RESEARCH ARTICLE OPEN ACCESS

A Review of Privacy-Preserving Machine Learning Methods Using Cryptography

Pankaj Sarde [1], Vaishali Sarde [2]

[1] Department of Mathematics, Rungta College of Engineering and Technology Bhilai(CG), India

[2] Department of Computer Application, Govt. J. Yoganandan Chhattisgarh College Raipur(CG), India

ABSTRACT

In the age of extensive data and artificial intelligence, protecting confidential data while allowing the valuable insights of machine learning has become of utmost importance. Privacy-preserving machine learning (PPML) is an emerging discipline that use cryptographic methods to protect data throughout the training and inference stages of machine learning models. The cryptographic techniques used in PPML, such as homomorphic encryption, safe multi-party computation, differential privacy, federated learning with cryptographic upgrades, and zero-knowledge proofs, are thoroughly examined in this review paper. The analysis of each technique includes an evaluation of its fundamental principles, practical uses, and the difficulties it encounters in achieving a balance between privacy, security, and computational efficiency. By examining the latest advancements in cryptographic methods for PPML, this review seeks to inform and direct future research efforts toward developing more robust and scalable privacy-preserving solutions for machine learning across various application domains.

Keywords — Machine learning models, Privacy-preserving machine learning (PPML), Cryptographic techniques, Homomorphic encryption, Secure multi-party computation, Differential privacy, Federated learning, Zero-knowledge proofs, Data security, Artificial intelligence.

I. INTRODUCTION

With the ongoing transformation of industries through machine learning [1, 5, 6, 9, 17, 22, 27], the need for protecting sensitive information has grown more crucial. In industries such as healthcare, banking, and personal data analytics, where the protection of privacy is of utmost importance, the potential for data breaches or unauthorized access can result in significant effects. Privacy-preserving machine learning (PPML) ensures the security of data used in training and inference, even in potentially untrusted situations. Cryptography [3, 23, 24] is essential in PPML since it gives the required tools to protect data at every stage of the machine learning process. Methods such as homomorphic encryption [6], secure multi-party computation [18], and differential privacy [9] enable the processing of machine learning models on encrypted data, preventing the underlying information from being revealed. In addition, federated learning [5] and zero-knowledge proofs [4] provide novel methods for collectively training models or validating computations while preserving the confidentiality of sensitive data. This paper provides an overview of the present condition of cryptographic techniques employed in Privacy-Preserving Machine Learning (PPML), with a specific emphasis on their fundamental concepts, real-world applications, and the barriers they encounter. Through the analysis of these methods, our objective is to emphasize the significance of cryptography in the creation of reliable and effective machine learning systems. Additionally, we aim to identify specific areas that require additional investigation and innovation to meet the changing requirements of privacy and security in the era of artificial intelligence.

The organization of the paper is as follows:

Section 2 provides a detailed overview of Homomorphic Encryption (HE), including its principles, applications, and associated challenges.

Section 3 explores Secure Multi-Party Computation (SMPC), outlining how this technique facilitates secure computations among multiple parties while preserving privacy.

Section 4 covers Differential Privacy (DP), focusing on its methods for safeguarding individual data contributions and maintaining model utility.

Section 5 discusses Federated Learning with Privacy Enhancements, examining how cryptographic techniques are employed to secure collaborative model training across decentralized data sources.

Section 6 addresses Zero-Knowledge Proofs (ZKP), explaining their use in verifying computations without disclosing sensitive information.

Section 7 concludes the paper with a summary of the findings and suggestions for future research and advancements in privacy-preserving cryptographic methods for machine learning.

II. HOMOMORPHIC ENCRYPTION (HE) [6, 7, 12, 13, 15, 28]

Homomorphic Encryption (HE) is a cryptographic method that enables computations to be carried out on encrypted data without requiring prior decryption. This characteristic makes HE particularly effective in privacy-preserving machine learning (PPML) and other situations where data privacy is of utmost importance.

2.1 Key Concepts of Homomorphic Encryption

2.1.1 Encryption and Decryption

Similar to conventional encryption techniques, Homomorphic Encryption (HE) entails the process of encrypting plain data to generate cipher data. However, Homomorphic Encryption (HE) systems enable specific operations, such as addition and multiplication, to be executed on the encrypted data. Once decrypted, the result of these operations is identical to that obtained when the operations were conducted on the plaintext.

2.1.2 Types of Homomorphic Encryption

- **Partially Homomorphic Encryption (PHE)**: Only supports a singular type of operation (either addition or multiplication) on encrypted data. ElGamal encryption and the RSA cryptosystem are two examples.
- **Somewhat Homomorphic Encryption (SHE)**: Performs both addition and multiplication operations, but is limited to a specific level of complexity.
- **Fully Homomorphic Encryption (FHE)**: Allows for infinite multiplication and addition on ciphertexts. The strongest type is FHE, but it also requires the most processing power.

2.2 Applications

- PPML utilizes the HE technique to train models on encrypted datasets, ensuring the preservation of data privacy throughout the processing phase.

- Users have the option to delegate the storage and processing of encrypted data to cloud services, ensuring that the data remains secure and confidential during computations.

- HE enables secure querying of encrypted databases, ensuring that both queries and results are kept encrypted, thus ensuring data privacy.

2.3 Challenges:

- **Performance:** Specifically, FHE is slower than conventional encryption techniques due to its large computational overhead. This creates an important difficulty to its broad adoption.
- **Complexity**: For developers, this can be a barrier because of the mathematical complexity of HE schemes, which demands extensive understanding and execution.
- **Security and Key Management**: Homomorphic encryption key security is an important problem that requires careful consideration, particularly in distributed systems.

III. SECURE MULTI-PARTY COMPUTATION (SMPC) [8, 14, 18, 29]

Secure Multi-Party Computation (SMPC or MPC) is a cryptographic technique that allows many participants to collaboratively calculate a function using their inputs while ensuring the privacy of those inputs. The fundamental concept is that no individual party acquires any additional knowledge beyond their own input and the output of the computation, regardless of their lack of trust in one another.

3.1. Key Concept of Secure Multi-Party Computation (SMPC)

- **Semi-Honest (Passive) Adversary**: Assumes that parties observe the protocol exactly while attempting to extract more details from the messages they receive.
- **Malicious (Active) Adversary**: Assumes that parties have the ability to depart from the protocol in a random manner in order to obtain additional information or disrupt the calculation process.

3.2. Security Guarantees:

- **Privacy:** Ensures the confidentiality of the parties' inputs.
- **Correctness**: Ensures the accuracy of the computation's results, even in the presence of malevolent behavior by certain parties.
- **Fairness:** Prevents one party from learning the result while others do not by guaranteeing that either all parties receive the output or none do.

3.3. Applications of SMPC

- **Privacy-Preserving Data Analysis**: SMPC is used in cooperative research and joint statistics collection to analyze data from various sources without disclosing the underlying data.
- **Financial Computations**: Parties may compute financial functions such as auctions, benchmarking, or risk analysis without disclosing their confidential information.
- **Secure Voting**: Without disclosing specific votes, SMPC can guarantee that the votes are counted accurately.

IV. DIFFERENTIAL PRIVACY (DP) [1, 2, 10, 11]

Differential Privacy (DP) is a robust mathematical framework that guarantees the confidentiality of people in a dataset while enabling the retrieval of valuable statistical insights. It ensures that the output of a computation does not disclose an excessive amount of information about a particular individual, even if an adversary has access to other data sources.

4.1 Key Concepts of Differential Privacy

- **Privacy Guarantee**: Differential Privacy guarantees that the inclusion or exclusion of any individual's data in a dataset has minimal impact on the result of a computation. This creates a challenge for an opponent to deduce whether the dataset contains the data of a certain individual.
- **Noise Addition**: Usually, noise is added to the output of a function that is computed on the dataset in order to establish differential privacy.
- **Composition**: Since differential privacy is composable, when numerous calculations are carried out on the same dataset, the privacy guarantees deteriorate gradually. The total privacy loss is the sum of the ϵ values from each computation.

4.2 Applications of Differential Privacy

- **Machine Learning**: Differential privacy is utilized to train models on sensitive data, guaranteeing that the predictions made by the model do not disclose any information about individual training examples.
- **Data Sharing**: Organizations have the ability to distribute collective statistics or artificially generated datasets that ensure differentiated privacy, enabling data analysis without affecting the privacy of individuals.

V. FEDERATED LEARNING WITH PRIVACY ENHANCEMENTS [5, 16, 17, 19, 20]

Federated Learning (FL) is a distributed method of machine learning in which numerous clients (such as devices or organizations) work together to train a common model while maintaining their data locally. This approach enables training of models using data from multiple sources without the requirement to consolidate the data, hence improving privacy and minimizing expenses associated with data transfer.

5.1 Key Concepts of Federated Learning

- **Decentralized Data Storage:** The training data in FL is stored on the client devices, which might be edge servers, cellphones, or Internet of Things sensors. Each device trains the model locally, and only sends model changes (gradients or model weights) to a central server. This decentralized method guarantees that the original data remains on the device, greatly improving privacy.
- **Federated Averaging:** The basic FL algorithm, Federated Averaging, was first presented by McMahan et al. (2017). To create a single global model, it aggregates the locally learned models from several devices. Local computations are made by each device, and the central server averages them to update the global model.
- **Cryptographic Enhancements**: Combining federated learning with SMPC, HE, or DP to further protect privacy.

5.2 Applications of Federated Learning with Privacy Enhancements

- **Healthcare:** Without disclosing private information, hospitals and research centers can work together to train machine learning models with patient data.
- **Finance:** Financial institutions should engage in collaborative efforts to develop models that can identify fraudulent transactions by sharing valuable insights derived from local data, while ensuring the protection of sensitive consumer information.
- **Smartphones and Mobile Applications:** FL is employed to train models for predictive text input, autocorrect, and personalized keyboard recommendations on a large scale, while ensuring that user typing data is stored on the device.
- Federated Learning with privacy enhancements can be used in any field where confidentiality and data sensitivity are important considerations. In companies handling sensitive data, it creates potential for innovation by enabling collaborative learning while maintaining privacy.

VI. ZERO-KNOWLEDGE PROOFS (ZKP) [4, 14, 21, 25, 26]

Zero-Knowledge Proofs (ZKP) are cryptographic techniques that enable a party (the "prover") to prove to another party (the "verifier") that they are aware of a particular piece of information without actually disclosing the details. Researchers Goldwasser, Micali, and Rackoff established this hypothesis in the 1980s.

6.1 Key Concepts of Zero-Knowledge Proofs

- **Zero-Knowledge Property:** The fundamental concept of ZKP is that the verifier is not provided with any knowledge of the actual information, except for the fact that the prover is aware of it.
- **Completeness**: If the assertion is true and both the prover and the verifier agree to the protocol appropriately, the verifier will be convinced of its truth.

- **Soundness**: There is very little chance that any dishonest prover could persuade the verifier that the statement is true if it is false.
- **Zero-Knowledge**: Other than knowing that the statement is true, the verifier gains no new information. This indicates that no further information that might be utilized to deduce the secret is revealed by the prover.

6.2 Types of Zero-Knowledge Proofs

- **Interactive Zero-Knowledge Proofs**: In this format, the prover and the verifier engage in a sequence of exchanges that result in the proof. The prover is asked to do specific tasks or give specific answers by the verifier, which the prover could only complete if they are aware of the secret.
- **Non-Interactive Zero-Knowledge Proofs (NIZK)**: NIZK does not require back-and-forth communication, in contrast to interactive proofs. Rather, the prover produces an evidence that everyone can independently verify. This is very beneficial for blockchains and other distributed systems.

6.3 Applications of Zero-Knowledge Proofs

- **Privacy-Preserving Data Sharing**: ZKPs can be employed to authenticate sensitive data without disclosing the actual data, particularly in situations involving the exchange of confidential information such as medical records or financial data.
- **Voting Systems**: ZKPs can help protect the integrity and confidentiality of electronic voting systems. Voters may confirm they voted without disclosing their vote, keeping both the accuracy and privacy of the voting process.
- **Authentication**: Zero-knowledge proofs (ZKPs) can be employed in secure authentication systems, allowing users to authenticate their identity without disclosing their password or any other confidential data.

VII. CONCLUSION

In this survey, we examined different cryptographic techniques that have a crucial impact on improving privacy in machine learning. Given the growing importance of data privacy, methods such as Homomorphic Encryption (HE), Secure Multi-Party Computation (SMPC), Differential Privacy (DP), Federated Learning with Privacy Enhancements, and Zero-Knowledge Proofs (ZKP) provide effective solutions to protect sensitive information while learning. Every approach possesses unique advantages and constraints, which vary according to the particular application circumstances. Homomorphic Encryption (HE) enables computations to be performed on encrypted data, ensuring secure data processing without exposing the original information. On the other hand, Secure Multi-Party Computation (SMPC) enables secure collaborative computations among several participants without compromising the privacy of the data. Differential privacy (DP) offers a strong method for protecting individual data contributions while preserving the effectiveness of the overall model.

Federated Learning guarantees that data is distributed, and the learning process is additionally strengthened with cryptographic upgrades. Zero-knowledge proofs (ZKPs), however, provide robust mechanisms for validating computations while keeping confidential information undisclosed. Although cryptographic approaches have the potential for various applications, they also come with problems, including the demand for significant processing resources, complications in communication, and the requirement to find a balance between privacy and utility. Hence, continuous research is vital to enhance these techniques, rendering them more effective, adaptable, and feasible for real-world use. With the ongoing development of privacy-preserving machine learning, it is crucial to promptly tackle the current constraints of these cryptographic techniques. Future research ought to focus on the development of hybrid systems, enhancing the efficiency of cryptographic protocols, and investigating novel strategies that can more effectively address the varied privacy needs of machine learning applications.

REFERENCES

[1] Abadi, M., et al. (2016). "Deep learning with differential privacy." Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 308-318.

[2] Abowd, J. M. (2018). The U.S. Census Bureau adopts differential privacy. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 2867-2867). https://doi.org/10.1145/3229553.3236600

[3] Armknecht, F., Chen, L., Katzenbeisser, S., Peter, A., Schneider, T., & Smart, N. P. (2015). Outsourced secure computation. Cryptology and Network Security, 9543, 147-166. https://doi.org/10.1007/978-3-319-26823-1_10

[4] Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., & Virza, M. (2014). Succinct non-interactive zero knowledge for a von Neumann architecture. In Proceedings of the 23rd USENIX Security Symposium (pp. 781-796). USENIX Association.

[5] Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečný, J., Mazzocchi, S., McMahan, H. B., Van Overveldt, T., Petrou, D., Ramage, D., & Roselander, J. (2019). Towards federated learning at scale: System design. In Proceedings of the 2nd SysML Conference. https://doi.org/10.48550/arXiv.1902.01046

[6] Brakerski, Z., Gentry, C., & Vaikuntanathan, V. (2014). (Leveled) fully homomorphic encryption without bootstrapping. ACM Transactions on Computation Theory, 6(3), 1-36. https://doi.org/10.1145/2633600

[7] Chillotti, I., Gama, N., Georgieva, M., & Izabachène, M. (2016). TFHE: Fast fully homomorphic encryption over the torus. Journal of Cryptology. https://doi.org/10.1007/s00145-018-9283-5

[8] Cramer, R., & Damgård, I. (2000). Secure multi-party computation. Cambridge University Press.

[9] Dowlin, N., Gilad-Bachrach, R., Laine, K., Lauter, K., Naehrig, M., & Wernsing, J. (2016). CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy. In Proceedings of the 33rd International Conference on Machine Learning (pp. 201-210).

[10] Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science, 9(3-4), 211-407. https://doi.org/10.1561/0400000042

[11] Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In Proceedings of the 3rd Theory of Cryptography Conference (TCC) (pp. 265-284). Springer. https://doi.org/10.1007/11681878_14

[12] Gentry, C. (2009). A fully homomorphic encryption scheme (Doctoral dissertation, Stanford University).

[13] Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. In Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC) (pp. 169-178).

[14] Goldwasser, S., Micali, S., & Rackoff, C. (1989). The knowledge complexity of interactive proof systems. SIAM Journal on Computing, 18(1), 186-208. https://doi.org/10.1137/0218012

[15] Halevi, S., & Shoup, V. (2014). Algorithms in HElib. In Advances in Cryptology – CRYPTO 2014 (pp. 554-571). Springer. https://doi.org/10.1007/978-3-662-44381-1_31

[16] Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., ... & Yang, Q. (2021). Advances and open problems in federated learning. Foundations and Trends® in Machine Learning, 14(1-2), 1-210. https://doi.org/10.1561/2200000083

[17] Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. IEEE Signal Processing Magazine, 37(3), 50-60. https://doi.org/10.1109/MSP.2020.2975749

[18] Lindell, Y. (2017). A practical guide to secure multi-party computation. Cryptology ePrint Archive, Report 2017/189. https://eprint.iacr.org/2017/189

[19] Lu, Y., Huang, X., Dai, Y., Maharjan, S., & Zhang, Y. (2020). Blockchain and federated learning for privacy-preserving data sharing in industrial IoT. IEEE Transactions on Industrial Informatics, 16(6), 4177-4186. https://doi.org/10.1109/TII.2019.2942190

[20] McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS 2017) (pp. 1273-1282). https://doi.org/10.48550/arXiv.1602.05629

[21] Miers, I., Garman, C., Green, M., & Rubin, A. D. (2013). Zerocoin: Anonymous distributed e-cash from bitcoin. In IEEE Symposium on Security and Privacy (SP) (pp. 397-411). https://doi.org/10.1109/SP.2013.34

[22] Mo, Y., Qu, X., Tian, F., Liu, S., & Zhang, W. (2020). DarkneTZ: Towards model privacy at the edge using trusted execution environments. In Proceedings of the 18th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys 2020) (pp. 161-174). https://doi.org/10.1145/3386901.3388930

[23] Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. In Advances in Cryptology – EUROCRYPT '99 (pp. 223-238). https://doi.org/10.1007/3-540-48910-X_16

[24] Rivest, R. L., Adleman, L., & Dertouzos, M. L. (1978). On data banks and privacy homomorphisms. Foundations of Secure Computation, 4(11), 169-180.

[25] Sahai, A., & Waters, B. (2014). Attribute-based encryption for circuits: Achieving constant-size ciphertexts with adaptive security. In Advances in Cryptology – EUROCRYPT 2014 (pp. 239-257). https://doi.org/10.1007/978-3-642-55220-5_14

[26] Sasson, E. B., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., & Virza, M. (2014). Zerocash: Decentralized anonymous payments from bitcoin. In IEEE Symposium on Security and Privacy (pp. 459-474). https://doi.org/10.1109/SP.2014.36

[27] Shokri, R., & Shmatikov, V. (2015). Privacy-preserving deep learning. In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (pp. 1310-1321). https://doi.org/10.1145/2810103.2813687

[28] Van Dijk, M., Gentry, C., Halevi, S., & Vaikuntanathan, V. (2010). Fully homomorphic encryption over the integers. In Advances in Cryptology – EUROCRYPT 2010 (pp. 24-43). https://doi.org/10.1007/978-3-642-13190-5_2

[29] Yao, A. C. (1986). How to generate and exchange secrets. In Proceedings of the 27th Annual Symposium on Foundations of Computer Science (FOCS) (pp. 162-167). IEEE. https://doi.org/10.1109/SFCS.1986.25