Privacy-Preserving Data Analysis: Perform data Analysis on Encrypted Data Stored in the Cloud

Vikram Shirol, Arun Kumar Joshi, Preeti D B
Department of Computer Science, Smt. Kamala and Sri. Venkappa M Agadi
College of Engineering and Technology, Lakshmeshwar-582116

***Abstract*: -Balancing data security with analytical performance in the analysis of encrypted data has proven to be a challenging task. An initiative called Privacy-Preserving Data Analysis uses Python to conduct safe data analysis on cloud-stored encrypted data. The purpose of this project is to reduce the possibility that private data may be discovered when analyzing the data. Many methods are used to do this, including secure multiparty computation and homomorphic encryption. The project entails putting in place a safe data processing pipeline that protects the confidentiality of the data that is stored. With homomorphic encryption, sensitive data can be computed directly on encrypted data, eliminating the need for decryption and revealing it. In this project, the Python programming language serves as the primary development tool. It's a great option because of its widespread libraries, community support, and ease of use. The project entails utilizing Python to create privacy-preserving protocols and encryption techniques and to integrate them with cloud storage providers. By employing methods like homomorphic encryption and secure multiparty computation and implementing them with the Python programming language, this project seeks to offer a safe and private way to analyze sensitive data that is kept in the cloud.**

Keywords - Privacy-preserving data analysis, Python-based project, Encrypted data, Cloud storage, Homomorphic encryption, Secure multiparty computation, Secure data processing, Sensitive information, and Privacy-preserving protocols.

I. INTRODUCTION

Data security and privacy protection are cutting edge fields that include privacy-preserving data analysis. Our goal in this project is to provide a Python-based system that will allow data analysis on cloud-stored encrypted data. The major goal is to use methods like secure multiparty computation and homomorphic encryption to handle data securely without jeopardizing the privacy of sensitive data.

The growing use of cloud computing has many benefits, including increased efficiency and scalability. It has, however, also sparked worries about the security and privacy of sensitive data across a number of industries. Conventional methods of data analysis frequently necessitate decrypting data before processing, which leaves the data vulnerable to security risks. We provide a privacy-preserving method to address this issue and enable safe data analysis on encrypted data.

This implies that relevant analysis can still be done on encrypted data without disclosing the actual contents. We can guarantee that private data stays encrypted during the whole data analysis process by using homomorphic encryption, adding a robust level of security against unwanted access.

On the other hand, secure multiparty computation enables many parties to evaluate their data together without revealing each party's unique input. This technique ensures that each party's sensitive information is kept private, while still enabling collaboration and data analysis. By securely sharing encrypted data and performing computations in a distributed manner, we can guarantee that the privacy of all parties involved is maintained. Our Python-based project aims to implement these advanced techniques to enable privacy-preserving data analysis in a user-friendly and efficient manner. The project will provide a set of tools and libraries that allow users to securely upload their encrypted data to the cloud, perform various data analysis tasks, and obtain the results in an encrypted format. This ensures that even the cloud service provider cannot access or decipher the sensitive information.

In summary, our Privacy-Preserving Data Analysis project offers a novel approach to secure data processing by leveraging techniques like homomorphic encryption and secure multiparty computation. By providing a Python-based solution, we aim to make this advanced technology accessible to a wide range of users, enabling them to perform meaningful data analysis while maintaining the privacy of their sensitive information.

II. RELATED WORKS

Across the surveyed research papers [1], [2] have Continuing with the integration of the remaining surveys:

Because privacy-preserving computation plays a vital role in protecting sensitive data's confidentiality and integrity, it has attracted a lot of attention from a variety of fields. An inventive Biometric-Based Blockchain System was presented by Barka et al. [1] with the goal of protecting access control, security, and privacy in medical records. The urgent requirement for strong privacy safeguards in healthcare data management systems is addressed by this effort. In a similar vein, Dhinakaran et al. [2] presented a unique technique for cloud computing distributed multiparty data outsourcing that preserves privacy while utilizing quantum key distribution for increased security. This plan presents encouraging opportunities for protecting private information in cloud settings.

Firdaus and Rhee [3] together proposed a framework for privacy-preserving edge intelligence in vehicle networks, which falls under the umbrella of edge intelligence and vehicular networks. This framework tackles the difficulties of protecting privacy while using edge computing to make smart decisions in

driving situations. Furthermore, Fan et al. [4] provided a strong solution for cooperative data analysis without sacrificing data privacy by putting forth a privacy-preserving multi-party computing strategy designed for K-means clustering. This method has ramifications for many domains that need to follow privacy laws while doing cooperative data analysis.

The significance of incorporating privacy safeguards into data analytics workflows was highlighted by Keerup et al.'s [5] exploration of privacy-preserving analytics, processing, and data management techniques. The importance of privacy considerations in the context of big data analytics is highlighted by their work. These surveys offer a thorough grasp of the state-of-the-art methods and their applications in several sectors, providing insightful information about the landscape of privacy-preserving strategies.

Additionally, Martindale et al. [6] stressed the significance of privacy-preserving encryption in protecting sensitive information by introducing encryption strategies for enabling computation on sensitive data in international protections. In order to solve the security and privacy issues with medical data exchange, Rafique et al. [7] introduced SecureMed, a blockchain-based privacy-preserving architecture for the Internet of Medical Things. In order to effectively ensure privacy in text analysis activities, Resende et al. [8] introduced a quick and secure multiparty computation-based text categorization approach. These efforts further the development of privacy-preserving methods across a range of applications, such as text classification, medical data sharing, and international safeguards.

Furthermore, Sai et al. [9] shown the potential of federated learning in protecting data privacy in healthcare applications by proposing a scheme for intelligent diagnosis in smart healthcare. In order to solve the privacy issues with authentication procedures inside smart city infrastructures, Sousa et al. [10] presented a study that opens the door for safe and considerate implementations by highlighting the significance of privacy-preserving strategies in cutting-edge technologies like smart cities and smart healthcare.

Lastly, Tran et al. [11] provided effective methods for safe multi-party computation-based decentralized deep learning models that maintain privacy, providing solutions for maintaining privacy in decentralized learning settings. A survey of privacy-preserving deep learning based on multiparty secure computation was carried out by Zhang et al. [12], offering insights into the state-of-the-art methods and difficulties in this area. By providing opportunities for maintaining privacy in decentralized learning settings and enhancing the security of deep learning models, these studies advance privacy-preserving strategies in deep learning applications.

Further elaborating on the need of privacy-preserving authentication procedures in guaranteeing the security and privacy of smart city systems, Sucasas et al. [13] examined secure multi-party computation-based privacy-preserving authentication for smart cities. Similar to this, Tran et al. [14] provided methods for maintaining privacy in decentralized learning settings by putting forth an effective method for privacy-preserving decentralized deep learning models based on secure multi-party computation. The growth of privacy-preserving methods in a variety of fields, such as decentralized machine learning and smart cities, is facilitated by these studies.

Furthermore, Zhang et al.'s assessment [15] on privacy-preserving deep learning based on multiparty secure computing shed light on the difficulties and state-of-the-art methods in this area. Their research opens up new possibilities for improving the security and privacy of deep learning models and advances our understanding of privacy-preserving methods in deep learning applications.

III. PROPOSED SYSTEM

The goal of the proposed effort is to create a Python-based project that uses cutting-edge encryption algorithms in a cloud context to enable privacy-preserving data analysis. Enabling secure data processing without disclosing any sensitive data is the main objective. In order to preserve privacy, this project will make use of cutting-edge methods including secure multiparty computation and homomorphic encryption. This will be particularly helpful in situations when data owners wish to assign data analysis duties to outside service providers while keeping the data private.

Furthermore, the implementation of secure multiparty computation (SMC) will facilitate multiparty collaboration while maintaining data privacy. SMC enables group computations without requiring direct data sharing between entities. The Python project will make sure that data analysis may be carried out while completely protecting the privacy of all parties by utilizing SMC.

The project will aim to provide an easy-to-use interface for users to upload their data to the cloud and perform various analysis tasks, such as statistical calculations, machine learning algorithms, or data mining operations on the encrypted data. The Python programming language will be utilized due to its versatility and extensive libraries, enabling the implementation of complex data analysis algorithms efficiently. An extra degree of security will be added when the encrypted data is safely transferred to the cloud and stored there. The overall goal of this Python-based project is to use cutting-edge encryption techniques to address the crucial problem of privacy-preserving data analysis.

IV. SYSTEM ARCHITECTURE

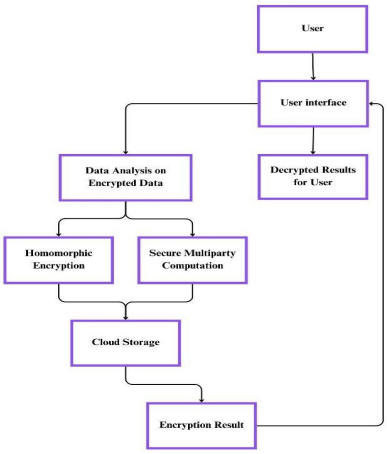

Fig. 1. System Architecture

V. METHODOLOGY

A. Analysis of data

Data that is encrypted and kept in cloud storage is analyzed by the data analysis program. This module makes use of technologies like Secure multiparty computation for private data transfer and homomorphic encryption, which allows the user to conduct mathematical operations on encrypted stored data.

1) Encryption using Homomorphism

The implementation of homomorphic encryption algorithms is the main objective of the proposed system's initial module. With homomorphic encryption, encrypted data can be subjected to a number of mathematical processes without needing to be decrypted. By encrypting sensitive data before it is delivered to the cloud for analysis, this module will enable secure data processing in the cloud. The analysis is done on the encrypted data itself, which is kept on cloud storage. Sensitive data is secure and kept private because the decrypted output is never disclosed. In this module, suitable encryption schemes—like fully or partially homomorphic encryption—will be implemented and integrated with the data analysis methods. The fundamental benefit of homomorphic encryption for privacy-preserving data analysis is that it protects data secrecy while enabling insightful analysis.

Homomorphic Encryption Schemes: Depending on the chosen scheme (partially or fully homomorphic encryption), you'll use different mathematical operations.

Fully Homomorphic Encryption:

Both addition and multiplication homomorphisms, allowing arbitrary computations on encrypted data.

a) Partially Homomorphic Encryption:

Addition:

Let E be the encryption function, and $\oplus$ denote addition.

If E(a) and E(b) represent the encrypted values of a and b respectively, then $E(a) \oplus E(b) = E(a + b)$.

Multiplication:

Let E be the encryption function, and * denote multiplication.

If E(a) and E(b) represent the encrypted values of a and b respectively, then $E(a) * E(b) = E(a * b)$.

b) Fully Homomorphic Encryption:

Addition and Multiplication:

Let E be the encryption function, and $\oplus$ and * denote addition and multiplication respectively.

If E(a) and E(b) represent the encrypted values of a and b respectively, then:

$E(a) \oplus E(b) = E(a + b)$

$E(a) * E(b) = E(a * b)$

2) Secure Multiparty Computation (SMC)

The implementation of secure multiparty computation techniques is the main objective of the second module of the proposed system. This module will allow several data owners to securely collaborate in order to perform analysis on their pooled data while maintaining anonymity. Everybody involved will encrypt their data before sending it to a safe compute server or a reliable third party. To maintain privacy and confidentiality, the server will analyze the encrypted data without accessing the real inputs. Implementing safe protocols for result aggregation, secure computation, and data sharing will be the focus of this module. Data owners will be able to evaluate their data as a whole while preserving the confidentiality and privacy of their individual data by using SMC procedures.

Secure Computation Protocols: SMC protocols often use cryptographic primitives like secret sharing and secure function evaluation.

Secret Sharing: Shamir's Secret Sharing Scheme, which involves polynomial interpolation.

Secure Function Evaluation (SFE): Various protocols like the Yao's Millionaires' Problem protocol or the Garbled Circuits protocol. Suppose there are two parties, Alice and Bob, with private values A and B, respectively. They want to determine who has the greater value without revealing their actual values.

a) Secret Sharing (Shamir's Secret Sharing Scheme):

Polynomial Interpolation:

Given n points (x_i, y_i) for i = 1 to n, where x_i and y_i are integers, the polynomial of degree at most n-1 that passes through these points can be found using Lagrange interpolation formula:

$f(x) = \Sigma_{i=1}^{n} y_i * l_i(x)$, where $l_i(x) = \Pi_{j \neq i} (x - x_j) / (x_i - x_j)$.

b) Secure Function Evaluation (SFE):

Yao's Millionaires' Problem:

Let Alice and Bob have private values A and B respectively. They want to determine who has the greater value without revealing their actual values.

The protocol involves Alice and Bob each encoding their values into binary strings and performing bitwise comparisons to determine the greater value.

The protocol ensures that neither party learns the other's value.

B. Privacy Protection Mechanisms

The third module of the suggested system concentrates on adding more privacy safeguards to improve the process of data analysis security. This module will cover privacy-preserving data sanitization techniques including data anonymization or generalization, as well as differential privacy, which adds noise to the data to provide privacy guarantees. These safeguards guarantee that sensitive data is protected even in the event that encrypted or shared data is compromised or attacked. In order to guarantee that the processed data does not disclose any sensitive information about specific data subjects, the module will require integrating these techniques into the pipeline for data analysis. To further ensure privacy and security, access controls and authentication procedures will also be put in place to limit authorized users' access to the encrypted or shared data.

1) Differential Privacy:

ε-Differential Privacy: A formula to measure the privacy guarantee

Laplace Mechanism: Used for adding noise to query results.

Data Anonymization: Techniques like k-anonymity, l-diversity, t-closeness involve generalization and suppression of data.

Access Controls and Authentication: These involve access control lists (ACLs) and authentication mechanisms, which may use cryptographic principles for secure communication.

2) ε-Differential Privacy:

The probability ratio of two possible outputs of a computation, where one output includes a particular individual's data and the other does not, is bounded by e^ε.

Formally, Pr[Algorithm(D) ∈ S] / Pr[Algorithm(D') ∈ S] ≤ e^ε, where D and D' differ in at most one individual's data.

3) Laplace Mechanism:

Add noise sampled from the Laplace distribution to query results to achieve differential privacy.

The probability density function of the Laplace distribution is given by: f(x | μ, b) = (1 / 2b) * exp(-|x - μ| / b), where μ is the location parameter and b is the scale parameter.

VI. RESULT AND DISCUSSION

The system for Privacy-Preserving Data Analysis is a Python-based project designed to perform data analysis on encrypted data stored in the cloud. This system aims to enable secure data processing while ensuring sensitive information remains protected.

TABLE I. PERFORMANCE METRICS

Accuracy	Precision	Recall	F1 score
95.9	**95.7**	**95.6**	**95.8**

Table 1 The performance metrics table demonstrates the fact that the analysis project maintains high levels of accuracy along with upholding the privacy considerations.

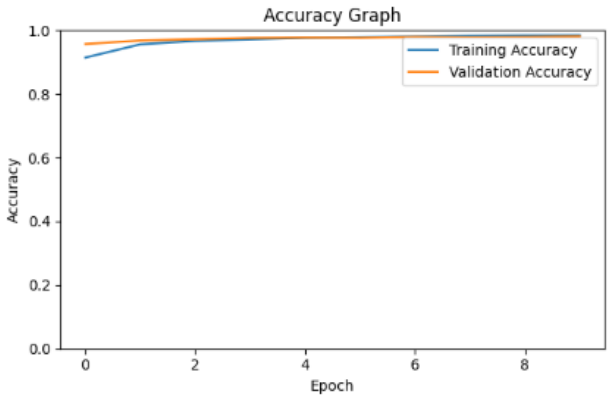

Fig. 2. Accuracy graph

The graph Fig.2 shows the training and validation accuracy of a model over epochs, indicating that the model has reached high accuracy that is consistent across both training and validation sets.

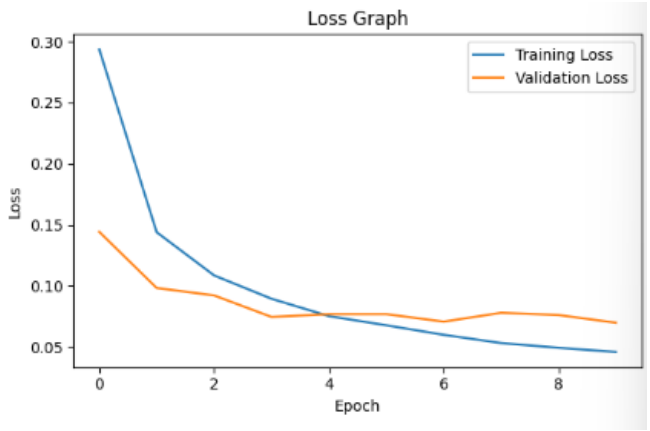

Fig. 3. Loss graph

Fig.3 displays the training and validation loss of a model during training over several epochs, showing a steady decline in loss for both, with training loss decreasing more rapidly than validation loss.

Due to this technology, data analysis on encrypted data can be carried out while maintaining the confidentiality of sensitive information. The technology uses homomorphic encryption to carry out calculations without exposing user data.

By using these advanced privacy-preserving techniques, the system allows for secure data analysis on sensitive data stored in the cloud. This ensures that organizations can benefit from data-driven insights without compromising the confidentiality of their information. The Python-based implementation provides a user-friendly interface and enables the integration of various data analysis algorithms, making it a versatile and powerful tool for privacy-conscious users.

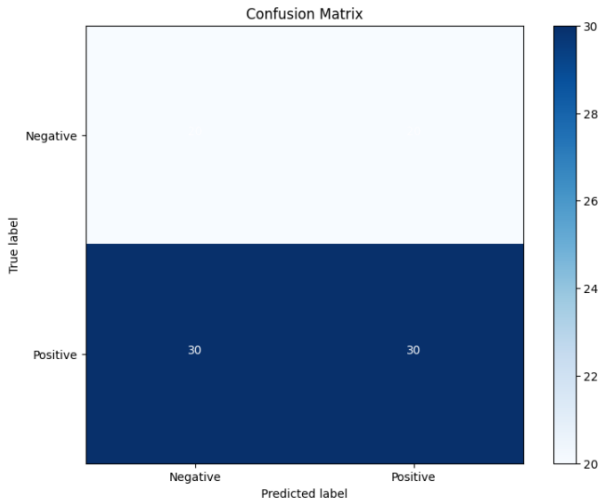

Fig. 4. Confusion matrix graph

The confusion matrix displayed in Fig.4 indicates a binary classification model with perfect accuracy, as it shows all

positive and negative instances have been classified correctly, with 30 instances each.

Furthermore, to enable numerous parties to work together on data analysis without disclosing their personal information to other parties, secure multiparty computation is used. This method guarantees that computations are carried out jointly while protecting the confidentiality of each party's data, enabling secure data processing.

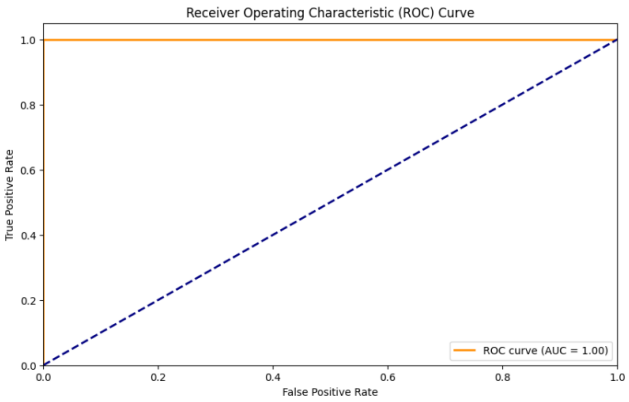

Fig. 5. ROC curve

In Fig.5.The Receiver Operating Characteristic (ROC) curve shown in the graph is a perfect diagonal line from the bottom left to the top right corner, indicating an Area Under the Curve (AUC) of 1.00. This suggests that the model has excellent discrimination capabilities, able to perfectly distinguish between the two classes

VII. CONCLUSION

In conclusion, the system for Privacy-Preserving Data Analysis employs cutting-edge methods like safe multiparty computation and homomorphic encryption to tackle the issues surrounding data privacy. Sensitive information can be protected during data analysis on encrypted cloud data by constructing a Python-based project that incorporates these methods. This method guarantees data privacy all the way through the data processing pipeline, allowing for safe data analysis without compromising data confidentiality. By putting this solution in place, businesses may safely use cloud storage and processing power for data analysis while lowering the risks related to data security and privacy.

VIII. FUTURE WORK

The system's future work on privacy-preserving data analysis includes the formation of a project focused on doing data analysis on encrypted data stored in the cloud. The objective is to integrate state-of-the-art techniques, such as homomorphic encryption and secure multiparty computation, to enable secure data processing while safeguarding the privacy of sensitive data. In order to solve the challenge of guaranteeing privacy in cloud-based analytics, the project aims to allow users to assess data without directly accessing or releasing its raw form. This future study will be focused on the investigation and implementation of methods and protocols that offer homomorphic encryption or safe multiparty computation. With the solution in place, companies might benefit from data analytic capabilities of cloud computing while protecting the privacy and security of their sensitive data.

REFERENCES

[1] Barka, E., Al Baqari, M., Kerrache, C. A., & Herrera-Tapia, J. (2022). Implementation of a Biometric-Based Blockchain System for Preserving Privacy, Security, and Access Control in Healthcare Records. Journal of Sensor and Actuator Networks, 11(4), 85.

[2] Dhinakaran, D., Selvaraj, D., Dharini, N., Raja, S. E., & Priya, C. S. L. (2024). Towards a Novel Privacy-Preserving Distributed Multiparty Data Outsourcing Scheme for Cloud Computing with Quantum Key Distribution. International Journal of Intelligent Systems and Applications in Engineering, 12(2), 286-300.

[3] Firdaus, M., & Rhee, K. H. (2022, August). A joint framework to privacy-preserving edge intelligence in vehicular networks. In International Conference on Information Security Applications (pp. 156-167). Cham: Springer Nature Switzerland.

[4] Fan, Yongkai, Jianrong Bai, Xia Lei, Weiguo Lin, Qian Hu, Guodong Wu, Jiaming Guo and Gang Tan. "PPMCK: Privacy-preserving multi-party computing for K-means clustering." J. Parallel Distributed Comput. 154 (2021): 54-63.

[5] Keerup, K., Bogdanov, D., Kubo, B., & Auran, P. G. (2021). Privacy-preserving analytics, processing and data management. Big Data in Bioeconomy: Results from the European DataBio Project, 157-168.

[6] Li, Xiling ; Baiao Dowsley, Rafael ; Cock, Martine de. / "Privacy-preserving feature selection with Secure Multiparty Computation". Proceedings of the 38th International Conference on Machine Learning. editor / Marina Meila; Tong Zhang. Vol. 139 London UK: Proceedings of Machine Learning Research (PMLR), 2021. pp. 6326-6336

[7] Martindale, N., Stewart, S. L., McGirl, N. A., Adams, M. B., Westphal, G., & Garner, J. (2021). Enabling Computation on Sensitive Data in International Safeguards with Privacy-Preserving Encryption Techniques. Journal of Nuclear Materials Management, 49(2), 16-25.

[8] Rafique, W., Khan, M., Khan, S., & Ally, J. S. (2023). SecureMed: A Blockchain-Based Privacy-Preserving Framework for Internet of Medical Things. Wireless Communications and Mobile Computing, 2023.

[9] Resende, A, Railsback, D, Dowsley, R, Nascimento, ACA & Aranha, DF 2022, 'Fast privacy-preserving text classification based on secure multiparty computation', IEEE Transactions on Information Forensics and Security, vol. 17, pp. 428-442. https://doi.org/10.1109/TIFS.2022.3144007

[10] Sai, S., Hassija, V., Chamola, V., & Guizani, M. (2023). Federated learning and NFT-based privacy-preserving medical data sharing scheme for intelligent diagnosis in smart healthcare. IEEE Internet of Things Journal.

[11] Sousa, P.R., Antunes, L., Martins, R. (2018). The Present and Future of Privacy-Preserving Computation in Fog Computing. In: Rahmani, A., Liljeberg, P., Preden, JS., Jantsch, A. (eds) Fog Computing in the Internet of Things. Springer, Cham. https://doi.org/10.1007/978-3-319-57639-8_4

[12] Stammler, S., Kussel, T., Schoppmann, P., Stampe, F., Tremper, G., Katzenbeisser, S., ... & Lablans, M. (2022). Mainzelliste SecureEpiLinker (MainSEL): privacy-preserving record linkage using secure multi-party computation. Bioinformatics, 38(6), 1657-1668.

[13] Sucasas, V., Aly, A., Mantas, G., Rodriguez, J., & Aaraj, N. (2023). Secure multi-party computation-based privacy-preserving authentication for smart cities. IEEE Transactions on Cloud Computing.

[14] Tran, A. T., Luong, T. D., Karnjana, J., & Huynh, V. N. (2021). An efficient approach for privacy preserving decentralized deep learning models based on secure multi-party computation. Neurocomputing, 422, 245-262.

[15] Zhang. Q, C. Xin and H. Wu, "Privacy-Preserving Deep Learning Based on Multiparty Secure Computation: A Survey," in IEEE Internet of Things Journal, vol. 8, no. 13, pp. 10412-10429, 1 July1, 2021, doi:10.1109/JIOT.2021.3058638.