IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 25, Issue 3, Ser. II (May. – June. 2023), PP 46-49
www.iosrjournals.org

Enhancing Privacy In Edge Devices Using Federated Learning

Rohan S
Student, Department of Electronics and Communication, BNMIT, Bengaluru
Suhas R Vittal
Student, Department of Electronics and Communication, BNMIT, Bengaluru
Prajwal Anand
Student, Department of Electronics and Communication, BNMIT, Bengaluru
Sujaya B L
Assistant Professor, Department of Electronics and Communication, BNMIT, Bengaluru

Date of Submission: 09-06-2023 Date of Acceptance: 19-06-2023

I. INTRODUCTION

In today's world data is something that fuels the growth of a company. In order to improve growth of a company collection of previous data and its analysis play a vital role for use in business decision and strategy and other purposes. We have seen a lot of tremendous progress in the fields of Machine Learning. This was in order to enhance the computational power and efficiency of the devices. Standard machine learning approaches necessitate storing the training data in data centres. It is an issue for the organisation and the people connected to it if any unaffiliated candidate gains access to the data centre. According to Bloomberg, data breaches increased by 40% in 2016 compared to 2015. The costs of such incidents can be very high, and in some cases, they may jeopardise a company's ability to continue operations. As a result, it is critical for organizations to put its finger on threats and mitigate them. Various types of data are more or less worthwhile to third forces and embody different levels of risk to an organization. Third forces who scrutinize the data are to the information. Personal or confidential data can be used for illegal purposes. The competitors may illegally utilize competitive information or sell it to any competitive organizations to thwart the plans, and leaked legal information may harm legal position. IT security data is a lucrative target in and of itself since it gives unauthorised parties access to all other forms of data on your system.

So, in order to protect the private data, in this paper, we will put Federated Learning into practise, a machine learning (ML) strategy in which the private data does not

leave the edge devices of the consumers. We have chosen this strategy because better compute efficiency and user data security are provided by this new strategy.

Federated learning works without the requirement for user data to be stored on the cloud. Federated learning is the process of training several machine learning models on edge devices (known as clients) and then pooling the results of all such models into a single model that remains at a server. As a result, a model is trained utilising ground truth data on the devices themselves, and only the learned model is shared with a server. Only this model update is transferred to the cloud through encrypted connection, where it is instantly averaged with other user updates to enhance the shared model. No individual updates are kept in the cloud, and all training data remains on your device. In this manner, it may be swiftly averaged with changes from other users to enhance the model which was shared. Federated learning can be approached in a variety of ways, with Flower being one of them.

This paper makes use of the Flower Framework. Flower is a framework for building federated learning systems. The design of Flower is based on a few guiding principles: Federated learning systems vary greatly depending on the use case. According to the requirements of each unique use case, Flower supports a broad variety of various configurations. Flower was created with AI research in mind since it was inspired by a research study at the University of Oxford. To create brand-new cutting-edge systems, several components can be enhanced and replaced. The goal of Flower's writing was fidelity. The community is urged to read the codebase and add to it. This project aims to implement federated learning on edge device, it is expected that the FL maintains decentralization of data with similar accuracy levels and slightly enhanced communication efficiency compared to the ML model while federated learning does not answer all the questions regarding data security it does help avoid sending some of user's critical data such as keystrokes, biometrics, locations and content engagement to a centralized location or cloud and such data could remain local while not sacrificing performance.

The paper is organized in the following manner. Section 2 speaks about the related work. The information about the datasets, software, algorithm, methodology and the implementation of the project is summarized in the Section 3 of the paper. In the Section 4 of the paper the results are discussed which includes accuracy of the model.

II. RELATED WORK

Novel False Rate (FR) paradigm, federated face recognition to design the FedFace method to train FR models via federated learning Partially Federated Momentum to correct client drift in federated training, Federated Validation to enhance the model's generalization ability. The accuracy of the model was 97.6%. Using MNIST dataset which consists of 60,000 training image samples and 10,000 test image samples, every sample is a 28*28 grayscale image having a handwritten number in the range between 0 and 9 and then simulates FL by constructing the model using the averaged parameters of multiple locally trained model parameters using the architecture of CNN parameters. In order to train FR models using federated learning, a unique FR paradigm called federated face recognition was presented. To correct client drift in federated training of FR while maintaining the training's effectiveness, they have created a Partially Federated Momentum. To increase the models' capacity for generalisation in federated learning for FR, they have developed federated validation. Numerous tests were run to assess and analyse our approach, which validates the potency of the Partially Federated Momentum (PFM) and Federated Validation (FV) algorithms that were suggested. In results under different party numbers, FedAvg got worse performance, but PFM was still comparable to the baseline. Flower – a complete FL framework that differs from existing platforms by providing new facilities for carrying out large-scale FL experiments and taking into account scenarios for richly heterogeneous FL devices. The tests demonstrate that Flower can run FL experiments with up to 15M clients utilising just a pair of top-tier GPUs. They have concluded that all our experimental data matches the expected complexities of SecAgg and SecAgg+. Evaluation of the on-device FL system expenses and discussion of how more effective FL algorithms might be created using this quantification. Their analysis focuses on estimating the system expenses related to using different edge devices to execute FL. Two datasets were used in the evaluation, namely CIFAR-10 and Office-31. The accuracy was 67% and 80% respectively. It was discovered that we could train a more precise object recognition model by increasing the number of clients. The model is exposed to a wider variety of training examples as more clients take part in the training, improving its generalizability to unknown test data. The more clients we use, the higher FL's overall energy consumption, negating the accuracy improvement at the cost of high energy usage.

III. METHODOLOGY

To evaluate federated learning for edge devices, we had to compare it to an existing benchmark, namely centralized learning, where data is gathered from the users and fed to a centralized model and then trained and re-trained as more data was gathered. This data was gathered from user devices, and there used to be a data transfer from the user to company servers, which was a potential data breach point. As a result, we first trained a centralized ML model on two widely used databases, CIFAR-10 and MNIST, where the model and data were on the same server.

MNSIT is a database created by the National Institute of Standards and Technology that contains at least sixty thousand images of various handwritten digits. The database has upwards of 60000 images to cover the various styles in which digits may be written. The goal of using an MNIST database is to evaluate the performance of the traditional setup for training models using an already "solved" example, so we already have benchmarks to compare model performance. To evaluate the MNIST database, we use sequential models, which are a type of convolution neural network that is adept at image classification by detecting patterns in images. This model's accuracy level is approximately 97%.

CIFAR-10 is a database created by the Canadian Institute for Advanced Research that contains at least ten images from ten different categories; the categories are chosen at the discretion of the database's intended user. Flowers, cats, and dogs are some popular categories. The goal of using a CIFAR-10 database is to evaluate the performance of the CNN, and it is a step up in complexity compared to the MNSIT database, which is less computationally demanding for the model. We use a convolution neural network, which is an ML model developed and adept at handling image classification, to evaluate the CIFAR-10 database and achieve an accuracy level of 64%.

Two models are trained in the conventional setup of having model and data in the same server and their performances are noted down. They will act as a benchmark to be compared with their counterparts trained in the federated method.

A federated network is made up of edge devices or user devices that are connected to a centralized server via the internet. Each edge device is pre-loaded with a base model, and the model is trained on data generated locally, yielding an updated local model; only updates over the base model are sent to the central server as weights. The Federated average algorithm, which is an efficient communication algorithm, collects all of these weights

from all of the edge devices and aggregates them to form a new global model, which is sent to all of the edge devices to act as the new updated base model.

As shown by the bar graphs, the database in federated MNIST has a heterogeneous split between two edge devices to train the model. Initially, the aggregation efficiency is around 18%, but after ten iterations of aggregation, the efficiency of the FedAvg algorithm settles at 86% and a loss factor of 14%, with the classification model having an accuracy of 95%, indicating that despite being trained on different data the final accuracy is comparable with centralized model.

The database in federated CIFAR has different categories for the two edge devices to train the model, initially, the aggregation efficiency is around 18%, but after twenty iterations of aggregation the efficiency of the FedAvg algorithm settles at 64% and the loss factor of 36% with the classification model has an accuracy of 65%, this indicates that federated learning performs better with diverse data and helps in bias reduction, promoting the ability to deploy edge devices.

IV. RESULTS AND DISCUSSION

For the federated approach to be deployed for both the CIFAR and MNIST datasets, we have set up a federated server along with two clients designated Client1 and Client2, which function as two edge devices. For the two datasets, two distinct federated models have been put up, and the accuracy and effectiveness of the individual models are being compared.

The federated server receives the parameters from the clients in order to assess performance and update the global model on the server, which is then distributed to all edge devices linked to this server.

The convolution neural network method is used by the centralised client to examine and recognise the numbers in the database. Here, we have provided the client with the entire undistributed MNIST dataset from 0 to 9 for the centralised MNIST model. With the help of this method, we were able to achieve an accuracy of 97–98%.

There are 10 different classes of images in the Centralized CIFAR model dataset. The centralised CIFAR algorithm has obtained an accuracy of 63% after training on all 10 classes of images that are present in the dataset.

Since we cannot predict when or how long a client will wait before requesting a service from the server, the federated server (flower server) must be started first in the federated learning model configuration. As a result, before beginning the client, we must first start the server.

We must start the clients one at a time after the server has been started. The flower framework assists in establishing a client-server connection utilizing the ephemeral port as its local port and connecting to the server port as the clients start the federated learning model to run on the client devices.

Along with the federated Server, we have defined two clients for our federated MNIST algorithm: Client1 and Client2.

The MNIST dataset is also shared equally between the two clients; Client1 receives data from the same MNIST dataset with numbers 0, 1, 2, 3, and 8; Client2 receives data from the same MNIST dataset with numbers 4, 5, 6, and 7, respectively.

The client 1 algorithm analyses handwritten MNIST data with the numbers 0, 1, 2, 3, and 8, whereas the client 2 algorithm analyses handwritten MNIST data with the numbers 4, 5, 6, and 7 and 9.

After a few iterations of aggregating the client parameters that are provided to the server through local port, the federated server has achieved an accuracy of 86% according to the server algorithm. The federated client algorithm received the global update from the federated server, and following the update, this federated client algorithm now has an accuracy of 95%.

Clients Client 1 and Client 2 have both received the server's global update. The dataset is now trained using this update process on the edge device rather than transferring the data to a centralized server, resulting in a more effective and stable model.

In federated MNIST, the database is heterogeneously split between two edge devices to train the model. Initially, the aggregation efficiency is around 18%, but after ten iterations of aggregation, the FedAvg algorithm's efficiency settles at 86% and loss factor of 14%, with the classification model having an accuracy of 95%. This shows that even though the final accuracy was trained on different data, it is comparable to the centralized.

We have defined two clients, Client1 and Client2, as well as the server, in the federated CIFAR dataset. In contrast to the MNIST client, the federated CIFAR client does not share the dataset between clients. To train the federated client algorithm, we were provided the identical undivided dataset for both clients.

The global update was received by Clients 1 and 2 from the federated server, and the algorithm's accuracy was 65.8%.

After this update, the algorithm is utilized to train the datasets on the edge device, rather than transferring the data to a centralized server, to produce a more effective and stable model.

This shows that federated learning performs better with diverse data and aids in bias reduction, promoting the ability to deploy edge devices in federated CIFAR where the database has a different category for the two

edge devices to train the model. Initially, the aggregation efficiency is around 18% but after twenty iterations of aggregation, the efficiency of FedAvg algorithm settles at 65% and loss factor of 35% with the classification model having an accuracy of 65%.

To execute federated learning using the Flower framework, two separate datasets MNIST and CIFAR were used, along with a comparison of the two networks' performance metrics.

DATABASE		MNIST	CIFAR-10
Accuracy	Centralized	97%	65%
	Federated	95%	64%

V. CONCLUSION

Using the flower framework, we have proposed a centralized learning algorithm and a federated learning algorithm in this paper.

We have constructed a safe and secure network to train the machine learning model without transferring the data to a central server and have achieved accuracy that is close to a centralized network. We have successfully obtained a stable and accurate federated model from both datasets.

Because no private information is ever exchanged with a server, the federated system effectively optimizes while protecting user privacy. By efficiently exploiting the data already present on the device, without the data ever leaving the device, and just transmitting the model's parameters to the server, the suggested technique is possible to improve the performance of machine learning models in edge devices.

REFERENCES

[1]. "Privacy-Preserving Blockchain-Based Federated Learning For Iot Devices" By Yang Zhao,Jun Zhao, Linshan Jiang, And Rui Tan.
[2]. "Implementation Of Federated Learning On Raspberry Pi Boards" By Rini Apriyanti Purba And Wen Hao.
[3]. "Federated Face Recognition" By Fan Bai1, Jiaxiang Wu2, Pengcheng Shen2, Shaoxin Li2, And Shuigeng Zhou1 Fudan University.
[4]. "Federated Learning For Internet Of Things: A Comprehensive Survey" By Dinh C.
[5]. Nguyen, Ming Ding, Pubudu N. Pathirana, Aruna Seneviratne, Jun Li, And H. Vincent Poor, Fellow,Ieee.
[6]. "Prototype Of Deployment Of Federated Learning With Iot Devices" By Pablo García Santa Clara.
[7]. "Industrial Internet Federated Learning Driven By Iot Equipment Id And Blockchain" By Xu Zhang,1 Haibo Hou,1 Zhao Fang,2 And Zhiqian Wang.
[8]. "Flower: A Friendly Federated Learning Framework" By Daniel J. Beutel 1 2 Taner Topal 1 2 Akhil Mathur 3 Xinchi Qiu 1 Javier Fernandez-Marques 4 Yan Gao 1 Lorenzo Sani 5 Kwing Hei Li 1 Titouan Parcollet 6 Pedro Porto Buarque De Gusmao˜ 1 Nicholas D. Lane 1.
[9]. "On-Device Federated Learning With Flower" By Akhil Mathur 1 2 Daniel J. Beutel 1 3 Pedro Porto Buarque De Gusmao˜1javier Fernandez-Marques 4 Taner Topal 1 3 Xinchi Qiu 1 Titouan Parcollet 5 Yan Gao 1 Nicholas D. Lane 1.
[10]. "Fedfog: Network-Aware Optimization Of Federated Learningover Wireless Fog-Cloud Systems" Van-Dinh Nguyen, Member, Ieee, Symeon Chatzinotas, Senior Member, Ie.
[11]. "Generative Models For Effective Ml On Private, Decentralized Datasets" Sean Augenstein, H. Brendan, Mcmahan Daniel, Ramage Swaroop Ramaswamy, Peter Kairouz, Mingqing Chen, Rajiv Mathews, Blaise Aguera Y Arcas.
[12]. "Federated Learning For Mobile Keyboard Prediction" By Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Franc¸Oise Beaufays Sean Augenstein, Hubert Eichner, Chloe Kiddon, Daniel Ramage.
[13]. "Central Server Free Federated Learning Over Single-Sided Trust Social Networks" By Chaoyang He, Conghui Tan, Hanlin Tang, Shuang Qiu, Ji Liu.
[14]. "Group Knowledge Transfer:Federated Learning Of Large Cnns At The Edge" By Chaoyang He, Murali Annavaram, Salman Avestimehr.
[15]. "A Systematic Literature Review On Federated Learning: From A Model Quality Perspective" By Yi Liu, Li Zhang, Ning Ge,Guanghao Li.
[16]. "Federated Learning In Smart City Sensing: Challenges And Opportunities" By Ji Chu Jiang, Burak Kantarci, *Orcid, Sema Oktug And Tolga Soyata.
[17]. "Emerging Trends In Federated Learning: From Model Fusion To Federated X Learning" By Shaoxiong Ji, Teemu Saravirta, Shirui Pan, Guodong Long, Anwar Walid.
[18]. "Federated Learning For Internet Of Things: Recent Advances, Taxonomy, And Open Challenges" By Latif U. Khan, Walid Saad, Fellow, Ieee, Zhu Han, Fellow, Ieee, Ekram, Hossain, Fellow, Ieee, And Choong Seon Hong, Senior Member, Ieee.
[19]. "Federated Learning: A Survey On Enabling Technologies, Protocols, And Applications" By Mohammed Aledhari, Rehma Razzak, Reza M Parizi, Fahad Saeed.
[20]. "Federated Learning For Vehicular Internet Of Things: Recent Advances And Open Issues" By Zhaoyang Du, Celimuge Wu, Tsutomu Yoshinaga, Kok-Lim Alvin Yau, Yusheng Ji, Jie Li.
[21]. "Federated Learning On Clinical Benchmark Data: Performance Assessment" By Geun Hyeong Lee, Soo-Yong Shin.