Homomorphic Encryption based Federated Learning for Financial Data Security

Shalini Dhiman
Department of Computer Science & Engineering
National Institute of Technology
Arunachal Pradesh, India
shalini.phd22@nitap.ac.in
0000-0003-1384-7238

Sumitra Nayak
Department of Computer Science & Engineering
National Institute of Technology
Arunachal Pradesh, India
sumitra.mtech.cse.21@nitap.ac.in
0000-0002-1910-6337

Ganesh Kumar Mahato
Department of Computer Science & Engineering
National Institute of Technology
Arunachal Pradesh, India
ganesh.phd20@nitap.ac.in
0000-0002-2925-7438

Anil Ram
Department of Computer Science & Engineering
National Institute of Technology
Arunachal Pradesh, India
anil.phd20@nitap.ac.in
0000-0002-3581-0519

Swarnendu Kumar Chakraborty
Department of Computer Science & Engineering
National Institute of Technology
Arunachal Pradesh, India
swarnendu@nitap.ac.in
0000-0002-6223-9362

2023 4th International Conference on Computing and Communication Systems (I3CS) | 979-8-3503-2377-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/I3CS58314.2023.10127502

***Abstract*—Federated Learning is a distributed machine learning technique that enables on-device training without exchanging the sensitive data over the centralized server. In this paper, Federated Learning is used to train financial data models on-devices with the help of IoT applications in financial or business systems. This creates more advanced and secured financial business models. We have applied the mechanism of homomorphic encryption and cryptographic primitives, including masking and local model protection to prevent any kind of inferring in private financial or business data where multiple attackers usually find a way of model inversion or reconstruction attack. We used datasets of various financial sectors as the primary measure, rather than taking the size of datasets that is generally used in deep learning, to get the correct measurement of the rate of contribution in every session of the local model's training to the training of the global model. If the calculated online number of clients exceeds a predetermined threshold, then the federated learning process will be continued with the dropout-tolerant plan. The security study demonstrates that the suggested solution fulfils data privacy requirements. The costs of computation and communication are also examined theoretically. According to research observations, the proposed approach achieved promising outcomes while assuring privacy preservation when compared to existing schemes.**

***Index Terms*—Homomorphic Encryption, Federated Learning, Privacy-preserving**

I. Introduction

The Internet of Things (IoT) is a beneficial concept that has many different applications. Through IoT-connected financial businesses, investors or market analysts can access financial or business data from their users in real-time from any location, enabling smart transactions and continuous monitoring. In addition, IoT offers computer-assisted financial management, financial analysis, and financial crisis prediction services. This allows users to achieve better results and financial operators to work more effectively. There are several benefits to using IoT-based business modules and financial analysis techniques that can enhance efficiency [1]. Sharing electronic financial data can assist market analysts in predicting fluctuations in investment plans, particularly in the financial sector. Sharing financial reports can help the entire industry develop. However, sharing resources is now difficult due to the sensitive nature of investment reports. Most market analysts are located in different regions and work under different administrative managements [2]. The concern of violating privacy laws or the risk of losing financial benefits, such as exposing business or market strategies or sensitive financial investments, makes market analysts hesitant to disclose investors' or industrially sensitive financial data. These privacy concerns are equally significant challenges that need to be addressed in other industries as well.

Researchers in [3] and [4] have developed a framework for federated learning, which enables data from local repositories to contribute to machine learning while preserving privacy. The aim is to achieve the benefits of data sharing without compromising user privacy. Federated learning has the potential to improve multiple fields through the application of artificial

intelligence in various public sectors. However, it can also pose a threat to user privacy. Implementing federated learning in real-world situations, such as IoT-based financial businesses, presents a significant challenge. Although local data is not shared directly, the local models are aggregated into a hub, making them vulnerable to attacks.

As a result of these concerns, there has been an increasing focus on implementing federated learning processes that protect user privacy. However, the application of federated learning can present various technical challenges, such as ensuring an accurate global model, managing client dropouts, and minimizing computation and communication costs.

To address these issues, the research proposes a privacy preservation system for federated learning in several financial sectors of IoT industrial applications, with a focus on financial data security. The federated learning architecture is implemented using multi-party computing, which is already a secure format, as well as homomorphic encryption to maintain data privacy. This system is also capable of handling collusion between multiple participants.

In addition, we have addressed the cost issue by enhancing the existing scheme. Furthermore, we have prioritized the accuracy of the model and simultaneously improved the efficiency of the security process by considering the diverse characteristics of data in different market sectors.

II. Background

A. Federated Learning

Federated Learning is a type of a distributed collaborative AI solution that helps to train a dataset and the training of data is done by the coordination of numerous devices with a central server, the actual datasets are not much used, For instance, in intelligent IoT networks, multiple IoT devices functions as workers to connect with a server, the server can be also known by the aggregator [3]. The server at first starts with a global model with some of the learning parameters, to be more precise. The current model is downloaded by each worker from the aggregator, who then updates the model using a local dataset and offloads the locally computed update to the aggregator. Finally, the server by the use of all the local model it creates a new global model. the server can improve the quality of training by the use of processing capacity of distributed workers that can reduce the privacy leakage of distributed workers [4]. Lastly the local worker from the aggregator downloads all the global updates and gets ready for their next local update till the completion of training.

There are three main types of federated learning, discussed below.

- Horizontal Federated Learning (HFL): HFL is a suitable approach when there is limited user overlap but significant user feature overlap between two datasets. In HFL, the datasets are horizontally divided by the user dimension, and the data that have the same user attributes but different users are removed for training. In other words, identical data features appear in various rows of data aligned by user features. As a result, Federated Horizontal Learning can broaden the user sample.
- Vertical federated learning (VFL): VFL is a scenario where there is little overlap between the user features of the two given datasets, but there is significant overlap in users. In VFL, datasets are divided vertically (by user feature dimension), and data with identical users but different user attributes is removed for training. In other words, identical users appear in the data across columns (aligned by user). The feature dimension of training data can therefore be increased using VFL.
- Federated Transfer Learning (FTL): FTL is a learning scheme that involves transferring knowledge from a dataset with a rich feature space to a dataset with limited features or labels in order to train a high-performing model. This is particularly useful when dealing with multiple datasets from different parties with different levels of features.

B. Homomorphic Encryption

Homomorphic encryption is a special type of mathematical operations, performed on data that is encrypted without compromising the encryption. The result gained after performing computation directly on plaintext is mostly equivalent to the result of computation that is performed on the cipher text [7] [20]. Here the service provider performing the computation gains no knowledge of the plain text, since they perform the computation at the encrypted state. Thereby, security of the plaintext is preserved.

III. Related Work

The authors in [4] have encountered a cybersecurity challenge where data aggregation becomes difficult. To address this challenge, the authors have used the concept of the smart grid, which provides solutions for enhancing operating systems and improving the distribution, generation, and transmission of electricity. However, the smart grid also faces security and privacy concerns and incurs high computational and communication costs. To overcome these concerns, the authors have proposed a privacy-preserving homomorphic encryption-based deep learning and data aggregation model. This model reduces the negative impact on the accuracy of flashed workload prediction models by performing smart metering using different authentication techniques. For enhanced security, the authors have designed a cloud computing and blockchain-based data aggregation framework. According to the results, this model is capable of achieving 80

According to [5], the traditional method of executing federated learning tasks results in reduced training speed, accuracy, and data security. When using Vertical Federated Learning for classifying tasks, the traditional method produces larger invalid parameters. To overcome this problem, the authors propose using gradient compression and parameter sharing for communication on both sides, as well as improving homomorphic encryption on parameter transfer. As a result, the authors achieve better classification model evaluation index compared

to the traditional federated learning logic regression algorithm. The authors also achieve faster training times and better data security.

In [6], [7], [17], the authors noted that trust issues arise in federated machine learning when multiple parties are involved in joint data analytics with distributed datasets. To address this problem, they proposed a scenario called Gradient Boosting Machines (GBM), which is a multi-party computation model based on semi-homomorphic encryption. In this scenario, each party can obtain their own model of shared Gradient Boosting machines, which enables them to protect their data from exploitation. The authors mostly worked in a "dual-party" scenario where both parties have a unique view and feature and label sharing is not allowed. To achieve this, they [18], [19] introduced LightGBM. For secure communication and computation, they introduced SecureGBM. As a result, the authors achieved a performance slowdown of between 3x to 64x, but the efficiency improved due to SecureGBM resulting in a lower slowdown ratio and a larger training set. Table I compares our proposed model with similar works that have been done previously.

IV. System Model

A. Model Design

This section discusses the FL system in the financial environment, which is composed of a distributed client and a model aggregation server as shown in Fig. 1. The system employs secure multi-party computing and homomorphic encryption to ensure data privacy even in the presence of collusions among sincere but curious participants. The research proposes an improved system that addresses cost problems that affected previous cryptography-based schemes while increasing model accuracy and performance outcomes for different financial industries' data properties.

To reduce communication overhead, the Advanced Paillier encryption technique's additive homomorphism is changed to multiplicative homomorphism by adjusting the algorithm. Additionally, homomorphic encryption in federated learning does not encrypt each model parameter, unlike traditional encryption. Homomorphic encryption with high-dimensional data, as seen in deep learning models, would result in significant computational costs. The proposed scheme in this paper only encrypts the data quality variable for each client in each achievement plan. This approach prevents a dramatic increase in computation overhead.

The proposed model works upon the mutual cooperation of the server and the client. Client in this system is basically a financial institutions, it consist of the large financial raw data. The client performs duty of training the local models on the local financial dataset and further they submit it to the server. The client submits all the masked financial local models, once the masked model is submitted then the server collects all those masked financial local model and performs the series of operation to execute the secure aggregation. While performing this process, it is required to perform some privacy preserving steps they can be seen in the given algorithms, Algorithm 1 discusses the training of the model with security whereas Algorithm 2 proposes the multi layer perceptron Federated Learning with data security.

Algorithm 1 Model Training algorithm with security

1: Input: $\left[Wt_l^1\right]_1$, $\left[Wt_l^2\right]_2$, …, $\left[Wt_l^{N_c}\right]_{N_c}$
2: CSP generates N_i random vectors $R_1, R_2, \ldots, R_{N_j}$ and encrypts them by the public keys.
3: for $j \leq N_i$ do
4: CSP $[S_k]_k \leftarrow \left[W_l^j\right]_j \cdot [R_j]_j$
5: CSP Partially decrypts $[S_i]_i$ using $SK_j^{(1)}$
6: end for
7: CSP Sends the vectors in decrypted form to CP

Algorithm 2 Multi Layer perceptron-federtaed learning with data security

Input: Financial Dataset D_x
Output: Model ϕ_{final}
1: Initialize the Models ϕ
2: for i iterations do
3: Forward propagate: out $_j = hq\left(x_j, \phi_j\right)$;
4: Computation of loss: $c_j = \text{loss}\left(h^*\left(x_j\right), \text{out } t_j\right)$;
5: if $c_j < \epsilon$ then
6: Break
7: else
8: Back propagate: grad $_j = bp\left(x_j, \theta_j, c_j\right)$;
9: Send gradient to the server performing computation;
10: Updation: $\phi_{j+1} = \phi_j - lr * g_{\text{rad}}$ new;
11: end if
12: end for
13: return Model with parameters ϕ_{final}

B. Experimental Environment

The proposed algorithm is developed using Python and evaluated on a machine having 3.6 GHz quad-core processor and 16 GB RAM. Federalted Learning Simulation Platform-FLUTE is used to carryout the experiments. Nearly 76000 datasets are trained and tested in this model.

C. Data Sets

In this context, data from financial sectors are used to train the model and perform homomorphic computation on the encrypted state. Over 76000 financial data on different parameters are used. Data are collected from [22] which is a mix of data from stock market, banking transactions, loan sectors etc.

D. Security Analysis

The computation of the data is being done in the encrypted form, where the cloud service provider and the computation server has no idea of the plain text. The cloud service provider sends the data to the computing server only after encrypting the sensitive data, there is no chance of getting the data in the hand of intruders.

TABLE I
COMPARISONS OF SIMILAR SCHEMES

Ref.	Scheme used	Data aggregation	Server	Encryption	Data access	Architecture	Homomorphic encryption method
[4]	Blockchain	Yes	Cloud	Yes	Partially	Centralized	Based on Data Aggregation
[5]	Blockchain	Yes	Nil	Yes	Fully	Centralized	Threshold Secret Sharing
[6]	Nil	Yes	Central Server	Yes	Nil	Centralized	Data Aggregation
[7]	Machine Learning	Yes	FL Server	Yes	Nil	Decentralized	Trusted Execution Environments
[8]	Machine Learning	Yes	FL Server	Yes	Nil	Decentralized	Decentralized Threshold Additive Homomorphic Encryption
[9]	Cloud Computing	No	Nil	Yes	Yes	Centralized	FL
[10]	Machine learning	No	Nil	Yes	No	None	Homomorphic Encryption for Arithmetic of Approximate Numbers
[11]	Machine Learning	Yes	Central server	Yes	Yes	Centralized	Additive Homomorphic Encryption
[12]	Nil	No	Cloud Server	Yes	No	Decentralized	Secret Sharing on Nodes at the Edge Layer
[13]	Machine Learning	Yes	Central Server	Yes	Yes	Decentralized	Additive Homomorphic Encryption
[14]	Artificial Intelligence	No	Host/Guest server	Yes	No	None	Data Aggregation
[15]	Machine Learning	Yes	Cloud Server	Yes	Yes	Centralized	Fashion-MNIST
[16]	Machine Learning	No	Nil	Yes	No	Centralized	Gradient Boosting Machines
Ours	Federated Learning	Yes	Cloud	Yes	No	Centralized	Additive and multiplicative both

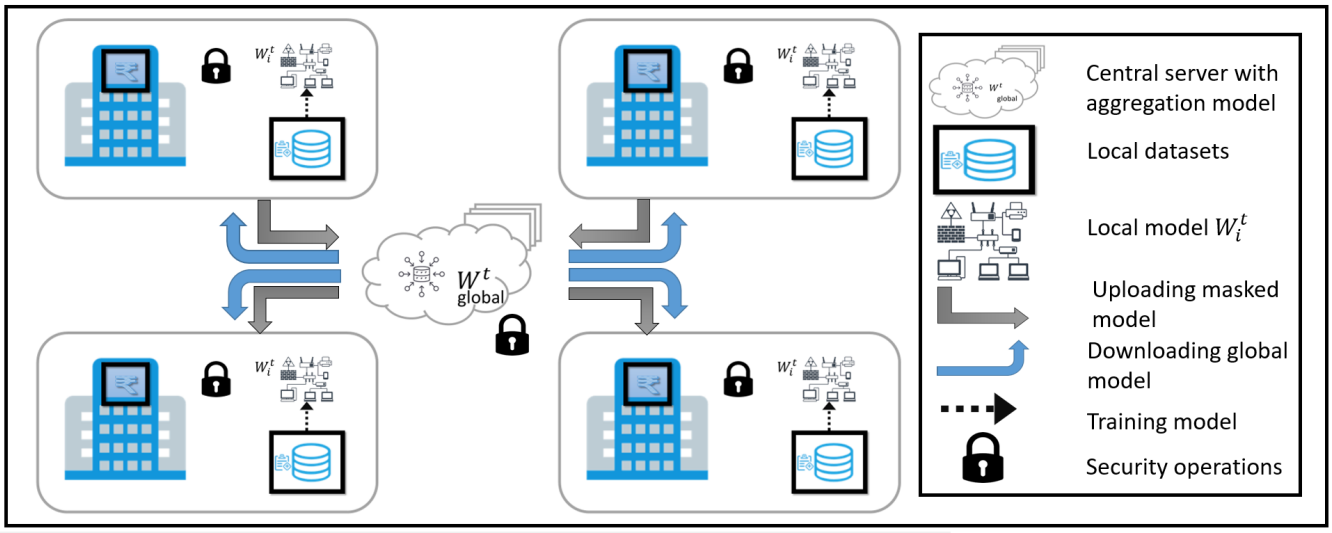

Fig. 1. Homomorphic Encryption based Federated Learning for secured Financial Data.

E. Performance Evaluation

The performance of the proposed model is tabulated in the TABLE II and TABLE III. Former depicts the time consumption in various iterations of different key size whereas the later shows the time taken in training the model at different layers.

Keys of different bit sizes, 128 to 2048 are used to secure the encrypted data. With the increase in key size, the time consumption also increases. Also, as the hidden layers increases the training time also increases. The graph in the Fig. 2. shows the time taken by the proposed model for key generation, encryption, computation and decryption with respect to the different key sizes. Fig. 3. shows the time spent on training the model at different iterations using the encrypted data of various key length.

CONCLUSION

In our proposed paper, we have presented a unique federated learning framework designed for an IoT-based financial marketing system that prioritizes preserving privacy. We con-

TABLE II
TIME CONSUMPTION (SEC) IN VARIOUS ITERATIONS OF DIFFERENT KEY SIZE

Algorithm	Key Size (Bits)	time (sec)
Advanced Paillier	128	547.65
	256	642.7
	512	1112.25
	1024	2025.27
	2048	3768.56

TABLE III
TIME CONSUMPTION (SEC) IN VARIOUS HIDDEN LAYERS FOR TRAINING

Datasets used	Hidden layers (unit)	time (sec)
Financial Data	3.25	2432.54
	3.65	3642.7
	4.23	4112.25
	4.54	5025.27
	5.25	8768.56

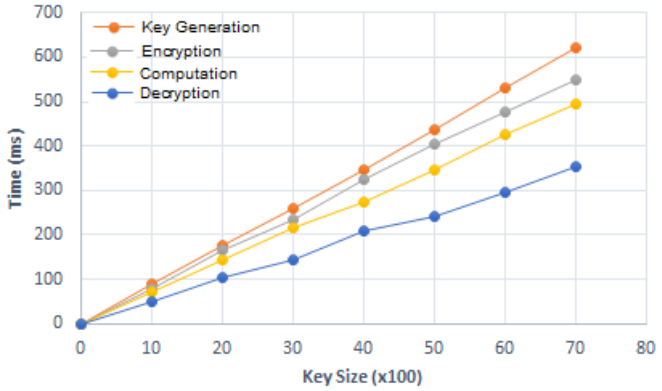

Fig. 2. Time spent on homomorphic key generation, encryption, computation and decryption

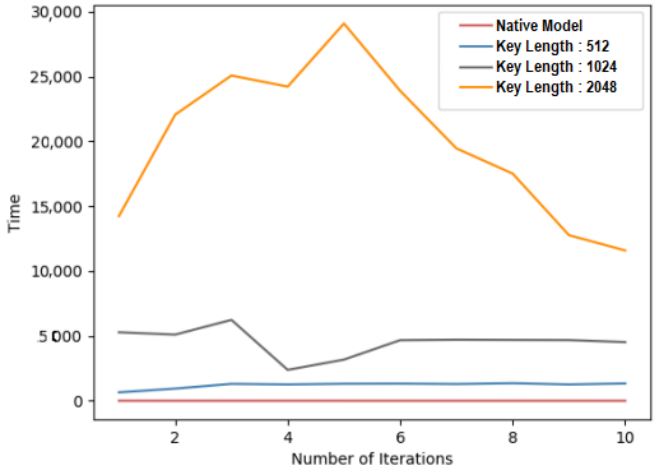

Fig. 3. Time spent on Financial data set for different key size

cluded that this framework could potentially revolutionize the current system by enabling more efficient data utilization while maintaining data privacy. To achieve this goal, we modified the calculations used in traditional system techniques [21] which were previously based on data volume. Instead, we used a weighted average algorithm based on data quality which we found to be more effective. In the context of federated learning, we also introduced a brand-new masking approach that utilized both secure multi-party computation and homomorphic encryption. This approach ensures that data from multiple devices is securely and efficiently utilized without compromising user privacy. However, we realized that homomorphic encryption on such high-dimensional data typically associated with deep learning models would result in significant computational overheads. As a result, we designed our homomorphic encryption model to not encrypt each model parameter individually to reduce the computational cost while still preserving privacy.

Our proposed technique aims to minimize the computational overhead associated with homomorphic encryption by encrypting only the data quality variable for each client in a fixed form. This approach prevents a significant increase in computation overhead, making it more feasible to implement in practice. To achieve this, we modified the Paillier encryption technique to transform its property of homomorphic multiplication to additive homomorphism, which is suitable for the system proposed in our study.

Overall, we believe that our proposed technique offers a promising solution to the problem of computational overhead associated with homomorphic encryption while maintaining a reasonable cost. By encrypting only the necessary variables and utilizing efficient encryption techniques, we are able to provide a practical and secure solution for federated learning in IoT-based financial marketing systems.

REFERENCES

[1] Zhang, Li, J. Xu, P. Vijayakumar, P. K. Sharma, and U. Ghosh, "Homomorphic encryption-based privacy-preserving federated learning in iot-enabled healthcare system," IEEE Transl, Network Science and Engineering, 2022.

[2] Kim, Andrey, A. Papadimitriou, and Y. Polyakov, "Approximate homomorphic encryption with reduced approximation error," Springer, Cham, RSA Conference, Cryptographers' Track, pp. 120-144, 2022.

[3] Kara, Mostefa, A. Laouid, M. A. Yagoub, R. Euler, S. Medileh, M. Hammoudeh, A. Eleyan, and A. Bounceur, "A fully homomorphic encryption based on magic number fragmentation and El-Gamal encryption: Smart healthcare use case," Expert Systems 39, no. 5, p.e12767, 2022.
[4] P. Singh, M. Masud, M.S. Hossain, and A. Kaur, "Blockchain and homomorphic encryption-based privacy-preserving data aggregation model in smart grid," Computers and Electrical Engineering, 93, p.107209, 2021.
[5] R. Kumar, J. Kumar, A.A. Khan, H. Ali, C.M Bernard, R.U. Khan, and S. Zeng, "Blockchain and homomorphic encryption based privacy-preserving model aggregation for medical images," Computerized Medical Imaging and Graphics, 102, p.102139, 2022.
[6] Jiang, Zhifeng, W. Wang, and Y. Liu, "Flashe: Additively symmetric homomorphic encryption for cross-silo federated learning," arXiv preprint arXiv, 2109.00675, 2021.
[7] Mondal, Arup, Y. More, R. H. Rooparaghunath, and D. Gupta. "Flatee: Federated Learning Across Trusted Execution Environments," arXiv preprint arXiv, 2111.06867 (2021).
[8] Tian, Haibo, F. Zhang, Y. Shao, and B. Li, "Secure linear aggregation using decentralized threshold additive homomorphic encryption for federated learning," arXiv preprint arXiv, 2111.10753, 2021.
[9] Li, Junyi, and H. Huang, "Faster secure data mining via distributed homomorphic encryption," [26th ACM SIGKDD International Conf. Knowledge Discovery & Data Mining, pp. 2706-2714, 2020].
[10] Li, Denghao, J. Wang, L. Kong, S. Si, Z. Huang, C. Huang, and J. Xiao, "A Nearest Neighbor Under-sampling Strategy for Vertical Federated Learning in Financial Domain," In Proceedings of the 2022 ACM Workshop on Information Hiding and Multimedia Security, pp. 123-128, 2022.
[11] Madi, Abbass, O. Stan, A. Mayoue, A. G. Sébert, C. G. Pailler, and R. Sirdey, "A secure federated learning framework using homomorphic encryption and verifiable computing," IEEE. Reconciling Data Analytics, Automation, Privacy, and Security: A Big Data Challenge, 2021 (RDAAPS), pp. 1-8, 2021.
[12] Salim, M. Mohammed, I. Kim, U. Doniyor, C. Lee, and J. H. Park, "Homomorphic Encryption Based Privacy-Preservation for IoMT," Applied Sciences 11, no. 18 (2021), 8757, 2021.
[13] Zhang, Chengliang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, "BatchCrypt: Efficient homomorphic encryption for Cross-Silo federated learning," USENIX Annual Technical Conf. (USENIX ATC 20), pp. 493-506, 2020.
[14] Yang, Kuihe, Z. Song, Y. Zhang, Y. Zhou, X. Sun, and J. Wang, "Model Optimization Method Based on Vertical Federated Learning," IEEE International Symposium on Circuits and Systems (ISCAS), 2021, pp. 1-5. IEEE, 2021.
[15] Zhou, Chunyi, A. Fu, S. Yu, W. Yang, H. Wang, and Y. Zhang, "Privacy-preserving federated learning in fog computing," IEEE Internet of Things Journal 7, no. 11 (2020): 10782-10793.
[16] Feng, Zhi, H. Xiong, C. Song, S. Yang, B. Zhao, L. Wang, Z. Chen, S. Yang, L. Liu, and J. Huan, "Securegbm: Secure multi-party gradient boosting," 2019 IEEE International Conf. Big Data, pp. 1312-1321, 2019.
[17] Mammen, P. Mary, "Federated learning: opportunities and challenges," arXiv preprint arXiv, 2101.05428 2021.
[18] Li, Tian, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," IEEE Signal Processing Magazine 37, no. 3, pp.- 50-60, 2020.
[19] Zhang, Chen, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," Knowledge-Based Systems 216, p.106775, 2021.
[20] Mahato, G. K., Chakraborty, S. K.: A compartive review on homomorphic encryption for cloud security. IETE Journal of Research. Taylor and Francis, pp. 1-10, 2021.
[21] Mahato, G. K., Chakraborty, S. K.: Privacy Protection of Edge Computing Using Homomorphic Encryption. In Pattern Recognition and Data Analysis with Applications (pp. 395-407). Springer, Singapore. (2022)
[22] https://www.kaggle.com/datasets/theworldbank/global-financial-development.