, respectively. We assume that $|z_1|, |z_2| \ll 1$. Let $k = |U_1|$ and the data of u_0 be t_0 , and the adversary needs to fit the unknown parameters k and t_0 in

FIGURE 1: The process of the attack.

the model of equation (2) with sets of $\{p_i\}$ and noisy $\{\hat{a}_{mul_1i}\}$ and $\{\hat{a}_{mul_2i}\}$, where $\hat{a}_{mul_1i} = (1 + z_{1i})a_{1i}$ and $\hat{a}_{mul_2i} = (1 + z_{2i})a_{2i}$ are the noisy outputs queried on the i -th query point p_i . a_{1i} and a_{2i} are the exact results queried on p_i , and the noises added on them are z_{1i} and z_{2i} , respectively:

$$a_2 = \frac{ka_1 + d(t_0, p)}{k + 1}. \quad (2)$$

We simplify the model as follows. Let $\mathbf{x} = [a_1, p]^T$ and $G = a_2$. Let $\alpha = [k, t_0]^T$ denote the parameters to be fit. We simplify the model in equation (2) as

$$G = F(\mathbf{x}, \alpha). \quad (3)$$

Then, the problem is to fit α in equation (3) with noisy $\{\hat{x}_{mul_i}\}$ and $\{\hat{G}_{mul_i}\}$, where $\hat{x}_{mul_i} = [\hat{a}_{mul_1i}, p_i]^T$ and $\hat{G}_{mul_i} = (1 + z_{2i})G_i$. Here, $G_i = a_{2i}$. We write $\hat{x}_{mul_i} = (1 + \mathbf{w}_i) \circ \mathbf{x}_i$, where $\mathbf{x}_i = [a_{1i}, p_i]^T$, $\mathbf{w}_i = [z_{1i}, 0]^T$, and the symbol $(\circ)$ denotes the Hadamard product.

Next, we derive the uncertainty of the estimation of the parameter α in equation (3) caused by the noise. Sader et al. [27] solved the problem of uncertainty analysis while fitting a function with noisy output data (G). We expand their work and enable uncertainty analysis while both input ($\mathbf{x}$) and output data (G) contain noises. The method of least squares is used to solve such a fitting problem with the goal that the residual S [27] is minimized, where

$$S = \sum_i [F((1 + \mathbf{w}_i) \circ \mathbf{x}_i, \alpha') - (1 + z_{2i})G_i]^2, \quad (4)$$

and α' is the estimated value of α . Thus, α' satisfies the following equation:

$$\sum_i [F((1 + \mathbf{w}_i) \circ \mathbf{x}_i, \alpha') - (1 + z_{2i})G_i] \times \nabla_{\alpha} F((1 + \mathbf{w}_i) \circ \mathbf{x}_i, \alpha') = 0, \quad (5)$$

where ∇_{α} is the gradient with respect to α .

We expand $\alpha' = \alpha_0 + \alpha_1$, where α_0 is the solution in the noiseless condition; i.e., α_0 is the ideal value of the fitting parameter; α_1 denotes the deviation between the estimation and the ideal value. Then, we have

$$\begin{aligned} F((1 + \mathbf{w}_i) \circ \mathbf{x}_i, \alpha') &= F(\mathbf{x}_i, \alpha_0) + \alpha_1 \cdot \nabla_{\alpha} F(\mathbf{x}_i, \alpha_0) \\ &\quad + (\mathbf{w}_i \circ \mathbf{x}_i) \cdot \nabla_{\mathbf{x}} F(\mathbf{x}_i, \alpha_0) \\ &\quad + o\left(\sqrt{\|\mathbf{w}_i \circ \mathbf{x}_i\|^2 + \|\alpha_1\|^2}\right), \end{aligned} \quad (6)$$

where $\nabla_{\mathbf{x}}$ is the gradient with respect to $\mathbf{x}$.

As we assume α_0 as the solution without the effect of noise, we can get

$$G_i = F(\mathbf{x}_i, \alpha_0). \quad (7)$$

Substituting equations (6) and (7) into equation (5) and ignoring the high-order small components, we get the expression of the deviation in the fit parameters caused by the noise:

$$\alpha_1 = \sum_i \mathbf{A}^{-1} \left[z_{2i} G_i \nabla_{\alpha} F(\mathbf{x}_i, \alpha_0) - \left(\nabla_{\alpha} F(\mathbf{x}_i, \alpha_0) \nabla_{\mathbf{x}}^T F(\mathbf{x}_i, \alpha_0) \right) (\mathbf{w}_i \circ \mathbf{x}_i) \right], \quad (8)$$

where

$$\mathbf{A} = \sum_i \nabla_{\alpha} F(\mathbf{x}_i, \alpha_0) \nabla_{\alpha}^T F(\mathbf{x}_i, \alpha_0). \quad (9)$$

The expectation and variance of the fitting parameters can be approximated in the form of integrals. We assume that the query points $\{p_i\}$ are uniformly picked from a query range D . We set δp as the average area in D segmented by the query points. z_{1i} and z_{2i} are noises added by the server and are assumed to follow the same distribution that $E[z_{1i}] = E[z_{2i}] = E[z_{11}]$ and $\text{Var}[z_{1i}] = \text{Var}[z_{2i}] =$

$\text{Var}[z_{11}]$. Also, $E[\mathbf{w}_i] = E[\mathbf{w}_1]$ and $\text{Var}[\mathbf{w}_i] = \text{Var}[\mathbf{w}_1]$. We approximate the sums in equations (8) and (9) with integrals and get the results for the expectation and variance of the m -th component, $\alpha_{1|m}$, of α_1 , under the multiplicative noise:

$$E_{\text{mul}}[\alpha_{1|m}] = \int_D (E[z_{11}]F(\mathbf{x}, \alpha_0) [\mathbf{B}^{-1} \nabla_{\alpha} F(\mathbf{x}, \alpha_0)]_m - [\mathbf{B}^{-1} (\nabla_{\alpha} F(\mathbf{x}, \alpha_0) \nabla_{\mathbf{x}}^T F(\mathbf{x}, \alpha_0)) (E[\mathbf{w}_1] \circ \mathbf{x})]_m) dp, \quad (10)$$

$$\begin{aligned} \text{Var}_{\text{mul}}[\alpha_{1|m}] = & \delta p \int_D (\text{Var}[z_{11}] (F(\mathbf{x}, \alpha_0) \\ & \times [\mathbf{B}^{-1} \nabla_{\alpha} F(\mathbf{x}, \alpha_0)]_m)^2 \\ & + [\text{Ewp}(\mathbf{B}^{-1} (\nabla_{\alpha} F(\mathbf{x}, \alpha_0) \nabla_{\mathbf{x}}^T F(\mathbf{x}, \alpha_0))) \\ & \times (\text{Var}[\mathbf{w}_1] \circ \text{Ewp}(\mathbf{x}))]_m) dp, \end{aligned} \quad (11)$$

where $[\mathbf{v}]_m$ denotes the m -th component of vector $\mathbf{v}$ and for matrix $\mathbf{M}$, $[\text{Ewp}(\mathbf{M})]_{ij} = \mathbf{M}_{ij}^2$. $\mathbf{B}$ is denoted as

$$\mathbf{B} = \int_D \nabla_{\alpha} F(\mathbf{x}, \alpha_0) \nabla_{\alpha}^T F(\mathbf{x}, \alpha_0) dp. \quad (12)$$

4.2. The Additive Noise. The adversary solves a similar fitting problem as the case of the multiplicative noise, where the fitting model is the same as equation (2). The only difference is that the noisy results of the AVGD queries with U_1 and U_2 under the additive noises are $\hat{a}_{\text{add},1} = a_1 + \tilde{a}_1$ and $\hat{a}_{\text{add},2} = a_2 + \tilde{a}_2$, respectively. We assume that $|\tilde{a}_1| \ll a_1$ and $|\tilde{a}_2| \ll a_2$. The residual which aims to be minimized in equation (4) changes into

$$S' = \sum_i [F(\mathbf{x}_i + \tilde{\mathbf{x}}_i, \alpha') - (G_i + \tilde{G}_i)]^2, \quad (13)$$

where $\tilde{\mathbf{x}}_i = [\tilde{a}_{1i}, 0]^T$ and $\tilde{G}_i = \tilde{a}_{2i}$.

We omit the detailed derivation. $\tilde{a}_{1i}$ and $\tilde{a}_{2i}$ are assumed to follow the same Laplace distribution that $E[\tilde{a}_{1i}] = E[\tilde{a}_{2i}] = E[\tilde{a}_{11}]$ and $\text{Var}[\tilde{a}_{1i}] = \text{Var}[\tilde{a}_{2i}] = \text{Var}[\tilde{a}_{11}]$. Also, $E[\tilde{\mathbf{x}}_i] = E[\tilde{\mathbf{x}}_1]$ and $\text{Var}[\tilde{\mathbf{x}}_i] = \text{Var}[\tilde{\mathbf{x}}_1]$. We give the expectation and variance of the uncertainty of the adversary's estimation, α_1 , as follows, where δp is the average area in D segmented by the query points and $\mathbf{B}$ is defined in equation (12):

$$E_{\text{add}}[\alpha_{1|m}] = \int_D (E[\tilde{\alpha}_{11}] [\mathbf{B}^{-1} \nabla_{\alpha} F(\mathbf{x}, \alpha_0)]_m - [\mathbf{B}^{-1} (\nabla_{\alpha} F(\mathbf{x}, \alpha_0) \nabla_{\mathbf{x}}^T F(\mathbf{x}, \alpha_0)) E[\tilde{\mathbf{x}}_1]]_m) dp, \quad (14)$$

$$\text{Var}_{\text{add}}[\alpha_{1|m}] = \delta p \int_D (\text{Var}[\tilde{\alpha}_{11}] [\mathbf{B}^{-1} \nabla_{\alpha} F(\mathbf{x}, \alpha_0)]_m^2 + [\text{Ewp}(\mathbf{B}^{-1} \nabla_{\alpha} F(\mathbf{x}, \alpha_0) \nabla_{\mathbf{x}}^T F(\mathbf{x}, \alpha_0)) \text{Var}[\tilde{\mathbf{x}}_1]]_m) dp. \quad (15)$$

4.2.1. Case 1: One-Dimensional Average-Distance Query. We consider a query of one-dimensional average distance, i.e., the queried records of users are one-dimensional real values. For instance, the queried data can be the diastolic blood pressure (diaBP) data of users from a medical database. In this case, given a user set U and a query point $p = v_p$, the query function is $\text{Avg}_1(U, v_p) = (1/k) \sum_{u \in U} |v_p - v_u|$, where k denotes the number of users in U and v_u denotes the data of user u . Assuming that the value of the target user u_0 is v_0 , the model for the adversary to fit is

$$a_2 = \frac{ka_1 + |v_p - v_0|}{k+1}. \quad (16)$$

We compute the expectation and variance of the adversary's estimation in the case of multiplicative and additive noises, as in equations (10), (11), (14), and (15). In both cases, we have $\alpha_0 = [k, v_0]^T$ and $\mathbf{x} = [a_1, v_p]^T$. We assume that the multiplicative noise obeys Gaussian distribution with an expected value of 0 and the variance of $\text{Var}[z]$, such that $E[z_{11}] = 0$ and $E[\mathbf{w}_1] = [0, 0]^T$ in equation (10), $\text{Var}[z_{11}] = \text{Var}[z]$ and $\text{Var}[\mathbf{w}_1] = [\text{Var}[z], 0]^T$ in

equation (11). The additive noise is assumed to obey Laplace distribution with zero mean and variance of $\text{Var}[\tilde{a}]$, and then $E[\tilde{a}_{11}] = 0$ and $E[\tilde{\mathbf{x}}_1] = [0, 0]^T$ in equation (14) and $\text{Var}[\tilde{a}_{11}] = \text{Var}[\tilde{a}]$ and $\text{Var}[\tilde{\mathbf{x}}_1] = [\text{Var}[\tilde{a}], 0]^T$ in equation (15).

To simplify the results, we give an approximation of a_1 as follows. Assume the data in the database follow a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$, and the query range is from v_m to v_n . Then, we approximate the value of a_1 as

$$a_1 = \int_{v_m}^{v_n} |x - v_p| f(x) dx, \quad (17)$$

where $f(x) = (1/\sqrt{2\pi}\sigma) e^{-((x-\mu)^2/2\sigma^2)}$ is the probability density function.

Then, the uncertainty of the adversary's estimation is as follows. Let v'_0 be the adversary's estimation of v_0 and $v_e = v'_0 - v_0$ be the error of the adversary's estimation. The adversary chooses a total of n_{query} query points, and thus $\delta p = (v_n - v_m)/n_{\text{query}}$. According to equations (10), (11), (14), and (15), we can get the variance and the expected value of v_e under the assumption of multiplicative noise and additive noise, respectively, as

$$\text{Var}_{\text{mul}}[v_e] = \frac{v_n - v_m}{n_{\text{query}}} \text{Var}[z] (c_0(s_0) + c_1(s_0)k + c_2(s_0)k^2), \quad (18)$$

$$\text{Var}_{\text{add}}[v_e] = \frac{v_n - v_m}{n_{\text{query}}} \text{Var}[\tilde{a}] c_3(s_0) (2k^2 + 2k + 1), \quad (19)$$

$$E_{\text{mul}}[v_e] = E_{\text{add}}[v_e] = 0, \quad (20)$$

where $s_0 = (v_0 - v_m)/(v_n - v_m)$ denotes the relative value of v_0 in the query range and $c_0(s_0) \sim c_3(s_0)$ are the coefficients whose values only rely on s_0 .

4.2.2. Case 2: Two-Dimensional Average-Distance Query.

We consider the query of the average distance between spatial locations in this case. Assume that the server has a database of users' location data. The coordinates of locations are in the polar coordinate system. The client queries the average distance between a set of k users U and a specified query location $p = (\rho_p, \theta_p)$, where $0 \leq \rho_p \leq R$ and $0 \leq \theta_p \leq 2\pi$. The query function $\text{Avg}_2(U, p) = 1/k \sum_{u \in U} \sqrt{\rho_u^2 + \rho_p^2 - 2\rho_u \rho_p \cos(\theta_u - \theta_p)}$, where (ρ_u, θ_u) denotes the coordinate of user u . We denote the location of the target user as (ρ_0, θ_0) . In this case, the model for the adversary to fit is

$$a_2 = \frac{ka_1 + \sqrt{\rho_0^2 + \rho_p^2 - 2\rho_0 \rho_p \cos(\theta_0 - \theta_p)}}{k + 1}. \quad (21)$$

In this case, we have $\alpha_0 = [k, \rho_0, \theta_0]^T$ and $\mathbf{x} = [a_1, \rho_p, \theta_p]^T$ in equations (10), (11), (14), and (15). The multiplicative noise added on the query results in equations (10) and (11) is assumed to obey the Gaussian distribution with the expected value of 0 and the variance of $\text{Var}[z]$. Thus, we have $E[z_{11}] = 0$, $E[\mathbf{w}_1] = [0, 0, 0]^T$, $\text{Var}[z_{11}] = \text{Var}[z]$, and $\text{Var}[\mathbf{w}_1] = [\text{Var}[z], 0, 0]^T$. The additive noise in equations (14) and (15) is assumed to obey Laplace distribution with zero mean and variance of $\text{Var}[\tilde{a}]$, and then $E[\tilde{a}_{11}] = 0$ and $E[\tilde{x}_1] = [0, 0, 0]^T$ in equation (14) and $\text{Var}[\tilde{a}_{11}] = \text{Var}[\tilde{a}]$ and $\text{Var}[\tilde{x}_1] = [\text{Var}[\tilde{a}], 0, 0]^T$ in equation (15).

We give an approximation of a_1 in this case as follows. We find that users' check-in locations show concentrations at the center of the city and disperse towards the edge of the city in real datasets. Figure 2 shows the distribution of the check-ins of users in New York from a real-world dataset [28]. The distribution is shown as a heat map that the red area shows higher concentrations of check-ins than the blue area. The check-ins concentrate in the center of the city and decrease towards the outskirts, and this comes from the reason that people are more active in the city center. Accordingly, we make an assumption that the coordinates of users follow the distribution that the radial coordinate and the angular coordinate of each user are uniformly selected from $0 \sim R$ and $0 \sim 2\pi$, which gives concentrations of points in the center. Then, a_1 can be represented as

FIGURE 2: Distribution of check-ins in New York.

$$a_1 = \frac{1}{2\pi R} \int_0^R \int_0^{2\pi} \sqrt{\rho_p^2 + \rho^2 - 2\rho \rho_p \cos(\theta)} d\theta d\rho. \quad (22)$$

The uncertainty of the adversary's estimation in this case is given as follows. Let the adversary's estimation of (ρ_0, θ_0) be (ρ'_0, θ'_0) . $\rho_e = \rho'_0 - \rho_0$ and $\theta_e = \theta'_0 - \theta_0$ denote the errors of the adversary's estimations of ρ_0 and θ_0 . The adversary queries on a total of n_{query} query points. According to equations (10) and (11) and letting $\delta p = \pi R^2/n_{\text{query}}$, we can get the variances of ρ_e and θ_e under the assumption of multiplicative and additive noises, respectively, as

$$\text{Var}_{\text{mul}}[\rho_e] = \frac{\pi R^2}{n_{\text{query}}} \text{Var}[z] (c_4(s_0) + c_5(s_0)k + c_6(s_0)k^2), \quad (23)$$

$$\text{Var}_{\text{mul}}[\theta_e] = \frac{\pi}{n_{\text{query}}} \text{Var}[z] (c_7(s_0) + c_8(s_0)k + c_9(s_0)k^2), \quad (24)$$

$$\text{Var}_{\text{add}}[\rho_e] = \frac{\pi}{n_{\text{query}}} \text{Var}[\tilde{a}] c_{10}(s_0) (2k^2 + 2k + 1), \quad (25)$$

$$\text{Var}_{\text{add}}[\theta_e] = \frac{\pi}{n_{\text{query}} R^2} \text{Var}[\tilde{a}] c_{11}(s_0) (2k^2 + 2k + 1), \quad (26)$$

where $s_0 = \rho_0/R$ denotes the relative location of u_0 in the query range and $c_4(s_0) \sim c_{11}(s_0)$ are the coefficients whose values are determined by s_0 . The expected values of ρ_e and θ_e are both 0:

$$E_{\text{mul}}[\rho_e] = E_{\text{mul}}[\theta_e] = E_{\text{add}}[\rho_e] = E_{\text{add}}[\theta_e] = 0. \quad (27)$$

4.2.3. Findings. An obvious finding in equations (18), (19), and (23)~(26) is that the variances of v_e , ρ_e , and θ_e are in direct proportion to $\text{Var}[z]$ or $\text{Var}[\tilde{a}]$ and in inverse proportion to n_{query} . In other words, the noise of the query results brings difficulties for adversaries to reveal the target user's data. Besides, as the number of returned query results increases, the risk of data leakage also increases since more

equations can be built to solve the fitting parameters. Moreover, fewer number of users in a query would also lead to a higher accuracy of malicious estimation of users' data.

5. Privacy-Preserving Mechanism

In this section, we give a privacy-preserving mechanism for AVGD queries. We define the privacy metric for the AVGD query. Then, we design a mechanism to decide the minimum amount of noise for a query given the desired privacy requirement.

5.1. Privacy Metric for AVGD Query. We define the privacy metric for the AVGD query as follows. Hoh et al. [29] and Shokri et al. [30] quantified the location privacy as the expected distance between the user's location and the adversary's estimation. We follow the idea and quantify the privacy in the attack on AVGD queries as the expected error distance (eed), i.e., the expectation of the distance between the target user's data and the adversary's estimation. The definition of eed is as follows:

$$\text{eed} = E[d(t - t')], \quad (28)$$

where t is the real data of the target user and t' is the value estimated by the adversary. Therefore, a smaller eed corresponds to a higher risk of privacy leakage. Then, we show the detailed description of eed in the one-dimensional case (eed₁) and the two-dimensional case (eed₂).

5.1.1. One-Dimensional Case. According to equation (8), the error of an adversary's estimation is the sum of random variables. In the case of multiplicative noises, the noises are assumed to obey Gaussian distribution. Thus, the error is the sum of Gaussian noises, which obeys Gaussian distribution as well. The noises are assumed to obey Laplace distribution in the case of the additive noise. However, the sum of Laplace noises may not obey Laplace distribution. We empirically approximate the distribution of the error in the case of additive noises as a Gaussian distribution.

The expectations of the errors in the multiplicative noise case and the additive noise case are both 0, and the variances are described in equations (18) and (19), respectively. We omit the subscripts in $\text{Var}_{\text{mul}}[v_e]$ and $\text{Var}_{\text{add}}[v_e]$ and simply write them as $\text{Var}[v_e]$. According to equation (28), we have

$$\text{eed}_1 = \sqrt{\frac{2\text{Var}[v_e]}{\pi}}. \quad (29)$$

5.1.2. Two-Dimensional Case. Denote the real coordinate of the target user as (ρ_0, θ_0) and the estimation of the adversary as (ρ'_0, θ'_0) . Let $\rho_e = \rho'_0 - \rho_0$ and $\theta_e = \theta'_0 - \theta_0$ denote the errors of the adversary's estimations on ρ_0 and θ_0 , respectively. Then, the distance between the real coordinate of the target user and the adversary's estimation is

$$\Delta l = \sqrt{\rho_0^2 + (\rho_0 + \rho_e)^2 - 2\rho_0(\rho_0 + \rho_e)\cos(\theta_e)}. \quad (30)$$

Approximating $\cos(\theta_e)$ with $1 - \theta_e^2/2$, we get

$$\Delta l = \sqrt{\rho_e^2 + \rho_0(\rho_0 + \rho_e)\theta_e^2}. \quad (31)$$

eed₂ in the two-dimensional case is

$$\text{eed}_2 = E[\Delta l]. \quad (32)$$

To calculate the expected value of Δl , we use the following approximation. For a function $g(x)$ on a random variable x , using Taylor approximation [31], we can get

$$E[g(x)] \approx g(E[x]) + \frac{\text{Var}[x]}{2}g''(E[x]). \quad (33)$$

ρ_e and θ_e are random variables that obey Gaussian distribution in the case of multiplicative noise and are approximated to obey Gaussian distribution in the case of additive noise. Both the expected values of ρ_e and θ_e are 0, and the variances are $\text{Var}[\rho_e]$ and $\text{Var}[\theta_e]$ (subscripts are omitted) respectively, as described in equations (23) to (26). We have

$$E[\rho_e^2 + \rho_0(\rho_0 + \rho_e)\theta_e^2] = \text{Var}[\rho_e] + \rho_0^2\text{Var}[\theta_e], \quad (34)$$

$$\text{Var}[\rho_e^2 + \rho_0(\rho_0 + \rho_e)\theta_e^2] = 2\text{Var}^2[\rho_e] + 2\rho_0^4\text{Var}^2[\theta_e] + 3\rho_0^2\text{Var}[\rho_e]\text{Var}^2[\theta_e]. \quad (35)$$

Substituting equations (31), (34), and (35) into (33), we can get the expression for eed₂.

5.2. Privacy-Preserving Algorithm. Given the privacy metric for the AVGD query, we design a mechanism for the privacy-preserving AVGD query, as shown in Algorithm 1. Since the precision of an adversary's estimation increases when he/she gets more results, and when there are fewer users in each query, we assume that the server limits the number of queries on a specified user set that each client can request as n_{query} and limits the minimum number of users in each query as k . Each user u in the database has the privacy requirement eed _{u} ; i.e., eed of the adversary's estimation on the data of u must be less than eed _{u} . Assume that a client specifies a user set U and a query point q . Then, the server calibrates the noise magnitude such that the privacy requirement of each user in U is satisfied and responses the client with noisy results.

6. Experiment

In this section, we perform the attack on real-world datasets and seek to answer the following questions:

How do different factors influence the uncertainty of the adversary's estimation?

How is the severity of the attack in real-life cases?

How is the correctness of our theoretical analysis?

How much utility is lost when enabling privacy preserving?

Input: n_{query} : the limited number of queries on a specified user set; k : the minimum number of users in each query; U : the user set that the client specifies; q : the query point that the client specifies.

Output: $\hat{a}$: the noisy output.

- (1) **if** $|U| < k$ **then**
- (2) **return false**
- (3) **end if**
- (4) **for** each $u \in U$ **do**
- (5) Get the privacy requirement eed_u of u .
- (6) Solve the variance σ_i^2 such that the privacy requirement eed_u can be met.
- (7) **end for**
- (8) $\sigma^2 = \max \sigma_i^2$ in $\{\sigma_i^2\}$
- (9) Compute the correct answer $a = \text{Avg}(U, q)$.
- (10) $z \sim \mathcal{N}(0, \sigma^2)$ for multiplicative noises, or $\tilde{a} \sim \text{Laplace}(0, \sigma/\sqrt{2})$ for additive noises.
- (11) $\hat{a} = (1 + z)a$ for multiplicative noises, or $\hat{a} = a + \tilde{a}$ for additive noises.
- (12) **return** $\hat{a}$

ALGORITHM 1: Privacy-preserving AVGD query processing.

6.1. Dataset

6.1.1. One-Dimensional Case. We use a medical dataset from the Framingham Heart Study [32] that contains health data of 4,240 users. We use the diastolic blood pressure (DiaBP) data of each user as the inputs in our experiments. The DiaBP data in this dataset range from 48 mmHg to 142.5 mmHg and obey Gaussian distribution with a mean value of 81.31 mmHg and a standard deviation of 10.98 mmHg.

6.1.2. Two-Dimensional Case. We use a location dataset collected from Foursquare [33]. The dataset contains check-ins contributed by users in New York from 12 April 2012 to 16 February 2013 [28]. Each check-in is associated with a user-id, a timestamp, and the GPS coordinate. The original dataset contains 1,083 unique users and 227,428 check-ins. In our settings, we only consider one check-in for each user in each query. For simplicity, we choose the latest check-in of each user as the target of the clients in our experiments. The radius of this dataset is 28.5 km.

6.2. Measurement. We aim to measure the severity of the attack and the correctness of our theoretical analysis. The severity of the attack is measured as eed_1 and eed_2 defined in equations (29) and (32) for the two cases. Smaller values of eed_1 and eed_2 mean higher severity. We measure the correctness of the standard deviations in equations (18), (19), and (23) to (26) and eed_1 and eed_2 in equations (29) and (32). The correctness here is measured as the error rate between the theoretical estimation and the experimental value, i.e.,

$$\text{error rate} = \frac{|y_t - y_e|}{y_t}, \quad (36)$$

where y_t and y_e denote the theoretical estimation and the experimental value, respectively.

6.3. Experimental Setup and Statistical Result. The behavior of an adversary is simulated as follows. The dataset and a

query range specified by the server are given. The adversary focuses on a target user u_0 (assume that u_0 is within the query range). The adversary randomly chooses a user set U_1 within the query range containing k users and a neighboring user set $U_2 = U_1 \cup \{u_0\}$. Assume that the maximum number of queries that each client can request on a specified user set is n_{query} . The adversary chooses n_{query} query points within the query range. In the one-dimensional case, we set the maximum query range from 20 mmHg to 145 mmHg. In the two-dimensional case, the query range for the angular coordinate is from 0 to 2π , and the maximum query range for the radial coordinate is from 0 m to 28,500 m. For each query point p , the adversary requests AVGD queries on U_1 and U_2 , respectively. The server computes results $a_1 = \text{Avg}(U_1, p)$ and $a_2 = \text{Avg}(U_2, p)$ and adds multiplicative noises z_1 and z_2 (resp., additive noises $\tilde{a}_1$ and $\tilde{a}_2$) on the query results such that the noisy outputs are $\hat{a}_1 = (1 + z_1)a_1$ and $\hat{a}_2 = (1 + z_2)a_2$ (resp., $\hat{a}_1 = a_1 + \tilde{a}_1$ and $\hat{a}_2 = a_2 + \tilde{a}_2$), where z_1 and z_2 (resp., $\tilde{a}_1$ and $\tilde{a}_2$) are both random variables from Gaussian distribution $\mathcal{N}(0, \text{Var}[z])$ (resp., Laplace distribution $\text{Laplace}(0, \sqrt{\text{Var}[\tilde{a}]/2})$). The adversary gets a total of $2n_{\text{query}}$ noisy results and solves equations like equation (1) to recover the data of u_0 . We use the *lsqnonlin* function with the default trust-region-reflective algorithm [34] in MATLAB R2018b to solve the nonlinear least-square problems.

We study the influences on the adversary's estimation caused by different factors. The influences include the uncertainty of the adversary's estimation, eed of the adversary, and the error rate of the theoretical estimation. The factors we consider in our experiments are as follows: the magnitude (standard deviation) of the noise added on the query results ($\text{SD}[z]$ and $\text{SD}[\tilde{a}]$), the number of users in the query requested by the adversary (k), the limited number of queries on a specified user set for each client (n_{query}), the relative position of the target user's data in the query range (s_0), and the size of the query range.

In our experiments, we observe the influences by simulating attacks with different values of the parameters. Under each configuration of parameters, we perform the simulation of the attack described above 2,000 times. The

detailed settings of the parameters and the results are as follows. Note that in Section 6.3, we only demonstrate the results, and the detailed analysis is in Section 6.4–6.6.

We first demonstrate the influence of the noise magnitude. In the experiments, we fix the query range with the maximum query range and set $k = 50$, $n_{\text{query}} = 2,000$, and $s_0 = 0.5$ (in the two-dimensional case, s_0 denotes the relative position of the radial coordinate of the target user. The angular coordinate does not affect the uncertainty of the adversary as in equations (23)–(26), thus we set it with 0). For the one-dimensional case, we change the multiplicative noise $\text{SD}[z]$ (resp., additive noise $\text{SD}[\tilde{a}]$) from 0.01 to 0.1 (resp., 0.1 mmHg to 1 mmHg). For the two-dimensional case, we change $\text{SD}[z]$ (resp., $\text{SD}[\tilde{a}]$) from 0.001 to 0.01 (resp., 10 m to 100 m). The results of the first experiment are shown in Figures 3 and 4.

Secondly, we study the influence caused by the number of users in each query. We fix $\text{SD}[z] = 0.01$, $\text{SD}[\tilde{a}] = 0.1$ mmHg for the one-dimensional case and $\text{SD}[z] = 0.001$, $\text{SD}[\tilde{a}] = 10$ m for the two-dimensional case. For both the cases, we fix the query range with the maximum query range and fix parameters $n_{\text{query}} = 2,000$ and $s_0 = 0.5$ and change k from 20 to 200. The results are shown in Figures 5 and 6.

We then show the effect of the number of queries on a specified user set. We set $\text{SD}[z] = 0.01$, $\text{SD}[\tilde{a}] = 0.1$ mmHg for the one-dimensional case and $\text{SD}[z] = 0.001$, $\text{SD}[\tilde{a}] = 10$ m for the two-dimensional case and fix $k = 50$ and $s_0 = 0.5$ and fix the query range with the maximum query range for both cases. We choose n_{query} from 200 to 2,000 for both cases. The results are shown in Figures 7 and 8.

We also study the effect of the relative position (s_0) of the target user's data in the query range. Note that in the two-dimensional case, we only consider the impact of the radial coordinate of the target user because the angular coordinate has no effect on the uncertainty of the estimation. The parameter $\text{SD}[z]$ (resp., $\text{SD}[\tilde{a}]$) is set with 0.01 (resp., 0.1 mmHg) for the one-dimensional case and 0.001 (resp., 10 m) for the two-dimensional case. We fix the query range with the maximum query range and fix $k = 50$ and $n_{\text{query}} = 2,000$ and choose s_0 from 0.1 to 1 for both cases. The results are shown in Figures 9 and 10.

At last, we demonstrate the influence of the size of the query range. We fix $\text{SD}[z] = 0.01$, $\text{SD}[\tilde{a}] = 0.1$ mmHg for the one-dimensional case and $\text{SD}[z] = 0.001$, $\text{SD}[\tilde{a}] = 10$ m for the two-dimensional case. For both the cases, parameters $k = 50$, $n_{\text{query}} = 2,000$, and $s_0 = 0.5$ are fixed. We change the size of the query ranges. Specifically, we fix the central point of the query ranges and change the ranges around the central point. The results are shown in Figures 11 and 12.

We omit the results of $E[v_e]$, $E[\rho_e]$, and $E[\theta_e]$. Changes of these factors do not show obvious impacts on $E[v_e]$, $E[\rho_e]$, and $E[\theta_e]$ in the experimental results.

6.4. Effects of Different Factors

6.4.1. The Noise Magnitude. As shown in Figures 3 and 4, the standard deviation of the adversary's estimation error and

eed of the adversary are in direct proportion with the standard deviation of the noise. The precision of the adversary's estimation decreases when the server adds larger noises on the query results.

6.4.2. The Number of Users in Each Query. The standard deviation of the adversary's estimation error and eed of the adversary's estimation are in direct proportion to the number of users in each query, as shown in Figures 5 and 6. The precision of the adversary's estimation decreases when the number of users in each query increases. The reason is as follows. For $a_1 = \text{Avg}(U_1, p)$ and $a_2 = \text{Avg}(U_2, p)$, where $U_2 = U_1 \cup \{u_0\}$, the difference between a_1 and a_2 is smaller when there are more users in U_1 , and thus it is more difficult for an adversary to recover the target user's data.

6.4.3. The Number of Queries. As per the results shown in Figures 7 and 8, the standard deviation of the estimation error and eed decrease when n_{query} increases. More queries generate more results which can be used for the fitting problem, and thus the adversary gets a more precise estimation.

6.4.4. The Data of the Target User. As shown in Figures 9 and 10, the influences of s_0 on the uncertainty of the adversary do not show a consistent pattern, which depend on specific cases. Also, the multiplicative noises and the additive noises in the same case affect the results differently.

6.4.5. The Size of the Query Range. As shown in Figures 11 and 12, for both one-dimensional and two-dimensional cases, eed of the adversary's estimation increases under multiplicative noise when the query range gets larger. When the noise is additive, eed does not change much with different query ranges. Actually, the value of eed is closely related with the absolute noise magnitude. The additive noise itself is the absolute noise and remains the same with different query ranges. The multiplicative noise is the relative noise. When the value of $\text{SD}[z]$ is fixed, the absolute noise magnitude increases with larger query ranges, and thus the value of eed rises.

6.5. Severity of the Attack. We demonstrate the severity of the attack based on the theoretical estimation and the results on real-world datasets. First of all, the severity of the attack depends on the sensitivity of the data in different cases, i.e., to what extent the information leakage threatens a user's privacy. For example, in the two-dimensional case, when an adversary has $\text{eed}_2 = 1,000$ m, it is difficult for an adversary to link the estimated location with a specific sensitive place (e.g., hospital and church) with such uncertainty. But, in the one-dimensional case, if the adversary has $\text{eed}_1 = 5$ mmHg, a health problem such as high blood pressure of the target user may be leaked. Besides the sensitivity of different cases, the severity of the attack increases when the adversary gets a

FIGURE 3: Impact of the noise magnitude (diaBP dataset).

(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 4: Continued.

FIGURE 4: Impact of the noise magnitude (check-in dataset).

FIGURE 5: Continued.

(e)

(f)

FIGURE 5: Impact of the number of users in each query (diaBP dataset).

(a)

(b)

(c)

(d)

FIGURE 6: Continued.

FIGURE 6: Impact of the number of users in each query (check-in dataset).

higher accuracy of the estimation, which is mainly influenced by the following three facts.

6.5.1. The Noise Magnitude. When the server reduces the noise magnitude, eed of the adversary decreases, thus the severity rises. In Figure 3, eed_1 decreases from 5.6 mmHg to 0.45 mmHg (resp., 1.25 mmHg to 0.13 mmHg) when $SD[z]$ changes from 0.1 to 0.01 (resp., $SD[\tilde{a}]$ changes from 1 mmHg to 0.1 mmHg). In Figure 4, eed_2 decreases from 631 m to 71.8 m (resp., 277 m to 28 m) when $SD[z]$ changes from 0.01 to 0.001 (resp., $SD[\tilde{a}]$ changes from 100 m to 10 m). Moreover, the proposed attack can resist the noise to some extent. As shown in Figure 3, the adversary gets the eed_1 of about 5.6 mmHg even when the server adds the noise with $SD[z] = 0.1$.

6.5.2. The Number of Users in Each Query. The adversary gets a lower value of eed when the number of users in each query decreases. As shown in Figures 5 and 6, when

changing k from 200 to 20 (in the multiplicative noise case), eed_1 decreases from 1.8 mmHg to 0.19 mmHg and eed_2 decreases from 251 m to 29.5 m. It is the server's responsibility to take measures to limit the minimum number of users in each query.

6.5.3. The Number of Queries. The more queries the adversary requests, the lower the eed is. As shown in Figures 7 and 8, eed_1 decreases from 1.4 mmHg to 0.44 mmHg and eed_2 decreases from 211 m to 71.7 m when n_{query} increases from 200 to 2,000 (in the multiplicative noise case). The server has to limit the number of queries on a specified user set requested by each client to mitigate the risk of privacy breaches.

6.6. Correctness of the Theoretical Estimation. The error rate of our theoretical estimations increases when the magnitude of the noise gets larger, as shown in Figures 3 and 4. This comes from the assumption in our theoretical analysis that

FIGURE 7: Impact of the number of queries (diaBP dataset).

(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 8: Continued.

(g)

(h)

FIGURE 8: Impact of the number of queries (check-in dataset).

(a)

(b)

(c)

(d)

FIGURE 9: Continued.

(e)

(f)

FIGURE 9: Impact of the data of the target user (diaBP dataset).

(a)

(b)

(c)

(d)

FIGURE 10: Continued.

FIGURE 10: Impact of the data of the target user (check-in dataset).

$|z| \ll 1$ and $|\tilde{a}| \ll a_1$. In Figure 3, the error rate of eed_1 is less than 0.01 under multiplicative noises when $SD[z] = 0.01$ for the one-dimensional case and increases to 0.06 when $SD[z] = 0.1$. In the two-dimensional case, the error rate of eed_2 increases from 0.01 to 0.12 when $SD[z]$ changes from 0.001 to 0.01, under multiplicative noises, as shown in Figure 4.

The magnitude of the noise determines the utility of the results. The server may not add a large noise (e.g., with $SD[z] > 0.1$) on the results due to the utility concern, and thus the error rates of our theoretical estimations can be controlled. The correctness of the theoretical analysis ensures the effectiveness of the privacy-preserving mechanism in Algorithm 1.

It is worth noting that the error rate of eed_1 increases when k is small, as shown in Figure 5. This comes from the error in the approximation of a_1 in equation (17), which rely on an assumption of the distribution of the target users. The accuracy of the approximation decreases when the user set is small.

As shown in Figure 11, the error rate is high when the query range is small. The reason is that when we narrow down the query range, the distribution of the queried data

limited in the query range deviates from our assumption in Section 6.1 (we assume the data in the one-dimensional case to obey a Gaussian distribution) and thus affects the theoretical results.

6.7. Utility Loss under the Privacy-Preserving Mechanism. We show the utility loss when the privacy-preserving mechanism adds noise on the query results to meet different privacy requirements. The precision of the query results determines the utility, and thus we use the standard deviation of the noise to measure the utility; i.e., the more the standard deviation of the noise is, the less utility the query results have. The privacy for the two cases is quantified as eed_1 and eed_2 as defined in equations (29) and (32). We give examples to show the minimum amount of noise to meet the desired privacy requirement in Figure 13. For the one-dimensional case, we set $n_{\text{query}} = 200$ and $k = 50$ and change eed_1 from 0 mmHg to 15 mmHg. For the two-dimensional case, we set $n_{\text{query}} = 2,000$ and $k = 50$ and change eed_2 from

FIGURE 11: Impact of the size of the query range (diaBP dataset).

FIGURE 12: Continued.

FIGURE 12: Impact of the size of the query range (check-in dataset).

FIGURE 13: The minimum amount of noise added on the query results to meet the desired privacy requirement. (a) One-dimensional case. (b) Two-dimensional case.

0 m to 1,000 m. We show the corresponding minimum amount of noise when s_0 is set to 0.3, 0.6, and 0.9, respectively.

The standard deviation of the minimum noise is in direct proportion to the desired privacy requirement, i.e., a higher privacy requirement causes the lower utility of the query results. In the one-dimensional case with $s_0 = 0.6$, the standard deviation of the minimum noise $SD[z]$ (resp., $SD[\tilde{a}]$) increases from 0.05 to 0.15 (resp., 1.81 mmHg to 5.44 mmHg), when the privacy requirement increases from 5 mmHg to 15 mmHg. In the two-dimensional case with $s_0 = 0.3$, the standard deviation $SD[z]$ (resp., $SD[\tilde{a}]$) increases from 0.008 to 0.014 (resp., 173 m to 347 m), when the privacy requirement increases from 500 m to 1,000 m.

To decrease the magnitude of the noise and thus preserve the utility of the results, one possible way is to restrict the maximum number of queries that each client can request and the minimum number of users in each query.

7. Conclusion and Future Work

The AVGD query serves as basic components of real-world applications such as business decision and medical diagnosis. The work of this paper is instructive to the service providers to enable AVGD queries while preserving the privacy of users' data. Specifically, we first propose an attack on the AVGD query in this paper. An attacker can recover a target user's data by querying with carefully selected user sets and query points. To understand the severity of such an attack and further propose the privacy-preserving mechanism, we formalize the process of the attack as a fitting problem with noisy data. We give the theoretical analysis of the uncertainty of the adversary. The results show that the severity of such an attack is mainly related to the factors such as the noise magnitude, the number of queries, and the number of users in each query. Other factors such as the data of the target user and the size of query ranges show low correlations with the severity of the attack. Based on the theoretical analysis, we design an algorithm for the privacy-preserving AVGD query. Experiments on two real-life cases show the severity of such an attack and the effectiveness of our theoretical analysis. We also evaluate the effectiveness of the service quality when enabling the privacy-preserving mechanism.

The attack proposed in this paper is somewhat idealized. Future works can focus on more practical situations. First, recall that the proposed attack is based on queries on two groups of users where the groups merely differ in one user. In the future work, the attack can be extended by selecting more than two user sets while it can be more flexible in choosing the sets. Besides, recall that an adversary can get a higher precision of estimated private data by using more query results. How to limit the number of queries in practical applications should be studied, as several attackers can collude to query on the same user sets. However, this is a nontrivial task as discussed in previous works [26].

Data Availability

All data included in this study are available upon request from the corresponding author.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The research was supported by the National Key R&D Program of China (2018YFB0803400 and 2018YFB2100300), the National Natural Science Foundation of China, under Grant nos. 61972369, 61572453, 61520106007, and 61572454, and the Fundamental Research Funds for the Central Universities, no. WK2150110009.

References

- [1] HIPAA, "Largest healthcare data breaches of 2018," 2018, <https://www.hipaajournal.com/largest-healthcare-data-breaches-of-2018/>.
- [2] Becker's Healthcare, "Healthcare breaches cost \$6.2 billion annually," 2016, <https://www.beckershospitalreview.com/healthcare-information-technology/healthcare-breaches-cost-6-2b-annually.html>.
- [3] Z. Cheng, J. Caverlee, K. Lee, and D. Z. Sui, "Exploring millions of footprints in location sharing services," in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, Barcelona, Spain, July 2011.
- [4] Z. Huo, X. Meng, and R. Zhang, "Feel free to check-in: privacy alert against hidden location inference attacks in geosns," in *International Conference on Database Systems for Advanced Applications*, pp. 377–391, Springer, Berlin, Germany, 2013.
- [5] L. Willenborg and T. De Waal, *Elements of Statistical Disclosure Control*, vol. 155, Springer Science & Business Media, Berlin, Germany, 2012.
- [6] S. Scellato, A. Noulas, R. Lambiotte, and C. Mascolo, "Socio-spatial properties of online location-based social networks," in *Proceedings of the Fifth international AAAI Conference on Weblogs and Social Media*, Barcelona, Spain, July 2011.
- [7] N. Armetatzoglou, R. Ahuja, and D. Papadias, "Geo-social ranking: functions and query processing," *The VLDB Journal*, vol. 24, no. 6, pp. 783–799, 2015.
- [8] E. Yilmaz, H. Ferhatosmanoglu, E. Ayday, and R. C. Aksoy, "Privacy-preserving aggregate queries for optimal location selection," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 2, 2017.
- [9] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in *Proceedings of the INFOCOM, 2014 Proceedings IEEE*, pp. 754–762, IEEE, Toronto, Canada, April 2014.
- [10] B. B. Mehta and U. P. Rao, "Improved l-diversity: scalable anonymization approach for privacy preserving big data publishing," *Journal of King Saud University-Computer and Information Sciences*, 2019, In press.
- [11] Y. Sei, H. Okumura, T. Takenouchi, and A. Ohsuga, "Anonymization of sensitive quasi-identifiers for L-diversity and T-closeness," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 4, 2017.
- [12] A. Gutscher, "Coordinate transformation-a solution for the privacy problem of location based services?" in *Proceedings of*

- the 20th IEEE International Parallel & Distributed Processing Symposium, p. 7, April 2006.
- [13] M. L. Yiu, C. S. Jensen, J. Möller, and H. Lu, "Design and analysis of a ranking approach to private location-based services," *ACM Transactions on Database Systems (TODS)*, vol. 36, no. 2, p. 10, 2011.
 - [14] J. H. Cheon, K. Han, A. Kim, M. Kim, and Y. Song, "Bootstrapping for approximate homomorphic encryption," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Darmstadt, Germany, pp. 360–384, May 2018.
 - [15] C. Dwork, "Differential privacy," in *Encyclopedia of Cryptography and Security*, pp. 338–340, Springer, Berlin, Germany, 2011.
 - [16] I. Dinur and K. Nissim, "Revealing information while preserving privacy," in *Proceedings of the Twenty-Second ACM SIGMOD-SIGACTSIGART Symposium on Principles of Database Systems*, ACM, San Diego, CA, USA, pp. 202–210, June 2003.
 - [17] C. Dwork and S. Yekhanin, "New efficient attacks on statistical disclosure control mechanisms," in *Proceedings of the Annual International Cryptology Conference*, Springer, Santa Barbara, CA, USA, pp. 469–480, August 2008.
 - [18] C. Dwork and K. Nissim, "Privacy-preserving datamining on vertically partitioned databases," in *Proceedings of the Annual International Cryptology Conference*, Springer, Santa Barbara, CA, USA, pp. 528–544, August 2004.
 - [19] D.-N. Yang, C.-Y. Shen, W.-C. Lee, and M.-S. Chen, "On socio-spatial group query for location-based social networks," in *Proceedings of the 18th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining*, ACM, Beijing, China, pp. 949–957, August 2012.
 - [20] D. Zhang, Y. Du, T. Xia, and Y. Tao, "Progressive computation of the min-dist optimal-location query," in *Proceedings of the 32nd international Conference on Very Large Data Bases*, VLDB Endowment, Seoul, Korea, pp. 643–654, September 2006.
 - [21] J. Qi, R. Zhang, Y. Wang, A. Y. Xue, G. Yu, and L. Kulik, "The min-dist location selection and facility replacement queries," *World Wide Web*, vol. 17, no. 6, pp. 1261–1293, 2014.
 - [22] L. Ou, H. Yin, Z. Qin, S. Xiao, G. Yang, and Y. Hu, "An efficient and privacy-preserving multiuser cloud-based lbs query scheme," *Security and Communication Networks*, vol. 2018, Article ID 4724815, 11 pages, 2018.
 - [23] A. Rohilla, M. Khurana, M. Khurana, and L. Singh, "Location privacy using homomorphic encryption over cloud," *International Journal of Computer Network and Information Security*, vol. 9, no. 8, pp. 32–40, 2017.
 - [24] A. Gutub, N. Al-Juaied, and E. Khan, "Counting-based secret sharing technique for multimedia applications," *Multimedia Tools and Applications*, vol. 78, no. 5, pp. 5591–5619, 2019.
 - [25] S. P. Kasiviswanathan, M. Rudelson, A. Smith, and J. Ullman, "The price of privately releasing contingency tables and the spectra of random matrices with correlated rows," in *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, ACM, Cambridge, MA, USA, pp. 775–784, June 2010.
 - [26] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical privacy: the sulq framework," in *Proceedings of the Twenty-Fourth ACM SIGMODSIGACT-SIGART Symposium on Principles of Database Systems*, ACM, Baltimore, MD, USA, pp. 128–138, June 2005.
 - [27] J. E. Sader, B. D. Hughes, J. A. Sanelli, and E. J. Bieske, "Effect of multiplicative noise on least-squares parameter estimation with applications to the atomic force microscope," *Review of Scientific Instruments*, vol. 83, no. 5, Article ID 055106, 2012.
 - [28] D. Yang, D. Zhang, V. W. Zheng, and Z. Yu, "Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 1, pp. 129–142, 2015.
 - [29] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *Proceedings of the 2011 IEEE Symposium on Security and Privacy*, IEEE, Oakland, CA, USA, pp. 247–262, May 2011.
 - [30] B. Hoh and M. Gruteser, "Protecting location privacy through path confusion," in *Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, pp. 194–205, IEEE, Athens, Greece, September 2005.
 - [31] R. P. Kanwal and K. C. Liu, "A taylor expansion approach for solving integral equations," *International Journal of Mathematical Education in Science and Technology*, vol. 20, no. 3, pp. 411–414, 1989.
 - [32] Framingham, "Framingham heart study," 2020, <https://www.framinghamheartstudy.org/>.
 - [33] Foursquare, "About Foursquare," 2020, <https://foursquare.com/>.
 - [34] T. F. Coleman and Y. Li, "An interior trust region approach for nonlinear minimization subject to bounds," *SIAM Journal on Optimization*, vol. 6, no. 2, pp. 418–445, 1996.