TFL-DT: A Trust Evaluation Scheme for Federated Learning in Digital Twin for Mobile Networks

Jingjing Guo, Zhiquan Liu, Siyi Tian, Feiran Huang, *Member, IEEE*, Jiaxing Li, Xinghua Li, Kostromitin Konstantin Igorevich, and Jianfeng Ma, *Member, IEEE*

***Abstract*— Due to the distributed collaboration and privacy protection features, federated learning is a promising technology to perform the model training in virtual twins of Digital Twin for Mobile Networks (DTMN). In order to enhance the reliability of the model, it is always expected that the users involved in federated learning have trustworthy behaviors. Yet, available trust evaluation schemes for federated learning have the problems of considering simplex evaluation factor and using coarse-grained trust calculation method. In this paper, we propose a trust evaluation scheme for federated learning in DTMN, which takes direct trust evidence and recommended trust information into account. A user behavior model is designed based on multiple attributes to depict users' behavior in a fine-grained manner. Furthermore, the trust calculation methods for local trust value and recommended trust value of a user are proposed using the data of user behavior model as trust evidence. Several experiments were conducted to verify the effectiveness of the proposed scheme. The results show that the proposed method is able to evaluate the trust levels of users with different behavior patterns accurately. Moreover, it performs better in resisting attacks from users that alternately execute good and bad behaviors compared with state-of-the-art scheme. The code for the method proposed in this paper is available at: https://web.xidian.edu.cn/jjguo/en/code.html.**

***Index Terms*— Federated learning, digital twin, mobile networks, trust evaluation.**

Manuscript received 31 October 2022; revised 15 May 2023; accepted 4 August 2023. Date of publication 30 August 2023; date of current version 26 October 2023. This work was supported in part by the Foundation for Innovative Research Groups of the National Natural Science Foundation of China under Grant 62121001, in part by the Natural Science Basic Research Program of Shaanxi under Grant 2022JQ−603, in part by the National Natural Science Foundation of China under Grant 62272195 and Grant 61932010, in part by the Science and Technology Program of Guangzhou of China under Grant 202201010421, in part by the National Science Foundation for Distinguished Young Scholars of China under Grant 62125205, in part by the Fundamental Research Funds for the Central Universities under Grant ZYTS23161 and Grant 21618332, in part by the Key Research and Development Program of Shaanxi under Grant 2022GY−029, in part by the Russian Science Foundation (RSF) under Grant 22-71-10095, and in part by the 111 Project of China under Grant B16037. *(Corresponding author: Zhiquan Liu.)*

Jingjing Guo, Siyi Tian, Jiaxing Li, and Xinghua Li are with the School of Cyber Engineering, Xidian University, Xi'an 710071, China (e-mail: jjguo@xidian.edu.cn; 20181214403@stu.xidian.edu.cn; 22151214588@stu.xidian.edu.cn; xhli1@mail.xidian.edu.cn).

Zhiquan Liu and Feiran Huang are with the College of Cyber Security, Jinan University, Guangzhou 510632, China (e-mail: zqliu@jnu.edu.cn; huangfr@jnu.edu.cn).

Kostromitin Konstantin Igorevich is with the Department of Information Security, South Ural State University, 454080 Chelyabinsk, Russia (e-mail: kostromitinki@susu.ru).

Jianfeng Ma is with the State Key Laboratory of Integrated Services Network, Xidian University, Xi'an 710071, China (e-mail: jfma@mail.xidian.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSAC.2023.3310094.

Digital Object Identifier 10.1109/JSAC.2023.3310094

I. Introduction

With the commercial deployment of 5G, the rapid development of Internet of Things and the consequent new network services, the network scale is constantly expanding with the continuous increasing network load. Digital Twin (DT) technology offers great potential to bridge the gap between the data generation from mobile networks and the rapid and real-time data analysis requirement. A DT is an intelligent and constantly evolving system, that monitors, controls and optimizes the physical object throughout its life cycle. It is mainly composed of three parts, namely 1) a physical object; 2) a virtual twin; and 3) a mapping between the physical object and its virtual twin that enables the co-evolution of both physical and virtual sides. The virtual twin is an accurate digital replica of the corresponding physical object across multiple levels. The physical object could be a mobile device, a machine, a robot or an industrial process. Digital Twin for Mobile Network (DTMN) is a kind of Digital Twin Network (DTN), which is a many-to-many mapping network constructed by multiple one-to-one DTs. The physical objects in a DTMN could be various entities in a mobile network, e.g., smartphones, vehicle terminals, laptops and so on. In a DTMN, physical objects and their corresponding virtual twins can communicate, collaborate and share information to complete various tasks. During this process, DT modeling is the foundation to build the entire DTMN. Now, several modeling frameworks for DT have been proposed. One of the widely recognized modeling frameworks is a four-layer model including the data assurance layer, modeling calculation layer, DT function layer, and immersive experience layer [1]. In the modeling calculation layer, a model which gathers and processes the objects' information to model the objects plays a key role during the modeling process.

As the scale of a DTMN expands, the amount of data collected by physical objects will also increase. The congestion problem will arise if such large amount of data is directly transmitted through the communication system. Moreover, in some scenarios (e.g. the medical scenario), private information leakage and network security issues cannot be ignored. The raw data transmitted to the virtual twins may leak the privacy of the physical objects, which will make physical objects reluctant to provide their raw data to virtual twins. Therefore, it is a challenging problem to centrally build DTMN models at the modeling calculation layer, where the communication and privacy issues faced may hinder the development of DTMN.

As one of the most promising distributed machine learning paradigms with low delay and high privacy properties, federated learning is particularly suitable for constructing the DTMN model. During the whole learning process, none of the users' raw data or training process is exposed to others, including the aggregation server. Therefore, federated learning enables model training in virtual twins of a DTMN without collecting the physical objects' raw data together. A critical challenge posed to federated learning is the reliability of the global model. The majority of federated learning algorithms assume that users (participators) involved in the collaborative model training process are trusted. However, the practical situation is not consistent with this. Participators involved in a federated learning system usually have no trust relationship with each other. A participator may suffer from external attacks or be influenced by its limited resources, which will lead to unreliable behavior. Malicious users may disseminate false data or low-quality models to the aggregation server to adversely affect the modeling result in digital twins. In addition, physical objects with heterogeneous resources (communication, computation and data resources) may also behave differently, which may include unreliable or abnormal behavior. It is always expected that the participators involved in federated learning have trustworthy behaviors that can provide high-accuracy local models stably and timely according to predefined training rules.

Trust evaluation is an efficient way to measure the reliability of an entity's behavior. Based on the users' trust level, the aggregation server is able to select the local models submitted by users with a high trust level to update the global model so as to enhance the reliability of the global model [2]. Most of the existing trust evaluation methods for federated learning only take the interaction results (positive or negative) of a participator in each round of model training as the trust evaluation factor. However, the trust level of a participator is influenced by numerous factors. Moreover, the existing schemes lack fine-grained modeling of participators' behavior. In a federated learning system, the behavior data of participators is multidimensional and heterogeneous. However, most of the existing trust evaluation schemes adopt subjective logic model or simply use four arithmetic operations to calculate the trust value of a participator, which lacks the correlation analysis of the behavior data. Overall, the existing trust evaluation schemes for federated learning cannot comprehensively and accurately evaluate the trust level of a participator.

In order to solve the problems mentioned above and then enhance the reliability of the federated learning, we design a trust evaluation scheme for federated learning in a DTMN. The main contributions of this paper are summarized as follows.

- We propose a federated learning framework in DTMN, which is in charge of constructing the models in virtual twins of DTMN.
- To improve the reliability of the DTMN models trained by federated learning algorithm, we design a method to evaluate the trust level of users involved in the federated learning which takes multiple behavior attributes and temporal correlation of behavioral data of the participators into account. We implement the proposed trust evaluation method and prove its efficiency. The results show that the proposed method can accurately evaluate the trustworthiness of participators with different behavior patterns. Moreover, compared with the widely used subjective logic based trust evaluation scheme, the proposed scheme can detect more kinds of attack behaviors from participators.

The remainder of this paper is organized as follows. Section II reviews the related work of this paper, followed by the preliminaries of this paper in Section III. Section IV introduces the proposed scheme in detail. In Section V, we give the experiments to verify the effectiveness of our approach. Finally, Section VI concludes this paper and shows the future work.

II. Related Work

In this section, we review the latest related work of federated learning for DT and trust evaluation schemes for federated learning.

A. Federated Learning for Digital Twin

The digital twin paradigm emerges as one of the most promising technologies in mobile networks that can enable the near-instant communication and extreme-reliable mobile services [1]. While, the rising concern of data privacy and the collision between the massive data transmission requirement and the limited communication resource makes the conventional centralized AI algorithms are no longer suitable for the construction of digital twin model. Federated learning is a recent advance in distributed machine learning which has been applied in various fields [3], [4], [5]. Due to the distributed collaboration and privacy protection features, it is a promising framework to perform model training in DTMN. Lu et al. proposed an asynchronous federated learning framework for the DT-empowered Industrial IoT to achieve privacy protection in DTN [6]. They also proposed a blockchain empowered federated learning framework running in DTWN (Digital Twins for Wireless Networks) and DITEN (DIgital Twin Edge Networks) for collaborative computing [7]. Jiang et al. used federated learning technology to help resource-limited smart devices in constructing digital twin at the network edges belonging to different mobile network operators [8]. Sun et al. used federated learning in digital twin of air-ground networks where a drone works as the aggregator and the ground clients collaboratively train the model based on the network dynamics captured by digital twins [9]. They also studied dynamic digital twin and federated learning for air-ground networks, where an incentive scheme based on the Stackelberg game was designed for federated learning in order to motivate clients to collaboratively train the model [10]. Zhang et al. brought federated learning into the DT-enabled Industrial IoT system to achieve instant intelligence services for Industry 4.0 [11]. In these above-mentioned works, the model training takes place on end devices. Then, the obtained models are uploaded to corresponding servers (digital twins of the end devices) to accomplish the model aggregation. The resulting global model can be used to analyze the data of physical objects in real time and support the corresponding applications.

B. Trust Evaluation Schemes for Federated Learning

Trust evaluation is a useful means to measure the reliability of an object's behavior. When an object (a trustor) needs to evaluate the trust level of another object (a trustee), it can use the obtained information related to the trustee as input to a trust calculation function or a trust inference model. The calculation or inference result will be seen as the trust level of the trustee, which can be used as the basis of decision making.

Many trust models and trust evaluation methods have been proposed for various systems using different theories and techniques [12], [13], [14], [15], [16], [17]. In these schemes, most trust calculation functions use the weighted arithmetical operation or subjective logic model on the considered trust factors to calculate the trust value of a trustee. The trust factors are determined by the concrete network, therefore the existing concrete trust assessment methods proposed for specific application scenarios, such as ad hoc networks, social networks and so on, are not suitable for federated learning.

The research on trust evaluation method for federated learning is still in its infancy. To date, a few trust evaluation methods for federated learning have been proposed. Several researchers used the reputation value of a user to measure its trust level. Song et al. proposed reputation calculation method for users in federated learning using subjective logic model [18]. In this scheme, the reputation value of a user represents its trust level and local models uploaded by users with a high trust level will be assigned a greater weight in the global model aggregation process. The authors considered learning effects, the failure probability of packet transmission and dataset quality as the trust factors in their scheme. In order to find reliable users (participators) in federated learning, Kang et al. proposed a reputation evaluation method based on subjective logic model [19], [20]. In their scheme, the interaction timeline (recent or past interactions) and interaction effects (positive or negative interactions) are considered in the reputation calculation function. The final reputation of a user is the combination of the local reputation opinion of the aggregation server and the recommended reputation opinions from other learning task publishers. In order to identify trustworthy users to participate in federated learning tasks, Maslamani et al. proposed a reputation management mechanism based on deep reinforcement learning to evaluate the trustworthiness of a user [21]. In this scheme, the reputation calculation is also based on the subjective logic model. In some reputation calculation methods for users of a federated learning system, the reputation value of a user is calculated based on the extent to which its local model contributes to the global model [2], [22], [23], [24]. Gholami et al. proposed a trust evaluation scheme for users involved in federated learning to enhance the security of the federated learning [25]. In their scheme, the factor that determines the trust value of a user is the number of times up to now that the user's behavior has been benign and malicious. The model they used in the trust calculation function is the Beta distribution. Bao et al. proposed a trust evaluation scheme for federated learning [26]. In this scheme, the trust value of a participator is calculated based on the evaluation of the co-participators in model training and the model users.

From the above discussion, we can see that in available trust evaluation methods, the trust level of a user involved in federated learning is calculated mainly based on a single trust factor, the contribution of the user's local model to the global model or the interaction results (positive and negative) between the user and the aggregation server. They did not construct a fine-grained model to depict participators' behavior. Moreover, the behavioral data of participators is multi-dimensional and heterogeneous, while most existing trust evaluation schemes adopt a simple quantitative method to analyze those data, which lacks the correlation analysis of those data. Therefore, the existing trust evaluation models cannot comprehensively and accurately evaluate the trust level of a participator in a federated learning system.

Based on the above issues, we propose a trust evaluation scheme to calculate the trust level of users in a federated learning system in a fine-grained manner. In the proposed scheme, we combine the local (direct) trust value with recommended (indirect) trust value of a user to obtain its final trust level. We design a user's behavior model from multiple dimensions so as to evaluate the local trust value and recommended trust value of a user in a comprehensive way. In addition, we propose a temporal correlation analysis method to measure a user's behavioral stability and its familiarity with the aggregation server to more accurately describe its trust level.

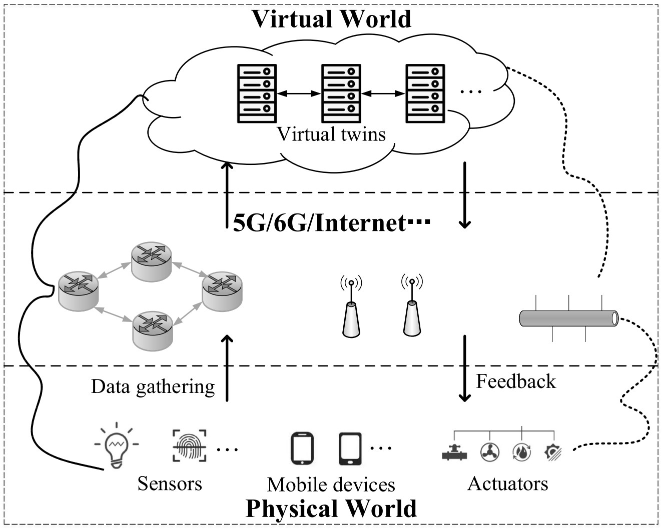
III. Preliminaries

In this section, we discuss the preliminaries and assumptions for our scheme.

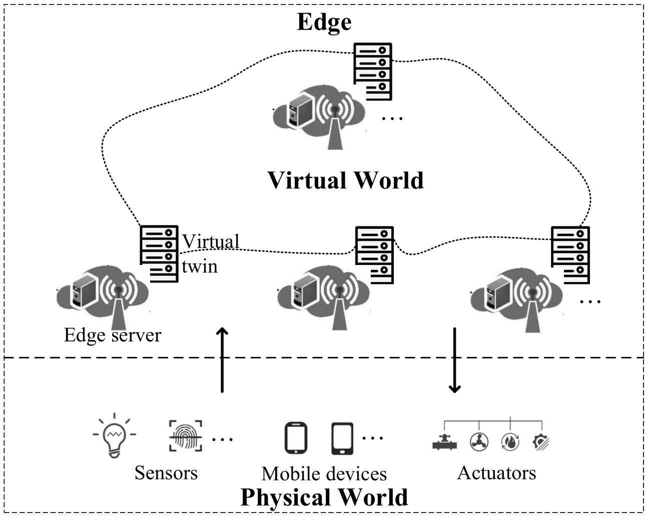
A. System of Digital Twin for Mobile Network

DTN is a many-to-many mapping network that is constructed by multiple one-to-one DTs. It uses advanced communication technologies to realize real-time information interaction between the physical object and its virtual twin, the virtual twin and other virtual twins, and the physical object and other physical objects. DTMN is a kind of DTN in which the physical world is made up of a wide variety of mobile devices connected by mobile networks.

Fig. 1 gives a brief infrastructure of the DTMN. Various mobile devices, such as sensors, cameras and smart phones, can connect with IoT gateways, Wi-Fi access points (APs) and base stations to form the physical world. Real-time data generated in the physical world can be transmitted to the virtual world via mobile networks. In this process, the physical object (device) is a mobile terminal, connected to the mobile access network through a mobile network access point, and finally connected to the virtual twin on the Internet. The virtual world is composed of several virtual twins corresponding to different physical objects. There are mainly two kinds of virtual twin deployment schemes. One is that all virtual twins are centrally deployed on a cloud server. Unlike the communications between physical objects that consume wireless spectrum resources and radio power, this virtual mode mainly



(a) Virtual Twin Centralized Deployment



(b) Virtual Twin Decentralized Deployment

Fig. 1. Framework of DTMN.

depends on DT servers' computing capability to model the data transmission behavior. In this case, the real-time data of the physical objects will be transmitted to the virtual twins first, and then a centralized AI algorithm will be run based on these data to accomplish the modeling process of the DTMN. The other one is to have virtual twins distributed deployed across multiple edge servers rather than centrally deployed on a single server, which is able to improve the security and efficiency of the model training process. Under the circumstance, model training in the virtual twin can be accomplished by federated learning, and the raw data of the physical objects will not be transmitted to the virtual twins. In this paper, we assume the virtual twins are deployed in the decentralized manner and the model training process is accomplished by federated learning.

The framework of a virtual twin is shown in Fig. 2. It mainly includes a data sharing module, a modeling module and a digital twin management module. The data sharing module is responsible for collecting and storing various network data, and providing data services and a unified interface to other modules. The modeling module completes data-based modeling and provides the obtained model for various network applications. The digital twin management module is in charge of the lifecycle management and visualization of the digital twin.

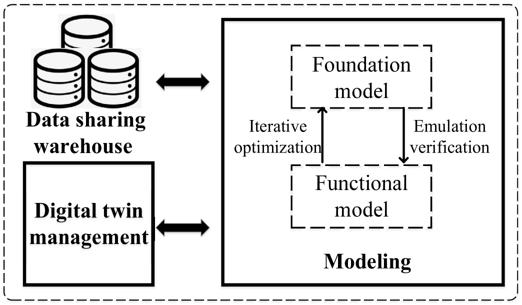


Fig. 2. Framework of a Virtual Twin.

B. Federated Learning System

Federated learning (FL) is a new machine learning paradigm allowing multiple users (FL users), such as mobile phones, sensors or drones, to train a model (e.g. a neural network) collaboratively, without exchanging their local raw data, thereby preserving data privacy to a great extent. There are mainly three types of entities in a federated learning system, namely user (participator), aggregation server and task publisher. The aim of the training process is to optimize a global loss function through minimizing the weighted average of every user's local loss function on its local data. The model training process usually consists of the following steps.

1 The task publisher publishes the federated learning task. Then, the aggregation server exposes a shared initial global model to users.
2 Each user trains its local model over its local data and the received global model. Then, it uploads the weights or gradients (i.e., local model update) of its latest local model to the aggregation server for updating the global model.
3 The aggregation server updates a new global model according to a predefined aggregation rule over the received local models. Then it sends the updated global model to users.
4 Users and the aggregation server will repeat steps 2 and 3 above until the model converges or reaches a predefined number of iterations.

Users with high-quality local models can lead to faster convergence of the local loss function and the global loss function. In addition, a reliable and high-quality communication environment can also decrease the training time. Consequently, trustworthy users with high-quality local models and reliable communication performance can significantly improve the learning efficiency of federated learning, e.g., with less training time and higher accuracy.

IV. Proposed Method

In this section, we give the proposed trust evaluation method for federated learning in detail. We assume federated learning is used to accomplish the model training in virtual twins of a DTMN. The proposed scheme can be used to solve the trustworthy user selection problem in a federated learning scenario, so as to enhance the security of federated learning and ensure the performance of the DTMN. The main notations used in this paper and their meanings are summarized in Table I.

TABLE I

MEANING OF NOTATIONS USED IN THE FOLLOWING SECTIONS

Notation	Meaning
N	Number of virtual twins in the system
M	Number of users (devices) in the system
u_i	Identity of the i-th user
vt_j	Identity of the j-th virtual twin
s_i	Number of users participate in modeling in vt_i
$bh_{i,k}^{c}$	Behavior record of u_i when it participates in the iterative learning on a given virtual twin for the k-th time under context c
BM_i^c	Behavior record list of u_i on a given virtual twin under context c
$lg_{i,c}$	Length of BM_i^c
G	Maximum value of $lg_{i,c}$
$rl_{k,c}^{i}$	Recommended trust of u_i from vt_k under context c
RL_i^c	Recommended trust set of u_i under context c
Q	Maximum length of RL_c^i
$T_c^j(i)$	Trust value of vt_j to u_i under context c
$local\_T_c^j(i)$	Local trust value of vt_j to u_i under context c
$hist\_T_c^j(i)$	History trust value of vt_j to u_i under context c
$curr\_T_c^j(i)$	Current interaction based trust value of vt_j to u_i under context c
$recom\_T_c^j(i)$	Recommended trust value of vt_j to u_i under context c
$\omega_{hist,i}$	Weight of history trust value
$\omega_{curr,i}$	Weight of current trust value
$\omega_{l,i}$	Weight of local trust value
$\omega_{r,i}$	Weight of recommended trust value
c_{af}^{i}	Abnormal factor of u_i
c_{df}^{i}	Delay factor of u_i
$acc_{i,k}^{j}$	Abnormal degree of u_i when it participates in the learning at vt_j for the k-th time
$delay_{i,k}^{j}$	Delay of u_i when it participates in the learning at vt_j for the k-th time
θ	Time forgetting factor when calculating the abnormal and delay factor
Δt_k	Interval between the happening time of $bh_{i,k}^{c}$ and the current time
φ	Time forgetting factor when calculating the history trust value
R_c^i	Reliable record set of u_i under context c
$r_{c,k}^{i}$	k-th reliable record of u_i under context c
S_c^i	Stability of $u_i^{\prime}s$ behavior under context c
$\Omega_{i,j}$	Familiarity of vt_j to u_i
ϕ	Familiarity regulatory factor
$h_{i,j}^{c}$	Number of interactions between u_i and vt_j under context c
$H_{i,j}^{c}$	Total number of interactions between u_i and other virtual twins under context c
ε	Threshold used to judge the reliability of a user in the current interaction
$g(h_{i,j}^{c})$	Adjustment function used in the calculation of current trust value
α_i, β_i, σ_i	Coefficients used to calculate $\omega_{r,i}$
$f_\omega(\sigma)$	Adjustment function used to calculate $\omega_{r,i}$

A. System Model

In this section, we will introduce the system model we considered in this paper. As shown in Fig. 3, in order to enhance the efficiency of operation and users' privacy, mobile network operators adopt digital twin technology to copy the real-time running of the real network and further simulate and analyze the data from the physical world by federated learning. There are mainly four modules in the considered system. They are task publisher, virtual twins deployed in edge servers, mobile devices and APs. The task publisher publishes model learning tasks based on the mobile network operator's specific requirements (e.g., deploying new services, resource allocation and so on) and makes a decision based on the obtained model. A huge number of mobile devices signed up to the mobile network operators are randomly distributed in the coverage areas of their corresponding access points (APs). These devices are able to connect to the APs located within their communication area. Here, each AP can be a Wi-Fi or a femtocell AP. Both the smart devices and the APs synchronize their data with the corresponding digital twins that are maintained by the associated edge servers. During the modeling process, mobile devices or APs train their local models using the data they owned. Only the parameters or the gradients of the obtained local models will be uploaded to the corresponding aggregation servers, which is deployed at virtual twins.

We assume there are N virtual twins in a DTMN in total, which is denoted as $VT = \{vt_1, vt_2, \cdots, vt_N\}$. The users (devices) set in the physical world is denoted as $MD = \{u_1, u_2, \cdots, u_M\}$. Here, MD includes all entities that perform local model training and upload the local model to the aggregation server, which may include mobile devices and APs. These users who participate in learning are also called participators. Each virtual twin has several corresponding users that participate in its model training process. Denote set $U_i = \{u_i^1, u_i^2, \cdots, u_i^{s_i}\}$ as the corresponding user set of vt_i, and we have $MD = \cup_{i \in [1,N]} U_i$. The workflow of federated learning in DTMN is shown as follows.

1 **System initialization.** First, a task publisher publishes a model learning task. Virtual twins associated with the task publisher will generate an initial global model related to this task. Then, the initial global model and the task will be broadcasted to users of the system.

2 **Local model training.** If a mobile device receives the learning task and the initial global model, it will train a local model based on its local data. Then, the local model will be uploaded to the associated AP. For devices with limited computation and energy resources, which may be insufficient for local models training and uploading, it will transmit local data to the corresponding AP to accomplish the local model training. These local models will be transmitted to the aggregation server of the modeling module in corresponding virtual twin.

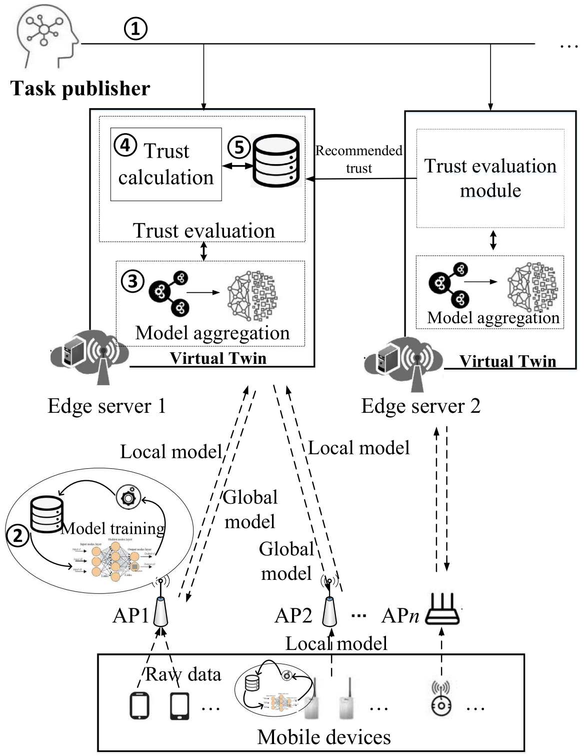


Fig. 3. Framework of Federated Learning in a DTMN.

3 **Trust-based global model aggregation.** The modelling module of each virtual twin that is deployed on edge servers will update the global model based on the local models it receives using a certain aggregation rule (such as FedAvg [27]). During the aggregation process, it will first select local models uploaded by trustworthy users (participators) based on their trust values, and then aggregate the selected local models to get the updated global model.

4 **Trust calculation.** After each round of global model aggregation, the aggregation server (virtual twin) will calculate the local trust value of participators in this iteration. Meanwhile, during the operation of the DTMN system, each virtual twin constantly receives the recommended trust value of users from other virtual twins. Based on the local trust value and the recommended trust value, virtual twins can obtain the global trust value of each user that participated in its model training process.

5 **Trust update.** virtual twins update the trust value information stored in the trust information storage database based on the latest trust calculation results.

Mobile devices, APs, and virtual twins will repeat steps 2-5 until either the aggregated global model converges or the number of iterations reaches a preset limit. After the training process, the task publisher is able to use the final obtained global model to make a decision.

B. Trust Behavior Model

In order to evaluate a user's trust level for participating in federated learning in a certain virtual twin, the virtual twin needs to record the user's behavior of participating in federated learning as comprehensive as possible. The trust evidence of user u_i recorded by vt_j under context c can be formalized as a quadruple shown in Eq. 1, where BM^c_i is an ordered set that records the behavior when u_i interacts with vt_j, RL^c_i is a set that records the recommended trust of u_i from other virtual twins under context c. Here, u_i interacts with vt_j means that u_i participates in the federated learning in vt_j. With the information recorded in BM^c_i and RL^c_i, vt_j can obtain the local trust profile (denoted as $L\_TP^c_i$) and nonlocal trust profile (denoted as $NL\_TP^c_i$) of u_i respectively, which will be used in the proposed trust calculation functions.

$$TE^c_{i,j} =< BM^c_i, RL^c_i, L\_TP^c_i, NL\_TP^c_i > \tag{1}$$

Context here refers to information related to the user's behavior, such as when the behavior occurred, the type of learning task being performed when the behavior occurred, and so on. In this paper, we only consider using behaviors and recommended trust information under the same context to complete the trust evaluation, so we do not give the specific content of the context here, which should be determined according to the specific application scenario.

The behavioral record set BM^c_i can be formalized as Eq. 2. We assume the maximum length of BM^c_i is G because of the limitation of the storage resource, and the length of BM^c_i is denoted as $lg_{i,c}$. Each element ($bh^c_{i,k}$) in BM^c_i records the behavior of u_i on the k-th interaction with vt_j. Eq. 3 shows the formalization of $bh^c_{i,k}$, where $acc_{i,k}$ is the abnormal degree of u_i's k-th interaction with vt_j, and $delay_{i,k}$ represents the delay of uploading the local model when u_i participates in the federated learning iteration for the k-th time with vt_j. We assume virtual twins have the ability to detect the anomaly of the local model uploaded by each user using available local model anomaly detection methods [28]. The more abnormal the detection result is, the smaller the value of $acc_{i,k}$ is. The value range of the abnormal degree ($acc_{i,k}$) is $[0, 1]$.

$$BM^c_i =< bh^c_{i,k} >, i \in [1, M], k \in [1, lg_{i,c}] \tag{2}$$

$$bh^c_{i,k} =< u_i, c, acc_{i,k}, delay_{i,k} > \tag{3}$$

Each element $rl^i_{k,c}$ within the recommended trust record set RL^c_i (shown in Eq.4) is a piece of message, including the recommended trust for u_i under context c sent by other virtual twins. Eq. 5 shows the formalization of $rl^i_{k,c}$, where vt_k is the identify of the recommender, $rt^c_{i,k}$ is the recommended trust of vt_k to u_i, $h^c_{i,k}$ represents the number of interactions between vt_k and u_i under c, and t is the sending time of $rl^i_{k,c}$. Due to the storage resources limitation and time forgetting factor, only the last Q pieces of recommended trust messages are recorded in RL^c_i. If RL^c_i is full when a new recommended message is received, the message with the earliest sending time will be deleted and the latest message will be inserted.

$$RL^c_i = \{rl^i_{k,c} | k \neq j\} \tag{4}$$

$$rl^i_{k,c} =< vt_k, rt^c_{i,k}, h^c_{i,k}, t > \tag{5}$$

Based on the information recorded in BM^c_i, vt_j is able to depict the local profile of u_i in terms of the reliability and stability of its behavior, which can be formalized as Eq. 6. Here, we use R^i_c and S^i_c to represent the reliability and stability of u_i's behavior under context c respectively. R_c^i is an ordered set shown in Eq. 7 that records the behavioral reliability of u_i under context c. The value of $r_{c,k}^i$ can be calculated by Eq. 8, where c_{af} and c_{df} are parameters used to describe the abnormal and delay conditions of u_i respectively.

$$L\_TP_i^c = (u_i, c, R_c^i, S_c^i) \tag{6}$$

$$R_c^i = \{r_{c,k}^i | k \in [1, lg_{i,c}]\} \tag{7}$$

$$\begin{aligned} r_{c,k}^i = sigmoid(c_{af,k}^i) \times c_{df,k}^i = \frac{1}{1 + e^{-c_{af,k}^i}} \\ \times c_{df,k}^i \end{aligned} \tag{8}$$

The abnormal factor of u_i, which is denoted as $c_{af,k}^i$, can be obtained by Eq. 9. As a rule of thumb, the probability of a user having abnormal behavior in the future is more closely related to its recent behavior. So, we use $\theta^{\Delta t_q}$ to show this property. Here, θ is a time-forgetting factor which falls within $[0, 1]$ and Δt_q is the interval between the happening time of u_i's q-th interaction with vt_j and the current time. Thus, the earlier the interaction happens, the lower proportion it takes in the calculation of the abnormal factor. In the same way, we calculate the value of delay factor $c_{df,k}^i$ by Eq. 10.

$$\begin{aligned} c_{af,k}^i &= f_a(\{acc_{i,q} | q \in [1, k]\}) \\ &= 1 - e^{-\sum_{q=1}^{k}(acc_{i,q} \times \theta^{\Delta t_q})} \end{aligned} \tag{9}$$

$$\begin{aligned} c_{df,k}^i &= f_d(\{delay_{i,q} | q \in [1, k\}) \\ &= 1 - e^{-\sum_{q=1}^{k}(delay_{i,q} \times \theta^{\Delta t_q})} \end{aligned} \tag{10}$$

Eq. 11 gives the way to calculate the behavior stability of user u_i under context c, which is the fourth element in $L\_TP_i^c$. From Eq. 11, we can see that the more similar the reliability of user u_i's behavior is between adjacent interactions in context c, the higher the stability of u_i's behavior will be. For users with high stability behaviors, there may be two kinds of behavior patterns. The first pattern is that the user performs poorly all the time (all values of $acc_{i,k}$ are small), and the other one is that the user performs well all the time (all values of $acc_{i,k}$ are large).

$$S_c^i = 1 - \frac{\sum_{k=1}^{lg_{i,c}}(|r_{c,k+1}^i - r_{c,k}^i|)}{lg_{i,c} - 1} \tag{11}$$

The trust profile of the recommended trust of u_i under context c ($NL\_TP_i^c$) is shown in Eq. 12, which is made up of two aspects. The first one is the total number of interactions of u_i with other virtual twins except vt_j under context c. The second one is the latest recommended trust value of u_i under context c ($recom\_T_c(i)$), the calculation method of which will be described in next section.

$$NL\_TP_i^c =< H_{i,j}^c, recom\_T_c(i) > \tag{12}$$

C. Trust Evaluation Method

In this section, we will introduce the proposed trust evaluation method. The goal of our method is to calculate the trust value of users participating in federated learning. Here, we consider that the higher the probability of delivering a high accurate local model to the aggregation server (deployed in virtual twin) in a timely manner, the higher the trust value of a user, and vice versa. In this way, the aggregation server can select users with high trust values to update the global model, so as to improve the training efficiency and accuracy of the global model as much as possible.

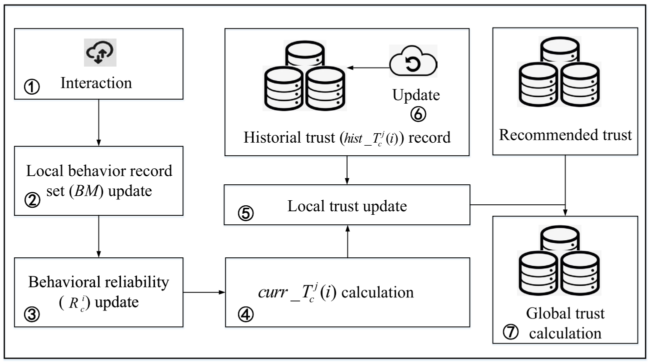

① Interaction
Local behavior record set (*BM*) update ②
Behavioral reliability (R_c^i) update ③
$curr\_T_c^j(i)$ calculation ④
⑤ Local trust update
Update ⑥
Historial trust ($hist\_T_c^j(i)$) record
Recommended trust
Global trust ⑦ calculation

Fig. 4. Workflow of the proposed trust evaluation scheme.

The workflow of the proposed trust evaluation method is shown in Fig. 4. Upon there is direct interaction between vt_j and u_i, vt_j will record u_i's behavior and update its behavioral record BM_i^c. Then, based on the latest behavioral record set, the behavioral reliability of u_i will be updated as well. Using the latest obtained reliability (the last element in R_c^i), vt_j is able to evaluate u_i's trust level based on the current iteration. Then, the local trust of u_i can be calculated. While computing the local trust, vt_j keeps receiving recommended trust information sent by other virtual twins. Once the recommended trust information is received, the recommended trust of the corresponding user will be updated immediately. Finally, with the latest recommended trust value of u_i and its local trust value, the global trust value of u_i will be obtained by Eq. 13. The calculation methods for the local trust value and the recommended trust value of u_i will be introduced in detail below.

Eq. 13 shows the function to calculate the global trust value of vt_j to u_i under context c. We can see $T_c^j(i)$ depends on the local trust value ($local\_T_c^j(i)$) and the recommended trust value ($recom\_T_c^j(i)$) of u_i under context c. $\omega_{l,i}$ and $\omega_{r,i}$ are the weights of u_i's local trust value and recommended trust value respectively. We have $\omega_{l,i} \geq 0, \omega_{r,i} \geq 0$ and $\omega_{l,i} + \omega_{r,i} = 1$. Here, $local\_T_c^j(i)$ is calculated based on the behavior of u_i when it interacts with vt_j, and $recom\_T_c^j(i)$ is obtained based on the recommended trust information of u_i from other virtual twins.

$$T_c^j(i) = \omega_{l,i} \cdot local\_T_c^j(i) + \omega_{r,i} \cdot recom\_T_c^j(i) \tag{13}$$

The reason we take both local interaction information and recommended information into account in the trust evaluation function is that we cannot guarantee that there is always available trusted local interaction information or recommended information. Therefore, we need coefficients $\omega_{l,i}$ and $\omega_{r,i}$ to determine whether and to what extend we can count on local trust information and recommended trust information in the trust calculation process. The value of $\omega_{l,i}$ and $\omega_{r,i}$ varies in different situations based on the relative quantity of local trust information and recommended trust information.

Eq. 14 shows the method for calculating $\omega_{l,i}$. After we get the value of $\omega_{l,i}$, we can also easily calculate the value of $\omega_{r,i}$ according to the above assumption, as shown in Eq. 15. We can see that the value of $\omega_{l,i}$ depends on two parameters α_i, β_i and a function $f_\omega(\sigma_i)$. The greater the value of α_i and $f_\omega(\sigma_i)$, the greater the weight of the local trust value. α_i reflects the proportion of the number of user u_i's direct interactions with vt_j to the average number of direct interactions between all users within U_j and vt_j. Similarly, β_i represents the ratio of the number of direct interactions between u_i and other virtual twins to the average number of direct interactions between all users within U_j and other virtual twins. σ_i indicates the relative quality of u_i's local trust information to its recommended trust information. We can see that when there is no direct interaction, the value of α_i and $\omega_{l,i}$ will be 0. Similarly, if there is no indirect interaction between u_i and other virtual twins, β_i will be 0 and the value of $\omega_{l,i}$ will be 1. Here, we use the relative number of direct interactions between u_i and vt_j and indirect interactions between u_i and other virtual twins to measure the weight of local trust and recommended trust, so that the weight of both can be measured more accurately when the numbers of direct interactions and indirect interactions are both large or small. The calculation method of parameters α_i, β_i, σ_i and function $f_\omega(\sigma_i)$ are shown in Eqs. 16, 17, 18 and 19 separately. The parameter ξ in Eq. 19 is a predefined threshold to determine the upper bound of $f_\omega(\sigma_i)$.

$$\omega_{l,i} = \frac{\alpha_i \cdot \frac{f_\omega(\sigma_i)}{f_\omega(\sigma_i)+1}}{\alpha_i \cdot \frac{f_\omega(\sigma_i)}{f_\omega(\sigma_i)+1} + \beta_i \cdot \frac{1}{f_\omega(\sigma_i)+1}} \tag{14}$$

$$\omega_{r,i} = 1 - \omega_l \tag{15}$$

$$\alpha_i = \frac{h^c_{i,j}}{\frac{1}{s_i} \cdot \sum_{u_k \in U_j} h^c_{k,j}} \tag{16}$$

$$\beta_i = \frac{H^c_{i,j}}{\frac{1}{s_i} \cdot \sum_{u_k \in U_j} H^c_{k,j}} \tag{17}$$

$$\sigma_i = \frac{h^c_{i,j}}{H^c_{i,j}} \tag{18}$$

$$f_\omega(\sigma) = \begin{cases} \xi, & if\ \sigma \geq \xi \\ \sigma, & otherwise \end{cases} \tag{19}$$

D. Local Trust Calculation

In this section, the calculation method of $local\_T^j_c(i)$ (local trust of vt_j to u_i under context c) is given in detail. After each direct interaction, vt_j will evaluate the trust value of u_i based on u_i's behavior during that interaction, which is denoted as $curr\_T^j_c(i)$. However, this evaluation result cannot fully reflect the trust level of u_i, so we must make a comprehensive evaluation by combining it with the trust level of u_i's behavior during other direct interactions (namely history interactions) with vt_j over a certain period of time, which is denoted as $hist\_T^j_c(i)$. Therefore, the value of $local\_T^j_c(i)$ is the weighted sum of $curr\_T^j_c(i)$ and $hist\_T^j_c(i)$, which is shown in Eq. 20.

$$local\_T^j_c(i) = \omega_{curr,i} \cdot curr\_T^j_c(i) + \omega_{hist,i} \cdot hist\_T^j_c(i) \tag{20}$$

Eq. 21 shows the calculation method of $hist\_T^j_c(i)$. It is the weighted sum of the behavioral reliability of u_i under context c. The weight φ_v ensures that the influence of the reliability of a user's behavior during a given period on its trustworthiness diminishes over time. The value of φ_v can be calculated by Eq. 22, where $ct_{i,v}$ is the time to calculate $r^i_{c,v}$.

$$hist\_T^j_c(i) = \Sigma^{lg_i}_{v=1}(r^i_{c,v} \times \frac{\varphi_v}{\Sigma^{lg_i}_{r=1}\varphi_r}) \tag{21}$$

$$\varphi_v = 2^{-(t-ct_{i,v})} \tag{22}$$

The value of $curr\_T^j_c(i)$ is calculated based on Eq. 23, which is the result of the most recent evaluation of the reliability of u_i (r^i_{c,lg_i}). If the reliability of u_i during its latest interaction with vt_j is less than 0.5, we assign $curr\_T^j_c(i)$ as 0. Otherwise, the value of $curr\_T^j_c(i)$ should be between the value of r^i_{c,lg_i} and 0.5 (uncertain). If vt_j has enough direct interaction experience with u_i (the number of interactions exceeds a certain threshold), the more times vt_j interacts with u_i, the more confident vt_j has in its reliability assessment result, so the closer the value of $curr\_T^j_c(i)$ is to r^i_{c,lg_i}. Otherwise, the more the value of $curr\_T^j_c(i)$ tends to be uncertain (0.5). The function $g(h^c_{i,j})$ in Eq. 23 is used to adjust the value of $curr\_T^j_c(i)$, which can be obtained by Eq. 24. From Eq. 24 we can see that the smaller the value of λ, the slower the value of $\lambda h^2 + \frac{1}{2}$ goes to 1. It ensures the value of $curr\_T^j_c(i)$ grows slowly and falls fast, which is consistent with the nature that trust value is difficult to increase and easy to decrease. Eq. 25 gives the calculation method for parameter ε in Eq. 23.

$$curr\_T^j_c(i) = (0.5 + g(h^c_{i,j}) \cdot (r^i_{c,lg_i} - 0.5)) \cdot \varepsilon \tag{23}$$

$$g(h) = \begin{cases} \lambda \cdot h^2 + \frac{1}{2}, & if\ 0 \leq h \leq \frac{1}{\sqrt[2]{2\lambda}} \\ 1, & otherwise \end{cases} \tag{24}$$

$$\varepsilon = \begin{cases} 1, & if\ r^i_{c,lg_i} > 0.5 \\ 0, & otherwise \end{cases} \tag{25}$$

The weights of $curr\_T^j_c(i)$ and $hist\_T^j_c(i)$ are denoted as $\omega_{curr,i}$ and $\omega_{hist,i}$ separately and can be obtained by Eq. 26 and Eq. 27. The more familiar vt_j is with u_i and the more stable the u_i's behavior, the more confident vt_j has in its evaluation result of u_i's local trust value. So, the value of $\omega_{hist,i}$ is the product of $\Omega_{i,j}$ and S^i_c. Here, $\Omega_{i,j}$ represents the familiarity of vt_j to u_i. The greater the number of interactions between vt_j and u_i, the larger the value of $\Omega_{i,j}$, which means that vt_j is more familiar with u_i. Eq. 28 gives the way to calculate $\Omega_{i,j}$.

$$\omega_{curr,i} = 1 - \omega_{hist,i} \tag{26}$$

$$\omega_{hist,i} = \Omega_{i,j} \cdot S^i_c \tag{27}$$

$$\Omega_{i,j} = \begin{cases} 1 - \frac{1}{\sqrt[\phi]{e^{h^c_{ij}} - 1} + 1}, & if\ h^c_{ij} \geq 1 \\ 0, & otherwise \end{cases} \tag{28}$$

E. Recommended Trust Calculation

The recommended trust of u_i under context c can be calculated by Eq. 29. It is the weighted sum of the recommended trust values of other virtual twins to u_i, where $rt^c_{i,k}$ is the second element of $rl^i_{k,c}$ which is shown in Eq. 5. We can

TABLE II

MEANINGS AND VALUES OF PARAMETERS USED IN EXPERIMENTS

Parameter	Meaning	Value
N	Number of virtual twins in the system	100
M	Number of users in the system	100
λ	Used in the calculation of $curr\_T_c^j(i)$	0.0001
θ	Time forgetting factor	0.5
ϕ	Familiarity regulatory factor used in Eq. 28	10

see that the more direct interaction experience a recommender has with u_i under c, the more weight the recommended trust ($rt^c_{i,k}$) from this recommender occupies in the calculation of $recom\_T^j_c(i)$.

$$recom\_T^j_c(i) = \Sigma_{k=1}^{|RL^i_c|}(\frac{h^c_{i,k}}{H^c_{i,j}} \times rt^c_{i,k}) \tag{29}$$

V. NUMERICAL EXPERIMENTS

In this section, we will describe the simulations and analysis that were undertaken to verify the effectiveness of the proposed scheme.

A. Experiments Setup

All of the experiments were conducted on a desktop with an Intel(R) Core(TM) i5-10400 processor and a CPU of 160 GB memory. The programming platform is Python 3.8.8 and Anaconda 4.10.1. Table II gives the meanings and values of parameters used in the following experiments.

We assume there are 100 virtual twins (vt_1 to vt_{100}) and 100 users involved in the system. We mainly simulate the trust evaluation of vt_1 to the users who participated in its model training task. Other virtual twins (vt_2 to vt_{100}) provide recommended trust information about the users to vt_1. We assume the virtual twins are trusted that the recommended trust value provided by vt_j ($j \in [2, 100]$) to vt_1 is its actual evaluation result and there is no collusion between virtual twins. The initial trust values of all users are set to 0.5. Virtual twin considers users whose trust values fall into $[0.8, 1]$ as benign users. A user with a trust value of less than 0.4 will be considered as a malicious user. If the trust value of a user is within $[0.4, 0.8]$, the virtual twin will consider its as an uncertain user.

There are two types of users in the experiments, namely benign user and malicious user. In one round of model iteration, a user can show good performance, that is, provide a high-accuracy local model timely. It may also behave poorly, such as uploading abnormal local models and experiencing timeouts. A benign user always performs well. A malicious user may have different strategies to compromise the federated learning. All the possible behavior patterns of users are listed in Table III. Pattern 1 is the behavior pattern of benign users. Pattern 2 to pattern 6 are the behaviors of malicious users.

In order to measure the feasibility of the proposed method, we first analyze the evolution of trust values of users with different behaviors. Then, it is obvious that the trust value of a user is related to three parameters, namely the familiarity degree with virtual twin vt_1 ($\Omega_{i,1}$), the time forgetting factor (φ) when calculating the historical trust value, and the threshold ε. Therefore, we study the influence of these three parameters on the trust evolution of a user. Finally, we compare the performance of the proposed method and the subjective logic based trust evaluation method which is widely used in available trust evaluation schemes.

TABLE III

BEHAVIOR PATTERNS OF USERS CONSIDERED IN THE EXPERIMENTS

Behavior	Description
Pattern 1	Benign user. It always performs well when interacting with virtual twins.
Pattern 2	Malicious user. It always performs poor.
Pattern 3	Malicious user. It alternates between performing well 10 times and performing poorly 5 times during the interaction with virtual twins.
Pattern 4	Malicious user. It alternates between performing well 10 times and performing poorly 10 times during the interaction with virtual twins.
Pattern 5	Malicious user. It alternates between performing well 1 time and performing poorly 1 time during the interaction with virtual twins.
Pattern 6	Malicious user. It alternates between performing well 5 times and performing poorly 10 times during the interaction with virtual twins.

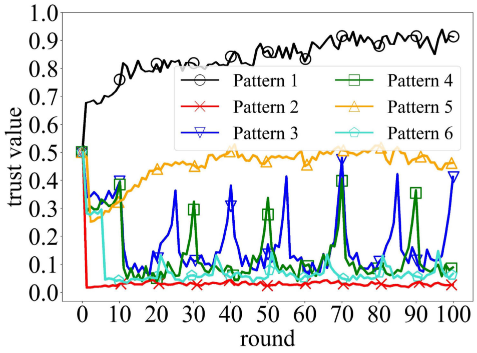


Fig. 5. Trust value evolution of users with different behavioral patterns.

B. Experimental Result

Fig. 5 shows the trust evolution of users with different behavior patterns. We can see that the trust value of a user with pattern 1 behavior rapidly rises to 0.8 after it interacts with vt_1 for a few times, and then slowly increases as the increasing number of interactions. For users with other behavioral patterns (pattern 2 to pattern 6), their trust values can drop below 0.4 in a short period of time. Therefore, the proposed method is able to evaluate the trust value accurately for users with different behavior patterns, so as to provide an effective decision support for the virtual twin. More specifically, the trust evolutions of malicious users with different behavior patterns are also different. A user with behavior pattern 2 always has the lowest trust value because it always performs poorly. For users who alternate between good and malicious behavior (pattern 3, pattern 4, pattern 5 and pattern 6), the higher the proportion of the number of good behaviors in an alternating cycle, the greater the trust value, except for pattern 5. Comparing behavior pattern 4 and pattern 5, although the time of good performance and bad performance are the same, the trust value of a user with pattern 5 is higher than that with pattern 4. Even so, the user

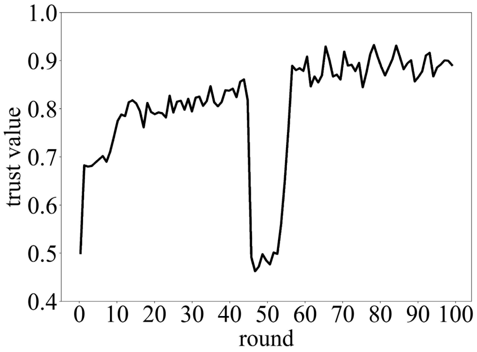


Fig. 6. Trust value evolution of a benign user in case of sudden failure.

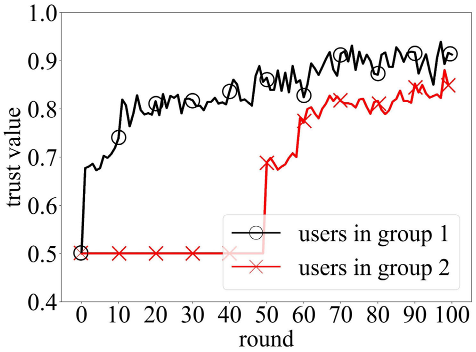


Fig. 7. Influence of the familiarity between the virtual twin and a user on the user's trust level.

with behavior pattern 5 will still be able to be recognized as a malicious user after a few times of interaction.

Fig. 6 shows the trust value evolution of a benign user in the case of a sudden failure during interaction with the aggregation server. We can see that the sudden failure of a benign user will lead to a sharp decline in its trust value. While, with the subsequent good interactions, its trust value rises slowly at first, and then after several successive rounds of good interactions, its trust value reaches the previous height soon. Therefore, if the trust of a benign user decreases due to occasional failures, the trust value can still return to the same level before the failure after several consecutive good interactions. During the whole interaction process, its trust value is never lower than 0.5, so it will not be judged as a malicious user by the virtual twin. It can be seen that the proposed scheme has good robustness.

In order to study the influence of the familiarity between the virtual twin and a user on the user's trust level, we assume all users in the system are benign users and we divided them into two groups equally. After the task publisher publishes a learning task, users within the first group participate in the learning from the first iteration, while users within the second group participate in the learning from the 50-th iteration. Under the above configuration, the familiarities between vt_1 and users within the two groups are different.

Fig. 7 gives the trust evolution of users within the two groups separately. We can see that the trust value of users within group 1 increases with the increasing number of iteration. After the 50-th iteration, users within group 1 have a higher familiarity with vt_1 and the trust value of them have reached 0.8. While the users within group 2 have no iteration with vt_1, so their trust value remains at the initial trust value of 0.5 and the familiarity with vt_1 is 0. From the 51-th round, all users participated in the learning iteration with different familiarities with vt_1. As the number of iterations increases, the familiarity of users within group 2 with vt_1 is getting closer to that of users within group 1. Thus, the trust value of users within group 2 is also getting closer and closer to that of users within group 1 which represents the real trust level of benign users.

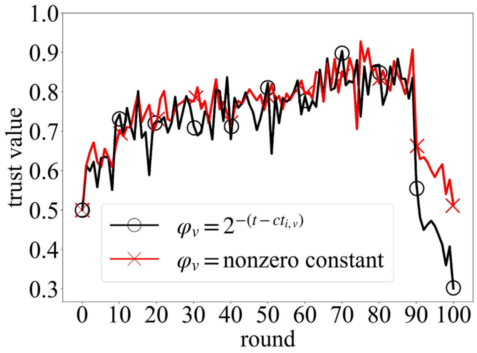
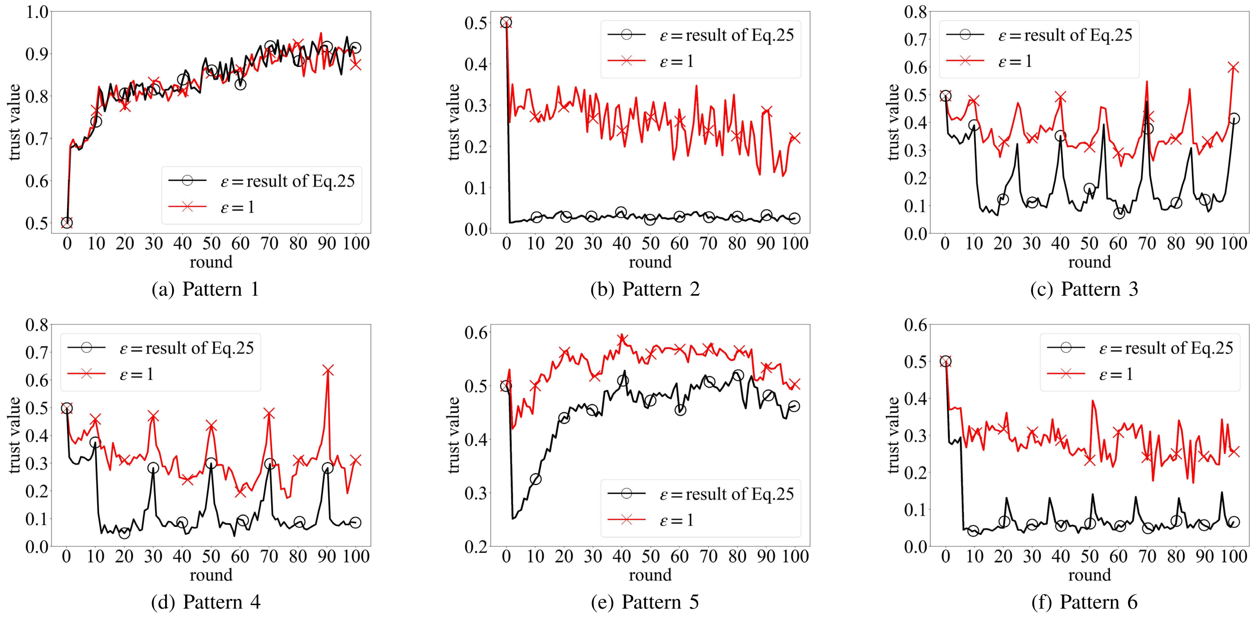


Fig. 8. Influence of the time forgetting factor on user's trust value evolution.

Fig. 8 gives the influence of the time forgetting factor (φ_ν) on a user's trust evolution. We set the value of φ_ν as $2^{-(t-ct_{i,v})}$ as shown in Eq. 21 and a nonzero constant separately. When φ_ν is a nonzero constant, it means that all historical behaviors' reliability has the same weight when calculating the historical trust value, no matter how long ago the reliability information was obtained. In this case, a user's historical trust value is the average value of all elements in the reliable record set of u_i (R_c^i). Then, we evaluate the trust value of a malicious user (called mu) who performs well in the first 90 rounds of iteration with vt_1 to accumulate a high trust value and then performs poorly in the following rounds. We can see with the time forgetting property, the trust value of mu could decrease below 0.4 quickly from the 91-th round, and the virtual twin could quickly find mu is a malicious user. If we do not consider the time decay factor, the trust value of mu will also decrease. While the decline is so small that even after executing 10 malicious behaviors, its trust value is still higher than 0.5, and it is difficult for the virtual twin to find malicious user mu in a short time.

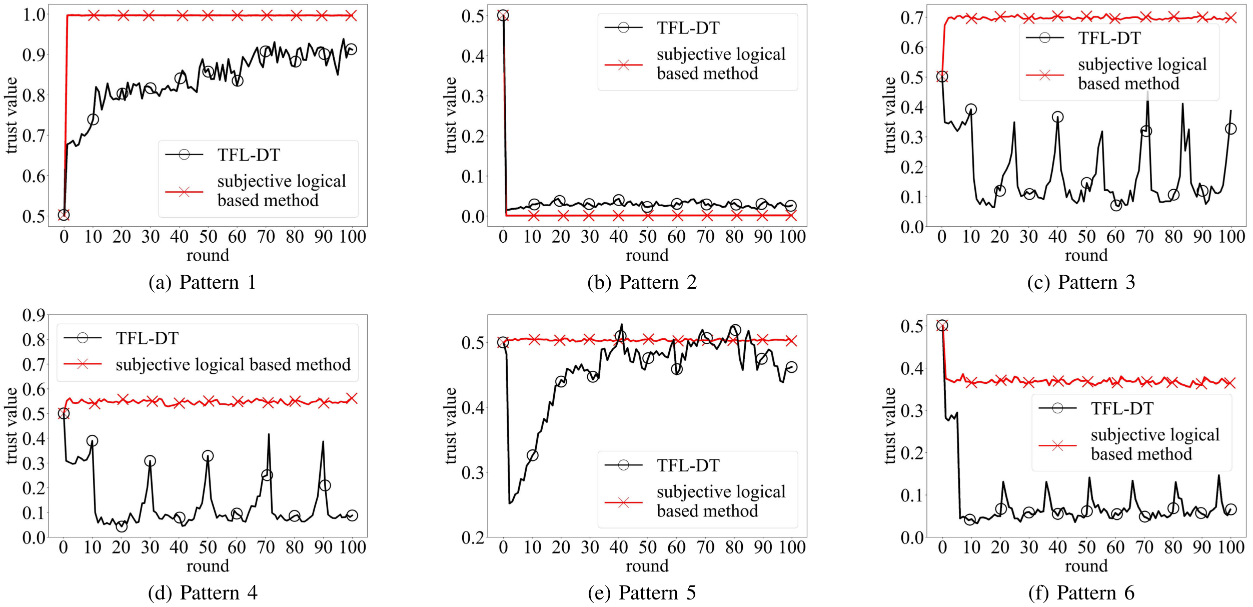
Fig. 9 shows the influence of ε on the trust evolution of users with different behavior patterns. In this experiment, we compare the trust value evolution of users with different behavior patterns when the value of ε is assigned as a constant 1 and the result of Eq. 25 separately. When ε equals 1, it can be deleted from Eq. 23, which means the ε will not influence the trust value of a user. We can see for a benign user with behavior pattern 1, there is almost no difference in its trust value when ε takes constant 1 and the result of Eq. 25, respectively. This is because the reliability of the user is always greater than 0.5, the result of Eq. 25 is always equal to 1. While for malicious users, whether it behaves good and bad alternatively (pattern 3, pattern 4, pattern 5 and pattern 6) or performs malicious behavior all the time (pattern 2), the existence of ε (ε equals the result of Eq. 25) can inhibit the



(a) Pattern 1 (b) Pattern 2 (c) Pattern 3

(d) Pattern 4 (e) Pattern 5 (f) Pattern 6

Fig. 9. Influence of the value of ε on the trust value evolution of users.



(a) Pattern 1 (b) Pattern 2 (c) Pattern 3

(d) Pattern 4 (e) Pattern 5 (f) Pattern 6

Fig. 10. Comparison between the proposed method and the subjective logic based method.

growth of trust value of these malicious users. This is because for malicious users, when its reliability is lower than 0.5 in a certain iteration, its trust value for this interaction will be 0, which will lower its local trust value.

We compare the performance of the proposed method and subjective logic based trust evaluation method and the comparison result is shown in Fig. 10. We can see that for users with behavior pattern 1 and pattern 2 (always performs well or poorly), both the proposed method and the subjective logic based method can accurately evaluate their trust values. For users with behavior pattern 3, pattern 4 and pattern 6, the proposed method always performs better than the subjective logic based method. For users with behavior pattern 5, during the first several rounds of iteration, the trust value obtained by the proposed method and the subjective logic based method are about the same (0.5). With the increasing number of interactions, the trust value calculated by the proposed scheme shows a downward trend, while the trust value obtained by the subjective logic based method remains unchanged. Therefore, the proposed scheme performs better in resisting attacks that alternately execute good and bad behaviors.

VI. Conclusion and Future Work

In this paper, we propose a trust evaluation scheme for federated learning in DTMN. The trust value of a user participating in the federated learning system depends on the direct trust of the trustor and the recommended trust information from other virtual twins. We design behavior model for users in a fine-grained and comprehensively manner. Based on a user's behavior model, its local trust value and recommended trust value can be obtained. In the proposed trust calculation functions, not only the factors like interaction results and time decay are considered, behavior properties including stability and reliability are also taken into account, which makes the evaluation results more accurate. In addition, an adaptive weight calculation method is proposed to determine the weights of local trust and recommended trust when calculating the global trust. A series of simulation experiments have done to verify the performance of our method. The results show that our scheme is able to evaluate the trust value of users with different behavioral patterns. Moreover, our scheme also performs better than its state-of-the-art counterparts in terms of detecting malicious users.

For the proposed scheme, we assume all the trust information about a user is obtained under the same context. While in real scenarios, the trust information, especially the recommended trust value of a user, may be evaluated under different contexts. In the future, we will study the trust evaluation problem in federated learning under different contexts.

Acknowledgment

The authors would like to appreciate the anonymous reviewers for their insightful suggestions to improve the quality of this article.

References

[1] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, “Digital twin in industry: State-of-the-art,” *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019.

[2] Q. Zhang, Q. Ding, J. Zhu, and D. Li, “Blockchain empowered reliable federated learning by worker selection: A trustworthy reputation evaluation method,” in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Mar. 2021, pp. 1–6.

[3] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong, “Federated learning over wireless networks: Optimization model design and analysis,” in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2019, pp. 1387–1395.

[4] S. Abdulrahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, “A survey on federated learning: The journey from centralized to distributed on-site learning and beyond,” *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5476–5497, Apr. 2021.

[5] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, “Federated learning: Challenges, methods, and future directions,” *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.

[6] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, “Communication-Efficient Federated learning for digital twin edge networks in industrial IoT,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5709–5718, Aug. 2021.

[7] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, “Low-latency federated learning and blockchain for edge association in digital twin empowered 6G networks,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 5098–5107, Jul. 2021.

[8] L. Jiang, H. Zheng, H. Tian, S. Xie, and Y. Zhang, “Cooperative federated learning and model update verification in blockchain-empowered digital Twin edge networks,” *IEEE Internet Things J.*, vol. 9, no. 13, pp. 11154–11167, Jul. 2022.

[9] W. Sun, N. Xu, L. Wang, H. Zhang, and Y. Zhang, “Dynamic digital twin and federated learning with incentives for air-ground networks,” *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 1, pp. 321–333, Jan. 2022.

[10] W. Sun, P. Wang, N. Xu, G. Wang, and Y. Zhang, “Dynamic digital twin and distributed incentives for resource allocation in aerial-assisted Internet of Vehicles,” *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5839–5852, Apr. 2022.

[11] J. Zhang, Y. Liu, X. Qin, and X. Xu, “Energy-efficient federated learning framework for digital twin-enabled industrial Internet of Things,” in *Proc. IEEE 32nd Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Helsinki, Sep. 2021, pp. 1160–1166.

[12] D. Wu, S. Si, S. Wu, and R. Wang, “Dynamic trust relationships aware data privacy protection in mobile crowd-sensing,” *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2958–2970, Aug. 2018.

[13] Z. Liu et al., “Lightweight trustworthy message exchange in unmanned aerial vehicle networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2144–2157, Feb. 2023.

[14] J. Guo et al., “TROVE: A context-awareness trust model for VANETs using reinforcement learning,” *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6647–6662, Jul. 2020.

[15] Y. Wang et al., “CATrust: Context-aware trust management for service-oriented ad hoc networks,” *IEEE Trans. Services Comput.*, vol. 11, no. 6, pp. 908–921, Nov. 2018.

[16] X. Meng and D. Liu, “GeTrust: A guarantee-based trust model in chord-based P2P networks,” *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 1, pp. 54–68, Jan. 2018.

[17] Z. Liu et al., “PPTM: A privacy-preserving trust management scheme for emergency message dissemination in space–air–ground-integrated vehicular networks,” *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5943–5956, Apr. 2022.

[18] Q. Song, S. Lei, W. Sun, and Y. Zhang, “Adaptive federated learning for digital twin driven industrial Internet of Things,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2021, pp. 1–6.

[19] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, “Reliable federated learning for mobile networks,” *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 72–80, Apr. 2020.

[20] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, “Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory,” *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10700–10714, Dec. 2019.

[21] N. Al-Maslamani, M. Abdallah, and B. S. Ciftler, “Secure federated learning for IoT using DRL-based trust mechanism,” in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, May 2022, pp. 1101–1106.

[22] Y. Wang and B. Kantarci, “A novel reputation-aware client selection scheme for federated learning within mobile environments,” in *Proc. IEEE 25th Int. Workshop Comput. Aided Model. Design Commun. Links Netw. (CAMAD*, Sep. 2020, pp. 1–6.

[23] J. Zhang, Y. Wu, and R. Pan, “Auction-based Ex-Post-Payment incentive mechanism design for horizontal federated learning with reputation and contribution measurement,” 2022, *arXiv:2201.02410.*

[24] J. Zhang, Y. Wu, and R. Pan, “Incentive mechanism for horizontal federated learning based on reputation and reverse auction,” in *Proc. Web Conf.*, Apr. 2021, pp. 947–956.

[25] A. Gholami, N. Torkzaban, and J. S. Baras, “Trusted decentralized federated learning,” in *Proc. IEEE 19th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Jan. 2022, pp. 1–6.

[26] X. Bao, C. Su, Y. Xiong, W. Huang, and Y. Hu, “FLChain: A blockchain for auditable federated learning with trust and incentive,” in *Proc. 5th Int. Conf. Big Data Comput. Commun. (BIGCOM)*, Aug. 2019, pp. 151–159.

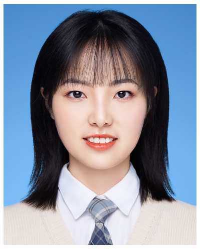
[27] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Proc. 20th Int. Conf. Artif. intell. Statist. (AISTATS)*, vol. 54, A. Singh and X. J. Zhu, Eds., Fort Lauderdale, USA, Apr. 2017, pp. 1273–1282.

[28] J. Guo et al., “ADFL: A poisoning attack defense framework for horizontal federated learning,” *IEEE Trans. Ind. Informat.*, vol. 18, no. 10, pp. 6526–6536, Oct. 2022.

Jingjing Guo received the M.Sc. and Ph.D. degrees from the School of Computer Science and Technology, Xidian University, Xi’an, China, in 2012 and 2015, respectively. She is currently an Associate Professor with the School of Cyber Engineering, Xidian University. Her research interests include trust management, access control, the Internet of Things security, and artificial intelligence security.

Zhiquan Liu received the B.S. degree from the School of Science, Xidian University, Xi'an, China, in 2012, and the Ph.D. degree from the School of Computer Science and Technology, Xidian University, in 2017.

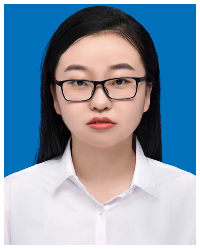
He is currently an Associate Professor with the College of Cyber Security, Jinan University, Guangzhou, China. His current research interests include trust management and privacy preservation in vehicular networks and UAV networks.



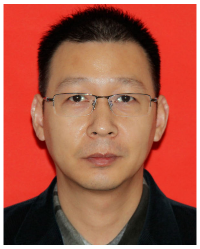
Siyi Tian received the B.S. degree from the School of Computer, Xi'an University of Posts and Telecommunications, Xi'an, China, in 2020. She is currently pursuing the master's degree with the School of Cyber Engineering, Xidian University. Her research interests include trust management and federated learning.

Feiran Huang (Member, IEEE) received the B.S. degree from the School of Physics and Electronics, Central South University, Changsha, China, in 2011, and the Ph.D. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2019.

He is currently an Associate Professor with the College of Cyber Security, Jinan University, Guangzhou, China. His current research interests include multi-modal data analysis, social media analysis, and network security.



Jiaxing Li received the B.S. degree from the School of Computer Science and Engineering, Xi'an University of Technology, Xi'an, China, in 2022. She is currently pursuing the master's degree with the School of Cyber Engineering, Xidian University. Her research interests include network security and federated learning.



Xinghua Li received the M.E. and Ph.D. degrees in computer science from Xidian University, Xi'an, China, in 2004 and 2007, respectively. He is currently a Full Professor and a Ph.D. Supervisor with Xidian University. His main research interests include wireless networks security, privacy protection, cloud computing, software-defined networks, and security protocol formal methodology.

Kostromitin Konstantin Igorevich received the B.S., M.S., and Ph.D. degrees from Chelyabinsk State University, Russia, in 2008, 2010, and 2013, respectively. He is currently a Researcher with the Department of Information Security, South Ural State University, Chelyabinsk, Russia. His current research interests include physics and the Internet of Things security.

Jianfeng Ma (Member, IEEE) received the M.E. and Ph.D. degrees in computer software and communications engineering from Xidian University, Xi'an, China, in 1988 and 1995, respectively. He is currently a Full Professor with Xidian University. His main research interests include information security, coding theory, and cryptography.