Reliable Federated Learning for Mobile Networks

Jiawen Kang, Zehui Xiong, Dusit Niyato, Yuze Zou, Yang Zhang, and Mohsen Guizani

Abstract

Federated learning, as a promising machine learning approach, has emerged to leverage a distributed personalized dataset from a number of nodes, for example, mobile devices, to improve performance while simultaneously providing privacy preservation for mobile users. In federated learning, training data is widely distributed and maintained on the mobile devices as workers. A central aggregator updates a global model by collecting local updates from mobile devices using their local training data to train the global model in each iteration. However, unreliable data may be uploaded by the mobile devices (i.e., workers), leading to frauds in tasks of federated learning. The workers may perform unreliable updates intentionally, for example, the data poisoning attack, or unintentionally, for example, low-quality data caused by energy constraints or high-speed mobility. Therefore, finding out trusted and reliable workers in federated learning tasks becomes critical. In this article, the concept of reputation is introduced as a metric. Based on this metric, a reliable worker selection scheme is proposed for federated learning tasks. Consortium blockchain is leveraged as a decentralized approach for achieving efficient reputation management of the workers without repudiation and tampering. By numerical analysis, the proposed approach is demonstrated to improve the reliability of federated learning tasks in mobile networks.

Introduction

Mobile devices, such as smart phones or vehicles, equipped with a variety of sensors, generate a huge amount and diverse types of user data [1]. Recently, for greatly improving mobile services and enabling smarter mobile applications, it is increasingly popular to utilize machine learning technologies to train models on such user data, for example, service recommendation and mobile healthcare [2]. However, a majority of machine learning technologies require a large amount of user data with sensitive privacy information to be aggregated in a central server for model training and analysis. This results in exorbitant communication and storage cost, and the mobile users are at risk of serious privacy leakage [3].

To address the privacy challenges, a decentralized machine learning paradigm called federated learning has been proposed to enable mobile devices (e.g., vehicles) to collaboratively train a global model required by a central aggregator (i.e., a task publisher) in a decentralized manner, without the need of centrally storing raw training data. In federated learning, the mobile devices download a global model from the central aggregator in each iteration, and then train and improve the current global model by using their local raw data. The mobile devices send the local model updates to the central aggregator. By aggregating these local model updates, the central aggregator generates a new global model for the next iteration. Both the mobile devices and the central aggregator repeat the above process until the global model achieves a certain accuracy [4]. This paradigm significantly reduces risks of sensitive privacy leakage by decoupling of model training from the need for direct access to the raw training data [3].

Although federated learning brings great benefits for mobile networks, it is still susceptible to various adversarial attacks in its primary stage. That is, during a federated learning process, data owners may mislead a global model by intentional or unintentional behaviors [5]. For intentional behaviors, an attacker can send malicious updates, that is, the poisoning attack, to affect the global model parameters resulting in the failure of current collaborative learning. The authors in [6] demonstrated the vulnerability of federated learning to sybil-based poisoning through experiments, and showed that existing defenses to such attacks are ineffective.

In addition, much more dynamic mobile networking environments indirectly result in some unintentional behaviors of data owners. The data owners may also indeliberately update low-quality models caused by high-speed mobility or energy limitation, thus adversely affecting federated learning. Therefore, it is of paramount importance for federated learning to defend against such intentionally and unintentionally unreliable local model updates.

In this article, we propose that reputation can be used to provide solutions to select reliable and trusted workers for the federated learning tasks. Existing studies show that reputation can reflect the rating of how reliable or trusted an entity is in certain activities according to its historical behaviors [1, 7]. Along with this direction, we are motivated to treat the reputation as a fair metric and design a reputation-based worker (i.e., data owner) selection

Jiawen Kang, Zehui Xiong (corresponding author), and Dusit Niyato are with Nanyang Technological University; Yang Zhang is with Wuhan University of Technology, and also with Nanyang Technological University; Yuze Zou is with Huazhong University of Science and Technology; Mohsen Guizani is with Qatar University.

Digital Object Identifier:
10.1109/MWC.001.1900119

1536-1284/20/$25.00 © 2020 IEEE

scheme for reliable federated learning. With the help of reputation, each task publisher selects only high-reputation workers to eliminate the impact from unreliable workers, thereby leading to high accuracy of the learning task [5]. Each task publisher calculates reputation opinions of every interacting worker through a subjective logic model. In the subjective logic model, the task publishers integrate their own opinions based on past interactions and recommended opinions from other task publishers [1, 7]. All the reputation opinions of the task publishers for the workers should be recorded in a non-repudiation and tamper-resistance manner for reliable reputation calculation.

To realize reliable reputation calculation as well as reputation management in federated learning, we design a consortium blockchain acting as a trusted and decentralized ledger to record and manage the data owners' reputation. The consortium blockchains are specific blockchains that perform the consensus process on pre-selected miners with mild cost in a short time [1, 7]. In mobile networks, edge nodes, for example, roadside units and base stations, are commonly deployed over the networks and easily reachable by task publishers and mobile devices, can be the pre-selected miners owing to having sufficient storage and computation resources [7]. The reputation values of the data owners are securely managed and stored on the consortium blockchain consisting of the edge nodes. The consortium blockchain is an efficient and practical blockchain technology running lightweight and fast consensus mechanisms on the miners.

The major contributions of this article are summarized as follows:

- To defend against unreliable model updates, reputation is introduced as a reliable metric to select trusted workers for reliable federated learning.
- A multi-weight subjective logic model is applied to design an efficient reputation calculation scheme according to both task publishers' interaction histories and recommended reputation opinions.
- To achieve secure reputation management, the reputation is managed in a decentralized manner by employing the consortium blockchain deployed at edge nodes.

Federated Learning and Its Vulnerabilities

Federated Learning and Its Mobile Applications

Traditional machine learning methods train models by using training data stored in a centralized server or dataset. But these methods face several critical challenges including single point of failure, sensitive data leakage, and huge overhead to collect and store the training data. To overcome these challenges, Google introduced a promising technique named federated learning that allows distributed mobile devices to collectively train a global model using their raw data while keeping these data locally stored on the mobile devices. Every mobile device computes a local update, for example via a distributed Stochastic Gradient Descent (SGD) algorithm, and uploads the local update, that is, weight parameters of current global model, to a central aggregator. The central aggregator, for example, a central server, collects all the local updates and calculates the average value of these local updates as a new global model. Federated learning significantly improves privacy protection of the mobile devices by blocking attack surfaces for direct access to the raw training data [3].

With the increasing popularity of federated learning, more and more mobile applications with federated learning have emerged. Some typical applications are listed as follows.

Google Keyboard: Gboard (https://ai.googleblog.com/2017/04/federated-learning-collaborative.html) as a virtual keyboard application from Google employs federated learning to improve language model quality, while simultaneously offering security and privacy protection for users by training input data locally.

Service Recommendation: Service providers collect searching and location histories from mobile devices to train entertainment and restaurant recommendation systems for enhancing service quality, but may cause serious privacy leakage risks for the mobile users. To ensure privacy preservation, the mobile devices join training recommendation models without concerning about privacy leakage by using federated learning.

Traffic Monitoring and Prediction: UberEATs (https://eng.uber.com/michelangelo/) leverages real-time traffic information to calculate estimated time of food delivery in a distributed learning manner. However, the distributed vehicles are not willing to share local traffic sensing data because of the concern of privacy leakage. To address this problem, the federated learning technique can be used to train prediction models without direct access to the personal data on the vehicles, which not only enhances traffic prediction accuracy but also protects data privacy of vehicles [8].

Mobile Healthcare: Health data from patients can be shared among hospitals or medical researchers to improve clinical services and healthcare analytics. Sharing such data with sensitive privacy information is facing serious challenges in mobile healthcare. Federated learning therefore is introduced to avoid centrally health data collection and collaboratively train models by using local health data in the mobile devices. NVIDIA Clara (https://blogs.nvidia.com/blog/2018/10/10/kings-college-london-nvidia-clara/) is used to deploy federated learning tasks to recommend the best treatment or automatic biomarker determination.

Security Challenges and Motivations

Although federated learning is promising to be applied in mobile environments, some critical challenges exist including reliable and trusted worker selection problems for model training. On the one hand, due to the openness and complexity of mobile network architectures, the data owners performing maliciously unreliable updates may result from: sensing data from malicious intent or tampered devices may include deceptive information, which is similar to false data injection attacks in smart grids [9]; the data can be arbitrarily manipulated when being transmitted through insecure communication channels [3, 9]. If a malicious data owner is selected to be a worker, the malicious worker may intentionally launch or collude with other workers to launch attacks, such as poisoning attacks [5]. For the poisoning attacks,

Health data from patients can be shared among hospitals or medical researchers to improve clinical services and healthcare analytics. Sharing such data with sensitive privacy information is facing serious challenges in mobile healthcare. Federated learning therefore is introduced to avoid centrally health data collection and collaboratively train models by using local health data in the mobile devices.

Inspired by the great potential of reputation in solving data quality problems in crowdsensing, we adopt the reputation metric to the selection of trusted and reliable workers for enhancing model training performance in federated learning. The reputation can reflect how well a worker has performed about model training, which can be measured from its training task completion history with the past behaviors of good or unreliable activities.

malicious workers deliberately tamper with a fraction of training data or inject poisonous data into the training datasets to increase the probability of misclassification, thus manipulating the results of training models [7]. On the other hand, the data owners may inadvertently provide unreliable local updates from low-quality raw data because of energy constraints or high-speed mobility. Both the intentional and unintentional behaviors can degrade the quality of the global model managed by a central aggregator,[1] hence affecting the final outputs of the global model [5]. Therefore, it is vitally important to design a reliable worker selection scheme during model training. Nevertheless, in federated learning, the following challenges for the worker selection need to be addressed.

No Reliable and Fair Metrics to Evaluate Workers: A majority of federated learning systems randomly select mobile devices to be the workers through verifiable random functions [5] or resource conditions [4]. However, the existing schemes cannot measure the trustworthiness level of workers to remove unreliable or untrusted workers.

No Efficient and Universal Worker Selection Schemes: The workers in existing federated learning schemes are selected either by a centralized authority, or by a decentralized method that all mobile devices join model training at will. As a result, the worker selection schemes are suffering from negative influence of unreliable or untrusted workers. For federated learning in mobile networks, it is difficult to design an efficient and universal worker selection scheme for identifying high-quality data contributors and malicious worker candidates.

No Timely Monitoring Methods for Workers: It is hard for the central aggregator/server (i.e., task publisher) to monitor the large-scale worker behaviors in real-time. The central aggregator without timely and dynamic monitoring methods cannot detect and remove the malicious or unreliable workers from the system. As a result, a malicious or unreliable worker may be selected to be a worker again for a new federated learning task because of the lack of time-accumulated metrics to evaluate the worker's historical performance and the synchronous information of malicious and unreliable worker lists.

To cope with the above challenges, we introduce a reliable metric and design a reliable worker selection scheme for federated learning in mobile networks.

Reputation Management for Reliable Federated Learning

Overview of Reputation Management in Crowdsensing

Regarding data quality problems, recent studies mainly focus on introducing reputation as a metric to identify whether a data provider is honest or malicious and evaluate its data quality [1, 7], especially in crowdsensing scenarios [10–12]. The reputation is used to select data providers who are more likely to provide high-quality data in crowdsensing.

An *et al.* [10] proposed a data provider selection scheme by using credit matching degree and trajectory matching degree for improving data quality in crowdsensing. The credit matching degree is calculated to measure the possibility that the worker submits high-quality data. Xie *et al.* [12] designed a reputation mechanism to prevent low-skilled workers and encourage high-skilled workers to participate in the crowdsensing tasks. The reputation values of workers are obtained and updated according to their historical contributions. Pouryazdan *et al.* [11] proposed a collaborative reputation scoring method based on statistical and vote-based user reputation scores to quantify the data trustworthiness, which improves platform utility and data trustworthiness in mobile crowdsensing.

Inspired by the great potential of reputation in solving data quality problems in crowdsensing, we adopt the reputation metric to the selection of trusted and reliable workers for enhancing model training performance in federated learning. The reputation can reflect how well a worker has performed about model training, which can be measured from its training task completion history with the past behaviors of good or unreliable activities [10]. With the help of reputation, task publishers select trusted and reliable workers to train the global model well, which can prevent the poisoning attacks launched by malicious workers and also remove unreliable data providers for obtaining high accuracy of the global model. Recent studies on federated learning [5, 13] have indicated that a central aggregator is vulnerable to security problems, for example, single point of failure [10]. In this article, to avoid the potential risks of central reputation calculation and management, we employ a decentralized reputation calculation method named subjective logic model [7], and a consortium blockchain with the properties of immutability and decentralization to realize secure reputation management [1]. Compared with centralized reputation management, consortium blockchain as a decentralized ledger can manage the reputation in a real-time and parallel manner without large computation overload. Similar to [7], the consortium blockchain performs the consensus process on pre-selected miners with mild cost in a short time, which is particularly suitable and practical for mobile networks because of lightweight and faster consensus agreement. More details about the reputation management and the subjective logic model for reputation calculation are presented below.

Reputation-Based Worker Selection Scheme with Consortium Blockchain

As shown in Fig. 1, the mobile devices collect local sensing data and generate various user data from mobile applications. Mobile applications with federated learning perform model training by using these data without the need of data aggregation for privacy preservation. The detailed steps about the federated learning are shown as follows [7].

Step 1: Task Publishment: Federated learning tasks from task publishers are first broadcast with specific data requirements (e.g., data sizes, types and time range). Mobile devices, that want to join one task and also satisfy the specific data requirements, will send a joining request with identity and data resource information back to one task publisher.

[1] We assume that the central aggregator is not compromised or malicious.

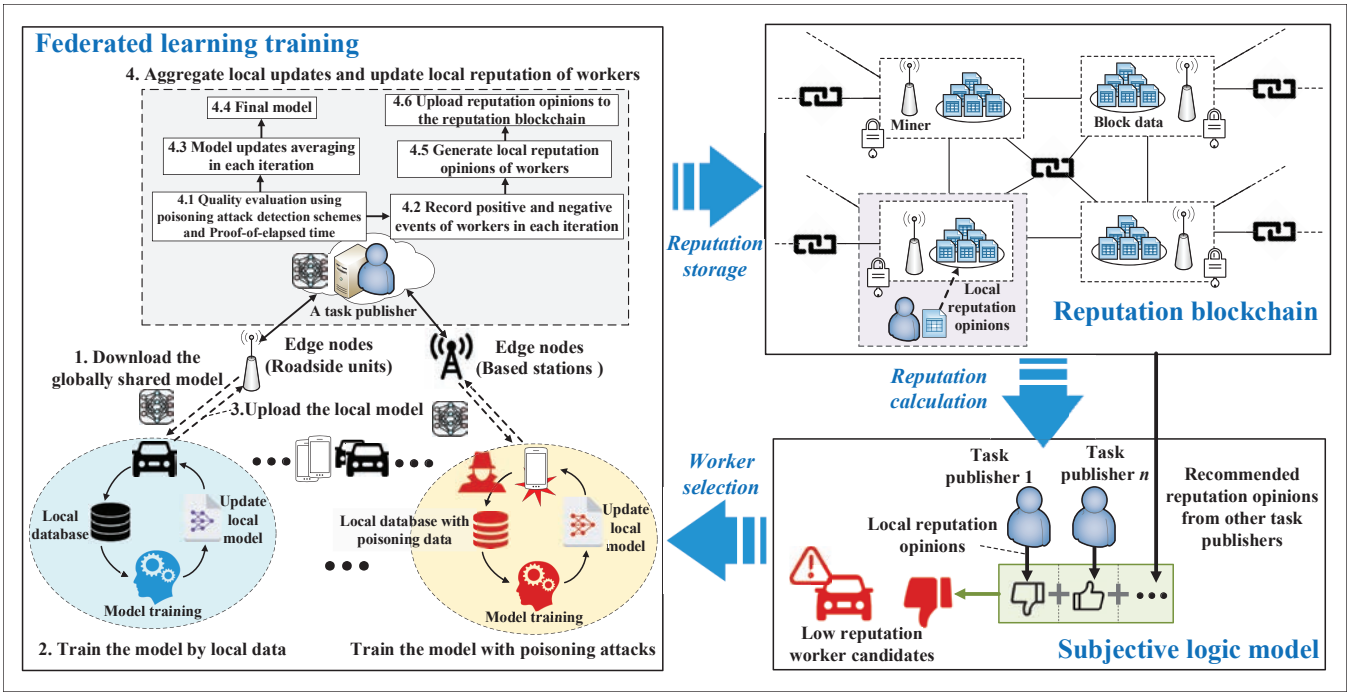

FIGURE 1. Consortium blockchain-based reputation management for secure federated learning.

Step 2: Worker Selection: The task publisher validates the identity and data resource information of the requesters, then the legal requesters can be the worker candidates. The task publisher starts to select its workers from the worker candidates according to their reputation values calculated by the subjective logic model in Section IV. The worker candidates with reputation values above a threshold can be selected as the workers. Here, the task publishers can set different reputation thresholds by themselves according to their own security level requirements. Without loss of generality, all task publishers can choose the same reputation threshold for current federated learning tasks. The reputation thresholds can also be adjusted by some statistical metrics based on the mean and standard deviation of reputation values of their worker candidates. The reputation values of the worker candidates are calculated according to local reputation opinions generated from direct interaction histories, and recommended reputation opinions of other task publishers stored on an open-access consortium blockchain named reputation blockchain. The reputation blockchain with decentralization and tamper-resistant natures is a public ledger established on the pre-selected miners, which records the reputation opinions into data blocks. These reputation opinions in the data blocks are transparent and tamper-proof evidence even if damage occurs [1, 7].

Step 3: Reputation Calculation: The task publisher utilizes the subjective logic model to generate local reputation opinions for the worker candidates based on interaction histories. The subjective logic model takes three weights about the past interactions into consideration to form the local opinions for each worker candidate. By combining the local reputation opinions with recommended reputation opinions, the task publishers generate a composite reputation as the final reputation for each worker candidate. The recommended reputation opinions can be downloaded from the reputation blockchain and obtained from the latest block data. More details about reputation calculation are depicted in Section IV.

Step 4: Federated Learning: We can adopt different optimization algorithms to train a federated learning model. In this article, we utilize the SGD algorithm[2] that iteratively selects a batch of training examples to calculate their gradients against the current model parameters and takes gradient steps in the direction that minimizes the loss function [5]. The task publisher first randomly chooses an initial SGD model (i.e., initial parameters) from predefined ranges as the global model. This initial SGD model is received by selected workers and the workers collaboratively train the global model by using their own local data. The workers generate local model updates and the corresponding local computation time and upload this information to the task publisher. The local computation time is used to verify the reliability and authenticity of local model updates by comparing the data size of the training data, in which the local computation time is proportional to the data size. To ensure the truthfulness of local computation time, we consider employing the proof of elapsed time method under Intel's SGX technology [13]. After validating the computation time, the task publisher can determine the "lazy" workers that have not trained all of the local data. Moreover, some poisoning attack detection schemes are carried out by the task publisher to identify the poisoning attacks and unreliable workers. Typical detection schemes include the Reject on Negative Influence (RONI) scheme [5] for Independent and Identically Distributed (IID) scenarios and the FoolsGold scheme [6] for non-IID scenarios. With the help of these schemes, the task publisher removes malicious updates from poisoning attacks and unreliable local model updates from the lazy or untrusted workers. Then, the task publisher generates a new global model

[2] For obtaining ε-accuracy, the SGD algorithm needs $O(\mu^2/\varepsilon)$ iterations on each worker, where μ is the condition number defined as the ratio of smoothness and strong convexity parameters of a strongly convex problem P.

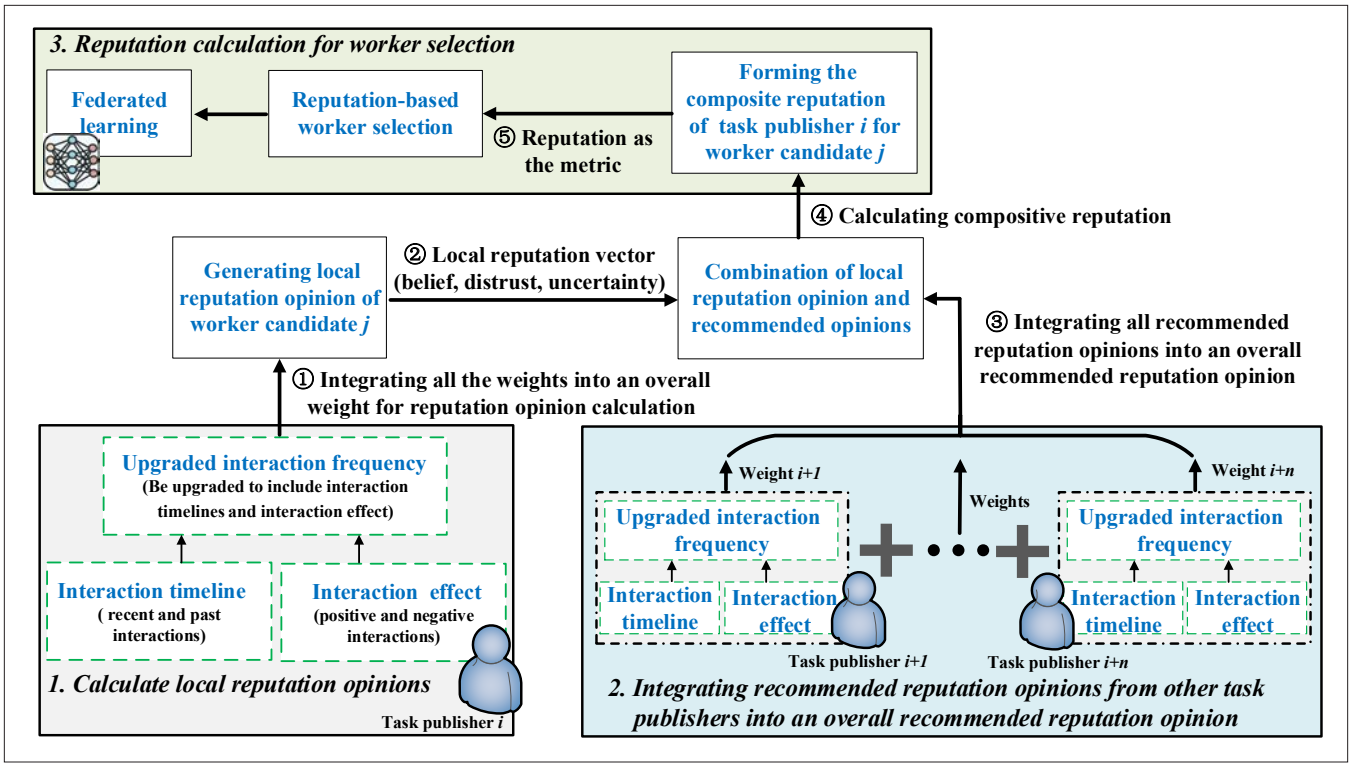

FIGURE 2. An overview of reputation calculation.

by calculating the average of the rest of the local model updates. Similar to [5], we consider that the distribution of data among mobile devices is sufficiently uniform to enable the RONI to work well in gradient validation. The task publisher sends the new global model to the selected workers for the next model iteration until the global model meets the predefined convergence conditions. The workers obtain rewards from the task publisher according to their data contribution and model training behaviors for the federated learning task [5, 13]. During the federated learning process, either the lazy and unreliable workers or the workers with poisoning attackers in each model iteration are recorded as a negative interaction by the task publisher. Finally, the task publisher generates local reputation opinions for the workers based on their performance in the federated learning task.

Step 5: Reputation Updating: To achieve secure reputation management, the task publisher's interaction histories and local reputation opinions for the workers with digital signatures are recorded as "transactions" and uploaded to the pre-selected miners in the reputation blockchain. These miners execute consensus algorithms, such as PBFT, and the reputation opinions and interaction histories are stored as a data block to be added into the reputation blockchain. After that, all task publishers can obtain the latest reputation opinions for a certain worker candidate from the reputation blockchain. Lastly, with the help of the reputation blockchain, the task publishers are able to select high-reputation workers for federated learning tasks.

Efficient Reputation Calculation Scheme

To assess the trustworthiness of a worker candidate, reputation opinions from task publishers should be collected and integrated into a composite reputation value of the worker candidate for secure worker selection. We therefore utilize the subjective logic model to calculate composite reputation values of worker candidates. Subjective logic is widely used to evaluate the trust level between different entities in the networks [1, 7], which is a specific framework of uncertain reasoning that uses a belief metric named "opinion" to represent a subjective belief about the world. The opinion is denoted by a tuple consisting of belief, distrust, and uncertainty to express the subjective belief of an entity or an event. For example, in vehicular networks, a task publisher performs a federated learning-based traffic prediction service with the help of vehicles. The task publisher's subjective belief for a vehicle increases if the publisher believes that the model updates provided by the vehicle are high-quality without the external impacts of unstable communication link between them, and vice versa.

All reputation opinions from task publishers are securely updated and stored in the decentralized reputation blockchain. Every task publisher selects workers by calculating composite reputation values according to its local reputation opinion and recommends reputation opinions. More details about reputation calculation are given as follows.

Subjective Logic Model for Reputation Calculation

During a federated learning task, for example, vehicular service recommendation, a task publisher interacts with different vehicles (i.e., workers) for training model corporately in each training iteration. By using poisoning attack detection schemes and the proof of elapsed time scheme (Step 4 above), the task publisher i treats a training iteration as a positive interaction event if the publisher perceives that the local model update from a worker j is reliable, and vice versa. The task

publisher records the numbers of positive and negative interaction events of all workers after a learning task, that is, α_j and β_j, and generates local reputation opinions for the workers. Each local reputation opinion is formally denoted as a local opinion vector consisting of belief degree $b_{i\to j}$, distrust degree $d_{i\to j}$, and uncertainty degree $u_{i\to j}$. The sum of these degrees is one.

Similar to [1, 7], the uncertainty degree is determined by the quality of the communication link between the worker j and the task publisher i, that is, the unsuccessful probability of data packet transmission (e.g., a worker unintentionally ignoring or dropping communication packets). The belief (distrust) degree is expressed by the positive (negative) interaction percentage of all interactions with good communication quality, denoted as

$$b_{i\to j} = \left(1-u_{i\to j}\right)\frac{\alpha_j}{\alpha_j+\beta_j}$$

and

$$d_{i\to j} = \left(1-u_{i\to j}\right)\frac{\beta_j}{\alpha_j+\beta_j}.$$

From the local opinion vector, a local reputation value is generated to represent the task publisher's expected belief that the worker provides high-quality local model updates during federated learning. The local reputation value is expressed as $T_{i\to j} = b_{i\to j} + \gamma u_{i\to j}$, where γ is the given constant indicating an effect level of the uncertainty for the reputation.

Multi-Weight Subjective Logic Model

Multi-weight subjective logic is an extension of subjective logic that takes different attributes of interaction events into consideration for more accurate and reliable reputation calculation [7]. As shown in Fig. 2, we consider the following three attributes as the weights to calculate reputation opinions.

Interaction Frequency: The interaction frequency represents the familiarity degree between a task publisher and a worker, which is expressed by the ratio of the number of times that the publisher interacts with the worker to the average number of times that the publisher interacts with other workers during a time window. The higher interaction frequency brings more prior knowledge about the worker to the publisher, hence leading to a higher local reputation opinion for the worker.

Interaction Timelines: Mobile devices acting as workers are vulnerable if there is no sufficient security protection. Therefore, the workers are not always reliable or trusted in federated learning. The trust level and the local reputation opinion of a worker for the same task publisher are changing over time. To evaluate the time effects on interactions, a time scale, for example, three days, is utilized to divide the interaction events into recent and past interactions. The recent interactions have a higher weight on the task publisher's reputation opinions.

Interaction Effects: Different interaction events have different effects on reputation opinions. We classify the interaction events into positive and negative interactions. The negative interactions, for example, the interactions with malicious workers or "lazy" workers (judged by Step 4 above), decrease the reputation of the workers, and vice versa. The positive interactions have a higher weight on the reputation opinion calculation.

Taking the interaction timelines and interaction effects into consideration, the interaction frequency is upgraded to contain the above two weights. Therefore, the interaction frequency is determined by both the two weights and the average number of times of interactions with other workers during a time window. After that, the upgraded interaction frequency is used to generate an overall weight for local and recommended reputation opinion calculation (as shown in ① and ②) [7].

Recommended Reputation Opinions

For a task publisher, the local reputation opinions from other task publishers are treated as recommended reputation opinions. These opinions are integrated into an overall recommended opinion according to the task publisher's weights for each recommended opinion (as shown in ③). The overall recommended opinion is also denoted as a recommended belief degree, a recommended distrust degree, and a recommended uncertainty degree. These degrees are calculated by weighted arithmetic mean of the belief degrees, distrust degrees and uncertainty degrees from other task publishers, respectively.

Combining Local Reputation Opinions with Recommended Reputation Opinions

When calculating the composite reputation value of a worker, the task publisher takes not only the overall recommended opinions, but also its own local reputation opinion into consideration to avoid collusion cheating from other task publishers (as shown in ④). The composite reputation of the task publisher to the worker is denoted as a final reputation opinion vector including three elements: the final belief degree, the final distrust degree, and the final uncertainty degree. The composite reputation value is determined by the final belief degree and the final uncertainty degree. More details about the reputation calculation can be found in [1, 7]. With the help of the reputation metric, high-reputation worker candidates can be selected as the worker for federated learning tasks (as shown in ⑤). These high-reputation workers will train local model honestly and maintain good behaviors in the federated learning tasks for earning more profits from the system. Therefore, the reputation-based worker selection scheme can defend against unreliable local model update from intentional or unintentional data providers, hence ensuring reliable federated learning in mobile networks.

To assess the trustworthiness of a worker candidate, reputation opinions from task publishers should be collected and integrated into a composite reputation value of the worker candidate for secure worker selection. We therefore utilize the subjective logic model to calculate composite reputation values of worker candidates.

Numerical Results

Simulation Setting

In order to evaluate the performance of the proposed schemes, we perform simulation on a well known digit classification dataset named MNIST by using Tensorflow 1.12.0 for a digit classification. This dataset consists of 60,000 training examples and 10,000 test examples [6]. We consider ten workers in this federated learning task including two malicious workers who launch poisoning attacks, four unreliable workers with low-quality data, and four well-behaved workers.

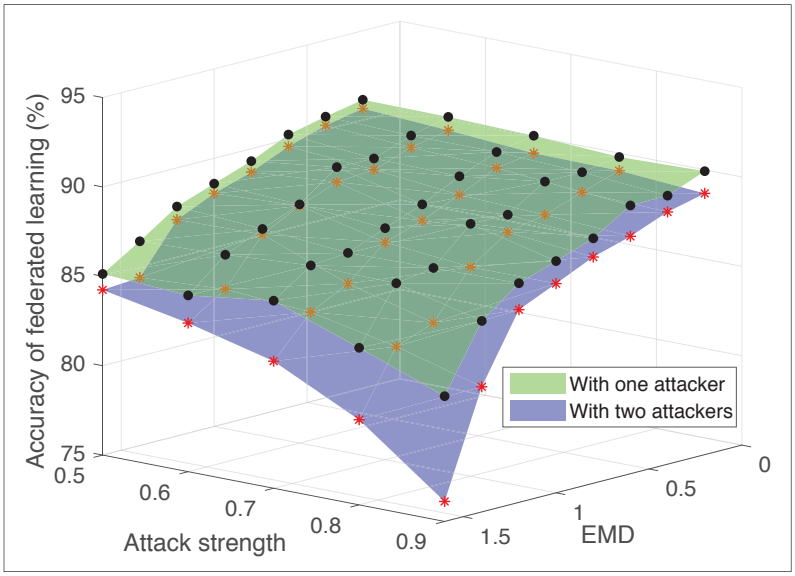

FIGURE 3. The accuracy comparison with respect to attack strengths and data quality levels.

The training sets of the well-behaved workers are randomly assigned but follow a uniform distribution over 10 classes. The data in each unreliable worker is only assigned a certain number of classes randomly. We employ the Earth Mover's Distance (EMD) as a metric to measure training data quality of the unreliable workers. Here, the EMD is expressed by the probability distance for a worker's data distribution compared with the actual distribution for the whole population [14]. For the malicious workers launching poisoning attacks, they randomly receive training data with 10 classes. However, the labels of some training examples are intentionally modified for misleading training.

The percentage of the modified training examples is used to indicate the attack strength. The workers use a batch of 32 randomly sampled training examples to produce a local SGD update, and every global model is trained with five synchronous iterations [6, 7]. Without loss of generality, the computation overhead of local model training is a constant overhead on each worker in the simulation. We establish the reputation blockchain system on the Hyperledger Fabric v1.4.0 and use the practical and efficient PBFT algorithm with mild overhead and latency as the consensus algorithm [1, 7].

For reputation calculation, the interaction frequency between task publishers and workers is from 20 to 40 federated learning tasks every week. The weight parameters of negative, positive, recent, and past interactions, and the time scale in the proposed Multi-weight Subjective Logic (MSL) scheme are referred to [1]. The unsuccessful transmission probability of data packets ranges from 0 percent to 40 percent, and the initial reputation of all the workers is 0.5.

We compare the proposed MSL scheme with a Traditional Subjective Logic (TSL) scheme from [7], and an Aggregated Trust Value (ATV) scheme referred to in [15]. In the ATV scheme, reputation is calculated by aggregating trust value offsets with different weights from the task publishers. The trust value offset is determined by the ratio of the difference between positive events and negative events to the total number of events.

Performance Results

Figure 3 shows the federated learning accuracy with respect to different poisoning attack strengths and EMDs. There are three factors that affect the learning accuracy: EMD, attacker number, and attack strength. An increase of any one of the above factors leads to a decrease of accuracy. The unreliable and untrusted workers with low-quality training data have negative impacts on the accuracy. For example, as shown in Fig. 3, the learning accuracy in the case with two attackers is only 76.12 percent, which is 7.7 percent lower than that with one attack when EMD is 1.6 and the attack strength is 0.9.

To illustrate the reputation change of a malicious or unreliable worker, we set that this worker performs well in the former six federated tasks on purpose to increase its reputation value. Then, the worker trains local models on its poisoning or unreliable examples for 30 task publishers with the probability of 0.8. As shown in Fig. 4, when the worker performs misbehaviors, its reputation begins to decrease in the MSL, TSL and ATV schemes, but is still linearly increasing in the scheme without reputation defenses. Due to considering the interaction effects, frequency and timeline, the reputation of the MSL scheme has a sharper and larger decrease than those in the ATV and TSL schemes in a short time. Moreover, the reputation of the ATV scheme drops faster than that of the MSL scheme after 12 iterations because the ATV scheme merely focuses on the interaction effects when calculating trust value offsets.

Figure 5 shows the impact of reputation thresholds of successful detection on the accuracy of a federated learning task (EMD = 1.6, attack strength = 0.9). If a worker's calculated reputation is below the given reputation threshold, the worker will be treated as a malicious or unreliable worker. Figure 5 illustrates that the higher reputation threshold brings a higher federated learning accuracy. Although the accuracy of the MSL scheme is lower than that of the ATV scheme under lower reputation thresholds, the MSL scheme has the same performance as that of the ATV scheme when the reputation is higher than 0.35. The reason is that the ATV scheme is sensitive to current negative events but ignores the well-behaved histories for good worker candidates with unintentional mistakes. This can result in false-positive errors and partial reputation calculation to reduce the incentive of the worker candidates. The TSL, MSL and ATV schemes achieve the same performance when the reputation threshold is above 0.45. The reason is that the malicious and unreliable workers are easier to be detected and hence removed in the case of high EMD and attack strength. In summary, the MSL scheme can achieve a more accurate and fair reputation calculation, thereby leading to a more reliable worker selection in federated learning.

Conclusion and Future Directions

In this article, we addressed worker selection issues to ensure reliable federated learning in mobile networks. A reputation-based scheme was designed to select reliable and trusted workers. For efficient and secure reputation man-

agement, we calculated workers' reputation by using a multi-weight subjective logic model, and employed consortium blockchain to manage the reputation with tamper resistance and non-repudiation in a decentralized manner. Numerical results showed that our schemes can bring reliable federated learning to mobile networks. There are several possible directions that are worth being studied:

- Due to model update validation limitation of the RONI scheme in non-IID settings, more accurate and efficient validation schemes for non-IID datasets should be designed to improve the detection performance of poisoning attacks in the proposed worker selection schemes.
- Considering the high overhead of a large number of workers will join in federated learning, efficient schemes for optimizing the number of workers are worth investigation in order to balance learning performance and resource cost.
- It still remains to be an open issue on how to dynamically optimize the reputation threshold to minimize negative effects from malicious workers, for example, by using advanced machine leaning methods.

Acknowledgment

This work was supported in part by Singapore NRF National Satellite of Excellence, Design Science and Technology for Secure Critical Infrastructure NSoE DeST-SCI2019-0007; A*STAR-NTU-SUTD Joint Research Grant Call on Artificial Intelligence for the Future of Manufacturing RGANS1906, WASP/NTU M4082187 (4080); Singapore MOE Tier 1 2017-T1-002-007 RG122/17, MOE Tier 2 MOE2014-T2-2-015 ARC4/15, Singapore NRF2015-NRF-ISF001-2277, and Singapore EMA Energy Resilience NRF2017EWT-EP003-041; and the National Natural Science Foundation of China under Grant 61601336.

References

[1] J. Kang *et al.*, "Towards Secure Blockchain-Enabled Internet of Vehicles: Optimizing Consensus Management Using Reputation and Contract Theory," *IEEE Trans. Vehicular Technology*, vol. 68, no. 3, Mar. 2019, pp. 2906–20.
[2] L. Cui *et al.*, "A Survey on Application of Machine Learning for Internet of Things," *Int'l. J. Machine Learning and Cybernetics*, 2018, pp. 1–19.
[3] X. Zhu *et al.*, "Blockchain-Based Privacy Preserving Deep Learning," *Proc. Int'l. Conf. Information Security and Cryptology*, Springer, Cham, 2018, pp. 370–83.
[4] T. T. Anh *et al.*, "Efficient Training Management for Mobile Crowdmachine Learning: A Deep Reinforcement Learning Approach," *IEEE Wireless Commun. Letters*, vol. 8, no. 5, Oct. 2019, pp. 1345–48.
[5] M. Shayan *et al.*, "Biscotti: A Ledger for Private and Secure Peer-to-Peer Machine Learning," 2018; available: https://arxiv.org/abs/1811. 09904.
[6] C. Fung *et al.*, "Mitigating Sybils in Federated Learning Poisoning," 2018; available: https://arxiv.org/abs/1808.04866.
[7] J. Kang *et al.*, "Incentive Mechanism for Reliable Federated Learning: A Joint Optimization Approach to Combining Reputation and Contract Theory," *IEEE Internet of Things J.*, vol. 6, no. 6, Dec. 2019, pp. 10700–714.
[8] L. Liang *et al.*, "Towards Intelligent Vehicular Networks: A Machine Learning Framework," *IEEE Internet of Things J.*, vol. 6, no. 1, 2019, pp. 124–35.
[9] P. Zhuang *et al.*, "False Data Injection Attacks Against State Estimation in Multiphase and Unbalanced Smart Distribution Systems," *IEEE Trans. Smart Grid*, vol. 10, no. 6, Nov. 2019, pp. 6000–13
[10] J. An *et al.*, "Crowdsensing Quality Control and Grading Evaluation Based on a Two-Consensus Blockchain," *IEEE Internet of Things J.*, vol. 6, no. 3, June 2019, pp. 4711–18.
[11] M. Pouryazdan *et al.*, "Quantifying User Reputation Scores, Data Trustworthiness, and User Incentives in Mobile Crowdsensing," *IEEE Access*, vol. 5, 2017, pp. 1382–97.

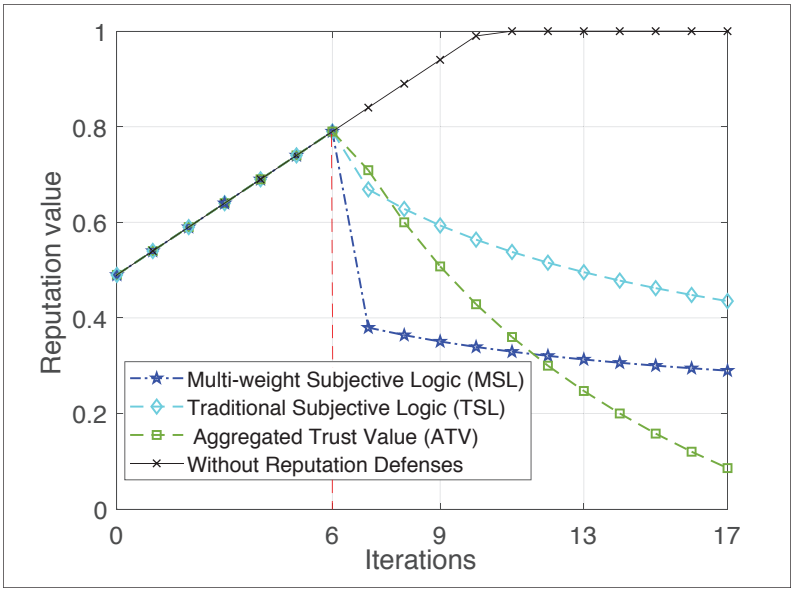

FIGURE 4. Reputation changes based on behavious of a worker.

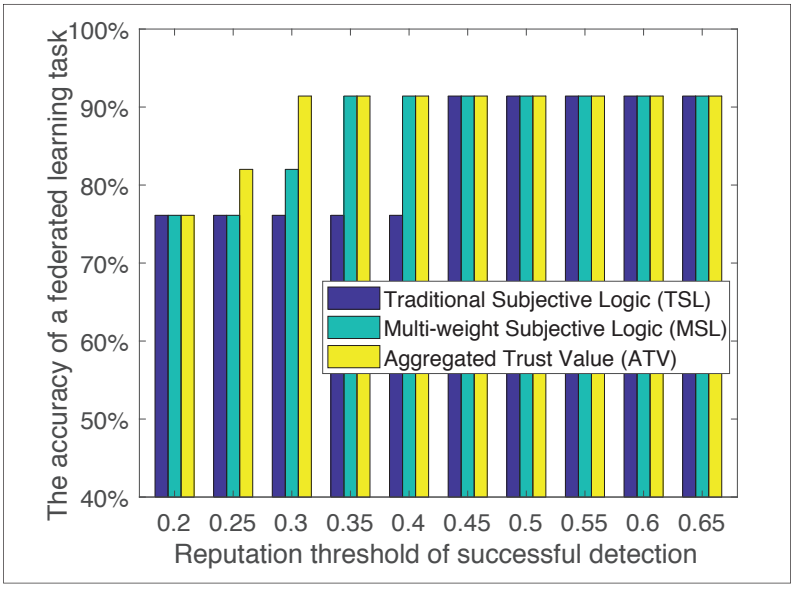

FIGURE 5. The impact of reputation thresholds of successful detection on the federated learning accuracy.

[12] H. Xie *et al.*, "Incentive and Reputation Mechanisms for Online Crowdsourcing Systems," *Proc. 23rd IEEE Int'l. Symposium Quality of Service (IWQoS)*, 2015, pp. 207–12.
[13] H. Kim *et al.*, "Blockchained On-Device Federated Learning," *IEEE Commun. Letts.*, in press, 2019. DOI: 10.1109/LCOMM. 2019.2921755
[14] Y. Zhao *et al.*, "Federated Learning with Non-Iid Data," 2018; available: https://arxiv.org/abs/1806.00582
[15] Z. Yang *et al.*, "Blockchain-Based Decentralized Trust Management in Vehicular Networks," *IEEE Internet of Things J.*, vol. 6, no. 2, April 2019, pp. 1495–1505.

Biographies

Jiawen Kang (kavinkang@ntu.edu.sg) received the M.S. degree from the Guangdong University of Technology, China, in 2015, and the Ph.D. degree from the same school in 2018. He is currently a postdoc at Nanyang Technological University, Singapore. His research interests mainly focus on blockchain, security and privacy protection in wireless communications and networking.

Zehui Xiong [S'17] (zxiong002@e.ntu.edu.sg) received his B.Eng. degree with the highest honors in telecommunication engineering from Huazhong University of Science and Technology, Wuhan, China, in 2016. He is currently working toward the Ph.D. degree in the School of Computer Science and Engi-

neering, Nanyang Technological University, Singapore. He was a visiting student at Princeton University in 2019. His research interests include network economics, wireless communications, blockchain, and deep reinforcement learning.

DUSIT NIYATO [M'09, SM'15, F'17] (dniyato@ntu.edu.sg) is currently a professor in the School of Computer Science and Engineering, Nanyang Technological University. He received his B.Eng. from King Mongkut's Institute of Technology Ladkrabang, Thailand, in 1999 and his Ph.D. in electrical and computer engineering from the University of Manitoba, Canada, in 2008. His research interests are in the area of energy harvesting for wireless communication, Internet of Things, and sensor networks.

YUZE ZOU (zouyuze@hust.edu.cn) received the B.E. degree in electronic information engineering (EIE) from Huazhong University of Science and Technology, Wuhan, China, in 2015, where he is currently pursuing the Ph.D. degree in the Department of EIE. His research interests include wireless power transfer, backscatter communications, and game theory and its applications in networked systems.

YANG ZHANG [M'11] (yangzhan2@whut.edu.cn) received the B.Eng. degree from Beihang University, and the Ph.D. degree from Nanyang Technological University, Singapore. He is currently an associate professor at Wuhan University of Technology, China, and a research fellow in Nanyang Technological University, Singapore. His current research interests include market-oriented modeling for network resource allocations, multiple objective optimization, and deep reinforcement learning.

MOHSEN GUIZANI [M'89, SM'99, F'09] (mguizani@ieee.org) received all of his degrees from Syracuse University, New York, in 1984, 1986, 1987, and 1990, respectively. He is currently with the Computer Science and Engineering Department, College of Engineering, Qatar University. He serves on the Editorial Boards of several international technical journals, and is the founder and Editor-in-Chief of *Wireless Communications and Mobile Computing* (Wiley). His research interests include wireless communications, mobile computing, computer networks, IoT, security, and smart grid.