Multi-tenant Optimal Cooperative Offloading based on Clustered Federated Learning in Optical and Wireless Converged Networks

Peng Chang
State Key Laboratory of information Photonics and Optical Communications
Beijing University of Posts and Telecommunications
Beijing, China
changpeng@bupt.edu.cn

Hui Yang*
State Key Laboratory of information Photonics and Optical Communications
Beijing University of Posts and Telecommunications
Beijing, China
yanghui@bupt.edu.cn

Yang Zhao
Department of fundamental networks
China mobile research institute
Beijing, China
zhaoyang@chinamobile.com

Sheng Liu
Department of fundamental networks
China mobile research institute
Beijing, China
liushengwl@chinamobile.com

Yunbo Li
Department of fundamental networks
China mobile research institute
Beijing, China
liyunbo@chinamobile.com

Jie Zhang
State Key Laboratory of information Photonics and Optical Communications
Beijing University of Posts and Telecommunications
Beijing, China
lgr24@bupt.edu.cn

***Abstract*—This paper proposes a cluster federated learning-based multi-tenant industrial Internet computing offloading scheme. Simulation results show that the scheme can improve model accuracy, reduce task failure rate, and improve user quality of experience.**

Keywords—clustered federated learning, computing offload, optical and wireless converged network

I. INTRODUCTION

According to IDC's prediction, more than 41.6 billion IoT devices will be connected to the Internet in 2025, and more than 45% of the data will be generated by edge devices. For example, various production links in the industrial Internet require a large number of edge devices with perception and control. The optical and wireless converged network has become the network architecture supporting the Industrial Internet because it can provide flexible access methods and high-quality communication for devices in the Industrial Internet [1]. As more and more edge devices join the industrial Internet, a large number of tasks that require offloading of computing will be generated in the production process [2], such as real-time interactive tasks required for multi-tenant collaborative production of computers. If too many tasks choose the same node for offloading, unloading delays will be unavoidable, or even impossible to offload. Therefore, research on computational offloading for these tasks is significant [3].

Recently, machine learning (ML)-based methods have emerged as powerful tools for solving task-computing offloading problems [4]. However, in the distributed scenario of the Industrial Internet, there are differences in the local industrial production data stored by the edge nodes, and the local resources of the edge nodes are limited, which makes it impossible to generate a complete computing offloading model to guide the offloading server of multi-tenant tasks. Federated learning is a promising candidate to solve the above problems [5], which can aggregate local models initially trained by edge nodes on cloud nodes. At the same time, cloud-edge collaboration can offload computing tasks from edge nodes to cloud nodes, reducing the computing pressure on edge nodes. Therefore, the combination of federated learning and cloud-edge collaboration can be used as one of the computing offloading solutions to realize multi-tenant real-time interactive tasks in the industrial Internet.

In this paper, a clustered federated learning (CFL)-based offloading solution for multi-tenant industrial Internet computing is proposed. In this solution, firstly, cloud nodes use a clustering algorithm based on the similarity of model parameters to divide edge nodes into multiple federated learning groups, so as to reduce the heterogeneity among edge data and improve the accuracy of model training. Then, in each federated learning group, the leader node is elected by time difference, so as to reduce the scale of model

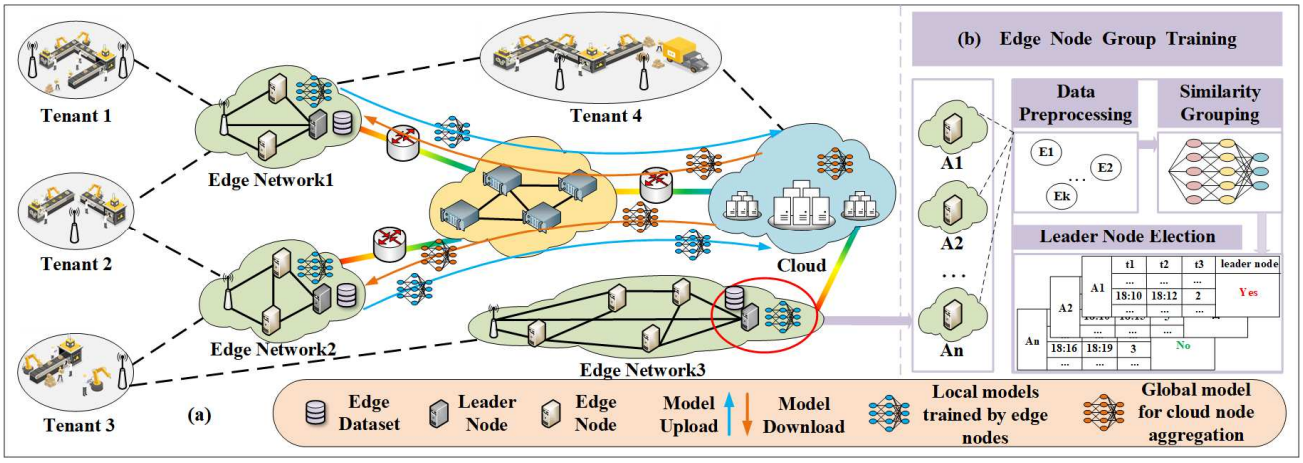

Fig. 1. An illustration of (a) the CFL architecture, and (b) leader node election.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

parameters transmitted to cloud nodes and improve the efficiency of model training. Finally, considering the limited computing and storage capabilities of edge nodes, a cloud-edge collaborative computing offloading model based on cluster federated learning is proposed to guide task offloading goals. The simulation results show that the scheme can effectively reduce the average service time and task failure rate, and improve the quality of experience (QoE) of multi-tenant users in the industrial Internet.

II. PROPOSED CFL FRAMEWORK

A. Optical and Wireless Converged Network Architecture

Fig. 1(a) shows the optical and wireless converged network architecture based on clustering federated learning. The multi-tenant Industrial Internet includes industrial equipment, edge networks and cloud computing center. Industrial equipment is connected to the edge network through wireless LAN, and the edge network is connected to the cloud computing center through optical transmission. However, there are a large number of devices in the industrial Internet. If too many devices choose to offload to the same edge node, it may cause the node to crash and cannot continue to calculate the offloading task. In order to develop an efficient computing offloading scheme, this study considers offloading tasks to the current edge node, adjacent edge nodes, and cloud nodes for processing.

B. Computational Offloading Model based on CFL

In a distributed scenario, there may be differences in the local industrial data stored by edge nodes. In order to improve the training accuracy of the computational offloading model, this paper proposes an edge node clustering method based on model parameter similarity to cluster edge nodes with similar model parameters. To implement this approach, it is first necessary to define pre-learned computation offloading model parameters for each edge node. In view of the high-dimensional characteristics of model parameters, principal component analysis (PCA) is used to reduce the dimension of model parameters. After the edge node completes the model training, it uploads the parameters to the cloud node. Finally, the spherical K-means algorithm, that is, the K-means algorithm of cosine similarity, is used to implement clustering. The clustering decision is determined by calculating the similarity of the unloaded model parameters. The schematic diagram of node clustering based on model parameter similarity is shown in Fig 2.

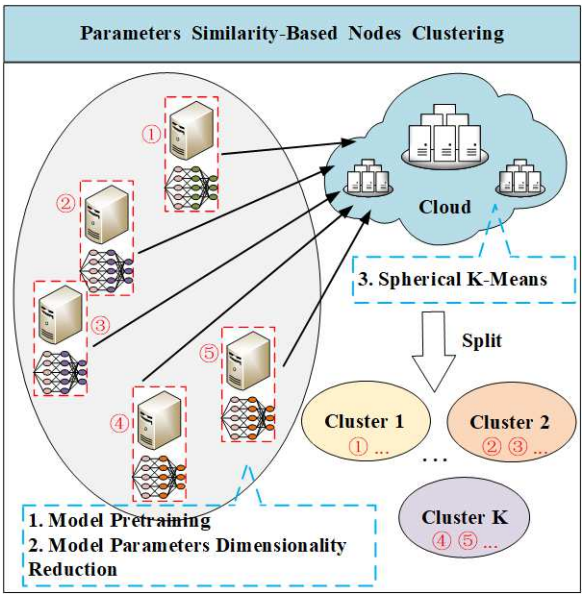

Fig. 2. Node clustering diagram based on similarity.

Algorithm1: Generating CFL-based Computational Offloading Model

Input: Edge dataset $A = \{A_1, A_2, \dots, A_n\}$,Groups set $P = \{P_1, P_2, \dots, P_k\}$, Initial model parameters ω_0, Learning rate η, Number of local training epochs E_{local}.
Output: High-accuracy Computational Offloading Model

1: The cloud node sends the model parameters ω_0 to all edge nodes
2: Edge nodes perform 5 rounds of model training based on local datasets $A = \{A_1, A_2, \dots, A_n\}$
3: Each round of model training includes deep learning models and linear regression models
4: Edge nodes upload model parameters ω_i to leader nodes for initial aggregation
5: Leader nodes upload the aggregated model parameters $\{\theta_1^{P_1}, \theta_2^{P_2}, \dots, \theta_k^{P_k}\}$ to the cloud node
6: The cloud node uses FedAvg algorithm $\theta_{global} = \frac{1}{k}\sum_{i \in k} \theta_i^{P_i}$ to aggregate models
7: **When** model accuracy $< \theta_{global}^{best}$ or training epochs $E < E_{local}$
8: Repeat above 1-6
9: **End when**
10: **End**

In order to reduce the scale of model parameters transmitted to cloud node, leader nodes are elected in each federated learning group. The leader node completes the local aggregation of model parameters, and then transmits the aggregated model parameters to cloud node for further aggregation. First, the edge nodes in each group send election requests with timestamp t_1 to other edge nodes, and other edge nodes convert the received election request time into timestamp t_2. The time from initiating a request to receiving a request is defined as $t_3 = t_2 - t_1$, and each edge node records the candidate node information and t_3 in the table. After receiving all the node election requests, the edge nodes broadcast their recorded information to each other, and the smallest sum of t_3 in each group is the leader node of the group, followed by the candidate nodes. The two nodes supervise each other, and if both fail, a new leader node election will be started. The schematic diagram of leader node election is shown in Fig. 1(b).

In the distributed scenario of the Industrial Internet, edge nodes collect and store multi-tenant industrial data. However, due to the limited computing power and the privacy and security issues involved in transmitting data to the cloud, a computational offloading model based on federated learning is needed to solve this problem. The model is divided into two stages: the first stage is to estimate the success of the offloading process through a deep neural network-based federated classification model. If the task execution and transmission are successful, the task offloading is considered successful. The second stage is to use the federated regression model to estimate the service time of the options predicted to be successful in the first stage, and the one with the smallest promised service time is the calculation offloading target. In the regression phase, the linear regression model is directly used to predict the service time. The training process of the model includes the following steps: 1) The cloud computing center sends the

model to all edge nodes; 2) The edge nodes train the local model, and after completing 5 rounds, the model parameters are transmitted to the leader node; 3) The leader node completes the preliminary Aggregate and transmit to the cloud node; 4) The cloud node uses the federated average algorithm to generate a global model, and then sends it to the edge node for training until it reaches the preset training rounds or threshold. The specific flow of the algorithm and the communication process involved are shown in Algorithm 1.

III. SIMULATION

This section focuses on the experimental validation and evaluation of the CFL model. The indicators evaluated in the experiment include model accuracy and performance indicators of the computational offloading model, where the performance indicators of the computational offloading model include average service time, task failure rate and QoE service satisfaction rate. During the experiment, a multi-core server with GPU is used to build a simulation environment, including a cloud node and 24 edge nodes, and a docker virtual environment is deployed. The CFL-based computing offload model considers short-term load, LAN delay, task size, and average node utilization on the edge side, and short-term load and WAN delay on the cloud. Among them, the short-term load refers to the number of tasks offloaded to the relevant nodes in the last 5 seconds. In order to evaluate the model accuracy of CFL, we choose models based on FL and local training of edge nodes as a comparison. In order to evaluate the computational offloading performance of CFL, we selected the computational offloading schemes based on simple moving average (SMA), multi-armed bandit (MAB) and Baseline (ordinary federated learning) as comparison schemes for benchmarking. Finally, the QoE service satisfaction rate is calculated by (1), which considers the actual service time T_i, task delay requirement time R_i, task delay sensitivity D_i and task failure rate S_i. Wherein, the value D_i is between 0 and 1, and higher for the delay intolerant tasks.

$$QoE_i = \begin{cases} 0 & if\ T_i \geq 2R_i \\ (1-S_i)\left(1-\dfrac{T_i-R_i}{R_i}\right)(1-D_i) & if\ R_i < T_i < 2R_i \\ 1 & if\ T_i \leq R_i \end{cases} \quad (1)$$

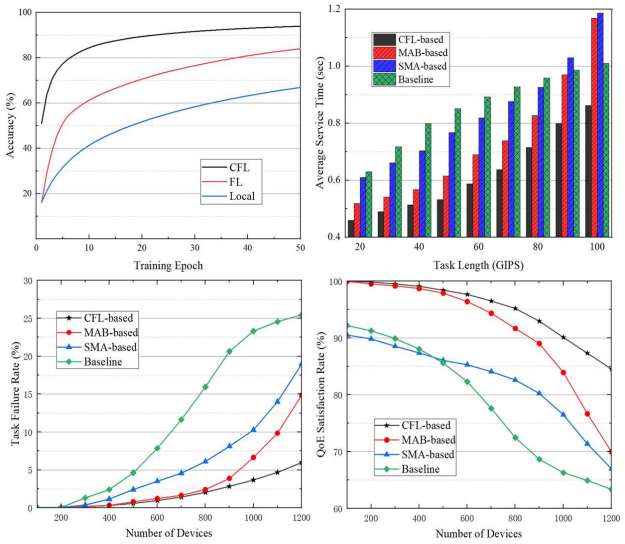

Fig. 3. Performance comparison in (a) model accuracy, (b) average service time, (c) task failure rate, and (d) QoE satisfaction rate.

As can be seen from Fig.3(a), compared with common federated learning and edge node local training models, the CFL proposed in this paper has higher model accuracy and faster convergence speed when edge nodes store multi-tenant data differences.

Fig.3(b) compares the average service time of CFL with MAB, SMA, and Baseline under different task lengths when the number of edge devices is fixed at 800. As the task length increases, the CFL-based computational offloading scheme consistently outperforms other comparative algorithms in the task length range from 10GIPS to 100GIPS. Fig.3(c) and (d) show the comparison of task failure rate and user experience quality satisfaction rate of different schemes when the task length is selected as 20GIPS. With the increase of edge devices, especially after reaching 1000 edge devices, a small amount of decision-making mistakes will have a large impact on the system. Through comparison, we found that the CFL-based computing offloading scheme we proposed can reduce the task failure rate and improve the user service quality satisfaction rate.

IV. CONCLUSION

This paper proposes a multi-tenant computing offload solution based on CFL. First, the cloud nodes divide the edge nodes into multiple federated learning groups according to the clustering algorithm based on the similarity of model parameters. Then, in each federated learning group, the leader node is elected by time difference, that is, the local model parameters are first aggregated at the leader node, and then further aggregated by the cloud node. Finally, a multi-tenant computing offloading model based on CFL is used to guide the task offloading target. The simulation results show that the scheme can effectively reduce the average service time and task failure rate, and improve the quality of experience (QoE) of multi-tenant users.

ACKNOWLEDGMENT

This work has been supported in part by NSFC project (62122015, 62271075), and Fund of SKL of IPOC (BUPT) (IPOC2021ZT04).

REFERENCES

[1] K. Bober, et al., "Distributed Multiuser MIMO for LiFi in Industrial Wireless Applications," J. Lightwave Technol. 39, 3420-3433 (2021).

[2] S. Yin, et al., "Dependency-aware task cooperative offloading on edge servers interconnected by metro optical networks," J. Opt. Commun. Netw. 14, 376-388 (2022).

[3] J. Borromeo, et al., "Experimental Demonstration of Scalable and Low Latency Crowd Management Enabled by 5G and AI in an Accelerated Edge Cloud," in Proc. OFC 2021.

[4] Zhang, Wei, et al. "Parallel computation offloading between MEC servers with metro optical network." 2021 Opto-Electronics and Communications Conference (OECC). IEEE, 2021.

[5] C. Li, et al, "High-Precision Edge-Cloud Collaboration with Federated Learning in Edge Optical Network," in Proc. OFC 2021.

[6] H. Yang, et al., "BrainIoT: brain-like productive services provisioning with federated learning in industrial IoT," IEEE Internet of Things Journal, vol. 9, no. 3, pp. 2014-2024, 2022.

[7] Sattler F, Müller K R, Samek W. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints[J]. IEEE transactions on neural networks and learning systems, 2020, 32(8): 3710-3722.