A Performance Study of Algorithms and Frameworks for Federated Learning

Shuyan Zhang[1, *]

[1]Beijing Institute of Technology, Beijing, 100081, China
[*]Corresponding auther: Arccsc@outlook.com

Abstract. Federated learning is a type of distributed machine learning that focuses on solutions for the properties of training data on edge devices as well as the privacy of the training set. Federated learning is a discipline highly relevant to real-world applications, and thus the emphasis on different perspectives requires an adaptation of the federated learning framework. Although almost every newly proposed federated learning algorithm is compared with some existing algorithms, current research on testing comparisons between commonly used federated learning algorithms remains vague and complex. Therefore, the purpose of this paper is to test and compare several federated learning frameworks, including the representative FedAvg, MOON, FedProx, and MOON. Based on revisiting the theory and key steps of these algorithms, an analysis of the performance performance will be conducted, evaluating their advantages for applications. Furthermore, a summary based on the test results will be provided, pointing out possible future challenges as well as research directions.

Keywords: Federated learning; FedAvg; Machine learning

1. Introduction

As the use of personal computing devices becomes more widespread, the amount of information that can be collected, either from sensors of all kinds or from the users themselves, is incalculable in scale. This information seems to make a good training dataset, however, it also naturally involves issues of privacy and data security. The act of storing user-generated information directly implies the responsibility for security concerns and is subject to strict legal oversight. In order to enable users to benefit from global models trained on their data, and to avoid directly involving user's raw data, Mcmahan et al. [1] put forth the concept of federated learning, where the training task is realized through a loose federation between clients. The transfer of source data between servers and clients is replaced by the parameters of the training process, allowing the information transferred to be reduced to the minimum required to update a particular model, thus guaranteeing the privacy involved during the training process.

In a typical federated learning process, clients receive an initial model from the server at first before training it locally on the basis of global state along with their local dataset. After a certain round of training, local updates are subsequently transferred to the server. Then, the information collected is aggregated by the server so as to update the global state, after which new model is sent to clients for the next iteration.

Distinct from typical distributed optimization, federated optimization has the following properties due to its data source for the training process. (1) Non-Independent and Identically Distributed (Non-IID). The local training set of clients could hardly reflect the global distribution. (2) Unbalanced. The local training sets of clients are of different sizes. (3) Massively distributed. The scale of clients is anticipated to be significantly higher than the scale of clients in the training dataset on average. (4) Limited communication.

Based on these features and the initial motivation of privacy-orientation, the challenges that federated learning needs to address are as follows: (1) Communication. The communication pattern between servers and clients (e.g., frequency of communication, the content of communication) is a factor that directly affects the efficiency and results of federated learning. (2) Heterogeneity. Heterogeneity could be manifested at two major aspects. Systematic heterogeneity is manifested in the variability of clients, leading to differences in dimensions such as training time among different clients. Statistical heterogeneity is manifested in the bias between global and local models in terms of updated trends. (3) Privacy. Although the parameters transferred during the process of federated learning are not raw data from the clients, they still contain information about the model and affect model updates. Therefore, the possibility of threats such as inference attacks and Byzantine attacks are inevitably taken into account.

In recent years, several studies have put forth their respective solutions for these three major aspects. In terms of communication, Jakub et al. [2] introduced both structured updates as well as sketched updates with an idea of a compression scheme over the content of the communication. Lalitha et al. [3] proposed fully decentralized federation learning, which restricts communication to one-hop neighbors. To address the heterogeneity problem, Active federated learning was put forth by Goetz et al. [4] to improve the efficiency of training by controlling the probability of selecting a particular client. Diao et al. [5] proposed HeteroFL to face scenarios where clients are extremely different in terms of computational and communication performance. Huang et al. [6] proposed FedAMP, which improves the performance of the algorithms by employing the collaboration of clients with similar data. For privacy, Wei et al. [7] introduce differential privacy in federated learning and analyze the performance. Xu et al. [8] propose VerifyNet for privacy in deep neural networks. All of these methods make their own trade-offs for the dimensions of efficiency, accuracy, and security under the framework of federated learning.

Although the above efforts have greatly contributed to the research progress in federated learning, the efficiency of various algorithms in various contexts has received scant study. In order to provide some decision basis for algorithm selection in real scenarios, this paper introduces basic theories of representative algorithms in detail and quantitatively compares their experimental results in different scenarios. The major contributions of this paper are as follows: (1) Test a variety of federated learning algorithms in different contexts and compare their test results. (2) Analyze possible reasons and speculate on the practical applicable contexts based on the test results. (3) Infer subsequent challenges and research trends of federated learning.

2. Related Work

Federated learning has become one of the hotspots of research in recent years as a solution for principles of focused collection as well as data minimization applied to reality. FedAvg proposed by Mcmahan et al. [1] is the first and relatively fundamental framework of federated learning, which achieves the goal of reducing the number of communication rounds by increasing computation on individual clients, and datasets with unbalanced or non-IID properties are taken into account for the first time. Li et al. [9] put forth Model-Contrastive Federated Learning (MOON) for the context of image-based datasets under deep learning models. This method utilizes resemblance between model representations to instruct the learning process of individuals for the purpose of optimizing the performance of federated learning on image training sets.

When facing systematic heterogeneity problems, FedAvg generally drops clients with relatively poor performance based on a certain standard, which may lead to bias in the global model. Li et al. [10] proposed Federated Optimization in Heterogeneous Networks (FedProx) as an adapted and generalized version of FedAvg to address this problem and improve average performance in highly heterogeneous scenarios. For the problem of inconsistency within the minimum of experience loss between local device-level and global-level during communication, Acar et al. [11] proposed Federated Learning based on Dynamic Regularization (FedDyn). This method introduces a dynamic parameter to each client for every update to ensure consistency between global and local solutions. This entitles the framework theoretically independent of factors such as heterogeneity, communication quality, etc.

3. Performance Comparison and Analysis

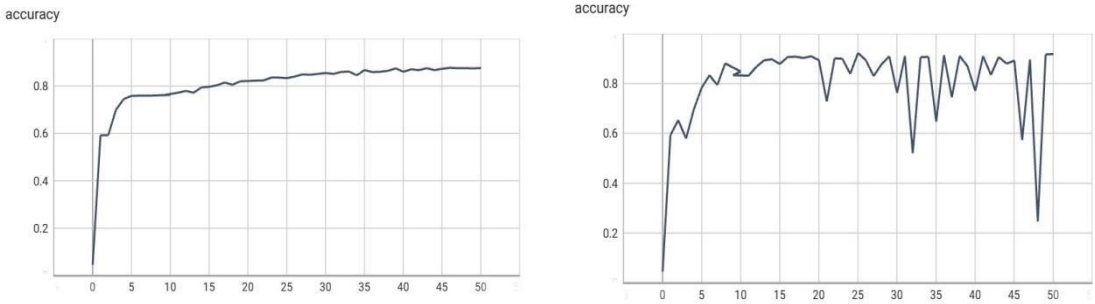
3.1. Experimental Setup

In this paper, four state-of-the-art methods, FedAvg, MOON, FedDyn, and FedProx, are selected for comparison. The experiments are mainly set on two datasets, including CIFAR-10, and Fashion-MNIST. Furthermore, two different network architectures are used for testing. For CIFAR-10, the DNN network was selected as the model, while the CNN network was selected for Fashion-MNIST. All federated learning frameworks tested were based on PyTorch implementations. For all the framework, batch size was set to 10 and local learning rate was 0.005. The number of global rounds was set to 50 since the tested frameworks obtained little or almost no accuracy improvement in the subsequent rounds. The training data was set to be Non-IID and unbalanced to simulate realistic scenarios. Dirichlet distribution was used to generate these datasets to characterize the unbalanced distribution of data samples.

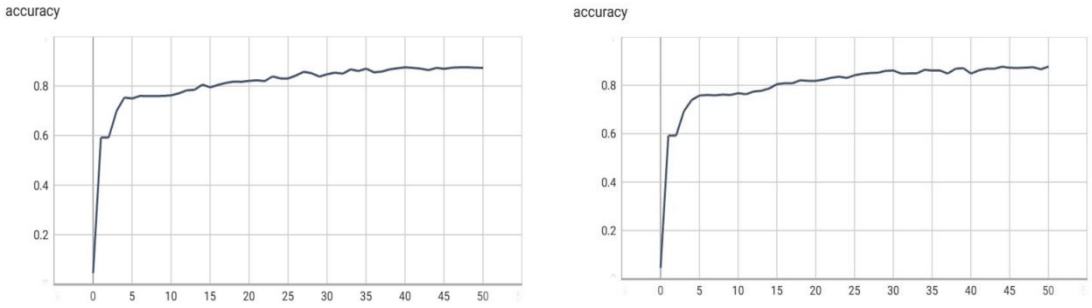
3.2. Accuracy Comparison

For the four federated learning frameworks, the accuracy of the fixed global number of rounds is tested first. The accuracy corresponding to the number of rounds could to some extent reflect the efficiency of the algorithm and the performance of the output results. For frameworks that include additional hyperparameters such as FedProx, parameters that ensure better performance ($\mu = 0.01$ for the CIFAR-10 dataset) are selected.

Figure 1 demonstrates the accuracy of the four federated learning algorithms based on the Fashion-MNIST dataset. As can be seen, FedAvg, MOON, and FedProx perform relatively similar to each other. All three reach a relatively high accuracy rate at the early stage, indicating that they could perform well and consistently in smaller contexts. FedDyn, on the other hand, exhibits larger fluctuations and unstable performance, which might be a result caused by the framework's greater emphasis on communication performance at a large scale.



(a) FedAvg (b) FedDyn

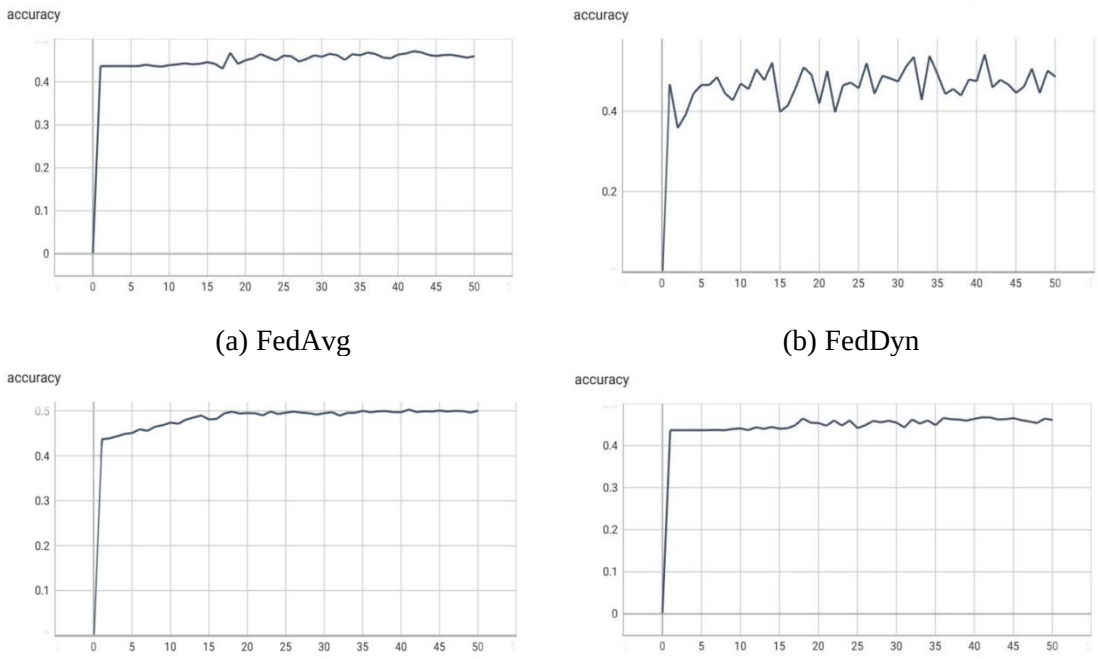


(c) FedProx (d) MOON

Figure 1. Accuracy in different number of global rounds on Fashion-MNIST

Figure 2 demonstrates the accuracy of the four federated learning algorithms on the basis of CIFAR-10 dataset. FedAvg and MOON perform similarly and close to their performance on the Fashion-MNIST dataset. FedProx, on the other hand, subsequently achieves a higher accuracy rate while performing similarly at early rounds to FedAvg. Possibly due to the small number of clients in this test, FedDyn's performance is still relatively unstable, but its final result is similar to that of FedAvg and MOON.

(a) FedAvg (b) FedDyn



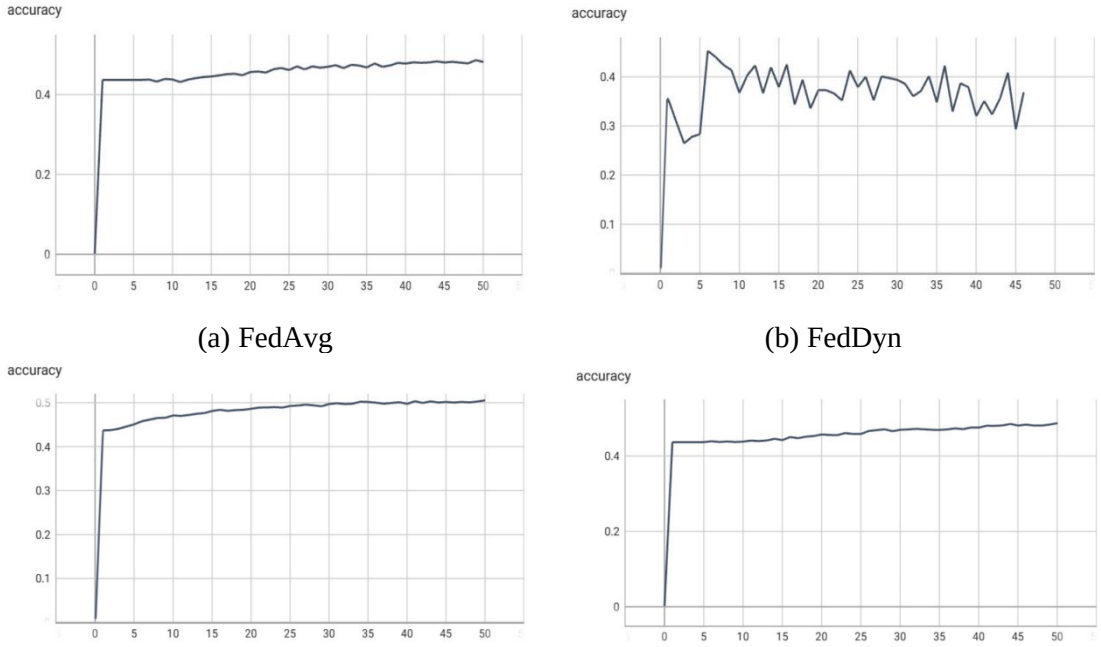
(c) FedProx (d) MOON

Figure 2. Accuracy in different number of global rounds on CIFAR-10

3.3. Scalability

In this paper, the scalability performance of different frameworks is also tested by the adjustment over the size of the clients. The effect of increasing the number of clients creates inevitable challenges for aspects such as the communication performance of the frameworks. Therefore, scalability performance is likely to have a direct impact on the decision of which federated learning framework to apply.

Figure 3 demonstrates the accuracy of the four federated learning algorithms based on the Fashion-MNIST dataset as the number of local rounds increases. When local epoch is set to 1, the magnitude of updates per round is relatively small, making the gain obtained from communication low. When this parameter is increased, although the gain in updates per round is greater, the local-global consistency is also more unstable and even generates errors. According to the test results, this parameter has a very small impact on the optimal accuracy. The main factor to be weighed in choosing this parameter lies in balancing communication gains with model stability.



(a) FedAvg (b) FedDyn

(c) FedProx (d) MOON

Figure 3. Accuracy in different number of global rounds on Fashion-MNIST with increased number of local epochs

Figure 4 illustrates the accuracy of the four federated learning algorithms based on the CIFAR-10 dataset after increasing the scale of clients. FedDyn shows a more pronounced advantage with the rise in the scale of clients and achieves the highest accuracy of 0.4 under the four federated learning frameworks. FedProx has the most stable convergence process but ultimately achieves the lowest accuracy. FedAvg and MOON perform close to each other, roughly between FedDyn and FedProx. In addition to this, Table 1 shows how efficiently these frameworks run in time. MOON requires the largest increase in time as the number of clients increases. All four methods show varying degrees of decrease in accuracy, with FedDyn being the least affected.

On the whole, the performance of FedAvg and FedProx is comparatively similar. the performance of

FedProx is slightly more stable than that of FedAvg in the test. Nevertheless, the running time of FedProx is slightly increased due to its need to aggregate the client data that has not completed the calculation. It is also worth mentioning that this experiment does not fully demonstrate the advantage of FedProx in terms of tolerance of heterogeneous systems. MOON does not show a significant advantage in the test, and its performance suffers more with the number of clients in the test, probably due to the fact that the method introduces additional calculation over loss function. FedDyn has a relatively unstable training process and has its advantage when the scale of clients rises. The fact that the model for each device is required to converge towards the global optimum during the training process may be one of the contributing factors.

(a) FedAvg (b) FedDyn

(c) FedProx (d) MOON

Figure 4. Accuracy in different number of global rounds on CIFAR-10 with increased number of local epochs

Table 1. Accuracy and time efficiency on CIFAR-10 with different number of clients

	original (5)		**extra clients (15)**	
	Time cost	**Accuracy**	**Time cost**	**Accuracy**
FedAvg	100% (85.67s)	0.47	350%	0.40
FedDyn	122%	0.53	422%	0.48
FedProx	110%	0.51	372%	0.38
MOON	187%	0.47	556%	0.40

4. Discussion and Conclusion

Federated Learning acts as a distributed learning solution for privacy and training data characterization

under real-world applications. The framework fully exploits the trend of large total size of client datasets and stronger client computational performance while satisfying privacy. Since its introduction, federated learning has developed various branches to suit different application environments and has been embraced by a large number of researchers and organizations. On the one hand, with the help of federated learning architecture, clients can overcome the bottleneck of small local datasets and obtain more accurate models and higher efficiency. On the other hand, the global model indirectly applies data generated by huge collection of edge devices under the condition of minimizing privacy issues.

In this paper, the performance of several federated learning algorithms is tested in the same environment. The performance of these frameworks in different contexts is analyzed. In subsequent research work, federated learning will be further optimized to address issues such as its heterogeneity. On the other hand, considering the outstanding scalability of federated learning frameworks, progress in other fields is likely to be applied in the improvement over performance of federated learning in specific dimensions, including scheduling on distributed machine learning and new approaches to data privacy.

References

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In Artificial intelligence and statistics, pages 1273– 1282. PMLR, 2017.

[2] Jakub Koneˇcny`, H Brendan McMahan, Felix X Yu, Peter Richt´arik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. arXiv preprint arXiv:1610.05492, 2016.

[3] Anusha Lalitha, Shubhanshu Shekhar, Tara Javidi, and Farinaz Koushanfar. Fully decentralized federated learning. In Third workshop on bayesian deep learning (NeurIPS), volume 2, 2018.

[4] Jack Goetz, Kshitiz Malik, Duc Bui, Seungwhan Moon, Honglei Liu, and Anuj Kumar. Active federated learning. arXiv preprint arXiv:1909.12641, 2019.

[5] Enmao Diao, Jie Ding, and Vahid Tarokh. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. arXiv preprint arXiv:2010.01264, 2020.

[6] Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. Personalized cross-silo federated learning on non-iid data. In Proceedings of the AAAI conference on artificial intelligence, volume 35, pages 7865–7873, 2021.

[7] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H Yang, Farhad Farokhi, Shi Jin, Tony QS Quek, and H Vincent Poor. Federated learning with differential privacy: Algorithms and performance analysis. IEEE Transactions on Information Forensics and Security, 15:3454–3469, 2020.

[8] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems, 2:429–450, 2020.

[9] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10713–10722, 2021.

[10] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems, 2:429–450, 2020.

[11] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. arXiv preprint arXiv:2111.04263, 2021.