Federated Learning - A step towards privacy
San Jose State University
Research Paper
CMPE 294

December 14, 2022

Charanveer Singh
SJSU ID: 011826308
Contact: Charanveer2014@gmail.com

***Abstract* -** This term paper is going to major on the topic of federated learning – a step towards privacy. First, the paper includes a brief introduction by shedding light on the topic. Afterward, the paper will discuss the concept of federated learning. To build this topic, the paper will also incorporate other supporting topics that go hand in hand with the topic of federated learning. To make the topic more captivating and resourceful for learning, the paper will include some advantages, disadvantages, and challenges associated with federated, centralized, and decentralized learning. The paper will outsource references from various peer-reviewed scholarly articles from accredited sources.

I. Introduction

Before going deep into the topic of federated learning, it is thoughtful to define what it is. Incorporating interpretations from various researchers, it can conclude that multiple clients or entities can work together to solve a machine learning problem in a federated learning environment, which is governed by a central server or service provider.. Each client's raw data is preserved locally in federated learning and isn't shared or made public.; intensive updates are projected for instant accretion to realize objectives. On the other hand, data is a set of values describing a situation, process, or system. Data can be collected through observation, experimentation, or survey. Data is essential because it can help us to understand the world around us. It can help us to find trends and to make predictions. Data can also help us make decisions by providing information we would not otherwise have.

II. Body

Data science and machine learning are crucial instruments in public policy, science, and the design of services and products. “Thanks to the increment in the availability of processing, storing and collecting vast quantities of data despite the act that centralized data collection can unveil individuals to privacy risks and organizations to legal risks if data management is appropriately controlled” [12]. Standard machine learning approaches need data centralization in a data center or a single machine [1]. In order to enhance its services, Google has built one of the most reliable and secure cloud infrastructures..

“With federal learning, mobile phones can collaboratively learn a collective prediction model while storing all training data on a device, decoupling the ability to do machine learning from the urgency of storing data in the cloud” [2]. This extends beyond the application of the local models that makes forecasts on mobile phones. Such as Device Smart Reply and the mobile Vision API by availing the model training to devices.

It operates in this manner: the device retrieves the current model and improves it by learning from the phone's data. It then summarizes the changes as a targeted update. Only the updated model is sent to the cloud using encrypted communication, where it is combined with other users' updates to improve the overall model. The device retains the training data and no individual updates are stored in the cloud.

In this model, the phone customizes itself to the individual user's usage and then the updates from many users are combined to create a consensus change to the collective model. This process is then repeated. Federated learning is known for its ability to create more effective models while also maintaining privacy. It also results in lower latency and lower power consumption.. “Another immediate benefit of this approach is that; in addition to giving an update to the collective model, the improved model on the phone is used immediately; powering the experiences personalized by how the phone user uses it” [12]. At present, federated learning is being trialed on Gboard, Google's keyboard app for Android. If a user chooses one of Gboard's recommended queries, the phone records information about the current context. Using this data, federated learning can suggest enhancements to the query suggestion model in future versions of Gboard. This approach allows for the improved creation of more effective models without compromising user privacy.

In keyboard prediction, a neural network is trained to predict the user's next character based on typed characters. Federated Learning is used to train a keyboard prediction model on many different devices without collecting any data from the devices. Federated Learning can train an accurate keyboard prediction model without all the data being collected in a single location. Likewise, Federated Learning can be used to keep the keyboard prediction model up to date as the user types, without the need to collect any data from the user.

To make ease federated learning, many technical and algorithmic challenges are solved. In a system of archetypal machine learning, optimization algorithm such as Stochastic Gradient Descent operates on huge dataset apportioned consistently across all servers in the cloud. In order to achieve the best results, iterative algorithms require fast and well-designed connections to the training data. When using federated learning, this data is distributed across millions of devices in a very fragmented manner. These devices often have slower connections with higher latency, which can be inconsistent and unreliable for learning. In order to overcome these challenges and make the most of federated learning, it is important to carefully consider the design of connections between the devices and training data. This includes optimizing the connections for speed and reliability, as well as ensuring that they are able to handle the large amounts of data being shared among the devices. By doing so, it is possible to create highly effective models using federated learning, even in challenging environments with limited resources.

The Federated Averaging algorithm is designed to address the challenges of latency and bandwidth limitations. It can train deep learning networks using 10-100 times less communication compared to the federated SGD version. The key idea is to use the powerful processors in modern mobile phones to calculate more effective updates, rather than small gradient steps. Because fewer iterations of these high-quality updates are needed to create a good model, training requires less communication overall. Additionally, uploading speeds are typically slower than downloading speeds, so this approach can further reduce the cost of communication by up to 100 times through the use of techniques like quantization and random rotations. These methods are particularly useful for training large networks, and there are also algorithms designed to excel at tasks like click-through rate prediction for high-dimensional, sparse, convex models.

The trained models are downloaded to the phone via a server. The server uses a higher-level TensorFlow model trained on all the different phone types to generate the device-specific models. This "meta-model" enables the server to generate models for any device that doesn't already have a model. The server also validates the downloaded models, checking that they do not contain any malicious code.

“Google uses Tensor Flow Federated to implement Federated Learning, and it is hoped that by contributing it as an open-source project, its adoption will accelerate beyond the research community” [2]. Federated Learning infrastructure is also built on TensorFlow Federated, making it easier to get started—it is called TensorFlow Federated Learning (TFF) [2].

The server creates a new model update in the form of a TensorFlow Saved Model and sends it to each client. The client uses this update to improve its local version of the model, resulting in a new local model update. This update is then sent back to the server, which incorporates it into the global model and sends the updated global model to all the clients. This process is repeated several times, typically 2-3 times, in order to improve the model. By using federated learning, it is possible to train deep learning models without compromising data privacy or requiring large amounts of data to be transmitted over the network.

The training and validation process is automated, which means the models are updated automatically as the training data changes. The system is also designed to work with multiple languages in parallel. In order to train the meta-model, a lot of data is needed. The data is collected from Gboard users who opt into the on-device training. The data is collected anonymously and is not used for any other purpose.

The device training system is designed to be privacy-preserving. The models are trained on the phone and never leave the device. The training data is collected anonymously and is not used for other purposes. The on-device training system is designed to be secure. The models are downloaded from a server and are validated before they are used.

Federated learning allows for the creation of effective models without storing user data in the cloud. To further improve this process, a safe aggregation protocol is being developed that uses cryptographic techniques to coordinate the server and decrypt the average update if more users participate. This approach allows for the creation of highly effective models without compromising user privacy.

III. Centralized Machine Learning

“There is no precise definition of centralized machine learning, but it generally refers to training machine learning models on a central server rather than on individual devices. This can make it easier to manage data and resources but can also create privacy and security concerns” [14].

ADVANTAGES

There are a few advantages to centralized learning; Centralized learning can be more efficient than decentralized learning since there is only one central point that needs to be updated rather than multiple points. Additionally, Centralized learning can be more accurate than decentralized learning since there is only one source of information that needs to be processed, rather than multiple sources. And lastly, Centralized learning can be more secure than decentralized learning since there is only one central point that needs to be protected, rather than multiple points. Centralized learning has advantages, but the system also has some limitations.

DISADVANTAGES

Increased cost: A centralized setup requires additional hardware and software, which can increase the overall cost of the system The course of outlining security requirements starts with the creation of business requirements. The security requirements are then used to create the security design which outlines how the system will be secured. The security design is used to create the security architecture which defines the security components and how they work together.

The importance of business requirements cannot be underestimated. Business requirements provide the foundation for the system and define the parameters used to decide whether the system is successful [7].

Increased complexity: A centralized setup can be more complex to manage and maintain than a distributed setup. Lack of scalability: A centralized setup can become a bottleneck if the load increases. It isn't easy to scale a centralized setup.

IV. Decentralized Machine Learning

“Federated machine learning is a distributed machine learning approach where data is stored and processed across a network of nodes rather than in a centralized location” [6]. This approach has several advantages, including improved scalability, privacy, and security. A traditional machine learning model is trained using data from a single source. Then data is processed by various feature extractors and fed into a machine-learning algorithm. The algorithm then produces predictions about the target variable that is being learned.

This is a very simplistic view of how traditional machine learning works. In practice, many more steps are involved, such as data pre-processing, feature selection, model selection, and model evaluation. The main problem with traditional machine learning is that it doesn't scale well. As the size of the data increases, so does the computational cost of training the model. This is why most traditional machine learning models are limited to data sets that are relatively small in size.

Another problem with traditional machine learning is that it is vulnerable to over-fitting. Over-fitting occurs when a model is trained on a data set too small. The model then learns the Noise in the data instead of the actual signal. This results in the model making poor predictions on new data. Finally, traditional machine learning models are often difficult to deploy and maintain. This is because they must be retrained every time the data set changes. This can be a time-consuming and expensive process.

ADVANTAGES

Decentralized learning has several advantages, including the following: Decentralized learning can give learners more control over their learning experience. With this type of learning, learners can choose when and where they want to learn. They can also choose the pace at which they want to learn.

Flexibility: Decentralized learning can be more flexible than traditional learning models. Learners can learn at their own pace and on their schedule. This can be especially beneficial for learners who have busy schedules or live in different time zones.

Cost-Effective: Decentralized learning can be more cost-effective than traditional learning models. Learners can access learning materials from anywhere in the world. They can also learn at their own pace, saving time and money.

Accessibility: Decentralized learning can be more accessible than traditional learning models. Learners can access learning materials from anywhere in the world. They can also choose the pace at which they want to learn [12].

DISADVANTAGES

Generalization: Decentralized learning can be challenging to generalize, as each learner may be working on a different data set. This can make it challenging to develop a general model that can be applied to new data.

Communication: Decentralized learning can also be challenging to communicate, as each learner may be working on a different set of data. This can make it challenging to share results or discuss strategies.

Coordination: Decentralized learning can be challenging to coordinate, as each learner may be working on a different data set. This can make it challenging to ensure that all learners are making progress or that the learning process is efficient.

Implementation Cost: Deploying and maintaining the infrastructure required to support it. This includes the cost of developing and maintaining the software required to manage the decentralized learning process and the cost of deploying and maintaining the hardware required to support it. Additionally, there may be costs associated with training staff to use the decentralized learning process and costs associated with marketing the decentralized learning process to potential users [2].

Decentralized learning enables Model Evaluation and Selection to be distributed among the agents or nodes in a cooperative way. The agents or nodes can be independent of each other and do not need a central coordinator. Decentralized learning is also used to improve the efficiency of resource utilization.

BEST PRACTICES AND APPROACHES

There are a few proposed solutions to the challenge of generalization in decentralized learning. “One solution is to use a federated learning approach, where a central model is trained on various data sources and then shared with the participating nodes” [15]. Another solution is to use a hybrid approach, where a central model is trained on various data sources and then shared with the participating nodes. The hybrid approach also uses a local model trained on the data from the participating nodes.

A subset of machine learning in which the goal is to fool a machine learning model. “TensorFlow - An open-source software library for numerical computation using data flow graphs. An open-source software library for numerical computation using data flow graphs Deep Learning might also solve the challenge of decentralized learning. A subset of machine learning uses a deep neural network” [16]. A Neural Network is a subset of machine learning, that uses an artificial neural network, a computer system created to simulate the workings of a human brain [7]. In this manner, a “Data Flow Graph is a directed graph; nodes represent mathematical operations, and edges represent the data that flow between them” [16].

Another proposed solution to the challenge of decentralized learning is the Training AI Model without touching the data. However, there exist some critics of this solution. Some scientists are against the possibility of training and learning AI without data. They claim that this is impossible since, for one to train a machine learning model, one must provide training data to the algorithm. The algorithm will then learn from this data in order to be able to make predictions on new data.

For privacy protection, “Federated learning is a good privacy protection technique because it allows data to be shared without revealing any personally identifiable information; however other challenges with decentralized learning are that the system; is difficult to manage and control, is vulnerable to cheating and fraud, it is decentralized and thus harder to coordinate” [16]. In addition, decentralized learning relies on peer-to-peer interactions, which can be unreliable and, therefore, less effective than traditional learning methods.

However, Which this phase of technological advancement, In five years, you may see federated learning being used more in healthcare and in other industries where data privacy is a concern.

V. Federated Learning

As earlier, discussed, “Federated learning is a type of machine learning where models are trained on data from multiple distributed devices” [16]. This can help to improve the accuracy of the models by using more data and can also help to protect the privacy of the data by keeping it on individual devices.

Federated learning is a distributed machine learning method that can be used on a variety of devices, including smartphones, sensors, and embedded systems. Unlike traditional machine learning, where data is typically stored on a central server, federated learning enables machine learning models to be trained on data that is distributed across multiple devices. This approach allows for the creation of more effective models without compromising user privacy or requiring large amounts of data to be transmitted over the network.

Federated learning has some advantages over traditional machine learning. First, it allows for training on data that is not centrally located [2]. This can be useful in cases where data is distributed across several devices, such as sensors, or where data is sensitive and cannot be centrally located, such as medical data.

Second, federated learning can train models on various devices, including mobile phones, sensors, and embedded systems. This is different from traditional machine learning, which is typically limited to training on data centralized on a server.

Federated learning has many applications. One application is training models on mobile devices. This is different from traditional machine learning, which is typically limited to training on data centralized on a server.

Another application is training models on sensitive data that cannot be centrally located, such as medical data. This is different from traditional machine learning, which is typically limited to training on data centralized on a server [5].

Federated learning is still in its early stages, and there is a lot of potential for growth in this area. One potential growth area is federated learning for other data types, such as text data.

Additionally, there is potential for federated learning to be used in more complex applications, such as predictive modeling and machine learning. Google is an example of a company that has already shared its federated learning podium in the form of TensorFlow Federated at the beginning of this year. With iOS 14, Apple is adding support for federated learning and allowing developers to use on-device intelligence to power their apps [6].

The app developer can create their federated learning algorithm or use TensorFlow Federated. Once you have created an algorithm, you can use the new Federated Learning API in iOS 14 to access the federated learning framework. Apple has also shared a few sample apps written in Swift and TensorFlow Federated. You can check out the sample apps on Apple's GitHub page.

Data privacy and security will be significant concerns in the future. As we continue to generate more and more data, businesses will need to ensure that this data is protected and used ethically**.**

DATA PARTITION

There are several different ways to partition data in federated learning, each of which has its own advantages and disadvantages. These include:

1. Stratified partitioning: In this approach, the data is divided into multiple groups based on certain characteristics or attributes, and each device is responsible for training on a different group of data. This ensures that each device is training on a representative sample of the overall data.
2. Random partitioning: In this approach, the data is randomly divided among the devices, with each device responsible for training on a different subset of the data. This approach can be useful for ensuring that each device is contributing to the training of the model in a balanced way.
3. Adaptive partitioning: In this approach, the data is partitioned in a dynamic way, with the assignment of data to devices based on their individual characteristics and capabilities. For example, a device with more computational power might be assigned a larger subset of the data to train on.
4. Federated transfer learning: In this approach, a pre-trained model is used to initialize the training process, and each device is responsible for fine-tuning the model on its local data. This can be useful for leveraging the knowledge learned by the pre-trained model while still allowing each device to tailor the model to its local data.

WAYS TO REDISTRIBUTE PARTITION

Rebalance is one way of redistributing partitions. The load is redistributed across all partitions by moving some partitions to other nodes. Reassign: The partitions are assigned to different nodes.

Repartition is another solution. Data in the partitions is redistributed across all partitions.

Single point of failure:

A centralized setup can be more susceptible to failure than a distributed setup because all components are located in one location. A few key things will need to happen for federated learning to truly take off: More organizations will need to adopt the technology and start using it for real-world applications. The technology will need to become more robust and scalable. Privacy and security concerns will need to be addressed. Federated learning will need to be integrated with other AI technologies. And lastly, the costs associated with federated learning will need to be reduced [10].

FUTURE OF FEDERATED LEARNING

Though much has been discussed concerning the downsides and advantages of federated learning, there are a number of critical open directions in federated learning that have not been explored. Here is a brief list of open problems;

Communication and privacy: Can we gain a deeper understanding of the relationships between communication and privacy in federated learning? For example, can we quantify the specific trade-offs between privacy and communication in federated learning?

Multi-agent systems: In federated learning, each device is a "multi-agent" that interacts with other agents in the system. Traditional distributed optimization typically assumes only two agents (i.e., a server and a client), whereas federated learning involves thousands or even millions of agents. Can we adapt results from multi-agent systems to this new setting, or develop new theories?

Federated reinforcement learning: While federated learning has been studied extensively for supervised learning tasks, its application to reinforcement learning tasks is still in its infancy. In particular, federated reinforcement learning has the potential to enable agents to jointly learn a cooperative policy by sharing experiences, which may lead to more efficient and robust learning. However, federated reinforcement learning presents several unique challenges, such as how to handle non-stationarily, how to implement distributed optimization algorithms, and how to manage the trade-off between exploration and exploitation. In addition, federated reinforcement learning can be used to study and understand how agents can learn to cooperate in a decentralized and distributed system.

Federated optimization with constrained clients: Many real-world applications of federated learning involve clients that are constrained in some way. For example, in federated reinforcement learning, it is often the case that agents can only communicate and share experiences with a limited number of other agents. In addition, federated optimization methods often need to be designed to handle clients that are constrained in terms of their data size or computing resources.

These and many other challenges need a collaborative effort from a broad range of research communities.

VI. Conclusion

With advances in technology, Federated learning is upgrading the solutions to improve privacy, and reliance on the data to train local models which collaborates to update a global model that enhances generalization behaviours though in a real sense no computer system is entirely safe from cyber-attacks. In instances when an attacker can compromise the privacy of a user by gaining access to their local data, research is being done to upgrade a novel FL method that will be combining both local and global models to protect the global model from cyber-attacks. Such a novel FL method is named as Distributed Federated Learning (DFL). It is developed on the basis of the two-tier training method of the original FL, i.e., data-level and model-level. The local model updates the global model by observing the local data, while the global model updates the local model by observing the global data. To protect the global model against cyber-attacks, scientists are proposing a novel mechanism named as security-aware-generalization, which updates the global model by observing the local data. The security-aware-generalization mechanism is based on security-aware data-level learning and security-aware model-level learning. Experiments have shown that the proposed DFBl-based method can provide a better generalization performance than the original FL-based method while protecting the privacy of the global model.

Lastly, machine learning is rapidly evolving and shaping the future of technology. Like other Machine learning technique, applications of federated learning has its benefits and challenges too. Federated learning can overcome these drawbacks and become game-changing aspect for many industries. Sooner, federated learning would make significant progress in its applications. Soon, many enterprises are going to embrace distributed learning model which in return will result to quick responses to dynamic consumer behaviour at reduced costs.

VII. References

[1] AbdulRahman, S., Tout, H., Ould-Slimane, H., Mourad, A., Talhi, C., & Guizani, M. (2020). A survey

on federated learning: The journey from centralized to distributed on-site learning. IEEE Internet of Things Journal, 8(7), 5476-5497.

[2] Augenstein, Sean, Andrew Hard, Lin Ning, Karan Singhal, Satyen Kale, Kurt Partridge, and Rajiv Mathews. "Mixed Federated Learning: Joint Decentralized and Centralized Learning." arXiv preprint arXiv:2205.13655 (2022).

[3] Elbir, Ahmet M., and Sinem Coleri. "A family of hybrid federated and centralized learning architectures in machine learning." arXiv preprint arXiv:2105.03288 (2021).

[4] Elbir, Ahmet M., Sinem Coleri, and Kumar Vijay Mishra. "Hybrid federated and centralized learning." In 2021 29th European Signal Processing Conference (EUSIPCO), pp. 1541-1545. IEEE, 2021.

[5] Kairouz, Peter, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, et al. "Advances and open problems in federated learning." Foundations and Trends® in Machine Learning 14, no. 1–2 (2021): 1-210.

[6] Konečný, Jakub, H. Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. "Federated learning: Strategies for improving communication efficiency." arXiv preprint arXiv:1610.05492 (2016).

[7] Li, Li, Yuxi Fan, Mike Tse, and Kuo-Yi Lin. "A review of applications in federated learning." Computers & Industrial Engineering 149 (2020): 106854.

[8] Li, Tian, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. "Federated learning: Challenges, methods, and future directions." IEEE Signal Processing Magazine 37, no. 3 (2020): 50-60.

[9] Liu, Jessica Chia, Jack Goetz, Srijan Sen, and Ambuj Tewari. "Learning from others without sacrificing privacy: Simulation comparing centralized and federated machine learning on mobile health data." JMIR mHealth and uHealth 9, no. 3 (2021): e23728.

[10] Mohri, Mehryar, Gary Sivek, and Ananda Theertha Suresh. "Agnostic federated learning." In International Conference on Machine Learning, pp. 4615-4625. PMLR, 2019.

[11] Moradi, Milad. "A centralized reinforcement learning method for multi-agent job scheduling in Grid." In 2016 6th International Conference on Computer and Knowledge Engineering (ICKE), pp. 171-176. IEEE, 2016.

[12] Navarro, Inaki, Ezequiel Di Mario, and Alcherio Martino. "Distributed vs. centralized particle swarm optimization for learning flocking behaviors." In ECAL 2015: the 13th European Conference on Artificial Life, pp. 302-309. MIT Press, 2015.

[13] Niknam, Solmaz, Harpreet S. Dhillon, and Jeffrey H. Reed. "Federated learning for wireless communications: Motivation, opportunities, and challenges." IEEE Communications Magazine 58, no. 6 (2020): 46-51.

[14] Rieke, Nicola, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R. Roth, Shadi Albarqouni, Spyridon Bakas et al. "The future of digital health with federated learning." NPJ digital medicine 3, no. 1 (2020): 1-7.

[15] Yang, Qiang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, and Han Yu. "Federated learning." Synthesis Lectures on Artificial Intelligence and Machine Learning 13, no. 3 (2019): 1-207.

[16] Zhang, Chen, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. "A survey on federated learning." Knowledge-Based Systems 216 (2021): 106775.