A Comparative Study of Federated LearningFrameworks to Flower Framework

Singanamalla. Jaya Mohnish

Department of Computer Science and EngineeringKoneru Lakshmaiah Education Foundation Vaddeswaram, A.P., India-522502. sjayamohnish@gmail.com

Kota.Venkata Narayana

Department of Computer Science and EngineeringKoneru Lakshmaiah Education Foundation Vaddeswaram, A.P., India-522502. kotavenkat1206@gmail.com

Thatavarti. Satish

Department of Computer Science and EngineeringKoneru Lakshmaiah Education Foundation Vaddeswaram, A.P., India-522502. drsatishthatavarti@kluniversity.in

Gandra. Shiva Krishna

Department of Computer Science and EngineeringKoneru Lakshmaiah Education Foundation Vaddeswaram, A.P., India-522502. shivagandra9664@gmail.com

Jonnalagadda. Surya Kiran

Department of Computer Science and EngineeringKoneru Lakshmaiah Education Foundation Vaddeswaram, A.P., India-522502. kiransurya93@kluniversity.in

Abstract—

Federated learning constitutes a decentralized ma- chine learning methodology, enabling the training of models on distributed data without necessitating centralized aggregation. The selection of an appropriate federated learning framework assumes paramount significance in achieving optimal model performance. This research endeavors to conduct a comparative assessment of various prominent federated learning frameworks using the Flower framework, a widely recognized benchmark dataset for evaluating federated learning algorithms. Our findings reveal that the Flower Framework exhibits superior performance with respect to flexibility and customization when juxtaposed with alternative frameworks. These outcomes suggest that the Flower Framework holds promise as a judicious choice for practitioners embarking on the deployment of federated learning within the context of the Flower framework. In summary, this investigation underscores the critical nature of the selection of a

federated learning framework in the practical application of this technique to real-world challenges.

Index Terms—Federated Learning, Flower Federated Frame- work, Machine Learning

I. INTRODUCTION

Federated learning presents a highly promising avenue within the realm of machine learning, facilitating the col- laborative training of machine learning models by multiple entities on their respective datasets, all without necessitating the exchange of data among the parties or with a central server. This innovative methodology holds the capacity to address a multitude of challenges intrinsic to conventional centralized machine learning paradigms, including but not limited to issues related to data privacy, data compartmentalization, and the demand for transferring substantial data volumes acrossnetworks that may be characterized by sluggishness or unre- liability.

There are several frameworks available for implementing federated learning, including the Flower Framework. However, little is known about how these frameworks compare in terms of their performance, ease of use, and compatibility with different types of data and machine learning models.

Within this research paper, our objective is to present a com-prehensive comparative analysis of federated learning frame- works, with particular emphasis on the Flower Framework. Our goal is to provide an in-depth analysis of the strengths and weaknesses of different federated learning frameworks, and to identify key factors that should be considered when choosing a framework for a particular application.

To execute our comparative study, we assessed various federated learning frameworks, considering multiple criteria. These criteria encompassed their capacity to facilitate the training of precise machine learning models, their proficiency in addressing data privacy and security concerns, their user- friendliness and compatibility with complementary tools and frameworks, as well as their scalability in the context of sub- stantial datasets and a multitude of participating entities. We also considered factors such as the availability of documenta- tion and support, the degree of flexibility and customization offered by the different frameworks, and their potential for future development and improvement.

Overall, our study aims to provide a comprehensive overview of the current state of the art in federated learning frameworks, and to help practitioners and researchers make informed decisions about which framework is best suited to their needs. By providing a detailed analysis of the strengths and weaknesses of different federated learning frameworks, we hope to contribute to the continued development and advancement of this critical area of machine learning.

II. LITERATURE SURVEY

A. *Title: Data privacy preservation algorithm with k- anonymity*

Authors: Waranya Mahanan, W.Art Chaovalitwongse, Jug- gapong Natwichai

K-anonymity is a privacy technique in which a dataset is modified so that no individual can be identified within a group of at least k individuals. The proposed algorithm uses k-anonymity to preserve the privacy of individuals in a dataset by replacing sensitive attribute values with generalized values. The generalized values are chosen such that they are representative of a group of at least k individuals, and the algorithm ensures that the number of individuals in each group is equal to or greater than k. To preserve privacy, the algorithm uses a clustering technique to group individuals based on their sensitive attribute values. It then replaces the sensitive attribute values with generalized values that are representative of the group. The algorithm also includes a check to ensure that no sensitive information is lost in the process. The authors assessed the algorithm’s performance across diverse datasets, demonstrating its effectiveness in safeguarding the privacy of individuals while concurrently enabling valuable data analysis. In summary, the algorithm presented herein furnishes a valuable means of preserving individuals’ privacy within a dataset, all the while permitting the conduct of data analysis. Its utility is particularly pronounced in scenarios mandating the sharing of sensitive data with external parties, as exemplified in domains such as medical and financial data sharing. [1]

B. *Title: A machine sound monitoring for predictive mainta- nence focusing on very low frequency band*

Authors: Kazuki Tsuji, Shota Imai, Ryota Takao, Tomonori Kimura, Hitoshi Kondo & Yukihiro kamiya

In this research paper, a machine sound monitoring system is introduced for the purpose of predictive maintenance, with a specific focus on the very low frequency (VLF) band. Predictive maintenance is a proactive approach to maintaining equipment that involves regularly monitoring equipment per- formance and identifying potential issues before they become serious problems. The proposed system uses sensors to moni- tor the VLF band of machine sounds, which is typically in the range of 0.1 to 10 Hz. The authors argue that monitoring this frequency range is particularly useful for predictive mainte- nance because many mechanical faults produce characteristic sounds in the VLF band. The system processes the collected sound data using machine learning techniques to identify and classify different types of mechanical faults. The authors evaluate the performance of the proposed system using data collected from a range of different machines, including pumps, fans, and motors. They show that the system can accuratelyidentify different types of mechanical faults, and that it can be used to detect problems before they cause equipment fail- ures. Overall, the proposed machine sound monitoring system provides a promising approach for predictive maintenance, as it is able to accurately identify potential issues before they become serious problems. It is particularly useful foridentifying mechanical faults in the VLF band, which are oftendifficult to detect using other methods. [2]

C. Title: Secure Aggregation for Federated Learning in Flower

Authors: Kwing Hei Li, Pedro Porto Buarque de Gusmão, Daniel J.Beutel, Nicholas D.Lane

In this research paper, a secure aggregation technique is introduced for the application of federated learning within the ”Flower” framework. Federated learning represents a machine learning paradigm enabling model training on decentralized data, thereby obviating the requirement for data centralization.This approach proves especially advantageous in scenarios where data privacy assumes paramount importance, affording individuals or organizations the ability to maintain sovereignty over their respective data while concurrently participating in the collective model training process. The proposed secure aggregation method is designed to ensure the privacy of the data used in federated learning by using secure multi-party computation (MPC) techniques. MPC allows multiple parties to jointly compute a function over their private inputs, withoutrevealing their inputs to each other. [16] The authors demon- strate the effectiveness of the proposed method by applying it to a range of different federated learning tasks and showing that it can achieve good performance while preserving the privacy of the data. Overall, the proposed secure aggregation method provides a useful approach for preserving the privacy of data in federated learning systems. It is particularly useful in situations where data privacy is a concern, and it allows individuals or organizations to retain control over their own data while still contributing to the training of a model. [3]

D. *Title: Privacy preservation techniques in big data analyt- ics: a survey*

Authors: P.Ram Mohan Rao, S.Murali Krishna, A.P.Siva Kumar

Big data analytics pertains to the intricate procedure of scrutinizing extensive and intricate datasets in order to extract invaluable insights and knowledge. However, the use of big data often raises privacy concerns, as it can potentially ex- pose sensitive information about individuals or organizations. The authors survey a range of different privacy preservation techniques that can be used to protect the privacy of individ- uals or organizations in big data analytics. These techniques include data anonymization, data perturbation, data masking, and data encryption, among others. The authors deliberate on both the strengths and limitations inherent in each technique, supplementing their discussions with practical applications as illustrative cases. In essence, this survey furnishes a compre- hensive panorama of the diverse array of privacy preservation techniques accessible for incorporation within the domain of big data analytics. It serves as a valuable resource for both researchers and practitioners seeking to deepen their understanding of safeguarding the privacy of individuals or organizations within the realm of big data analytics. [4]

E. *Title: Federated Machine Learning: Survey, Multi-Level Classification, Desirable Criteria and Future Directions in Communication and Networking Systems*

Authors: Omar Abdel Wahab, Azzam Mourad, Hadi Otrok, Tarik Taleb

The research paper offers a thorough and inclusive ex- ploration of federated machine learning (FML) within the context of communication and networking systems. FML is a machine learning methodology that empowers multiple entities to collaborate on model training, all without necessitating data centralization. This is particularly useful in situations where data privacy is a concern, as it allows individuals or organizations to retain control over their own data while still contributing to the training of a model. The authors survey a range of different FML approaches and classify them according to a multi-level classification framework. Addition- ally, the paper outlines a set of essential criteria that can be employed to assess the efficacy of FML methodologies, along with a thorough examination of the challenges and potential future avenues for FML within communication and networking systems. In summary, this survey delivers an all-encompassing examination of the diverse FML techniques that have been devised in the realm of communication and networking sys- tems. It serves as a valuable reference for researchers and practitioners keen on gaining a deeper understanding of the implementation of FML in these specific contexts.

The authors also provide insight into the challenges and future directions of FML, which can be useful for guiding future research inthis area. [5]

F. Title: Data Poisoning Attacks on Federated Machine Learn-ing

Authors:Gan Sun, Yang Cong, Jiahua Dong, Qiang Wang, Lingjuan Lyu, Ji Liu

The research paper discusses the issue of data poisoning attacks in context of federated machine learning (FML). To defend against these attacks, the authors propose several algorithms and tools that can be used. One approach that the au- thors propose is the use of a ”poisoning detection algorithm,” which is designed to identify and remove malicious data from the training process. The algorithm functions through the assessment of the model’s performance on distinct data sub- sets, pinpointing any subsets that exhibit a disproportionately influential effect on the model’s performance. Subsequently, these identified subsets are flagged as potentially malicious, and the data they encompass is excluded from the training process. An alternative approach put forth by the authors isthe adoption of ”adversarial training,” a method entailing the training of the model to enhance its resilience against data poisoning attacks. This is achieved by generating syntheticdata that is designed to mimic the effects of a data poisoning attack and using this data to train the model. The authors further delve into the application of supplementary tools and methodologies, including data sanitization, a process involving the preprocessing of data to eliminate any malevolent content, as well as ”secure aggregation,” a technique enabling multiple entities to collaboratively compute a function using their private inputs without disclosing said inputs to each other. Collectively, the paper provides a comprehensive overview of numerous algorithms and tools that can be harnessed to safeguard federated machine learning (FML) systems against data poisoning attacks. These approaches are designed to identify and remove malicious data, and to train the modelto be robust against these attacks. [6]

G. Title: SAFE: Secure Aggregation with Failover and En- cryption

Authors: Thomas Sandholm, Sayandev Mukherjee, Bernardo A.Huberman

The SAFE (Secure Aggregation with Failover and Encryp- tion) approach for safe aggregation is suggested in this study for use in federated machine learning (FML). The SAFE approach enables safe data aggregation in the context of FML by combining secure multi-party computation (MPC) with failover and encryption mechanisms. Through the employment of Multi-Party Computation (MPC), multiple participants can collectively compute a function using their respective private inputs, all while preserving the confidentiality of those inputs from each other. The SAFE technique leverages MPC to achieve secure data aggregation from various sources, ensuring that none of the parties involved has access to the underlying data. The aggregation process is made reliable and secure by using failover and encryption mechanisms. Although encryp- tion is used to protect the data during transmission, failover refers to the use of redundant systems to guarantee that the aggregation process can continue in the case of a failure. The authors demonstrate the effectiveness of the SAFE method by applying it to a range of different FML tasks and showing that it can achieve good performance while preserving the privacy of the data. Overall, the SAFE method provides a useful approach for securely aggregating data in the context of FML. It combines MPC with failover and encryption techniques to ensure the reliability and security of the aggregation process, and it has been shown to be effective in a range of different FML tasks. [7]

III. METHODOLOGY

Typically, in Machine Learning, we train on data that has been collected from a wide variety of endpoints, such as smartphones, laptops, etc., and then uploaded to a single server. Machine learning algorithms then use this information to self-train, before using this knowledge to make predictions about future data. Federated learning is a distributed machine learning technique that enables several users to cooperatively train a machine learning model without requiring them to ex- change raw data. A central server oversees the training process by providing model updates to each client and receiving modelupdates from each client.

A. *How does Federated Learning work*

Federated learning involves multiple people sharing their data remotely to train a single deep learning model collabo- ratively and frequently, as in a team report or presentation. The model, which is often a pre-trained foundation model, is downloaded by each party from a cloud-based data center. They use their private data to train the model, then condense and encrypt its updated configuration. Model updates are transmitted back to the cloud, where they are combined, averaged, and encrypted before being added to the central model. Iteration after iteration, the collaborative training is conducted until the model is fully trained.

B. *Difference between traditional and federated learning ap- proach*

Here the Traditional Model Approach refers to old Central- ized Model building Architecture where in a repository, many data sources for machine learning models are centralized. This central location may be a data warehouse, a data lake, or a novel hybrid of the two, a lake house. You may choose a single algorithm, such as the decision tree, or a set of algorithms, such as neural networks, to train on the collected data. The model may then be executed instantly on the central server or disseminated to several devices.

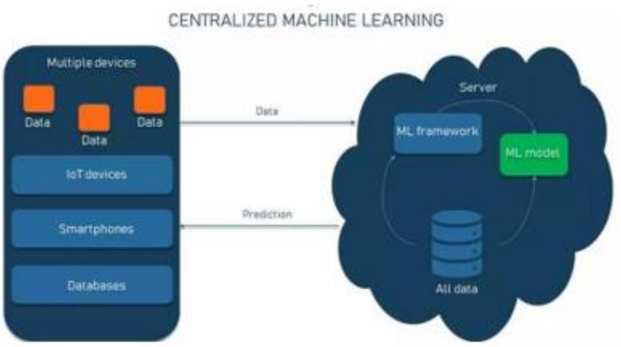


Fig. 1. Centralized Machine Learning Architecture

C. *De-centralized Machine Learning Architecture*

But where in the Federated Learning Approach, The Com- plete Architecture of Model building is De-Centralized to build a model in an effective way to fit the model across most of the devices it is required on. In this approach we have separate models on our Client and Server Machines for training, The initial training process of the model happens on the Client Machine and then the insights are sent to the Server model for an Aggregated training of the Server Model and then the modifications are sent to the Clients to train its model according to the results it gained from the Server Model.

Federate Learning helps us in De-Centralizing the whole process of model building and its Iterative Training for future developments to accommodate the requirements rather than replacing the model.

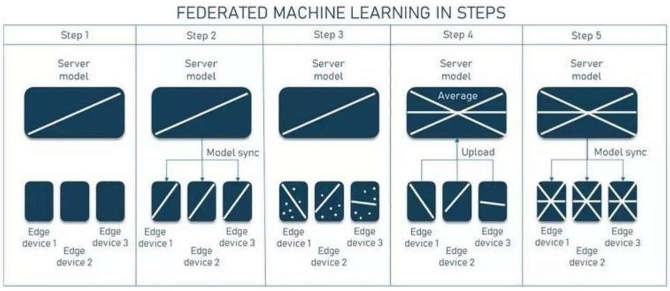


Fig. 2. De-centralized machine learning architecture

D. Various Frameworks for implementing Federated Learning

1) Flower: A comprehensive FL framework that distin- guishes itself from other platforms by offering extra capabili- ties for carrying out extensive FL experiments and examining a wide range of FL device scenarios. Flower provides several features to support this process, including:

- Data Sharding: Flower can divide the data among the clients in a way that ensures that each client receives a representative sample of the overall data set.
- Model aggregation: Flower provides algorithms for ag- gregating the updates from the clients in a way that ensures the resulting model is accurate and unbiased.
- Communication protocols: Flower includes support for various communication protocols, such as HTTP and gRPC, to enable communication between the clients and the server.

Overall, Flower is designed to make it easy to build and deploy federated learning systems, allowing developers to focus on building the machine learning models and algorithms rather than the infrastructure needed to support federated learning.

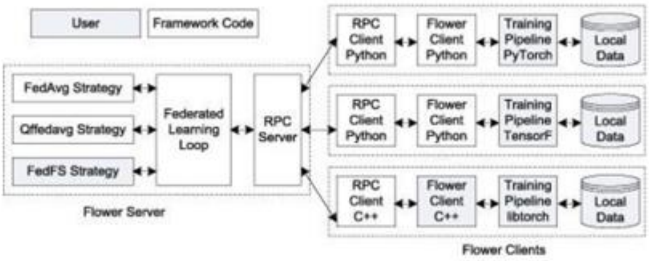


Fig. 3. Architecture of Flower Federated Learning

In federated learning, secure aggregation algorithms are used to combine the model updates from multiple participating devices or clients in a way that preserves the privacy of the data used to compute the updates. Federated Averaging (FedAvg) is a popular technique in federated learning and one example of a secure aggregation algorithm. In FedAvg, the model updates from each participating device are first locally averaged on the device, and then these averaged updates are securely aggregated on the server using a secure multi-party computation (MPC) protocol. This ensures that the server doesnot have access to the raw [8] model updates, or the data used to compute them, and as a result, the privacy of the data is preserved.

Other secure aggregation algorithms that have been pro- posed for use in federated learning include Secure Aggregation with Encrypted Gradients (SAEG) and Secure Aggregation with Randomized Response (SARR)[10]. These algorithms use different techniques to achieve privacy-preserving ag- gregation, such as encrypting the model updates or adding noise to the updates before sending them to the server. To guarantee that the information used to train a machine learningmodel stays private and is not shared with the server or other participating devices, federated learning uses privacy preservation methods. In federated learning, privacy can be maintained by a number of methods, such as:

- Several parties may collaboratively calculate a function on their private inputs using the secure multi-party com- puting (MPC) approach without disclosing their secret inputs to one another. In a federated learning environ- ment, MPC may be utilized to safely aggregate the modelupdates from participating devices [14].
- Differential privacy: This technique adds noise to the data or model updates in a controlled way to obscure the data and prevent it from being re- identifiable. In a federated learning environment, differential privacy can be utilized to safeguard the confidentiality of the data used to traina machine learning model.
- Homomorphic encryption [14]: This method makes it possible to do mathematical operations on encrypted data while maintaining the encryption of the operation’s output. In order to maintain data privacy, homomorphic encryption can be utilized to allow participating devices to use encrypted data for model training and evaluation.

2) LEAF: A federated learning framework called LEAF (Learning with Fewer Labels) attempts to enhance the per- formance of machine learning models in situations when each participating device or client has access to a limited quantityof labelled data.

In LEAF, the participating devices are organized into a tree structure, with a central server at the root and the devices as the leaves. The devices receive model updates from the central server, and they utilize these updates to train a local model on their own data. The servers then receive the updated models from the devices and compile the changes [9] using a secure aggregation algorithm such as Federated Averaging (FedAvg). One key feature of LEAF is that it allows the participating devices to communicate with each other directly, in addition to communicating with the central server. By enabling greater information sharing and mutual learning across the devices, this can aid in enhancing the model's performance. LEAFalso includes several mechanisms to improve the efficiency of federated learning, such as a technique called "model distillation"[13] that allows the participating devices to learn from each other's model updates without requiring them tocommunicate the raw updates.

3) *FED Scale:* FED Scale is an emerging machine learning setting[15] that aims to improve the efficiency and scalability of federated learning by using a decentralized approach to model training and aggregation. In FED Scale, the participat- ing devices are organized into a decentralized network, rather than a centralized tree structure as in some other federated learning frameworks. Every device oversees using its own datato train a local model, which it then transmits to a group of chosen devices known as "neighbors" for aggregation. Subsequently, the neighbors combine the model updates and transmit them back to the original device, which utilizes the combined updates to refine its local model.

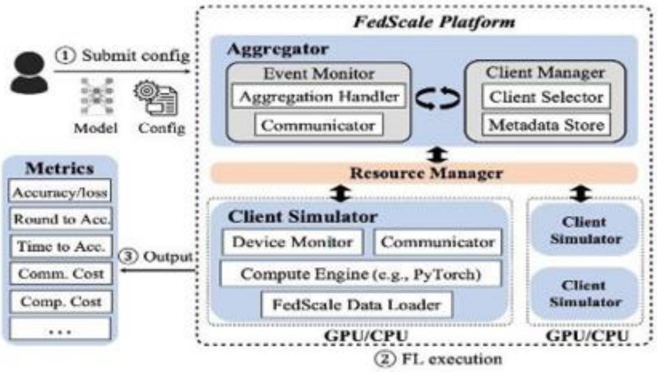
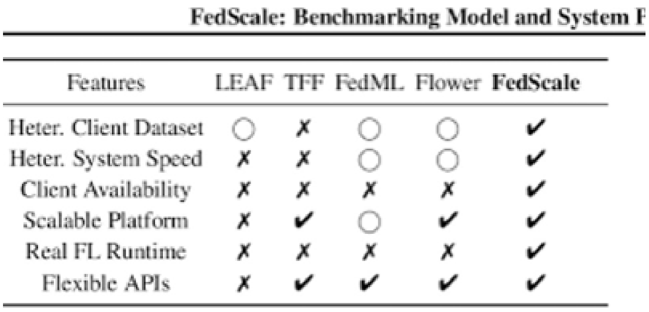


Fig. 4. Fed Scale Architecture

This decentralized approach allows FED Scale to scale to large numbers of devices and to adapt to changing network conditions, such as devices joining or leaving the network. It also allows FED Scale to use a variety of secure aggregation algorithms, such as Federated Averaging (FedAvg) and Secure Aggregation with Randomized Response (SARR), to preserve the privacy of the data used to compute the model updates.[17]



FedScale: Benchmarking Model and System F

Features	LEAF	TFF	FedML	Flower	**FedScale**
Heter. Client Dataset	○	✗	○	○	✔
Heter. System Speed	✗	✗	○	○	✔
Client Availability	✗	✗	✗	✗	✔
Scalable Platform	✗	✔	○	✔	✔
Real FL Runtime	✗	✗	✗	✗	✔
Flexible APIs	✗	✔	✔	✔	✔

Fig. 5. Comparison between various Frameworks with FedScale

4) *TensorFlow Federated:* TensorFlow Federated (TFF) is a framework for federated learning developed by Google and built on top of TensorFlow, a popular machine learning library. TFF provides a high-level API for implementing federated learning algorithms and running federated learning experiments. Federated learning in TFF is accomplished by a series of ”federated computations,” or functions that work with federated data and models. Federated data is a type of data that is spread among several participating clients or devices. In TFF, it is represented as a group of ”federated tensors.” Federated models—represented as ”federated variables” in TFF—are machine learning models that have been trained on federated data.

TFF includes several built-in federated computations for common tasks such as federated averaging (FedAvg), which is a popular method for aggregating model updates in federated learning. TFF also includes support for custom federated computations, allowing users to implement their own federatedlearning algorithms or variations on existing algorithms[12]. At a high level, the architecture of a federated learning system built using TFF comprises of the following components:

- Participating devices: These are the gadgets, like smart- phones or Internet of Things devices, that take part in the federated learning process. To accomplish model training and evaluation, each device has its own data and model and connects with the server and other devices.
- Server: The model updates from the participating devices are aggregated by the server,

ISSN PRINT 2319 1775 Online 2320 7876

Research paper © 2012 IJFANS. All Rights Reserved, UGC CARE Listed (Group -I) Journal Volume 8, Issue 3, 2019

which also manages the federated learning process. The server may also provide additional resources, such as model initialization or ad- ditional data, to the participating devices as needed.

- TensorFlow Federated API: The TFF API provides a set of functions and classes for implementing federated learning algorithms and running federated learning exper- iments. It includes built-in support for common tasks such as federated averaging (FedAvg) and model evaluation, aswell as support for custom federated computations.
- TensorFlow: TensorFlow is the underlying machine learn- ing library used by TFF. It provides a wide range of machine learning and numerical computing capabilities, including support for neural networks, gradient descent, and linear algebra.

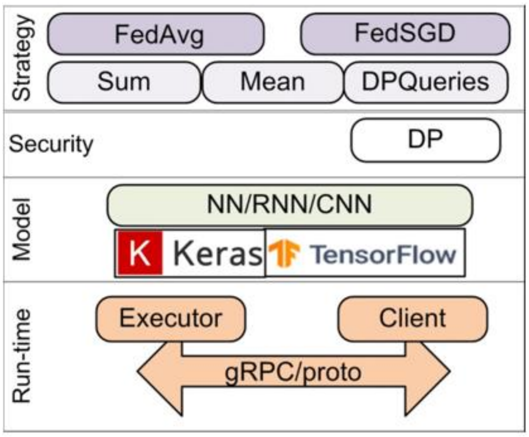


Fig. 6. TensorFlow Federated Architecture [18]

Fed AI: FedAI is a framework for federated learning thataims to provide a high-level API for implementing federated learning algorithms and running federated learning experi-ments.

In FedAI, federated learning is implemented as a sequence of ”federated computations,” which are functions that operate on federated data and models. Federated data is defined as information that is dispersed among several participating clients or devices and is represented in FedAI as a set of”federated tensors.” Federated variables, as they are called in FedAI, are machine learning models that have been trained using federated data. FedAI includes several built-in federatedcomputations for common tasks such as federated averaging (FedAvg), which is a popular method for aggregating model updates in federated learning. FedAI also includes support for custom federated computations, allowing users to implement their own federated learning algorithms or variations on exist- ing algorithms.

FedAI is built on top of PyTorch, a popular deep learning library, and is designed to be easy to use and integrate into existing PyTorch workflows.

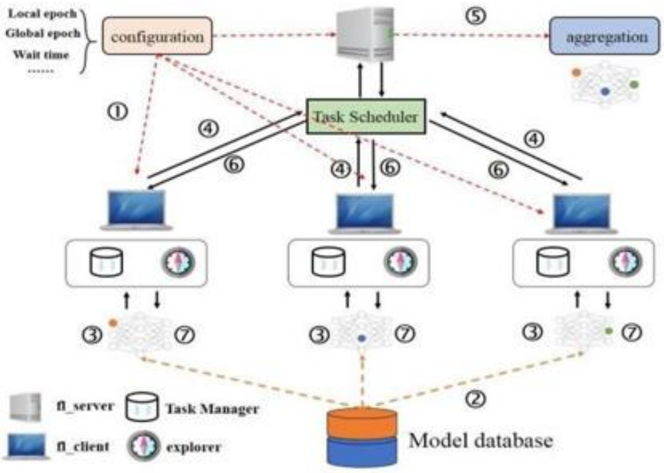


Fig. 7. FedAI Architecture

- Configuration: Users can set up training parameters with this function, which includes the number of iterations, reconnections, server URL for uploading model parame- ters, and other important components.
- Task Scheduler: It evens out the use of local computer resources during federated model training by managing communications between the federated learning server and clients. To maximize the quality of the final federated model, the load balancing technique takes into account the quality of the clients' local models as well as the current demand on their local computing resources.
- Task Manager: When numerous clients teach various model training algorithms simultaneously, this sectionmanages concurrent federated model training activities.
- Explorer: To affect the Task Scheduler's load-balancing choices, this component tracks client-side resource use (such as CPU utilization, memory consumption, network demand, etc.).
- FL SERVER: This server makes it possible for federated learning. It covers crucial federated learning components including uploading model parameter sets, model aggre- gation, and model dispatching[11].
- FL CLIENT: It executes the essential federated learning phase of local model training and houses the Task Man- ager and Explorer components.

IV. CONCLUSION AND FUTURE WORK

By emphasizing a knowledge of federated learning frame- works and their designs, this research aims to demonstrate why the flower federated framework is more practical than several other frameworks. We initially described the architectures of all the frameworks and then contrasted them with flower to see how flow is adaptable and straightforward to meet our aims in federated learning. We hypothetically evaluated all the designs and their suitability for usage in various situations in order to swiftly discover the best design available. Flower is clearly more adaptable and practical than any other framework, allowing us to swiftly build a federated learning architecture.

REFERENCES

[1] Mahanan, W., Chaovalitwongse, W. A., & Natwichai, J. (2001). Data privacy preservation algorithm with K-anonymity. World Wide Web, 24(5), 1551-1561. https://doi.org/10.1007/s11280-021-00922-2

[2] Tsuji, K., Imai, S., Takao, R., Kimura, T., Kondo, H., & Kamiya, Y. (2001). A machine sound monitoring for predictive maintenance focusing on very low frequency band. SICE Jour- nal of Control, Measurement, and System Integration, 14(1), 27-38. https://doi.org/10.1080/18824889.2000.1863611

[3] Li, K. H., De Gusmão, P. P., Beutel, D. J., & Lane, N. D. (2001). Secure aggregation for federated learning in flower. Proceedings of the 2nd ACM International Workshop on Distributed Machine Learning. https://doi.org/10.1145/3488659.3493776

[4] Ram Mohan Rao, P., Murali Krishna, S., & Siva Kumar, A. P. (2018). Privacy preservation techniques in big data analytics: A survey. Journal of Big Data, 5(1). https://doi.org/10.1186/s40537-018-0141-8

[5] Wahab, O. A., Mourad, A., Otrok, H., & Taleb, T. (2001). Fed- erated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking sys- tems. IEEE Communications Surveys & Tutorials, 23(2), 1342-1397. https://doi.org/10.1109/comst.2001.3058573

[6] Sun, G., Cong, Y., Dong, J., Wang, Q., Lyu, L., & Liu, J. (2002). Data poisoning attacks on federated machine learning. IEEE Internet of Things Journal, 9(13), 11365-11375. https://doi.org/10.1109/jiot.2001.3128646

[7] Sandholm, T., Mukherjee, S., & Huberman, B. A. (2000). Demo — SPoKE. Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. https://doi.org/10.1145/3548606.3563701

[8] Wang, Z., Kang, Q., Zhang, X., & Hu, Q. (2002). Defense strategies toward model

poisoning attacks in federated learning: A survey. 2022 IEEE Wireless Communications and Networking Conference (WCNC). https://doi.org/10.1109/wcnc51071.2002.9771619

[9] Costa, G., Pinelli, F., Soderi, S., & Tolomei, G. (2002). Turning federated learning systems into covert channels. IEEE Access, 10, 130642-130656. https://doi.org/10.1109/access.2002.3229124

[10] Lim, W. Y., Ng, J. S., Xiong, Z., Niyato, D., & Miao, C. (2002). Feder- ated learning at mobile edge networks: A tutorial. Wireless Networks, 1-51. https://doi.org/10.1007/978-3-031-07838-5 1

[11] Sun, Z., Kairouz, P., Suresh, A. T., & McMahan, H.B. (2019). Can you really backdoor federated learning? arXiv preprint arXiv:1911.07963.

[12] Li, D., & Wang, J. (2019). Fedmd: Heterogenous federated learning via model distillation. arXiv preprint arXiv:1910.03581.

[13] Truex, S., Baracaldo, N., Anwar, A., Steinke, T., Ludwig, H., Zhang, R., & Zhou, Y. (2019). A hybrid approach to privacy-preserving fed- erated learning. Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security. https://doi.org/10.1145/3338501.3357370

[14] Ma, J., Naas, S., Sigg, S., & Lyu, X. (2002). Privacy-preserving federated learning based on multi-key homomorphic encryption. International Journal of Intelligent Systems, 37(9), 5880-5901. https://doi.org/10.1002/int.22818

[15] Lai, F., Dai, Y., Zhu, X., Madhyastha, H. V., & Chowdhury,

M. (2001). FedScale. Proceedings of the First Workshop on Systems Challenges in Reliable and Secure Federated Learning.https://doi.org/10.1145/3477114.3488760

[16] Bellini, A., Bellini, E., Bertini, M., Almhaithawi, D., & Cuomo, S. (2003). Multi-party computation for privacy and security in machine learning: A practical review. 2023 IEEE International Conference on Cyber Security and Resilience (CSR). https://doi.org/10.1109/csr57506.2003.10224826

[17] Guendouzi, B. S., Ouchani, S., EL Assaad, H., & EL Zaher, M. (2003). A systematic review of federated learning: Challenges, aggregation methods, and development tools. Journal of Network and Computer Applications, 220, 103714. https://doi.org/10.1016/j.jnca.2003.103714