

Trust-aware reliability-based aggregation framework for resilient federated learning in critical infrastructure systems

Stanislava Kudrenko^{1,*†}, Vitalii Alkema^{1†}, Valerii Kozlovskiy^{1†} and Diana Kozlovskaya^{1†}

¹Scientific Cyber Security Association of Ukraine, Mykhaila Dontsia Str., 2A, Kyiv, 03161, Ukraine

Abstract

Federated learning is a promising approach for training machine learning models in distributed environments where raw data cannot be centralized due to privacy, security, or organizational constraints. However, the aggregation stage remains vulnerable to unreliable, abnormal, or adversarially manipulated client updates. This problem is especially important for critical infrastructure systems, where degraded model behavior may affect anomaly detection, intrusion detection, traffic monitoring, and early warning mechanisms. This paper proposes a trust-aware reliability-based aggregation framework for resilient federated learning in critical infrastructure systems. The framework extends conventional FedAvg by introducing a reliability coefficient for each local client update. This coefficient is calculated using the normalized deviation of a local update from a reference update formed by the aggregation server. The final aggregation weight combines the nominal dataset-size-based contribution with the estimated reliability of the corresponding update. As a result, suspicious or strongly deviated updates are softly down-weighted instead of being rigidly excluded. The proposed framework was evaluated in a server-client federated learning simulation under mixed unreliable-client behavior. The results show higher accuracy, lower loss, and more stable convergence compared with conventional FedAvg. The analysis of aggregation weights and reliability coefficients confirms that unreliable clients receive substantially lower final contributions during aggregation.

Keywords

cybersecurity, anomaly detection, federated learning, critical infrastructure, trust-aware aggregation, reliability-based aggregation, edge-fog computing, unreliable clients, model poisoning attacks, resilient machine learning

1. Introduction

Federated learning has become an important paradigm for training machine learning models in distributed environments where raw data cannot be directly centralized due to privacy, security, technical, or organizational constraints. Instead of transferring local datasets to a central server, participating clients train local models and exchange only model updates for global aggregation. This approach is especially relevant for critical infrastructure systems, industrial Internet of Things environments, cyber-physical systems, edge-fog architectures, and distributed monitoring platforms, where data are naturally generated and stored across heterogeneous nodes.

In critical infrastructure environments, distributed learning may support anomaly detection, intrusion detection, traffic monitoring, equipment state assessment, predictive diagnostics, and early warning mechanisms. However, the reliability of the global model strongly depends on the quality and integrity of local updates received from participating clients. In real-world edge and fog environments, some nodes may operate under unstable communication conditions, limited computational resources, noisy data, local distribution shifts, software failures, or adversarial influence. As a result, local model updates may become unreliable, biased, abnormal, or intentionally manipulated.

CMiGIN'26: Fifth International Workshop on Conflict Management in Global Information Networks, June 26–27, 2026, Krakow, Poland

*Corresponding author.

†These authors contributed equally.

✉ stanislava@i.ua (S. Kudrenko); 9010908@stud.kai.edu.ua (V. Alkema); valerii.kozlovskiy@npp.kai.edu.ua (V. Kozlovskiy); 8542876@stud.kai.edu.ua (D. Kozlovskaya)

ORCID 0000-0002-0759-3908 (S. Kudrenko); 0009-0000-0009-8237 (V. Alkema); 0000-0002-8301-5501 (V. Kozlovskiy); 0009-0004-6223-0319 (D. Kozlovskaya)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

The conventional Federated Averaging approach combines local updates mainly according to the size of local datasets. Although this strategy is effective under normal operating conditions, it does not explicitly assess the trustworthiness or reliability of each client. Therefore, abnormal or compromised clients may still significantly influence the global model if their updates are included in the aggregation process with regular weights. This limitation becomes especially critical in adversarial environments, where malicious clients may attempt to degrade model performance, introduce hidden backdoors, or distort the decision boundary of the global model.

Existing robust aggregation methods attempt to reduce the impact of abnormal updates by using statistical distance, coordinate-wise filtering, or Byzantine-resilient selection rules. These approaches are important for improving the security of federated learning, but many of them rely mainly on hard rejection or static assumptions about client behavior. In critical infrastructure systems, client reliability may change dynamically over time. A previously reliable node may become unstable or compromised, while a temporarily abnormal update should not always lead to complete exclusion of the client from the learning process. Therefore, there is a need for aggregation mechanisms that combine robustness, interpretability, adaptive trust assessment, and soft reduction of suspicious contributions.

This paper proposes a trust-aware reliability-based aggregation framework for resilient federated learning in critical infrastructure systems. The main idea of the proposed framework is to introduce a reliability coefficient for each local update and use it during global aggregation. The coefficient reflects the degree of consistency between a client update and a reference aggregation behavior. Instead of treating all clients equally or fully excluding suspicious participants, the proposed framework adaptively reduces the contribution of unreliable updates while preserving useful information from legitimate clients.

The proposed framework is designed for scenarios in which edge or fog nodes participate in federated training under uncertain or adversarial conditions. It focuses on improving the resilience of the global model against abnormal local behavior, node degradation, and manipulated updates. The framework includes a threat-oriented problem formulation, reliability estimation, adaptive weighting of local updates, and simulation-based evaluation under different levels of unreliable client participation.

2. Related work

Federated learning was introduced as a distributed machine learning paradigm that enables collaborative model training without direct centralization of raw data. The classical Federated Averaging algorithm proposed by McMahan et al. [1] became a baseline approach for combining local model updates received from multiple clients. Further studies extended this concept by considering communication efficiency, system scalability, non-IID data distribution, privacy preservation, and heterogeneous client participation [2, 3, 4]. These features make federated learning especially relevant for distributed cyber-physical systems, industrial IoT infrastructures, and edge-fog environments.

A broad analysis of federated learning challenges is presented in the works of Kairouz et al. [3] and Li et al. [4]. These studies emphasize that federated learning differs from conventional distributed optimization because of unstable client availability, heterogeneous data, limited communication resources, and privacy constraints. For critical infrastructure systems, these problems are especially important, since participating nodes may differ in computational capabilities, local data quality, network stability, and exposure to cyber threats.

Security and privacy in federated learning have also been widely studied. Secure aggregation protocols, such as the approach proposed by Bonawitz et al. [5], are aimed at protecting the confidentiality of local model updates during aggregation. However, secure aggregation mainly addresses privacy preservation and does not fully solve the problem of unreliable or malicious updates. A client may keep its local data private and still send a distorted, abnormal, or intentionally manipulated update to the server. Therefore, privacy-preserving aggregation should be complemented by mechanisms for evaluating the reliability of local updates.

A significant group of studies is devoted to Byzantine-robust aggregation. Blanchard et al. [6]

proposed Krum as a method for selecting an update that is close to the majority of other updates and thus less likely to be malicious. Other robust aggregation approaches, such as coordinate-wise median and trimmed mean, reduce the influence of extreme values in model parameters or gradients [7]. Bulyan further combines selection and robust aggregation mechanisms to improve Byzantine tolerance [8]. These methods demonstrate that robust aggregation can significantly improve the resistance of distributed learning compared with simple averaging.

At the same time, many Byzantine-resilient methods are based mainly on statistical distances or coordinate-level filtering. Such approaches are effective for detecting strongly deviated updates, but they may be less flexible when client reliability changes dynamically over time. In edge-fog and industrial IoT environments, a node may become unreliable because of communication degradation, local data drift, software faults, temporary overload, or adversarial influence. Conversely, a single abnormal update does not always mean that the client should be permanently excluded from the training process. This motivates the development of adaptive aggregation mechanisms that use soft reliability-based weighting instead of only hard rejection.

Model poisoning and data poisoning attacks represent another important threat category in federated learning. In such attacks, adversarial clients attempt to manipulate the global model by sending malicious local updates or by training on corrupted local data. Backdoor attacks and local model poisoning attacks show that even a limited number of compromised clients can significantly affect global model behavior [7, 9, 10]. These threats are especially dangerous for critical infrastructure monitoring systems, where incorrect model behavior may lead to missed anomalies, delayed incident detection, or incorrect interpretation of system states.

Trust-based and reliability-aware approaches provide a promising direction for improving the resilience of federated learning. FLTrust [11] introduced the idea of trust bootstrapping for Byzantine-robust federated learning. This approach shows that trust estimation can be used to reduce the impact of malicious clients during aggregation. However, the development of lightweight, interpretable, and simulation-friendly reliability mechanisms remains relevant, especially for resource-constrained IoT, edge, and fog environments. In such systems, aggregation should not only be robust, but also computationally simple and suitable for repeated use across multiple communication rounds.

The issue of node reliability is also closely related to critical infrastructure edge-fog monitoring. In previous work, Kudrenko, Nimych, and Makieiev [12] considered the problem of predicting node compromise in edge and fog environments for critical infrastructure. That study emphasizes the importance of early identification of unreliable or potentially compromised nodes. The present paper continues this direction in the context of federated learning by focusing not only on detecting unreliable behavior, but also on reducing its influence during global model aggregation.

Thus, existing studies provide a strong foundation for federated learning, secure aggregation, Byzantine robustness, and poisoning attack analysis. However, there remains a need for aggregation mechanisms that combine trust assessment, reliability-based weighting, and resilience to abnormal client behavior in critical infrastructure environments. The proposed framework addresses this gap by introducing a reliability coefficient into the aggregation process and using it for adaptive correction of client contributions during federated learning.

3. Problem statement

Federated learning is considered as a distributed training process involving a central aggregation server and a set of participating clients. In critical infrastructure systems, such clients may represent edge devices, fog nodes, industrial monitoring units, or local data centers. Each client trains a local model using its private dataset and sends only model updates to the central server.

The main problem addressed in this paper is the vulnerability of standard aggregation to unreliable or abnormal client updates. In conventional Federated Averaging, local updates are usually weighted according to the size of local datasets. However, this approach does not explicitly evaluate whether a client update is reliable, suspicious, or significantly deviated from the expected global learning behavior.

In critical infrastructure environments, unreliable updates may appear due to local data drift, communication instability, sensor faults, software failures, or adversarial manipulation. If such updates are included in the global aggregation with regular weights, they may slow down convergence, reduce model accuracy, or distort the global decision function.

Therefore, the task of this study is to develop an aggregation framework that combines the nominal contribution of each client with an additional reliability assessment. The framework should reduce the influence of suspicious updates while preserving useful information from legitimate clients. The formalization of this task and the proposed reliability-based aggregation mechanism are presented in the following section.

4. Formalization of the trust-aware reliability-based aggregation framework

The proposed framework considers federated learning as a sequence of communication rounds between a central aggregation server and a set of distributed clients. In critical infrastructure systems, such clients may correspond to edge devices, fog nodes, industrial monitoring units, local control subsystems, or geographically distributed data centers. The main objective of the framework is to increase the resilience of global model training when some local updates become unreliable, abnormal, or adversarially manipulated.

Let the federated learning system include N participating clients. At the t -th communication round, the central server sends the current global model w^t to the selected clients. Each client trains the model on its local dataset and forms a local update:

$$\Delta w_i^t = w_i^t - w^t, \quad i = 1, 2, \dots, N. \quad (1)$$

Here, w^t is the current global model, w_i^t is the locally trained model of the i -th client, and Δw_i^t is the local model update submitted to the aggregation server.

In the conventional Federated Averaging approach, the contribution of each client is usually determined by the size of its local dataset:

$$\alpha_i^t = \frac{n_i}{\sum_{j=1}^N n_j}, \quad (2)$$

where n_i is the number of training samples stored by the i -th client. The coefficient α_i^t reflects the nominal contribution of the client and satisfies

$$\sum_{i=1}^N \alpha_i^t = 1. \quad (3)$$

Under normal conditions, the global model update can be calculated as

$$w^{t+1} = w^t + \sum_{i=1}^N \alpha_i^t \Delta w_i^t. \quad (4)$$

However, in adversarial or unreliable environments, the size of the local dataset does not guarantee the correctness or usefulness of the corresponding update. A client may have a large local dataset and still generate an abnormal, noisy, or intentionally manipulated update. Therefore, the proposed framework introduces an additional reliability assessment stage before global aggregation.

To estimate the typical behavior of client updates in the current communication round, the server forms a reference update:

$$\Delta w_{ref}^t = \text{median}(\Delta w_1^t, \Delta w_2^t, \dots, \Delta w_N^t). \quad (5)$$

The median operation is applied component-wise. This makes the reference update less sensitive to extreme values and strongly deviated local updates. Therefore, Δw_{ref}^t can be interpreted as an approximation of the typical update behavior of the majority of clients.

The deviation of the i -th local update from the reference update is calculated using the normalized Euclidean distance:

$$d_i^t = \frac{\|\Delta w_i^t - \Delta w_{ref}^t\|_2}{\|\Delta w_{ref}^t\|_2 + \varepsilon}, \quad (6)$$

where $\varepsilon > 0$ is a small constant used to avoid division by zero. The normalization reduces the dependence of the deviation measure on the absolute scale of model updates, which may change significantly during different training stages.

On the basis of the normalized deviation, the reliability coefficient of the i -th client update is defined as

$$r_i^t = \exp(-\lambda d_i^t), \quad (7)$$

where $\lambda > 0$ is a sensitivity parameter. If the local update is close to the reference update, then d_i^t is small and r_i^t remains close to 1. If the update strongly deviates from the reference behavior, then d_i^t increases and r_i^t decreases.

The final reliability-corrected aggregation weight is calculated as

$$\gamma_i^t = \frac{\alpha_i^t r_i^t}{\sum_{j=1}^N \alpha_j^t r_j^t}. \quad (8)$$

This normalization ensures that

$$\sum_{i=1}^N \gamma_i^t = 1. \quad (9)$$

The global model is then updated according to the reliability-based aggregation rule:

$$w^{t+1} = w^t + \sum_{i=1}^N \gamma_i^t \Delta w_i^t. \quad (10)$$

Thus, the proposed framework does not replace the classical FedAvg principle completely. Instead, it extends it by introducing a reliability-aware correction of local client contributions. The nominal weight α_i^t preserves the dataset-size-based logic of FedAvg, while the coefficient r_i^t reflects the estimated reliability of the corresponding update. As a result, reliable updates preserve their influence on the global model, whereas suspicious or strongly deviated updates are softly down-weighted.

The workflow of the proposed framework is illustrated in Figure 1.

At each communication round, the central server distributes the current global model w^t to participating clients. Each client trains the model locally and forms the update Δw_i^t . The server then receives the set of local updates, calculates the nominal FedAvg weights α_i^t according to (2), and constructs the reference update Δw_{ref}^t according to (5). After that, the normalized deviation d_i^t is estimated for each local update according to (6), and the reliability coefficient r_i^t is calculated according to (7). Based on these values, the final reliability-corrected aggregation weight γ_i^t is determined according to (8). Finally, the global model is updated according to (10), and the process continues at the next communication round. The proposed framework is especially suitable for heterogeneous critical infrastructure environments because it does not require access to raw local data and does not rely on rigid exclusion of suspicious clients. Instead, it provides a soft adaptive mechanism for reducing the influence of unreliable updates while preserving potentially useful information from legitimate but heterogeneous participants.

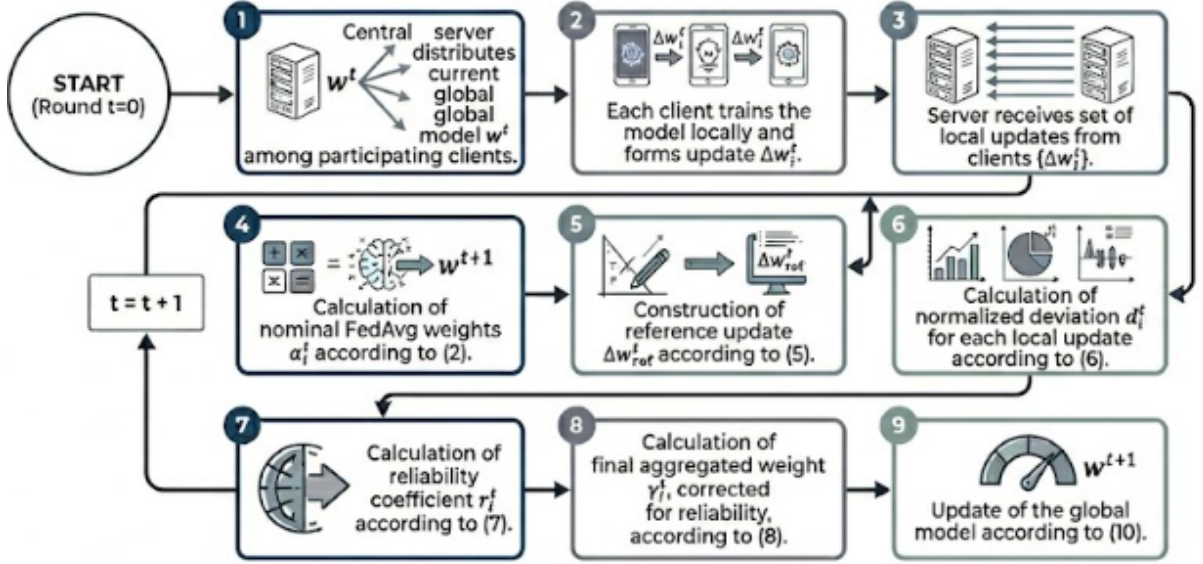


Figure 1: Workflow of the proposed trust-aware reliability-based aggregation framework.

5. Simulation model and experimental setup

The experimental simulation was implemented in Python using PyTorch for local neural network training, NumPy for numerical operations, and Matplotlib for visualization of the obtained results. The federated learning process was reproduced as a server-client simulation without using an external federated learning framework. This made it possible to directly implement and compare the conventional FedAvg aggregation rule and the proposed trust-aware reliability-based aggregation framework.

The simulation was based on a synthetic binary classification dataset. The dataset was generated to imitate two classes of network states: normal behavior and anomalous behavior. Each sample was represented by a numerical feature vector, and the target label belonged to one of two classes. The use of a synthetic dataset made it possible to control the level of separability between classes and to reproduce the behavior of unreliable clients under repeatable conditions.

Let the test dataset be denoted as

$$\mathcal{D}_{test} = \{(x_k, y_k)\}_{k=1}^M, \quad (11)$$

where x_k is the feature vector of the k -th sample, $y_k \in \{0, 1\}$ is the corresponding class label, and M is the number of test samples.

The training data were divided among $N = 10$ simulated clients. A moderately non-IID partitioning strategy was used, so that each client received data from both classes, but with different class proportions. This allowed the simulation to reproduce a more realistic federated learning setting, where local client datasets are not identical.

The global model was represented by a lightweight feed-forward neural network with two hidden layers. At each communication round, the server sent the current global model to all clients. Each client trained the model locally for a fixed number of local epochs and returned only the model update to the server. The simulation was performed for $T = 25$ communication rounds.

The local update of the i -th client at the t -th communication round was calculated as

$$\Delta w_i^t = w_i^t - w^t, \quad (12)$$

where w^t is the current global model and w_i^t is the model obtained after local training on the i -th client.

To evaluate the behavior of the aggregation methods under unreliable conditions, a part of the clients was marked as unreliable. In the performed simulation, the share of unreliable clients was set to

$$\rho = 0.3, \quad (13)$$

which corresponds to 30% of the participating clients. These clients generated distorted local updates before sending them to the server.

A mixed distortion scenario was used. In this scenario, one part of unreliable clients generated noisy updates, while another part generated sign-reversed updates. The noisy update was represented as

$$\widetilde{\Delta w}_i^t = \Delta w_i^t + \eta_i^t, \quad (14)$$

where η_i^t is a random noise term. The sign-reversal distortion was represented as

$$\widetilde{\Delta w}_i^t = -\beta \Delta w_i^t, \quad (15)$$

where β is the attack intensity parameter. In the experiment, $\beta = 1.2$ was used. This scenario made it possible to imitate both weakly abnormal behavior and stronger adversarial manipulation of local updates.

At each communication round, the same set of submitted updates was aggregated by two methods. The first method was conventional FedAvg, where aggregation weights depend only on the relative size of local client datasets. The second method was the proposed reliability-based framework, where the nominal FedAvg weights were corrected using reliability coefficients calculated from the deviation of local updates from the reference update.

The simulation parameters are summarized in Table 1.

Table 1
Simulation Parameters

Parameter	Value	Description
N	10	Number of simulated clients
T	25	Number of communication rounds
E	2	Number of local training epochs
ρ	0.3	Share of unreliable clients
β	1.2	Intensity of sign-reversal distortion
λ	0.6	Sensitivity parameter of the reliability coefficient
Dataset type	synthetic binary	Normal/anomalous state classification
Model type	feed-forward neural network	Two hidden layers
Software	Python, PyTorch, NumPy, Matplotlib	Simulation and visualization tools

The simulation procedure included the following stages. First, the synthetic binary dataset was generated and divided into training and test subsets. Second, the training data were distributed among ten clients using a moderately non-IID partitioning scheme. Third, at each communication round, each client trained a local neural network model and produced a local update. Fourth, unreliable clients distorted their updates according to the mixed distortion scenario. Fifth, the same set of updates was aggregated using FedAvg and the proposed reliability-based aggregation rule. Sixth, both global models were evaluated on the same test set.

The classification performance was evaluated using accuracy, while convergence behavior was analysed using the loss value over communication rounds. In addition, the average final aggregation weights and average reliability coefficients were monitored separately for reliable and unreliable clients.

The purpose of this experimental setup was not to reproduce a specific industrial network dataset, but to verify the behavior of the proposed aggregation framework under controlled and repeatable conditions. The obtained simulation results make it possible to analyse whether the proposed method

can reduce the influence of unreliable local updates and improve the stability of federated learning compared with conventional FedAvg.

6. Simulation results and discussion

The simulation results were obtained for the experimental setup described in Section 5. The federated learning process was simulated for $T = 25$ communication rounds with $N = 10$ participating clients. A mixed unreliable-client scenario was used, in which 30% of clients generated distorted local updates before sending them to the aggregation server. The conventional FedAvg algorithm and the proposed trust-aware reliability-based aggregation framework were evaluated under the same input conditions.

The first stage of the analysis was focused on the classification accuracy of the global model. The accuracy dynamics over communication rounds are shown in Figure 2.

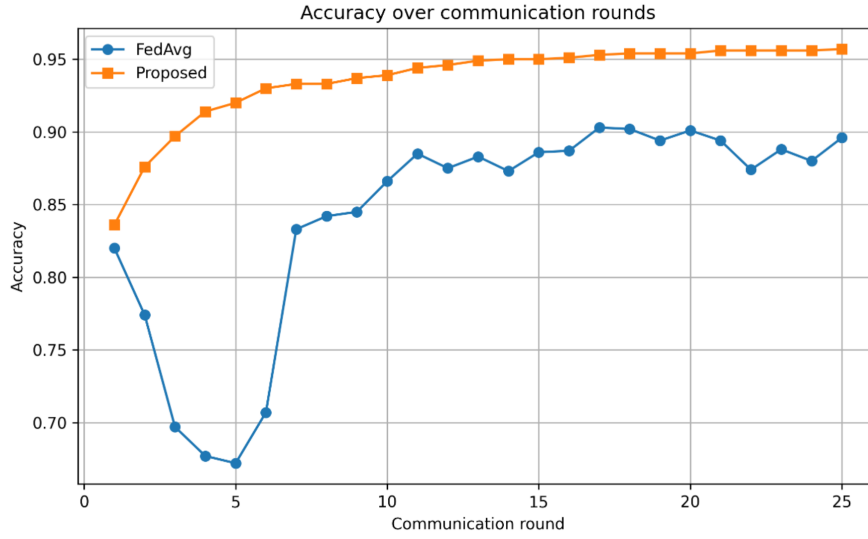


Figure 2: Accuracy dynamics over communication rounds.

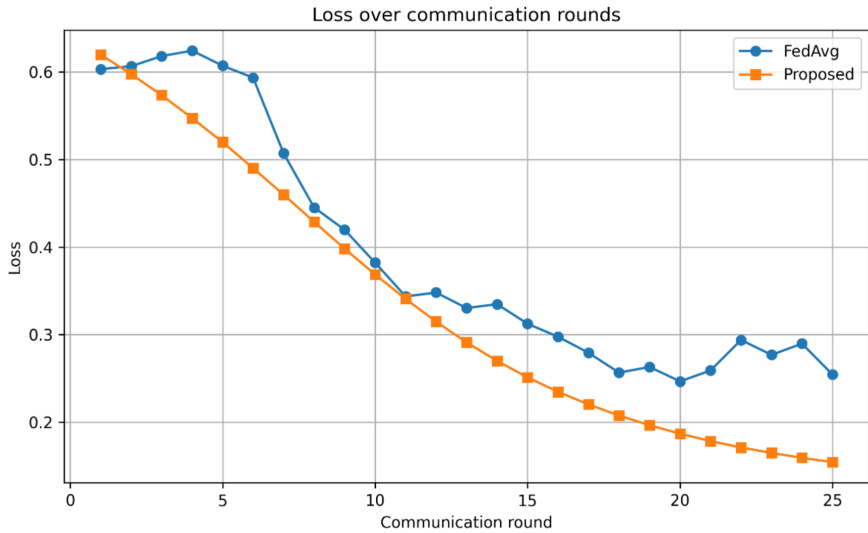


Figure 3: Loss dynamics over communication rounds.

As shown in Figure 2, the proposed framework demonstrates more stable accuracy growth compared with conventional FedAvg. Under mixed unreliable client behavior, FedAvg shows a noticeable decrease

in accuracy during the early communication rounds and remains more unstable during further training. In contrast, the proposed method provides smoother growth and reaches a higher final classification performance. This indicates that reliability-based correction of aggregation weights reduces the negative influence of distorted local updates on the global model.

The convergence behavior of both methods was analysed using the loss value over communication rounds. The corresponding results are presented in Figure 3.

The loss dynamics confirm the advantage of the proposed framework. FedAvg demonstrates higher sensitivity to abnormal client updates, which leads to less stable convergence. The proposed method provides a more consistent decrease in the loss value and reaches a lower final loss. This behavior shows that the use of reliability coefficients improves the stability of the global model training process under unreliable client participation.

The internal behavior of the proposed aggregation mechanism was analysed by comparing the average final aggregation weights assigned to reliable and unreliable clients. The results are shown in Figure 4.

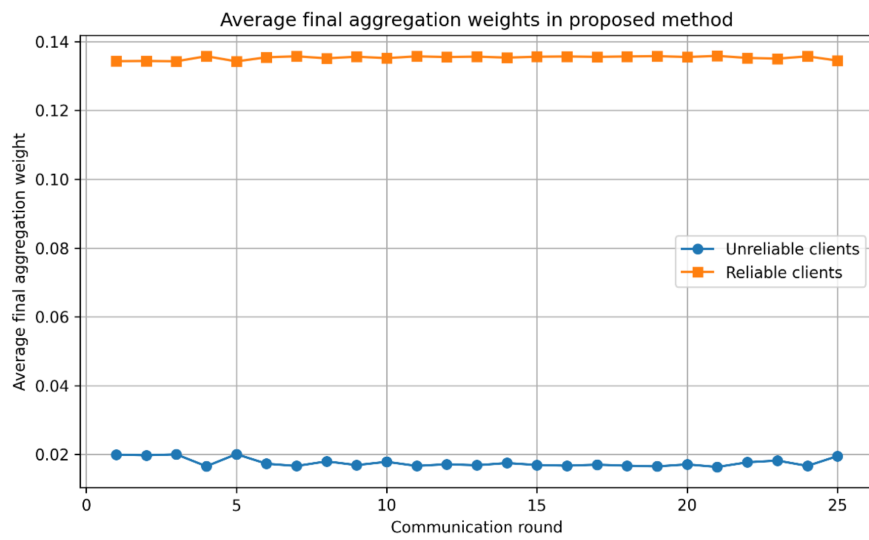


Figure 4: Average final aggregation weights for reliable and unreliable clients.

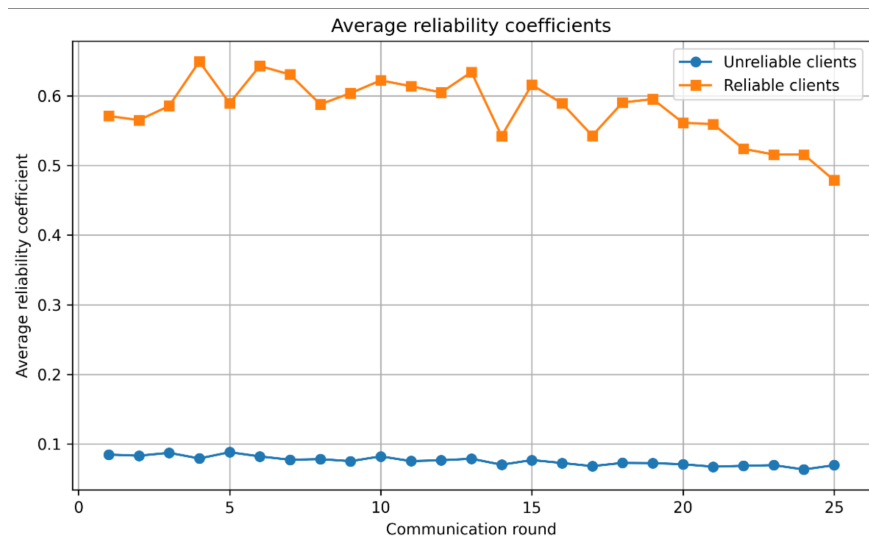


Figure 5: Average reliability coefficients for reliable and unreliable clients.

Figure 4 demonstrates that unreliable clients receive substantially lower final aggregation weights

than reliable clients. This means that the proposed framework does not simply improve the final classification metrics, but also directly changes the aggregation process. Suspicious updates are not fully removed from training, but their contribution to the global model is significantly reduced. Such soft down-weighting is important for heterogeneous federated learning environments, where complete exclusion of clients may lead to the loss of potentially useful local information.

The behavior of reliability coefficients is presented in Figure 5.

Figure 5 shows that the reliability coefficients of unreliable clients remain considerably lower than those of reliable clients throughout the simulation. This confirms that the normalized deviation from the reference update can be used as an informative indicator of abnormal local behavior. Reliable clients preserve higher reliability values, while distorted updates are penalized by lower coefficients. As a result, the final aggregation weights are corrected in favor of more consistent local updates.

Overall, the obtained results show that the proposed trust-aware reliability-based aggregation framework improves the resilience of federated learning under unreliable client behavior. Compared with FedAvg, the proposed method demonstrates higher accuracy, lower loss, and more stable convergence. At the same time, the analysis of aggregation weights and reliability coefficients confirms that the improvement is achieved through adaptive reduction of suspicious client contributions.

These results support the main idea of the proposed framework: aggregation in federated learning should take into account not only the nominal contribution of each client, determined by the size of its local dataset, but also the reliability of the submitted update. In critical infrastructure systems, where edge or fog nodes may become unstable, degraded, or partially compromised, such reliability-aware aggregation can reduce the influence of abnormal updates and improve the stability of distributed model training.

7. Conclusions

This paper proposed a trust-aware reliability-based aggregation framework for resilient federated learning in critical infrastructure systems. The main idea of the framework is to extend conventional FedAvg aggregation by introducing an additional reliability assessment of local client updates. The proposed approach combines the nominal dataset-size-based contribution of each client with a reliability coefficient calculated from the deviation of the local update from the reference update.

The framework was formalized through the definition of nominal FedAvg weights, reference update construction, normalized update deviation, reliability coefficient, and final reliability-corrected aggregation weights. Unlike rigid filtering mechanisms, the proposed approach applies soft down-weighting of suspicious updates. This makes it suitable for heterogeneous edge-fog and industrial IoT environments, where abnormal client behavior may be caused not only by adversarial manipulation, but also by local data drift, sensor faults, communication instability, or temporary node degradation.

A simulation model was implemented in Python using PyTorch, NumPy, and Matplotlib. The federated learning process was reproduced as a server-client simulation with ten clients and twenty-five communication rounds. A synthetic binary classification dataset was used to imitate normal and anomalous network states. To evaluate the resilience of the proposed framework, a mixed unreliable-client scenario was considered, in which part of the clients generated noisy or sign-reversed updates.

The obtained simulation results showed that the proposed framework provides more stable learning behavior compared with conventional FedAvg. Under unreliable client conditions, the proposed method demonstrated higher accuracy, lower loss, and smoother convergence of the global model. The analysis of aggregation weights and reliability coefficients confirmed that unreliable clients received substantially lower final aggregation weights than reliable clients. This indicates that the improvement is achieved not only at the level of final classification performance, but also through direct reduction of suspicious client contributions during aggregation.

The results confirm that reliability-aware aggregation can improve the resilience of federated learning in distributed critical infrastructure environments. The proposed framework preserves the privacy-oriented nature of federated learning because it does not require access to raw local data and operates

only with model updates. At the same time, it increases the robustness of the aggregation process under abnormal or adversarial client behavior.

Future research may include validation of the proposed framework on real intrusion detection and industrial IoT datasets, analysis under non-IID data distributions of different severity, comparison with other Byzantine-robust aggregation methods, and extension of the reliability coefficient by incorporating historical client behavior or reputation-based indicators.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

- [1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, B. Agüera y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, volume 54, 2017, pp. 1273–1282. URL: <https://proceedings.mlr.press/v54/mcmahan17a.html>.
- [2] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, D. Bacon, Federated learning: Strategies for improving communication efficiency, 2016. doi:10.48550/arXiv.1610.05492.
- [3] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., Advances and open problems in federated learning, Foundations and Trends in Machine Learning 14 (2021) 1–210. doi:10.1561/22000000083.
- [4] T. Li, A. K. Sahu, A. Talwalkar, V. Smith, Federated learning: Challenges, methods, and future directions, IEEE Signal Processing Magazine 37 (2020) 50–60. doi:10.1109/MSP.2020.2975749.
- [5] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, K. Seth, Practical secure aggregation for privacy-preserving machine learning, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 2017, pp. 1175–1191. doi:10.1145/3133956.3133982.
- [6] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, J. Stainer, Machine learning with adversaries: Byzantine tolerant gradient descent, in: Advances in Neural Information Processing Systems, volume 30, 2017. URL: <https://papers.nips.cc/paper/6617-machine-learning-with-adversaries-byzantine-tolerant-gradient-descent>.
- [7] D. Yin, Y. Chen, K. Ramchandran, P. Bartlett, Byzantine-robust distributed learning: Towards optimal statistical rates, in: Proceedings of the 35th International Conference on Machine Learning, volume 80, 2018, pp. 5650–5659. URL: <https://proceedings.mlr.press/v80/yin18a.html>.
- [8] E. M. El Mhamdi, R. Guerraoui, S. Rouault, The hidden vulnerability of distributed learning in byzantium, in: Proceedings of the 35th International Conference on Machine Learning, volume 80, 2018, pp. 3521–3530. URL: <https://proceedings.mlr.press/v80/mhamdi18a.html>.
- [9] V. Tolpegin, S. Truex, M. E. Gursoy, L. Liu, Data poisoning attacks against federated learning systems, in: Computer Security – ESORICS 2020, 2020, pp. 480–501. doi:10.1007/978-3-030-58951-6_24.
- [10] M. Fang, X. Cao, J. Jia, N. Z. Gong, Local model poisoning attacks to byzantine-robust federated learning, in: Proceedings of the 29th USENIX Security Symposium, 2020, pp. 1605–1622. URL: <https://www.usenix.org/conference/usenixsecurity20/presentation/fang>.
- [11] X. Cao, M. Fang, J. Liu, N. Z. Gong, Fltrust: Byzantine-robust federated learning via trust bootstrapping, in: Proceedings of the Network and Distributed System Security Symposium, 2021. doi:10.14722/ndss.2021.24434.
- [12] S. Kudrenko, O. Nimych, I. Makieiev, Method for predicting node compromise in edge and fog environments for critical infrastructure, Ukrainian Information Security Research Journal 27 (2025) 87–93. doi:10.18372/2410-7840.27.21183.