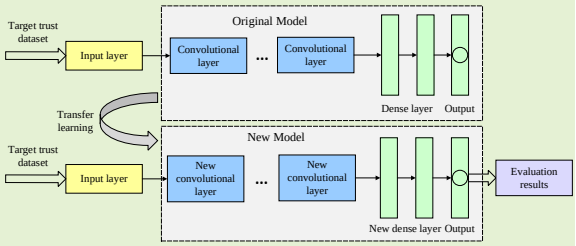
TLTrust: A Flexible Trust Model Based on Transfer Learning for Underwater Wireless Sensor Networks

Zhenquan Qin, *Member, IEEE*, Weicheng Meng, Yuxin Cui, and Bingxian Lu*, *Member, IEEE*

***Abstract*— Recently, research on trust models for underwater wireless sensor networks has been continuously deepening. However, most current trust models are trained using machine learning methods and require high costs for the purpose of gathering copious data. In addition, when facing with unknown attacks, these models are unable to detect new attacks in time. In order to solve these challenges, we propose a flexible trust model based on transfer learning, called TLTrust. TLTrust aims to update the model with a small amount of data, resist unknown attacks, and improve the flexibility of the model. TLTrust includes trust evidence collection and trust evaluation. In addition to interactive trust, data trust and energy trust, we also use location trust to measure node mobility. TLTrust model is based on transfer learning. It trained a trust model suitable for target attacks and updated trust values. Simulation results demonstrate that TLTrust has good performance in improving the accuracy of trust evaluation and dealing with unknown attacks. Under the experimental settings of different number of nodes, the average accuracy of TLTrust reaches 98.87%, 99.60% and 99.61% respectively under three different attacks. Compared with LTrust and STMS, the average improvement is about 10.44%, 6.06% and 27.19%.**



***Index Terms*— Underwater wireless sensor networks(UWSN), Trust model, Transfer learning**

I. Introduction

The deployment of underwater wireless sensor networks has expanded rapidly in recent years, covering areas like marine monitoring, disaster management, and resource exploration [1] [2] [3]. However, due to the harsh marine environment, the acoustic transmission process of underwater wireless sensor networks is unstable, and the data acquisition is difficult [4]. In this case, the sensor nodes are highly susceptible to malicious assaults, thus compromising the security of the network [5]. Therefore, the security research of underwater wireless sensor network is particularly important .

UWSN nodes are at risk of encountering different types of attacks underwate, which is related to the security and reliability of UWSN [6]. To address this issue, the trust model has gradually become the security mechanism of UWSN [6]. An adaptive trust evaluation model was proposed to detect abnormal nodes in UASN, effectively addressing node misbehavior in harsh underwater environments [7]. At present, trust models are widely used in wireless networks such as social networks and the Internet of Vehicles [8] [9]. A blockchain-driven trust management system has been introduced for vehicular ad hoc networks, enhancing trust evaluation through secure and tamper-proof transaction records. Moreover, it gradually develops into the field of underwater wireless sensor networks [10]. A trust model with multiple dimensions was developed for the detection of misbehavior in vehicular ad hoc networks, incorporating various trust factors to improve detection accuracy [11].Furthermore, it evolves gradually into the domain of UWSN [12]. Nevertheless, in light of the distinctions among differences between underwater and terrestrial environments, these trust models can not fully adapt to the application scenarios of UWSN [13]. Consequently, it is essential to take the uniqueness of the underwater environment into account and devise a trust model applicable to underwater scenarios.

In the past period of time, the studies on the trust model in UWSN has gradually deepened and made new progress. For example, Jiang *et al*. suggested a trust model relying on cloud theory (TMC) for wireless networks [14], which had significant advantages in describing the trust's indeterminacy and vagueness. In order to obtain accurate trust evaluation result, Du *et al*. put forward a trust model named ITTrust which is based on isolation forest [15]. ITrust integrated communication trust, data trust, energy trust and environment trust into a trust data set. Subsequently, the isolation forest algorithm was employed to assess the trust evidence. To reduce the latency of trust evaluation, Du *et al*. proposed a edge-computing-enabled trust mechanism, called ATrust [16]. ATrust introduced edge computing technology and attention mechanism into the trust model, and adjusted the weight of

Zhenquan Qin is with School of Software, Dalian University of Technology, Dalian, China and Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology, Dalian 116024, China. (email: qzq@dlut.edu.cn)

Corresponding author: Bingxian Lu, email: lubingxian@neu.edu.cn

each trust parameter adaptively.

Although the above methods solve the application problem of trust model in underwater to a certain extent, there are still some challenges. Taking into account the particularity of the underwater environment, trust model for UWSN should consider the following aspects:

- The trust models in existence relying on machine learning depend on from a mass of underwater nodes [17]. However, due to the poor underwater environment and the constrained energy of sensor nodes, it is impossible to obtain enough labeled data underwater to judge the reliability of nodes. The labeling process is costly in terms of both labor and time. Therefore, how to maximize the use of limited labeled data to analyze and detect the reliability of nodes is very important for improving the security of UWSN.
- When the attack mode changes, the trust model based on machine learning cannot timely update the detection model to adapt to new attacks [18].SLIP suggests a framework based on self - supervised learning for detecting model reversal and contaminating attacks, which makes it possible for zero-trust systems to dynamically adapt to evolving threats in vehicular networks. [19].Traditional machine learning methods are often trained for specific scenarios. However, the mobility of UWSN and changes in node data requirements can severely impact the performance of machine learning models and reduce their applicability. Therefore, the existing model cannot be updated when the attack mode changes.

To tackle the aforementioned challenges, the study introduces a flexible trust model based on transfer learning for UWSN named TLTrust. TLTrust consists of two parts: the first step is to collect trust evidence and process it; the second step is to input the preprocessed trust evidence into the trust model for detection and identify malicious nodes.

Compared with reinforcement learning and online learning, transfer learning has four obvious advantages:

- Data requirements: Transfer learning can accelerate learning by using existing knowledge with a small amount of target data, while reinforcement learning and online learning usually require a large amount of training data.
- Learning speed and adaptability: Transfer learning can adapt to new tasks or data faster through knowledge transfer, especially when facing new attacks or scenarios, which can improve the adaptation speed.
- Resource consumption: Transfer learning is relatively more computationally efficient because it does not require training from scratch.
- Ability to handle new tasks: Transfer learning can adapt to new tasks more quickly, especially when data is limited, while reinforcement learning and online learning often require a large amount of interactive data to adapt to new tasks.

When collecting evidence of trust, we consider the location changes of nodes to reduce the negative impact of ocean current movement on trust evaluation. Bidirectional interaction is considered during node communication to avoid inaccurate trust evaluation caused by unidirectional communication. When detecting nodes, we use the transfer learning mechanism to optimize and update the initialization model. Then we use the updated model to evaluate the reliability of nodes. The principal achievements presented within this paper are concisely encapsulated below.

- We propose a flexible trust model based on transfer learning for underwater wireless sensor networks, named TLTrust. In TLTrust ,we integrate and consider multiple factors to enhance the precision of identifying malicious nodes substantially.
- We introduce transfer learning into trust model to reduce the requirement of data volume. It is capable of efficiently resolving the issue of a small amount of data in the underwater environment.
- Through simulation experiments on different unknown attacks, the accuracy and performance of TLTrust have been greatly improved. It can verify the effectiveness of TLTrust.

II. Related Work

In this section, we outline the key studies on trust evaluation and summarize recent advancements in trust models within underwater wireless sensor networks.

A. Trust Evaluation

Trust evaluation typically encompasses three steps: evidence collection, trust modeling, and evaluation. We will review notable contributions in these areas below.

Willink introduced a possibility-based trust model for mobile wireless networks [20], which accounted for network topology changes and contextual factors. This model utilized node link capacity and data integrity as evidence, updating trust values through direct observations and reputation data. Simulation results demonstrated that this approach effectively reduced trust uncertainty by enhancing evidence credibility, offering a robust framework for trust management in mobile Ad Hoc networks.

To address underwater link instability affecting trust calculations, Su *et al*. developed a trust model leveraging fast link quality assessment (LQA) [21]. This model incorporated node energy, computational capacity, and LQA metrics such as packet reception rate, quality index, and signal-to-noise ratio. By analyzing various internal attacks, it integrated communication, data, and energy trust to compute node trust values. Simulation findings showed that the model mitigated the impact of link instability, ensuring accurate node trust representation.

Guo *et al*. proposed TROVE, a context-aware trust management model designed for vehicular networks [22]. TROVE encompassed data formalization, trust evaluation, and adaptive strategy adjustment. Using information entropy theory, it quantified evaluation-related information. A reinforcement learning framework dynamically refined strategies based on historical feedback, enabling TROVE to optimize trust evaluation and maintain high performance across diverse scenarios.

Suresh *et al*. proposed a trust evaluation method for mobile ad hoc networks (MANETs). They integrated the Bayesian Best - Worst Method with a modified Grey PROMETHEE - AL model [23]. This method efficiently reduces problems such as rank bias, rank reversal, and information uncertainty, enhancing the recognition and isolation of malicious and selfish nodes.

B. Trust Model for UWSN

Underwater Wireless Sensor Networks (UWSN) are essential for ocean monitoring and disaster warning, continuously generating large volumes of data. However, the harsh underwater environment makes sensor nodes highly susceptible to malicious attacks, and accurately evaluating node trustworthiness is crucial for ensuring the security and stability of UWSNs.

To assess node trust comprehensively, Du *et al*. proposed ITrust, a mechanism based on the isolated forest algorithm [15]. ITrust integrates communication, data, energy, and environmental trust into a single dataset, using the isolated forest algorithm to quantify environmental effects. This approach proves particularly effective in noisy environments.

To enhance trust detection flexibility in dynamic environments, He *et al*. introduced TUMRL, a reinforcement learning-based trust update mechanism for Underwater Acoustic Sensor Networks (UASN) [6]. TUMRL incorporates node criticality, making nodes with higher criticality more responsive to malicious attacks. This adaptability allows TUMRL to effectively respond to changing underwater threats while maintaining high detection accuracy.

Existing trust models overlook the impact of defective recommendations. Du *et al*. addressed this gap with LTrust, an adaptive trust model based on Long Short-Term Memory (LSTM) [24]. LTrust accounts for node movement and its effect on network topology, and filters recommendations to improve the reliability of trust evaluations.

To resolve trust conflicts from dishonest recommendations, Jiang *et al*. proposed the Controversy-Adjudication-Based Trust Management mechanism (CATM) [25]. CATM considers link and node reliability to adjudicate disputes, while an incentive mechanism based on the prisoner's dilemma encourages broader participation in recommendations, enhancing the accuracy of trust evaluations.

Despite the advancements in trust models, challenges remain. Many existing models for UWSN require large datasets, making them more suitable for dense networks, but less effective in sparse ones. To address the data collection challenges in underwater environments, we propose a transfer learning-based trust model. This model can update and optimize itself using a small amount of data, significantly reducing data requirements while improving detection efficiency and model flexibility.

III. Preliminary

A. Network Model

Fig. 1 shows our network model. We used N sensor nodes randomly to distribute in a certain range of sea area. The surface sinks are deployed on the water surface. Each sensor node has equal initial energy, finite computing power and storage, powered by a non-rechargeable battery. The sensor nodes are responsible for communication underwater. the communication range is a given communication radius $maxCom$, and the information of other nodes is collected within the communicable range. The area's sensor nodes are grouped into clusters, where nodes within each cluster communicate directly. The cluster head node CH collects data from the cluster members and periodically transmits the trust evidence to the surface sink. Surface sinks receive messages from cluster heads and processes them. Then sinks can communication with base station and satellite.

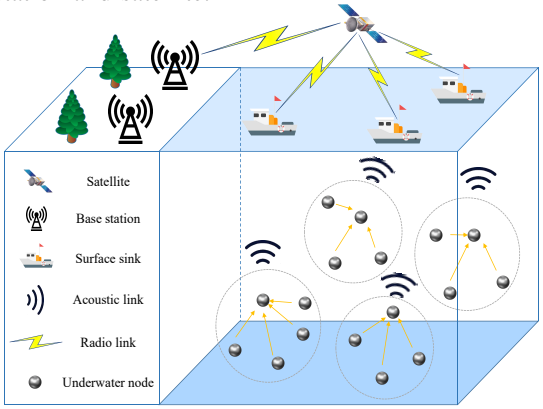


Fig. 1. Network Model

B. Problem Statement

This section details the research problem. Malicious node detection involves generating trust parameters and evaluating trust evidence. Based on the interactions of underwater sensor nodes, we select appropriate parameters as trust evidence, and send the trust evidence into the trust model for detection after integrated processing. A hybrid trust model was proposed to counter insider packet drop attacks in wireless sensor networks, effectively integrating multiple trust parameters to enhance attack detection [26]. Malicious nodes make up a small proportion of sensor networks. And when malicious nodes execute different attacks, these malicious nodes are different from normal nodes in communication and energy consumption. In this paper, a transfer learning mechanism is used to build a trust model. When the nodes change their attack methods, a small amount of data can still be used to effectively detect malicious nodes. Some parameters are defined as follows:

a) The trust evidence dataset $Trust_{Matrix}$ includes n sets of data generated by n sensor nodes in UWSN, $Trust_{Matrix} = \{T_1, T_2, \cdots, T_n\}$.
b) The trust parameter T_i of node n_i is a row of data of the trust matrix. And T_i includes $Trust_{loc}$, $Trust_{inter}$, $Trust_{data}$, $Trust_{energy}$.

IV. Design of TLTrust

The structure of the TLTrust model is shown in Fig. 2. The TLTrust model consists of three parts: trust evidence collection, trust modeling, and trust evaluation. In the trust

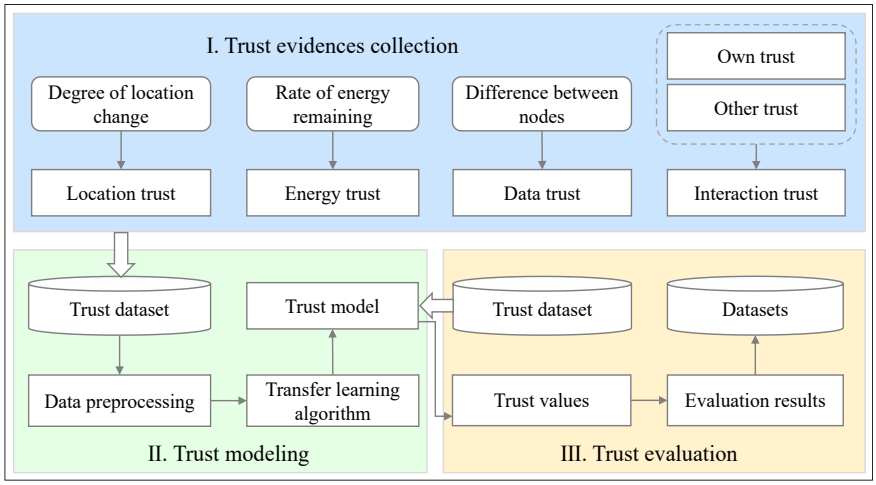


Fig. 2. The Structure of TLTrust

evidence collection stage, we focus on the location change, interaction behavior, data and remaining energy of nodes. These four parts serve as the parameters of the trust evidence data set. In the trust modeling stage, a transfer learning mechanism is adopted. First, we input the source data into the pre-trained model for updates. Next, the target data is fed into the new trust model to calculate the node's trust value. Finally, surface sinks assess the node's trustworthiness based on this value.

A. Trust Evidence Collection

Trust evaluation relies on collecting trust parameters, focusing on node communication, data transmission, and energy usage. By analyzing the effects of internal attacks, three outcomes are identified: communication failure, packet errors, and abnormal energy consumption [27]. As shown in Table I.

TABLE I
ATTACKS AND CONSEQUENCES

Attack types	Consequences
Black-hole	Packets loss and low energy consumption
Grey-hole	Some packets loss and lower energy consumption
Bad-mouthing	Deliberately decrease trust value of normal node
Good-mouthing	Deliberately increase trust value of malicious node

In addition, the position of underwater sensor nodes will change due to the movement of ocean currents, which will affect the communication of UWSN. Therefore, we consider the change of node position into the trust evidence.

1) Calculation of Location Trust: Due to ocean currents, the positions of sensor nodes may shift over time. To model this movement, we introduce the mobility model of sensor nodes [28]. As shown in Fig. 3, the node SN has a maximum communication range of $maxCom$, with the coordinate system centered at point O. The angles between node SN and the Z axis and X axis are α and β, respectively. Let the node's speed at time t be $v_i(t)$, and its direction be represented by $(\mathrm{d}\alpha_i(t), \mathrm{d}\beta_i(t))$, where $\alpha, \beta \in (0, 2\pi)$. The node's position at time t is $(x_i(t), y_i(t), z_i(t))$, and its location at time $t+1$ can be expressed as:

$$loc_i(t+1) = \begin{cases} x_i(t) + v_i(t) \sin d\alpha_i(t) \cos d\beta_i(t) \\ y_i(t) + v_i(t) \sin d\alpha_i(t) \sin d\beta_i(t) \\ z_i(t) + v_i(t) \cos d\alpha_i(t). \end{cases} \tag{1}$$

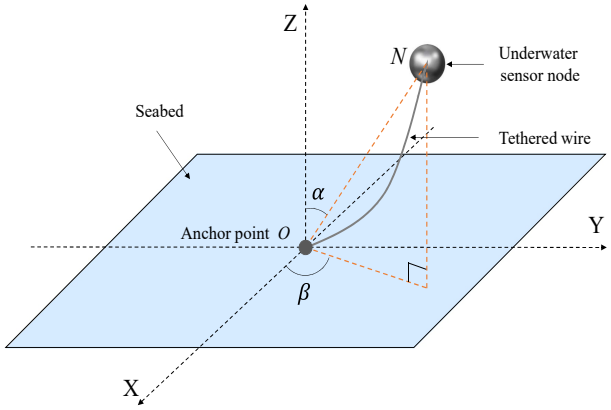


Fig. 3. The Structure of Mobility Model

The distance between node i and node j at time $t+1$ could be expressed as follows:

$$dist_{ij}(t+1) = |loc_i(t+1) - loc_j(t+1)|. \tag{2}$$

$dist_{ij}$ represents the change of the distance between two nodes with time. Location trust represents the impact of ocean current movement on node location stability, and is defined as follows:

$$Trust_{loc} = 1 - \frac{\triangle dist_{ij}}{maxCom}. \tag{3}$$

$maxCom$ is the maximum communication radius. $\triangle dist_{ij}$ is the distance change between node i and node j.

2) Calculation of Interaction Trust: In this paper, we adopt the improved K-means clustering algorithm to cluster sensor nodes in the area. K-means++ selects the point farthest from the currently selected center as the new cluster center, making the initial selection of cluster centers more uniform, avoiding the local optimal solution caused by random selection of initial cluster centers in traditional K-means, and helping to reduce

the instability of clustering results. Each cluster member can communicate directly with other members and send information to the cluster head. When malicious nodes launch malicious attacks, they may deliberately drop all or part of the data packets and destroy communication security. We take the successful communication rate of nodes as an indicator of node communication behavior [29]. The successful communication rate of nodes $SCom$ is expressed as follows:

$$SCom = \frac{c_s + 1}{c_s + c_f + 2}. \tag{4}$$

Here, c_s represents the count of successful data transmissions, while c_f indicates the quantity of failed data transmissions.

In UWSNs, each sensor node functions as both a sender and receiver. To better evaluate node communication, we use bidirectional communication as a measure of interaction trust.

We store the communication of each node in the asymmetric matrix $Com_{n\times n}$.

The diagonals of the matrix are all 1, indicating that the node trusts itself. We divide the trustworthiness of nodes in the interaction process into two types. The first is the trustworthiness of the current node to other nodes. For example, the data in the i_{th} row represents the trustworthiness of the i_{th} node to other nodes; the second is the trustworthiness of the current node to other nodes. One is the trust degree of other nodes to the current node. For example, the j_{th} column of data represents the trust degree of other nodes to the j_{th} node. According to these two trusts, we comprehensively define the interactive trust of node i, which is expressed as follows:

$$Trust_{inter} = \frac{\lambda \sum_{i=0}^{n} Com[j][i] + \mu \sum_{j=0}^{n} Com[i][j]}{n}. \tag{5}$$

Where $\lambda, \mu \in (0, 1)$ are the weights of the two factors, and $\lambda + \mu = 1$.

3) Calculation of Energy Trust: The energy of the underwater sensor node is a critical factor in maintaining its reliability and trustworthiness in the network. Since malicious nodes may exhibit abnormal energy consumption patterns, it is crucial to accurately estimate the energy of a node. We propose a comprehensive method to estimate the energy consumption of nodes by considering both their transmission activity and the associated communication overhead.

To calculate the energy trust of a node, we first define the total energy consumption of the node, considering factors such as the number of transmissions, data size, and time spent in communication.

The paper mainly focuses on the energy loss caused by communication between sensor nodes and node movement, and considers the node communication distance d_c, transmission power p_c and mobile energy consumption power p_m. The node communication distance d_c of t is:

$$d_c = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - zj)^2}. \tag{6}$$

Where (x_i, y_i, z_i) is the location of the node i at time t.

The transmission speed of the sensor node is v_c, then the transmission time t_c can be expressed as $t_c = \frac{d_c}{v_c}$.

The distance d_m moved by the node at time t is:

$$d_m = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - zi - 1)^2}. \tag{7}$$

Where $(x_{i-1}, y_{i-1}, z_{i-1})$ is the location of the node i at $t - 1$.

If the sensor movement speed is v_m, the movement time t_m can be expressed as $t_m = \frac{d_m}{v_m}$.

E is the initial energy of the node, and the remaining energy of the node E_{res} is:

$$E_{res} = E - p_c t_c - p_m t_m. \tag{8}$$

We then estimate the remaining energy (E_{res}) of the node using these parameters. The initial energy (E) of the node is known, and the energy trust is defined as:

$$Trust_{energy} = \frac{E_{res}}{E}. \tag{9}$$

Where E represents the initial energy of the node, and E_{res} represents the remaining energy of the node. This method ensures that the energy consumption is measured based on actual network activity rather than a simple static value, reducing the risk of falsification.

To prevent the potential falsification of energy values by malicious nodes, we propose incorporating additional security measures. One approach is to employ hardware-based energy monitoring, where a trusted hardware component on the node records energy consumption directly, preventing software manipulation. Additionally, nodes can periodically report energy consumption data to a central or distributed verification system, where inconsistencies can be flagged for further investigation. These steps mitigate the risk of malicious nodes artificially inflating their remaining energy to evade detection.

4) Calculation of Data Trust: In wireless sensor networks, malicious attacks will cause abnormal data packets. Thus, the packets received from maliciously attacked nodes will be very different from those received from normal nodes [30]. Since the packages are spatially related, the packets sent by neighbors are always similar [30]. The values of these packets follow a normal distribution $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\varphi)^2}{2\sigma^2}}$ [31]. The average of the data items best represents the data similarity [32]. We determine if a node is suffering from a malicious attack by analyzing the data packet values. The concept of data trust is defined in the following manner:

$$Trust_{data} = 2\left(0.5 - \omega \int_{\varphi}^{v_d} f(x)dx\right) = 2\omega \int_{v_d}^{\infty} f(x)dx. \tag{10}$$

Where ω is the weight factor assigned to critical events ($\omega > 1$ for outliers), v_d represents the data value from the target node and φ is the mean of the dataset. The greater the difference between v_d and φ, the lower the data trust value. This weighting method ensures that the model is more sensitive to abnormal events, thereby improving its ability to detect rare but important malicious behaviors.

B. Trust Modeling and Evaluation

1) Trust Modeling Based on Transfer Learning: Trust modeling based on transfer learning includes two parts: a) Preprocessing the collected trust evidence; b) Performing transfer learning on the pre-trained model to extract and classify the target trust data set.

a) Data preprocessing

Due to the instability of underwater acoustic communication, the trust evidence collected by sensor nodes may contain some irregular values. The normalization process makes the trust evidence between $[0, 1]$. Taking the location trust as an example, the normalization process is as follows:

$$Trust_{loc} = \frac{Trust_{loc} - Trust_{loc}^{min}}{Trust_{loc}^{max} - Trust_{loc}^{min}}. \tag{11}$$

Where $Trust_{loc}^{max}$ represents the maximum value of location trust $Trust_{loc}$, $Trust_{loc}^{min}$ represents the minimum value of $Trust_{loc}$, and $Trust_{loc}$ is the normalized location trust. The same process is used for interaction trust, data trust and energy trust.

b) Training based on transfer learning algorithm

In the process of trust evaluation, this paper adopts the transfer learning algorithm to evaluate and classify the trust value. In previous work, algorithms such as decision tree [5], isolation forest [15], SVM [33] and LSTM [24] have been used, which have good results in the detection of malicious nodes. However, when the attack method of malicious nodes changes, the data of the new attack method may be scarce. These algorithms cannot make the original model fit the characteristics of the new data with a small amount of data, while transfer learning can solve the new classification task with a small amount of data [34].

For deep convolutional networks, usually the first few layers of the network learn general features, and deeper networks tend to learn specific features. Therefore, the general features can be transferred to other fields [34] [35]. The key to transfer learning is to fine-tune the model to reduce data requirements and training time, as shown in the abstract. Specific steps are as follows:

- Build and initialize the CNN-based network model M, set the trust threshold $\theta \in [0, 1]$, the number of training rounds $epoch \in [100, 200]$.
- Input the sample data set $D = \{D_{loc}, D_{inter}, D_{data}, D_{energy}\}$ to retrain the initial network M and obtain a new parameter set $W_1 = \{w_1, w_2, w_3, \cdots, w_k\}$ of the model. The new network model is M_1.
- Select a certain amount of K data from the preprocessed data T_s as the transfer training set, and input it into the model M_1.
- Get the parameter W_1 of the model M_1, add the regularization term $\Omega(M_1)$, and get the target model M_2 for detecting new attacks. The regularization process is described as follows:

$$\widetilde{J}(M_1; D, y) = J(M_1; D, y) + \sigma\Omega(M_1) \tag{12}$$

Where J is the original objective function, $\widetilde{J}$ is the objective function with the regularization term $\Omega(M_1)$, D is the high-dimensional input, y is the output, and the regularization term uses the regularization weight σ to regularize the parameters of model M_1.

- The target model parameters and the source model parameters can be modeled as follows:

$$\begin{aligned} M_s &= M_0 + v_s \\ M_t &= M_0 + v_t \end{aligned} \tag{13}$$

Among them, M_0 is a task-independent parameter, namely the energy loss and whether there is packet loss during communication, which is the part that is transferred in model transfer learning. v_s and v_t are specific features.

- Fine-tune model M_1 with K sample data to obtain new parameters $\widetilde{W_2}$. Update the model M_1 to obtain a detection model $\widetilde{M}$ suitable for the new attack method. The remaining sample data is used as the target domain data $T_t = [T_{t_{loc}}, T_{t_{inter}}, T_{t_{data}}, T_{t_{energy}}]$ for verification.

In the process of detecting malicious attacks on underwater wireless sensor networks, it is assumed that the sensor network is first attacked by Bad-mouthing. At this time, the trust model is trained to obtain model M_1. After n rounds, the attack method is changed to Grey-hole attack. After a small amount of target domain data, the parameters of model M_1 are fine-tuned to obtain model M_2 suitable for target domain data.

2) Trust Evaluation: In the trust evaluation stage, we evaluate the trust value of each node according to the output result of the TLTrust model. In the same time, the node category is classified. The classification is based on the following:

$$Node_{class} = \begin{cases} 1, & (tv < 0.5); \\ 0, & (tv \geq 0.5). \end{cases} \tag{14}$$

Where tv is the trust value of sensor node. Trust threshold is 0.5.

In the trust evaluation, we use a sliding time window to evaluate the nodes in each time period to ensure real-time detection of abnormal behavior of sensor nodes and maintain network security.

The details of trust evaluation are shown in Algorithm 1.

Algorithm 1 Trust evaluation.

Require:
Trust matrix, TLTrust;

Ensure:
$Detection$: predictions of n nodes;
$accuracy$: prediction accuracy;

$Detection = \emptyset, accuracy = 0$;
for $i \leftarrow 1$ to n **do**
  $probability \leftarrow TLTrust(TD)$;
  if $p[i] >= 0.5$ **then**
    $Detection[i] \leftarrow 0$;
  else
    $Detection[i] \leftarrow 1$;
  end if
end for
$accuracy \leftarrow accuracy\_score(Detection, Label)$;

This section evaluates the performance of the TLTrust model by comparing it with STMS [33] and LTrust [24]. Trust evidence classification and evaluation are based on transfer learning. The default simulation parameters are listed in Table II. For a fair comparison, the same simulated UWSN is used to test all three models.

V. Experiments and Results

A. Experimental Setup

The implementation of the TLTrust model takes place in Python3.6 and tensorflow2.0.0. A total of 100 sensor nodes are randomly distributed within a cubic region with a side length of 500 meters.

For comparison, the STMS and LTrust models are chosen, each leveraging multi-faceted trust evidence. We will describe them in detail below.

STMS [33]: STMS is a trust prediction model based on support vector machine. In this approach, the network is partitioned into several clusters, with cluster heads and members working together to perform tasks. Moreover, a dual-cluster head strategy is employed to enhance both the security and the lifetime of the network.

LTrust [24]: LTrust is an adaptive trust model based on LSTM, which considers the influence of underwater node movement on network topology. LTrust adopts recommendation filtering algorithm in the model to ensure the accuracy of recommendation trust.

TABLE II
Simulation Parameters

Parameters	Default value
Nodes	100
Volume	$500 \times 500 \times 500m^3$
Communication range r	$200m$
Initial energy E	$100J$
Ratio of malicious nodes	0.3
Trust threshold θ	0.5

Evaluation metrics: The performance of the TLTrust model is assessed through the following metrics.

The $Accuracy$ is the percentage of the total sample that predicts the correct result. As accuracy increases, the overall prediction accuracy improves, leading to better model performance.

The $Precision$ indicates the degree of accuracy of prediction in positive sample results.

The $F1-score$ is calculated as the weighted average of $precision$ and $recall$. A high $F1-score$ is achieved only when $precision$ and $recall$ are both at elevated levels. $F1-score$ is particularly suited for imbalanced datasets with a significant disparity between positive and negative sample proportions.

The $trust\ value$ indicates the result of assessing the trustworthiness of the nodes, and the $trust\ values$ of normal and malicious nodes are evaluated in the simulation experiments.

B. Transfer Performance of the TLTrust

This section analyzes the importance of the transfer learning mechanism in TLTrust.

1) The Performance of TLTrust in Different Experimental Settings: This section analyzes the importance of transfer learning mechanism in TLTrust. In this part, we execute three different sets of experiments. The source dataset is the Black-hole attack dataset and the test set is the Grey-hole attack dataset. The three sets of experiments are set up as follows.

a) The number of nodes is 100, the ratio of malicious nodes is 30%, and the time period $[1, 30]$.
b) The time period is 30, the ratio of malicious nodes is 30%, and the number of nodes is $[10, 100]$.
c) The number of nodes is 100, the time period is 30, and the ratio of malicious nodes is $[0.05, 0.5]$.

TLTrust is evaluated in the case of direct attack initiation. We compared the $accuracy$ of TLTrust on the test set when transfer the training data set for 0, 5, 10, 15, and 20 time periods for the target data. 0 indicates that no transfer learning is performed. Fig. 4 displays the experimental results.

As illustrated in Fig. 4, when TLTrust transfers, the accuracy of detecting malicious nodes in the target data set is much higher than that without transferring. Even if the data set used for transfer training is only 5 time periods, the accuracy of TLTrust can reach more than 90%, which is much higher than the performance without transferring. Therefore, only a minimal amount of data from the target domain is necessary to improve the detection performance of the trust model in the relevant domain.

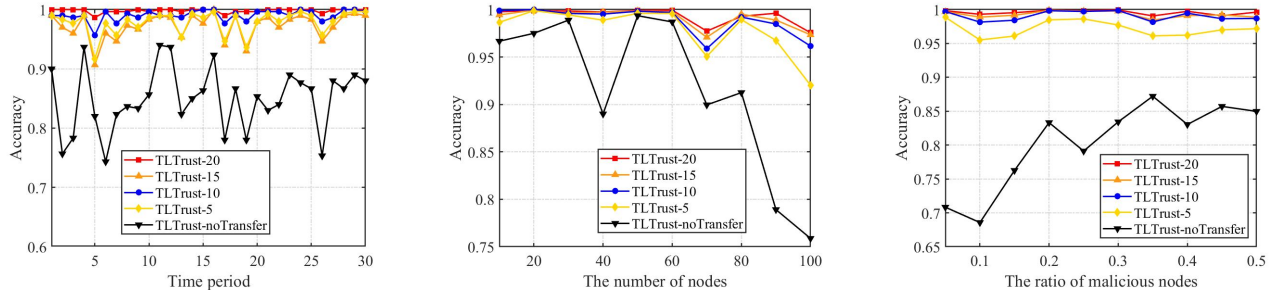
Fig. 4 (b) shows that when the transfer training data set size is 10 time periods, the accuracy of TLTrust has stabilized above 95%, and there will not be large fluctuationsdue owing to the growing node density. In the absence of transferring, the performance of TLTrust fluctuates greatly and decreases significantly. This is because TLTrust does not learn the characteristics of Grey-hole attacks, so it cannot accurately detect the harmful nodes that perform Grey-hole attacks.

The findings presented in Fig 4 (c) demonstrate that as the proportion of malicious nodes rises, TLTrust performance with a transfer learning mechanism fluctuates slightly. But the accuracy can keep above 95% by using only 5 time periods of transfer training data. In contrast, TLTrust performance without transferring shows an overall upward trend, but the fluctuation range was large and does not reach 90%.

To sum up, when we introduce the transfer learning mechanism, no matter the time period, number of nodes or ratio of malicious nodes changes, TLTrust can quickly learn the characteristics of target attacks and detect the malicious nodes launching new attacks with a small amount target attack data.

2) The Performance of TLTrust in Different Evaluation metrics: In order to show the performance of TLTrust more comprehensively, we calculate the $precision$ and $F1-score$ of TLTrust under the same experimental setting. The results are shown in Table III and Table IV.

The results in Table III and Table IV show that compared with TLTrust, the $precision$ and $F1-score$ of TLTrust without transferring mechanism are much lower, and the performance is more volatile. This is because TLTrust without transferring only relies on the attack characteristics of Black-hole learned in the initial training and cannot adapt to Grey-hole attack well. When the size of the transfer training data set



(a) Accuracy over time. (b) Accuracy & Number of nodes. (c) Accuracy & The ratio of malicious nodes.

Fig. 4. Comparison of accuracy.

TABLE III
PRECISION OF TLTRUST

Number of transfer	20	15	10	5	0
Time period	0.9968	0.9356	0.9845	0.9556	0.6922
Number of nodes	0.9805	0.9790	0.9650	0.9480	0.8241
Ratio of malicious nodes	0.9812	0.9675	0.9576	0.8838	0.5482

TABLE IV
PRECISION OF TLTRUST

Number of transfer	20	15	10	5	0
Time period	0.9969	0.9571	0.9845	0.9662	0.7963
Number of nodes	0.9892	0.9858	0.9814	0.9664	0.8878
Ratio of malicious nodes	0.9899	0.9827	0.9767	0.9343	0.6770

is the data of 10 time periods, the precision and $F1-score$ indexes are stable above 95%, which indicates that after 10 time periods of transfer training, TLTrust has been able to timely update the model according to the existing knowledge and new knowledge extracted from a small amount of target data. Thus, the malicious nodes that launch Grey-hole attacks can be identified accurately.

C. Comparison with Existed Trust Model

In this section, we compare the performance of TLTrust with LTrust and STMS. According to the experimental results of the ratio of transfer training data set in Sec. IV-B, when the size of the training data set is the target domain data of 10 time periods, TLTrust can achieve ideal effects in the three indexes of $accuracy$, $precision$ and $F1-score$. Therefore, in the experiment in this section, we use data of 10 time periods as the transfer training set. All three trust models are evaluated using the same set of test data.

Within this segment, we conduct a performance comparison among TLTrust, LTrust, and STMS. As per the experimental outcomes regarding the proportion of the transfer training data set detailed in Sec. IV-B, when the scale of the training data set amounts to the target domain data spanning 10 time periods, TLTrust is capable of attaining optimal results across the three metrics: $accuracy$, $precision$ and $F1-score$. Consequently, during the experiment in this section, we utilize data from 10 time periods as the transfer training set. All three trust models are appraised by means of an identical set of test data.

1) Comparison of Performance under Different Attack Modes: In this part, we carried out three groups of experiments. The source dataset was the Black - hole attack dataset, while the test datasets consisted of the Grey - hole attack dataset, Bad - mouthing attack dataset, and Good - mouthing attack dataset. For these experiments, we configured the number of nodes as 100, the proportion of malicious nodes at 30%, and the time range from 1 to 30. We compared the $accuracy$ of TLTrust, LTrust and STMS in three attack type test sets, and Fig. 5 illustrates the results of the experiments.

As can be seen from Fig. 5, the performance of TLTrust is significantly better than that of LTrust and STMS models under three attack modes. When the time period increases, the performance of TLTrust is always stable at more than 90%, while LTrust and STMS both produce large fluctuations. This is because TLTrust has learned the knowledge of three attack modes, Grey-hole, Bad-mouthing and Good-mouthing. And TLTrust can accurately identify the node that launched the malicious attack based on the trust evidence.

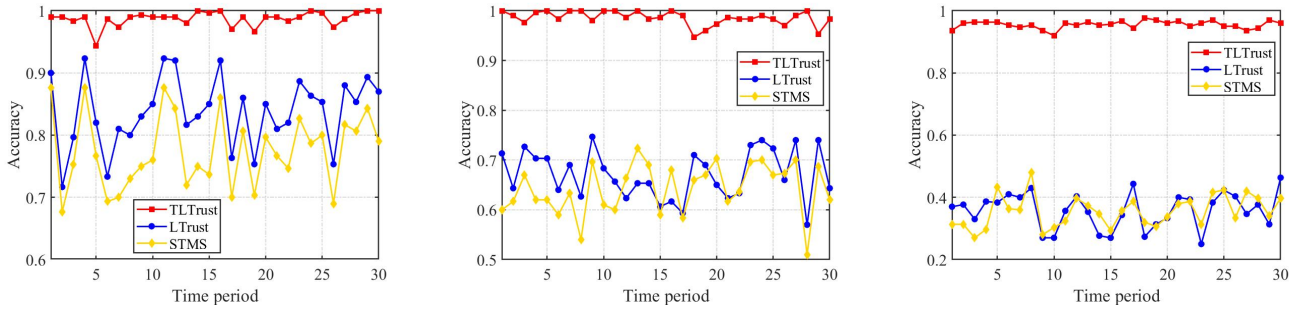
In order to test the performance of TLTrust more comprehensively, we test the $F1-score$ of three models for Bad-mouthing attack under the same experimental setting, and the results are shown in Table V.

TABLE V
COMPARISON OF F1-SCORE UNDER BAD-MOUTHING ATTACK

Model	TLTrust	LTrust	STMS
F1-score	0.9901	0.5106	0.5520

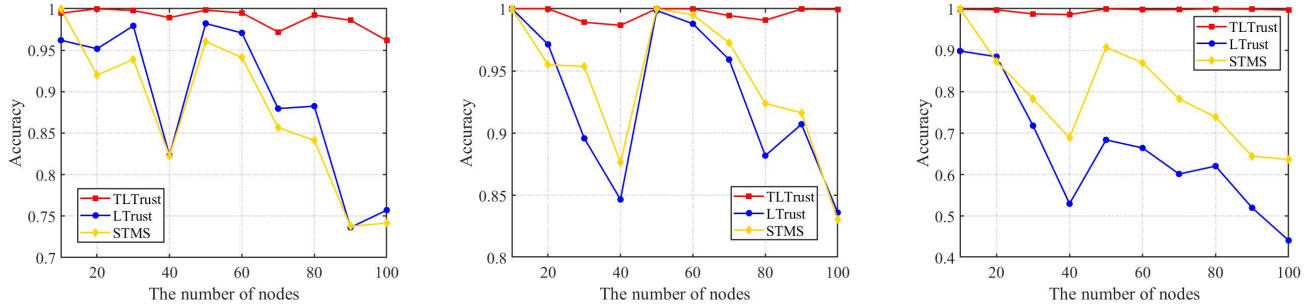
Table V shows that the average $F1-score$ of TLTrust reaches 99% in 30 time periods, while $F1-score$ of LTrust and STMS is only 51.06% and 55.20% under the same conditions. Therefore, the transfer learning mechanism enables TLTrust to quickly learn the characteristics of malicious nodes that launch Bad-mouthing attacks, so as to accurately detect malicious nodes in the test set. However, LTrust and STMS can not learn knowledge about Bad-mouthing attack, so their performance in the test set was not ideal.

2) Comparison of Performance under Different Experiment Settings: In order to further test the versatility of TLTrust, two sets of experiments were conducted in this section. The source data set was Black-hole attack data set, and the test set was Grey-hole attack data set, Bad-mouthing attack data set



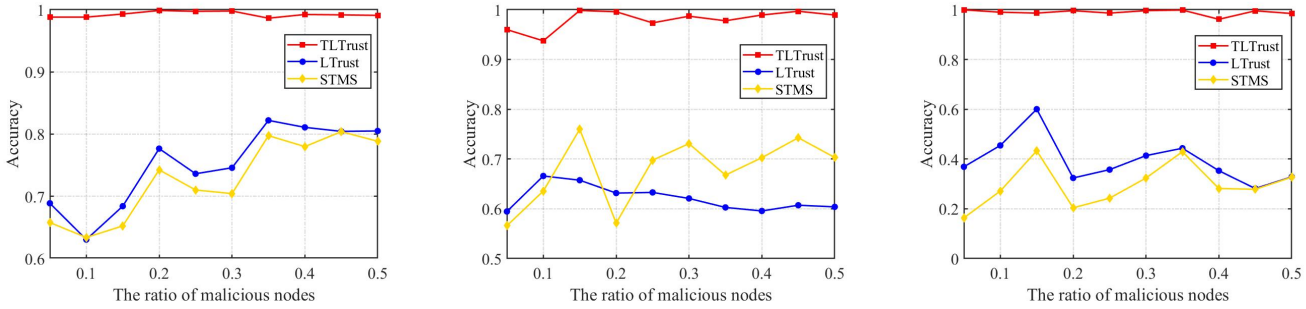
(a) Grey-hole attack. (b) Bad-mouthing attack. (c) Good-mouthing attack.

Fig. 5. Comparison of accuracy over time.



(a) Grey-hole attack. (b) Bad-mouthing attack. (c) Good-mouthing attack.

Fig. 6. Comparison of accuracy under different number of nodes.



(a) Grey-hole attack. (b) Bad-mouthing attack. (c) Good-mouthing attack.

Fig. 7. Comparison of accuracy under different ratio of malicious nodes.

and Good-mouthing attack data set. The experimental settings are:

a) The time period is 30, the percentage of malicious nodes is 30%, and the number of nodes is $[10, 100]$.
b) The number of nodes is 100, the time period is 30, and the ratio of malicious nodes is $[0.05, 0.5]$.

We evaluated and contrasted the $accuracy$ of TLTrust, LTrust, and STMS on the test set. The outcomes of these experiments are presented in Fig. 6 and Fig. 7.

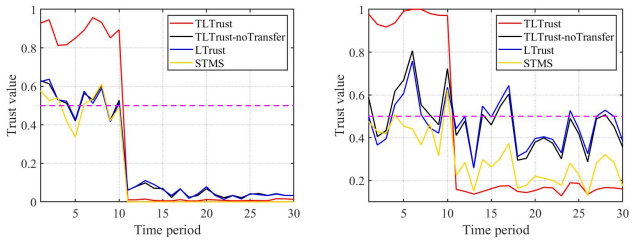
In Fig. 6, as the number of nodes grows, TLTrust maintains an accuracy of over 95% across the three attack modes. While LTrust and STMS initially exhibit high accuracy, their detection rates decline substantially as the number of nodes keeps increasing. Notably, when the node count reaches 100, LTrust's accuracy under Good - mouthing attack plummets below 50%.

Turning to Fig. 7, as the proportion of malicious nodes rises, TLTrust's performance remains consistently stable, with an accuracy above 90%. In contrast, LTrust and STMS show varying degrees of performance fluctuations, and their malicious node detection accuracy is far from optimal, particularly during Good - mouthing attacks. The accuracy of these two models is frequently below 50%.

Therefore, when we adopt the transfer learning mechanism in TLTrust, the model quickly obtain the characteristics of the target attack and update the model. By using this approach, the trust model's accuracy in identifying malicious nodes within the target domain can be enhanced.

3) Comparision of Average Trust Value: In this experiment, we set the malicious node to launch the attack at $Time = 10$, and test the trust evaluation of TLTrust, TLTrust without transferring, LTrust and STMS to the malicious node and normal node under Grey-hole attack and Bad-mouthing attack in $[1, 30]$ time periods respectively. The results are shown in Fig. 8.

As shown in Fig. 8 when $Time = 10$, trust values of malicious nodes in all four model decline rapidly. However, the TLTrust that adopts the transfer learning mechanism has a high trust evaluation value when malicious nodes do not launch attacks, which does not affect the normal communication



(a) Grey-hole attack. (b) Bad-mouthing attack.

Fig. 8. Trust value of malicious nodes over time.

behavior of the network. In the other three cases, the evaluation of malicious nodes without attack is not accurate enough, and the normal network communication will be affected when the nodes are further processed.

In conclusion, transfer learning mechanism significantly improves the ability of TLTrust to learn new attack features. So TLTrust can distinguish malicious nodes from normal nodes accurately and ensure network security.

D. The Performance of TLTrust on NSLKDD Dataset

To better validate the performance of the TLTrust model, we re-validated it on the NSLKDD dataset [36]. The features of the NSLKDD dataset are as follows:

a) The training set and test set of the NSLKDD dataset do not contain redundant records, which makes the detection more accurate.
b) The NSLKDD test set contains attack types not in the training set.

We choose KDDTrain as the training set and KDDTest as the validation set to verify the performance of TLTrust and LTrust under different ratio of the transfer training dataset.

In this set of experiments, we compare the performance of TLTrust-Transfer, TLTrust-noTransfer and LTrust on the test set. When applying TLTrust, we split the KDDTest dataset into a transfer training set and a test set in a certain ratio, with a split ratio of $[0.05, 0.5]$ and a stride of 0.05. We set the transfer training epoch to 100. The experimental results are shown in Fig. 9.

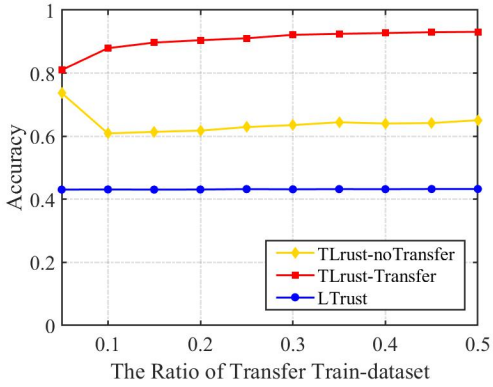


Fig. 9. Accuracy of different models on the NSLKDD dataset.

Fig. 9 shows that compared to LTrust and TLTrust without transferring, our TLTrust model with transfer has the best performance on the test set, and the performance is more stable. This is because, the NSLKDD test set contains some attack types that are not in the training set. LTrust and TLTrust without transfer cannot identify the features of the new model well, which results in lower detection accuracy. When we obtain a part of the test set data from the migrated TLTrust model, we first use a small part of the test set data to fine-tune the model through transfer learning, so that the updated model is more suitable for the attack characteristics of the test set. Therefore, the detection accuracy of malicious attack types can be significantly improved.

Conclusion

We propose a flexible trust model named TLTrust to detect malicious nodes in underwater sensor networks and maintain network security. In this model, we cluster the sensor nodes using improved K-means algorithm. The cluster heads are responsible for collecting the location, interaction, data, and energy information of nodes as trust evidence. The key of TLTrust is to use a small amount of target domain data for transfer learning and update the model. The updated model can detect target attack, which significantly improves the ability of TLTrust to deal with new attacks. Extensive experiments on simulated datasets and NSLKDD datasets demonstrate the effectiveness of TLTrust.

In the future, we plan to explore the fusion mechanism of transfer learning, federated learning, and blockchain-based trust models to further improve the performance and reliability of trust assessment of malicious nodes in underwater sensor networks. First, we study how to combine transfer learning with federated learning, use transfer learning to extract reusable knowledge from other related fields or historical data, provide initial models or optimize parameters for federated learning, and achieve efficient distributed trust assessment model training. Secondly, we combine the blockchain-based trust model with the above fusion solution, and use the decentralized, tamper-proof, and traceable characteristics of blockchain to ensure the security and credibility of trust data interaction between nodes.

References

[1] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, 2005.
[2] L. A. Villas, A. Boukerche, R. B. D. Araujo, and A. A. F. Loureiro, "A reliable and data aggregation aware routing protocol for wireless sensor networks," in *Acm International Conference on Modeling*, 2009.
[3] Erol-Kantarci, Melike, Mouftah, Hussein, T., Oktug, and Sema, "Localization techniques for underwater acoustic sensor networks." *IEEE Communications Magazine*, 2010.
[4] X. Wei, H. Guo, and M. Wang, Xiaonan Qiu, "Reliable data collection techniques in underwater wireless sensor networks: A survey," *Communications surveys & tutorials*, vol. 24, no. 1, pp. 404–431, 2022.
[5] J. Jiang, X. Zhu, G. Han, M. Guizani, and L. Shu, "A dynamic trust evaluation and update mechanism based on c4.5 decision tree in underwater wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. PP, no. 99, pp. 1–1, 2020.
[6] Y. He, G. Han, J. Jiang, H. Wang, and M. Martinez-Garcia, "A trust update mechanism based on reinforcement learning in underwater acoustic sensor networks," *IEEE Transactions on Mobile Computing*, vol. PP, no. 99, pp. 1–1, 2020.
[7] C. Liu, J. Ye, F. An, and W. Jiang, "An adaptive trust evaluation model for detecting abnormal nodes in underwater acoustic sensor networks," *Sensors*, vol. 24, no. 9, 2024.

[8] A. Boukerch, L. Xu, and K. El-Khatib, "Trust-based security for wireless ad hoc and sensor networks," *Computer Communications*, vol. 30, no. 11-12, pp. 2413–2427, 2007.

[9] X. Li, F. Zhou, and J. Du, "Ldts: A lightweight and dependable trust system for clustered wireless sensor networks," *IEEE Transactions on Information Forensics & Security*, vol. 8, no. 6, pp. 924–935, 2013.

[10] J. Zhao, F. Huang, L. Liao, and Q. Zhang, "Blockchain-based trust management model for vehicular ad hoc networks," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 8118–8132, 2024.

[11] J. Qi, N. Zheng, M. Xu, X. Wang, and Y. Chen, "A multi-dimensional trust model for misbehavior detection in vehicular ad hoc networks," *Journal of Information Security and Applications*, vol. 76, p. 103528, 2023.

[12] N. Fan and C. Q. Wu, "On trust models for communication security in vehicular ad-hoc networks," *Ad hoc networks*, vol. 90, no. JUL., pp. 101 740.1–101 740.13, 2019.

[13] P. Sun and A. Boukerche, "Modeling and analysis of coverage degree and target detection for autonomous underwater vehicle-based system," *IEEE Transactions on Vehicular Technology*, 2018.

[14] J. Jiang, G. Han, L. Shu, S. Chan, and K. Wang, "A trust model based on cloud theory in underwater acoustic sensor networks," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 1, pp. 1–1, 2017.

[15] J. Du, G. Han, C. Lin, and M. Martinez-Garcia, "Itrust: An anomaly-resilient trust model based on isolation forest for underwater acoustic sensor networks," *IEEE Transactions on Mobile Computing*, vol. PP, no. 99, pp. 1–1, 2020.

[16] R. Zhu, A. Boukerche, P. Li, and Q. Yang, "A traffic-aware trust model based on edge computing for underwater wireless sensor networks," in *ICC 2024 - IEEE International Conference on Communications*, 2024.

[17] C. T. Nguyen, N. Van Huynh, N. H. Chu, Y. M. Saputra, D. T. Hoang, D. N. Nguyen, Q. V. Pham, D. Niyato, E. Dutkiewicz, and W. J. Hwang, "Transfer learning for future wireless networks: A comprehensive survey," *Proceedings of the IEEE*, vol. 110, no. 8, pp. 1073–1115, 2022.

[18] J. Li, Yunwei Ma, "Transfer learning based intrusion detection scheme for internet of vehicles," *Information Sciences: An International Journal*, vol. 547, no. 1, 2021.

[19] S. A. Khowaja, L. Nkenyereye, P. Khowaja, K. Dev, and D. Niyato, "Slip: Self-supervised learning based model inversion and poisoning detection-based zero-trust systems for vehicular networks," *IEEE Wireless Communications*, vol. 31, no. 2, pp. 50–57, 2024.

[20] J. Willinktricia, "Possibility-based trust for mobile wireless networks," *IEEE Transactions on Mobile Computing*, 2020.

[21] Y. Su, S. Mal, Z. Jin, X. Fu, Y. Li, and X. Liu, "A trust model for underwater acoustic sensor networks based on fast link quality assessment," in *Conference on Global Oceans : Singapore – U.S. Gulf Coast*, 2020.

[22] J. Guo, X. Li, Z. Liu, J. Ma, C. Yang, J. Zhang, and D. Wu, "Trove: A context-awareness trust model for vanets using reinforcement learning," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6647–6662, 2020.

[23] J. Suresh, J. M. Sahayaraj, B. Rajakumar, and N. Jayapandian, "Hybrid bayesian and modified grey promethee-al model-based trust estimation technique for thwarting malicious and selfish nodes in manets," *Wireless Networks (10220038)*, vol. 30, no. 3, 2024.

[24] J. Du, G. Han, C. Lin, and M. Martínez-García, "Ltrust: An adaptive trust model based on lstm for underwater acoustic sensor networks," *IEEE Transactions on Wireless Communications*, vol. 21, no. 9, pp. 7314–7328, 2022.

[25] J. Jiang, S. Hua, G. Han, A. Li, and C. Lin, "Controversy-adjudication-based trust management mechanism in the internet of underwater things," *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2603–2614, 2023.

[26] Y. Cho and G. Qu, "A hybrid trust model against insider packet drop attacks in wireless sensor networks," *sensors*, vol. 23, no. 9, 2023.

[27] I. Souissi, N. Ben Azzouna, and L. Ben Said, "A multi-level study of information trust models in wsn-assisted iot," *Computer Networks*, vol. 151, no. MAR.14, pp. 12–30, 2019.

[28] H. Chang, J. Feng, and C. Duan, "Reinforcement learning-based data forwarding in underwater wireless sensor networks with passive mobility," *Sensors*, vol. 19, no. 2, 2019.

[29] W. Gao, G. Zhang, W. Chen, and Y. Li, "A trust model based on subjective logic," in *International Conference on Internet Computing in Science and Engineering*, 2009, pp. 272–276.

[30] J. Jiang, G. Han, F. Wang, L. Shu, and M. Guizani, "An efficient distributed trust model for wireless sensor networks," *IEEE Transactions on Parallel & Distributed Systems*, vol. 26, no. 5, pp. 1228–1237, 2016.

[31] S. Kun, L. Fei, M. Niao-Xiong, and L. Zong-Tian, "Normal distribution based dynamical recommendation trust model," *Journal of Software*, vol. 23, no. 12, pp. 3130–3148, 2012.

[32] G. Han, J. Jiang, L. Shu, and M. Guizani, "An attack-resistant trust model based on multidimensional trust metrics in underwater acoustic sensor network," *IEEE transactions on mobile computing*, vol. 14, no. 12, pp. 2447–2459, 2015.

[33] G. Han, Y. He, J. Jiang, N. Wang, and J. A. Ansere, "A synergetic trust model based on svm in underwater acoustic sensor networks," *IEEE Transactions on Vehicular Technology*, vol. PP, no. 99, pp. 1–1, 2019.

[34] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014, pp. 3320–3328.

[35] L. Mingsheng, C. Yue, C. Zhangjie, W. Jianmin, and M. I. Jordan, "Transferable representation learning with deep adaptation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 3071–3085, 2018.

[36] S. Revathi and A. Malathi, "A detailed analysis on nsl-kdd dataset using various machine learning techniques for intrusion detection," *ESRSA Publications*, no. 12, 2013.

Zhenquan Qin received the B.S. and Ph.D.degrees in security engineering from the University of Science and Technology of China, in2002 and 2007, respectively. He is currently an Associate Professor with the School of Software,Dalian University of Technology. His research interests include UWSNs, industrial IoT, and Federated Learning.

Weicheng Meng is currently pursuing an M.S. degree in the School of Software at Dalian University of Technology, China. He received the B.S. degree in Software Engineering from Dalian University of Technology, China, in 2021. His research interests include fault detection within the Industrial Internet of Things, Federated Learning, and Underwater Wireless Sensor Networks (UWSNs).

Yuxin Cui received the M.S. degree in Software Engineering from Dalian University of Technology, China, in 2023. Her research interests include fault detection within the Industrial Internet of Things and Underwater Wireless Sensor Networks.

Bingxian Lu is currently an Associate Professor of the School of Computer Science and Engineering, Northeastern University, China. He received his B.S., M.E., and Ph.D. in 2012, 2014, and 2019 from Dalian University of Technology. His research interests include wireless networks, mobile computing, and pervasive computing applications. He is a member of the IEEE and the ACM.