SCIENCE CHINA
Information Sciences

• LETTER •

November 2021, Vol. 64 219101:1–219101:3
https://doi.org/10.1007/s11432-018-9899-8

Locally differentially private distributed algorithms for set intersection and union

Qiao XUE[1], Youwen ZHU[1,3*], Jian WANG[1], Xingxin LI[1] & Ji ZHANG[2]

[1]*College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing* 210016*, China;*
[2]*Faculty of Health, Engineering and Science, University of Southern Queensland, Toowoomba* 4350*, Australia;*
[3]*Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing* 210023*, China*

Received 5 November 2018/Revised 4 February 2019/Accepted 21 May 2019/Published online 13 May 2021

Citation Xue Q, Zhu Y W, Wang J, et al. Locally differentially private distributed algorithms for set intersection and union. Sci China Inf Sci, 2021, 64(11): 219101, https://doi.org/10.1007/s11432-018-9899-8

Dear editor,

Privacy-preserving distributed set intersection and union (PPSI, PPSU) have received much attention in recent years because of their wide applications. Most of existing solutions [1, 2] utilize secure multiparty computation protocols (SMCP) [3, 4] to settle the problem, but the SMCP methods are expensive in computation and communication. Even worse, most SMCP methods hardly continue to work if some participants disconnect.

To address the drawbacks of the SMCP schemes, we propose novel solutions for privacy-preserving set operations. The key technique in the novel solutions is local differential privacy (LDP) [5]. LDP is a variant model of differential privacy (DP) [6] which is a state-of-the-art privacy definition that is independent of the adversary's background knowledge. The algorithm with LDP has low computation and communication costs. Besides, the mechanism satisfying LDP guarantees that the output cannot be impacted badly by any change of data in the input, which means that it is hard for an adversary to infer the data from the outputs, and so data are protected. Usually, in the system under LDP, each data owner processes his data independently and does not need to cooperate with other participants like the SMCP schemes, thus the system under LDP is more robust.

System model. We consider a distributed model consisting of n data owners (users) and one collector. Each user u_i $(i = 1, 2, \ldots, n)$ possesses a private dataset D_i which is a subset of a universal set U. The collector is interested in obtaining union and intersection of private sets of users. Considering that the collector may be an adversary, we design PPSI/PPSU protocols with LDP to provide a privacy guarantee for uses. About each user's private set, there are two cases: (1) it is a normal set which is a collection of distinct items, such as $\{1, 2, 3\}$; and (2) it is a multiset which allows multiple instances for each item, such as $\{1, 1, 2, 2, 3\}$. And, items in each user's set are supposed to be independent. The solutions on PPSI (PPSU) for normal sets have been elaborated in [7]. Hence, we mainly focus on PPSI and PPSU mechanisms under LDP for multisets in this study.

Mechanisms design. Each user u_i has a private multiset D_i, $D_i \subseteq U$, where U is a public universal multiset:

$$U = \{\underbrace{1, \ldots, 1}_{\max}, \underbrace{2, \ldots, 2}_{\max}, \ldots, \underbrace{l, \ldots, l}_{\max}\}.$$

U contains l distinct items and max is the maximal multiplicity (the number of times an item occurs in a multiset) which is allowed in a multiset. Let $f_j(D_i)$ denote the multiplicity of the j-th different item of U in D_i. Then, for each $j \in \{1, \ldots, l\}$, it has

$$\begin{aligned} f_j\big(\cap_{i=1}^n D_i\big) &= \min(f_j(D_1), \ldots, f_j(D_n)), \\ f_j\big(\cup_{i=1}^n D_i\big) &= \max(f_j(D_1), \ldots, f_j(D_n)). \end{aligned} \tag{1}$$

Accordingly, D_i's complement $\overline{D}_i$ satisfies $f_j(\overline{D}_i) = \max - f_j(D_i)$. We also denote $\overline{D}_i = U - D_i$.

We first introduce the locally differentially private method for the intersection operation (DMSI-LDP). The protocol on the perturbation of each user's multiset is presented as follows.

Encoding. Each user u_i first encodes his multiset D_i as a group of binary vectors $S_{i1}, S_{i2}, \ldots, S_{i\max}$, and each one consists of l bits. When $t \leqslant f_j(D_i)$, the j-th bit of S_{it} will be set to 1, otherwise 0. For example, given the universal multiset $U = \{1, 1, 2, 2, 3, 3, 4, 4, 5, 5\}$ with $l = 5$ and $\max = 2$, multiset $D_i = \{1, 1, 3, 3, 4\}$ will be encoded as $S_{i1} = [10110]$, $S_{i2} = [10100]$.

Sanitizing. After encoding, each user uniformly randomly picks a value T_i from $\{1, 2, \ldots, \max\}$ and only sanitizes the T_i-th vector S_{i,T_i} by randomized response mechanism [8] with the privacy budget ϵ, that is, for each $j \in \{1, \ldots, l\}$,

$$\widetilde{S}_{iT_i}[j] = \begin{cases} 1 - S_{iT_i}[j], & \text{with probability } \dfrac{1}{\mathrm{e}^\epsilon + 1}, \\ S_{iT_i}[j], & \text{with probability } \dfrac{\mathrm{e}^\epsilon}{\mathrm{e}^\epsilon + 1}. \end{cases} \tag{2}$$

Uploading. Lastly, each user sends the perturbed vector $\widetilde{S}_{i,T_i}$ and the selected value T_i to the collector together.

Based on the received vector $\widetilde{S}_{i,T_i}$ and T_i from each user, the collector first builds max perturbed matrixes

* Corresponding author (email: zhuyw@nuaa.edu.cn)

$\widetilde{M}_1, \ldots, \widetilde{M}_{\max}$. The t-th perturbed matrix $\widetilde{M}_t$ consists of the perturbed vectors $\{\widetilde{S}_{it}, \ldots, \widetilde{S}_{i't}\}$ from the users $\{u_i, \ldots, u_{i'}\}$ who upload the value $T_i = t$. Each row of $\widetilde{M}_t = [\widetilde{S}_{it}^{\mathrm{T}}, \ldots, \widetilde{S}_{i't}^{\mathrm{T}}]$ corresponds to the perturbed vector of one user. For example, if $T_i = 3$, u_i's perturbed vector $\widetilde{S}_{i3}$ will be included in the third noisy matrix $\widetilde{M}_3$. From each noisy matrix $\widetilde{M}_t$, the collector can derive an estimation ($\hat{\boldsymbol{\rho}}_t = [\hat{\rho}_{t1}, \ldots, \hat{\rho}_{tl}]$) to the frequency of '1' in every column of the noise-free matrix $M_t = [S_{it}^{\mathrm{T}}, \ldots, S_{i't}^{\mathrm{T}}]$ by $\hat{\rho}_{tj} = \frac{p-1}{2p-1} + \frac{\tilde{\rho}_{tj}}{2p-1}$, where $p = \frac{\mathrm{e}^{\epsilon}}{\mathrm{e}^{\epsilon}+1}$, $j \in \{1, \ldots, l\}$ and $\tilde{\rho}_{tj}$ stands for the frequency of '1' in the j-th column of $\widetilde{M}_t$. Intuitively, the true percentage of '1' in a column should be equal to 100%, if and only if the corresponding item belongs to the intersection. Nevertheless, there exists deviation in the estimation, thus we relax the threshold to $1-s$, instead of 100%, to determine items in the intersection. Here, s denotes the standard deviation of the estimation, and previous work in [8] has shown that s can be computed by

$$s = \sqrt{\frac{\frac{1}{4} - (\frac{1}{2} - p)^2}{(2p-1)^2 n}}. \tag{3}$$

After that, the collector can gain intermediate intersections $\mathrm{AI}_1, \ldots, \mathrm{AI}_{\max}$ with $\hat{\boldsymbol{\rho}}_1, \ldots, \hat{\boldsymbol{\rho}}_{\max}$ and s. The intermediate intersection AI_t indicates that items in it can repeat at least t times in the final intersection I. Generally, if an item is in a latter intermediate intersection, e.g., AI_{t_a}, the item must be included in every intermediate intersection AI_{t_b}, where $t_b < t_a$. Thus, to decide the multiplicity of one item in the final intersection I, the collector only find the last intermediate intersection where the item occurs. However, due to the deviation of the estimation, the estimated intersection may include some items which are not possessed by all users, or exclude some items in the true intersection. For example, one item does not appear in the $(t'-1)$-th intermediate intersection $\mathrm{AI}_{t'-1}$, but appears in $\mathrm{AI}_{t'}$ lastly. Then it is hard to determine which intermediate intersection the item truly occurs in at last. Hence, it is of low accuracy to determine one item's multiplicity in I only by the last intermediate intersection where the item occur. Theorem 1 demonstrates that the probability of one item being excluded from two successive intermediate intersections by mistake is low, unless this item is not in these two intermediate intersections indeed. Therefore, the collector can use a new rule: only when one item is excluded from two successive intermediate intersections, the multiplicity of the item in I can be decided by the intermediate intersection before these two successive intersections. For instance, if one item is excluded by $\mathrm{AI}_{t'}$ and $\mathrm{AI}_{t'+1}$, its multiplicity in I can be regarded as $t'-1$. By following this rule, the collector derives the final intersection I with all intermediate intersections $\mathrm{AI}_1, \ldots, \mathrm{AI}_{\max}$.

Theorem 1. For one item which is in the t'-th and the $(t'+1)$-th true intermediate intersections, the probability that the item is excluded from these two estimated intersections $\mathrm{AI}_{t'}, \mathrm{AI}_{t'+1}$ by mistake is less than $\mathrm{e}^{\frac{-4\mathrm{e}^{\epsilon}}{(\mathrm{e}^{\epsilon}+1)^2}}$.

We then briefly describe the LDP-method for distributed union operation (DMSU-LDP). As De Morgan's laws still holds for multiset, we can utilize the multiset intersection scheme to solve multiset union. Firstly, each user computes the complement $\overline{D}_i$ of the secret multiset D_i. Then, each user processes the complement of his multiset by the perturbation protocol in the intersection scheme. After receiving the perturbed data from users, the collector estimates the intersection of users' complementary multisets, and derives the multiset union from complement of the intersection.

Security analysis. It is not hard to prove that the perturbation protocol designed for private sets of users satisfies ϵ-local differential privacy. The proof is provided in supporting information. Users' sensitive information can be sanitized by the perturbation protocol, and each bit in users' noisy vectors satisfy ϵ-LDP. Hence, even if there are $n-1$ users who offer their true data to the collector, the collector can still hardly deduce more information about the dataset of the last user other than the estimated results. That indicates the designed schemes can resist the collusion of $n-1$ users and the collector. Further more, each user completes his data perturbation independently and the communication between each user and the collector is only required once. Even if some users fail, the collector still can recover the intersection or union by the dataset from other users. Thus, our schemes enjoy strong robustness.

Evaluation results. We conduct experiments on two datasets[1)] to evaluate the proposed schemes.

(1) T10I4D100K (TK): It is a synthetic dataset with 100000 records and 999 different items.

(2) Accidents: It is a real dataset with 340183 records and 469 different items.

We employ the following metric to measure the utility of the designed schemes:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{j=1}^{l}(E[j] - T[j])^2}{l}}, \tag{4}$$

where $E[j]$ ($T[j]$) is the multiplicity of the j-th different item of U in the estimated (true) results. For example, suppose that the estimated result is $\{2, 2, 4, 5\}$. Then $E[1] = 0$, $E[2] = 2$ and so on.

In experiments, we suppose the maximal multiplicity max of items is 10, 20, 40 or 80, respectively. To ensure that the multiplicity of the items in each user's set does not exceed max, we divide the dataset TK into 10000, 5000, 2500, 1250 subsets correspondingly and divide the dataset Accidents into 34000, 17000, 8500, 4250 subsets correspondingly with each user keeping one subset.

Next, we conduct experiments with different values of max to compare the utility of the basic solution and the designed schemes. The description of the basic solution and the experimental figures are provided in supporting information. As max raises, the error (RMSE) of all schemes increases. With an increase in the privacy budget ϵ, the error of DMSI-LDP and DMSU-LDP is reduced, and yet the error of basic solutions only decreases when max is small. Additionally, the error of basic solutions is always higher than that of DMSI-LDP and DMSU-LDP except for the experiments on the union of TK. We find out that the dataset TK is too sparse, i.e., each record in the dataset only includes few items, which results in that the multiplicity of items in each subset can hardly exceed 10 even if we divide TK less than 10000 subsets. Further, the multiplicity of items in the true union is mostly below 5 when the dataset TK is divided into 5000 or 2500 subsets. In this situation, basic solutions and the designed schemes (DMSI-LDP, DMSU-LDP) have similar results. Overall, the proposed methods have good utility and usually surpass the basic solutions.

Conclusion. This study designed schemes to obtain distributed multiset intersection and union by exploiting LDP. In the schemes, the private items in each data owner's set

1) http://fimi.ua.ac.be/data/.

were sanitized to satisfy ϵ-LDP, and meanwhile the collector could derive a high-accuracy intersection and union from noisy sets. Through theoretical analysis and experiments, we presented that the proposed schemes enjoy good utility and strong robustness.

Acknowledgements This work was partly supported by National Key Research and Development Program of China (Grant No. 2017YFB0802300) and National Natural Science Foundation of China (Grant No. 61602240).

Supporting information Appendixes A–D. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Freedman M J, Nissim K, Pinkas B. Efficient private matching and set intersection. In: Proceedings of Eurocrypt 2004, Interlaken, 2004. 1–19

2 Samanthula B K, Jiang W. Secure multiset intersection cardinality and its application to Jaccard coefficient. IEEE Trans Dependable Secure Comput, 2016, 13: 591–604

3 Yao A C C. How to generate and exchange secrets. In: Proceedings of the 27th Annual Symposium on Foundations of Computer Science, Toronto, 1986. 162–167

4 Li X X, Zhu Y W, Wang J, et al. On the soundness and security of privacy-preserving SVM for outsourcing data classification. IEEE Trans Dependable Secure Comput, 2018, 15: 906–912

5 Kasiviswanathan S P, Lee H K, Nissim K, et al. What can we learn privately? SIAM J Comput, 2011, 40: 793–826

6 Dwork C, Mcsherry F, Nissim K. Calibrating noise to sensitivity in private data analysis. In: Proceedings of Conference on Theory of Cryptography, New York, 2006. 265–284

7 Xue Q, Zhu Y W, Wang J, et al. Distributed set intersection and union with local differential privacy. In: Proceedings of IEEE 23rd International Conference on Parallel and Distributed Systems, Shenzhen, 2017. 198–205

8 Holohan N, Leith D J, Mason O. Optimal differentially private mechanisms for randomised response. IEEE Trans Inform Forensic Secur, 2017, 12: 2726–2735