Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/39/20230591

Federated learning method for dynamic weight adjustment: An in-depth analysis of FedDyn algorithm

Rongchuan Cao

Qi lu University of Technology,Jinan250353,China

jmoore32028@student.napavalley.edu

Abstract. With the gradual enrichment of the application of deep learning in daily life, distributed machine learning has played a greater role in daily life, among which federated learning and its optimization algorithm FedDyn algorithm have begun to receive widespread attention. After introducing federated learning and traditional algorithms, this paper starts with the operation mode and framework of the FedDyn algorithm, takes the combination method of port selection strategy and global distribution as an example, analyzes the characteristics of some links in the operation process and uses the innovation points of the FedDyn algorithm : Dynamic weight adjustment is the entry point, which illustrates the advantages of the FedDyn algorithm in current use, and also points out some of the current problems. Finally, from the aspect of market application, the current application status of the algorithm is introduced from the aspects of personalized recommendation, medical treatment and finance, and the future development of the algorithm is prospected.

Keywords: Federated Learning, FedDyn Algorithm, Privacy Protection, Dynamic Weight Adjustment.

1. Introduction

1.1. Introduction to Distributed Machine Learning and Federated Learning

Distributed machine learning is a processing method that runs a large number of machine learning methods and computing task data on multiple computing nodes [1].

Since multiple devices are working at the same time, it avoids the disadvantage of traditional machine learning being limited by the performance of a single device. At the same time, the distribution method of the master node and multiple working nodes facilitates the data distribution of model training and improves the training efficiency. As an emerging machine learning method, federated learning improves data privacy and security under the condition of distributed machine learning, and integrates the update of the global model [2,3].

Compared with traditional machine learning, federated learning no longer needs to aggregate all data in one server, but allows model training on the device side.

1.2. Limitations and Problems of Traditional Federated Learning Approaches

First of all, because traditional federated learning does not use the differential privacy method, it has great hidden dangers in terms of privacy and security, and it is easy to be maliciously attacked to discover the user's privacy.

Second, there is a lack of unified standards and cooperation mechanisms, which leads to low reliability of traditional federated learning in multi-device collaboration, resulting in biased data selection. At the same time, traditional federated learning attaches the same weight to each device, which can easily cause some devices to be overloaded or some devices to be idle, resulting in a waste of resources and affecting resource utilization and efficiency.

Finally, due to the need to upload to the central server, due to differences in equipment performance in different regions, limited by network resources and the server itself, redundant costs may occur.

1.3. Introduction and significance of the FedDyn algorithm

The FedDyn algorithm is an optimization algorithm for federated learning. Its full name is "Federated Dynamic". Devices in areas with large differences will not only assign weights based on data samples, causing one party to be ignored. At the same time, due to the real-time adjustment capability of dynamic allocation, it can more flexibly and accurately adapt to changes in data distribution during model training. Finally, because of the introduction of some algorithms of differential privacy, the privacy security of users is effectively improved.

The FedDyn algorithm has greatly accelerated the speed and convergence speed of global model training and proposed a new idea of dynamic adjustment in the traditional federated learning algorithm.

1.4. Overview of Article Structure

This article is divided into six parts.

The first part introduces the relevant knowledge of deep learning.

The second part is related research, which cites many documents to explain the origin, development, and partial application of the algorithm.

The third part introduces the principle and operation process of the FedDyn algorithm in detail and explains in detail the port selection strategy and global update method of the algorithm.

The fourth part is the advantages of the FedDyn algorithm, which explains the improvement of model training by dynamic weight adjustment, which is a feature of the algorithm.

The fifth part of the application scenario of the FedDyn algorithm explains the current market application status in terms of user personalized recommendation, medical care, and finance.

Finally, the sixth part of the conclusion summarizes the whole article, puts forward the current problems, and looks forward to the future. Algorithm development.

2. Related research

2.1. Federated learning overview and development history

Federated learning is a concept proposed by Google in 2016. In 2017, an article proposed the initial concept of federated learning and introduced an algorithm called FedAvg [4]. After improving communication efficiency through local training and aggregation schemes, etc., it has received widespread attention [5]. At present, great progress has been made in the personalized integration model of mobile applications and the financial field. Now, while the training effect is gradually improving, it can also realize cross-platform and device model sharing and collaboration [6].

2.2. Research status in the field of dynamic model personalized recommendation

Next, through three articles, we will introduce the optimization of its personalized recommendation in the field of dynamic model federated learning.

The first article uses federated learning from the perspectives of personalized service systems and intelligent service systems to demonstrate the advantages of federated learning in improving data accuracy through cases such as smart assistants, but at the same time, there are privacy leaks and poor communication capabilities hidden dangers [7].

The second article uses an improved bias_RSNMF method converts user behavior into user preference for items, and collects data by constructing a user rating matrix, which effectively solves the problem that the rating matrix is difficult to obtain. The fuzzy recommendation when accurately describing one's own needs and preferences is improved to improve the accuracy of the scoring matrix [8].

In the third article, with the widespread use of federated learning in different fields, it is pointed out that there is still a large research space on communication efficiency, privacy protection, unbalanced data, and other issues. After improving the solution proposed in the article, carry out After comparison, it is found that compared with the original method, there is a significant improvement. Finally, the article puts forward new views on different future application scenarios such as cross-device learning or cross-domain learning [9]. The third article puts forward new views on different future development directions of federated learning, such as cross-device learning or cross-domain learning, after proposing solutions to problems such as communication efficiency, privacy protection, and unbalanced data [9].

3. Related research

3.1. Federated learning overview and development history

Federated learning is a concept proposed by Google in 2016. In 2017, an article proposed the initial concept of federated learning and introduced an algorithm called FedAvg [4]. After improving communication efficiency through local training and aggregation schemes, etc., it has received widespread attention [5]. At present, great progress has been made in the personalized integration model of mobile applications and the financial field. Now, while the training effect is gradually improving, it can also realize cross-platform and device model sharing and collaboration [6].

3.2. Research status in the field of dynamic model personalized recommendation

Next, through three articles, we will introduce the optimization of its personalized recommendation in the field of dynamic model federated learning.

The first article uses federated learning from the perspectives of personalized service systems and intelligent service systems to demonstrate the advantages of federated learning in improving data accuracy through cases such as smart assistants, but at the same time, there are privacy leaks and poor communication capabilities hidden dangers [7].

The second article uses an improved bias_RSNMF method converts user behavior into user preference for items, and collects data by constructing a user rating matrix, which effectively solves the problem that the rating matrix is difficult to obtain. The fuzzy recommendation when accurately describing one's own needs and preferences is improved to improve the accuracy of the scoring matrix [8].

In the third article, with the widespread use of federated learning in different fields, it is pointed out that there is still a large research space on communication efficiency, privacy protection, unbalanced data, and other issues. After improving the solution proposed in the article, carry out After comparison, it is found that compared with the original method, there is a significant improvement. Finally, the article puts forward new views on different future application scenarios such as cross-device learning or cross-domain learning [9]. The third article puts forward new views on different future development directions of federated learning, such as cross-device learning or cross-domain learning, after proposing solutions to problems such as communication efficiency, privacy protection, and unbalanced data [9].

3.3. Comparison and analysis of other federated learning algorithms

Compared with the traditional FedAvg algorithm, in terms of algorithm performance, the convergence speed of the FedDyn algorithm is faster. FedDyn algorithm is more complex than FedAvg when written to run. At the same time, in terms of weight allocation, the FedAvg algorithm adopts a fixed weight allocation scheme, while the FedDyn algorithm adopts a dynamic weight adjustment method, which can more flexibly and accurately allocate weights according to training requirements, and obtain more ideal results.

4. FedDyn Algorithm Introduction

4.1. Algorithm Principles and Process

The Federated Dynamic Regularization Algorithm (FedDyn) proposes “a dynamic regularizer for each device at each round, so that in the limit the global and device solutions are aligned” [10]

The principle can be understood as follows:

First, each participant selects the local model that best meets the requirements at the device port as the training model, and after the participant selects the best model according to the requirements of the model training indicators, uploads it to the central The server performs model aggregation. After aggregation, according to the method of dynamically adjusting weights, dynamic weight distribution is performed with contribution and reliability as weights for training, and then the training data is returned to each device port. The local data has certain changes, so in the end, the data of different ports will be shared to balance the final results.

Referring to Figure 1, the specific operation process can be obtained according to the principle:

First, assume that there are three different devices, namely device1 device2 device3, and initialize them to the same model, namely model A.

After that, the data content is divided into several subsets, each device can only be accessed by itself, and the device that gets the new data set is set as device A device B device C.

Then start local training for each device, that is, use different data sets for each device to train the model to obtain train A train B train C.

After the training, the model obtained by the training result is uploaded to the central server, and then the central server aggregates the parameters of the uploaded model to obtain new global model parameters.

Send the obtained new global parameters to the respective devices to update the original data, and use this data as the new data for the next round of training.

Before the output, judge whether the training stop condition is satisfied. If it is satisfied, the result will be output and the training will end. If the condition is not met, the next round of iteration will be performed with new data. Repeat the iteration until the output result is reached, then stop the iterative output.

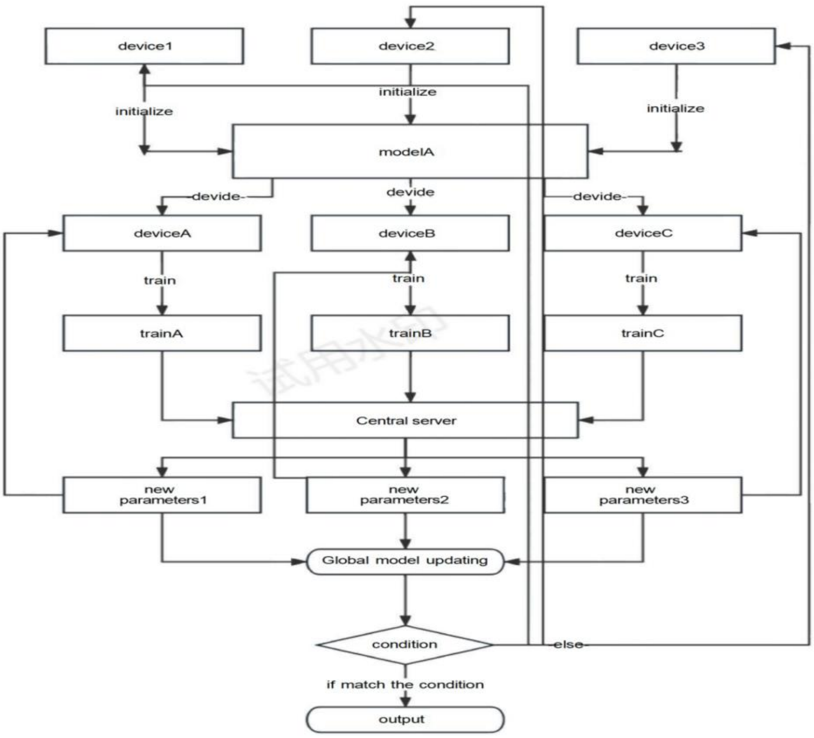

Figure 1. FedDyn Algorithm running flow chart(Picture credit:Original).

4.2. Detailed port selection strategy

The port selection of the FedDyn algorithm needs to consider two different indicators after the system is initialized.

One is the contribution of the data update of different devices to the model update, and the other is the degree of trustworthiness of the port participants. The combination of the two Adds weights to the data model according to requirements, and after determining the weights, updates the data according to the weighted model. Not only that, but the algorithm also evaluates and evaluates indicators for ports including training speed, number of training rounds, and convergence speed. If the weight result is lower than the evaluation or average level, the FedDyn algorithm will reassign the port task and assign it to a port with a higher performance index to complete more training tasks, to ensure the best training results as much as possible

4.3. Global model merging method

After the new data results are obtained, the global model update of the FedDyn algorithm will undergo a dynamic selection of participants, and model training will be allocated according to the performance of the equipment itself. After the dynamic adjustment is completed, the model data will be updated and re-weighted to ensure that the impact of different data set sizes on the overall model will also change according to the data set size. The impact of devices with more data sets on model updates will also be Larger, to achieve the logic that large data sets have a greater impact on the overall model.

5. IV. FedDyn Algorithm advantage

5.1. Advantages of port selection strategies

Compared with the port selection of some traditional federation algorithms, the FedDyn algorithm can dynamically select the appropriate port for model training, reducing the occurrence of port device overload or idle phenomena, improving resource utilization efficiency and effectively increasing flexibility. and accuracy.

At the same time, the fusion of the two indicators of contribution and participant reliability makes the FedDyn algorithm have more weight parameters to be considered in port selection. For heterogeneous model training, even in the case of large differences in regional data, can still conform to the logic that more contributed data accounts for a large proportion of the model, and at the same time allocate more private data to trusted ports, effectively protecting the privacy and security of participants.

5.2. The meaning of dynamic weight adjustment

In terms of ports, dynamic adjustment enables each port device to adjust the weight in time according to the contribution and reliability of each node port. Such dynamic changes can enable port devices with better performance to obtain corresponding data sizes and improve their training performance. efficiency. The dynamic weighting method can change the weight in a time when the port device fails or the performance deteriorates, and the port can be replaced flexibly, which reduces the adverse impact of the fault on the overall model.

On the other hand, the dynamic adjustment of the weight distribution gives full play to It maximizes the utilization of different port nodes, greatly avoids the extreme phenomenon of some ports being overloaded and some ports being limited, fully utilizes the performance of all ports, improves resource utilization, and saves the loss of part of the calculation data. In terms of global model updates, the weight of the data transmitted from the port to the central server is adjusted, which reduces the time for training and evaluation. When the new data is returned to each port device, it can better adapt to changes in environmental conditions, avoiding the If the device is damaged and the overall model training fails, the data set on the device will be dynamically redistributed to maintain the stability and accuracy of the global model.

5.3. Advantages of data privacy protection

The FedDyn algorithm itself is an algorithm under the framework of federated learning. Under the security protection of federated learning, it also starts model training for each communication port after anonymous processing. At the same time, the introduction of differential privacy technology has also improved its protection of user privacy. While using noise interference, it also hides the sensitive data privacy of users. It also strengthens the anti-interference ability of the global model and further improves the performance of the algorithm. This makes The FedDyn algorithm superior to the privacy security of the traditional FedAvg and other algorithms in terms of data privacy protection.

On the other hand, due to the method of dynamically adjusting weights, if strictly confidential data is used, the FedDyn algorithm can increase the weight of reliability for trusted devices at the port, and the dynamic adjustment also enables each training port to be more timely and strictly controlled Update the model, and do as much precaution as possible. While improving data security, these improvements are also conducive to improving the generalization ability of the federated learning model and improving the overall performance of the model.

6. FedDyn Algorithm Application Scenarios

6.1. Personalized Recommendation System

Compared with other federated learning algorithms, the FedDyn algorithm has the advantage of adopting a differential privacy protection method [11], and due to the dynamic weight adjustment of the reliability, it will be more vulnerable to malicious attacks based on the server or client than other algorithms. Algorithms are more protective [11].

On the other hand, the recommendation for user preferences can also be changed in time due to dynamic weight adjustment. Compared with traditional optimization algorithms, new personalized recommended products can be generated according to changes in demand faster, meeting user needs and timeliness.

DOI: 10.54254/2755-2721/39/20230591

6.2. Healthcare

In the medical field, federated learning makes it easier to collect a large amount of patient information than doctors to increase the data set, the training time is short, the training cost is low, and the accuracy rate is high. While reducing labor costs, it is beneficial to reduce the area due to data sharing The difference is large, and the level gap between regions with backward medical level. At the same time, since the private data of each medical institution is not required to interact, it only needs to be trained on the same model, which greatly protects the internal data of each medical institution and also protects the privacy of patients [12]. With the continuous generalization of federated learning, FedDyn, as a type of federated learning algorithm, is widely used in the medical industry.

6.3. Financial risk control

With the current rise of financial fraud, financial companies need more secure technology to provide financial services to customers while protecting customer privacy [12]. As a model algorithm for multi-device joint learning, federated learning can effectively deal with a variety of different fraud training methods. Among them, the FedDyn algorithm can be used to establish a model for joint construction of fraud detection by different financial institutions, and it is also used for the training of models such as transaction monitoring [13,14]. The training of these models is easy to obtain prevents different ways of fraud, and improves the monitoring and computing capabilities of the equipment. Because of the outstanding privacy protection performance of this algorithm, it can also better protect the privacy of a large number of user data of each institution, so this algorithm has been widely popularized and applied in the field of financial risk control.

7. Conclusion

7.1. Summary and Review of FedDyn Algorithm

As an optimization algorithm for federated learning under distributed machine learning, there are certain differences between the FedDyn algorithm and the traditional FedAvg algorithm in terms of weight adjustment and other aspects. The core of the FedDyn algorithm is to introduce a dynamic deep neural network structure and modify the required model in multiple iterations in the face of large differences in multi-party data sets. Its innovation is that dynamic weight adjustment improves the flexibility and accuracy of the algorithm, and at the same time hides private data through noise interference. In a word, the FedDyn algorithm, as an optimization algorithm for federated learning, can adapt to the performance and data quality of different devices and improve the performance of the model. and generalization ability.

7.2. Algorithm advantages and limitations discussion and future development direction and outlook

As an optimization algorithm, FedDyn has great advantages in dynamic allocation, unbalanced data allocation, and non-independent allocation. Due to the adoption of a dynamic weight adjustment algorithm, compared with other optimization algorithms, the FedDyn algorithm will be more flexible in the use of each port device, and the reasonable distribution of data training according to the performance of the device will not cause waste of resources, but also improve the accuracy and flexibility of training. At the same time, in terms of privacy and security, the FedDyn algorithm has also been significantly improved. However, as an optimization algorithm with a short research time, since it needs to connect to the central server and upload training update data, it has high requirements on the central server, so if the communication signal is poor, it may affect the accuracy and time of the training results. Failure of the central server can have a huge impact on the accuracy of the algorithm. At the same time, compared with traditional optimization algorithms such as FedAvg, this algorithm has a higher degree of complexity in operation and programming. At the same time, due to the use of the weight adjustment strategy of the contribution degree, sometimes the proportion of the model accounted for by the device port with huge data may make the contribution to the global model too obvious, resulting in the contribution of other devices being ignored, resulting in serious data errors. At the same time, due to the particularity of dynamic adjustment, it is difficult to use this algorithm in combination with other optimization algorithms in practical applications, resulting in isolated cases of use.

At present, this algorithm is widely used in the financial industry, and it is also used in personalized recommendation and medical treatment. With the further development of federated learning, it is believed that this algorithm will break through the constraints of isolated algorithms. In the future, the system linkage and personalization in smart cities The optimization and protection of the recommendation algorithm play a huge role. As an emerging optimization algorithm, the future application scenarios will be broader, and it also provides a new way of thinking for the further development of federated learning.

References

[1] Shu Na, Liu Bo,Lin Weiwei,Li Pengfei,Overview of distributed machine learning platforms and algorithms[J].Computer science,2019,46(03):9-18.
[2] Yang Q., et al. “Federated Machine Learning: Concept and Applications”. IEEE Internet of Things Journal, 2019
[3] Li T., et al. “Federated Learning: Challenges, Methods, and Future Directions”. arXiv preprint, 2020.
[4] McMahan,H.Brendan, “Communication-efficient learning of deep networks from decentralized data.” arXiv preprint arXiv:1602.05629 (2016).
[5] Konečný,Jakub. “Federated learning: Strategies for improving communication efficiency.” NIPS Workshop on Private Multi-Party Machine Learning (2016).
[6] Li,Tianxing. “FedAvg-Mobile: ensemble model personalization for mobile applications via federated learning.” Proceedings of the 26th ACM SIGKDD International Conference onKnowledge Discovery & Data Mining. 2020.
[7] Zhao, Y., Li, A., Tuo, X., & Yan, X. Federated Learning for Personalized and Intelligent Services: A Comprehensive Survey. IEEE Journal on Selected Areas in Communications, 38(8), 1697-1711.. (2020)
[8] JIA Jun, ZHANG Bin, LI Zhiyuan, WEI Wei, WEI Hao✉,Personalized recommendation algorithm based on user behavior analysis,Journal of Intelligent Science and Technology, 1(4): 421-426 doi: 10.11959/j.issn.2096-6652.201947, 2019,
[9] Lian, P., Zhang, C., Zhang, H., & Sun, M. . Federated Learning: Challenges, Methods, and Future Directions. IEEE Signal Processing Magazine, 37(3), 50-60.(2020)
[10] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In International Conference on Learning Representations, 2021.
[11] FENG Han, YI Huawei, LI Xiaohui, LI Rui. Review of Privacy-Preserving Research in Recommendation Systems[J]. Journal of Frontiers of Computer Science and Technology, 2023, 17(8): 1814-1832.
[12] Chuanxin ZHOU, Yi SUN, Degang WANG, Huawei GE. Survey of federated learning research[J]. Chinese Journal of Network and Information Security, 2021, 7(5): 77-92.
[13] Chen, X., Wu, Y., Liu, X., Yannakakis, G., & Yuan, C. (2020). Federated Dynamics for Distributed Machine Learning. arXiv preprint arXiv:2002.08908.
[14] Liu, Y., Huang, S., Luo, L., & Shen, X. (2021). Federated Learning for Credit Scoring: A Privacy-Preserved Approach. IEEE Transactions on Industrial Informatics, 17(1), 543-553.