Heart Stroke Prediction Using Federated Learning

Dhanunjay Potti[1*], Mandavalli N V Saisandeep[2], V Madhu Viswanatham[3], Prasanth Ganapavarapu[4]

[1,2,3]*School Of Computer Science and Engineering; Vellore institute of Technology (VIT), Vellore, India; E-mail: dhanupotti2001@gmail.com*
[4]*R.V.R & J.C College of Engineering*

Abstracts: Heart attacks in youth have become a common issue, making it crucial to predict the chances of heart stroke in patients. However, the fragmented and private nature of healthcare data presents a significant challenge for producing reliable research results across populations. Federated Learning shows promise in connecting disparate healthcare data sources while also safeguarding privacy. To that end, this research aims to develop a Federated Learning algorithm that addresses the challenges and opportunities in healthcare by utilizing a central server to train a sharing globalized model while retaining sensitive data in local institutions where they originated. This approach is particularly important as electronic health records of diverse patient groups are owned by multiple institutions, making it difficult for hospitals to share such sensitive information. This proposed algorithm will contribute towards the creation of generalizable and efficient analytical techniques for predicting heart stroke risk in patients. This proposed algorithm produced better accuracies compared to traditional machine learning approaches. The study focused on working with numerical data sets and has the potential to contribute towards the creation of generalizable and efficient analytical techniques for predicting heart stroke risk in patients.

Keywords: Federated Learning, Neural Networks, Weights, Communication Overhead, Privacy and Security.

1. INTRODUCTION

Heart attacks, also known as myocardial infarctions. These usually come on by an abrupt disruption of the blood supply to the heart. The major blood channels that feed the heart with blood, the coronary arteries, become clogged with cholesterol deposits in the condition of CHD. The plaques formations which rupture before a heart attack, lead to the development of a blood clot at the site of the rupture. Heart attacks are typically associated with older adults, but they are becoming too common in young people these days. People with regular workouts are also on the verge to get a stroke. There has been a rise in the occurrence of heart attacks among youngsters primarily attributed to various factors including smoking, hypertension, obesity, diabetes, hereditary to heart disease. Education on heart disease and its risk factors is crucial to prevent future heart attacks in young people. Heart attacks in youth can have long-lasting effects on their physical and emotional well-being, including anxiety and depression. One might not prevent a heart attack but can always predict on who is going to affected by it.

The way of interaction with technology is disrupted by Artificial Intelligence (AI). It is transforming mode of communication, prediction and decision making. While the recent advancements in AI have been notable, for quite some time AI has been used in the healthcare industry. In initial days, AI was employed for tasks such as medical diagnosis and decision support. The healthcare industry is witnessing a remarkable expansion in the role of Artificial Intelligence (AI), which has the potential to revolutionize healthcare delivery and significantly enhance patient outcomes. AI algorithms are increasingly being utilized for predicting and preventing heart strokes. Through analysis of enormous medical data, these algorithms can detect patterns and identify risk factors that can indicate probability of heart stroke. Moreover, they help healthcare providers devise individualized treatment plans for patients based on their unique data. To ensure maximum accuracy and obtain optimal results, it is imperative to consider extensive amounts of data and edge cases while working on a large-scale problem like this.

Traditional machine learning typically requires data training, where the data is collected, and the total learning process takes place on a central server. However, Federated Learning offers a different approach. It is a machine learning environment that aims To construct an effective centralized model employing dispersed information for training across many clients, everyone with a usually poor and unpredictable connection to the network. This approach emphasizes data privacy and security. In simpler terms, Federated Learning is a machine learning training method that enables models to be trained without sharing the raw data that is present across multiple devices or locations.

The model is trained independently on each device or location, and the updated model weights are shared and combined to improve the overall model's performance. This approach facilitates the training of models on large amounts of diverse data without compromising data privacy or security.

This paper makes a comparison between the accuracies of machine learning models and federated learning follow. In contemplation to identify the covid-19 infections, a unique dynamic fusion-based federated learning technique was used by Zhang, W., Zhou, T., Lu, Q., Wang, X., Zhu in “Dynamic-Fusion-Based Federated Learning for COVID-19 Detection for Medical diagnostic picture analysis”. To evaluate medical diagnostic pictures, dynamic fusion-based federated learning systems are initially designed. Finally, they outlined the categories of diagnostic image datasets for covid 19 identification. Moreover, the assessment findings demonstrate that the suggested technique is workable and outperforms traditional federated learning [1].

On the other hand, the authors of FedZip: A compression framework for communication-efficient federated learning, used a new architecture called FedZip is used to transport deep learning model weights between clients and servers while reducing the number of updates. As we utilized Fedzip, the continuous transmission between clients and servers increased communication costs and was ineffective owing to the huge number of parameters. Fedzip uses three separate encoding methods to implement compression and achieves top-z sparsification using quantization with clustering. [10].

In July 2020, Briggs, Fan, and Andras published a paper titled "Federated Learning with hierarchical clustering of local updates to enhance training on non-IID Data". Their paper proposes an enhancement to the Federated Learning (FL) algorithm by adding a Hierarchical Clustering step. The intention of this phase aims to group clients into unique groups based on how similar their regional updates are in relation to the global joint model. Once the groups have been separated they begin working independently and concurrently on specialized models, demonstrating how federated learning plus hierarchical clustering enables the training of models to finish in a smaller amount of communication stages than federated learning despite clustering along with federated learning in addition to hierarchical clustering [5]

2. MATERIALS AND METHODS

2.1. The Dataset

The data source was obtained from Kaggle repository and serves as a cardiac attack dataset: https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset .

There are 11 features and 1 binary target variable. 1 indicates heart stroke, and 0 indicates no heart stroke.

The below text explains the dataset:

- Gender - sex of the patients
- Age - numerical representation of age
- Hypertension - Suffering from hypertension? 1 is a “Yes”, 0 indicates “No”
- Heart disease - history of heart diseases? 1- Yes, 0 - No
- Ever married - Married? 1- Yes, 0 - No

- Work type - Private, Independent, Government_employee
- residence – Urban or rural
- Average glucose level - The level of glucose in the body
- Smoking status - never smoked, previously smoked, smokes, not known.
- Stroke - 1 Heart stroke, 0 - no heart stroke

In the process of formulating a procedure, we formulate a prediction of heart stroke using Machine Learning models and Federated Learning algorithms and then compare the accuracies to show which is the best. The machine learning algorithms which are employed here include the random forest, Decision tree along with K-Nearest Neighbors.

2.2. Data Pre-processing

Firstly, label encoding is done, the dataset has a column named "gender" which has two attributes male and female. Male is denoted by 1, female is denoted by 0. In the column "ever-married", married person is denoted with 1 and bachelor is denoted 0. This way all the descriptive columns are converted to numeric. The missing vales are being checked. Every column has one or many missing values. All the missing values are replaced with the mean of column values.

In columns like "avg_glucose_level" and "BMI", the data is widely spread. For proper computation, the data must be normalized. Normalization is carried out to scale and transform features in a dataset to a standard range.

For every iteration, about 80% of the data collection is utilized for the purposed of the training set, while around 20% is used for the testing purpose.

Random forest is a suitable option for both classification and regression tasks, which is why it is being utilized. It is an ensemble method that builds multiple decision trees and combines their predictions to produce a final output. This can handle numerical data by splitting the data into ranges or bins and treating them as categorical variables. This helps to reduce the sensitivity of the algorithm to outliers and improves its overall performance. To generate a forecast the algorithm locates the K data points which are closest to a particular given query point and computes out the majority decision or average of those data points found there. KNN is capable of managing numerical data by utilizing distance metrics to determine the resemblance between different data points.

The Decision Tree is a form of supervised learning technique, which is suitable for both regression and classification tasks, though it is more often used for classification purposes. It acts as a classifier represented in the form of a tree. Here the data attributes are presented at the internal nodes, the rules are displayed through branches, and the outcomes are shown at the ending of every leaf. Due to its tree-like structure, this method is easy to understand and interpret, which makes it useful for various applications.

Required packages are installed before running the metrics where the classifiers are used to output the accuracies.

The goal of the algorithm is to clearly find out the attribute values on every individual device without jeopardizing its security. To N all of the clients, $C \subset \{C_0, C_1, ..., C_{tot}\}$, the server model ($w^0$) is broadcast. Three distinct learning rates are transmitted in addition to the distributed server models. The array (η_m), whose values vary from [1e-1, 1e-5], is used to choose the learning rates. Additionally, the sample size is determined at random, each edge device is provided with an identical value of w^0 therefore boosting its sample size may additionally improve out the training accuracy, which is copied for all values and separately monitored for an entire iteration.

The complexity, size, ambiguity, and variation of several data characteristics are exclusive to edge devices. The hyper-parameters are carefully chosen because these data properties will influence training. Only the model with the lowest loss w^0_{min}, is chosen out of the models at every individual device. Each edge device will provide the values *w^0_{min}, η_{min}, and $losses_{min}$*. Because they can display data about specific edge devices, these statistics are crucial. On the server, the models "w^0" learning rates "0," "1",... "n" and their associated losses are obtained. Using the model aggregation technique, the edge device models are combined to create the server model. The model weights (w^0_n) are incrementally added.

$$f(w) = \sum_{k=1}^{K} \frac{nk}{n} F_k(w) , \qquad \text{where } F_k = \frac{1}{nk} \sum_{i \in Pk} f_i(w)$$

2.3. Procedure

Algorithm 1 Clustering and Initial Broadcasting

```
Create a server
For every client
while j = 3 do
    while j ≠ clients do
            k=0
            while j ≠ 3 do
                $w_j^0$ on $\eta_j$
                los = loss($w_j^0$)
return train_losses, val_losses
$w^0 \leftarrow \frac{\Sigma(w_n^0)}{n}$
```

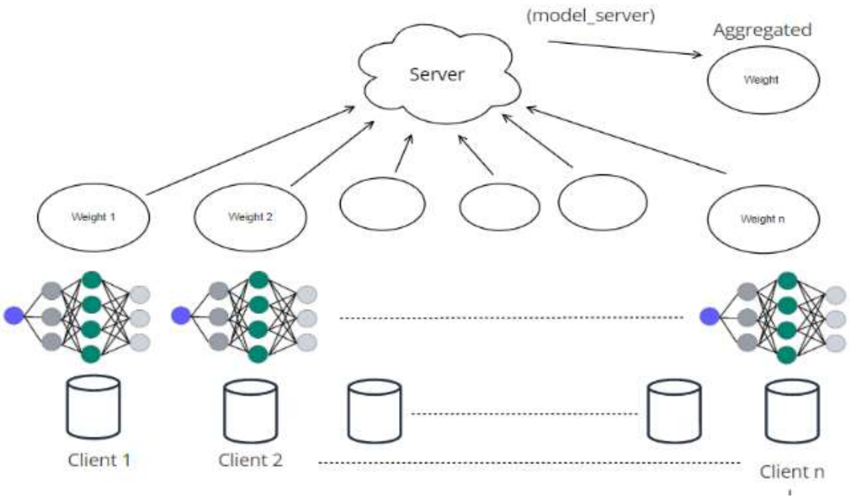
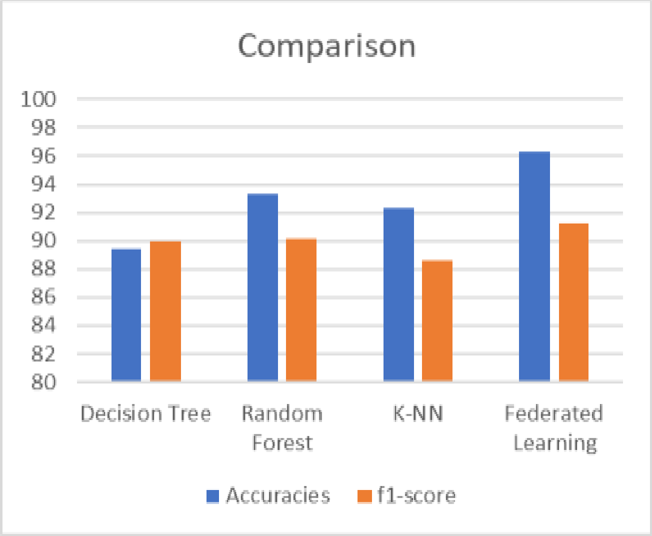


Figure 1: Architecture of Federated Learning model

3. RESULTS AND DISCUSSION

The results were shown in Table 1. A comparison was made between the accuracies of traditional machine learning models and Federated Learning. The models used under traditional machine learning included Random Forest, Decision Tree, and K-Nearest Neighbors. The Decision Tree model showed an accuracy of 89%, while K-NN and Random Forest models showed 93% accuracy. On the other hand, the Federated Learning algorithm showed an accuracy of about 94%, which is higher than that of the traditional machine learning models. Multi-layer perceptron was used to implement the Federated Learning algorithm. Not only did it provide higher accuracy, but it was also found to be more secure compared to other algorithms.

Models	Accuracy	F1 score
Decision Tree	89.4%	90.0%
Random Forest	93.3%	90.18%
K-Nearest Neighbor	92.3%	88.59%
Federated Learning	96.3%	91.2%



CONCLUSIONS AND FUTURE WORK

To summarize, this research paper demonstrated a successful implementation of a federated learning approach for training a neural network to solve a binary classification problem. Our study compared the performance of this approach with a traditional centralized learning method and found that federated learning can achieve similar accuracy and loss while mitigating concerns over data privacy and security. Going forward, we aim to improve the accuracy of our approach further and extend it to various real-world applications. We also plan to explore the possibility of integrating our algorithm with nature-inspired algorithms to enhance its performance.

REFERENCES

[1] Zhang, W., Zhou, T., Lu, Q., Wang, X., Zhu, C., Sun, H., Wang, Z., Lo, S. K., & Wang, F. Y. (2021). Dynamic-Fusion-Based Federated Learning for COVID-19 Detection. IEEE Internet of things journal, 8(21), 15884–15891. https://doi.org/10.1109/JIOT.2021.3056185

[2] Kumar, R., Khan, A. A., Kumar, J., Golilarz, N. A., Zhang, S., Ting, Y., ... & Wang, W. (2021). Blockchain-federated-learning and deep learning models for covid-19 detection using ct imaging. IEEE Sensors Journal, 21(14), 16301-16314.

[3] Nilsson, A., Smith, S., Ulm, G., Gustavsson, E., & Jirstrand, M. (2018, December). A performance evaluation of federated learning algorithms. In Proceedings of the second workshop on distributed infrastructures for deep learning (pp. 1-8).

[4] Sattler, F., Müller, K. R., Wiegand, T., & Samek, W. (2020, May). On the byzantine robustness of clustered federated learning. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 8861-8865). IEEE.

[5] Briggs, C., Fan, Z., & Andras, P. (2020, July). Federated learning with hierarchical clustering of local updates to improve training on non-IID

data. In 2020 International Joint Conference on Neural Networks (IJCNN) (pp. 1-9). IEEE.

[6] Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., & Khazaeni, Y. (2020). Federated learning with matched averaging. arXiv preprint arXiv:2002.06440.

[7] Ghosh, A., Chung, J., Yin, D., & Ramchandran, K. (2020). An efficient framework for clustered federated learning. Advances in Neural Information Processing Systems, 33, 19586-19597.

[8] Agrawal, S., Sarkar, S., Alazab, M., Maddikunta, P. K. R., Gadekallu, T. R., & Pham, Q. V. (2021). Genetic CFL: hyperparameter optimization in clustered federated learning. Computational Intelligence and Neuroscience, 2021.

[9] Khan, T. I., Jam, F. A., Akbar, A., Khan, M. B., & Hijazi, S. T. (2011). Job involvement as predictor of employee commitment: Evidence from Pakistan. International Journal of Business and Management, 6(4), 252.

[10] Qayyum, A., Ahmad, K., Ahsan, M. A., Al-Fuqaha, A., & Qadir, J. (2022). Collaborative federated learning for healthcare: Multi-modal covid-19 diagnosis at the edge. IEEE Open Journal of the Computer Society, 3, 172-184.

[11] Malekijoo, A., Fadaeieslam, M. J., Malekijou, H., Homayounfar, M., Alizadeh-Shabdiz, F., & Rawassizadeh, R. (2021). Fedzip: A compression framework for communication-efficient federated learning. arXiv preprint arXiv:2102.01593.

[12] Ibraimi, L., Selimi, M., & Freitag, F. (2021, December). BePOCH: Improving federated learning performance in resource-constrained computing devices. In 2021 IEEE Global Communications Conference (GLOBECOM) (pp. 1-6). IEEE.[16]

[13] Arikumar, K. S., Prathiba, S. B., Alazab, M., Gadekallu, T. R., Pandya, S., Khan, J. M., & Moorthy, R. S. (2022). FL-PMI: federated learning-based person movement identification through wearable devices in smart healthcare systems. Sensors, 22(4), 1377.

[14] Jabbar, W. A. A.-., Mekkey, S. M. ., Sahib, A. S. ., & Oudah, G. A. A.-. (2023). Hesperetin Alleviate Renal Ischemia-Reperfusion Injury in Male Mice Model by Suppressing Inflammation & Oxidative Stress Pathway . International Journal of Membrane Science and Technology, 10(3), 410-420. https://doi.org/10.15379/ijmst.v10i3.1549

[15] Long, G., Xie, M., Shen, T., Zhou, T., Wang, X., & Jiang, J. (2023). Multi-center federated learning: clients clustering for better personalization. World Wide Web, 26(1), 481-500.

[16] Kopparapu, K., Lin, E., & Zhao, J. (2020). Fedcd: Improving performance in non-iid federated learning. arXiv preprint arXiv:2006.09637.

[17] Gadekallu, T. R., Pham, Q. V., Huynh-The, T., Bhattacharya, S., Maddikunta, P. K. R., & Liyanage, M. (2021). Federated learning for big data: A survey on opportunities, applications, and future directions. arXiv preprint arXiv:2110.04160.

DOI: https://doi.org/10.15379/ijmst.v10i3.1800