Babylonian Journal of Artificial Intelligence
Vol. 2024, **pp**. 168–174
DOI: https://doi.org/10.58496/BJAI/2024/018 ISSN: 3006-5437
https://mesopotamian.press/journals/index.php/BJAI

Research Article

Enhancing Privacy in Artificial Intelligence Services Using Hybrid Homomorphic Encryption

Mustafa A Jalil[1],*,

[1] *Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI), University of Cordoba, Campus Universitario de Rabanales,Cordoba, 14071, Spain.*

ARTICLE INFO

Article History

Received 16 Aug 2024
Revised: 14 Sep 2024
Accepted 15 Nov 2024
Published 08 Dec 2024

Keywords

Hybrid Homomorphic Encryption (HHE)

Privacy-Preserving Artificial Intelligence (PPAI)

Encrypted Data Classification

Secure AI Framework

Resource-Constrained Devices

ABSTRACT

The increasing occurrence of cyberattacks specifically aimed at critical infrastructure has led to the adoption of network intrusion detection techniques for the Internet of Things (IoT). AI is transforming multiple sectors today, the growth of adversarial attacks on AI models and models present imperative privacy issues which hinder its larger implementation. Some of the Privacy-Preserving Artificial Intelligence (PPAI) methods including HE make it possible to secure data during the calculation process. Yet conventional HE techniques experience certain disadvantages at present with applicability to highly scalable and resource-limited applications. Moreover, this paper presents an HHE technique that is designed by integrating symmetric cryptography with HE to overcome the above-mentioned challenges successfully. To this end, we propose the GuardAI framework for end devices with limited resources such that encrypted data can be classified while preserving the privacy of input data and AI models. To show the effectiveness of the HHE, we apply it to the actual problem of heart disease classification based on the easily contaminated ECG signals. In this way, the proposed method maintains the privacy of the data with little computational and communication cost for analysts and devices and has a fairly reasonable level of accuracy in comparison with unencrypted inference. This work therefore provides a foundation for secure and private approach in AI especially for those developed to suit devices and systems with limited resources by incorporating HHE into the PPAI systems.

1. INTRODUCTION

AI has become one of the most effective innovations throughout various sectors since it improves automation, decision making and results in the forms of analytics. Nevertheless, recent upgrade of AI systems as components of applications operating with personal data has stirred up great concerns about privacy. These include, but are not limited to, bypassing of crucial security safeguards for Deep learning models that can be applied on resource-constrained devices or in joint environments, and thus raising the dangers of data leakage and malicious use. They dent public confidence in AI applications and keep them from being adopted broadly for fear of data leaks, breaches especially in sensitive areas like the medical, the financial, and the governmental.

Privacy- Preserving Artificial Intelligence (PPAI) has emerged as a key field of study to develop the protection of user data as well as the efficient safe execution of the AI model. From the existing techniques in the field of secure data computation, there is a technique known as Homomorphic Encryption (HE) that does allow computations to be carried out on data without the content ever being revealed. HE makes it possible to perform actual operations such as addition, multiplication of plaintext without actually exposing data to unauthorized persons. This work stems from Gentry's Fully Homomorphic Encryption (FHE) scheme [1] on which subsequent developments CKKS [2], TFHE [3], and BFV [4, 5] improved to make HE applicable in real life AI as in MLaaS [6-13].

All the same, HE has the following limitations, which can be seen as threats to its success: computational overhead and the increased size of the ciphertexts. It also has limitations in its application due to scalability and resources to be used in resource-demanding AI project. To overcome these problems, researchers proposed such an advanced form of the HE algorithm as Hybrid Homomorphic Encryption (HHE), the application of which is based on the combination of the

*Corresponding author. Email:Mustaf.a.76t@gmail.com

symmetric and the HE approaches [14, 15]. HHE incurs relatively low communication overhead and a compact ciphertext size relative to the basic HHE while requiring a relatively small number of iterations to establish a critical level of security. HHE empowers the first stage of transformation of data using a symmetric key algorithm, and then homomorphically encrypting the resultant symmetric key. This data is subsequently encrypted twice by the server and then the ciphertexts are combined in a way suitable for performing the homomorphic operations. This approach not only leads to the reduction of the Ciphertext size but also solves problems of high computational cost and an increase in the multiplicative depth that is inherent to pure HE schemes. Recent development in the HE Friendly symmetric cipher like HERA and Rubato have made the HHE more efficient which provides some light on the HHE as the solution to implement security in AI models.

2. LITERATURE REVIEW

The field of Privacy-Preserving Artificial Intelligence (PPAI) has experienced rapid growth in recent years, driven by the increasing demand for secure and efficient AI systems that protect sensitive data. Among the techniques designed to address these concerns, Homomorphic Encryption (HE) has emerged as a cornerstone due to its unique ability to perform computations directly on encrypted data. This capability ensures that data confidentiality is maintained throughout the computational process, making HE highly suitable for applications in privacy-sensitive domains.

The foundation of HE was laid by Gentry's Fully Homomorphic Encryption (FHE) scheme, which allowed unlimited computations on encrypted data for the first time. However, the initial FHE models were computationally expensive, leading to the development of more efficient schemes. The BFV scheme, for instance, enables arithmetic operations on integer ciphertexts and eliminates the need for costly bootstrapping, making it suitable for applications requiring moderate computational depth [16]. Similarly, the CKKS scheme, introduced by Cheon et al., facilitates computations on floating-point data, making it particularly useful for real-world AI applications that involve approximate numerical calculations [17]. The TFHE scheme, proposed by Chillotti et al., improves bootstrapping efficiency and supports an unlimited number of binary operations, making it ideal for binary data processing task [18].

These advancements have enabled HE schemes to be integrated into various AI applications, such as privacy-preserving machine learning (PPML). For example, TFHE has been used for implementing lookup table (LUT) searches for non-linear activations, while polynomial approximations have been adopted in BFV and CKKS for similar purposes [19]. Notable implementations like TAPAS and FHE-DiNN have demonstrated the potential of HE in achieving high accuracy in PPML tasks [20].

Despite its promise, HE faces significant challenges, particularly regarding computational overhead and ciphertext expansion. These limitations hinder its scalability and practical applicability in large-scale AI environments. Computationally intensive operations and the storage requirements for expanded ciphertexts remain barriers to the widespread adoption of HE in real-time and resource-constrained applications [21].

To address these challenges, researchers have explored Hybrid Homomorphic Encryption (HHE), which combines symmetric cryptography with HE to reduce computational costs and communication overhead. Early HHE implementations relied on symmetric ciphers such as AES. However, AES's high multiplicative depth proved inefficient for HHE, prompting the development of optimized symmetric ciphers tailored for HHE [22].

Several HHE schemes have been proposed to enhance the efficiency and practicality of HE. HERA, for instance, supports floating-point operations and integrates Weighted Modular Arithmetic (WMA) to improve performance. Elisabeth, designed for TFHE, optimizes operations on binary data, while PASTA, tailored for BFV, focuses on integer computations to ensure scalability in resource-constrained environments [23].

HHE has shown promise in real-world PPML applications, addressing the computational inefficiencies of traditional HE schemes. It has been successfully applied in secure computation, privacy-preserving AI, and scalable AI models. However, its practical deployment remains limited due to the complexity of implementation and the scarcity of large-scale use cases in the literature. For instance, HERA and CKKS enable operations on floating-point objects, while Elisabeth and TFHE optimize binary computations, highlighting the versatility of HHE in addressing diverse privacy-preserving requirements [24].

The advancements in HE and HHE have significantly improved the feasibility of privacy-preserving AI applications, providing robust solutions for secure computations. However, challenges such as high computational overhead and practical deployment persist, necessitating further research. This study builds on these advancements by proposing a novel framework leveraging HHE to enhance privacy and scalability in AI services, demonstrating its applicability in sensitive domains such as healthcare.

TABLE I. SUMMARY OF HE AND HHE SYSTEMS AND THEIR APPLICATIONS IN PRIVACY-PRESERVING PPAI

Scheme	Supported HE Framework	Focus Area	Applications	Challenges
BFV	Integer ciphertexts	Arithmetic computations	Machine learning, encryption	Limited scalability
CKKS	Floating-point data	Approximate operations	AI model training	High ciphertext expansion
TFHE	Binary operations	Bootstrapping efficiency	Real-time AI tasks	Computational overhead
HERA	CKKS	Floating-point objects	Secure computation	Requires advanced optimizations
Elisabeth	TFHE	Binary data optimization	Privacy-preserving AI	High computational cost
PASTA	BFV	Integer computations	Scalable AI models	Complexity in integration

3. METHODOLOGY

3.1. Model of System

The system for PPML involves a user group that encrypts data, with a unique key for each user. Many CSPs collect symmetrically encrypted data from users. An analyst with a machine-learning model interprets the outcomes of machine-learning operations on pre-encrypted data stored at CSP. Decrypted information is obtained from the HE evaluation of collected encrypted data to understand users' data. The system design is illustrated in Figure 1.

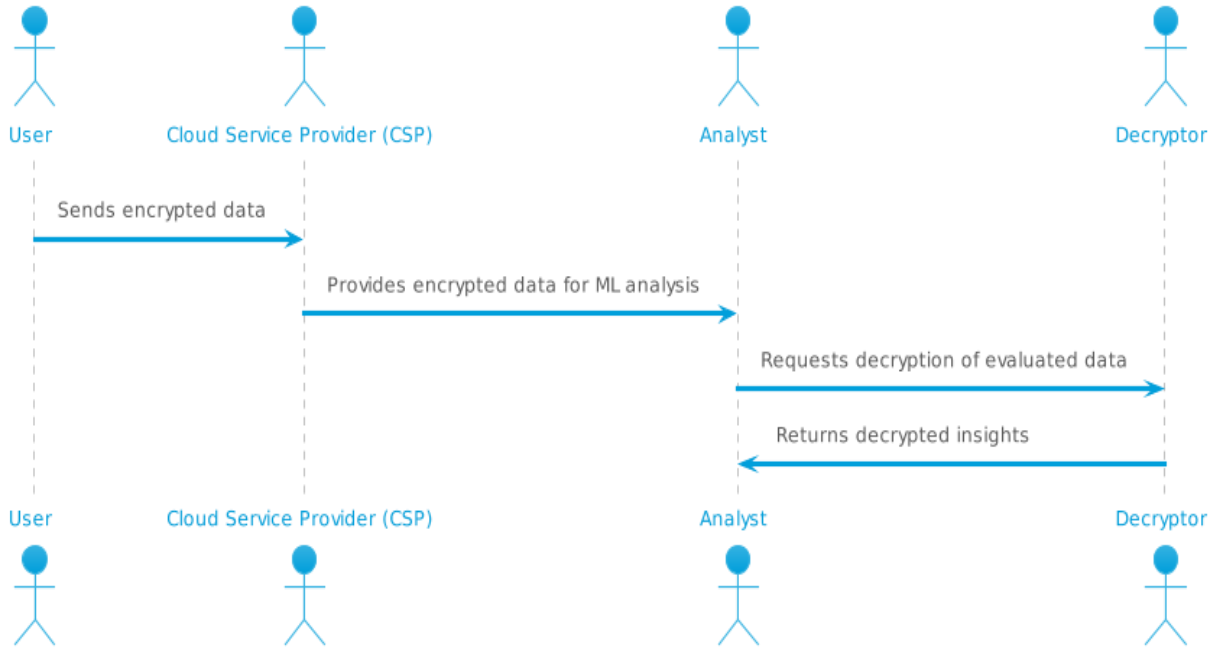

Fig. 1. Data Privacy Model for Secure ML.

3.2. 2GML Protocol in GuardML

This section outlines the development of the 2GML protocol within the GuardML framework. The 2GML protocol constitutes the second phase of the GuardML solution within the Hybrid Homomorphic Privacy-Preserving protocol. This phase is designed to address the requirements of specific machine learning applications intended for commercial use, with the models being owned by the CSP. Encrypted data and models stay confidential while Cloud Service Providers can perform computations. This configuration is effective for machine learning tasks when analysts do not want model information but require computational resources from the cloud service provider.

The architectural architecture of the 2GML Protocol comprises Secure Symmetric Encryption (SKE), ABFV-based Hybrid Homomorphic Encryption (HHE), Public-Key Encryption Scheme (PKE), Signature Scheme (σ), and Cryptographic Hash Function (H(·)). These components strengthen the security of the message's integrity, signature, decryption, and encryption. They are ideally suited for commercial environments as they allow CSPs to possess ML models and offer backend computing, as illustrated in Figure 2. As well as Table 2 delineates the essential processes and operations of the 2GML protocol, as outlined in GuardML, focusing on its core capabilities for enabling secure machine learning interactions between a user and a Cloud Service Provider.

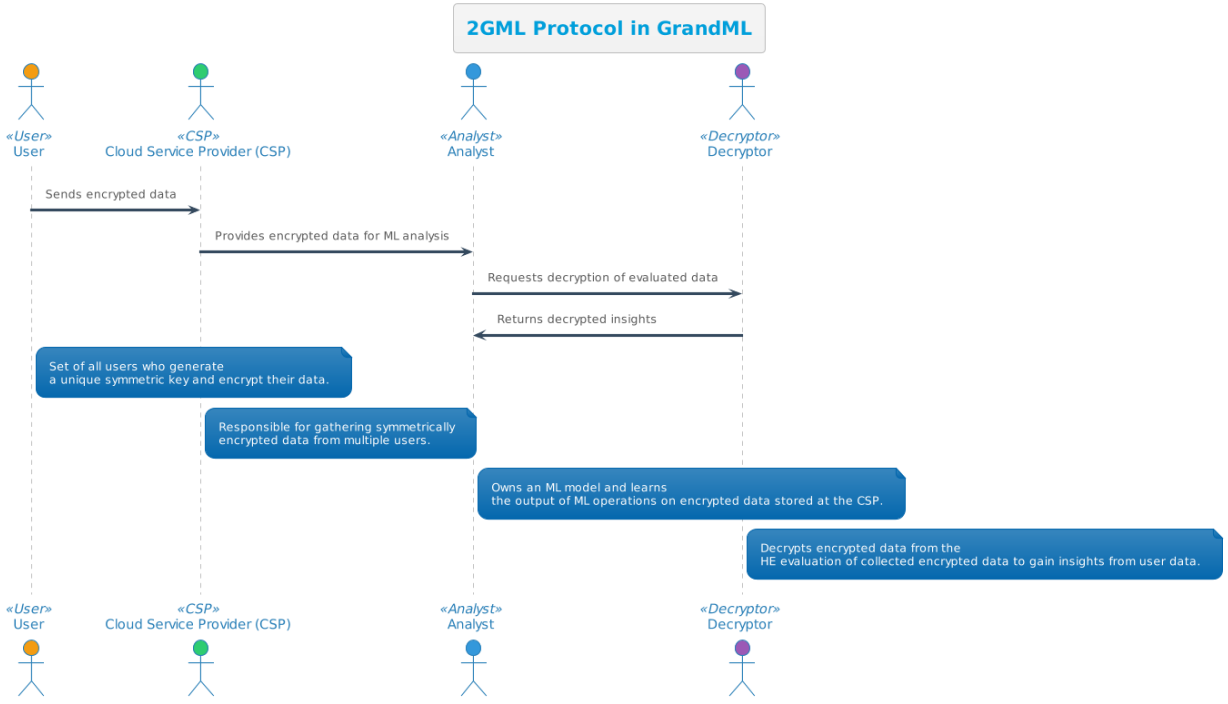

Fig. 2. Parts of the 2GML Protocol, Presumptions about Security, and Appropriate Use Cases.

TABLE II. PROCESS STAGES OF 2GML APPROACH.

Phase	Description
2GML.Setup	• User uiu_iui generates HHE keys (pkui, skui, evkui) and shares pkui with CSP while sending evkui separately. • CSP generates its PKE key pair (pkCSP, skCSP). • User uiu_iui signs and CSP verifies the setup message m1m1m1.
2GML.Upload	• User uiu_iui encrypts data xix_ixi with SKE.Enc using a symmetric key Ki, producing ciphertexts cxicxicxi and cKicKicKi. • User uiu_iui homomorphically encrypts Ki into cKicKicKi with HHE.Enc. • User uiu_iui signs and CSP verifies the upload message m2m2m2.
2GML.Eval	• CSP decrypts cxicxicxi into c′xic'xic′xi with HHE.Decomp. • CSP uses c′xic'xic′xi, ML model parameters (w,bw, bw,b), and evkui in HHE.Eval to compute crescrescres. • CSP signs and sends crescrescres to uiu_iui in m3m3m3.
2GML.Classify	• User uiu_iui decrypts crescrescres with HHE. • Dec to obtain resresres, the prediction.

3.3. Model of Attack and Security Analysis

GuardML's security can be assessed utilizing an attack model predicated on the adversary's capabilities; for instance, ADV may execute several assaults to undermine protocol security and privacy. Diverse attack methodologies, encompassing algebraic approaches such as Linearization and Gröbner Basis assaults, alongside statistical techniques like differential and linear assaults, have failed to compromise GuardML's cryptographic framework. The threat model focuses on the communication between entities within the protocol rather than the cryptographic system itself. Although mitigating the risk of basic man-in-the-middle attacks, ADV can compromise the Cloud Service Provider (CSP) and many users. The Ciphertext Substitution Attack and the ML Model Unauthorized Access Attack are two potential threats. In these attacks, the perpetrator unlawfully acquires access to the CSP's or analyst's ML model and substitutes the generated ciphertexts with undetectable alternatives.

TABLE III. A CRITICAL REVIEW OF THE 2GML PROTOCOL FOR GUARDML SECURITY

Attack Type	Description	Security Assurance
Ciphertext Substitution Attack	ADV attempts to replace genuine ciphertexts in 2GML.Upload or 2GML.Eval phases with indistinguishable fabricated ones.	EUF-CMA secure signature scheme (σ) ensures forgery resistance; negligible probability of success.
ML Model Unauthorized Access Attack	ADV colludes with users or compromises CSP to gain unauthorized access to the multi-layered ML model (f) used in 2GML.	Security relies on the complexity of multi-layered ML models and the semantically secure HE scheme.

4. RESULTS

The study evaluated encrypted inference in plaintext ECG data using floating-point and integer arithmetic across experiments with varying numbers of data inputs. The ecgPPML framework was evaluated for its effectiveness, focusing on high accuracy regardless of data volume and type. The results showed that the ecgPPML framework's efficiency and reliability were mainly focused on the high level of accuracy, even with newly implemented homomorphic encryption noise. The accuracy loss in comparison with plaintext techniques was minimal, even with newly implemented homomorphic encryption noise. The ecgPPML framework addresses specific issues in privacy-preserving machine learning, produces robust performances, and guarantees data protection, making it suitable for various real-world applications. The accuracy varied among different cases, with the ecgPPML framework maintaining robust performance even with larger datasets. The results provide a wide panoramic view of how the ecgPPML addresses specific issues in privacy-preserving machine learning, producing robust performances and ensuring data protection, making it suitable for various real-world applications.

In general, Table 4 delivers a wide panoramic view of how the ecgPPML addresses specific issues in privacy-preserving machine learning, produce robust performances and simultaneously guarantee the data protection, and therefore is fit for various real-world applications.

TABLE IV. EXAMINING ACCURACY WITH ECGPPML

AccuracyAnalysis

Accuracy Analysis - ecgPPML

Data Input	Plaintext (Float)	Plaintext (Integer)	Encrypted
1	100 %	100 %	100 %
10	90 %	90 %	90 %
20	90 %	95 %	90 %
50	88 %	92 %	90 %
100	86 %	91 %	90 %
500	87 %	87.2 %	86.8 %
1000	87.9 %	87.3 %	87.4 %
2000	88.2 %	87.4 %	87.55 %

Figure 3 illustrates the efficacy of 2GML in performing secure machine learning tasks. In point (a), we used the SKE approach to encrypt a symmetric key; the total result is that time. Due to the increased computational burden per data point, this time grows in relation to the input dimensions. The evaluation phase decryption, which involves breaking EKCT into sub-plaintexts, likewise increases dramatically when the inputs are numerous. It is efficient to decrypt results from homomorphic encrypted outputs because the amount of time it takes to do elliptic curve point multiplication is independent of the number of inputs, regardless of how difficult the computational issue is. A crucial component in the creation of secure data handling in sensitive applications is the capacity of the protocol to execute calculations on the encrypted data and address performance difficulties. This capability is illustrated in the following image. As shown in Figure 3.

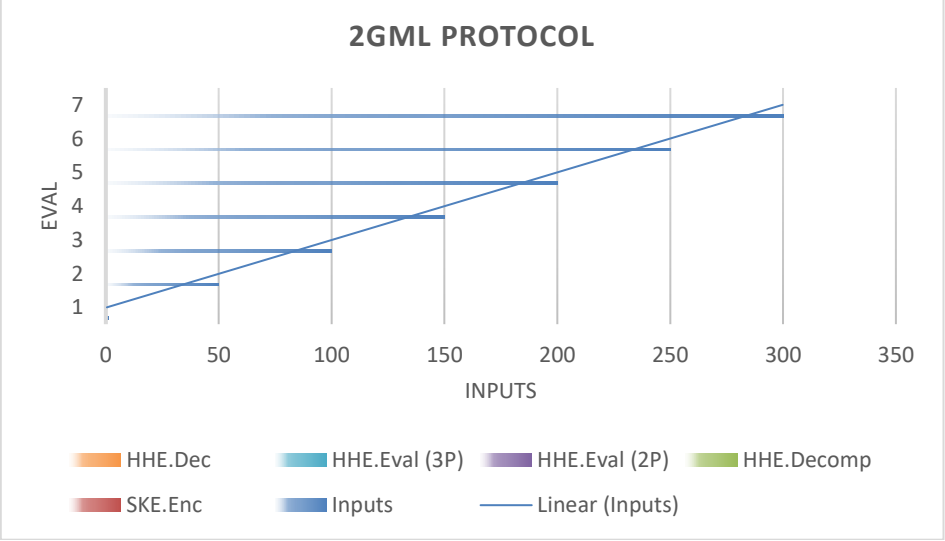

Fig. 3. outcome of the 2GML procedure.

Despite its user-friendliness, the 2GML framework requires a lot of processing power. Among the many processes involved are setup, data uploading, assessment, and data classification. However, setting everything up only takes 243 milliseconds, whereas uploading takes 607 milliseconds. After 300 data inputs, the assessment step takes 3597.7 seconds to finish on the server. Classification takes 900 milliseconds from the user's perspective. The overall efficiency and speed of the 2GML framework are enhanced by these operations, as shown in Table 5.

TABLE V. THERE ARE 300 DATA INPUTS WITH 2GML.

PhaseTimes

Phase Times

Phase	: User	: Server	: Total
2GML.Setup	: 243 ms	: 0	: 243 ms
2GML.Upload	: 607 ms	: 0	: 607 ms
2GML.Eval	: 0	: 3597.7 s	: 3597.7 s
2GML.Classify	: 900 ms	: 0	: 900 ms

5. CONCLUSION

This paper presents the PPML approach, which has been created independently, through HHE. This strategy aims to encompass many PPML methodologies and domains, including pervasive computing, by delivering optimal machine learning capabilities while safeguarding user privacy. The authors addressed the fundamental challenges of data collecting and administration at devices with constrained processing capacity, such as IoT sensors and mobile devices, by using HHE. This approach ensures robust security with minimal impact on the execution of machine learning. This paper elucidates the interplay between cryptography and machine learning in delivering efficient and secure privacy-preserving machine learning (PPML) services across various architectures, including cloud, edge, and resource-constrained environments. Data security and integrity are critical in various sectors, including healthcare, banking, and smart cities. This approach creates new opportunities for the development of private, secure apps across several domains. The paper's findings serve as a basis for future progress in privacy-preserving machine learning (PPML), facilitating the development of efficient, high-quality, and safe machine learning systems.

Conflicts Of Interest

The author's disclosure statement confirms the absence of any conflicts of interest.

Funding

No financial contributions or endorsements from institutions or sponsors are mentioned in the author's paper.

Acknowledgment

The author acknowledges the support and resources provided by the institution in facilitating the execution of this study.

References

[1] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Stanford University, 2009.

[2] Z. A. Abbood, N. A. F. Abbas, and B. Makki, "Spectrum Sensing Utilizing Power Threshold and Artificial Intelligence in Cognitive Radio," Int. J. Robot. Control Syst., vol. 2, no. 4, pp. 628–637, 2022, DOI: 10.31763/ijrcs.v2i4.771.

[3] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachene, "Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds," in Advances in Cryptology–ASIACRYPT 2016: 22nd Int. Conf. on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, Dec. 2016, pp. 3–33, Springer.

[4] Z. Brakerski, "Fully homomorphic encryption without modulus switching from classical GapSVP," in Annual Cryptology Conf., Springer, 2012, pp. 868–886.

[5] J. Fan and F. Vercauteren, "Somewhat practical fully homomorphic encryption," Cryptology ePrint Archive, 2012.

[6] T. Khan, A. Bakas, and A. Michalas, "Blind faith: Privacy-preserving machine learning using function approximation," in 2021 IEEE Symp. on Computers and Communications (ISCC), 2021, pp. 1–7.

[7] T. Khan and A. Michalas, "Learning in the Dark: Privacy-Preserving Machine Learning using Function Approximation," 2023.
[8] T. Khan, K. Nguyen, and A. Michalas, "A More Secure Split: Enhancing the Security of Privacy-Preserving Split Learning," in Nordic Conf. on Secure IT Systems, Springer, 2023, pp. 307–329.
[9] J.-W. Lee, H. Kang, Y. Kim, et al., "Privacy-Preserving Machine Learning with Fully Homomorphic Encryption for Deep Neural Networks," IEEE Access, vol. 10, pp. 30039–30054, 2022, DOI: 10.1109/ACCESS.2022.3148782.
[10] J. Cho, J. Ha, S. Kim, et al., "Transciphering Framework for Approximate Homomorphic Encryption," in ASIACRYPT 2021, Springer, pp. 640–669, DOI: 10.1007/978-3-030-92075-3_20.
[11] J. Ha, S. Kim, B. Lee, et al., "Rubato: Noisy Ciphers for Approximate Homomorphic Encryption," in EUROCRYPT 2022, Springer, pp. 581–610, DOI: 10.1007/978-3-030-92537-6_20.
[12] C. Dobraunig, L. Grassi, L. Helminger, et al., "PASTA: A Case for Hybrid Homomorphic Encryption," Trans. on Cryptographic Hardware and Embedded Syst., vol. 2023, no. 3, pp. 97–128, 2023, DOI: 10.46586/tches.v2023.i3.97-128.
[13] A. Bakas, E. Frimpong, and A. Michalas, "Symmetrical Disguise: Realizing Homomorphic Encryption Services from Symmetric Primitives," in Security and Privacy in Communication Systems, Springer, 2022, pp. 353–370.
[14] O. Cosseron, C. Hoffmann, P. Méaux, et al., "Towards Case-Optimized Hybrid Homomorphic Encryption: Featuring the Elisabeth Stream Cipher," in ASIACRYPT 2023, Springer, pp. 32–67.
[15] S. Sav, A. Pyrgelis, J. R. Troncoso-Pastoriza, et al., "POSEIDON: Privacy-Preserving Federated Neural Network Learning," in NDSS 2021, DOI: 10.14722/ndss.2021.24222.
[16] Q. Lou, B. Feng, G. C. Fox, and L. Jiang, "Glyph: Fast and Accurately Training Deep Neural Networks on Encrypted Data," in Advances in Neural Information Processing Systems, vol. 33, pp. 9193–9202, 2020.
[17] E. Hesamifard, H. Takabi, M. Ghasemi, and R. N. Wright, "Privacy-Preserving Machine Learning as a Service," in Proc. on Privacy Enhancing Technologies, vol. 2020, no. 3, pp. 123–142, 2020.
[18] A. Sanyal, M. Kusner, A. Gascon, and V. Kanade, "TAPAS+: Tricks to Accelerate Prediction as a Service on Encrypted Data," in Mach. Learn. Syst., 2022, DOI: 10.1007/s10994-022-06227-1.
[19] A. A. Badawi, C. Jin, J. Lin, et al., "Towards the AlexNet Moment for Homomorphic Encryption: HCNN, the First Homomorphic CNN on Encrypted Data with GPUs," IEEE Trans. Emerg. Topics Comput., vol. 9, no. 3, pp. 1330–1343, 2021, DOI: 10.1109/TETC.2021.3065413.
[20] Q. Lou and L. Jiang, "Efficient Privacy-Preserving Deep Learning for Secure Inference and Training," in Advances in Neural Information Processing Systems, vol. 35, pp. 2451–2462, 2022.
[21] K. Nguyen, T. Khan, and A. Michalas, "Split Without a Leak: Reducing Privacy Leakage in Split Learning," in 19th EAI Int. Conf. on Security and Privacy in Communication Networks (SecureComm'23), 2023.
[22] A. Bakas and A. Michalas, "Exploring Lightweight Hybrid Homomorphic Encryption for Mobile AI Systems," Cryptographic Innovations J., 2023.
[23] Z. A. Abbood, D. Ç. Atilla, and Ç. Aydin, "Enhancement of the Performance of MANET Using Machine Learning Approach Based on SDNs," Optik, vol. 272, p. 170268, 2023, DOI: 10.1016/j.ijleo.2022.170268.
[24] A. Canteaut, S. Carpov, C. Fontaine, T. Lepoint, M. Naya-Plasencia, P. Paillier, and R. Sirdey, "Stream ciphers: A practical solution for efficient homomorphic-ciphertext compression," J. Cryptol., vol. 31, no. 3, pp. 885–916, 2018.