Privacy-preserving Collaborative Computation: Methods, Challenges and Directions

Ikhlas Mastour, Layth Sliman, Benoît Charroux, Raoudha Ben Djemaa, Kamel Barkaoui

Privacy-preserving Collaborative Computation: Methods, Challenges and Directions

Ikhlas Mastour
ISITCom of H. Sousse
University of Sousse
H. Sousse, Tunisia
Efrei Paris
Paris Pantheon-Assas University
Paris, France
ikhlas.mastour@efrei.fr

Layth Sliman
Efrei Paris
Paris Pantheon-Assas University
Paris, France
layth.sliman@efrei.fr

Benoît Charroux
Efrei Paris
Paris Pantheon-Assas University
Paris, France
Benoît.charroux@efrei.fr

Raoudha Ben Djemaa
ISITCom of H. Sousse
University of Sousse
H. Sousse, Tunisia
raoudhaham@yahoo.fr

Kamel Barkaoui
CÉDRIC Laboratory
Conservatoire National des Arts et Métiers
Paris, France
kamel.barkaoui@cnam.fr

2023 International Conference on Computer and Applications (ICCA) | 979-8-3503-0325-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCA59364.2023.10401829

***Abstract*—Although data mining is very relevant to the medical sector, it has also raised privacy concerns since it is applied to sensitive data, which undoubtedly affects citizens' rights and freedoms, which are strictly regulated by the EU through the General Data Protection Regulation (GDPR). This concern creates a big gap between the data owner and the data analyst, and it is not easy to connect them. Thus, it is evidently important to ensure privacy. This need for privacy becomes a necessity when data from multiple entities aim to collaborate. To tackle this gap, several techniques worth mentioning can be employed during data analysis to ensure privacy, including secure multiparty computation, homomorphic encryption, and federated learning. In this paper, we present the state-of-the-art of existing approaches and discuss their drawbacks to finally identify outstanding challenges in this field.**

***Index Terms*—privacy-preserving, secure multiparty computation, homomorphic encryption, federated learning**

I. Introduction

Over the past decades, many medical institutions have transformed their paper-based systems into electronic systems to increase work efficiency and results. As a result of this digital transformation, a large amount of data (Big Data) is being gathered from various sources such as X-rays, computed tomography scans (CT), magnetic resonance images (MRI), ultrasound, etc. However, most of these data are not very well structured and suitable for diagnostic purposes. Therefore, the evaluation of these data requires robust analysis methods such as machine learning and data mining methods, since their complexity makes them unmanageable by conventional methods. Unfortunately, while the application of data mining for analytic purposes is very relevant, it has also raised privacy concerns. The use of medical data as a basic component in data mining creates a conflict with the principles of data protection, which is strictly regulated by the EU through the General Data Protection Regulation (GDPR). This concern creates a big gap between the data owner and the data analyst, and it is not easy to connect them. Healthcare providers are typically reluctant to share their data with analysts in order to avoid the risk of violating patient privacy. Thus, it is evidently important to enhance security in IT systems and ensure individual privacy. The need for privacy preserving becomes a necessity when data from multiple entities aim to collaborate. Collaboration can occur between organizations in the same healthcare industry or even between organizations in different industries, such as combining data from hospitals and insurance companies to link medical data with data about treatment costs. Privacy concerns arises in scenarios where a group of n parties, $p_1, p_2, \ldots, p_n$, aims to collectively learn a machine learning model on the union of their confidential databases. The challenge lies in finding a solution that enables model training across distributed sources while ensuring the privacy of each party's data without requiring data disclosure among the entities.

In line with the aforementioned problem, a lot of researchers have been interested in privacy-preserving data mining (PPDM) by suggesting several methods to preserve privacy. According to the the current literature, two primary categories of PPDM methods have been recognized: non-cryptographic and cryptographic methods. [1]. Non-cryptographic methods are widely employed for lightweight privacy preservation in machine learning. They encompass various techniques such as data perturbation, data anonymization, and output perturbation. These methods involve distorting the original data or modifying outputs by employing strategies like noise addition [2], data swapping [3], k-anonymization, and its variations [4]. Another prominent approach in this domain is

Differential Privacy [5]. However, these methods fall short in providing adequate protection as it remains possible to infer certain sensitive information from the perturbed data. Furthermore, the introduction of noise frequently yields less reliable and less accurate results.

In contrast, cryptographic methods play a crucial role in PPDM by employing cryptographic techniques to provide significantly stronger privacy guarantees. Among cryptography-based methods, secure multi-party computation (SMPC) [6] is a well-known method that is required when multiple parties want to jointly compute a function over their inputs where participants do not reveal their inputs to each other. One of the most important directions of building SMPC is based on advanced cryptosystems such as homomorphic encryption (HE) [7]. HE presents an emerging cryptographic research area focused on preserving users' privacy by allowing a non-trusted party to perform computations on encrypted data without requiring the data to be decrypted. The result of the computation remains encrypted and represents the encrypted result that would be obtained if the same computation were performed on the original data. Despite the fact that cryptographic approaches maintain data quality and result accuracy while offering robust privacy guarantees, they introduce a significant computational / communication overhead.

A novel and highly promising framework has emerged with the primary goal of ensuring privacy in distributed machine learning. This innovative paradigm is referred to as Federated Learning (FL) [8]. FL-based systems can achieve significantly enhanced privacy preservation since they involve the sharing of local models among distributed parties, rather than sharing local data. Given these inherent advantages, the adoption of cryptographic methods and federated learning for privacy-preserving data in healthcare systems has attracted considerable attention in recent years, thereby ensuring a very high level of data privacy. Accordingly, our main interest in this paper is focused on current approaches based on SMPC, HE, and FL for privacy-preserving collaborative computations. The aim of this paper is to provide an overview of existing approaches and discuss their limitations in order to outline the challenges and open problems as well as to point out future directions.

The remainder of this paper provides a brief overview of SMPC, HE, and FL in section II. Section III presents the state-of-the-art of current approaches. The purpose of section IV is to highlight the drawbacks of the discussed approaches in order to identify outstanding challenges. Section V outlines promising directions for future research. Finally, section VI concludes the paper and highlights our perspective.

II. Background

The purpose of this section is to provide a brief overview of SMPC, HE and FL.

A. Secure Multiparty Computation

SMPC was introduced by Yao [6] in the 1980s which aims to build a secure protocol that allows distributed parties to jointly calculate a function over their inputs without disclosing any private information to each other. Based on the secret sharing method, the workflow of SMPC consists of three basic steps. As shown in Fig. 1, Given m participants and n computing parties: 1) Each participant sends a separate and different secret to each of the n computing parties. 2) Each computing party calculates the intermediate results on the m secrets and shares these results with the other $n-1$ computing parties. 3) Each computing party aggregates all the exchanged results between them to calculate the final results. Once the aggregation is finished, each participant must obtain the final result without acquiring any other information.

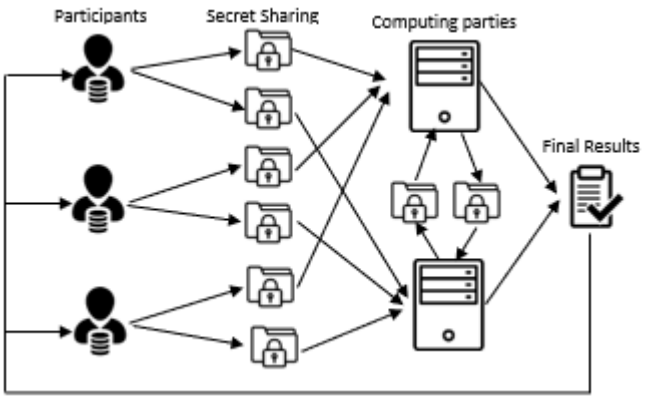

Fig. 1. Secure Multiparty Computation [9]

In SMPC, parties that are under the control of an adversary and consequently follow the adversary's instructions are commonly referred to as "corrupted parties". The security model in SMPC relies on two main types:

- Semi-honest (also known as "honest-but-curious"): in this scenario, a corrupted party follows the protocol specification correctly. However, the adversary has the capability to acquire the internal state of a corrupted party and may attempt to exploit this information to gain insights that should remain private.
- Malicious: in this scenario, a corrupted party can arbitrarily deviate from the protocol specification, according to the adversary's instructions. This can involve injecting arbitrary messages into the network or generating false results, potentially compromising the integrity and privacy of the computation.

B. Homomorphic Encryption

Homomorphic encryption is an emerging cryptographic research area that provides the ability to perform arbitrary addition and multiplication operations on encrypted data without the requirement for decryption. As illustrated in Fig. 2, HE-based systems typically follow three main steps: 1) The participating parties encrypt their data and send the encrypted

data to a computing party. 2) The Computing party executes the operations over the encrypted data and subsequently shares the results, which remain encrypted, with the participants. 3) The participants access the results by decrypting them.

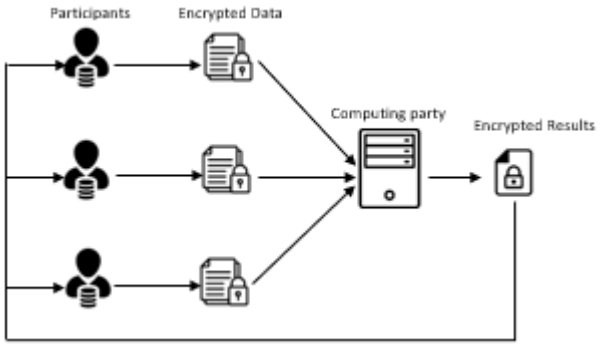

Fig. 2. Homomorphic Encryption [9]

Homomorphic cryptosystem type is defined according to the operation type and the number of mathematical functions that will be performed over encrypted data. Addition and multiplication operations constitute a complete base of functions since all mathematical functions can be expressed in polynomials which is a sequence of addition and multiplication operations. Three types of homomorphic cryptosystems have been defined in the literature: Partially homomorphic encryption, Somewhat homomorphic encryption and Fully homomorphic encryption. We refer the reader to [10] for more details.

C. Federated Learning

Federated learning is a novel machine learning paradigm in which multiple machines collaboratively train a machine learning model while keeping their data locally. Instead of sharing its private data with other participants, each client locally trains the model over its data and share local model with the server for aggregation purposes. As depicted in Fig. 3, FL is an iterative process that typically follow three steps: 1) Initialization: the server initializes model parameters and distributes them to the participants. 2) Training: each participant independently and locally trains the model over its own data and sends its model parameters to the server for aggregation purposes. 3) Aggregation: the server collects the local parameters from participants and aggregates them to update the current model for the next iteration. Step 2) and 3) will be repeated until the model converges.

III. State of the art

In this section, we will explore various relevant approaches, highlighting their strengths and discussing their limitations. Table I gives a comparative study of existing approaches.

Kumar et al. [11] proposed a novel approach for privacy-ensured self-care health management using SMPC. Through this approach patients can share their sensitive data to the hospital server through the online mode, the data will be shared in an encrypted format which will be matched with

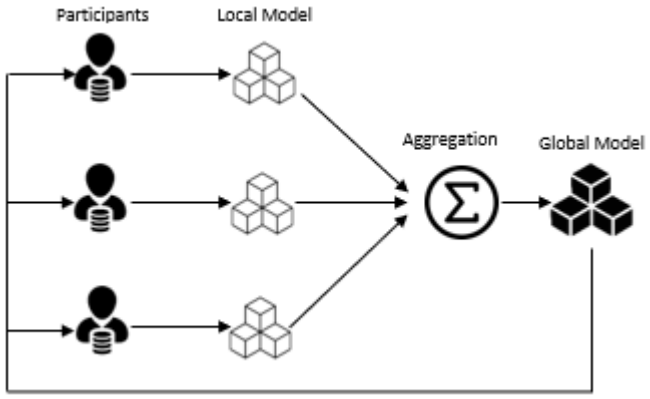

Fig. 3. Federated Learning [9]

the existing data at the hospital records, and the best relevant match based on the smart Index of disease. For statistical matching analysis, authors used random forest algorithm and data privacy is ensured through the use of Paillier's scheme. Although the proposed approach provides security and reliability of the user data, the model runs very slow as the operations are performed on the encrypted data.

Wibawa et al. [12] have proposed a privacy-preserving federated learning system that uses Brakerski-Fan-Vercauteren (BFV) scheme to protect a convolutional neural network (CNN) model trained on medical data. The authors evaluated their system using real-world COVID-19 X-ray scans, which were divided arbitrarily among clients. Despite achieving similar accuracy performances with a deviation of only 1% between the encrypted and unencrypted processes, there was an exponential difference in execution time. Furthermore, the authors failed to mention the security model they addressed in their protocol against adversarial attacks.

The approach of Lu et al. [13] presents a privacy-preserving cox regression protocol for analyzing survival data. The proposed protocol allows researchers to train models on horizontally or vertically partitioned datasets while ensuring privacy for both sensitive data and trained models. Two homomorphic encryption schemes are used in this protocol, namely, the BFV scheme and the CheonKim-Kim-Song (CKKS) scheme. There are two major limitations in this work. Firstly, it is imperative that all parties remain online throughout the entire training process; otherwise, the execution will fail. Secondly, when dealing with high-dimensional data at scale, the proposed solution becomes less practical.

Fan et al. [14] proposed a privacy-preserving multi-party computing scheme for K-means clustering (PPMCK) in order to ensure data privacy in both the cloud and at the local side for each party. PPMCK uses homomorphic encryption to protect data privacy. To deal with the problem of computing

the nearest clustering center in the ciphertext form and the problem of recalculating the cluster centers, PPMCK uses order-preserving encryption (OPE) and privacy-preserving weight average problem (PPWAP). The multiparty computing in this proposal involves only two participants, which may not guarantee a high level of privacy. Generally, in a SMPC with n computing entities, the larger n, the stronger the privacy.

Froelicher et al. [15] proposed a novel multiparty federated analytics system using lattice-based homomorphic encryption. Two essential biomedical tasks are addressed in this work which are Kaplan-Meier survival analysis and genome-wide association studies. While the proposed solution is efficient in terms of execution time and communication, using the differential privacy method to prevent privacy leakage by adding noise to intermediate data results in inaccurate models. Furthermore, there is a lack of consensus around how to set parameters for differential privacy in order to provide acceptable mitigation of inference risks in practice.

Van et al. [16] proposed a solution for hospital Erasmus MC and health insurance company Achmea which allows them to securely train a regression model on vertically-partitioned synthetic data in order to identify high-impact lifestyle factors for heart failure. The proposed solution uses SMPC, HE, and Shamir's scheme. Despite being the first solution to ensure security with Lasso regression, this work lacks scalability when dealing with multiple entities with a dataset with hundreds of features.

Fang and Qian [17] proposed a privacy-preserving machine learning framework for multi-layer perceptron (MLP) models, called PFMLP. This framework combines an enhanced version of Paillier's scheme with FL, resulting in improved encryption and decryption performance, with a 25-28% enhancement. However, the updating of key pairs during each iteration introduced computational and communication overhead, negatively affecting network training efficiency. This overhead can be aggravated by limited network bandwidth, making the PFMLP approach less suitable for large-scale deployments. Furthermore, in this work, the security assumption protects against colluding parties; however, the authors did not provide a formal proof for this assumption.

Paul et al. [18] proposed a collaborative learning protocol for sharing classified time-series data within entities. The protocol encrypts each data's feature using the CKKS encryption scheme and trains the last layers using encrypted logistic regression. The in-hospital mortality task was chosen for the experiments with long short-term memory (LSTM) architecture. While the proposed approach appears feasible for applying LSTM algorithms to time-series data using HE schemes, it is important to note that the authors have only considered a semi-honest threat model. Furthermore, the work lacks a formal proof to support this claim, and it does not address the potential scenario of corrupted parties collaborating outside the protocol to exchange information.

Boemer et al. [19] presents MP2ML, a machine learning framework that integrates nGraph-HE and secure two-party computation framework for artificial neural network (ANN). This work introduces a novel scheme based on the CKKS scheme to ensure the privacy of both input data and model weights during ANN inference. The primary concern with this approach is that it sends the data back to the users after every layer, requiring them to execute the non-linear activation function. This, in turn, leads to a significant computational overhead on the user side.

Son et al. [20] proposed a novel solution for the privacy-preserving Gated Recurrent Unit (GRU) inference model using CKKS scheme and secure two-party computation. The proposed approach was validated on breast cancer recurrence prediction with 13,117 patients' medical data. Similar to the aforementioned approach [18], the major limitation of this work is that it can only guarantee security against semi-honest model.

IV. Discussion

In this section, we evaluate the previously mentioned approaches from various perspectives. As illustrated in Table I, our evaluation is based on 8 criteria as follows:

- **Method:** indicates the privacy-preserving methods that are used to develop a secure computation environment, including SMPC, HE and FL.
- **Application:** indicates which use case scenario was addressed.
- **Data mining task:** presents the algorithms which have been used in a privacy-preserving manner.
- **Data distribution:** describes how data is distributed. Three scenarios of data partitioning are considered:1) Horizontal partitioning with the same attributes from different data instances; 2) Vertical partitioning data with the same data instances but with different attributes; 3) Arbitrary partitioning which combines aspects of both horizontal and vertical partitioning, where data providers hold different attributes for different data instances.
- **Security model:** presents the assumed adversarial behavior, which includes two types as defined in Section II-A: semi-honest and malicious.
- **Number of party:** specifies the number of parties involved in the computation task. The number of computing parties affects the level of privacy, with privacy being preserved as long as the majority of entities do not collude.
- **Computation / communication cost:** factor indicates how efficiently an approach consumes network bandwidth and computational resources. An approach is considered more communication-efficient when it involves less data traffic exchanged over the network. Indeed, computational cost serves as an indicator of the additional computational

TABLE I
COMPARISON BETWEEN EXISTING APPROACHES

Ref	Year	Method	Application	Data Mining task	Data distribution	Security model	N party	Cost	
								Computation	Communication
[11]	2020	SMPC[1]+HE[2]	assistant medical	random forest	centralized	-	-	-	-
[12]	2022	FL[3]+SMPC+HE	COVID-19 detection	CNN	arbitrary	-	multi	-	-
[13]	2021	SMPC+HE	survival analysis	cox regression	arbitrary	semi-honest	multi	44min	100 MB
[14]	2021	SMPC+HE	-	k-means	centralized	semi-honest/malicious	2	386.11s	-
[15]	2021	FL+SMPC+HE	survival analysis	Kaplan-Meier	distributed	semi-honest	96	12s	-
			GWAS	linear regression			12	60 min	1 GB - 500 MB
[16]	2021	SMPC+HE	heart disease causes	regression LASSO	vertically	semi-honest	2	60min	-
[17]	2021	FL+SMPC+HE	-	MLP	distributed	-	multi	-	-
[18]	2021	SMPC+HE	in-hospital mortality	LSTM	distributed	semi-honest	2	60min	-
[19]	2020	SMPC+HE	-	ANN	centralized	semi-honest	2	-	9.6 GB
[20]	2021	SMPC+HE	cancer prediction	GRU	centralized	semi-honest	2	-	1 GB

[1] Secure multiparty computation.
[2] Homomorphic encryption.
[3] Federated learning.

overhead that an approach incurs. This overhead arises when handling secret shares from a significant number of participants or dealing with a vast volume of data. Moreover, it arises from the use of HE since it performs operations on encrypted data.

Based on the state-of-the-art presented above and according to the evaluation outlined in Table I, we were able to highlight some limitations. First, most approaches assume a central authority orchestrates the computation task which potentially represents a single point of failure where in case that computing entity fails, the computation cannot be performed. Second, notice that to preserve the data privacy under two computing parties those latter must be non-colluding. In other words, computing parties should not share the data they compute; they should only share the results of their computation. Otherwise, the computing parties can then reveal the participants' data. Generally, in SMPC with n computing entities, privacy is protected as long as most $n-1$ computing entities are non-colluding with each other [21]. The larger the n, the stronger the privacy; however, this leads to communication and computation time overhead. The communication overhead problem is caused by the exchanged results (model parameters) during the computation process. As model parameters increases in size, communication overhead increases. Therefore, for a large model that requires thousands of iterations to converge it is extremely difficult and expensive for devices with limited bandwidth to communicate. On the other hand, high computation overhead is related to the use of homomorphic encryption since it performs operations on encrypted data which makes it unfeasible for complex tasks. For example, when developing nonlinear artificial intelligence models with deep neural networks. Moreover, for the security models in SMPC, most approaches only take into account semi-honest models and do not consider malicious security models. In SMPC, it is assumed that a protocol execution may be susceptible to attacks from an external adversary or a subset of corrupted parties. The aim of such attacks may include gaining access to some private information or manipulating the computation to produce incorrect results. Thus, secure computation protocols must satisfy two crucial requirements: privacy, which ensures that parties only learn the output and nothing more, and correctness, which guarantees that each party receives the correct result. Another persistent issue, is the susceptibility of FL systems to various types of attacks. Despite the revolutionary impact of FL in enabling collaboration among parties without sharing their local data, many research works have demonstrated that FL systems are highly vulnerable to various kinds of attacks, including "membership inference attacks, model poisoning, model inversion attacks", etc. [22]. Consequently, FL may not consistently offer adequate privacy guarantees, as communicating model updates throughout the training process can nonetheless reveal a certain amount of sensitive information.

V. RESEARCH DIRECTIONS

Privacy-preserving computation has been rapidly developing through active research programs across different scientific communities including data mining and machine learning, mathematics and statistics, cryptography and data management. While numerous approaches have been proposed to address privacy concerns, several critical challenges remain unaddressed. This section provides a summary of our insights into potential future research directions.

Communication is still a critical bottleneck in both FL and SMPC based approaches. This challenge arises from massive number of devices coupled with limitations in network communication bandwidth. To further tackle this issues, it's crucial to focus on two key aspects: 1) reducing the number of communication rounds, and 2) decreasing the size of exchanged results (models updates), possibly through the implementation of compression techniques.

Most of approaches assumed a central authority coordinates a set of parties to carry out computational tasks. To mitigate this challenge, future research could explore the

implementation of smart contracts to replace the central server, avoiding the risk of a single-point failure.

Moreover, model security remains an open problem that requires significant attention. The majority of approaches have been developed based on the semi-honest model assumption, which assumes that participating parties aim to learn some private information without causing harm or posing a threat to others. While this assumption may hold true in some cases, real-life examples often necessitate more strict considerations regarding malicious behavior. To address this issue, integrating blockchain technology with privacy-preserving computation systems is an interesting research topic. This approach holds the potential to enhance transparency and traceability within computation systems, given that all blockchain actions are inherently immutable. In more detail, a distributed ledger using smart contracts can act as a system controller, orchestrating all actions with the use of a Zero Knowledge Verifiable Computation scheme [23] to prevent dishonest behaviors where computing parties are enforced to produce a proof of correctness of computation.

Another notable challenge that remains open in FL, as previously mentioned, revolves around the risk of the exchanged local model parameters during the training process disclosing certain sensitive information. One potential solution to mitigate this issue is using HE schemes, particularly lattice-based ones, which are considered resilient to quantum computing attacks.

VI. Conclusion

There is no doubt that data mining has the potential to enhance decision-making and provide better services for many companies. However, using sensitive data in data mining tasks raised privacy concerns. In this regard, several methods worth mentioning such as secure multiparty computation, homomorphic encryption and federated learning, can be employed during data analysis to ensure privacy. In this paper, we discussed the state-of-the-art of existing approaches and identified their gaps and weaknesses. Additionally, we have outlined promising directions for future research, with the goal of achieving further significant enhancements in privacy-preserving data mining systems.

In our future work, our objective is to propose a notable approach for federated learning leveraging the capabilities of homomorphic encryption and blockchain. We will formulate our solution while maintaining alignment with the security assumptions inherent in the SMPC model, including both the semi-honest and malicious security models.

References

[1] Talbi, R. (2021). Robust and privacy preserving distributed machine learning (Doctoral dissertation, Université de Lyon).

[2] Mivule, K. (2013). Utilizing noise addition for data privacy, an overview. arXiv preprint arXiv:1309.3958.

[3] Dalenius, T., & Reiss, S. P. (1982). Data-swapping: A technique for disclosure control. Journal of statistical planning and inference, 6(1), 73-85.

[4] Rajendran, K., Jayabalan, M., & Rana, M. E. (2017). A study on k-anonymity, l-diversity, and t-closeness techniques. IJCSNS, 17(12), 172.

[5] Bugliesi, M., Preneel, B., Sassone, V., & Wegener, I. (Eds.). (2006). Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part I (Vol. 4051). Springer.

[6] Yao, A. C. (1982, November). Protocols for secure computations. In 23rd annual symposium on foundations of computer science (sfcs 1982) (pp. 160-164). IEEE.

[7] Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. ACM Computing Surveys (Csur), 51(4), 1-35.

[8] Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST), 10(2), 1-19.

[9] Torkzadehmahani, R., Nasirigerdeh, R., Blumenthal, D. B., Kacprowski, T., List, M., Matschinske, J., ... & Baumbach, J. (2022). Privacy-preserving artificial intelligence techniques in biomedicine. Methods of Information in Medicine.

[10] Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. ACM Computing Surveys (Csur), 51(4), 1-35.

[11] Kumar, A. V., Sujith, M. S., Sai, K. T., Rajesh, G., & Yashwanth, D. J. S. (2020, December). Secure Multiparty computation enabled E-Healthcare system with Homomorphic encryption. In IOP Conference Series: Materials Science and Engineering (Vol. 981, No. 2, p. 022079). IOP Publishing.

[12] Wibawa, F., Catak, F. O., Kuzlu, M., Sarp, S., & Cali, U. (2022, June). Homomorphic Encryption and Federated Learning based Privacy-Preserving CNN Training: COVID-19 Detection Use-Case. In EICC 2022: Proccedings of the European Interdisciplinary Cybersecurity Conference (pp. 85-90).

[13] Lu, Y., Tian, Y., Zhou, T., Zhu, S., & Li, J. (2021). Multicenter privacy-preserving Cox analysis based on homomorphic encryption. IEEE Journal of Biomedical and Health Informatics, 25(9), 3310-3320.

[14] Fan, Y., Bai, J., Lei, X., Lin, W., Hu, Q., Wu, G., ... & Tan, G. (2021). PPMCK: Privacy-preserving multi-party computing for K-means clustering. Journal of Parallel and Distributed Computing, 154, 54-63.

[15] Froelicher, D., Troncoso-Pastoriza, J. R., Raisaro, J. L., Cuendet, M. A., Sousa, J. S., Cho, H., ... & Hubaux, J. P. (2021). Truly privacy-preserving federated analytics for precision medicine with multiparty homomorphic encryption. Nature communications, 12(1), 1-10.

[16] van Egmond, M. B., Spini, G., van der Galien, O., IJpma, A., Veugen, T., Kraaij, W., ... & Kooij-Janic, M. (2021). Privacy-preserving dataset combination and Lasso regression for healthcare predictions. BMC medical informatics and decision making, 21(1), 1-16.

[17] Fang, H., & Qian, Q. (2021). Privacy preserving machine learning with homomorphic encryption and federated learning. Future Internet, 13(4), 94.

[18] Paul, J., Annamalai, M. S. M. S., Ming, W., Al Badawi, A., Veeravalli, B., & Aung, K. M. M. (2021). Privacy-Preserving Collective Learning With Homomorphic Encryption. IEEE Access, 9, 132084-132096.

[19] Boemer, F., Cammarota, R., Demmler, D., Schneider, T., & Yalame, H. (2020, August). MP2ML: A mixed-protocol machine learning framework for private inference. In Proceedings of the 15th International Conference on Availability, Reliability and Security (pp. 1-10).

[20] Son, Y., Han, K., Lee, Y. S., Yu, J., Im, Y. H., & Shin, S. Y. (2021). Privacy-preserving breast cancer recurrence prediction based on homomorphic encryption and secure two party computation. Plos one, 16(12), e0260681.

[21] Ranbaduge, T., Vatsalan, D., & Christen, P. (2020). Secure Multi-party Summation Protocols: Are They Secure Enough Under Collusion?. Trans. Data Priv., 13(1), 25-60.

[22] Jere, M. S., Farnan, T., & Koushanfar, F. (2020). A taxonomy of attacks on federated learning. IEEE Security & Privacy, 19(2), 20-28.

[23] Blum, M., De Santis, A., Micali, S., & Persiano, G. (1991). Noninteractive zero-knowledge. SIAM Journal on Computing, 20(6), 1084-1118.