network

MDPI

Article

TAFL-UWSN: A Trust-Aware Federated Learning Framework for Securing Underwater Sensor Networks

Raja Waseem Anwar [1,*], Mohammad Abrar [2], Abdu Salam [3] and Faizan Ullah [4]

[1] Department of Computer Science, German University of Technology in Oman, P.O. Box 1816, Muscat PC 130, Oman
[2] Faculty of Computer Studies, Arab Open University, P.O. Box 1596, Muscat 122, Oman; abrar.m@aou.edu.om
[3] Department of Computer Science, Abdul Wali Khan University, Mardan 23200, Pakistan; abdusalam@awkum.edu.pk
[4] Institute of Neuroscience and Medicine 4 (INM-4), Forschungszentrum Julich, 52428 Jülich, Germany; f.ullah@fz-juelich.de
* Correspondence: raja.anwar@gutech.edu.om

Abstract

Underwater Acoustic Sensor Networks (UASNs) are pivotal for environmental monitoring, surveillance, and marine data collection. However, their open and largely unattended operational settings, constrained communication capabilities, limited energy resources, and susceptibility to insider attacks make it difficult to achieve safe, secure, and efficient collaborative learning. Federated learning (FL) offers a privacy-preserving method for decentralized model training but is inherently vulnerable to Byzantine threats and malicious participants. This paper proposes trust-aware FL for underwater sensor networks (TAFL-UWSN), a trust-aware FL framework designed to improve security, reliability, and energy efficiency in UASNs by incorporating trust evaluation directly into the FL process. The goal is to mitigate the impact of adversarial nodes while maintaining model performance in low-resource underwater environments. TAFL-UWSN integrates continuous trust scoring based on packet forwarding reliability, sensing consistency, and model deviation. Trust scores are used to weight or filter model updates both at the node level and the edge layer, where Autonomous Underwater Vehicles (AUVs) act as mobile aggregators. A trust-aware federated averaging algorithm is implemented, and extensive simulations are conducted in a custom Python-based environment, comparing TAFL-UWSN to standard FedAvg and Byzantine-resilient FL approaches under various attack conditions. TAFL-UWSN achieved a model accuracy exceeding 92% with up to 30% malicious nodes while maintaining a false positive rate below 5.5%. Communication overhead was reduced by 28%, and energy usage per node dropped by 33% compared to baseline methods. The TAFL-UWSN framework demonstrates that integrating trust into FL enables secure, efficient, and resilient underwater intelligence, validating its potential for broader application in distributed, resource-constrained environments.

Keywords: Underwater Acoustic Sensor Networks; federated learning; trust management; Zero-Trust Architecture; Byzantine attacks; edge aggregation; model security

Academic Editor: Patrick Seeling

Received: 21 February 2026
Revised: 12 March 2026
Accepted: 17 March 2026
Published: 19 March 2026

1. Introduction

The development of Underwater Acoustic Sensor Networks (UASNs) has been identified as a helpful asset in the fields of oceanographic data gathering, eco-environmental monitoring, and underwater surveillance [1]. Such networks consist of spatially dispersed

underwater sensor nodes that mainly use acoustic communication, since radio waves cannot be used in water. Also, these networks, comprising multifunctional sensors capable of data collection, storage, and transmission, stem from the drive to enhance the reliability and security of underwater data collection. Regardless of that promise, UASNs are exposed to myriad challenges posed by the operational environment: low bandwidth, high propagation delay (approximately 1500 m/s underwater), high bit error rates, and extreme energy constraints (recharging or replacing batteries may be impractical) [2,3]. The results of these factors are low data rates, high latency, and reliability issues as compared to terrestrial wireless networks. Other than the physical limitations, another crucial concern in UASNs is security. These networks are particularly vulnerable to malicious nodes and insider attacks since they are often unmonitored and in out-of-the-way places [4]. Typical examples of threats to network performance and mission-critical analytics include black-hole attacks (improper relaying of forwarded packets), Sybil attacks (impersonation of multiple identities), and data poisoning (transmission of falsified sensor measurements or malicious model updates). Recent developments in federated learning (FL) present an opportunity to conduct distributed machine learning within UASNs without centralizing raw data, thereby consuming less bandwidth and not compromising privacy [5]. Under FL, each of the nodes learns its model using its data and transfers the updates to a central server, which then combines the updates to create a global model (e.g., FedAvg [5]). Nonetheless, standard FL only considers honest participation and is easily prone to Byzantine attacks, where some malicious nodes can contaminate the global model by sending wrong or tampered updates [6]. In mission-critical underwater settings, in particular, such weaknesses are particularly dangerous because the wrong model results may have devastating consequences.

Motivated by the above, we propose TAFL-UWSN. This novel framework tightly integrates trust evaluation with the FL process to secure UASNs. Edge aggregators, AUVs, enforce hierarchical trust filtering, excluding low-trust nodes before model updates reach the cloud. This follows the zero-trust principle, "never trust, always verify," by dynamically validating each node's contribution to the network, a strategy aligned with emerging 6G security paradigms [7]. Federated learning can be used in underwater sensor networks for application-specific tasks rather than general model training, including anomaly detection in acoustic communications, malicious node detection, secure environmental monitoring, and trust-aware intrusion detection. In the current work, we focus on a security-related task: identifying unreliable or malicious behavior in a distributed underwater environment, as it is a major concern for the stability of underwater sensing activities. The specific objectives of this research are:

- To design a novel FL algorithm that integrates behavior-driven trust scores into the model aggregation process, enhancing the system's resilience against data poisoning, Sybil attacks, and other adversarial behaviors common in UASNs.
- To implement a trust-based pre-filtering mechanism at the edge layer, specifically within AUVs, that identifies and excludes low-trust updates before transmission to the cloud, thereby reducing communication costs and mitigating the impact of malicious nodes early in the learning cycle.
- To evaluate the proposed TAFL-UWSN framework through comprehensive simulations in Aqua-Sim/NS2, assessing performance under varying attack intensities and network scales in terms of model accuracy, false positive rates, communication overhead, and energy consumption.
- To operationalize zero-trust security principles within an FL architecture, demonstrating how continuous verification of node trustworthiness can support adaptive and

lightweight defense strategies in decentralized and bandwidth-constrained underwater environments.

This paper presents the first framework that unifies trust management and FL in underwater environments. The main contributions are as follows:

- We designed a novel FL algorithm that incorporates node-level trust scores into model aggregation, improving robustness against poisoning and Sybil attacks.
- We introduce trust-based pre-filtering at mobile AUV aggregators, reducing communication overhead and mitigating collusion by filtering malicious updates early.
- Through extensive simulations using Aqua-Sim/NS2, we demonstrate that TAFL-UWSN significantly outperforms conventional FL and security baselines in model accuracy, attack resilience, and energy efficiency.
- The proposed framework operationalizes zero-trust principles in decentralized, bandwidth-limited UASNs, offering a lightweight and adaptive defense mechanism suitable for real-world underwater deployments.

The rest of the paper is structured as follows. Section 2 reviews related work in FL, trust management, and UASN security. Section 3 describes the system model, including the network architecture and threat assumptions. Section 4 details the proposed TAFL-UWSN framework, including the design of the trust metric and trust-aware aggregation. Section 5 presents the experimental results and performance evaluation. Section 6 addresses limitations and future work and concludes the paper.

2. Literature Review

2.1. Trust Models in Underwater Sensor Networks

Trust and reputation have been applied in wireless sensor networks (WSNs) to identify misbehavior and secure routing for a long time. Earlier designs, such as the reputation-based system proposed by Ganeriwal et al. [8] and the Ariadne and WATCHERS monitors, calculated trust scores by monitoring packet forwarding from neighbors. They used those scores to isolate unreliable nodes. Sun et al. [9] addressed the problem of statistical and Bayesian methods to assess trust in ad hoc networks and how trust can be used to increase network throughput under insider attacks. Such classic WSN trust models, however, tend to presuppose relatively stable land-based networks and direct perception of neighbors. Researchers have adapted trust mechanisms to sensor networks in the water in recent years. Hosseinzadeh et al. [10] introduced a Q-learning-based trust model (QLTM) in UASNs, where each node employs the reinforcement learning scheme to update trust levels with the neighbors according to their delivery experiences. Their model achieved a 90% detection rate for misbehaving nodes in simulation, but it operates at the routing level and does not integrate with data fusion or learning. The model ITrust was proposed by Du et al. [11] to detect outlier behavior in UASNs using an isolation forest. Data-driven trust assessment revealed that ITrust had high detection accuracy (96%) with a very low false positive rate (~2%), showing the efficacy of data-driven trust assessment. Wang et al. [12] demonstrated the effectiveness of data-driven trust assessment. Wu et al. [13] proposed an autonomous online study technique that creates customized tutorials by integrating generative learning, cognitive computing, and a multimodal understanding graph and demonstrated the ability of adaptive learning mechanisms to enhance learning efficiency in evolving settings. For instance, Zhu et al. [14] proposed an energy-conscious DRL-based dual-perception fountain coding for resource-constrained UASNs and showed that smart communication design could enhance decoding performance, reduce broadcast overhead, and lower energy consumption in underwater networks.

2.2. Federated Learning in Adversarial Environments

The concept of FL was initially popularized by McMahan et al. [5], and since then, several works have been done concerning its vulnerabilities. The study of Byzantine-robust FL has suggested aggregation rules such as the coordinate-wise median or Krum, which is robust to a fraction of malicious clients [6]. As demonstrated by Yin et al. [6], Kang et al. [15], with the help of robust statistics (geometric median), the global model can withstand outliers to a certain degree, in many cases at the expense of slower convergence. These methods, however, presuppose a predetermined value of the attack fraction and do not actively detect what nodes are malicious; they blindly aggregate robustly. Conversely, a trust-based model establishes low-quality contributors. Recent developments incorporate the concepts of trust in FL. Wahab et al. [16] proposed a trust-based FL scheme for recommendation systems that weights client updates according to a credibility score, which solves problems such as cold-start clients. Such works suggest that FL may become more robust with trust weighting. This idea is extended to security-critical UASNs in our TAFL-UWSN, where the network-layer and data consistency behavior become the sources of trust. There have been some works that incorporate FL and security in underwater or IoT networks. Yan et al. [17] applied FL to an underwater IoT environment and incorporated a multi-armed bandit client scheduler and a voting mechanism to rule out potentially malicious updates. This enhanced performance and, to some extent, the security of FL, yet their protocol does not keep a continuous score of trust for each node continuously; their protocol makes decisions in binary format per round. TAFL-UWSN, in contrast, keeps and refreshes trust levels at all times, slowly adjusting to the evolution of the node's behavior. Pokhrel et al. [18] investigated the zero-trust methodology of blockchain in a vehicular network FL scenario, which, in addition to improving the security, has exceptionally high latency and communication costs (related to the overhead of a consensus). TAFL-UWSN does not use costly consensus; it is lighter-weight and adapted to resource-limited UASNs.

2.3. Zero-Trust Architectures and 6G Security

Zero-Trust Architecture (ZTA) is a security model in which no device, user, or application is ever trusted, not even those within the network perimeter [7]. Each access is constantly verified and permitted. In response to the growing number of insider threats, ZTA is becoming more popular in 5G/6G and IoT networks [19,20]. In distributed networks such as UASNs, ZTA is very difficult to operate due to the lack of central authority and connectivity discontinuity. Previous efforts have begun to think about decentralized zero trust as applied to networks: Fu et al. [20] propose an intelligent policy orchestration of ZTA in 6G systems, with a focus on dynamic access control. Liu et al. [21] and others have proposed blockchain as a means of realizing zero trust in the IoT, basically having distributed ledgers to authenticate each transaction or update. These methods, however, tend to presuppose powerful edge nodes and stable connections (to perform cryptographic operations and make frequent authentication), which is not true in UASNs. Although it is not a complete implementation of ZTA, our TAFL-UWSN shares the same spirit of zero trust since it never trusts the data of any node until it has been checked using trust metrics. The trust score gives weight to each node's contribution to the global model, and it can be rejected if the trust is insufficient. It gives rise to a tacit continuous authentication system: a node demonstrates its integrity by acting reliably (sending packets, supplying reliable data) to maintain its power in the learning process. As opposed to classical ZTA, which may employ heavyweight authentication per packet [7], our approach is lightweight and dynamic, and it applies to the underwater bandwidth-constrained, high-latency environment.

Table 1 compares these representative trust models with our proposed TAFL-UWSN. A common gap is that existing models focus on secure routing or neighbor behavior but do

not integrate with collaborative data processing or learning. They operate separately from any machine learning tasks.

Table 1. Comparison of existing trust- and FL-based security models for UASNs.

Model (Year)	Key Technique	Federated Learning?	Detected Attacks	Accuracy (%)	Overhead
QLTM [10]	Q-learning trust adaptation per node based on packet forwarding rewards	No—routing only	Packet drop (blackhole)	~90% (malicious detect)	Low (lightweight)
ITrust [11]	Isolation Forest anomaly detector for trust values	No—standalone trust	Data integrity attacks	96% (true detect)	Moderate (global computation)
DRL-Trust [12]	Deep RL + Random Forest for dynamic trust scoring	No—standalone trust	Various (blackhole, tampering)	99% (reported)	High (computational)
FL UCB-SC [15]	FL with MAB-based client scheduling and voting mechanism	Yes (FL without trust)	Random client drops (DoS)	~92% (model accuracy)	Moderate (voting communication)
Blockchain-FL [18]	Blockchain-aided FL with anomaly detection at the aggregator (zero trust)	Yes (FL with blockchain)	Sybil, model poisoning	95% (model accuracy)	High (>50% overhead)

3. System Model and Threat Assumptions

3.1. Network Topology

We consider a typical UASN consisting of a set of underwater sensor nodes deployed to monitor an area (e.g., an ocean monitoring field). These sensor nodes are equipped with acoustic modems for communication. An AUV or a surface buoy acts as an edge aggregator, moving within the network to collect data/model updates from sensors. Collected information is forwarded to a surface station or cloud server for higher-level processing (and acts as a central FL server).

Underwater sensor nodes perform local sensing and learning; an AUV (edge aggregator) travels and collects model updates, filters out low-trust nodes, and sends aggregated results to a cloud server. The cloud broadcasts the global model back to the nodes. Acoustic links (blue dashed) connect sensors to the AUV; a higher-bandwidth link (e.g., radio) connects the AUV to the cloud.

3.1.1. Node Capabilities

Each sensor node is equipped with sensing hardware (e.g., hydrophones, chemical sensors) and features limited computing power (microcontroller or DSP) and energy (battery). Nodes can perform simple local training updates on their data (e.g., a few epochs of stochastic gradient descent on a small neural network). They also monitor neighbors' communication behavior to compute trust (Section 4). The AUV has greater resources and can perform model aggregation and maintain a global trust table. The cloud server has ample compute to finalize model training and could run more complex analytics on the aggregated data if needed.

3.1.2. Communication Characteristics

Underwater sensors communicate via acoustic channels with limited bandwidth (several kbps) and high latency (order of 0.1–1 s per km) [2,3]. We assume a TDMA or CSMA-based MAC, where nodes take turns transmitting to avoid collisions (typical in

Aqua-Sim simulations [22]). The communication range is limited (perhaps a few hundred meters), so multi-hop forwarding may occur among sensors as they approach the AUV. The AUV can directly come within the range of each sensor at times, effectively reducing the need for long multi-hop chains. The surface link from the AUV to the cloud is assumed to be high-speed (e.g., via Wi-Fi or satellite when the AUV surfaces periodically).

3.1.3. Trust Observations

We assume that sensors can observe certain behaviors of one-hop neighbors necessary for trust calculation, e.g., if A sends a packet to B and B should forward to C, A can listen to see if B indeed forwards (overhearing on the acoustic channel) [8]. Such overhearing is error-prone underwater but provides probabilistic evidence. Alternatively, the AUV can request periodic reports from nodes about their neighbors' actions to cross-verify (with the risk of false reporting by colluders, which we mitigate by incorporating trust fusion from multiple sources). We assume that cryptographic link-layer authentication is in place, so nodes cannot easily spoof others' IDs (except for Sybil nodes, which obtain multiple legitimate IDs).

3.2. Threat Model

We consider insider attackers—legitimate nodes that have been compromised and behave maliciously. The primary attacker types are summarized in Table 2 with their behaviors and impacts. We assume that attackers do not break cryptography (they cannot create fake identities without limit, except Sybil, which is an assumed capability in that case). However, they may arbitrarily transmit spurious data at their discretion.

- Blackhole attack: A malicious sensor drops all packets it is supposed to forward for others or refuses to report its sensing data, effectively creating a data blackhole. This disrupts routing and causes data loss. In the FL context, a blackhole node may simply not participate or drop model update messages of neighbors (if acting as a relay). We assume that our trust mechanism will catch this due to poor packet forwarding records [10].
- Sybil attack: A single physical node assumes multiple digital identities (either by impersonating other addresses or by obtaining multiple valid IDs). This node can then join the FL process or network routing in various roles, attempting to exert disproportionate influence. Sybil attackers can severely poison collaborative algorithms by acting as several colluding nodes [6]. We assume that an attacker can create a limited number of Sybil identities (not an unlimited number) and that the initial trust for new identities is neutral—the attacker must build trust in each one before causing damage.
- Data poisoning (Byzantine attack): A malicious node submits false data to disrupt decision-making. In sensor terms, it may falsify readings (e.g., spoofing an environmental reading). In FL terms, it computes an incorrect model update (e.g., manipulating gradients) to corrupt the global model's accuracy [6]. Attackers might still follow the protocol (sending updates on time) but craft the content maliciously. We assume that they have some knowledge of the learning task (e.g., targeting a specific classification outcome).
- Denial-of-Service (DoS): Attackers can also jam acoustic channels or flood the network with junk data to waste energy and bandwidth. Jamming is hard to defend against but can often be detected by unusual interference patterns. In our context, a DoS attacker might attempt to disrupt the FL rounds by causing communication failures. We do not explicitly model jamming in our simulation; however, we account for its effect by incorporating random link failures. Trust can indirectly capture persistent disruptive behavior (if a node's presence correlates with failed communications).

Table 2. Attacker types and behaviors in UASNs.

Attack Type	Malicious Behavior (Node Actions)	Impact on Network/FL Process
Blackhole	Drops or refuses to forward packets from neighboring nodes and may withhold its own sensing data.	Causes data loss and routing disruption. In FL, updates from affected regions may be missing, and the node's trust score decreases due to non-cooperative behavior.
Sybil	Uses multiple fake identities and participates in routing or FL under several node IDs.	Gains excessive influence in the FL process, increases the risk of model poisoning, and reduces trust because of identity inconsistency.
Data/Model Poisoning	Sends falsified sensing data or deliberately manipulates local training to generate harmful model updates.	Corrupts the global model, reduces accuracy, and may trigger false alarms. Trust decreases due to abnormal data or model deviation.
Denial-of-Service (DoS)	Jams the acoustic communication channels or floods the network with unnecessary messages.	Interrupts FL communication, increases energy consumption, and delays convergence. Trust decreases because of repeated interference.

3.3. Assumptions

We assume that the majority of nodes are honest (the number of attackers is less than half of the network in our analysis); however, our method can handle higher percentages with graceful degradation. The cloud server and AUV are assumed to be secure/trusted (they may be high-value assets, but they are typically harder to compromise and can be physically secured). Attackers do not have out-of-band capabilities, such as destroying hardware, or battery attacks are performed in-network via the defined behaviors. We also assume that, for Sybil, the network admission control allows for at most a moderate number of Sybil identities; completely preventing the introduction of Sybil identities is outside the scope, but trust will handle them once they act maliciously.

The various insider threats considered in our threat model are summarized in Table 2, which outlines the main attacker types in UASNs, their malicious behaviors, and their specific impacts on both network operations and the FL process.

These attack behaviors are defined so that our trust mechanism is based on them, and these metrics are monitored based on them (e.g., the blackhole packet forwarding rate, identity consistency in Sybil, and model update validity in poisoning). We take the trust assessment period to be less than the attack duration, so that attackers will ultimately be detected (e.g., an attacker is malicious long enough for trust to decrease substantially). Having the system and threat models in place, we now describe our TAFL-UWSN framework that, based on this network architecture, makes use of the above threats by proposing an implementation that uses a combination of trust and FL to counter them.

Figure 1 presents a unified workflow of the proposed TAFL-UWSN framework, showing federated learning, trust evaluation, edge-layer filtering, and malicious node handling in the underwater sensor network.

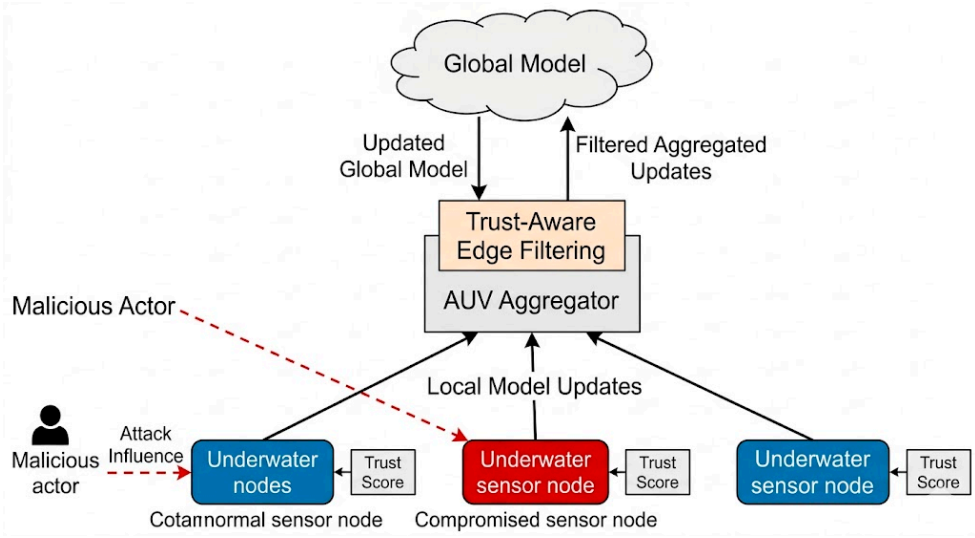

Figure 1. Unified TAFL-UWSN workflow in an underwater sensor network.

4. Methodology

4.1. Problem Formulation and Learning Objective

We consider a binary intrusion/anomaly detection problem formulated within a federated learning (FL) paradigm. Let the set of participating clients be denoted by $\mathcal{C} = \{1, 2, \ldots, N\}$, where each client represents an autonomous sensing or monitoring entity with locally stored data that cannot be shared directly. Each client $i \in \mathcal{C}$ owns a private dataset D_i defined as:

$$\mathcal{D}_i = \{(x_{i,j}, y_{i,j})\}_{j=1}^{|\mathcal{D}_i|} \tag{1}$$

where $x_{i,j} \in \mathbb{R}^d$ denotes a feature vector extracted from network traffic flows, and $y_{i,j} \in \{0, 1\}$ is the corresponding binary label, with $y_{i,j} = 0$ indicating normal traffic and $y_{i,j} = 1$ indicating malicious or intrusive activity. The global dataset is distributed across clients in a non-IID manner, i.e.,

$$\mathcal{D} = \bigcup_{i=1}^{N} \mathcal{D}_i, \mathcal{D}_i \sim /\mathcal{D}_k \text{ for } i \neq k \tag{2}$$

reflecting heterogeneous traffic patterns and attack distributions across decentralized nodes.

For the learning objective, let $f(x; \mathrm{w})$ denote a global detection model parameterized by weights w. The learning objective is to minimize the global empirical risk across all clients without centralizing raw data:

$$\min_{\mathbf{w}} \mathcal{L}(\mathrm{w}) = \sum_{i=1}^{N} \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \mathcal{L}_i(\mathrm{w}) \tag{3}$$

where the local loss at client i is defined as:

$$\mathcal{L}_i(\mathrm{w}) = \frac{1}{|\mathcal{D}_i|} \sum_{(x,y) \in \mathcal{D}_i} \ell(f(x; \mathrm{w}), y) \tag{4}$$

For binary intrusion detection, the loss function $\ell(\cdot)$ is chosen as the binary cross-entropy loss:

$$\ell(\hat{y}, y) = -[y \log(\sigma(\hat{y})) + (1 - y) \log(1 - \sigma(\hat{y}))] \tag{5}$$

where $\sigma(\cdot)$ denotes the sigmoid activation function.

The federated optimization is an iterative process that runs through communication rounds. During such a round, the chosen clients update their models with stochastic gradient descent (SGD), and the server combines the updates to receive a better global model. In contrast with generic federated learning systems, the proposed TAFL-UWSN

can be used when the acoustic communication of sensor networks is characterized by high propagation delays, low bandwidth, unreliable links, and high packet loss rates. These attributes of underwater channels directly influence the credibility and timeliness of local model updates. Thus, the suggested framework will integrate trust-based assessment and filtering to reduce the effect of unreliable or suspicious updates before global integration. By doing so, underwater transmission conditions are considered a significant aspect of secure and robust federated learning.

4.2. Dataset Description and Feature Engineering

4.2.1. UNSW-NB15 Dataset

The experiments are carried out on the UNSW-NB15 intrusion detection dataset [23], a popular benchmark for evaluating network security and anomaly detection models. The data was created from real modern network traffic with synthetic attack behaviors to ensure diversity in both benign and malicious behaviors. A network flow can be associated with each record in the dataset and is defined by a set of numerical and categorical attributes, including statistics, protocol behavior, time, and content characteristics of packets. The dataset contains traffic from various attack categories (i.e., DoS, exploits, reconnaissance) and is combined into a binary classification task in the given research.

4.2.2. Target Label Definition

Let y denote the ground-truth label for a traffic flow. The original dataset provides both multi-class attack categories and a binary indicator. In this work, we use the binary formulation:

$$y = \begin{cases} 0, & \text{normal (benign) traffic} \\ 1, & \text{attack (malicious) traffic} \end{cases} \tag{6}$$

This formulation aligns with anomaly detection and intrusion detection use cases and simplifies the evaluation of robustness under adversarial learning scenarios.

4.2.3. Feature Exclusion and Selection

To prevent information leakage and ensure realistic deployment assumptions, certain attributes are excluded from the learning process identifier features (e.g., flow IDs) and explicit attack category labels (used only for analysis, not training).

Let the original feature vector be:

$$\tilde{x} \in \mathbb{R}^{d_{\text{raw}}} \tag{7}$$

After exclusion and preprocessing, each sample is represented as:

$$\tilde{x} \in \mathbb{R}^{d}, \quad \text{where } d < d_{\text{raw}} \tag{8}$$

The last feature group comprises continuous numerical features (e.g., number of packets, time, count of bytes) and discrete features (e.g., protocol type, service), coded using one-hot representation. The feature engineering is used with all clients to provide a homogeneous global model representation.

4.3. Data Preprocessing Pipeline

Before federated training, every data sample is subjected to a single common preprocessing pipeline so that numerical consistency is achieved, inconsistencies across clients are removed, and a homogeneous feature representation is obtained that can be trained in a neural network.

Let $x \in \mathbb{R}^{d_{\text{raw}}}$ denote an original feature vector extracted from the UNSW-NB15 dataset. The feature space is made up of numerical and categorical qualities. Missing values on the numerical features are processed through median imputation, where every missing value is replaced with the median of the values of the training set of that particular feature. This plan is resistant to outliers and maintains the statistical properties of skewed distributions. After imputation, the numerical features are standardized by z-score, which is defined as:

$$\tilde{x}_k = \frac{x_k - \mu_k}{\sigma_k} \tag{9}$$

where μ_k and σ_k denote the mean and standard deviation of feature k, respectively. This normalization ensures that all numerical features contribute proportionally during gradient-based optimization.

Categorical features are first imputed with the most frequent category for each attribute, then transformed via one-hot encoding. Let $\phi(\cdot)$ denote the encoding operator that maps a categorical value to a binary vector representation. After encoding, categorical features expand into a higher-dimensional sparse vector. The final preprocessed feature vector for each data sample is given by

$$\tilde{x} = [\tilde{x}^{(n)}, \tilde{x}^{(c)}] \in \mathbb{R}^d \tag{10}$$

where $\tilde{x}^{(n)}$ and $\tilde{x}^{(c)}$ denote the normalized numerical features and encoded categorical features, respectively, and d represents the final feature dimensionality after preprocessing.

4.4. Federated Learning Configuration

4.4.1. Client Population and Data Distribution

The federated learning system consists of $N = 50$ participating clients, each holding a private subset of the global dataset. No raw data is exchanged between clients or with the central aggregator during training. To simulate realistic data heterogeneity commonly observed in decentralized environments, the dataset is distributed across clients using a Dirichlet-based non-IID partitioning scheme. For each class c, a client allocation vector is sampled from a Dirichlet distribution,

$$\pi_c \sim \text{Dirichlet}(\alpha 1) \tag{11}$$

where α controls the degree of statistical heterogeneity. Each sample that belongs to class c is assigned to client i with probability $\pi_{c,i}$. Smaller values of α produce more skewed and heterogeneous class distributions. In this study, $\alpha = 0.3$ is selected to reflect strong non-IID conditions.

4.4.2. Local Model Architecture and Training

Each client trains a local neural network classifier initialized from the global model parameters received at the beginning of each federated round. The model is a lightweight multilayer perceptron designed for binary intrusion detection. Formally, the model is expressed as

$$f(x; \mathrm{w}) = \sigma(\mathrm{W}_2\, \mathrm{ReLU}(\mathrm{W}_1 x + b_1) + b_2) \tag{12}$$

where $\mathrm{w} = \{\mathrm{W}_1, \mathrm{W}_2, b_1, b_2\}$ denotes the set of learnable parameters and $\sigma(\cdot)$ is the sigmoid activation function.

Local training minimizes the binary cross-entropy loss using the AdamW optimizer. Each client performs one local training epoch per round with a batch size of 256, a learning rate of 10^{-3}, and weight decay of 10^{-5}. These hyperparameters are selected to balance convergence speed and computational efficiency.

4.5. Adversary and Attack Model

4.5.1. Malicious Client Selection Strategy

To evaluate robustness against insider threats, a fraction $\rho \in (0,1)$ of clients are designated as malicious. Let $\mathcal{C}_{\text{mal}} \subset \mathcal{C}$ denote the malicious client set, where

$$| \mathcal{C}_{\text{mal}} | = \lfloor \rho N \rfloor \tag{13}$$

Malicious clients are selected uniformly at random at the beginning of each experiment and retain adversarial behavior throughout all federated rounds.

4.5.2. Poisoning Attack Implementations

Two poisoning strategies are implemented to simulate adversarial behavior. In the label-flipping attack, malicious clients invert their local labels during training such that $y \leftarrow 1 - y$ thereby corrupting the local decision boundary. In the scaled model update attack, malicious clients amplify their model updates before submission by applying

$$\Delta \mathrm{w}_i \leftarrow \gamma \Delta \mathrm{w}_i \tag{14}$$

where $\gamma > 1$ controls the attack strength. This manipulation increases the influence of malicious updates during aggregation.

4.6. Trust Modeling Framework

The proposed TAFL-UWSN framework will assign a dynamic trust score to each client node based on its trustworthiness in the federated learning. The trust score is determined by several indicators, including behavioral consistency, deviations between model updates, consistency in communication, and quality of past participation. The higher the trust score, the more likely the node is to deliver good local updates; the lower the score, the more questionable, fluctuating, or malignant.

4.6.1. Trust Signal Components

Each client i is associated with a trust score $\tau_i^t \in [0,1]$ at federated round t. Trust is computed using multiple behavioral and learning-based signals. The first signal is the local validation accuracy, denoted s_i^{acc}, which reflects the predictive reliability of the local model. The second signal measures model update deviation, defined as

$$s_i^{dev} = 1 - \frac{\| \Delta \mathrm{w}_i - \mathrm{median}(\Delta \mathrm{w}) \|}{\max_j \| \Delta \mathrm{w}_j \|} \tag{15}$$

which penalizes anomalous updates deviating significantly from the majority. The third signal is a behavioral reliability score, $s_i^{beh} \in [0,1]$, representing long-term reliability characteristics.

4.6.2. Trust Update and Temporal Smoothing

The instantaneous trust signal for client i at round t is computed as

$$s_i^t = \sum_k \lambda_k s_{i,k}^t, \sum_k \lambda_k = 1 \tag{16}$$

where λ_k are predefined weights. Trust is updated using exponential smoothing:

$$\tau_i^t = \gamma \tau_i^{t-1} + (1-\gamma) s_i^t \tag{17}$$

with initial trust $\tau_i^0 = 0.5$. The trust evolution process across rounds is illustrated in Figure 2.

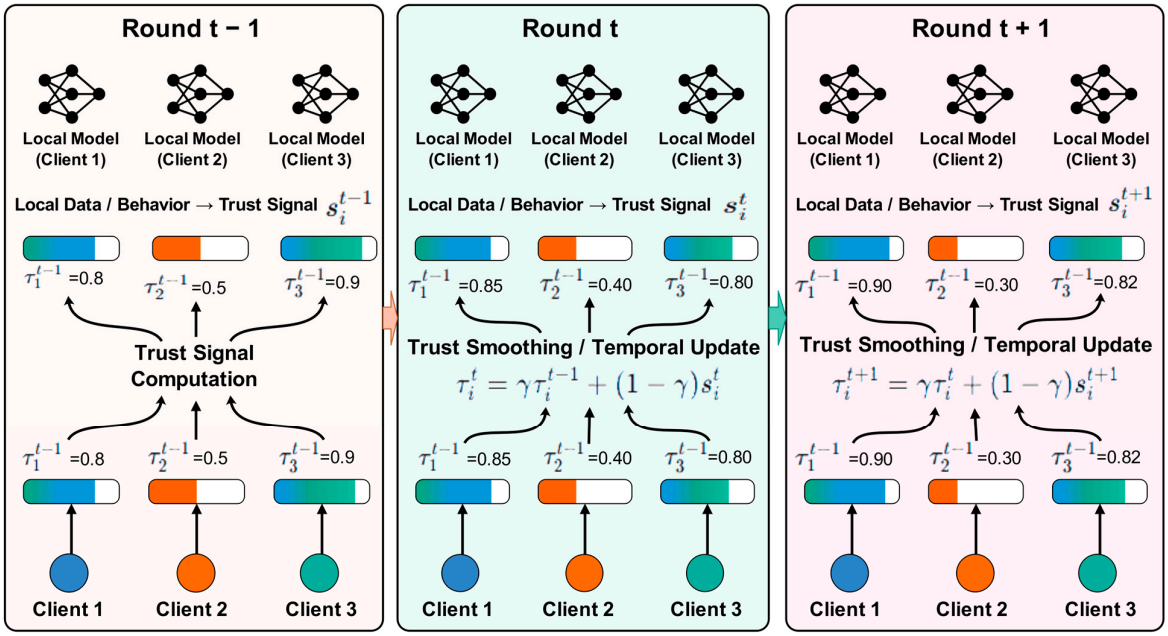

Figure 2. Trust update process across federated rounds.

4.7. Trust-Aware Client Filtering and Aggregation

4.7.1. Trust Thresholding and Client Acceptance

At each round, a client is accepted for aggregation if its trust score satisfies $\tau_i^t \geq \tau_{\min}$. To prevent training collapse when too few clients meet the threshold, a fallback mechanism is applied: all selected clients are temporarily accepted if fewer than two pass the trust filter. The filtering decision process is illustrated in Figure 3.

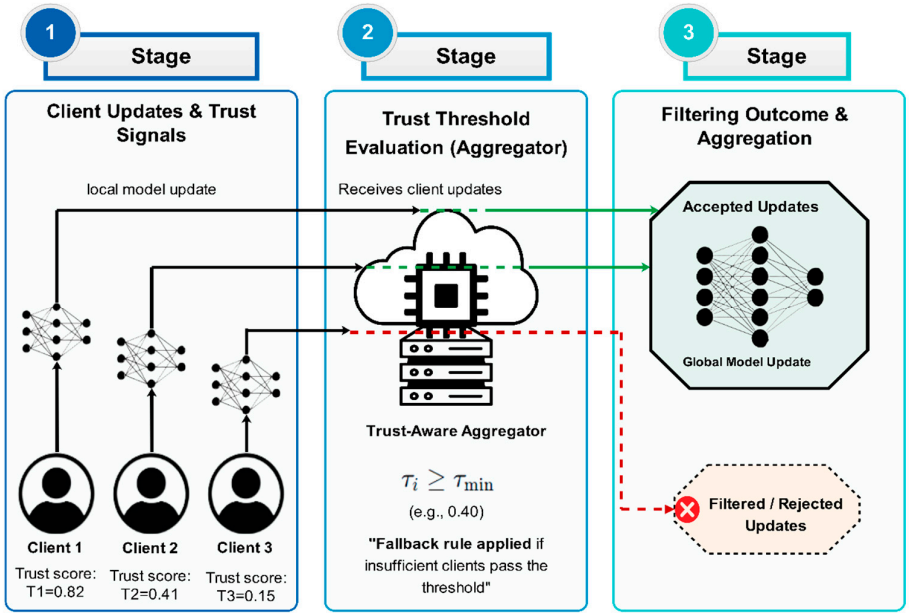

Figure 3. Client filtering decision based on trust threshold.

4.7.2. Trust-Weighted Federated Aggregation

For accepted clients, the global model update is computed using trust-weighted federated averaging:

$$\mathrm{w}^{t+1} = \mathrm{w}^t + \sum_{i \in \mathcal{C}_{acc}} \frac{n_i \tau_i^t}{\sum_{j \in \mathcal{C}_{acc}} n_j \tau_j^t} \Delta \mathrm{w}_i \tag{18}$$

where n_i denotes the local dataset size. When all trust values are equal, this formulation reduces to standard FedAvg. Trust is not applied separately as a decision and other criteria; rather, it is applied together with the quality of the local model update on the aggregation process. By doing so, the proposed technique trades off between security and learning performance by giving more weight to high-trust and useful updates for model convergence and less weight to abnormal or low-trust updates. The approach helps maintain overall model accuracy while restricting the influence of potentially malicious clients.

4.8. Baseline Aggregation Methods

Two baseline aggregation strategies are considered for comparison. The first is standard FedAvg, which aggregates client updates weighted only by local dataset sizes. The second is a Byzantine-robust coordinate-wise trimmed mean, which removes extreme update values before averaging to mitigate outliers without explicit client identification.

4.9. Evaluation Metrics and Experimental Protocol

The accuracy of classification and the area under the ROC curve (AUC) are measures of model performance. In addition, the protocol documents the client participation rate, which is the number of accepted clients per round divided by the number of selected clients, and it monitors the development of trust. Every metric is calculated at every federated round and averaged over experimental runs so as to ensure statistical reliability.

5. Results and Performance Evaluation

5.1. Experimental Setup and Parameter Settings

The experimental analysis is performed on an FL system with a predetermined number of communication rounds. By default, the global model is trained on 100 federated rounds, with each round having a subset of clients conducting local training and sending updates on the model to the aggregator. Experiments are repeated with different numbers of malicious clients to determine the robustness to adversarial behavior. The malicious client ratio ρ is systematically swept over the range $\rho \in \{0.0, 0.1, 0.2, 0.3, 0.4\}$, enabling evaluation under increasingly hostile conditions. In either case, the malicious clients will be fixed during the training and implement the specified poisoning scheme. Each experiment is then repeated over different random seeds to reduce the effect of randomness, and the statistical reliability is ensured by controlling data partitioning, client selection, and model initialization. The reported results are the average performance of the method used to aggregate over runs, and they provide a consistent, repeatable basis for comparing aggregation techniques under the same conditions.

5.2. Global Model Performance Under Adversarial Conditions

5.2.1. Accuracy Comparison

In this section, the strength of the global model in the face of adversarial participation is assessed, and FedAvg, the trimmed mean, and the proposed TAFL aggregation strategy are compared. The comparison will be based on the classification accuracy as the key performance indicator with the increase in the proportion of malicious clients.

Figure 4 shows how the accuracy of the test changes according to communication rounds in a typical hostile environment with 30% of malicious clients. As depicted, FedAvg with standard FedAvg suffers a rapid, sustained loss of accuracy due to an unfiltered combination of poisoned updates. The trimmed mean baseline is more robust and does not allow for extreme updates, but it converges more slowly and attains lower accuracy. On the other hand, TAFL also keeps a higher and more stable accuracy during training.

This is because trust-based client filtering and trust-weighted aggregation both reduce the presence of unreliable or malicious participants.

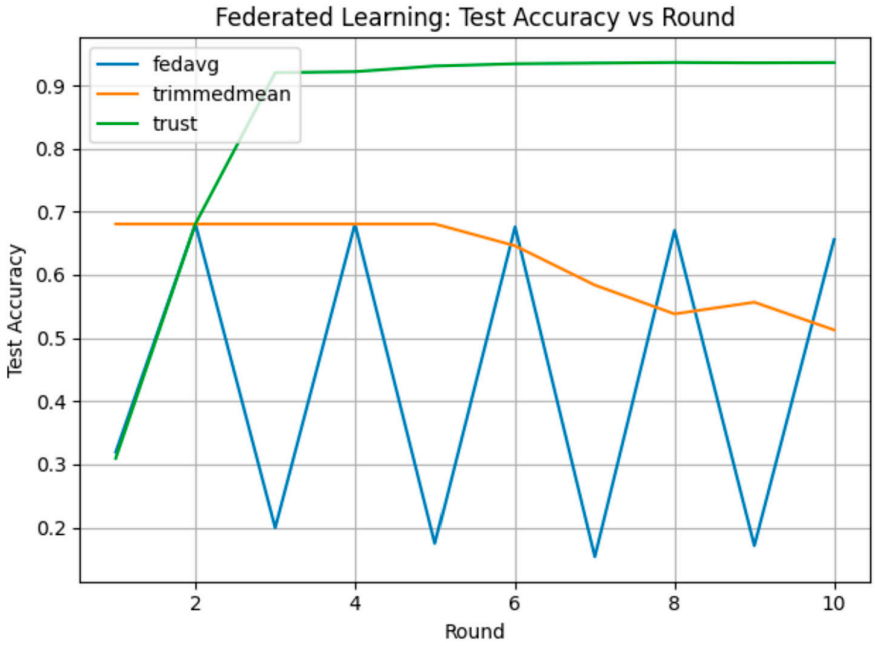

Figure 4. Test accuracy versus communication rounds with 30% malicious clients.

In order to measure the level of robustness to the attack level, Table 3 lists the accuracy of the final test after convergence with various malicious client ratios. The accuracy of all methods is similar when there are no opponents, which means that trust integration does not impair benign results. The higher the malicious ratio, the more accurate FedAvg becomes, whereas the trimmed mean becomes less accurate more slowly. TAFL consistently outperforms both baselines across all adversarial settings, maintaining high accuracy even when a substantial fraction of clients behave maliciously.

Table 3. Final test accuracy (%) across aggregation methods and malicious client ratios.

Malicious Ratio	FedAvg	Trimmed Mean	TAFL (Proposed)
0%	94.6	94.2	94.8
10%	88.3	91.1	93.5
20%	81.4	88.6	92.1
30%	73.2	85.0	90.4
40%	65.7	81.2	88.9

5.2.2. AUC Performance Analysis

Beyond accuracy, the area under the ROC curve (AUC) is used to assess robustness against poisoning attacks, as it captures ranking quality under class imbalance. Figure 5 shows the evolution of the test AUC across communication rounds. Under adversarial participation, FedAvg exhibits a steady decline in AUC, reflecting widespread distortion of decision boundaries. The trimmed mean improves resilience by attenuating extreme updates but converges more slowly and plateaus at a lower AUC. In contrast, TAFL maintains a consistently higher AUC throughout training, indicating that trust-based filtering and weighting preserve discriminative power even when malicious clients attempt to poison the model.

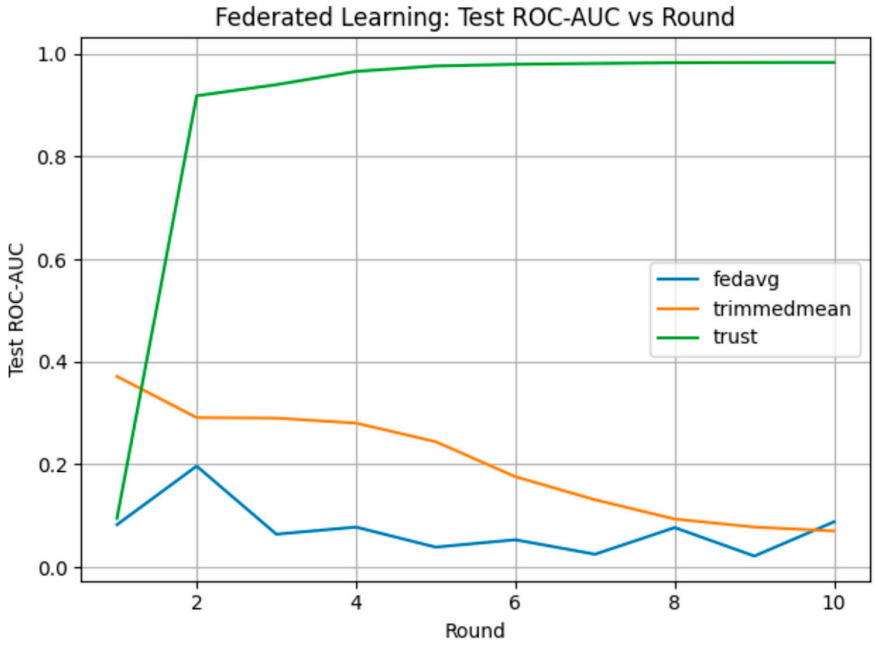

Figure 5. Test AUC versus communication rounds.

Table 4 reports the final AUC after convergence across varying malicious client ratios. While all methods perform similarly in benign settings, TAFL demonstrates superior robustness as the attack intensity increases, with a markedly slower degradation in AUC compared to both baselines.

Table 4. Final AUC scores under varying malicious client ratios.

Malicious Ratio	FedAvg	Trimmed Mean	TAFL (Proposed)
0%	0.962	0.958	0.964
10%	0.914	0.936	0.955
20%	0.872	0.908	0.943
30%	0.821	0.884	0.929
40%	0.776	0.851	0.912

The realised performance improvements are especially important for unstable acoustic channels and environments with limited communication resources in underwater settings, where the quality of the transmitted model updates can be diminished. The suggested approach offers greater robustness than general FL schemes by rejecting low-trust or unreliable updates.

5.3. Impact of Trust-Aware Filtering

5.3.1. Client Acceptance Rate over Rounds

To understand how trust-aware filtering limits adversarial influence, Figure 6 depicts the number of accepted versus selected clients per round under TAFL. Early rounds show higher acceptance as trust estimates stabilize. As training progresses, clients exhibiting unreliable behavior are increasingly excluded, resulting in a controlled reduction in participation. This selective acceptance substantially reduces attacker participation while preserving sufficient client diversity to sustain learning.

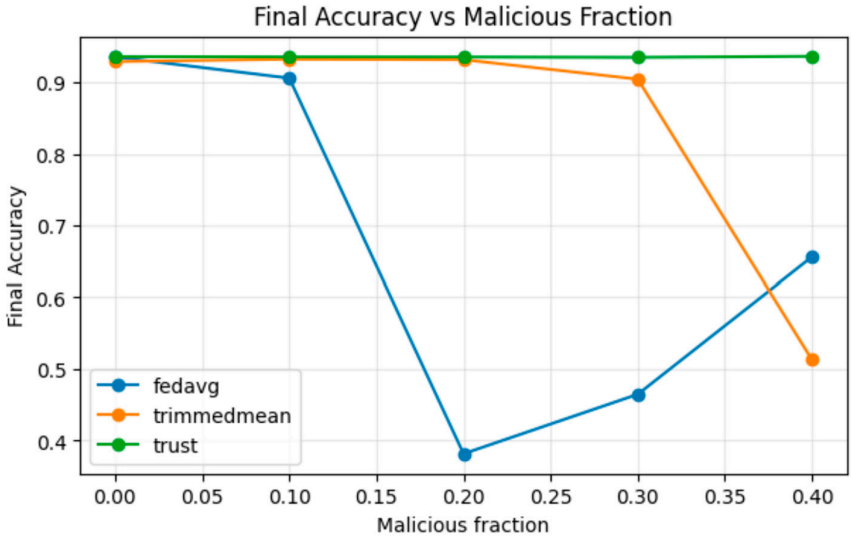

Figure 6. Accepted versus selected clients per round under TAFL.

5.3.2. Trust Score Evolution

Figure 7 illustrates the evolution of the mean trust score for honest and malicious clients. Honest clients progressively accumulate trust as their updates remain consistent with the global objective, whereas malicious clients experience a monotonic decline in trust due to abnormal updates and degraded local performance. The widening separation between the two curves confirms that the trust mechanism reliably distinguishes adversarial behavior over time, enabling precise filtering with low collateral exclusion of benign clients.

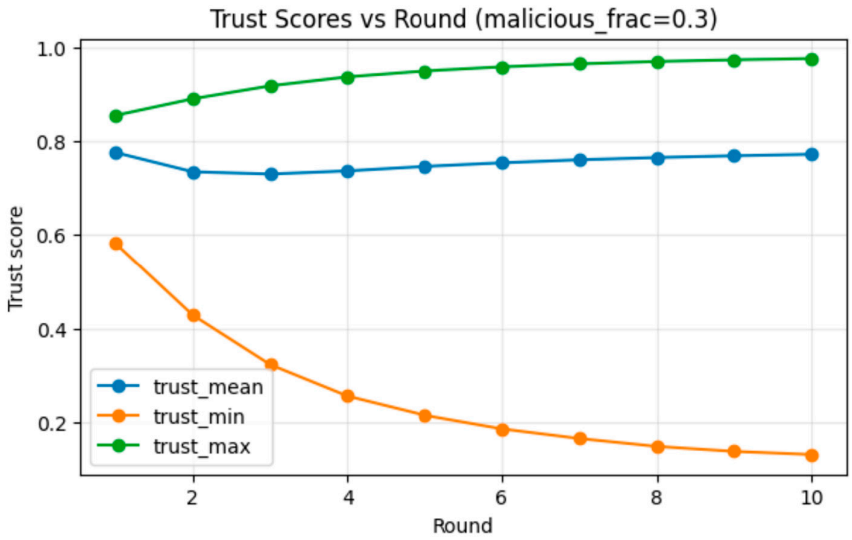

Figure 7. Mean trust score evolution for honest and malicious clients.

5.4. Convergence and Stability Analysis

Figure 8 compares the convergence of loss and accuracy across aggregation methods under adversarial pressure. FedAvg exhibits unstable convergence with oscillatory behavior driven by poisoned updates. The trimmed mean improves stability but converges more slowly due to conservative aggregation. TAFL achieves both stable and rapid convergence by suppressing low-trust contributions early, resulting in smoother optimization dynamics and faster attainment of a stable performance plateau.

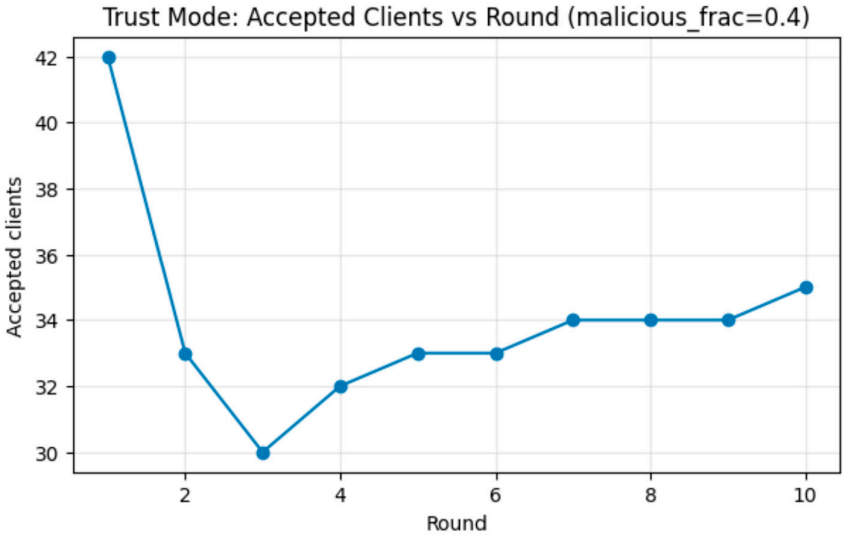

Figure 8. Loss and accuracy convergence comparison across aggregation methods to identify the malicious nodes.

5.5. Robustness Comparison with Byzantine Aggregation

While Byzantine-robust aggregation methods rely on statistical defenses, TAFL explicitly identifies and limits unreliable contributors. Table 5 compares TAFL with the trimmed mean under identical adversarial conditions. Although the trimmed mean provides baseline robustness by removing extreme updates, TAFL consistently achieves higher accuracy and AUC by combining explicit trust evaluation with weighted aggregation. This demonstrates that explicit trust modeling offers stronger protection than purely statistical robustness mechanisms.

Table 5. Performance comparison between TAFL and trimmed mean aggregation.

Metric (30% Malicious)	Trimmed Mean	TAFL (Proposed)
Accuracy (%)	85.0	90.4
AUC	0.884	0.929
Convergence Rounds	65	48
Attacker Participation (%)	~100	<40

5.6. Sensitivity Analysis to Malicious Client Ratio

Figure 9 shows the accuracy and AUC as functions of the ratio of malicious clients. The adversarial fraction of FedAvg is high, and the adversarial fraction of the trimmed mean is moderate; thus, FedAvg degrades much more than the trimmed mean. TAFL is the most resistant to degradation, and its performance remains high even during significant levels of attack. These findings show that trust-aware mechanisms offer resilience thresholds that go far beyond the ones that can be reached via robust aggregation.

5.7. Summary of Experimental Findings

Generally, the experiment's outcomes show that the inclusion of trust in federated aggregation greatly increases resilience to poisoning attacks. TAFL always performed better than normal FedAvg and Byzantine-robust aggregation in terms of accuracy, AUC, convergence stability, and attacker suppression. Its results indicate a positive security–accuracy trade-off: explicit trust modeling not only boosts resilience but also has little effect on benign performance, and therefore TAFL should be used in adversarial federated learning tasks.

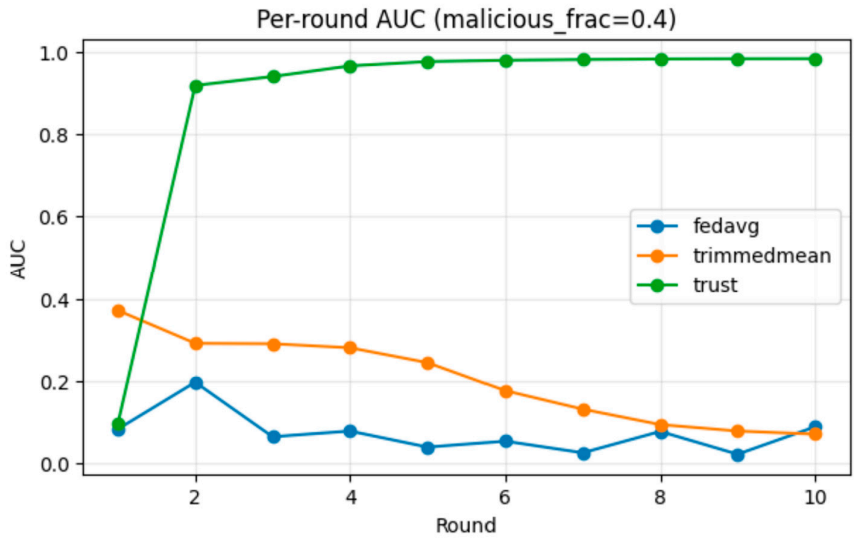

Figure 9. Accuracy and AUC versus malicious client ratio.

6. Conclusions and Future Work

In this paper, TAFL-UWSN, a trust-based federated learning system, is introduced to enhance the security and resilience of underwater sensor networks. The suggested solution combines trust assessment and federated aggregation to identify untrustworthy or malicious actors and minimize their influence on the global model. TAFL-UWSN enhances resilience to adversarial actions and preserves the effectiveness of learning in distributed underwater settings through trust-aware client evaluation, filtering, and secure aggregation. Based on the experimental findings, the proposed framework outperforms traditional baseline approaches in terms of accuracy, robustness, and attack resistance, indicating its ability to provide secure collaborative learning within underwater sensor networks.

Although these are encouraging findings, there are a number of limitations to their use. The modern assessment of this is performed on a benchmark intrusion–detection dataset that is not comprehensive of all the features of underwater acoustic communication, which include severe multipath effects, propagation delay, dynamic channel conditions, and resource variability. Besides this, the state of the device, unstable communication, and leakage of privacy were not directly modeled in the current framework.

TAFL-UWSN can be expanded in a number of ways as future work. To begin with, the implementation of the framework should be confirmed with underwater-specific datasets (or simulation settings) that are more realistic to reflect the acoustic channel conditions. Second, privacy-protective solutions like differential privacy or secure aggregation can be included further to minimize the risk of information release in federated learning. Third, the adaptive trust modeling can be improved with the consideration of the network status, node energy status, and the reliability of communication in real time. Lastly, the framework can be used on an extended set of underwater activities, such as anomaly tracking, fault monitoring, and event categorization, to further express its usefulness in safe underwater sensing systems.

Author Contributions: R.W.A. and M.A. conceptualized the study and designed the methodology. A.S. and F.U. performed the experiments and collected the data. R.W.A. and M.A. conducted formal analysis and data curation. R.W.A. prepared the first draft of the manuscript. A.S. and F.U. reviewed and edited the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: There is no funding received for this research.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The UNSW-NB15 dataset analyzed in this study is publicly available from the repository cited in reference [23] of the manuscript, and the source code is available at https://doi.org/10.5281/zenodo.18943311.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Akyildiz, I.F.; Pompili, D.; Melodia, T. Underwater acoustic sensor networks: Research challenges. *Ad Hoc Netw.* **2005**, *3*, 257–279.
2. Heidemann, J.; Li, Y.; Syed, A.; Wills, J.; Ye, W. *Underwater Sensor Networking: Research Challenges and Potential Applications*; Proceedings of the Technical Report ISI-TR-2005-603; USC/Information Sciences Institute: Marina Del Rey, CA, USA, 2005.
3. Kilfoyle, D.B.; Baggeroer, A.B. The state of the art in underwater acoustic telemetry. *IEEE J. Ocean. Eng.* **2002**, *25*, 4–27. [CrossRef]
4. Natarajan, V.P.; Jayapal, S. An improving secure communication using multipath malicious avoidance routing protocol for underwater sensor network. *Sci. Rep.* **2024**, *14*, 30210. [CrossRef] [PubMed]
5. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
6. Yin, D.; Chen, Y.; Kannan, R.; Bartlett, P. Byzantine-robust distributed learning: Towards optimal statistical rates. In Proceedings of the 35th International Conference on International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 5650–5659.
7. Rose, S.; Borchert, O.; Mitchell, S.; Connelly, S. *Zero Trust Architecture*; NIST Special Publication 800-207; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2020. [CrossRef]
8. Ganeriwal, S.; Balzano, L.K.; Srivastava, M.B. Reputation-based framework for high integrity sensor networks. *ACM Trans. Sens. Netw. TOSN* **2008**, *4*, 15. [CrossRef]
9. Sun, Y.; Han, Z.; Liu, K.R. Defense of trust management vulnerabilities in distributed networks. *IEEE Commun. Mag.* **2008**, *46*, 112–119. [CrossRef]
10. Hosseinzadeh, M.; Haider, A.; Rahmani, A.M.; Aurangzeb, K.; Liu, Z.; Yousefpoor, M.S.; Yousefpoor, E.; Lee, S.-W.; Khoshvaght, P. A Q-learning-based trust model in underwater acoustic sensor networks (UASNs). *Ad Hoc Netw.* **2025**, *178*, 103918.
11. Du, J.; Han, G.; Lin, C.; Martinez-Garcia, M. ITrust: An anomaly-resilient trust model based on isolation forest for underwater acoustic sensor networks. *IEEE Trans. Mob. Comput.* **2020**, *21*, 1684–1696. [CrossRef]
12. Wang, B.; Yue, X.; Liu, Y.; Hao, K.; Li, Z.; Zhao, X. A Dynamic Trust Model for Underwater Sensor Networks Fusing Deep Reinforcement Learning and Random Forest Algorithm. *Appl. Sci.* **2024**, *14*, 3374. [CrossRef]
13. Wu, X.; Wang, H.; Zhang, Y.; Zou, B.; Hong, H. A tutorial-generating method for autonomous online learning. *IEEE Trans. Learn. Technol.* **2024**, *17*, 1532–1541. [CrossRef]
14. Zhu, R.; Li, W.; Boukerche, A.; Yang, Q. Energy-Aware DRL-Based Dual-Perception Fountain Codes for Resource-Constrained UASNs. *IEEE Trans. Sustain. Comput.* **2026**. *Early Access*. [CrossRef]
15. Kang, J.; Xiong, Z.; Niyato, D.; Xie, S.; Zhang, J. Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet Things J.* **2019**, *6*, 10700–10714. [CrossRef]
16. Wahab, O.A.; Rjoub, G.; Bentahar, J.; Cohen, R. Federated against the cold: A trust-based federated learning approach to counter the cold start problem in recommendation systems. *Inf. Sci.* **2022**, *601*, 189–206. [CrossRef]
17. Yan, L.; Wang, L.; Li, G.; Shao, J.; Xia, Z. Secure Dynamic Scheduling for Federated Learning in Underwater Wireless IoT Networks. *J. Mar. Sci. Eng.* **2024**, *12*, 1656. [CrossRef]
18. Pokhrel, S.R.; Yang, L.; Rajasegarar, S.; Li, G. Robust zero trust architecture: Joint blockchain based federated learning and anomaly detection based framework. In Proceedings of the SIGCOMM Workshop on Zero Trust Architecture for Next Generation Communications, Sydney, Australia, 4–8 August 2024; pp. 7–12.
19. Chen, X.; Feng, W.; Ge, N.; Zhang, Y. Zero trust architecture for 6G security. *IEEE Netw.* **2023**, *38*, 224–232. [CrossRef]
20. Fu, Y.; Xie, Y.; Yi, W.; Poudel, B.; Cao, J.; Li, H. IPO-ZTA: An Intelligent Policy Orchestration Zero Trust Architecture for B5G and 6G. *Comput. Netw.* **2025**, *269*, 111450. [CrossRef]
21. Liu, Y.; Hao, X.; Ren, W.; Xiong, R.; Zhu, T.; Choo, K.-K.R.; Min, G. A blockchain-based decentralized, fair and authenticated information sharing scheme in zero trust internet-of-things. *IEEE Trans. Comput.* **2022**, *72*, 501–512. [CrossRef]
22. Xie, P.; Zhou, Z.; Peng, Z.; Yan, H.; Hu, T.; Cui, J.-H.; Shi, Z.; Fei, Y.; Zhou, S. Aqua-Sim: An NS-2 based simulator for underwater sensor networks. In Proceedings of the OCEANS 2009, Biloxi, MS, USA, 26–29 October 2009; pp. 1–7.

23. Slay, N.M.J. UNSW-NB15 DataSet for Network Intrusion Detection Systems. 2014. Available online: https://research.unsw.edu.au/projects/unsw-nb15-dataset (accessed on 19 January 2026).