Value-Aware Collaborative Data Pricing for Federated Learning in Vehicular Networks

Yilong Hui[1], Jie Hu[1], Xiao Xiao[1](✉), Nan Cheng[1], and Tom H. Luan[2]

[1] State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China
xiaoxiao@xidian.edu.cn

[2] School of Cyber Engineering, Xidian University, Xi'an 710071, China

Abstract. Vehicular federated learning (VFL) is a new paradigm that enables the use of data for distributed training under the premise of protecting the privacy of vehicular nodes (VNs). However, due to the heterogeneity of federated learning data, it is a challenge to evaluate the value of data and design an intelligent pricing scheme to effectively motivate the VNs in the vehicular networks (VNets) to complete learning tasks collaboratively. To this end, in this paper, we consider the value of data and propose a value-aware collaborative data pricing scheme for VFL. In the scheme, we first design a data transaction architecture based on the value of data and the cooperation among VNs. Then, by considering the non-independent and identically distributed (non-IID) degree and the age of data (AoD), we develop the data value model to evaluate the quality of data. Next, based on the requirement of the learning task and the data owned by each VN, we formulate the cooperation of the VNs as a coalition game, where the equilibrium of the coalition game is obtained by designing a distributed coalition formation algorithm. The simulation results show that the proposed scheme can lead to higher utility than the traditional methods.

Keywords: Federated learning · Vehicular networks · Data pricing · Coalition game

1 Introduction

With the rapid development of 6G and vehicular networks (VNets), a large number of machine learning tasks have been generated to facilitate intelligent transportation systems (ITS) [1]. As a new learning architecture, federated learning has attracted widespread attention to support various applications and services [2–4]. Different from traditional machine learning which needs to transmit the collected data to a central server for training, federated learning allows each data owner to locally train the collected data and to upload the local update to the central server, reducing the communication overhead and improving the data

W. Bao et al. (Eds.): ADHOCNETS 2021/TridentCom 2021, LNICST 428, pp. 289–300, 2022.
https://doi.org/10.1007/978-3-030-98005-4_21

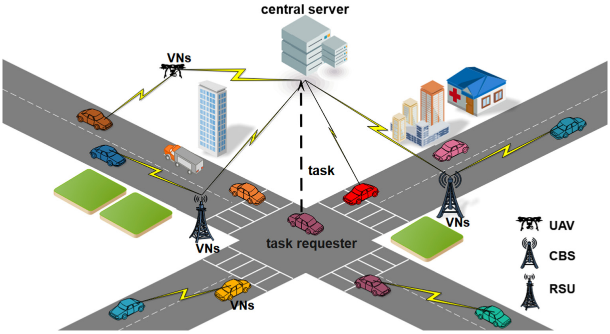

Fig. 1. System model.

security [5]. With these advantages, a new paradigm called vehicular federated learning (VFL) has been proposed to facilitate the applications and services in VNets such as vehicle positioning [6] and vehicle classification [7].

In the VFL, as shown in Fig. 1, a task requester can send its request to the central server which will select a number of vehicular nodes (VNs) including unmanned aerial vehicles (UAVs), cellular base stations (CBSs), roadside units (RSUs) and vehicles to train the learning task locally. After a round of training, the local update will be collected by the central server to update the global model. Repeat this process until a certain precision or the maximum number of the iteration is reached.

Obviously, in the model training process, the VNs with different data have different contributions to the global model training. According to [8], when performing learning tasks, the higher the quality of data, the higher the quality of local model updates. However, VNs usually generate and collect data in different ways. As a result, the number and characteristics of data owned by different VNs may vary greatly. Therefore, the data of VFL is non-independent and identically distributed (non-IID). For neural networks trained on highly skewed non-IID data, the accuracy will be significantly reduced [9]. Furthermore, most of the tasks in the VNs require the timeliness of data. Outdated data may reduce the accuracy of the training model, leading to wrong judgments and low quality of experience (QoE) of the task requester. Taking an autonomous driving task as an example, fresh driving environment data can significantly improve the accuracy of analysis of accident risks [10].

Apart from the data quality, another factor that affects VFL is data pricing. In the VFL, VNs may consume different resources when collecting data and training samples. As selfish and rational nodes, they will not actively participate in the training process of the VFL task. To solve this problem, a collection of mechanisms has been proposed to motivate VNs [11–13]. However, few of them consider the fact that VNs will declare various prices for joining in the VFL. In addition, the existing works typically think that VNs independently price data because they will train local models with their own data, which lacks the consideration of the cooperation among VNs to increase their data value and maximize their utility. Therefore, it is still an urgent problem to design a pricing scheme in VNets to facilitate the VFL by considering the value of data and the cooperation among VNs.

By considering the above issues, this paper proposes a value-aware collaborative data pricing scheme to facilitate VFL. In the scheme, we first establish a data transaction architecture for VFL in VNets by considering the value of data and the cooperation among VNs. Then, we design a value model to evaluate the quality of data by jointly considering the non-IID of data and age of data (AoD). Next, the interactions among VNs are modeled by a coalition game, where a distributed coalition formation algorithm is designed to achieve the equilibrium of the coalition game with the target of maximizing the utility of each VN. The simulation results show that the proposed scheme can bring the highest utilities to the VNs by comparing with the traditional methods.

The remainder of this paper is organized as follows. Section 2 describes the system model. Section 3 designs the proposed scheme. Section 4 formulates the cooperation among VNs as the coalition game. The simulation results are presented in Sect. 5. Finally, we conclude the paper in Sect. 6.

2 System Model

In this section, we introduce the system model which consists of network model and task model.

2.1 Network Model

In the VNets, as shown in Fig. 1. the networks consist of a central server, task requester and VNs.

- **Central Server:** As the controller of the VFL, the central server caches the data and price information of each VN. If a VN intends to update its information, it can send a revised message to the central server. In addition, the central server manages local updates, aggregates global models, and controls the collection of local updates and the distribution of global updates. After the central server receives a learning task, it can assign the task to a number of VNs to train the model locally and cooperatively.
- **Task Requester:** In the VNets, each VN can be a task requester. If a task requester has a learning task that needs to be completed, it can publish the task to the central server. After the central server completes the learning task, the trained model then can be delivered to the task requester.
- **Vehicular Nodes:** Let $I = \{1, ..., i, ..., I\}$ represent all the VNs in the VNets. For VN i, it can decide the price of its data to execute the local training based on the pricing model broadcasted by the central server. If the VN is selected by the central server to train the learning model, the reward will be paid to the VN after the learning task is completed.

2.2 Task Model

The set of learning tasks in the VNets is denoted as $R = \{1, ..., r, ..., R\}$. For task $r (r \in R)$, it can be represented as

$$r = \langle \mathrm{id}_r || \mathrm{des}_r || \mathrm{reward}_r \rangle, \tag{1}$$

where id_r is the unique identification of task r. des_r is the description of task r. reward_r is the reward for completing task r which is provided by the task requester. If a task requester intends to complete learning task r, it can deliver task r to the central server. Then the central server broadcasts a request message to the VNs in its communication coverage, shown as

$$\mathrm{req}_r = \langle r || h_r || \psi_r^0 || E || B || D_r \rangle, \tag{2}$$

where h_r is the machine learning model which is selected by the central server for task r. ψ_r^0 is the initial parameter of the machine learning model. E is the number of epochs. B is the local batch size used for local training. D_r is the requested number of samples in each round for training the learning task.

3 Value-Aware Collaborative Data Pricing for VFL

In this section, we introduce the proposed value-aware collaborative data pricing scheme.

3.1 Data Transaction Architecture

The data transaction architecture designed for the VFL has the following steps.

1) When a task requester has a task r to be completed, it first describes the detailed information of the task as a task requirement message and sends it to the central server.
2) After receiving task r from the task requester, the central server initializes the parameters E, B, and ψ_r^0. Meanwhile, the central server selects the machine learning model h_r for training task r.
3) Based on the information of the task request and the VNs, the central server starts to operate the coalition game. Specifically, for each VN $i (i \in I)$, it decides its data price based on the value of its data, where the price model is determined by the central server. Furthermore, it can form a coalition by itself to train the model individually or form a coalition with other VNs to increase its data value. The details of the designed coalition game among the VNs will be discussed in Section 4-A.
4) Based on the data prices of different coalitions, the central server selects a group of coalitions to form the optimal coalition set (OCS) and complete the learning task collaboratively. The selection of the OCS will be introduced in Section 4-B.

5) After the central server determines the OCS, it broadcasts the task request message and the determined OCS to the VNs within its communication coverage.
6) Once a VN receives the OCS, it then starts to execute the K rounds learning task if the VN is a member of the OCS. Specifically, for round $k(1 \leq k \leq K)$, the VNs in the OCS train the local model with their data sets. The purpose of local training is to minimize the loss function of task r. For VN i, the samples in its data set are denoted as $D_i = \{1, ..., d_i, ..., D_i\}$, where d_i is a pair of input and output $\{x_{d_i}, y_{d_i}\}$. Therefore, the loss function of sample d_i can be defined as $f_{d_i}(x_{d_i}, y_{d_i}, \psi_r^k)$.
   Denote by $S_J^* = \{S_1^*, ..., S_j^*, ..., S_J^*\}$ the set of coalitions in the OCS. Then, the samples in the data set of the VNs in coalition S_j^* can be expressed as $D_{S_j^*} = \sum_{i \in S_j^*} D_i$. Similarly, we can define the loss function of coalition S_j^* with data set $D_{S_j^*}$ shown as
$$f_{S_j^*}(x_{d_i}, y_{d_i}, \psi_r^k) = \frac{\sum_{d_i \in D_{S_j^*}} f_{d_i}(x_{d_i}, y_{d_i}, \psi_r^k)}{|D_{S_j^*}|}, \tag{3}$$
   where $|D_{S_j^*}|$ is the number of samples in $D_{S_j^*}$.
   Based on the loss function of the task, we then introduce the optimization process to find the parameter of (3) with the target of minimizing the value of the loss function.
   Similar to [14], we use the Mini-Batch Gradient Descent (MBGD) as the optimization method. The strategy of the MBGD is to reduce the value of loss function along the direction of gradient descent. For each epoch, there are $\lceil \frac{|D_{S_j^*}|}{B} \rceil$ iterations. Each iteration uses B samples to update the parameters. Therefore, the number of local updates of coalition S_j^* is $E * \lceil \frac{|D_{S_j^*}|}{B} \rceil$.
   Then, we update the local model of coalition S_j^* according to the gradient, where the local update can be defined as
$$\psi_r^k = \psi_r^{k-1} - \eta_r^k \nabla(\psi_r^{k-1}). \tag{4}$$
   In (4), ψ_r^{k-1} is the local model of coalition S_j^* in the k-1th round. η_r^k is the learning rate (step size), $\nabla(\psi_r^{k-1})$ is the gradient of ψ_r^{k-1}.
7) For each VN in the OCS, it sends the local update to the central server if the local training is completed.
8) The central server aggregates the local updates received from the VNs into a new global model Ψ_r^k. We have
$$\Psi_r^k = \sum_{S_j^*=1}^{S_J^*} \frac{|D_{S_j^*}|}{\sum_{S_j^*=1}^{S_J^*} |D_{S_j^*}|} \psi_r^k. \tag{5}$$
9) The central server sends the newly formed global model to the VNs in the OCS.

10) After the VNs in the OCS receive the global model, the k+1th round starts. Repeat the above training process until the given precision or the maximum number of iterations is reached, ending the execution process of the learning task.

4 Game Analysis

In this section, we first introduce the value of data. Then, we design the coalition game to formulate the interactions among VNs to maximize their utilities, followed by the coalition selection mechanism to form the OCS.

4.1 Value of Data

In the VFL, VNs with different data values have different contributions to the global model training. In other words, the contribution of nodes is related to the value of data. In general, the data owned by different VNs are heterogeneous. Therefore, we define the data value by jointly considering two factors, i.e., non-IID and AoD.

– **non-IID:** Based on the studies in [9], the average earth mover's distance (EMD) can be used to measure the heterogeneity of data distribution among different VNs. Considering the data sample $\{x_{d_i}, y_{d_i}\}$ follows the distribution P, we define the non-IID of coalition S_j as

$$\overline{EMD_{S_j}} \approx \sum_{c=1}^{Y} ||p_{S_j}(y=c) - p(y=c)||, \tag{6}$$

where y is the label of sample, and $y = \{0, 1, ..., Y\}$. $p_{S_j}(y=c)$ is the proportion of data with label c in coalition S_j's data and $p(y=c)$ is the proportion of data with label c owned by all the VNs. It can be seen from (6) that a high EMD value can result in a high non-IID degree of data.
In order to make EMD and AoD in the same order of magnitude, we adopt the min-max normalization to convert the range of EMD values to $[0, 1]$. The minimum value of EMD is 0. Therefore, (6) can be rewritten as

$$\begin{aligned} EMD_{S_j} &= \frac{\overline{EMD_{S_j}} - \min(\overline{EMD})}{\max(\overline{EMD}) - \min(\overline{EMD})} = \frac{\overline{EMD_{S_j}}}{\max(\overline{EMD})} \\ &= \frac{\sum_{c=1}^{Y} ||p_{S_j}(y=c) - p(y=c)||}{\max(||p_{S_j}(y=c) - p(y=c)||) * Y}. \end{aligned} \tag{7}$$

– **AoD:** AoD can be used to evaluate the freshness of data [15]. For coalition S_j, the AoD can be defined by

$$AoD_{S_j} = \sum_{i \in S_j} \frac{|D_i|}{\sum_{i \in S_j} |D_i|} \triangle_i, \tag{8}$$

where $\triangle_i$ is the AoD of data set owned by VN i, $\triangle_i \in [0, 1]$.

– **Data value:** We measure the value of data by its heterogeneity and age. In this way, based on the non-IID and AoD of the data owned by different VNs, the value of the data owned by coalition S_j can be expressed as

$$\begin{aligned} VoD_{S_j} &= \log(1 + \frac{\rho}{EMD_{S_j}} + \frac{1-\rho}{AoD_{S_j}}) \\ &= \log(1 + \frac{\rho * \max(||p_{S_j}(y=c) - p(y=c)||) * Y}{\sum_{c=1}^{Y} ||p_{S_j}(y=c) - p(y=c)||} \\ &+ \frac{1-\rho}{\sum_{i \in S_j} \frac{|D_i|}{\sum_{i \in S_j} |D_i|} \triangle_i}), \end{aligned} \tag{9}$$

where ρ is used to balance the importance of the non-IID and the AoD.

4.2 Coalition Game

1) Definition: a coalition game can be expressed as $[I, V(S_j)]$ where I is a set of VNs which take part in the coalition game, S_j is an arbitrary subset of I, and $V(S_j)$ is the characteristic function of coalition S_j.
2) Characteristic function: In the game, the characteristic function $V(S_j)$ represents the utility of the coalition. The data value and data volume of the coalition will affect the global model of VFL. Therefore, we can define the utility of the coalition by

$$\begin{aligned} V_{S_j} &= \alpha_r \log(1 + VoD_{S_j}) * |D_{S_j}| \\ &= \alpha_r \log(1 + \log(1 + \frac{\rho}{EMD_{S_j}} + \frac{1-\rho}{AoD_{S_j}})) * |D_{S_j}, \end{aligned} \tag{10}$$

where α_r is the basic price of task r. $\log(1 + VoD_{S_j})$ is the unit price of data.
3) Game rules: The rules of the coalition game consist of coalition conditions and coalition principles.
   – **Coalition conditions:** Considering that VN i is in coalition $S_{j'}$ and intends to split from this coalition and merge in coalition $S_{j''}$. It can join coalition $S_{j''}$, if the following two conditions are satisfied.
   For VN i, the utility after the node merge in the new coalition $S_{j''}$ is larger than the utility of the node in coalition $S_{j'}$. We have

$$U_i(V(S_{j''} \cup \{i\})) > U_i(V(S_{j'})). \tag{11}$$

   The utility obtained by each VN after VN i merged in coalition $S_{j''}$ is no less than the utility obtained by the VN in this coalition. It can be expressed as

$$U_{i'}(V(S_{j''} \cup \{i\})) > U_{i'}(V(S_{j''})), i' \in S_{j''}. \tag{12}$$

- **Coalition principles:** Given the coalition structure $S = \{S_1, ..., S_{j'}, ..., S_j, ..., S_{j''}, ..., S_J\}$, if VN i switches from coalition S_j' to coalition S_j'' , the coalition structure can be updated by

$$S_{j'} \leftarrow S_{j'} \setminus \{i\}, S_{j''} \leftarrow S_{j''} \cup \{i\}. \tag{13}$$

4) Node utility: If VN i in coalition S_j participates in this round of federated learning training and contributes to the global model, it will obtain the corresponding reward. Because the data value of different nodes is not the same, the contribution to the whole coalition is different. Therefore, we can calculate the utility of each VN in the coalition based on its contribution. We have

$$U_i(V(S_j)) = V(S_j) - V(S_j \setminus \{i\}). \tag{14}$$

Therefore, the utility model of the VN can be defined as:

$$U_i = \begin{cases} U_i(V(S_j)), i \in S_j, \\ V(\{i\}), \text{else}, \end{cases} \tag{15}$$

where $\{i\}$ is the coalition consisting VN i itself.
5) Game equilibrium: We can know from the game rules that each VN needs to operate some switches to find its coalition. Therefore, we use the iterative method to design a distributed coalition formation algorithm for finding the equilibrium of the coalition game [16]. The algorithm makes the coalition structure stable through the VNs continuously joining a coalition and leaving a coalition.
6) Coalition formation process: In the process of data transaction, each node tries to merge to form a coalition according to the coalition conditions and coalition principles, and finally forms a stable coalition structure through multiple rounds of the coalition formation process. The process of coalition formation can be described as follows.
   - **Initial coalition:** The nodes are divided into I coalitions, where each coalition has only one VN. The initial coalition structure can be given by $S = \{S_1, ..., S_j, ..., S_J\} = \{\{1\}, ..., \{i\}, ..., \{I\}\}$.
   - **Coalition formation:** For VN $i (i \in S_j)$, there is a candidate coalition sequence SQ_i that includes the remaining coalitions except S_j, shown as

$$SQ_i = \{S_1, ..., S_j - 1, S_j + 1, ..., S_J\}. \tag{16}$$

   Based on (16), VN i selects a coalition from SQ_i in turn. If the coalition conditions are satisfied, VN i merges in the coalition according to the coalition principles. In this way, each VN is analyzed by coalition game in turn until the stable coalition structure is achieved.

4.3 Coalition Selection

In the data transaction process, the central server intends to obtain the highest quality data with the lowest rewards. Therefore, we define the coalition type by considering value of data and price of data to help the central server make a decision.

Coalition type: The coalition type represents central server's preference for coalitions. The higher order of the type of S_j implies that the coalition has a larger possibility to contribute its data. Then, the type of coalition S_j can be defined as

$$TYPE_{S_j} = \log(1 + VoD_{S_j}/V(S_j)). \tag{17}$$

Coalition selection mechanism: When the coalition structure is stable, there are J^* coalitions. Sort by the coalitions based on the type in ascending order: $S_1 > S_2 > ... > S_{J^*}$. Central server selects the coalitions to join the training. There are two cases to form the OCS. First, if $D_r < D_{S_j}$, the central server selects D_r data from this coalition, we have $D_{S_j} = D_r$. In this case, the coalition provides D_r data to train the global model. Second, if $D_r > D_{S_j}$, we have $D_r = D_r - D_{S_j}$. Namely, this coalition provides all data to train the global model. Repeat this process until we have $D_r < D_{S_j}$. Then, the case is similar to case 1.

5 Simulation Results

In this section, the performance of the proposed scheme is evaluated by using a simulator in Matlab. We first introduce the simulation setup and then show the simulation results and discussions.

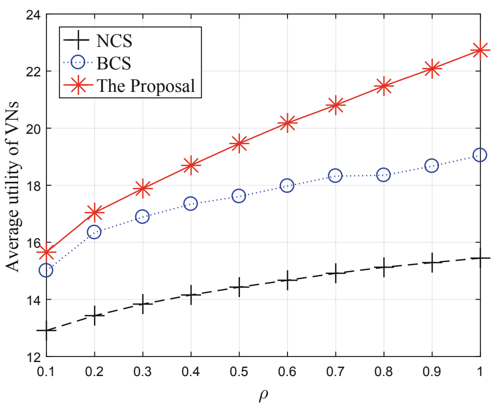

Fig. 2. Average utility of VNs by changing the EMD parameter ρ.

5.1 Simulation Setup

In our simulation, we consider an image classification task. The data set used in our simulation is Cifar-10 data set. Cifar-10 is divided into 5 batch training sets and 1 batch-test set, where the data set contains 10 labels: airport, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Each image belongs to one of the labels. The occurrence probability of each category of images is essentially equal. Each VN randomly extracts a fragment as its own data set. The values of ρ and α_r are 0.5 and 0.1, respectively. With this scenario, we evaluate the proposed scheme by changing the values of ρ and the number of VNs in the VNets. The performance of our proposed scheme is evaluated by comparing with the following conventional schemes.

- Non-cooperation scheme (NCS): In this scheme, all the VNs provide their data to train the learning model independently.
- Big coalition scheme (BCS): In this scheme, all the VNs form a big coalition and provide their data to complete the learning task collaboratively.

5.2 Simulation Results

Figure 2 is the average utility of VNs by changing the EMD parameter. As shown in Fig. 2, we can see that the proposed scheme can bring a higher utility to the VNs than the conventional schemes. This is because the proposed scheme allows VNs to form distributed coalitions to maximize their utility. In addition, it can be seen in this figure, the average utility of the VNs increases with the increase of ρ. The reason for this is that the importance of EMD increases with the increase of ρ. As a result, the value of data increases and the utility of VNs increases.

Figure 3 is the average utility of VNs by changing the number of VNs participating in the learning process. From this figure, we can see that the utility

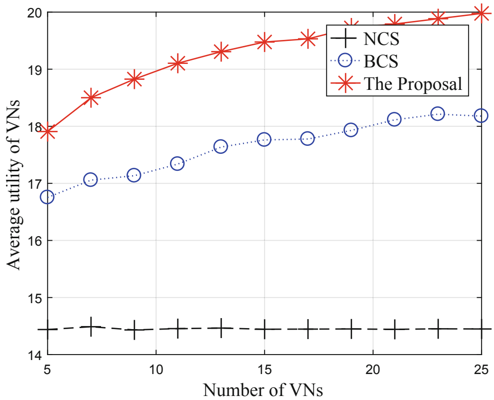

Fig. 3. Average utility of VNs by changing the number of VNs.

of VNs in the NCS is almost stable. In addition, as we can see in this figure, the utility of VNs increases with the increase of the number of VNs in the BCS and the proposal. This is because the increase of nodes promotes the variability of the coalition structure, where the probability of the coalition with low heterogeneity becomes high.

6 Conclusion

In this paper, we have proposed a value-aware data pricing scheme to facilitate the VFL in VNets. In the scheme, we have designed a novel data transaction architecture based on the value of data and the cooperation among VNs. Then, by considering the non-IID and AoD, we have developed a value model to evaluate the quality of data. Next, with different values of data owned by different VNs, a distributed coalition formation game has been designed to promote cooperation among VNs, where the equilibrium of coalition game has been obtained by designing a coalition formation algorithm. Simulation results have demonstrated that the proposed scheme can achieve the optimal strategy for VNs and outperform the conventional schemes in terms of the average utility obtained by VNs. For future work, the integration of blockchain and the proposed scheme will be considered to guarantee the security of the VFL in VNets.

Acknowledgement. This work was supported in part by the National Key R&D Program of China under Grant 2019YFB1600100, in part by the NSFC under Grant 61901341, in part by the China Postdoctoral Science Foundation under Grant 2021TQ0260, in part by the XAST under Grant 095920201322, in part by the National Natural Science Foundation of Shaanxi Province under Grant 2020JQ-301, and in part by the Fundamental Research Funds for the Central Universities under Grant XJS200109.

References

1. Hui, Y., et al.: Secure and personalized edge computing services in 6G heterogeneous vehicular networks. IEEE IoT J. **99**, 1–12 (2021). (Early Access)
2. Qu, Y., Pokhrel, S.R., Garg, S., Gao, L., Xiang, Y.: A blockchained federated learning framework for cognitive computing in industry 4.0 networks. IEEE Trans. Ind. Inf. **17**(4), 2964–2973 (2021)
3. Zhan, Y., Li, P., Qu, Z., Zeng, D., Guo, S.: A learning-based incentive mechanism for federated learning. IEEE IoT J. **7**(7), 6360–6368 (2020)
4. Qu, Y., et al.: Decentralized privacy using blockchain-enabled federated learning in fog computing. IEEE IoT J. **7**(6), 5171–5183 (2020)
5. Liu, Y., Yu, J.J.Q., Kang, J., Niyato, D., Zhang, S.: Privacy-preserving traffic flow prediction: a federated learning approach. IEEE IoT J. **7**(8), 7751–7763 (2020)
6. Kong, X., Gao, H., Shen, G., Duan, G., Das, S.K.: FedVCP: a federated-learning-based cooperative positioning scheme for social internet of vehicles. IEEE Trans. Comput. Soc. Syst. **9**, 1–10 (2021)

7. Abdellatif, A.A., Chiasserini, C.F., Malandrino, F., Mohamed, A., Erbad, A.: Active learning with noisy labelers for improving classification accuracy of connected vehicles. IEEE Trans. Veh. Technol. **70**(4), 3059–3070 (2021)
8. Duan, S., Zhang, D., Wang, Y., Li, L., Zhang, Y.: JointRec: a deep-learning-based joint cloud video recommendation framework for mobile IoT. IEEE IoT J. **7**(3), 1655–1666 (2020)
9. Ding, N., Fang, Z., Huang, J.: Optimal contract design for efficient federated learning with multi-dimensional private information. IEEE J. Sel. Areas Commun. **39**(1), 186–200 (2021)
10. Savazzi, S., Nicoli, M., Bennis, M., Kianoush, S., Barbieri, L.: Opportunities of federated learning in connected, cooperative, and automated industrial systems. IEEE Commun. Mag. **59**(2), 16–21 (2021)
11. Hui, Y., Su, Z., Luan, T.H.: Unmanned era: a service response framework in smart city. IEEE Trans. Intell. Transp. Syst. **99**, 1–15 (2021). (Early Access)
12. Kang, J., Xiong, Z., Niyato, D., Xie, S., Zhang, J.: Incentive mechanism for reliable federated learning: a joint optimization approach to combining reputation and contract theory. IEEE IoT J. **6**(6), 10700–10714 (2019)
13. Hui, Y., Su, Z., Luan, T.H., Li, C.: Reservation service: trusted relay selection for edge computing services in vehicular networks. IEEE J. Sel. Areas Commun. **38**(12), 2734–2746 (2020)
14. Mills, J., Hu, J., Min, G.: Communication-efficient federated learning for wireless edge intelligence in IoT. IEEE IoT J. **7**(7), 5986–5994 (2020)
15. Hu, H., Xiong, K., Qu, G., Ni, Q., Fan, P., Letaief, K.B.: AoI-minimal trajectory planning and data collection in UAV assisted wireless powered IoT networks. IEEE IoT J. **8**(2), 1211–1223 (2021)
16. Ng, J.S., et al.: Joint auction-coalition formation framework for communication-efficient federated learning in UAV-enabled internet of vehicles. IEEE Trans. Intell. Transp. Syst. **22**(4), 2326–2344 (2021)