Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 28 (2025), No. 5-A, pp. 1517–1526
DOI : 10.47974/JDMSC-2150

Homomorphic encryption performance in secure data processing for machine learning

V. Dankan Gowda *
Department of Electronics and Communication Engineering
BMS Institute of Technology and Management
Bangalore
Karnataka
India

V. Praveena †
Department of Computer Science and Engineering
Dr NGP Institute of Technology
Coimbatore
Tamil Nadu
India

M. Nagabhushanam §
Department of Electronics and Communication Engineering
M S Ramaiah Institute of Technology
MSR Nagar, MSRIT Post
Bengaluru
Karnataka
India

Pullela SVVSR Kumar ‡
Department of Computer Science & Engineering
Aditya University
Surampalem
Andhra Pradesh
India

* *E-mail:* dankan.v@bmsit.in (Corresponding Author)
† *E-mail:* drvpraveena@gmail.com
§ *E-mail:* nagabhushanam1971@msrit.edu
‡ *E-mail:* pullelark@yahoo.com

Christian Rafael Quevedo Lezama [@]
Department of Electrical and Electronics Engineering
Universidad Nacional Mayor de San Marcos
Avenida Carlos Germán Amezaga 375 - Cercado de Lima
Peru

Disha Pathak [#]
Department of Finance
Symbiosis Centre for Management Studies
Hyderabad
Telangana
India

and

Symbiosis International (Deemed University)
Pune
Maharashtra
India

Abstract

In terms of data privacy and security especially in the development of machine learning HE has the potential to solve it. This paper focuses on homomorphic encryption as a method for secure data processing with reference to encryption techniques that allow arithmetic operations to be performed on encrypted data for use in machine learning. When comparing different HE methods, their computational complexity, as well as the obtained precision, have been analyzed, and practical suitability of the methods in question has been discussed, that is, what actual problem-solving can be done using the methods in question. That was done in a way that does not in any way encroach into the security of the data to ensure that we do not compromise it in the proposed method used to incorporate HE with machine learning models. The provided findings of experiments show that the application of HE in experiments secured data is valid while establishing the cost between security measures and computation time. Chen's works provide knowledge on how more efficient, personal sensitive preserving machine learning can be developed and integrated.

Subject Classification (2010): *94A60, 20C05, 20C07.*

Keywords: *Homomorphic encryption, Secure data processing, Machine learning, Privacy-preserving, Computational efficiency, Encryption schemes, Data confidentiality, Performance evaluation.*

[@] *E-mail:* **christian.quevedo1@unmsm.edu.pe**
[#] *E-mail:* **disha.pathak@scmshyd.siu.edu.in**

1. Introduction

Together with the increase in the amount of data and the development of possibilities to use ML in different fields, many areas have been changed and opened the opportunity for further automation, prediction, and decision-making[1]. However, the exposure of involving personal, financial as well as health information poses a high risk especially in the area of privacy[2]. The traditional approaches of encrypting data protect data at the time of storing and even at the time of transferring but for computation, they have to be decrypted and are hence at risk[3]. Homomorphic encryption (HE) eradicates this problem by allowing computations on encrypted data with the end getting encrypted results that can only be decrypted to get actual values[4]. This capability preserves data privacy throughout the processing life cycle[5]; therefore, making HE suitable for solving secure data processing in the ML applications[6]. Furthermore, the integration of HE with ML algorithms[7] creates difficulties in controlling the encrypted arithmetic because some operations are complex and render a confined function[8]. Such challenges require the evaluation of HE's efficiency to determine the information-processing efficiency in HE for its applications in operation[9]. The objective of this paper is to identify any HE schemes that are both secure and relatively computational and evaluate the impact on the resulting ML model alongside identifying the potential applications[10,11]. The scope relates to the comparison of different forms of HE to each other in regards to their effectiveness for learning and the possibility of integrating them with ML, as well as assessing their viability.

2. Literature Survey

Several methods and techniques have been proposed since the time HE was first conceptualized for improving the practicality of HE. Fully HE, which allows any operations on encrypted data, is the ultimate realization of HE but bear heavy computational cost[12,13]. Efficient and proper FHE is possible due to techniques like Gentry scheme, BGV, and CKKS that enhanced the applicability of deep learning techniques[14]. Within the domain of secure computation specifically for machine learning in the present state[15], there is a substantial trend for the use of HE for providing data privacy[16]. Studied experience has confirmed the effectiveness of HE in different operations of ML in the training process and inferences with data protection[17]. Nonetheless, due to the overhead

of computation, the efficiency of HE is still a crucial issue, especially in the context of constrained environments[18]. However, the following gaps and limitations are evident in the current research works still: As for the limitations, one can consider the unavailability of HE for handling large datasets for training larger scale ML models since the computational complexity steeply rises with the data size. On the same note, the leverage of HE with the latest sophisticated networks such as deep learning has an enormous potentiality that has not been fully explored. There is also the problem of lack of standard measures that could be used to compare the performance of HE in different applications. Filling these gaps is critical for the furthering of practical application of HE in secure ML processes.

3. Proposed Method

The promised approach relies on FHE to implement encrypted data processing in the ML applications. The choice of our approach is based on the efficiency of the CKKS (Cheon-Kim-Kim-Song) scheme because many operations in ML involve approximate computations. Preprocessing is done on the raw data through encryption with the CKKS scheme to before the ML computations are Carried out on the data in its encrypted form. The results produced are also encrypted before being decrypted to yield the final output, hence preserving the data's confidentiality. Figure 1 presents the proposed system, specifically for the homomorphic encryption method for secure data processing in ML.

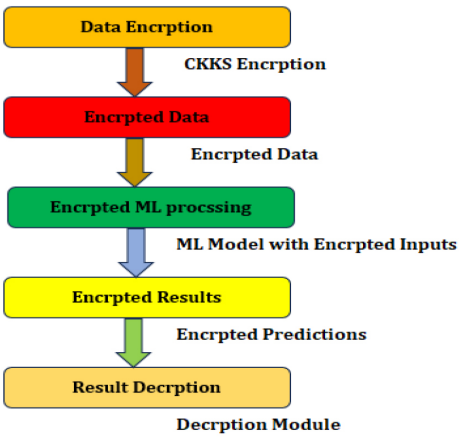

Figure 1

Proposed System Flowchart

This figure is a general representation of the whole process indicating the main processing stages and elements. The data encryption module employs the CKKS scheme to encrypt the raw data so that the data is protected from any processing medium to which it will be subjected. For the data level security, an encrypted ML processing module is provided and this contains the ml model that has been redesigned to do computations on encrypted data. At last, there is the decryption module which decrypted the processed data to yield the final result.

4. Algorithm

In the case of homomorphic encryption as used in this study, CKKS is preferred because of its usefulness in applying approximate calculations that are characteristic of machine learning. The pseudocode for the CKKS-based encryption and decryption process is as follows:

Pseudocode:

Step 1: Key Generation

Generate public and secret keys: (pk, sk)

Step 2: Encryption

Input: Plaintext data x

Encrypt x using the public key pk to obtain ciphertext c

c = Encrypt (pk, x) = $x + e \cdot q$ (mod q). where e is an error term, and q is a large modulus.

Step 3: Homomorphic Operations

Perform arithmetic operations (addition and multiplication) on ciphertexts

$$\text{Csum} = \text{c1} + \text{c2} \pmod{\text{q}}$$

$$\text{cprod} = \text{c1} \times \text{c2} \pmod{\text{q}}$$

Step 4: Decryption

Decrypt ciphertext ccc using the secret key sk

$$x = \text{Decrypt}(sk, c)$$

Key Generation involves creating a pair of keys: They used one key, called a public key for encrypting the information and the second, secret key to decrypt it. Encryption is when plaintext data is converted to ciphertext by the use of the public key so that the data will be secure.

Homomorphic Operations allow arithmetic calculations on the encrypted data and produce encrypted outcomes. Decryption reverses the encryption process to convert the obtained ciphertext back to plaintext with the help of the secret key.

5. Results & Discussion

The simulation setup was based on adopting CKKS homomorphic encryption with different models of ML incorporated in it to benchmark. For these experiments the datasets were consisted from the synthetic data sizes varied from 100KB to 1000KB to cover the large-scale real data processing. These were latency in both encryption and decryption, accuracy of models trained and tested with both encrypted and plaintext data, and computational cost in terms of the additional operations involved in each homomorphic operation; addition, multiplication and exponentiation among others. The results were compared with conventional encryption technique, the unencrypted data processing method for comparison of results. The Figure 2 presents the impact of data size on the time required for the encryption/decryption process when utilizing homomorphic encryption. The HE for secure data in machine learning is provided in Figure 3. where two plots are included for the evaluation of efficiency and performance characteristics. In Figure 3(a), the primary focus is to depict how the Model accuracy looks like when encryption on the data is done or when no encryption is made on the data given the data size. Figure 3(b) is related to computational expenses associated with different homomorphic operations. Figure 4 presents a comparative analysis of three different encryption schemes: FHE, SHE, PHE are defined as the types of HE.

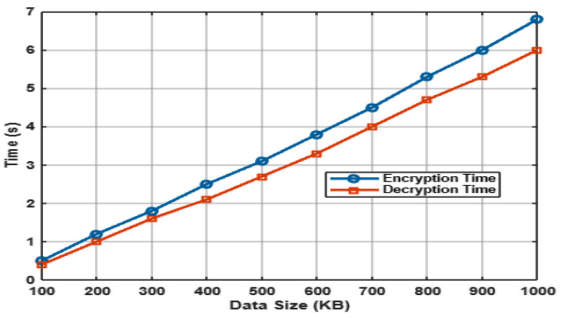

Figure 2

Encryption and Decryption Time vs. Data Size

Figure 5 presents the differences on the number of training time as impacted by homomorphic encryption to the learning algorithms. The x-axis puts into use five machine learning models referred to as model 1, model 2, model 3, model 4, and model 5 while the y-axis presents the training time in minutes.

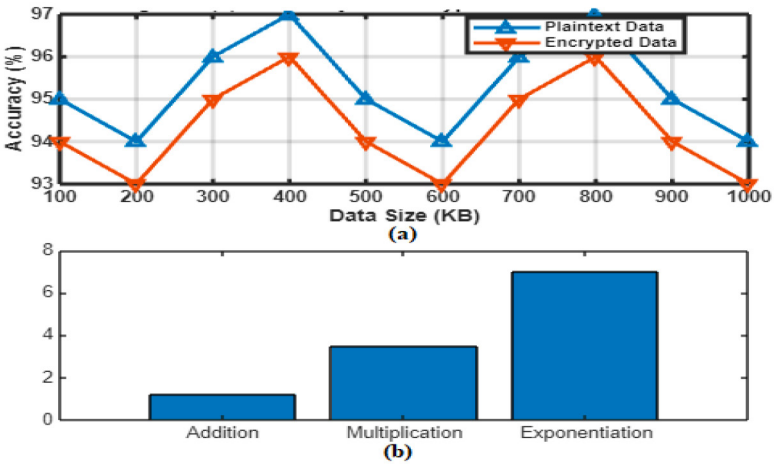

Figure 3

(a) Accuracy vs. Encrypted vs. Plaintext Data & (b) Computational Overhead of Homomorphic Operations.

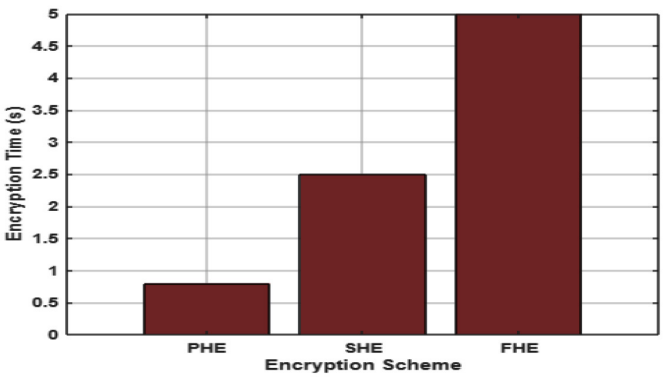

Figure 4

Comparison of Encryption Schemes

To summarize the results achieved in this work, it is necessary to emphasize the following: The outcomes that have been derived herein bear exciting implications to the safety of data in handling along with the

corresponding machine learning. But in homomorphic encryption there is the provision of data privacy since the data is encrypted before the operation is carried out; though, this consumes a lot of time. But it is justified particularly if the program needs to protect data and doesn't need to work with time-sensitive outputs.

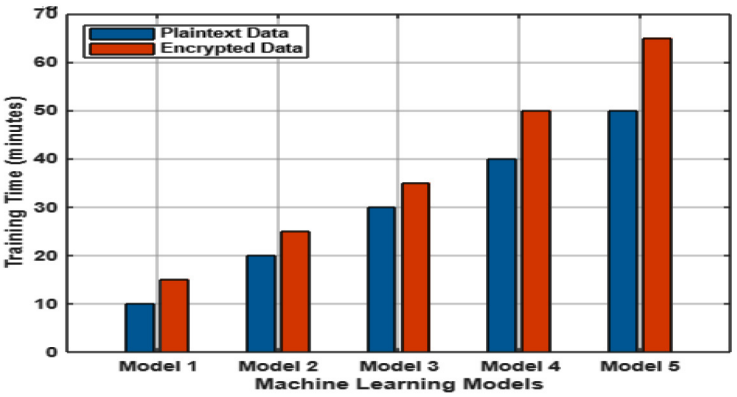

Figure 5

Homomorphic Encryption Impact on Model Training Time

From the above comparative analysis, it was found out that HE can be easily incorporated in the conventional machine learning processing if more computational complexity is to be regarded. However, it is clear from the study that there is a step to select the most appropriate encryption scheme based on the characteristics of the permissible application necessary and sufficient criteria from the perspectives of security and performance of the system.

6. Conclusion

In the paper, the accuracy of using homomorphic encryption for secure data processing of machine learning tasks was examined with the emphasis on the CKKS scheme. The experimental results showed that though, homomorphic encryption effectively comes with a privacy guarantee, it is however costly in terms of computation. Encryption slightly affected the model's accuracy; however, the performance remained at a high level, similar to that for plaintext data. When examining the amount of overhead for various homomorphic operations, it was discovered that addition had the smallest amount of overhead while exponentiation was

the most computationally expensive. The comparative analysis of the encryption techniques pointed out the fact that even though FHE provides the highest level of functionality by allowing homomorphic evaluation of functions on the encrypted data, it has the highest computation overhead compared with PHE and SHE. As with the test set accuracy, model training times also rose when operating on encrypted data; more efficient methods of encrypting the data or more computing resources may be necessary.

References

[1] J. Fan and F. Jiang, "Efficient Fully Homomorphic Encryption from (Standard) LWE," *IEEE Trans. Inf. Theory,* vol. 68, no. 1, pp. 262-280 (2022).

[2] L. Dai, Q. Xu, and C. Shen, "Homomorphic Encryption-Based Privacy-Preserving Data Aggregation for Wireless Sensor Networks," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1080-1089 (2022).

[3] A. Acar, H. Aksu, and A. S. Uluagac, "A Survey on Homomorphic Encryption Schemes: Theory and Implementation," *ACM Comput. Surv.,* vol. 51, no. 4, pp. 1-35 (2018).

[4] C. Gentry, S. Halevi, and N. P. Smart, "Homomorphic Evaluation of the AES Circuit," in Advances in Cryptology – CRYPTO 2012, Springer, Berlin, Heidelberg, pp. 850-867 (2012).

[5] W. Li, Y. Shi, and Y. Zhang, "Privacy-Preserving Data Mining on Social Networks: A Survey," *IEEE Access*, vol. 8, pp. 9437-9456 (2020).

[6] Z. Yang, S. Zhu, and X. Jiang, "Privacy-Preserving Deep Learning on Mobile Devices: A Survey," *IEEE Commun. Surv. Tutor.,* vol. 21, no. 3, pp. 1994-2019 (2019).

[7] H. Wu, X. Zhang, and Y. Zhang, "Secure Data Aggregation Technique Based on Homomorphic Encryption in Wireless Sensor Networks," *IEEE Access*, vol. 8, pp. 32523-32534 (2020).

[8] G. Manivasagam and K. D. V. Prasad, "Novel approaches to biometric security using enhanced session keys and elliptic curve cryptography," *J. Discrete Math. Sci. Cryptogr.*, vol. 27, no. 2-A, pp. 477–488 (2024).

[9] R. Senthil Kumar and K. D. V. Prasad, "Securing digital authentication with cryptographic innovations in intrinsic verification systems," *J. Discrete Math. Sci. Cryptogr.*, vol. 27, no. 2-A, pp. 317–328 (2024).

[10] K. D. V. Prasad and T. Thomas, "Cryptographic image-based data security strategies in wireless sensor networks," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 293–304, 2024.

[11] P. Sarma and M. Rahman, "Mathematical analysis of wavelet-based multi-image compression in medical diagnostics," *J. Discrete Math. Sci. Cryptogr.,* vol. 27, no. 2-B, pp. 675–687 (2024).

[12] A. Kumar, "A novel approach of unsupervised feature selection using iterative shrinking and expansion algorithm," *J. Interdiscip. Math.,* vol. 26, no. 3, pp. 519-530 (2023).

[13] Veera Sivakumar, "A novel RF-SMOTE model to enhance the definite apprehensions for IoT security attacks," *J. Discrete Math. Sci. Cryptogr.,* vol. 26, no. 3, pp. 861-873 (2023).

[14] A. Chaturvedi, "Securing networked image transmission using public-key cryptography and identity authentication," *J. Discrete Math. Sci. Cryptogr.*, vol. 26, no. 3, pp. 779-791 (2023).

[15] K. Raghavendra, "Vector space modelling-based intelligent binary image encryption for secure communication," *J. Discrete Math. Sci. Cryptogr.,* vol. 25, no. 4, pp. 1157-1171 (2022).

[16] Y. Huang, D. Evans, J. Katz, and L. Malka, "Faster Secure Two-Party Computation Using Garbled Circuits," in Proc. 20th USENIX Security Symp., pp. 539-554 (2011).

[17] J. Yuan and S. Yu, "Efficient Privacy-Preserving Biometric Identification in Cloud Computing," *IEEE Trans. Cloud Comput.,* vol. 8, no. 1, pp. 151-164 (2020).

[18] S. Halevi and V. Shoup, "Algorithms in HElib," in Advances in Cryptology – CRYPTO 2014, Springer, Berlin, Heidelberg, pp. 554-571 (2014).

Received November, 2024