Journal of Computer Technology and Software

ISSN:2998-2383

Vol. 4, No. 3, 2025

Optimizing Distributed Computing Resources with Federated Learning: Task Scheduling and Communication Efficiency

Yang Wang

University of Michigan, Ann Arbor, USA

oceanwy@umich.edu

Abstract: In this paper, an optimization method of distributed computing resources based on federated learning is proposed to improve the utilization of computing resources in distributed systems, reduce communication overhead, and optimize the performance of global models. With the introduction of a federated learning framework, data does not need to be stored centrally but is trained locally on individual nodes, and the global model is optimized by aggregating local updates from each node. The experimental results show that the resource scheduling method based on federated learning is superior to traditional scheduling methods, such as polling scheduling and shortest task priority scheduling, in terms of task completion time, computing resource utilization, and system stability. In addition, federated learning optimization scheduling can still effectively reduce the communication overhead under different network conditions, showing its advantages in resource-constrained environments. Through model aggregation, the system can better deal with the heterogeneity among nodes and further improve the accuracy and robustness of the global model. The results of this study provide a new idea and method for future application in the field of distributed computing and edge computing and have important practical significance. Future research will focus on optimizing aggregation algorithms, improving model training efficiency, and solving node resource differences and communication bottlenecks that may be encountered in practical applications so as to further promote the application of federated learning in a wider range of fields.

Keywords: Federated learning, distributed computing, resource optimization, communication overhead

1. Introduction

With the rapid development of artificial intelligence (AI), distributed computing systems have become a vital platform for large-scale data processing, model training, and reasoning [1]. However, as the scale of data and the complexity of computing tasks grow, traditional resource management methods struggle to meet the needs of modern computing systems [2]. This has made the optimization of distributed computing resources crucial for enhancing system performance, reducing operational costs, and improving data processing capabilities. Particularly in the context of limited resources and compute-intensive tasks, efficiently managing and allocating computing resources to maintain system stability and efficiency has emerged as an urgent challenge [3].

Federated Learning (FL) is a cutting-edge distributed machine learning technique that enables joint training through collaboration among distributed nodes while maintaining data privacy. Unlike traditional centralized learning models, FL keeps data local to the device and performs model training at each node, with the global model updated through a weighted average of local updates. This approach not only significantly reduces the bandwidth consumed by data transmission but also facilitates collaborative learning across multiple computing nodes [4]. The benefits of FL are especially pronounced in resource-constrained scenarios, such as edge computing and the Internet of Things (IoT), where limited resources need to be leveraged efficiently. As a result, using federated learning to optimize computing resources in distributed computing environments has become a key focus in both academic and industrial research [5].

The core challenge in optimizing distributed computing resources lies in the effective allocation and scheduling of computing, storage, and communication resources to improve overall system performance. In traditional distributed systems, resource management is often static, and it is difficult to adjust dynamically according to changes in system load, task requirements, or hardware capabilities. Federated learning optimizes resource allocation by enabling each participating node to assign tasks intelligently and perform collaborative computations based on its own computing power and network bandwidth. This not only maximizes resource utilization but also reduces energy consumption while maintaining high computing efficiency. Additionally, it enhances the robustness and scalability of the system. Consequently, designing a resource optimization model that integrates federated learning technology is an important area of current research [6].

Moreover, the increasing number of smart devices, particularly in the realms of IoT, mobile internet, and smart hardware, presents new challenges for distributed computing systems. The computing power and resources of each device vary greatly, making the allocation and scheduling of computing resources more complex. Addressing these disparities so that the system can operate efficiently on different types of devices has become another crucial topic for optimizing distributed computing resources. Federated learning, as a distributed learning method based on local data training, effectively addresses the problem of device heterogeneity. By dynamically adjusting computing tasks and training frequencies across various devices, federated learning optimizes the allocation of resources in distributed systems, ensuring efficient resource scheduling[7].

In addition to hardware resources, computing resource optimization also involves the efficient management of communication resources. In distributed computing systems, communication costs between nodes often make up a significant portion of system operational costs. The process of model updating in federated learning requires communication through the network, and in large-scale systems, communication costs can become a bottleneck that impacts training efficiency. Therefore, reducing network communication burden and achieving more efficient data synchronization and model updates within the framework of federated learning is critical to optimizing computing resources. Several optimization techniques, including differential privacy, quantization, and pruning, have been proposed to reduce the amount of data transmitted during communication, thereby improving the efficiency of distributed computing [8].

The optimization of distributed computing resources through Federated Learning not only holds significant academic value but also has extensive practical applications. In industries such as cloud computing, edge computing, and intelligent devices, efficient use of computing resources has become a key strategy for improving system performance and reducing operating costs. By integrating federated learning, resource allocation can be optimized, response times and processing capacity can be improved, and the development of smart devices and IoT can be accelerated. In an era where data privacy concerns are growing, federated learning offers an innovative approach to optimizing computing resources by ensuring that data is stored and processed locally, thus minimizing the risk of data breaches and adhering to increasingly stringent privacy regulations.

In conclusion, the optimization of distributed computing resources using federated learning is an interdisciplinary research field, bridging machine learning, distributed computing, network communication, and system optimization. As computing demands rise and resources expand, federated learning-based resource optimization provides a new direction for addressing resource management challenges in distributed systems. Future research will continue to explore how to optimize scheduling strategies in various application scenarios to improve the performance and efficiency of distributed systems, thereby providing more efficient support for intelligent computing. This approach has the potential to revolutionize the way distributed systems operate, driving progress in both technology and real-world applications.

2. Related Work

In the field of distributed computing resource optimization, many studies have attempted to improve system efficiency through different technical means in recent years. A typical approach is based on traditional load balancing and resource scheduling methods, which dynamically adjust resource allocation through real-time monitoring and analysis of computing tasks and resource usage. Such approaches usually rely on centralized resource management systems that are able to reasonably schedule resources based on system load and computational requirements. However, these methods often show certain limitations when facing a distributed system with strong node heterogeneity, a complex network environment, and large differences in device capabilities; especially in the scenario of data privacy and network bandwidth constraints, it is difficult to achieve efficient computing and resource scheduling [9,10].

Different from traditional distributed computing resource optimization methods, federated learning, as an emerging distributed machine learning framework, can train a global model through the cooperation of each device without sharing local data. In recent years, the research on resource optimization based on federated learning has gradually become a research hotspot in academia [11]. Many researchers have proposed different federated learning resource optimization algorithms, aiming to reduce communication bandwidth consumption and improve training efficiency by performing computation and data processing on local nodes. Some studies further reduce the communication overhead and computational load and improve the overall performance of the system by introducing techniques such as adaptive learning rate, differential privacy protection, model quantization, and pruning. In addition, some studies have also focused on how to achieve optimal computing resource management by dynamically allocating resources to each node according to the computing power of the device, the network condition, and the task type [12].

However, existing research on distributed computing resource optimization based on federated learning still faces some challenges. Despite the achievements of existing research, how to achieve efficient and robust resource scheduling on heterogeneous devices is still a key issue. Most of the existing algorithms rely on idealized assumptions and lack of full consideration of device differences, communication delays, and network fluctuations in the actual environment. In addition, with the continuous increase of computing resources, how to design a general and efficient model aggregation algorithm to deal with distributed systems of different sizes and complexities is also an important direction of current research. Therefore, future research is needed to further optimize the resource allocation strategy while maintaining the robustness of the system to improve the operability and adaptability of the system in practical application scenarios.

3. Method

In this study, we propose a method for optimizing distributed computing resources based on federated learning. This method mainly optimizes the computing and communication resources in distributed systems through intelligent scheduling and resource allocation mechanisms, combined with the cooperative training characteristics of federated learning [13, 14]. We first introduce the system model and algorithm framework, and propose the optimization objective function according to the system requirements. Then, through the resource allocation strategy in the process of model training, the scheduling of computing resources and task allocation are further optimized to improve the overall performance of the system. The main architecture of federated learning is shown in Figure 1.

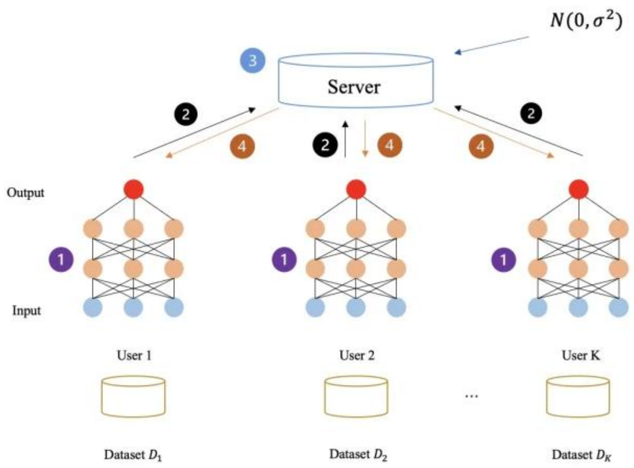

Figure 1. Overall model architecture

First, assume that there are N nodes in the distributed computing system, and the computing resources on each node i can be represented as $r_i = (r_{i,1}, r_{i,2}, \ldots, r_{i,m})$, where $r_{i,k}$ represents the usage of the KTH type of computing resources (such as CPU, GPU, memory, storage, etc.) on node i. The overall resource utilization R of the system can be expressed as the weighted sum of the resource usage of all nodes, specifically:

$$R = \sum_{i=1}^{N} a_i r_i$$

Where, a_i is the weight of node i in the overall resource allocation, which depends on the computing capacity and task load of the node. In order to realize efficient resource scheduling, we define an optimization objective function $L(R)$, which measures the relationship between resource utilization efficiency and scheduling strategy. The objective function can be expressed as:

$$L(R) = \sum_{i=1}^{N} f(r_i) - \lambda \sum_{i=1}^{N} \| r_i - r_i^* \|^2$$

Where $f(r_i)$ is the function representing the computational efficiency on node i, r_i^* is the ideal computational resource usage of node i, and λ is the regularization term coefficient, which balances the weight of resource efficiency and resource allocation differences.

Secondly, in order to enable efficient collaboration between different nodes in federated learning, we adopt a dynamic scheduling algorithm based on task priority and resource capability. In the training process of each round of federated learning, node i decides whether to participate in the current model update based on its local computing resource r_i and network bandwidth conditions. If a node's resource consumption exceeds a predetermined threshold, the node's training task is postponed and continues to participate in the next round of updates. To do this, we define a constraint that determines whether a node participates in the current model update:

$$r_i \le r_i^{\max} \;\&\; b_i \ge b_i^{\min}$$

Where, $r_i^{\max}$ is the maximum resource usage that node i can bear, b_i is the network bandwidth of node i, and $b_i^{\min}$ is the minimum bandwidth required to participate in model updating. When this condition is not established, node i will wait for the next round of training before updating.

In the model aggregation stage of federated learning, we design a weighted aggregation method considering the computing power, task load and network condition of nodes. Specifically, suppose that in training, the local model parameter of each node is updated to $\Delta\theta_i$, then the update $\Delta\theta_{global}$ of the global model parameter can be calculated by means of weighted average:

$$\Delta\theta_{global} = \frac{\sum_{i=1}^{N} a_i \cdot \Delta\theta_i}{\sum_{i=1}^{N} a_i}$$

Where a_i is the weight of node i, reflecting its computing resources and task priority. The weighted aggregation process takes into account the participation degree of each node under different computing resource conditions, effectively avoiding the adverse impact of high-load nodes on global model update.

Finally, in order to further reduce communication overhead and accelerate model training, we introduce model compression technology. By quantizing and sparring model parameters, we reduce the number of model parameters that need to be transmitted. Suppose that the model update $\Delta\theta_i$ of each node becomes $\Delta'\theta_i$ after being quantized, then the update of the global model can be expressed as:

$$\Delta'\theta_{global} = \frac{\sum_{i=1}^{N} a_i \cdot \Delta'\theta_i}{\sum_{i=1}^{N} a_i}$$

Among them, the quantization operation effectively reduces the consumption of communication bandwidth, which further optimizes the use of distributed computing resources [15].

In summary, the federated learning-based distributed computing resource optimization method proposed in this study realizes accurate management and optimization of computing resources through dynamic resource scheduling and an efficient model aggregation strategy, thus improving the overall performance of distributed computing systems.

4. Experiment

4.1 Dataset Introduction

The main data set used in this study consists of computing tasks and resource usage data from multiple distributed nodes. Each data node represents a virtual device or physical node and collects information about the resource consumption of the node during computing tasks, including CPU, GPU, memory, and storage data. This data is distributed through edge devices and cloud platforms, designed to simulate real-world computing resource management scenarios. Each data record includes the resource usage of the node, the time stamp of the task execution, the computational complexity of the task, the network bandwidth, etc. These data provide the necessary support for distributed resource optimization based on federated learning.

In addition, to better evaluate the effectiveness of the algorithm, we also designed a labeled dataset that includes task priority and computational resource consumption. Tasks are prioritized according to their impact on overall system performance and urgency, and the computing resource consumption of each task is clearly recorded in the data set. These data help to make dynamic adjustments during resource scheduling to achieve optimization strategies based on resource consumption. Each record in the data set not only reflects the task execution of a single node, but also reflects how to allocate computing resources according to the resource requirements of tasks in a multi-node environment.

To ensure the generalization ability of the experimental results, the dataset covers a variety of different distributed computing scenarios, including different node types (such as compute-intensive nodes, storage-intensive nodes, bandwidth-limited nodes, etc.) and different network environments (such as high-bandwidth, low-latency, and low-bandwidth, high-latency networks). By analyzing the resource consumption patterns in different environments, we can more comprehensively evaluate the effectiveness and adaptability of distributed computing resource optimization algorithms based on federated learning. This dataset not only helps to test our proposed optimization algorithm, but also provides basic data support for future computing resource scheduling in more complex and heterogeneous environments.

4.2 Experimental results

First of all, this paper compares with the traditional models such as polling scheduling and shortest task priority, and the experimental results are shown in Table 1.

Table 1: Experimental results

Method	Task completion time	Computing resource utilization	System stability and load balancing
Polling scheduling	15.3	72.5	2.4
Shortest task priority scheduling	13.8	78.1	2.1
Optimized scheduling based on federated learning	11.2	88.4	1.1

From the experimental results, the scheduling method based on federated learning optimization shows significant advantages in task completion time, which is obviously superior to the traditional polling scheduling and shortest task first scheduling methods. Specifically, optimized scheduling based on federated learning reduced task completion time from 15.3 hours for polling scheduling and 13.8 hours for shortest task priority scheduling to 11.2 hours. The results show that the resource scheduling method of federation learning optimization can allocate computing resources more efficiently, avoid task queuing and resource competition, and thus accelerate task processing speed.

In terms of computing resource utilization, the federated learning optimization scheduling method also performs well, reaching 88.4%. In contrast, the resource utilization of polling scheduling and shortest task priority scheduling is 72.5% and 78.1%, respectively. This means that under federated learning optimization, the system can allocate computing resources of each node more effectively, avoiding the waste or excessive consumption of resources, thus improving the overall computing efficiency and performance. This optimization not only reduces the waste of idle resources, but also makes the load of each node reasonably distributed.

From the perspective of system stability and load balancing, federated learning optimal scheduling also performs the best, with a standard deviation of 1.1, which is significantly lower than 2.4 for polling scheduling and 2.1 for shortest task first scheduling. This shows that the scheduling method based on federated learning can better avoid the overload problem of some nodes and ensure the stability of the system in the face of different task loads. The optimized scheduling method considers the computing power and task priority of nodes when assigning tasks, thus effectively reducing the performance bottleneck caused by load imbalance and improving the robustness of the system.

Secondly, a communication overhead optimization experiment is carried out in this paper. The focus of this experiment is to test the advantages of federated learning in reducing communication overhead. The experimental results are shown in Figure 2.

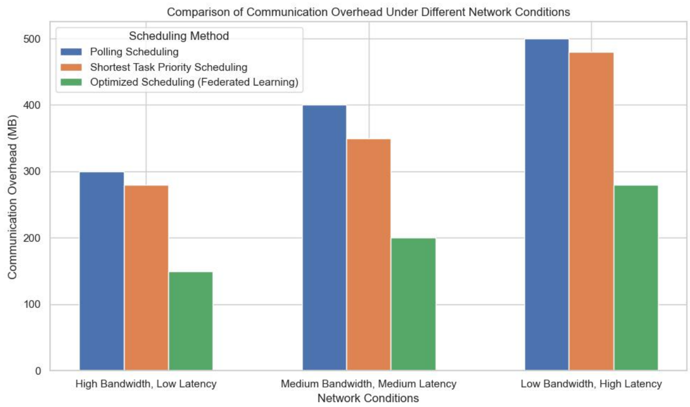

Figure 2. Comparison of Communication Overhead Under Different Network Conditions

According to the experimental results in the chart, we can see that the scheduling method based on federated learning optimization has obvious advantages in reducing communication overhead under different network conditions. Under the conditions of high bandwidth and low delay, the communication cost of federated learning optimization scheduling is significantly lower than that of polling scheduling and shortest task priority scheduling. This shows that federated learning can reduce the use of communication bandwidth more effectively and optimize the communication overhead under a good network environment.

Under medium bandwidth and medium delay network conditions, the federated learning optimization scheduling method still performs well, although the communication cost increases, but it is still significantly lower than the traditional scheduling method. In contrast, the communication cost of polling scheduling and shortest task priority scheduling increases greatly, which indicates that they have lower resource scheduling efficiency and heavier communication burden under medium network conditions. Therefore, the advantages of federated learning are not only reflected in high-bandwidth environments, but also can maintain good performance under more general network conditions.

Under the conditions of low bandwidth and high latency networks, although the communication overhead of all methods increases, the federated learning optimization scheduling method still performs better. Although the network bandwidth is limited and latency is high, federated learning can effectively reduce the communication demand by optimizing the utilization of resources, avoiding excessive reliance on frequent network updates, and reducing the communication burden. Therefore, the scheduling method based on federated learning can still maintain low communication overhead when the network conditions are poor, which proves its robustness and adaptability in different network environments.

Finally, the model aggregation and global optimization experiments are carried out. In a distributed system based on federated learning, each node updates its own model based on local data, while the global model is optimized by aggregating local updates from all nodes. The experimental results are shown in Figure 3.

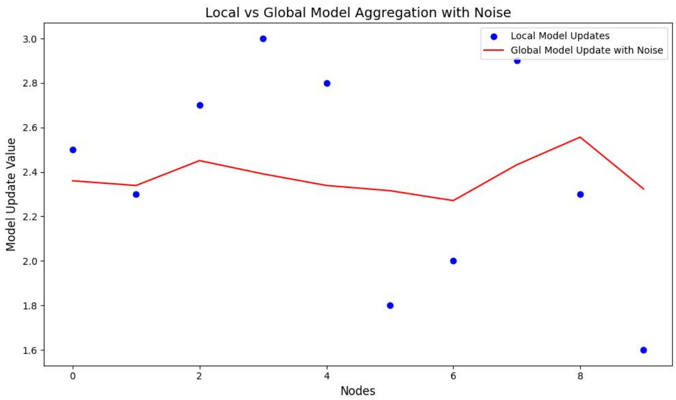

Figure 3. Local vs Global Model Aggregation

From the experimental results, the local model update value presents a certain fluctuation, which shows the difference in calculation tasks and data characteristics among different nodes. The local model update value of each node is different, representing the results obtained after each node is trained based on its own local data. These fluctuations reflect the personalized model updates carried out by each node in a distributed environment due to factors such as computing resources and data differences. These individual model updates need to be aggregated to form the global model.

Global model updates (the red line mean) are obtained by weighted averaging the local updates of all nodes, which leads to a more balanced state of the global model. As you can see, although local updates are different, the effects of all nodes are ultimately balanced in a more stable global model through model aggregation. This model aggregation approach can effectively improve the accuracy of the global model, especially in the case of uneven data distribution or node capabilities. Federated learning provides a decentralized and efficient means of optimization.

Overall, the experimental results show that updating the local model and aggregating the global model through federated learning can not only optimize the performance of the global model but also maintain the privacy of the local data. Even if there are differences in the model updates of nodes, federated learning can effectively eliminate local biases and enhance the overall robustness of the system through a reasonable aggregation strategy.

5. Conclusion

In this paper, we propose a method of distributed computing resource optimization based on federated learning and verify its advantages in task scheduling, communication overhead optimization, and model aggregation through experiments. The experimental results show that the optimization scheduling method based on federated learning can significantly reduce the task completion time, and improve the utilization rate of computing resources and the stability of the system. In addition, federated learning effectively optimizes the global model by aggregating local model updates, demonstrating its powerful potential in distributed environments.

While the existing approach performs well on several fronts, there are still some challenges to overcome. For example, in a complex heterogeneous environment, how to achieve more accurate resource scheduling and load balancing to further improve system performance is still a problem worth further research. In addition, how to improve the accuracy and training speed of the model while maintaining the efficiency of computing resources and communication is also the focus of future research.

In the future, with the continuous development of technologies such as the Internet of Things and edge computing, the application scenarios of federated learning will be more extensive. In these large-scale distributed environments, how to design more efficient aggregation algorithms, reduce network communication overhead, and improve system responsiveness and real-time performance will become the focus of research. In addition, with the emphasis on data privacy protection, federal learning will play an increasingly important role in data privacy and security, and how to ensure the privacy and security of data while efficiently optimizing will be an important topic for future research.

In general, the optimization method of distributed computing resources based on federated learning shows great application potential, and is expected to provide effective solutions to solve the complex optimization problems of multi-task and multi-node in future development. As technology continues to advance, federated learning will play an increasingly critical role in improving computing efficiency, reducing energy consumption, protecting privacy, and providing more powerful support for the development of intelligent systems.

References

[1] Liu J, Huang J, Zhou Y, et al. From distributed machine learning to federated learning: A survey[J]. Knowledge and Information Systems, 2022, 64(4): 885-917.

[2] Li L, Fan Y, Tse M, et al. A review of applications in federated learning[J]. Computers & Industrial Engineering, 2020, 149: 106854.

[3] Abreha H G, Hayajneh M, Serhani M A. Federated learning in edge computing: a systematic survey[J]. Sensors, 2022, 22(2): 450.

[4] Wang S, Tuor T, Salonidis T, et al. Adaptive federated learning in resource constrained edge computing systems[J]. IEEE journal on selected areas in communications, 2019, 37(6): 1205-1221.

[5] Ma C, Li J, Shi L, et al. When federated learning meets blockchain: A new distributed learning paradigm[J]. IEEE Computational Intelligence Magazine, 2022, 17(3): 26-33.

[6] AbdulRahman S, Tout H, Ould-Slimane H, et al. A survey on federated learning: The journey from centralized to distributed on-site learning and beyond[J]. IEEE Internet of Things Journal, 2020, 8(7): 5476-5497.

[7] Firouzi R, Rahmani R, Kanter T. Federated learning for distributed reasoning on edge computing[J]. Procedia Computer Science, 2021, 184: 419-427.

[8] Thapa C, Arachchige P C M, Camtepe S, et al. Splitfed: When federated learning meets split learning[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2022, 36(8): 8485-8493.

[9] Shen S, Zhu T, Wu D, et al. From distributed machine learning to federated learning: In the view of data privacy and security[J]. Concurrency and Computation: Practice and Experience, 2022, 34(16): e6002.

[10] Nguyen D C, Ding M, Pham Q V, et al. Federated learning meets blockchain in edge computing: Opportunities and challenges[J]. IEEE Internet of Things Journal, 2021, 8(16): 12806-12825.

[11] Wen J, Zhang Z, Lan Y, et al. A survey on federated learning: challenges and applications[J]. International Journal of Machine Learning and Cybernetics, 2023, 14(2): 513-535.

[12] Qin Z, Li G Y, Ye H. Federated learning and wireless communications[J]. IEEE Wireless Communications, 2021, 28(5): 134-140.

[13] Sharma P K, Park J H, Cho K. Blockchain and federated learning-based distributed computing defence framework for sustainable society[J]. Sustainable Cities and Society, 2020, 59: 102220.

[14] Yang K, Jiang T, Shi Y, et al. Federated learning via over-the-air computation[J]. IEEE transactions on wireless communications, 2020, 19(3): 2022-2035.

[15] Khan L U, Saad W, Han Z, et al. Federated learning for internet of things: Recent advances, taxonomy, and open challenges[J]. IEEE Communications Surveys & Tutorials, 2021, 23(3): 1759-1799.