Secure Multiparty Computation for the dataset weight Problem

Qiang Liu, Daewoon Lim
Dongguk University
liuqiang0321@gmail.com, daewoonlim@gmail.com

집합의 가중치 문제에 대한 안전한 다자간 계산

유강, 임대운
Dongguk University

Abstract

Secure Multiparty Computation (SMPC), a vital branch of cryptography, plays a crucial role in addressing the issue of joint computation of private data among multiple parties. Against this backdrop, the private computation of the weights of datasets becomes a significant scientific endeavor, particularly playing a key role in fields such as electronic voting. This paper investigates the transformation of finite datasets into binary arrays using 0-1 substitution techniques, followed by secure substitution employing the Lifted Elgamal threshold encryption algorithm, which possesses additive homomorphism properties. This process effectively defends against collusion attacks, ensuring that no private information is leaked during the protocol execution and only the final computational results are obtained. Ultimately, the security of the protocol is demonstrated through simulation paradigm.

Ⅰ. Introduction

Secure Multiparty Computation (SMPC) is designed to address the problem of collaborative computation among parties that do not trust each other, with a focus on protecting the privacy of involved parties. In this computational framework, all parties only receive the predetermined computational outcomes upon completion of the process, and are precluded from accessing any additional information related to the privacy of other participants [1].

The problem of dataset computation constitutes a critical aspect of SMPC within the domain of confidential scientific computations, notably influencing sectors like electronic voting and medical data analysis. The construction of SMPC protocols typically employs techniques such as oblivious transfer, homomorphic encryption, etc. Additionally, security models are primarily divided into semi-honest and malicious adversary models, with most studies targeting malicious models built upon the semi-honest model, enhancing them through techniques like zero-knowledge proofs [2]. Currently, the idel-real simulation paradigm is a prevalent method for depicting the security of protocols, distinguishing between an ideal world and the real world: a protocol is considered secure if it can be proven that its implementation in the real world achieves the security objectives of the ideal world [3].

The problem of confidential computation of dataset weights is a significant area of research whith the domain of dataset problem, primarily used to safely and privately compute the frequency of all items across datasets. To date, there are relatively few protocols addressing the issue of dataset weight, and in most practical scenarios, datasets tend to have finite limits. Thus, developing effective and confidential protocol suitable for finite datasets becomes imperative. The scheme proposed utilizes 0-1 encoding and the Lifted Elgamal threshold encryption algorithm [4] resistng collusion attacks by semi-honest adversaries. Even corrupt parties follow all protocol prescribed steps while sharing information among themselves and attempting to extract additional information during the interaction process, they cannot obtain any extra private information.

Ⅱ. Method

Problem Description. Assume there are n parties P_i, each possessing a datase $X_i = \{x_{i1}, x_{i2}, ..., x_{ini}\} \subseteq Q$, where $i \in \{1,2,...,n\}$ and $Q = \{q_1, q_2, ..., q_l\}$. Another party, Alice, with n parties wish to determine the frequency of each item in dataset Q across the datasets of all parties, without revealing any private data of the parties.

Protocol Details.

Public: Dataset $Q = \{q_1, q_2, ..., q_l\}$

Input: P_i: $X_i = \{x_{i1}, x_{i2}, ..., x_{ini}\} \subseteq Q$, where $i \in \{1,2,...,n\}$.

Protocol:

offline stage: Alice, along with n parties, jointly execute the Lifted Elgamal threshold encryption algorithm, thereby collaboratively generating private key sk_i and public key pk.

Online stage:

1. P_i constructs the array $m_i=(m_{i1},m_{i2},...,m_{il})$. When $q_j\in X_1$, $m_{ij}=1$; otherwise, $m_{ij}=0$, where $j\in\{1,2,...,l\}$. P_i encrypts the array m_i by pk, namely, $E(m_i)=((a_{i1},b_{i1}),(a_{i2},b_{i2}),...,(a_{il},b_{il}))$then sends $E(m_i)$ to Alice.

2. Alice computes$\prod_{i=1}^{m}E(m_i)=((\prod_{i=1}^{m}a_{i1},\prod_{i=1}^{m}b_{i1}),(\prod_{i=1}^{m}a_{i2},\prod_{i=1}^{m}b_{i2}),...,(\prod_{i=1}^{m}a_{il},\prod_{i=1}^{m}b_{il}))$, and encrypts by pk obtaining $(c_1,c_2)=((E(\prod_{i=1}^{m}a_{i1}),E(\prod_{i=1}^{m}b_{i1})),(E(\prod_{i=1}^{m}a_{i2}),E(\prod_{i=1}^{m}b_{i2}),...,(E(\prod_{i=1}^{m}a_{il}),E(\prod_{i=1}^{m}b_{il}))$. Alice announces c_1.

3. P_i and Alice compute $d_{ij}=c_{1j}^{sk_i}$ respectively and send the results to Alice.

4. Alice computes $w=(w_1,w_2,...,w_l)$, where $w_j=c_{2j}/(\prod_{i=1}^{n+1}d_{ij})\bmod p$, where p is the large prime selected in key generation. w is the final result obtained by Alice, that is, the frequency of each item in dataset Q in n datasets. Alice outputs w.

Correctness and security analysis.

For the correctness of the protocol, each party firstly constructs binary arrays using 0-1 substitution. These arrays are then encrypted using the pk and sent to Alice. Upon receiving the encrypted information from all parties, Alice computats them and encrypts, where the computation and encrypytion of Alice is to encrypt $\sum_{i=1}^{m}(m_{ij})$. In the final stage, each party uses their respective sk_i to perform threshold decryption, resulting in Alice's final output - the values within the array w - being exactly as anticipated by the protocol. Therefore, the correctness of the protocol is ensured by the reliability of the Lifted Elgamal threshold encryption algorithm.

For the security, it can be ensured that protocol is secure under the semi-honest model and resistant to arbitrary collusion. Since every party plays an identical role in the execution of the protocol, analyzing the ability of P_1 to resist collusion attacks from other parties is sufficient to demonstrate the security of the protocol. This paper uses the ideal-reality simulation example to analyze the security of the protocol, and the specific analysis is shown as follows.

Construct a simulator A_s for corrupt parties. A_s minics $X_1^*=\{x_{11}^*,x_{12}^*,...,x_{1n1}^*\}$, then uses 0-1 substitution to encode it into a binary array $m_1^*=(m_{11}^*,m_{12}^*,...,m_{1l}^*)$, which is then encrypted to obtain $E(m_1^*)=((a_{11}^*,b_{11}^*),(a_{12}^*,b_{12}^*),...,(a_{1l}^*,b_{1l}^*))$. A_s computes the arrays according to the protocol, then encrypts it resulting in $(c_1^*,c_2^*)=((E^*(\prod_{i=1}^{m}a_{i1}),E^*(\prod_{i=1}^{m}b_{i1})),(E^*(\prod_{i=1}^{m}a_{i2}),E^*(\prod_{i=1}^{m}b_{i2})),...,(E^*(\prod_{i=1}^{m}a_{il}),E^*(\prod_{i=1}^{m}b_{il}))$. At this point, A_s's perspective is represented as $A_s=\{X_2,X_3,...,X_n,E(m_1^*),c_1^*,c_2^*,output^*\}$. During the actual execution, the real view is $view_{As}^{\pi}=\{X_2,X_3,...,X_n,E(m_1),c_1,c_2,output\}$. Since the Lifted Elgamal is semantically secure, from A_s's perspective, the simulated view and the actual view are indistinguishable. Therefore, the protocol is secure.

Computational and communication complexity

Computational and communication Complexity are given as below, where computational complexity is in unit of modular exponential operation, as well as communication complexity is measured by the number of interactions.

Comp.	Comm.	Party	Res. col.
$n(5l+1)+2l+3$	$4n$	$n+1$	√

TABLE 1. Computational and communication complexity
Note. "Comp.": Computational complexity; "Comm.": Communication complexity; "Res. col.": Resist collusion

Ⅲ. Conclusion

Confidential computation of dataset weights plays a significant role in everyday life. Currently, there is substantial research on dataset theory, and many practical problems can be translated into computational operations on datasets. This paper primarily introduces a weight assessment protocol for finite field datasets, which resists collusion using the Lifted Elgamal threshold encryption algorithm under the semi-honest model. As the data scale increases, the complexity of the protocol grows linearly. In the future, it's essential to develop more efficient protocols and explore more general cases, as well as consider the design of security protocols for malicious models as new directions for our research.

ACKNOWLEDGMENT

This paper is a study conducted with the support of the National Research Foundation of Korea, funded by the Ministry of Science and ICT. (S-2024-A0496-00009).

REFERENCES

[1] Braun L, Damgard I, Orlandi C. " Secure multiparty computation from threshold encryption based on class groups," Annual International Cryptology Conference, 2023, pp. 613-645.

[2] Ben-Efraim A, Lindell Y, Omri E. "Optimizing semi-honest secure multiparty computation for the internet," Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016, pp. 578-590.

[3] Cramer R, Damgard I B. "Secure multiparty computation," Cambridge University Press, 2015.

[4] Liu X, Tu X F, Luo D, et al. "Secure multiparty computation of graphs' intersection and union under the malicious model," Electronics, 2023, 12(2): 258.