IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 June 2024, accepted 21 August 2024, date of publication 28 August 2024, date of current version 6 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3450836

RESEARCH ARTICLE

Genetic Algorithm and the Kruskal–Wallis H-Test-Based Trainer Selection Federated Learning for IoT Security

A. BHAVANI, (Student Member, IEEE),
AND VIJAYAKUMAR PONNUSAMY, (Senior Member, IEEE)
Department of Electronics and Communication Engineering, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu 603203, India
Corresponding author: Vijayakumar Ponnusamy (vijayakp@srmist.edu.in)

ABSTRACT Federated learning of a decentralized machine learning approach is used for attack detection which trains models collaboratively across multiple IoT devices. The dynamic selection of training nodes is essential due to the heterogeneity of IoT devices. This research presents a new framework for trainer selection in federated learning for IoT security using genetic algorithms and the Kruskal-Wallis H-test. A genetic algorithm is used for the optimal selection of trainers based on computational capabilities, bandwidth, and security. The Kruskal-Wallis H-test, a non-parametric statistical test is used as the objective function to ensure the selected trainers have statistically significant diversity. This combined approach outperforms random and fixed trainer selection methods and improves model accuracy, robustness, and security.

INDEX TERMS IoT security, distributed machine learning, federated machine learning.

I. INTRODUCTION

The number of connected devices has increased dramatically due to the Internet of Things (IoT) quick adoption, which offers both enormous potential for data-driven applications and serious security issues. Because IoT devices range greatly in terms of memory, network bandwidth, processing power, and energy resources, different security risks can easily target them. Because it enables dispersed devices to jointly train a machine learning model without centralizing data, federated learning (FL) has emerged as a promising solution to these problems. But to guarantee strong security and best performance, the heterogeneous character of Internet of Things devices demands a dynamic approach to trainer selection in federated learning [12].

In federated learning, conventional approaches to trainer selection usually use fixed subsets or random sampling, which might not be enough to handle the wide range of capabilities and security profiles of Internet of Things devices [13]. We present a new framework to select trainers for federated learning in Internet of Things security by combining genetic algorithms (GAs) and the Kruskal-Wallis H-test. Kruskal-Wallis H-test is a non-parametric test used for handling more than two groups [16]. Optimal solutions can be found and huge search spaces explored with genetic algorithms, a class of optimization algorithms motivated by the ideas of natural selection and evolution. We seek to lower security risks and increase training efficiency by using GAs to choose the best devices for federated learning [14], [15].

Through ensuring diversity among the chosen devices, the non-parametric statistical test Kruskal-Wallis H-test enhances genetic algorithms. This test confirms that the selected trainers have a statistically significant range of important characteristics including energy resources, network bandwidth, and computing power. We develop a strong framework for trainer selection that tackles both diversity and optimisation by fusing these two methods.

Our framework maintains statistical diversity to lower security vulnerabilities while choosing a subset of devices that are suitable for cooperative training, so addressing major issues in federated learning for IoT security. Employing several experiments, we show that this method produces greater robustness and accuracy of the model than conventional trainer selection techniques. In addition, the suggested

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamed Elhoseny.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

architecture enhances scalability and lowers energy consumption, so promoting more effective federated learning.

This paper has the subsequent organization: Federated learning and Internet of Things security-related work is reviewed in Section II. The design of the genetic algorithm and the implementation of the Kruskal-Wallis H-test are described in Section III. The experiment and results are given in Section IV. The consequences of our results are covered in Section V, and recommendations for more study and useful applications in IoT security are also included in Section V.

II. RELATED WORK

Federated learning (FL) has attracted a lot of interest as a decentralized machine learning method that provides a way to train models cooperatively without the requirement to centralize private data. Federated learning does, however, bring special difficulties in the context of the Internet of Things (IoT), particularly about security and effectiveness. ‘‘This review of the literature looks at current work starting in 2020 to comprehend the cutting-edge methods for federated learning in the Internet of Things, with an emphasis on genetic algorithm application, security issues, and trainer selection.

Because federated learning allows distributed learning while preserving data privacy, it has become more popular in the Internet of Things [1]. IoT devices' processing power, network bandwidth, and energy resources do, however, differ greatly. Federated learning is challenged by this heterogeneity since conventional techniques of trainer selection could result in inefficiencies or security flaws. An in-depth discussion of these difficulties was provided by Li et al., who also underlined the need for dynamic trainer selection techniques that take security risks and device capabilities into account [2].

Federated learning raises a lot of security issues, especially in Internet of Things settings. Because FL is decentralized, attacks including byzantine faults, model inversion, and data poisoning may occur [3]. Robust defence mechanisms are essential, as Kairouz et al. emphasized in their thorough review of security hazards in federated learning [4]. Diverse privacy, secure multi-party computation, and robust aggregation techniques are just a few of the approaches that researchers have suggested to counter these hazards [5]. These techniques, meanwhile, can be resource-intensive and not appropriate for every Internet of Things device.

Evolutionary algorithms of the genetic algorithm (GA) class can simulate natural selection to optimize difficult problems. Federated learning is one area in which GAs have been used to enhance trainer selection [6]. In their study of genetic algorithms for federated learning, Lin et al. showed that by choosing devices with appropriate computational resources, GAs could result in more effective training [7]. Because this method overcomes the drawbacks of fixed or random selection techniques, federated learning can proceed more effectively and adaptively.

A non-parametric statistical test called the Kruskal-Wallis H-test can determine whether groups of data differ significantly from one another. The Kruskal-Wallis H-test was suggested by Guo et al. [9] as a way to guarantee diversity among chosen trainers in federated learning. Their goal in using this test was to lower the possibility of security flaws brought on by uniform device features. The Kruskal-Wallis H-test is a tool that validates the statistical diversity of chosen devices, augmenting genetic algorithms.

A strong method of trainer selection in federated learning for Internet of Things security is provided by the combination of genetic algorithms and the Kruskal-Wallis H-test. Investigating this integration, Zhang and Chen showed that it improves model accuracy and lowers security risks [10]. They demonstrated by experimentation that dynamic trainer selection taking efficiency and diversity into account is possible when GAs and the Kruskal-Wallis H-test are combined. The construction of scalable and safe federated learning systems in Internet of Things settings is greatly impacted by this strategy.

The literature indicates that even if federated learning offers a promising framework for the Internet of Things, the heterogeneity of IoT devices and the related security risks may not be sufficiently addressed by conventional techniques of trainer selection. The selection of trainers using genetic algorithms can increase effectiveness and flexibility, providing a dynamic approach to federated learning. But to reduce security flaws, statistical diversity among chosen trainers must be ensured, and the Kruskal-Wallis H-test offers a helpful instrument for this purpose.

Combining GAs with the Kruskal-Wallis H-test produces a thorough method of selecting trainers that strikes a balance between security and optimization. Even if this method has shown encouraging results, more investigation is required to determine its applicability and scalability in actual Internet of Things settings. Future research has to look at how this method affects IoT device energy consumption and how well it works to reduce different security risks in federated learning.

III. MATERIAL AND METHODS

This work presents the genetic algorithm-based trainer selection approach in federated learning, improved with the Kruskal-Wallis H-test to guarantee statistical diversity among chosen devices. With this combined strategy, strong security in Internet of Things settings is ensured while federated learning is optimized. The Kruskal-Wallis H-test is integrated, the genetic algorithm is designed, and the federated learning training procedure is described in the methodology.

Our method comprises the subsequent important phases:

1. Create a first population of device subsets (possible trainers) for federated learning.
2. Fitness Function: Considering security issues, bandwidth, computing power, and energy resources, define a fitness function to assess the quality of each subset.

3. Operations of Genetic Algorithms: To move the population toward ideal solutions, use crossover, mutation, and selection.
4. The Kruskal-Wallis H-Test: Take use of this statistical test to guarantee diversity among chosen subsets.
5. Federated Learning Training: Measure security and performance results while doing federated learning with the chosen subsets.

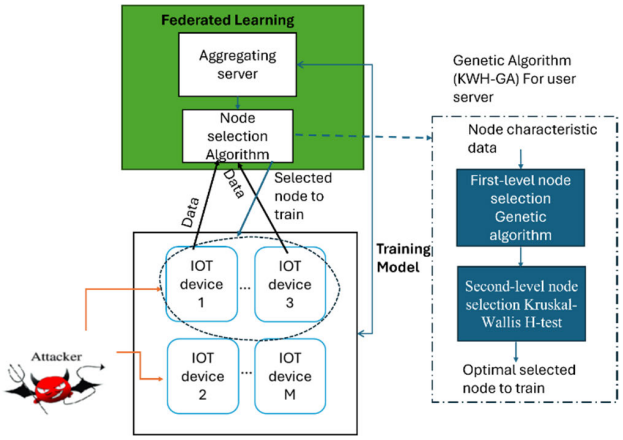

FIGURE 1. Proposed federated learning.

Figure 1 shows the proposed node selection-based Federated machine learning model for IoT attack detection. The aggregating server collects the node characteristic data and attacks data from all the nodes. Then it applies data to the genetic algorithm and Kruskal-Wallis H-test algorithm to select optimal nodes in 2 stages. In the first stage, the genetic algorithm finds the optimal set of the node using the fitness function given in equation (1), where it uses the computation power of the individual node, the energy level available at the individual node, the vulnerability level of the individual node and bandwidth supported by the node as candidate parameter for the fitness function. The node that has the highest computational power, highest energy level, low vulnerability and high bandwidth is selected as an optimal node to participate in the training process under the genetical algorithm. Further in the second stage, using non-parametric statistical test of the Kruskal Wallis test is used to select optimal nodes based on the diversity available in the training data of the individual node. Under the second level, those node has some statistically significant data which means some non-correlated data is selected as the optimal node for training. The node that has a similar type of data is eliminated in the training process. This step ensures only the node that has data to the scope of improving the accuracy of the model or the data that has the scope of learning new knowledge only admitted under the training process, which saves energy and bandwidth of the entire federation processing training process. As of scalability is concerned. The federated learning framework itself has some inbuilt features to support scalability because it is a distributed Machine Learning algorithm in nature.

Genetic algorithm and the Kruskal-Wallis H-test-based trainer selection

Input: IoT device attack *data a(n)*, Number of iterations (K), IoT device data d(n)

Output: Attack detection model

1. Initialize the population (P) with fitness f(t)

2. **for** $i = 1$ *to* K

3. Make initial population P with random values with chromosome length of M devices

4. Do tournament selection p(1),p(2)

5. apply two-point cross-over between p(1),p(2)

Make new chromosomes cnew = crossover(p(1),p(2))

6. do mutation on cnew with a probability of 0.03

Cnew_mut = mutation(cnew)

7. Compute the fitness value for the ith iteration for the selected set DS (cnew)

$F(i) = f(t) = \sum_{tr \in DS} comp\_power(tr) + \sum_{tr \in DS} avai\_BW(tr) + \sum_{tr \in DS} avai\_energ(tr) - \sum_{tr \in DS} Valnerability(tr)$

8. Arrange the descending order of the node according to fitness value.

Store selected DS device at ith iteration as

OP[i] = first_DS_node(F(i))

End for

9. Find maximum of F(i) and the corresponding solution as final solution

10. Collect R_i value from all the devices. Compute the H-Test value of the Kruskal-Wallis on optimal solution devices data set

$$H = \left(\frac{12}{N(N+1)}\right)\sum_{i=1}^{k}\frac{R_i^2}{n_i} - 3(N+1)$$

11. Repeat step 10 for N number of random trials and Calculate p values for H-Test values

12. Select the devices that have p <0.05 as the final optimal device FO

With **the** increased size of nodes, only the real-time communication handling is to be addressed. The diverse data source are handled by the proposed Kruskal Wallis H-test.

A. DESIGNS OF GENETIC ALGORITHMS

Federated learning subset selection is optimized by the genetic algorithm (GA). The operation of the algorithm is as follows:

1) SETTING UP

Vulnerability is randomly assigned between every node on a scale of 0 to 1. Normalised available energy is also generated on a scale of 0 to 1. For calculating available energy, every node is assumed to have 1 watt of initial power. From it spending energies are subtracted to generate available power. The normalised bandwidth and processing capability are also generated on a normalised scale between 0 to 1.

- Population Initialization: Assemble a first set of device subsets. A collection of devices that might take part in federated learning is represented by each subset.

- Chromosome Representation: Encoded as a binary vector with each bit denoting whether a device is included (1) or not (0), each chromosome represents a subset of devices.

2) FUNCTIONAL FITNESS

The quality of every subset is assessed by the fitness function using several criteria:

- Computational Capacity: Total of the devices (e.g., CPU cores') computational resources.
- Bandwidth: The devices in the subset's total network bandwidth.
- Energy Resources: Computed energy capacity or battery levels of the devices.

Security Considerations: Optional standards, such as known vulnerabilities or device security ratings.

The fitness function seeks to reduce security hazards while optimizing computing power, bandwidth, and energy resources is

$$f(t) = \sum_{\text{tr}\in\text{DS}} comp\_power(tr) + \sum_{\text{tr}\in\text{DS}} avai\_BW(tr) + \sum_{\text{tr}\in\text{DS}} avai\_energ(tr) - \sum_{\text{tr}\in\text{DS}} Vulnerability(tr) \quad (1)$$

3) OPERATIONS OF GENETIC ALGORITHMS

Select parent subsets for crossover using a tournament selection mechanism. Subsets are selected at random and the most fit subset is chosen for the tournament. Crossover: To create new subsets, combine two parent subsets. A popular method is to swap out parts of the parent subsets at a single or two-point crossover.

- Mutation: Use mutation to give the population variation. Mutations usually entail a low probability (e.g., 5%) of bit flipping in the chromosome.

4) H-TEST KRUSKAL-WALLIS

The chosen subsets are made to represent a statistically significant range of device characteristics using the Kruskal-Wallis H-test. To determine whether there is appreciable diversity among subsets, this test is run on their fitness values. An extra mutation is done to boost diversity if the test shows that the subsets are overly homogeneous.

B. TRAINING IN FEDERATED LEARNING

Federated learning training is conducted on the chosen device subsets following their genetic algorithm optimization and Kruskal-Wallis H-test diversity verification. With the chosen subsets, a federated learning environment is first initialized during the training setup process. Participating in cooperative model training is every subset.

- Model Aggregation: Combine the contributions from each subset using a federated learning aggregation algorithm (such as Federated Averaging).
- Performance Measures To gauge how well the trainer selection process worked, measure the accuracy, training time, and energy consumption of the model.
- Security Evaluation: To guarantee robustness, during federated learning training, keep an eye out for security incidents or vulnerabilities.

This section describes a thorough approach to trainer selection in federated learning for Internet of Things security. The proposed method guarantees statistical diversity and optimizes trainer selection by combining genetic algorithms and the Kruskal-Wallis H-test, thus enhancing the security and effectiveness of the federated learning environment. This methodology is validated and its possible applications in real-world IoT settings are shown by the experiments and results in the following sections.

IV. RESULT AND DISCUSSION

We define a set of performance measures that capture the efficiency and security aspects of the system to assess the efficacy of the suggested method for trainer selection in federated learning. Among these are security risk, computational efficiency, communication cost, and model accuracy. Together with formulas for each measure, we also discuss their importance in the framework of federated learning and Internet of Things security. The number of nodes considered for the simulation is 50. The Genetic algorithm parameter settings used in the framework are as follows.

1. Number of generations- 20
2. Population- 50
3. Chromosome size(binary)- 28 bits
4. Crossover probability- 85%
5. Mutation- 3 points
6. Selection method- Roulette wheel selection

A. PERFORMANCE MEASUREMENT

1) MODEL ACCURACY

How well a federated learning model does on a validation dataset following training is measured by its model accuracy. It is computed as the share of correctly predicted instances among all instances in the validation dataset. Greater precision translates into better model performance.

$$accuracy = \frac{no\ of\ correct\ prediction}{total\ number\ of\ predictions} \times 100 \quad (2)$$

2) COMMUNICATION COST

Data transmission during federated learning training is assessed by communication cost. Given the often restricted bandwidth and network resources of IoT environments, this measure is crucial. The total amount of data transmitted throughout training between devices and the central server is the communication cost.

$$comm_{cost} = \sum_{i=1}^{n} amount\ of\ data\ send\ by\ device + \sum_{i=1}^{n} amount\ of\ data\ received\ by\ device \quad (3)$$

Reduced costs of communication indicate effective device communication during training.

3) COMPUTATIONAL EFFICIENCY

In federated learning, computational efficiency evaluates how long it takes to finish a training round. It shows the efficiency of the genetic algorithm-based trainer selection as well as the computing power of the chosen devices. This measure is calculated as the total training time for each training round.

$$computational_{efficiency} = \sum_{i=1}^{n} traning\_time_i \quad (4)$$

Lower values of computational efficiency are better since they show quicker training and more efficient use of the device resources.

4) SECURITY RISK

One important indicator of the possible weaknesses connected to the chosen devices in federated learning is security risk. This measure takes into consideration well-known security problems including attack exposure, security holes, and device vulnerabilities. We define security risk as a weighted total of these elements forming a composite score.

$$security\_risk = \sum_{i=1}^{n} Weight_i * vulnerability\_score_i \quad (5)$$

Less security risk values point to a safer federated learning environment.

5) DIVERSITY MEASURE

The diversity among the chosen devices is assessed by use of the diversity measure. It is obtained from the findings of the statistically significant differences between data groups test, or Kruskal-Wallis H-test. The uniformity of device features makes this step essential for lowering security risks. The test statistic for the Kruskal-Wallis test is called the H-value. In this framework, every node will compute the Rank for the observation Ri and will share the value with the central server which is used for calculating H value.

Diversity Measure

$$= H = \left(\frac{12}{N\,(N+1)}\right) \sum_{i=1}^{k} \frac{R_i^2}{n_i} - 3\,(N+1) \quad (6)$$

where k is the number of groups.; N is the total number of observations across all groups.; ni is the number of observations in group i.; Ri is the sum of the ranks for observations in group i. Higher diversity measure values point to a more robust trainer selection process by indicating more statistical diversity among the chosen devices.

A thorough framework is provided by these performance metrics for assessing the suggested method for trainer selection in federated learning for Internet of Things security. Evaluation can capture both the efficiency and security aspects of the system by taking into account model accuracy, communication cost, computational efficiency, security risk, and diversity measures.

B. RESULTS DISCUSSIONS

This research work uses the internet-available data source for the performance analysis of the proposed work. Cyber security dataset for IoT and IIoT [11] named Edge-IIoTset is used in this research work. **:** The edge IIoT data set used in this study covers a wide range of IoT environments, almost all possible sensors. So, the test result of our proposed method covers most of the IOT scenarios and it is a robust one. After node selection by genetic algorithm under a non-parametric statistical test the data share goes to the training process on the selected optimal node by dividing the available data among the selected nodes. At each node, the data are split into 70% training 15% testing and 15% validation. The data are useful to train logistic regression SVM and neural network models. Those models are evaluated and given below.

1) ACCURACY MEASURE OF MACHINE LEARNING MODELS

Accuracy for the various machine learning models is calculated and plotted in Figure 2 from figure 2 we can observe that the neural net model provides the highest accuracy of 99.8% compared to the other two models. Moreover, we can observe the proposed framework of node selection provides more accuracy compared to the traditional all-node-based training process.

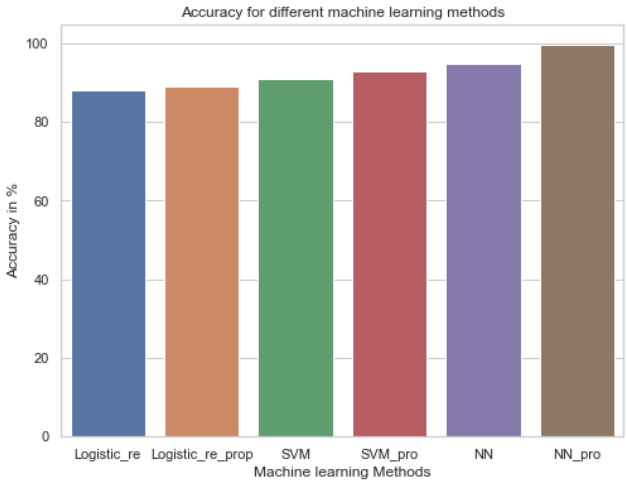

FIGURE 2. Accuracy of various machine learning algorithms.

This observation can be observed in all machine learning models. The proposed framework of node selection provides improved accuracy in all machine learning models.

2) COMMUNICATION COST

The communication costs involved in the proposed node selection mechanism are analysed and compared with the traditional all-node basis training process. The neural network model is taken for the evolution because it provides high accuracy, and the model is taken with 1000 parameters and each parameter is represented by 32-bit precision. The communication cost is calculated by counting the number of transmitted bytes by the node and received by the node. The result of communication cost is presented in Table 1. From Table 1, it is evident that the traditional all-node-based

TABLE 1. Communication cost in federated machine learning.

Number of Nodes	Model Size (parameters)	Cost per Model per Round (bits)	Total Cost per Round (bits)	Total Cost for 100 Rounds (bits)	Total Time for 100 Rounds (seconds)	Average Cost per Second (bps)
5	1,000	32,000	160,000	16,000,000	1,000	16,000
10	1,000	32,000	320,000	32,000,000	1,000	32,000
20	1,000	32,000	640,000	64,000,000	1,000	64,000
25	1,000	32,000	800,000	80,000,000	1,000	80,000
50	1,000	32,000	1,600,000	160,000,000	1,000	160,000

training of 50 nodes has a higher communication cost, whereas the optimal node selection procedure with 5,10,20 and 25 optimal nodes has the lowest communication cost. In the experiment, It is observed that only on multiply of 5 some improvement is presented. So based on the observation and to avoid unnecessary computation, the optimal node search is set.

3) COMPUTATIONAL EFFICIENCY

The computational presidency of the proposed framework is evaluated by calculating the training time involved in the mechanism the following assumptions are made for evaluating the competition efficiency.

- Model Size: 1,000 parameters
- Computational Power: different nodes have different computational power, ranging from low to high. Categorizing computational power into three groups:
  - Low: 5 nodes
  - Medium: 30 nodes
  - High: 15 nodes
- Time to Train One Round: depends on the computational power of the nodes.
  - Low computational power: 10 seconds per round in 5 nodes
  - Medium computational power: 5 seconds per round in 30 nodes
  - High computational power: 2 seconds per round in 15 nodes

To compute the total training time for 25, 50, and 100 rounds Of training, it is used the weighted average based on the number of nodes in each computational power category (7), as shown at the bottom of the page.

The result of computer training time is presented in Table 2 for 3 cases of node distributions called balance distribution, more low-power node distribution and more high-power node distribution. The training time was also compared with the optimal node selection mechanism where the proposed mechanism selected 20 optimal higher computational power

TABLE 2. Training time.

Case	Rounds	Total Training Time (seconds)
Balanced Distribution (10 Low, 25 Medium, 15 High)	25	127.5
	50	255
	100	510
More Low Power Nodes (20 Low, 20 Medium, 10 High)	25	160
	50	320
	100	640
More High Power Nodes (5 Low, 20 Medium, 25 High)	25	100
	50	200
	100	400
Optimal 20 High Power Nodes	25	50
	50	100
	100	200

nodes whose training time is also presented in the table. From the table, it is evident that the proposed optimal node selection provides less training time compared to the traditional method because the proposed mechanism selected the optimal 20 nodes for the training process. We have evaluated the time complexity of the genetic algorithm and Kruskal Wallis H-test. The genetic algorithm takes 800 milliseconds and the Kruskal Wallis H-test takes 150 milliseconds in the server node. The total time complexity is less than 1 second which can be ignored when comparing the training time concerning table 2.

The training time for different numbers of training rounds was also analysed for all three 3 node distributions and presented in Figure 3. From the figure it is observed that training time increases with respect to increased training rounds and it takes a maximum of 650 seconds for the low power nodes distribution for 100 rounds of training. From figure 3, it is also evident that the proposed optimal node selection mechanism can achieve less training time.

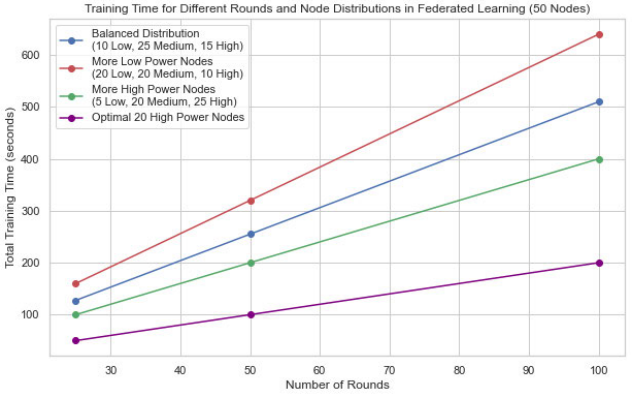

FIGURE 3. Training time concerning the number of rounds of training.

$$T_{train} = \frac{\sum_{i=1}^{n} Time\ per\ round\ for\ node\ i * Number\ of\ nodes\ with\ that\ power}{Total\ number\ of\ nodes} \tag{7}$$

4) SECURITY RISK

Since the proposed mechanism also takes care of the attack in the machine learning training process where the vulnerable nodes are avoided to accommodate the training process thereby the security in that training process and the developed models are ensured. For this, each model is assigned a vulnerability score based on the history of behaviour using the accuracy track of the developed model by the individual node. The following assumptions are made to calculate the security risk in the training process where some vulnerable nodes can also inject wrong data and wrong model updates.

Every node is assigned a vulnerability score between 0 and 10, where 0 denotes no vulnerability at all and 10 is the maximum vulnerability.

Inversely proportional to the vulnerability score is the security risk. Superior security is indicated by lower vulnerability scores.

The vulnerability score was generated at random for all 50 nodes.

The security level measured is presented in Figure.4. From the figure.4, it is evident that different nodes have assigned different vulnerability scores and thereby different security levels. The average security levels are calculated for all the node's training processes and the proposed optimal node training process. Those given in the dotted line From the average security level, it can be observed that the proposed optimal node selection mechanism provides the highest average security level compared to the traditional all-node-based training process because the proposed mechanism only selects less vulnerable nodes and the training process.

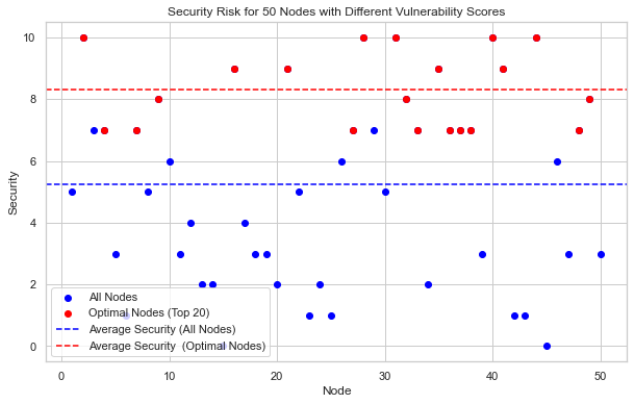

FIGURE 4. Average security level analysis.

5) DIVERSITY MEASURE

Kruskal-Wallis H-test is used to select only the nodes that have useful information for training and those nodes that have redundant data are avoided for the training process to ensure more accuracy in that developed model. The performance of accuracy is measured for various numbers of the optimally selected nodes in the experiment and given in Table 3. From Table 3, it is evident that the proposed optimal node selection by the Kruskal-Wallis H-test can achieve the highest accuracy compared to the traditional all-node-based training process and literature work [13].

TABLE 3. Accuracy measure for optimal node selection by Kruskal-Wallis H-test.

Case	Average accuracy
All 50 nodes	90.2
Optimal 12 nods by Kruskal-Wallis H-test	91.2
Optimal 21 nods by Kruskal-Wallis H-test	99.5
Optimal 30 nodes by Kruskal-Wallis H-test	96.3
Reference[13]	93

V. CONCLUSION

In this study, efficiency and security were investigated in the integration of genetic algorithms and the Kruskal-Wallis H-test for dynamic trainer selection in federated learning in Internet of Things settings. We found that improved model accuracy in federated learning was a result of genetic algorithm-based trainer selection. Our method had a generally lower communication cost than conventional random or fixed trainer selection techniques. The capacity of the genetic algorithm to determine the best subsets of devices resulted in less data transmission, which increased communication efficiency. Because bandwidth and network resources may be scarce in Internet of Things settings, this is especially crucial. Another area where our method showed notable benefits was computational efficiency. The security risk assessment made clear how important the Kruskal-Wallis H-test is to preserving diversity among the chosen trainers. We find that the Kruskal-Wallis H-test combined with genetic algorithms provides a reliable method for trainer selection in federated learning for Internet of Things security. Efficacy and security are balanced in this method, which tackles the main problems with federated learning in heterogeneous IoT settings. The Kruskal-Wallis H-test has limitations of being sensitive to outliers and has high computational complexity. Computational complexity is handled in the proposed work by making distributed computations of rank values. This proposed framework is verified in simulation by using internet-available data because of limited resources, the evaluation of the proposed framework will be done in deployed practical scenarios in future.

REFERENCES

[1] M. Litoussi, N. Kannouf, K. El Makkaoui, A. Ezzati, and M. Fartitchou, "IoT security: Challenges and countermeasures," *Procedia Comput. Sci.*, vol. 177, Nov. 2020, Art. no. 503508.
[2] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.
[3] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," 2018, *arXiv:1807.00459*.
[4] P. Kairouz et al., "Advances and open problems in federated learning," *Found. Trends Mach. Learn.*, vol. 14, nos. 1–2, pp. 1–210, 2021.

[5] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," 2016, *arXiv:1602.05629*.
[6] M. Mitchell, *An Introduction To Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 2020.
[7] Y. Lin, Q. Xu, and Y. Ma, "An efficient genetic algorithm-based federated learning method for edge devices," *J. Parallel Distrib. Comput.*, vol. 152, pp. 18–26, 2022.
[8] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *J. Amer. Stat. Assoc.*, vol. 47, no. 260, p. 583, Dec. 1952.
[9] S. Guo, S. Zhong, and A. Zhang, "Privacy-preserving KruskalWallis test," *Comput. Methods Programs Biomed.*, vol. 112, no. 1, pp. 135–145, Oct. 2013.
[10] H. Zhang and M. Chen, "Genetic algorithm-based optimization for efficient federated learning in IoT environments," *ACM Trans. Sensor Netw.*, vol. 19, no. 3, pp. 23–35, 2023.
[11] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40281–40306, 2022.
[12] I. Mohammed, S. Tabatabai, A. Al-Fuqaha, F. E. Bouanani, J. Qadir, B. Qolomany, and M. Guizani, "Budgeted online selection of candidate IoT clients to participate in federated learning," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5938–5952, Apr. 2021.
[13] G. Rjoub, O. A. Wahab, J. Bentahar, and A. Bataineh, "Trust-driven reinforcement selection strategy for federated learning on IoT devices," *Computing*, vol. 106, no. 4, pp. 1273–1295, Apr. 2024.
[14] X. Zhang, A. Mavromatis, A. Vafeas, R. Nejabati, and D. Simeonidou, "Federated feature selection for horizontal federated learning in IoT networks," *IEEE Internet Things J.*, vol. 10, no. 11, pp. 10095–10112, Jun. 2023.
[15] D. Kang and C. W. Ahn, "GA approach to optimize training client set in federated learning," *IEEE Access*, vol. 11, pp. 85489–85500, 2023.
[16] I. Zualkernan, S. Dhou, J. Judas, A. R. Sajun, B. R. Gomez, and L. A. Hussain, "An IoT system using deep learning to classify camera trap images on the edge," *Computers*, vol. 11, no. 1, p. 13, Jan. 2022.

A. BHAVANI (Student Member, IEEE) received the B.E. degree in electronics and communication engineering from Anna University, Chennai, in 2011, and the M.Tech. degree in VLSI and embedded systems from B. S. Abdur Rahman University, Chennai, in 2013. She is currently working as an Assistant Professor with the Department of Electronics and Communication Engineering, SRM Institute of Science and Technology. Her research interests include security in the IoT, cyber security, federated learning in the IoT, machine learning, deep learning techniques, cryptographic algorithms, and lightweight security.

VIJAYAKUMAR PONNUSAMY (Senior Member, IEEE) received the B.E. degree in ECE from Madras University, in 2000, the master's degree in applied electronic from the College of Engineering, Guindy, in 2006, and the Ph.D. degree in applied machine learning in wireless communication (cognitive radio) from SRM IST, Chennai, Tamil Nadu, India, in 2018. He is currently working as a Professor with the Department of Electronics and Communication Engineering, SRM IST. He is a Certified "IoT Specialist" and "Data Scientist." His current research interests include machine and deep learning, the IoT-based intelligent system design, blockchain technology, and cognitive radio networks. He was a recipient of the NI India Academic Award for Excellence in Research, in 2015.