Utility-Improved Key-Value Data Collection with Local Differential Privacy for Mobile Devices

TONG Ze, DENG Bowen, ZHENG Lele, ZHANG Tao

(School of Computer Science and Technology, Xidian University, Xi´an 710071, China)

DOI: 10.12142/ZTECOM.202204003

https://kns.cnki.net/kcms/detail/34.1294.TN.20221125.1108.002.html
published online November 25, 2022

Manuscript received: 2022-09-09

Abstract: The structure of key-value data is a typical data structure generated by mobile devices. The collection and analysis of the data from mobile devices are critical for service providers to improve service quality. Nevertheless, collecting raw data, which may contain various personal information, would lead to serious personal privacy leaks. Local differential privacy (LDP) has been proposed to protect privacy on the device side so that the server cannot obtain the raw data. However, existing mechanisms assume that all keys are equally sensitive, which cannot produce high-precision statistical results. A utility-improved data collection framework with LDP for key-value formed mobile data is proposed to solve this issue. More specifically, we divide the key-value data into sensitive and non-sensitive parts and only provide an LDP-equivalent privacy guarantee for sensitive keys and all values. We instantiate our framework by using a utility-improved key value-unary encoding (UKV-UE) mechanism based on unary encoding, with which our framework can work effectively for a large key domain. We then validate our mechanism which provides better utility and is suitable for mobile devices by evaluating it in two real datasets. Finally, some possible future research directions are envisioned.

Keywords: key-value data; local differential privacy; mobile devices; privacy-preserving data collection

Citation (IEEE Format): Z. Tong, B. W. Deng, L. L. Zheng, et al., "Utility-improved key-value data collection with local differential privacy for mobile devices," *ZTE Communications*, vol. 20, no. 4, pp. 15 – 21, Dec. 2022. doi: 10.12142/ZTECOM.202204003.

1 Introduction

With the development of mobile communication technologies, service providers are more willing to collect data from mobile devices to enhance the service experience for users. As a classical data structure, key-value data are widespread in practical mobile applications[1 – 2]. The structure of key-value data is a hybrid data structure, where the key is the identifier of data and the value is the content of data. The following three examples show its potential applications:

1) Mobile devices (such as wearable devices, smartphones, tablets, etc.) generate a large number of data, the majority of which are in a key-value format, i.e., $\langle device\_id, device\_value \rangle$ or $\langle timestamp, device\_value \rangle$. These data could show the usage habits of users on a specific device or during a particular period, which can help data collectors provide a personalized experience for the user. For example, the service center collects $\langle device\_id, sleep\ duration \rangle$ from the user's smart bracelet to remind the user to rest properly at a suitable time[3].

2) Software vendors, such as Android and iOS, collect users' data to enhance the users' experience, i. e. $\langle app\_name, user\ rating \rangle$ or $\langle app\_name, length\ of\ visit \rangle$, in a key-value format, where the key is the name of an APP, and the value is the length of time or a score to access the APP. These data could show the users' experience with a particular application, which can help software vendors study future product improvements. For example, software vendors provide users with personalized recommendations by collecting their specific interest[4].

3) Advertisers are interested in knowing whether the video advertisements they place on mobile devices appeal to potential customers[5]. Therefore, they are willing to collect advertisement ratings from users in the form of key-value, where the key is the ID of the advertisement, and the value is the number of minutes that users watch the advertisement.

However, the key-value data involves a lot of personal information, thus users may be reluctant to upload data from their mobile devices. To address the privacy-preserving data collection issue, some researchers proposed local differential privacy[6] to obfuscate local information in the data collection phase. Because of its decentralization and strict mathematical proof, it has been adopted by mainstream systems such as MacOS[7] and Windows[8] to collect data. In addition, local differential privacy (LDP) reduces the communication costs of large-scale computing and the frequent interaction with the data center, making it well-suited for mobile devices with limited resources and low computing power.

Recently, there has been extensive research on key-value data collections with LDP. YE et al. [9] first proposed PrivKVM to protect key-value data using synchronized key and value perturbation protocols. It adopted one iteration to obtain frequency estimation and several iterations to achieve an approximately unbiased mean estimation. The result of the last iteration is sent to the next iteration as input. However, it requires all users to be online in all the iterations, which is difficult to achieve in practical scenarios. Moreover, PrivKVM may lead to a high estimation error when the key domain is large. SUN et al. [2] proposed a series of LDP mechanisms based on PrivKVM and introduced conditional analysis for key-value data analysis. However, the mean estimation obtained by SUN et al. is biased. Subsequently, GU et al.[10] proposed a private correlated key-value (PCKV) data collection mechanism, which adopts the padding-and-sampling mechanism to solve the large key domain problem of previous work[9]. Moreover, a budget composition theorem for the relevant perturbation mechanism is further given to enhance the data utility using privacy budget relaxation. However, according to the definition of LDP, we cannot distinguish whether the output key is genuine. Because the virtual values significantly reduce the aggregation accuracy, the aforementioned mean estimation mechanisms perform poorly in the case of a small privacy budget. Therefore, there is a requirement to enhance the utility of key-value data collection under LDP.

Moreover, the mechanisms aforementioned regard all data as equally sensitive and thus provide excessive protection for some data and leave much room for improving data utility. In real-world scenarios, there is quite a lot of non-sensitive data. For example, when the server collects application names and ratings from cell phones, attackers cannot infer users' privacy preferences even if they know that the user logs in WeChat, which is a social APP that has a huge user base. Therefore, using WeChat provides non-sensitive data for users. In contrast, using some minority applications provides sensitive data. Based on this idea, MURAKAMI et al. [11] proposed the concept of utility-optimized LDP (ULDP), which only requires LDP protection for sensitive data to reduce the frequency estimation error. Nevertheless, ULDP is only suitable for frequency estimation, thus the accuracy of data collection under the privacy protection for key-value data needs further enhancements.

To address these issues, we propose a new framework for mobile devices called the utility-improved key value (UKV) data collection with LDP. In UKV, mobile devices take different perturbations based on whether the data are sensitive or not to achieve a balance between privacy and utility. We then introduce an initial implementation of the UKV framework and verify its performance in terms of data utility using public datasets.

The remainder of the paper is organized as follows. The overview and benefits of the UKV framework are introduced in Section 2. Some key challenges are presented in Section 3. In Section 4, we describe a case study of an initial implementation of the UKV framework for mobile devices. The performance of our mechanisms is evaluated in Section 5, and some possible future directions are given in Section 6. Finally, we conclude the paper in Section 7.

2 Overview and Benefits

In this section, we briefly introduce data collection under local differential privacy. Then we describe the UKV framework and its benefits for data protection.

2.1 Data Collection Under Local Differential Privacy

Data collection is an important means of obtaining data from mobile devices. By collecting and analyzing users' data, data collectors can mine users' characteristics (such as living habits and health status) and thus formulate more appropriate development strategies. However, users' data often contains a large amount of personal information. Collecting raw data may lead to serious personal privacy leaks, which not only harms privacy leakers but also brings a series of legal risks and economic losses to data collectors. Therefore, this issue needs to be solved urgently.

Differential privacy[12] provides a feasible solution to the problem of personal privacy leakage due to its characteristic of being plausible and deniable. It provides strictly provable privacy protection without relying on the background knowledge possessed by the attacker. LDP is one of the differential privacy technologies that specifically address the problem of personal privacy leakage during data collection. Unlike central differential privacy, which assumes the existence of a trusted data collector with access to the user's raw data, LDP does not require any qualification on the credibility of the data collector. In particular, LDP requires each user to locally perturb the raw data with a local perturbation mechanism before sending it to the data collector. Therefore, the data security of the users is guaranteed. Because of this unique advantage, LDP has been widely adopted in practice. A successful case is RAPPOR[13] on Google Chrome, which enables Google to collect users' browsing information while protecting user privacy.

A basic LDP mechanism is Generalized Randomized Response (GRR)[14]. The main idea of GRR is to set the output range to be the same as the input range, with a certain probability of providing a "fake" response while maximizing the likelihood of providing a "true" response. Specifically, each user perturbs x to itself with a large probability p, and perturbs x to other data with a small probability q. However, the utility of GRR drops rapidly when the data domain is large. UE[15] solves this problem. UE first encodes the input data as a one-hot d-dimensional vector with only the bit corresponding to the data set to 1, where d is the size of input domain. Then each bit is perturbed independently. Here, each user retains (only) input 1 with large probability p, and perturbs each 0 to 1 with probability q. Our work is based on the above scheme and achieves secure data collection adapted to mobile devices.

2.2 Overview

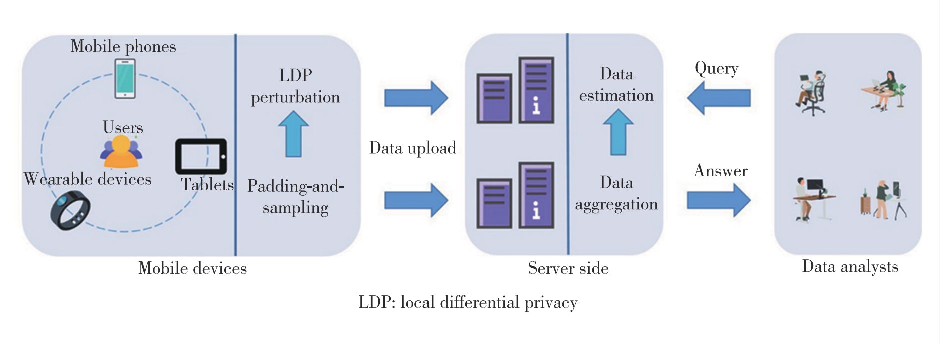
Fig. 1 shows the overview of our framework, which contains three parts: mobile devices, the server side, and data analysts. And, we will describe each part of our framework in detail.

•Mobile devices. Mobile devices are individual users who own personal data. They can not only generate and collect data of users but also perturb the raw data with local differential privacy mechanisms to protect information privacy.

•Server side. The server, which has a large number of computing and storage resources, is responsible for collecting the data sent by mobile devices, and aggregating and estimating the data. Finally, the server releases the data and its corresponding estimations. In this paper, we assume that the server is "semi-honest". Here, "semi-honest" means that the server honestly executes the data collection protocol while potentially leaking the user's historical data to attackers.

•Data analysts. Data analysts are the actual users of the data. The data analyst submits a query request to the server and gets the noise-added results. These analysts may be ordinary users or malicious attackers.

We briefly describe the data flow in a UKV framework to understand the data processing procedure. The raw data are generated by the mobile device and locally perturbed by UKV. Then, the mobile device sends it to the server side. In addition, the key-value data are divided into two categories: sensitive data and non-sensitive data. In the data perturbation phase, UKV divides the privacy budget ε into two parts, namely ε_1 and ε_2, where ε_1 is used for perturbation of key and ε_2 is used for perturbation of value. For sensitive key-value data, the key consumes all privacy budget ε_1; for non-sensitive key-value data, the key does not consume any privacy budget. Furthermore, UKV consumes the privacy budget of ε_2 to perturb the value for two types of key-value data. Finally, the server releases the estimated results to the data analyst.



LDP: local differential privacy

▲Figure 1. Overview of the proposed framework

2.3 Benefits

By applying the UKV framework, users perturb the raw data before sending key-value data to the server. In addition, the UKV framework also provides the following benefits.

• Privacy protection and efficient utility of data. In data collection, each user sends the noise-added data to the server. Then the server aggregates and analyzes the data where the frequency and mean estimation are important data analysis components. UKV can maintain high efficiency with a low privacy budget. As the number of non-sensitive keys increases, the data utility improves rapidly.

• No trusted third party is required. LDP transfers the part that adds noise to the raw data to local devices so that the third-party data collectors cannot get the raw data; thus it avoids the risk of privacy leakage by third parties.

3 Key Challenges

In order to take full advantage of the UKV framework for mobile devices, we still face challenges in the implementation of the proposed framework, which seriously hinders the booming development of related applications.

1) Individuals have different privacy needs for data. The difference between sensitive and non-sensitive data can vary from one user to another (e.g., some people even want to keep the name of the APP that they use and the scores of the movie they give private). Moreover, we concentrate on a situation in which users can easily choose, no matter it is sensitive or not. Nevertheless, there is also a situation in which the user knows nothing about the sensitive data type. For the latter case, the improvement of our UKV framework for data utility is greatly reduced. Therefore, how to divide sensitive data and non-sensitive data is crucial.

2) Association of sensitive data with non-sensitive data. First, we assume that each user sends a key-value data pair and each user's data are irrelevant. This makes sense for most personal data (e.g., application ratings). Yet, for certain types of personal data (e.g., flu status[16]), users may be extremely influenced by other users. Moreover, when users send more than one pair of data, sensitive and non-sensitive data may also be correlated with each other, which means that non-sensitive data release may lead to sensitive data leakage[17]. Therefore, designing a scheme for sending multiple data pairs per user is an important and challenging problem.

3) Lightweight. In practical applications, the communication band-

width cost between the mobile device and the server is proportionate to the domain size of the key. The time complexity of UKV proposed in this paper is $O(d)$. When the key domain is too large, the communication cost increases dramatically, which is unacceptable in many practical applications. Moreover, because of the limited computing resources of mobile devices, they lack the ability to perform complex computing tasks. Thus, designing a lightweight privacy-preserving algorithm for mobile devices is necessary.

4) Selection of parameters. The Padding-and-Sampling protocol is used in the UKV framework, where the padding length l needs to be set in advance. In theory, it should be set based on the data distribution. A small l will reduce the frequency estimation for the key, while a large l will increase the quantity of virtual key-value data, leading to a larger estimation error. However, the best selection of l is not possible because the purpose of UKV is to learn the data distribution. In addition, the choice of privacy budget is essential for balancing privacy and utility. Therefore, it is crucial to choose parameters to achieve near-optimal efficiency.

4 Case Study

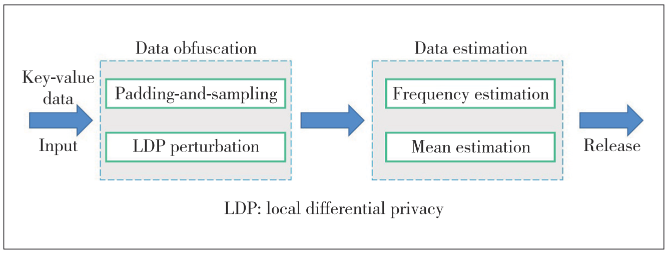
In this section, we provide a case study to introduce the initial implementation of the UKV framework for mobile device data collection.

Fig. 2 shows the implementation of the UKV framework for mobile devices. It consists of two parts:

1) The mobile device perturbs key-value data with LDP to provide provable privacy guarantees.

2) The server estimates data to generate usable data from the collected key-value data for analysis.

The two parts are combined to improve the accuracy of server data analysis while protecting the privacy of key-value data. We then describe the details of the two parts in the following sections.



▲Figure 2. Implementation of the utility-improved key value framework

4.1 Mobile Device

- Padding-and-sampling[18]: Each user samples one datum from possessed key-value data instead of sampling one datum from the domain of all key-value data. In order to make all samplings rate the same, each user first adds different random dummy data to possessed key-value data until he has l key-value data.

- Perturbation: The general overview of the perturbation method includes the key input domain, key output domain, and flip probability. The input domain has sensitive key $k_s \in K_S$ and non-sensitive key $k_n \in K_N$, where K_S and K_N are the sets of sensitive and non-sensitive keys, respectively. In the output domain, $k_r \in K_S \backslash k$ denotes the rest sensitive keys except k, where k may not belong to K_S. When the user inputs k_s to UKV, her output includes k_s with p_{ss} probability and k_r with p_{sr} probability, where the values of p_{ss} and p_{sr} are related to the privacy budget ε_1. When the user inputs k_n, her output includes k_r with the p_{nr} probability and k_n with the p_{nn} probability, where p_{nr} is equal to p_{sr}.

Combining the thought with UE[15], we instantiate a mechanism named UKV-UE under the UKV framework. For the data obtained by sampling, UKV-UE first transforms the input data into a one-dimensional array, e. g., the second data is $\langle id\_2, 0.9 \rangle$, which we transform into a vector $(\langle 0,0 \rangle, \langle 1,1 \rangle, \langle 0,0 \rangle, \cdots, \langle 0,0 \rangle)$ with vector length l, and perturbs each bit independently. Each array is divided into two parts: sensitive and non-sensitive bits. Here, we use k to denote the k-th bit of the array and i to denote the rest of the bits of the array. According to the transformation, we know that the k-th bit is $\langle 1, v \rangle$ and the rest bits are $\langle 0, 0 \rangle$. For the results of perturbation: 1) when k belongs to sensitive bits, -1 (or 1) indicates the presence of the key, where -1 (or 1) is obtained by a stochastic rounding (SR) mechanism[19] (the SR mechanism is to perturb the value to -1 or 1 with different probabilities depending on the input) and 0 indicates the absence of the key; 2) when k belongs to non-sensitive bits, v' obtained by perturbing v with the hybrid mechanism (HM)[20] indicates the presence of the key (HM output domain is boundedly continuous) and the specified out-of-domain element M indicates the absence of the key.

4.2 Server-Side

Data estimation: The server collects the data uploaded by users. For the sensitive key, the server computes the counts of 1 and -1 that support key k from all the data sent by users, denoted by n_0 and n_1, respectively. Then we could calculate the frequency estimation $\hat{f}_k$ and the corresponding mean estimation $\hat{m}_k$ by

$$\hat{f}_k = \frac{(n_0 + n_1)/n - p_{ss}}{p_{ss} - p_{sn}},$$

$$\hat{m}_k = \frac{l(n_0 - n_1)(e^{\varepsilon_2})}{n(e^{\varepsilon_2} - 1)\hat{f}_k p_{ss}}, \tag{1}$$

where n is the number of the users.

For the non-sensitive key, the server computes the number of v that supports key k from all the data sent by users, denoted by n_2. Then we could calculate the frequency estimation $\hat{f}_k$ and the corresponding mean estimation $\hat{m}_k$ by

$$\hat{f}_k = \frac{n_2}{n \cdot p_{nn}},$$

$$\hat{m}_k = \frac{\sum_{\langle k,v \rangle \in P} v}{n_2}, \tag{2}$$

where P is the set of perturbed values sent by the users.

In summary, UKV improves the data utility by slackening privacy protection for non-sensitive data.

5 Performance Evaluation

In this section, we evaluate the privacy and utility assurance performance of UKV for data collection in public datasets.

Datasets: In this paper, we use the e-commerce (Ec) dataset[21] and the clothing (Cl) dataset[22] to evaluate the performance of UKV on privacy protection and utility assurance. The Ec dataset includes 23 486 key-value pairs and 1 206 category keys for a total of 23 486 users. The Cl dataset includes 192 544 key-value pairs and 5 850 types of keys, with a total of 105 508 users.

To demonstrate the advantages of the UKV framework, we compare it with the most advanced key-value data collection mechanism PCKV.

Evaluation metrics: We evaluate the frequency and mean estimations by comparing the averaged mean square error (MSE) among non-sensitive keys:

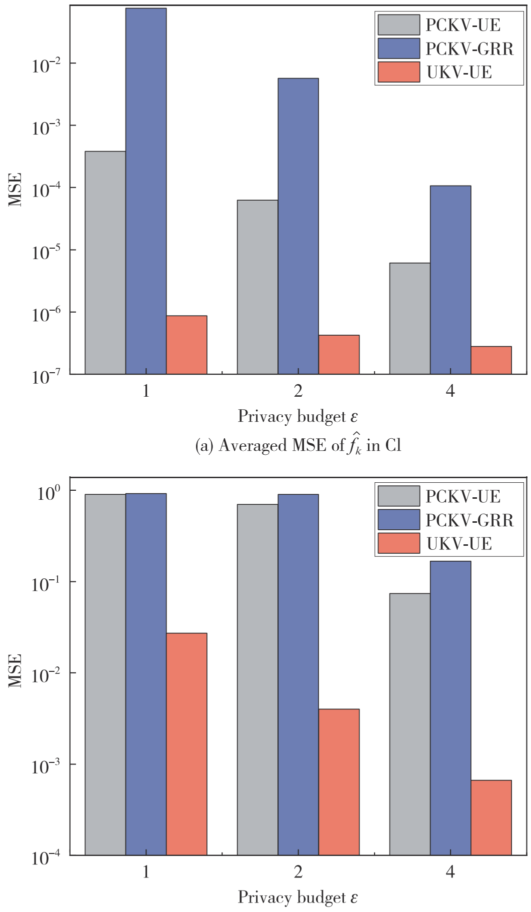
$$\mathrm{MSE}_{\mathrm{freq}} = \frac{1}{|K_N|} \sum_{k \in K_N} \left(\hat{f}_k - f_k \right)^2,$$

$$\mathrm{MSE}_{\mathrm{mean}} = \frac{1}{|K_N|} \sum_{k \in K_N} \left(\hat{m}_k - m_k \right)^2, \tag{3}$$

where K_N is the domain of non-sensitive keys, $\hat{f}_k$ and $\hat{m}_k$ are the frequency and mean estimations of the key-value data, and f_k and m_k are the actual frequency and mean values of the key-value data.

We use the ten most frequent keys as non-sensitive keys, because the frequency of non-sensitive keys is usually higher in practice.

Figs. 3 and 4 show the MSE of non-sensitive keys in two real-world datasets, from which the effect of privacy budget on data utility can be observed. We double the privacy budgets in our experiments. The larger the privacy budget, the lower the mean square error and the higher the data utility. The MSE of the Cl dataset does not change much compared with the results of the Ec dataset because all algorithms benefit from the number of users, which makes up for the effect of the large key domain. As shown in Figs. 3(a) and 4(a), our UKV-UE mechanism performs the best as it does not decrease the privacy budget of frequency estimation while discriminating the key sensitivity in UKV. The theory of dividing the sensitivity to decrease the frequency estimation errors is detailed in Ref. [12].



(a) Averaged MSE of $\hat{f}_k$ in Cl

(b) Averaged MSE of $\hat{m}_k$ in Cl

Cl: clothing
GRR: Generalized Randomized Response
MSE: mean square error
PCKV: Private Correlated Key-Value
UE: unary encoding
UKV: utility-improved key value

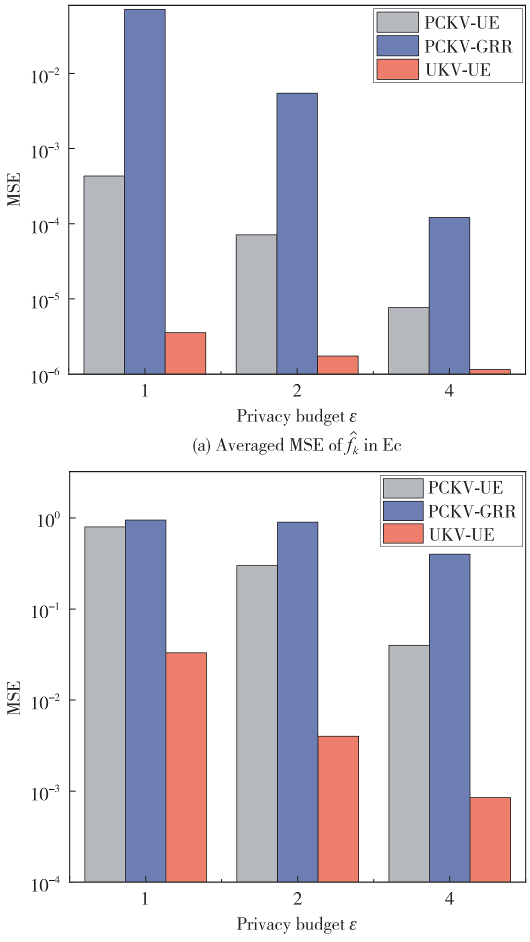
▲Figure. 3. MSE of Cl dataset

Similarly, in Figs. 3(b) and 4(b), the UKV-UE mechanism performs well for any privacy budget about the mean estimation. In the case of a small privacy budget, only the UKV-UE achieves higher accuracy.

6 Future Directions

This work, key-value data collection with LDP for mobile devices, still needs further research to advance its development. In this section, we envision some possible future directions.

1) Statistical analysis of key-value data for mobile devices. To the best of our knowledge, the current work is limited to frequency estimation and mean estimation of key-value data. In



(a) Averaged MSE of $\hat{f}_k$ in Ec

(b) Averaged MSE of $\hat{m}_k$ in Ec

Ec: e-commerce
GRR: Generalized Randomized Response
MSE: mean square error
PCKV: private correlated key-value
UE: unary encoding
UKV: utility-improved key value

▲Figure 4. MSE of Ec dataset

contrast, other applications of key-value data are less explored (e.g., maximum-minimum estimation of key-value data). Therefore, other aggregation statistics of key-value data for mobile devices are a direction worthy of attention.

2) Machine learning on mobile devices. In a distributed machine learning system on mobile devices, the mobile devices collect data and send it to the server. Then, the server divides the subsets of data items according to certain rules and finally distributes the subsets to each device for training. Currently, only a few mobile machine learning frameworks support key-value data formats to submit training data, like searching English words in dictionaries (the dictionary data structure is in key-value format, where the key is the alphabet and the value is their sequence number). Therefore, exploring more training frameworks that support key-value data formats and adding LDP protection to them is quite worthwhile.

3) High-dimensional key-value data in mobile devices. Most of the current major differential privacy protection frameworks are for two-dimensional data sets. However, there are a lot of complex high-dimensional data in mobile devices, and it is necessary to protect them using local differential privacy techniques. Moreover, shifting data protection from two dimensions to multiple dimensions will inevitably bring more challenges, like dimensional disasters. In short, designing a differential privacy protection framework for mobile devices that can be extended to multi-dimensional data protection is an important challenge for data analysis work.

4) Mobile real-time data release. With the need for some particular scenarios (such as a health code and a nucleic acid test), people have an increasing demand for real-time query response and data updates. Real-time data release has high requirements for the stability of data transmission. However, the data transmission stability of mobile devices is doubtful, which may cause frequent dropouts for users. In addition, problems such as repeated data release and dynamic data update significantly increase the risk of privacy leakage in the real-time data release. Therefore, the privacy protection for real-time data release of mobile devices deserves much attention.

5) Preventing poisoning attacks by mobile devices. Poisoning attacks against key-value data aim to reduce data availability by sending carefully crafted data from some fake users to the server while changing the frequency and mean value of the target key chosen by the attacker[23]. For example, an attacker successfully changed a road segment in Google Maps from "clear" to "congested" using 99 mobile phones. Existing defense methods are effective in some cases but ineffective in others. Therefore, researching methods to defend against poisoning attacks from mobile devices is a worthwhile endeavor.

7 Conclusions

We researched the utility improvement of key-value data collection for mobile devices and proposed a novel framework, UKV, which has improved the data utility by providing LDP privacy protection for sensitive key-value data only. We also introduced the main challenges that hindered key-value data collections from maximizing their benefits. Then, we introduced an initial implementation of the UKV framework and validated the excellent utility of our mechanism on two real datasets. Finally, we envisioned some possible future directions to attract more research in this area.

References

[1] ZENG J A, PLALE B. KVLight: A lightweight key-value store for distributed access in cloud [C]//The 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid). IEEE, 2016: 473 – 482. DOI: 10.1109/CCGrid.2016.55

[2] SUN L, ZHAO J, YE X, et al. Conditional analysis for key-value data with local differential privacy [EB/OL]. (2019-07-11) [2022-09-20]. https://arxiv.org/abs/

1907.05014v1

[3] ANGELINI L, CAON M, CARRINO S, et al. Designing a desirable smart bracelet for older adults [C]//Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication. ACM, 2013: 425 – 434. DOI: 10.1145/2494091.2495974

[4] MENG J K, ZHENG Z B, TAO G H, et al. User-specific rating prediction for mobile applications via weight-based matrix factorization [C]//Proceedings of 2016 IEEE International Conference on Web Services. IEEE, 2016: 728 – 731. DOI: 10.1109/ICWS.2016.104

[5] BALAKRISHNAN S, CHOPRA S, APPLEGATE D, et al. Computational television advertising [C]//Proceedings of 2012 IEEE 12th International Conference on Data Mining. IEEE, 2012: 71 – 80. DOI: 10.1109/ICDM.2012.129

[6] DUCHI J C, JORDAN M I, WAINWRIGHT M J. Local privacy and statistical minimax rates [C]//Proceedings of 2013 IEEE 54th Annual Symposium on Foundations of Computer Science. IEEE, 2013: 429 – 438. DOI: 10.1109/FOCS.2013.53

[7] TANG J, KOROLOVA A, BAI X, et al. Privacy loss in apple's implementation of differential privacy on macOS 10.12. [EB/OL]. [2022-09-20]. https://www.researchgate. net/publication/319622426_Privacy_Loss_in_Apple's_Implementation_of_Differential_Privacy_on_MacOS_1012

[8] DING B, KULKARNI J, YEKHANIN S. Collecting telemetry data privately [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems, IEEE, 2017: 3574 – 3583

[9] YE Q Q, HU H B, MENG X F, et al. PrivKV: key-value data collection with local differential privacy [C]//Proceedings of 2019 IEEE Symposium on Security and Privacy (SP). IEEE, 2019: 317 – 331. DOI: 10.1109/sp.2019.00018

[10] GU X, LI M, CHENG Y, et al. PCKV: locally differentially private correlated key-value data collection with optimized utility [EB/OL]. (2019-11-28) [2022-09-20]. https://arxiv.org/abs/1911.12834

[11] MURAKAMI T, KAWAMOTO Y. Utility-optimized local differential privacy mechanisms for distribution estimation [C]//Proceedings of the 28th USENIX Conference on Security Symposium. SEC, 2019: 1877 – 1894

[12] DWORK C, MCSHERRY F, NISSIM K, et al. Calibrating noise to sensitivity in private data analysis [J]. Theory of cryptography, 2006: 265 – 284. DOI: 10.1007/11681878_14

[13] ERLINGSSON Ú, PIHUR V, KOROLOVA A. RAPPOR: randomized aggregatable privacy-preserving ordinal response [C]//The 2014 ACM SIGSAC Conference on Computer and Communications Security. CCS, 2014: 1054 – 1067. DOI: 10.1145/2660267.2660348

[14] WANG T H, LI N H, JHA S. Locally differentially private heavy hitter identification [J]. IEEE transactions on dependable and secure computing, 2021, 18 (2): 982 – 993. DOI: 10.1109/TDSC.2019.2927695

[15] WANG T H, BLOCKI J, LI N H, et al. Locally differentially private protocols for frequency estimation [C]//The 26th USENIX Conference on Security Symposium. USENIX, 2017: 729 – 745

[16] SONG S, WANG Y Z, CHAUDHURI K. Pufferfish privacy mechanisms for correlated data [C]//Proceedings of the 2017 ACM International Conference on Management of Data. ACM, 2017: 1291 – 1306. DOI: 10.1145/3035918.3064025

[17] NARAYANAN A, SHMATIKOV V. Myths and fallacies of "personally identifiable information" [J]. Communications of the ACM, 2010, 53(6): 24 – 26. DOI: 10.1145/1743546.1743558

[18] WANG T H, LI N H, JHA S. Locally differentially private frequent itemset mining [C]//Proceedings of 2018 IEEE Symposium on Security and Privacy. IEEE, 2018: 127 – 143. DOI: 10.1109/SP.2018.00035

[19] DUCHI J C, JORDAN M I, WAINWRIGHT M J. Minimax optimal procedures for locally private estimation [J]. Journal of the American statistical association, 2018, 113(521): 182 – 201. DOI: 10.1080/01621459.2017.1389735

[20] WANG N, XIAO X K, YANG Y, et al. Collecting and analyzing multidimensional data with local differential privacy [C]//Proceedings of 2019 IEEE 35th International Conference on Data Engineering. IEEE, 2019: 638 – 649. DOI: 10.1109/ICDE.2019.00063

[21] KAGGLE. Ecommerce rating dataset [EB/OL]. [2022-09-20]. https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews

[22] KAGGLE. Clothing fit and rating dataset [EB/OL]. [2022-09-20]. https://www.kaggle.com/rmisra/clothing-fit-dataset-for-size-recommendation

[23] WU Y J, CAO X Y, JIA J Y, et al. Poisoning attacks to local differential privacy protocols for key-value data [EB/OL]. (2021-11-22) [2022-09-20]. https://arxiv.org/abs/2111.11534

Biographies

TONG Ze received his BS degree from Chang'an University, China in 2019, where he is currently pursuing the MS degree with the School of Computer Science and Technology, Xidian University, China. His research interests include differential privacy and network security.

DENG Bowen received his BS degree from Xidian University, China in 2020, where he is currently pursuing the MS degree with the School of Computer Science and Technology, Xidian University. His research interests include differential privacy and social networks.

ZHENG Lele received his BS degree from Xidian University, China in 2018, where he is currently pursuing the PhD degree with the School of Computer Science and Technology, Xidian University. His research interests include differential privacy and the IoT data security.

ZHANG Tao (taozhang@xidian.edu.cn) received his MS and PhD degrees in computer science from Xidian University, China in 2011 and 2015, respectively. He is currently an associate professor with the School of Computer Science and Technology, Xidian University. His research interests include network security and privacy protection.