TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

THESIS NO: 076MSICE011

INTERNET OF VEHICLE (IOV) BASED DRIVER EMOTION DETECTION USING FEDERATED LEARNING

by
Pankaj Nidhi Regmi

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
LALITPUR, NEPAL

April, 2023

INTERNET OF VEHICLE(IOV) BASED DRIVER EMOTION DETECTION USING FEDERATED LEARNING

by
Pankaj Nidhi Regmi
076MSICE011

Thesis Supervisor

Sharad Kumar Ghimire, Associate Professor

Dr. Babu R. Dawadi, Assistant Professor

A final Thesis Report submitted in partial fulfillment of the requirements for the degree of Master of Science in Information and Communication Engineering

Department of Electronics and Computer Engineering
Institute of Engineering, Pulchowk Campus
Tribhuvan University
Lalitpur, Nepal

April, 2023

COPYRIGHT ©

DECLARATION

I hereby declare that the report of the thesis entitled **"INTERNET OF VEHICLE(IOV) BASED DRIVER EMOTION DETECTION USING FEDERATED LEARNING"** which is being submitted to the **Department of Electronics and Computer Engineering, IOE, Pulchowk Campus**, in the partial fulfillment of the requirements for the award of the Degree of MSc in **Information and Communication Engineering**, is a bonafide report of the work carried out by me. The materials contained in this report have not been submitted to any University or Institution for the award of any degree and I am the only author of this complete work and no sources other than the listed here have been used in this work.

____

Pankaj Nidhi Regmi

076MSICE011

Date: April, 2023

CERTIFICATE OF APPROVAL

The undersigned certify that he has read and recommended to the **Department of Electronics and Computer Engineering, IOE, Pulchowk Campus**, a thesis work entitled "**INTERNET OF VEHICLE (IOV) BASED DRIVER EMOTION DETECTION USING FEDERATED LEARNING**" submitted by **Pankaj Nidhi Regmi** in partial fulfillment for the award of the degree of MSc in Information and Communication Engineering. The thesis work was carried out under special supervision and within the time frame prescribed by the syllabus.

We found the student to be hardworking, skilled and ready to undertake any related work to his field of study and hence we recommend the award of Masters of Science Degree in Information and Communication Engineering.

Thesis Supervisor
Sharad Kumar Ghimire, Associate Professor
Department of Electronics and Computer Engineering, IOE, Pulchowk Campu

Thesis Supervisor
Dr. Babu R. Dawadi, Assistant Professor
Department of Electronics and Computer Engineering, IOE, Pulchowk Campus

External Examiner
Dr. Suresh Pokhrel, CEO
Facet Technology

Committee Chairperson and
MSICE Progarm Co-ordinator
Dr. Baburam Dawadi
Department of Electronics and Computer Engineering, IOE, Pulchowk Campus

Date of Approval: April, 2023

DEPARTMENTAL ACCEPTANCE

The thesis entitled "**INTERNET OF VEHICLE (IOV) BASED DRIVER EMOTION DETECTION USING FEDERATED LEARNING**", submitted by **Pankaj Nidhi Regmi** in partial fulfillment of the requirement for the award of the degree of "Master of Science in Information and Communication Engineering" has been accepted as a bonafide record of work independently carried out by him in department.

.

Associate Prof. Dr. Jyoti Tandukar

Head of Department

Department of Electronics and Computer Engineering

IOE, Pulchowk Campus

ACKNOWLEDGEMENT

I would like to give acknowledge and warmest sincere thanks to Associate Professor Dr. Jyoti Tandukar, HOD sir for letting me to do the thesis. I would like to express my thankfulness to supervisors Mr. Sharad Kumar Ghimire sir and Dr. Babu Ram Dawadi sir guiding me and suggesting me for my topic. I am thankful to all the teacher and faculties' member of Department of Electronics and Computer Engineering.

Lastly, I would also like to thank my family who continuously motivated me to complete thesis work. Also, thank to friends, past and previous office staffs who directly or indirectly supported me to do the work.

Sincerely,

Pankaj Nidhi Regmi

076 MSICE 011

ABSTRACT

With the advancement of edge smart computing devices and Internet of Vehicle (IoV) technologies emotion detection has become one of the most used methods in smart vehicle while driving. Many models have been employed however, privacy disclosure and communication cost are still a question. To address this question a federated learning driver emotion detection system model is proposed. It intelligently utilizes collaboration between edge, client and cloud for realizing dynamic model also protecting edge data privacy.

Federated Learning has an advantage on privacy. In this thesis two different algorithm FedAvg and FedSGD are compared. It is found that accuracy of FedAvg is better than FedSGD. Also, FedSGD takes more steps to converge than FedAvg.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
DCCN	Deep Convolution Neural Network
FL	Federated Learning
IID	Independent and Identically Distributed
IOT	Internet of Things
ML	Machine Learning
NN	Neural Network
UE	User Equipment
V2I	Vehicle-To-Infrastructure
V2V	Vehicle-To-Vehicle

1. OVERVIEW

1.1 Introduction

Internet of Things (IoT) has revolutionized the world and the way we live. With the advancement of edge smart computing devices and IoT technologies we are able to control our smart device from any part of the world. Internet of Vehicle (IoV) is a subset of IoT which rather focus on vehicle to on vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communication transferring the way we interact with cars, and enabling the new features that were previously impossible.

Detection of driver emotion is a driver safety technology which prevents vehicle accidents and save drivers by detecting if they are emotionally fit. From various studies it has been found that most road accident are fatigue and emotion related whereas accident may occur due to road condition and driver negligence. Driver's fatigue has been taken as a significant factor in high portion of road accident. The technological enhancement on detection and prevention of driver emotion has become a major topic of interest to control road accident. System needs to be advanced for responding to the effect of driver emotion. Inattention of driver can be taken while driving due to drowsiness and distraction of driver due to emotion. Drivers get distracted when they feel drowsy, and their attention goes away from driving the car which may cause a serious accident.

The classical ML model (also called isolated learning) is used to learn from the stored data. But in many applications, we need the data that is generated over time and needs to be dealt with accordingly. IoV based driver emotion detection involves in collecting data from the different vehicles which have different sensors, driver characteristics, and driving condition. Federated learning helps this diverse dataset used in training model without need of sensitive data to the centralize server. Thus it helps to preserve the data privacy. The training process is performed locally with multiple clients, where only sensitive data is transferred to the server. This allow the training process to be complete more quickly with the less computational resource and reduced communication cost. Also, client can learn independently and they also have self-judgment capacity. By aggregating the training results that is computed on different vehicles, the model can learn wide variety of data thus is more robust to different driving condition. Overall, the motivation in IoV based driver emotion detection using FL is to prevent accidents and save lives. FL helps to preserve the data privacy, reduce to computational burden for the server and communication cost associated while transferring the data.

Machine learning (ML) is a sub-set of Artificial Intelligence (AI), which emphasizes on the development of algorithms and predicts the model based on data set[1]. It prioritizes on objectives and applications, mostly optimization and prediction. Whenever we talk about FL, the reference is ML. In the past decades, ML has transformed the processing of the data for a large-scale application. There is high consumption of data and is increasing rapidly. Classical ML also called centralized ML learns the model from the stored data. Thus, this type of model does not store information about past. In particular, it is a centralized data training algorithm. The data is gathered, and the overall process is performed in the central server. Unlike classical machine learning, FL is a machine learning technique which trains an algorithm across many decentralized devices which holds local data samples and also maintain the privacy. The key idea in FL is the decentralized machine learning framework. It contains central server and the client. Server transmit initial model to the several nodes. At the local client the model is trained with the local data. The central server pools the model result from different client and generate its global model. It uses averaging of the model at the server and that update will be used in the client for the next round of training.

Detection of driver emotion works based on the acquirement of video captured by camera that is placed in front of the driver. Incoming video stream is processed to alert the driver's emotion level if drowsiness or sadness is estimated. Output is sent through the alarm system which will make the driver alert.

1.2 Problem Statement

In a long drive or due to excessive work (fatigue condition) causes drowsiness and instability in emotion. This may lead to accident-causing severe damage of human body and even may lead to death. Intelligence emotion detection has become one of the most used models but still there is a question.

Many emotion detection model have emerged with the improvement of intelligent equipment and Internet of Technology. The model judges the behavior of driver with its facial expression like blinking rate of eyes, yawn, eye closure duration, sadness, joy etc. Intelligent driver emotion detection can be broadly classified into two categories: Client Based Model and Cloud Based Model. In client-based model the client directly judges emotion whereas in cloud-based model the server that is located in the cloud judge the uploaded information. The information is finally passed to the client. Both of this model has its own disadvantage. In client-based model we cannot obtain the as accurate result as needed because it is lacking the dynamic optimization. Whereas in cloud-based model there is a serious issue in user privacy. Also, high security risk to the client because data is collected to the central server in cloud for processing.

Thus, Federated Learning (FL) can solve these both issue by proper training models to the clients, sending only useful information to the central server, optimizing the model through constant aggregation and finally uploading the aggregated model to the clients.

1.3 Objectives

This thesis aims for fulfilling the following objectives:

- To analyze driver behavior using federated learning algorithm (FedSGD and FedAvg).
- To compare performance of Federated Learning and Centralized Machine Learning.

1.4 Organization of the Report

The thesis work is organized in the following order.

Chapter 1 is about Introduction. It has details on the background introduction, problem statement and objectives for this thesis work.

In chapter 2 various literature review is done. Its related work is to thesis are discussed with different works carried out by different authors. This provides the clear gap to setup research work.

Chapter 3 talks about system architecture and the model used in this thesis. It also summarizes the model used.

Chapter 4 gives details results and discussions during research work.

Chapter 5 draws conclusions based on results. It insight on how the work can be further increased and more modifications can be made.

2. LITERATURE REVIEW

The term Federated Learning was coined by Google in 2016. Since then it has been an area of research as evidenced by papers published on arXiv. There are several steps in federated learning. At first central server choose a statistical model to be trained which transmits the initial mode to the several nodes. Different nodes train the model locally with their own available data. Finally, server pools model results then averages which gives one global mode without accessing any data. Thus, privacy is preserved. Data privacy and computation at the edge are the key features of federated learning

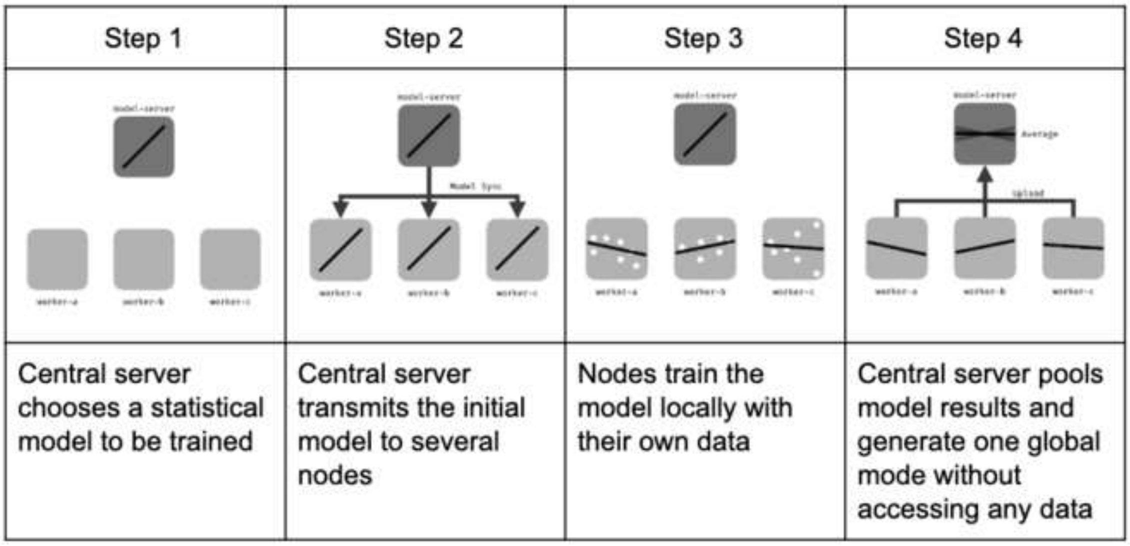

Figure 1: Federated Learning Steps

Federated learning also promises cloud computing, shared economy, business-to-business collaboration. However, heterogeneity between the different local datasets is the limitation.

There are several emotion detection system have developed with different technique like physiological, vehicle based and behavioral based[7]. EGG ECG, EMG etc all comes under physiological. It is expensive as well as time consuming. It makes subject stressful[8][9]. In vehicle based method steering movements, sudden change in acceleration can be used to predict the drowsiness[10]. It need sensor that are attached to the body and its performance decreases with time[7]. In the behavioral based model eye blink rate, yawning etc is used to identify the drowsiness[11]. Machine learning algorithm has played important role for identifying the driver behavior and emotion using the captured images and videos.

There are few research in federated learning as compared to emotion detection. The literature is very limited combining both FL and emotion detection. Here are some of the FL research that is being implemented in with dataset that is different from emotion.

Communication- Efficient Learning of Deep Network from Decentralized Data has implemented FL. These experiments have demonstrated on approach is robust to unbalanced and non-IID data distributions which is characteristic of this setting[2]. Communication costs are principal obligation, and shows there is a significant reduction in required communication rounds as compared to synchronized stochastic gradient descent test. In the experiment carried out in MNIST dataset the accuracy is much higher. However, in case of CIFAR dataset the accuracy is about 85% only after performing 200 communication rounds.

Federated learning with shared level of distribution enables training associatively, a joint model keeping decentralized data for the multiple centers. But its optimizations often suffer from data heterogeneity. The proposed Federated Learning with shared label distribution for classification assumes label data distribution knowledge for all participating clients that is involved in federated learning[3]. It also helps to adjusts the contribution of each data sample to meet the local objective, thus mitigating the instability problem. The accuracy is about 50% with FedAvg and FEDSLD in a CIFAR10 dataset.

Hybrid Facial Expression model has used Deep CNN and Haar Cascade deep learning architecture to classify human emotion. Plot of training loss as well as validation loss declines to the point of stability with the generalization gap of minimal difference[4]. The efficiency is about 70% using the proposed architecture in about 70^{th} communication round. However previous work shows that the efficiency is only about 60% if CNN architecture is used.

In a paper based on vehicular edge computing driver recommendation system using federated learning enables Road Side Units to do all computing of data on it[5]. The model is tested on the UAH-DriveSet dataset. It can be observed that model predicts the stress of a driver with higher accuracy of which assists in enhancing the driving quality and experience and driving quality[5].

Similarly, another paper on A Communication Efficient Federated Learning Fatigue Driving Behaviors Supervision Framework has implanted intelligent fatigue detection. It has used FedSup intelligently that utilizes the collaboration between client, cloud server and edge to realize dynamic model optimization also protecting edge data privacy[6].

From the above research most the emotion detection technique in classical ML is client based. There is limited literature combining both emotion and federated learning. In a client based model all computation is carried locally. It need high computational resource. If emotion detection is implemented in personal car then its scope is limited to that car only. All the learning and knowledge is only for single user. Now a days, due to booming of internet, smartphone and internet supported device the world has transferred. The data can be stored in the cloud. Emotion detection that is computed locally can be done in cloud based model. Client do not need a high computational resource to train the model. However, it need large bandwidth to transfer the data. All the information is transferred so there is a high security risk as well as user privacy to the client. To solve disadvantage of client and cloud based model FL is implemented. FL only transfer the useful information preserving privacy. Thus, if FL for emotion detection is implemented its scope is broader than classical ML. All the learning, computation and knowledge can be shared to the unknown users.

3. SYSTEM ARCHITECTURE AND METHODOLOGY

3.1 Theoretical Modeling

3.1.1 Convolutional Neural Network (CNN)

CNN mainly consists of a Convolutional Layer, Pooling layer and Fully Connected Layer.

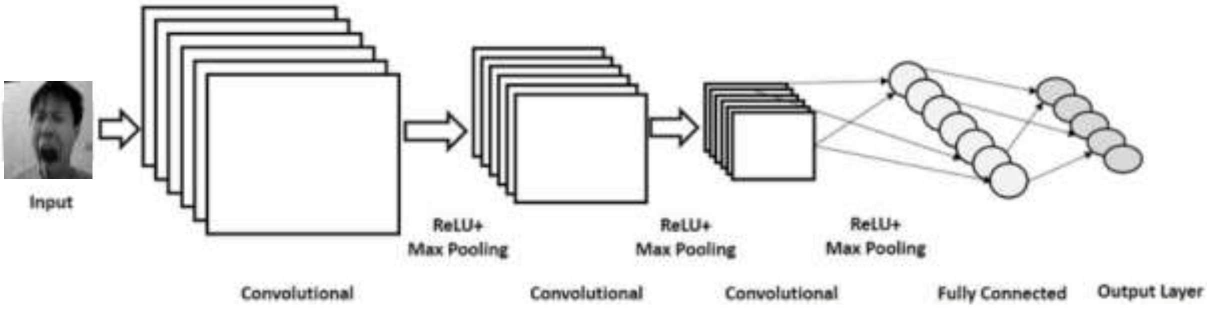

Figure 2: Convolutional Neural Network (CNN)

Convolutional Layer

It is the main building block of CNN where most of the computation is done. It consists of input data (generally color image), filter, and a feature map. Color image is made up of a matrix of pixels in 3D i.e., height, width, and depth. It corresponds to RGB in an image. It also contains a feature detector, also known as a kernel. Kernel moves across the receptive fields of the image which checks whether the feature is present or not. The entire process is called convolution.

The feature detector is 2-Dimension array of weights is the part of the image. The feature detector can vary in accordance to its size. Typically, 3x3 filter size matrix is used. Thus, it also helps for determining the size of the receptive field. The filter is applied to an area of the image. After that dot product is calculated. Dot product is then fed into an output array. Afterwards, the filter shifts by stride. The process repeats until the kernel has swept across the entire image. The final output from the series of dot products from the filter and input image which is also known as a feature map.

Pooling Layer

It performs dimensionality reduction Similar to the convolutional layer, the pooling operation sweeps a filter across the entire input. Here the difference is that this filter does not have any weights, but kernel applies an aggregation function to the values within the receptive field. There are two main types of pooling:

Max Pooling

As the filter moves across the input, it selects the pixel with the maximum value to send to the output array. It is used more often compared to average pooling.

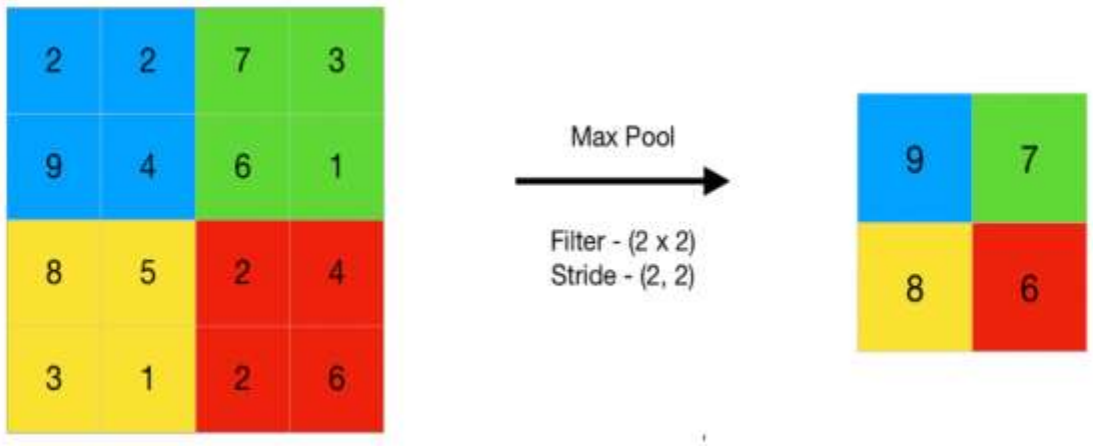

Figure 3: Max Pooling

Average pooling

As the filter moves across the input, it calculates the average value within the receptive field to send to the output array.

Figure 4: Average Pooling

It helps to reduce the risk of over fitting, reduce complexity and improve efficiency.

Fully-Connected Layer

In the fully-connected layer, each node in the output layer connects directly to a node in the previous layer. It performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers different functions are used

Activation Function

ReLU (Rectified Linear Unit) Function

It is mostly used activation function is in CNN. It do not saturate.Its value is zero for negative value of x.

$$\boldsymbol{f(u) = max(0,u)}$$

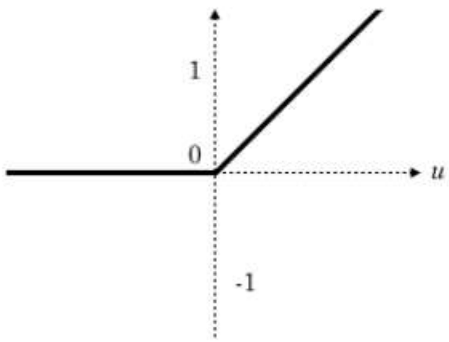

Figure 5: Relu Function

3.2 Federated Learning

There are several steps in federated learning. At the very beginning central server model choose a statistical model to be trained. Then the server model transmits the initial mode to the several nodes which is available online. Different nodes train the model locally with their own available data. Finally, server pools model results, averages them and generate one global mode without accessing any data. Thus, privacy is preserved.

3.3 System Architecture

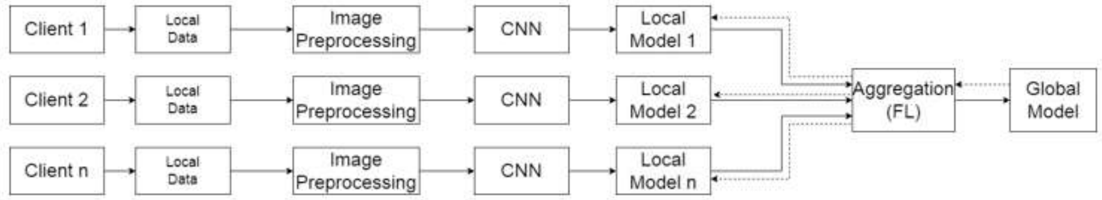

Figure 6: Methodology of IoV based Driver Emotion Detection Using Federated Learning

The system is divided into two parts:

- Clients contain data and performs the local training. It also updates model
- Server initializes the initial model and aggregates the model that is provided by clients.

The detail is explained as:

There are n number of clients. The dataset is divided into n number of clients. At the local edge, local device classifies the dataset and generate the local model. Here we have used CNN to generate the Local Model. All the models that are available is then passed to the aggregator of FL. Finally with the help of an aggregator the global server generates the global model and the model is passed to the available client. This process continues.

3.4 Centralize Machine Learning

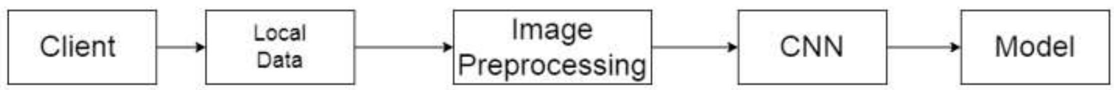

Figure 7: Methodology of IoV based Driver Emotion Detection Using Centralize ML

But in case of centralize ML there is only one client. All the dataset is used but the client for training and testing. Image is preprocessed and passed through CNN which will generate the model.

3.5 Dataset Description

Federated learning system relies heavily on the availability of datasets. The dataset is taken from Kaggle. It consists of 35887 gray images of dimension 48*48 of different facial expressions of Angry (4953 images), Disgust (436 images), Drowsiness (7268 images), Neutral (6198 images), Happy (7215 images), Sadness (6077 images).

Figure 8: Dataset

3.6 Image Augmentation

Augmentation techniques is used to increase the data content simply by adding modified copies of existing data. Techniques like twisting, zooming, rotation etc. are often used for image augmentation.

3.7 Image Preprocessing

The dataset taken contains image of different dimensions. Image preprocessing helps to normalize the data height and width. The color images with different length and breadth are preprocessed.

3.8 Analysis Metrics

To calculate the performance of the proposed system Accuracy, Precision, Recall, F1-Score and Error Rate is calculated.

Accuracy

It is the closeness of measurement to a specific value, given as,

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \tag{1}$$

Precision

Precision is the fraction of correctly classified positive examples divided by the number of examples labeled by the system, given as,

$$Precision = \frac{TP}{TP+FP} \tag{2}$$

Recall

It is also known as true positive rate or sensitivity. It is the probability that the model correctly identifies the anomaly detected.

$$Recall = \frac{TP}{TP+FN} \tag{3}$$

F1-Score

If we want to consider both precision and recall, we take F1-Score. It is the harmonic mean of precision and recall. It gives the single measure of comparison and higher is better.

$$F1 - Score = \frac{2*Precision*Recall}{Precision+Recall} \tag{4}$$

3.9 Tools to be Used

ML is implemented in different types of environments. Programming and modeling is done in Python with the help of different libraries like Tensorflow and Keras.

3.9.1 Python

Python is the most common language today used to build and train neural networks. All major deep learning framework supports Python including Tensorflow, Keras and PyTorch. Many Python frameworks and libraries available for machine learning and deep learning which includes NumPy, scikit-learn, etc. It has a large community supporting the language and thus can be easier to find the solutions to problems[12].

Python language used for following purpose:

- Preprocess images

- Train the image
- Test the image
- Client-server connection
- Combine the model

3.9.2 Scikit-Learn

It is an open-source library which is developed as an extension to SciPy (library) in Python. It provides the different machine learning algorithm implementation such as classification and clustering. It also provides specialized modules such as feature extraction and model review. It is a very popular library among the researches who closely works in machine learning field.

3.9.3 Tensorflow

It is also open-source software library by Google. Tensorflow is popular specially in deep learning for research and production. It is also used to design, build and train deep learning models. It performs numerical computations along with data flow graphs. The data are multidimensional array called tensors.

3.9.4 Keras

Keras is one of the open source neural network library written in Python which runs on top of Tensorflow[13]. Keras also uses another library called "Backend" which is used to handle low-level computations. So, Keras is a high-level API wrapper for low-level API which can run on top of Tensorflow, CNTK, or Theano.

3.9.5 PyTorch

PyTorch is a python package that provides tensor computation (like NumPy) with strong GPU acceleration and deep neural networks built on a tape-based autograd system.

3.10 Proposed System

3.10.1 Simulation Environment

All the experiments are done on the Acer Aspire A515-55.

- Operating System: Microsoft Windows 11 Home (10.0.22621 Built 22621)
- Processor: 1190 MHz Intel Core i5 (4 cores 8 logical Processors)
- RAM: 8 GB
- Software: ANACONDA NAVIGATOR Jupyter Notebook 6.4.12

3.10.2 Proposed Approach

The proposed approach for driver emotion detection is split into three parts:

1. Data pre-processing
2. Feature learning
3. Classification

Data Pre-processing:

Before training the model, a pre-processing is done to extract the face from the images. Data augmentation is also done as a pre-processing step for the training set if there is insufficient data. Two data pre-processing methods are adopted in the proposed approach:

- Face Alignment
- Data Augmentation.

Feature Learning

For training, we used a batch size of 32, and the training is done for 45-49 epochs. We have tested the performance in different learning rate.

Classification

After learning features the final step for the emotion is to classify the faces into one of the existing emotion categories. There are two approaches for feature classification,

- The feature extraction step
- The feature classification step

Both steps are independent. The feature extraction step and the feature classification steps are performed in an end-to-end way. For our approach, we have used the architecture as shown in Figure. The classification step is performed using a Labeled data.

Input Image
|
Convolutional Layer
Batch Normalization Layer
ReLU Activation Function
Dropout Layer
|
Convolutional Layer
Batch Normalization Layer
ReLU Activation Function
Dropout Layer
|
Convolutional Layer
Batch Normalization Layer
ReLU Activation Function
Max Pooling Layer
Dropout Layer
|
Convolutional Layer
Batch Normalization Layer
ReLU Activation Function
Dropout Layer
|
Convolutional Layer
Batch Normalization Layer
ReLU Activation Function
Dropout Layer
|
Convolutional Layer
Batch Normalization Layer
ReLU Activation Function
Max Pooling Layer
Dropout Layer
|
Convolutional Layer
Batch Normalization Layer
ReLU Activation Function
Dropout Layer
|
Convolutional Layer

Batch Normalization Layer
ReLU Activation Function
Max Pooling Layer
Dropout Layer
|
Fully Connected Layer
ReLU Activation Function
Fully Connected Layer
ReLU Activation Function
Fully Connected Layer
SoftMax Activation Function
|
Output(Class Probabilities)

Figure 9: Architecture used in CNN

FL is attached with the proposed classifier. In FL there are two steps

- Server Initialization phase
- participant end

For this thesis there are two algorithms used.

- FedAvg
- FedSGD

3.10.3 Algorithm used in FedAvg.

At Server:

The following steps are performed in server initialization phase:

Let,

N number of participant

n number of selected nodes

where, N<=n

d number of instances that is used for training purpose

w Weight parameter

t Time stamp

At first weight is initialized

For each time stamp "t" from 1 to T do

Shuffle N Participants

Select "n" participant from the list

For each participant "p" from 1 to n Parallelly Do

d,p←Local Training(p , w)

w(t+1)←Weighted Average of w(t)

At Client

Let, Input

P Participant

E Epochs

D Dataset Length

B minibatch Size

Δl Gradient

w Weight

n learning Rate

For each p participant do

Split dataset into minibatch size B

Each minibatch size is represented as Bs

For each "e" epoch in 1 to E do

For each "b" in Bs do

w←w-n.Δl(w,b)

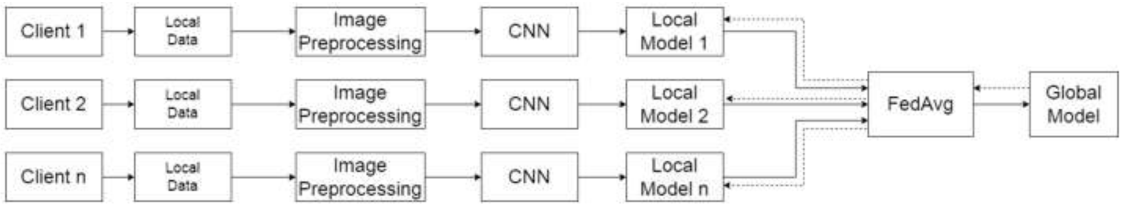

Figure 10: Block Diagram of FedAvg

Figure shows the block diagram of FedAvg. There are n number of clients. The client that are active will participate in learning process. The local dataset available will be preprocessed and passed through the CNN. The detail in CNN architecture is already explained in figure 9. Output of CNN will generate the local model. Different model generated by different client is then passed through the FedAvg. The algorithm used is explained above. The output of FedAvg generates the global model. This global model is finally transfer to all the clients.

3.10.4 Algorithm used in FedSGD

Same as FedAvg but instead of weight gradient values are updated in each round.

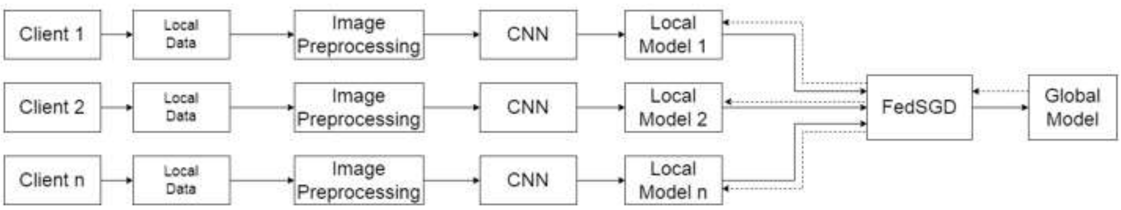

Figure 11: Block Diagram of FedSGD

3.10.5 Federated Learning Averaging to Generate a Global Model

After the classification of image, all the image is passed through the aggregator and generates the global model. This global model is then again passed to the local client.

3.11 Verification and Validation

The Verification and Validation is a method used to evaluate the implemented model of the system. The verification process includes checking of the output of the model. It is divided into three parts. We train the model using the training data set. Parameter tuning is done by validation set and finally performance is evaluated using test set.

4. RESULT

Before emotion test the FL algorithm is tested in the MNIST dataset. The motivation behind is to validate the FL algorithm with the standard dataset. The total number of clients is 100, and at a time 10 client is selected. The number of rounds performed is 5 with the 5 epochs.

The following performance is observed.

S.NO.	Training Accuracy	Test Accuracy
1	0.275	0.864
2	0.766	0.927
3	0.857	0.944
4	0.9659	0.952
5	0.9788	0.959

Table 1 : Training Accuracy and Test Accuracy for federated learning using MNIST dataset

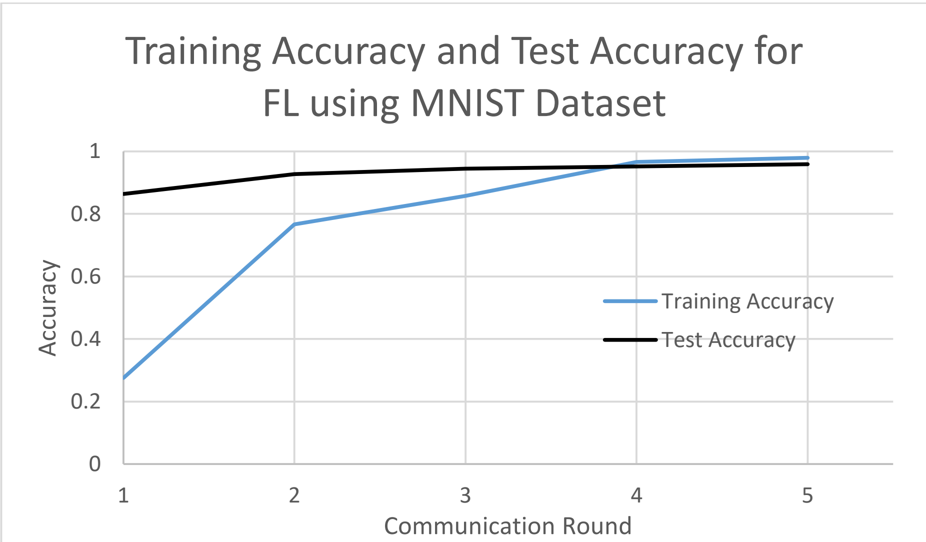

Figure 12: Training Accuracy for Federated Learning using MNSIT Dataset

From the table it is clear that the accuracy of the model is increased in the MNIST dataset using federated learning. Thus the algorithm is valid.

Different experiment setup is carried out to analyze the result. Due to time complexity, we have carried all the experiment maximum up to 45-49 communication rounds. The result is then extrapolated to analyze the result.

```
loss 1.8811540603637695
loss 3.770725727081299
loss 5.60853111743927
loss 7.484653353691101
loss 9.3439382314682
0-th round
average train loss 1.87 |  test acc: 0.245
loss 1.713935136795044
loss 3.571321129798889
loss 5.353814721107483
loss 7.0908414125442505
loss 8.920645594596863
1-th round
average train loss 1.78 |  test acc: 0.361
loss 1.81290864944458
loss 3.5794615745544434
loss 5.339846611022949
loss 7.151318311691284
loss 8.881652474403381
2-th round
average train loss 1.78 |  test acc: 0.407
```

Figure 13: Loss while training

Figure 12 is instant of loss during training phase.

First, the performance of Federated Learning algorithm is compared with the centralized ML algorithm. In this case learning rate is taken as 0.0001 for ML as well as Federated ML. In this experiment FedAvg algorithm is used. All the other parameter are same in both the case.

The following performance is observed.

S.No	Loss using FL	Accuracy using FL	Loss without FL	Accuracy without FL
1	1.91	0.246	1.83	0.312
2	1.86	0.273	1.79	0.385
3	1.78	0.362	1.74	0.402
4	1.85	0.386	1.72	0.439
5	1.77	0.395	1.7	0.464
6	1.69	0.41	1.69	0.478
7	1.72	0.449	1.68	0.476
8	1.7	0.466	1.67	0.503
9	1.71	0.47	1.66	0.502
10	1.72	0.476	1.65	0.502
11	1.69	0.487	1.65	0.507
12	1.65	0.478	1.64	0.507
13	1.64	0.487	1.64	0.533
14	1.67	0.494	1.63	0.535

15	1.71	0.498	1.63	0.543
16	1.71	0.495	1.62	0.547
17	1.64	0.501	1.62	0.538
18	1.65	0.508	1.62	0.534
19	1.67	0.503	1.61	0.538
20	1.63	0.508	1.61	0.548
21	1.6	0.505	1.61	0.549
22	1.61	0.508	1.61	0.556
23	1.64	0.511	1.6	0.545
24	1.61	0.51	1.6	0.558
25	1.62	0.509	1.6	0.56
26	1.6	0.514	1.6	0.56
27	1.59	0.518	1.59	0.551
28	1.68	0.516	1.59	0.564
29	1.66	0.525	1.59	0.571
30	1.66	0.528	1.59	0.566
31	1.58	0.529	1.59	0.573
32	1.68	0.533	1.59	0.567
33	1.63	0.534	1.58	0.572
34	1.66	0.536	1.58	0.582
35	1.61	0.546	1.58	0.567
36	1.59	0.546	1.58	0.58
37	1.61	0.554	1.58	0.582
38	1.67	0.544	1.58	0.571
39	1.58	0.546	1.57	0.583
40	1.58	0.55	1.57	0.578
41	1.61	0.558	1.57	0.58
42	1.58	0.557	1.57	0.579
43	1.55	0.555	1.57	0.584
44	1.58	0.555	1.57	0.591
45	1.57	0.557	1.57	0.602

Table 2 :Loss and Accuracy with and without using Federated Learning

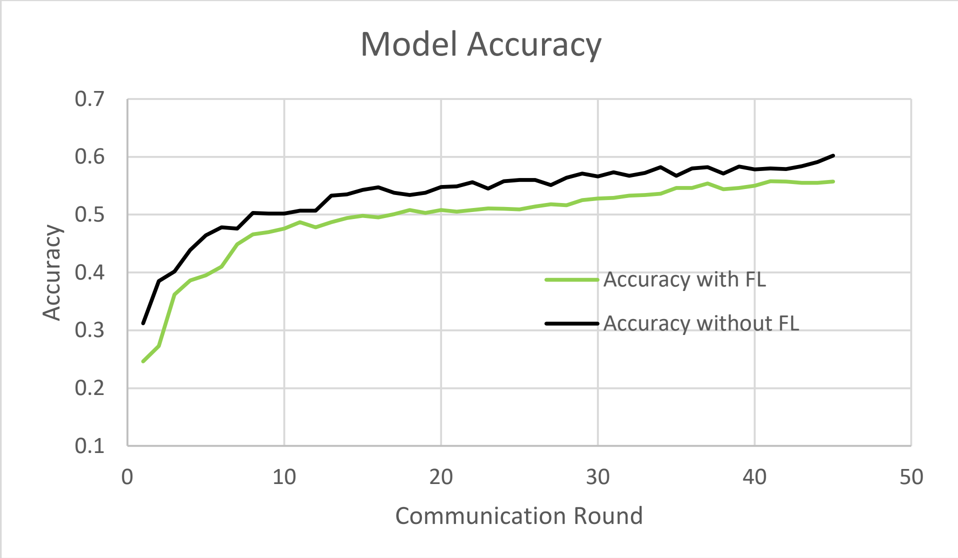

Figure 14: Accuracy with and without using FL

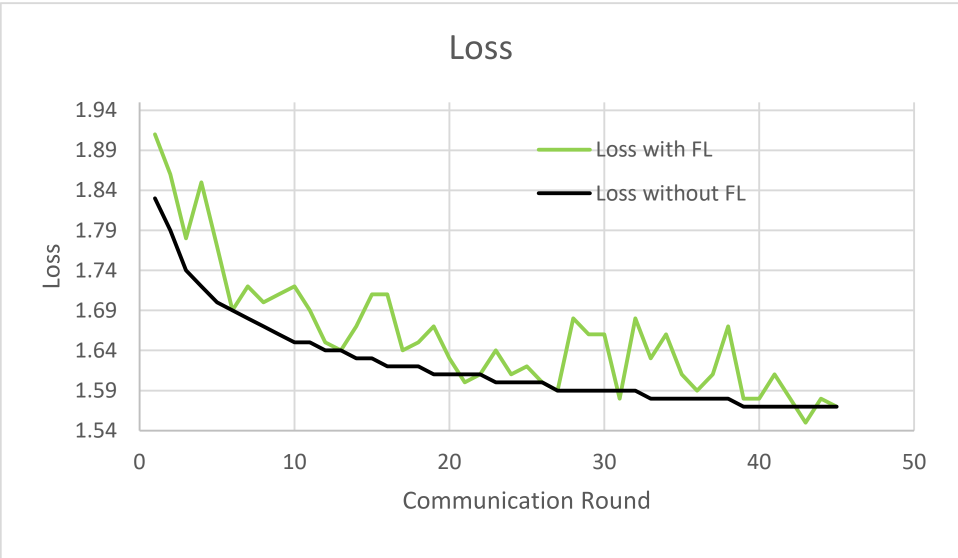

Figure 15: Loss with and without using FL

From the above experiment it can be observed that accuracy of emotion detection is better in centralized ML than FL. Out of 45 communications round the best accuracy can be observed in 41st communication round in FL with accuracy of 55.8%. The accuracy using centralize ML algorithm is 60.2% in the 45^{th} communication round. There is no significance difference in the accuracy with and without using FL however performance of Federated Learning algorithm is slightly lower as compared to centralize ML algorithm. This is quite obvious because data is distributed evenly. Their accuracy is almost same but due to preservation of privacy of the user data FL algorithm is preferred. From the losses curve it is observed that loss in centralized ML algorithm is smoother than Federated Learning Algorithm. The randomness has occurred due to uneven distribution of data.

In another experiment Federated Learning algorithms (FedSGD and FedAvg) is compared. Learning rate is taken as 0.00001. The number of clients is 10 and 10 clients take participation. All the other parameter are same in both the case.

S.No.	Loss FedSGD	Accuracy FedSGD	Loss FedAvg	Accuracy FedAvg
1	1.91	0.245	1.91	0.247
2	1.91	0.245	1.81	0.305
3	1.9	0.245	1.79	0.36
4	1.87	0.245	1.75	0.394
5	1.88	0.245	1.76	0.425
6	1.9	0.245	1.71	0.44
7	1.92	0.245	1.69	0.455
8	1.89	0.245	1.7	0.46
9	1.88	0.245	1.69	0.464
10	1.87	0.245	1.68	0.477
11	1.88	0.245	1.65	0.482
12	1.87	0.245	1.65	0.484
13	1.87	0.252	1.66	0.495
14	1.82	0.267	1.62	0.502
15	1.84	0.279	1.68	0.502
16	1.84	0.298	1.66	0.515
17	1.84	0.319	1.65	0.52
18	1.86	0.332	1.65	0.519
19	1.82	0.339	1.7	0.513
20	1.84	0.342	1.64	0.529
21	1.78	0.35	1.57	0.527
22	1.81	0.355	1.65	0.523
23	1.86	0.352	1.66	0.537
24	1.78	0.368	1.64	0.527
25	1.83	0.368	1.63	0.541
26	1.8	0.376	1.66	0.54
27	1.78	0.377	1.63	0.547
28	1.81	0.373	1.58	0.543
29	1.78	0.369	1.63	0.548
30	1.77	0.371	1.61	0.545

31	1.78	0.375	1.65	0.547
32	1.77	0.379	1.63	0.551
33	1.78	0.383	1.61	0.554
34	1.78	0.382	1.6	0.554
35	1.84	0.391	1.62	0.556
36	1.78	0.386	1.56	0.559
37	1.79	0.387	1.54	0.562
38	1.77	0.396	1.62	0.563
39	1.73	0.399	1.61	0.565
40	1.74	0.402	1.56	0.562
41	1.77	0.39	1.6	0.561
42	1.78	0.399	1.56	0.567
43	1.78	0.395	1.65	0.57
44	1.76	0.397	1.65	0.571
45	1.75	0.395	1.55	0.561
46	1.78	0.403	1.51	0.565
47	1.79	0.398	1.57	0.575
48	1.75	0.408	1.57	0.57

Table 3:Loss and accuracy using FedAvg and FedSGD

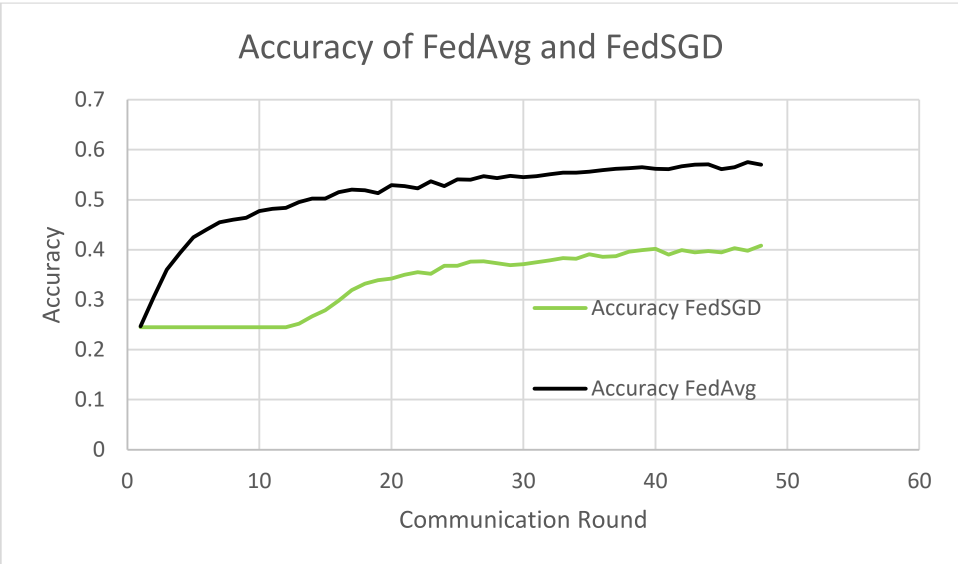

Figure 16: Accuracy using FedAvg and FedSGD

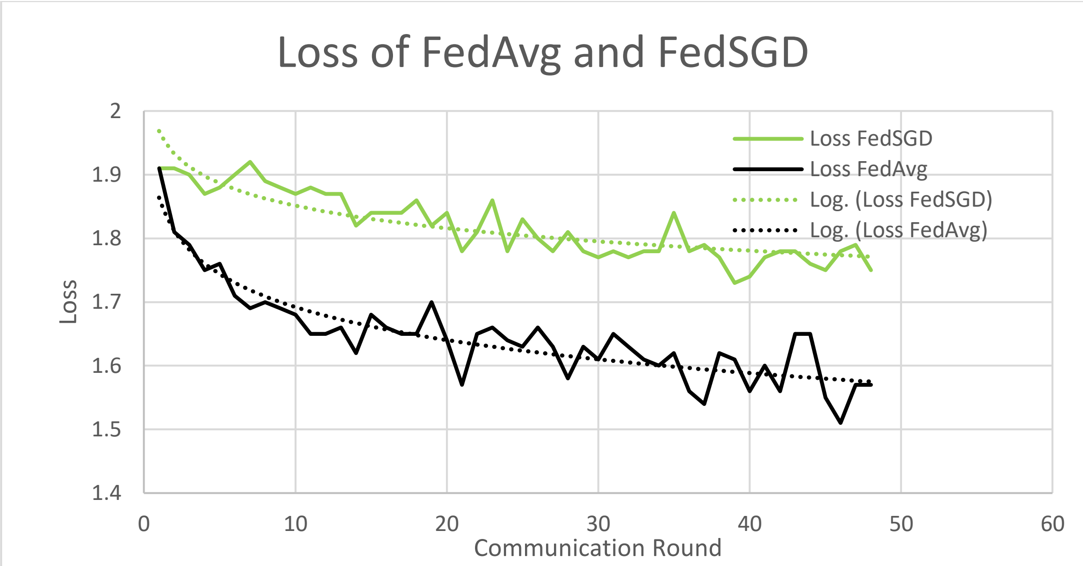

Figure 17: Loss using FedAvg and FedSGD

From the above graph it is seen that accuracy of FedAvg is better than accuracy of FedSGD. The accuracy is almost constant for FedSGD up to round 12 and then increases gradually. The maximum accuracy obtained while computing 48th round is 40.8% for FedSGD at 48th round whereas accuracy of FedAvg is 57.5% at 47th round. The convergence step of FedSGD is lower than that of FedAvg. It requires much more step for conversion. Losses is also higher in FedSGD as compared to FedAvg. The dotted line is extrapolated result.

Another experiment set up for comparing performance of FedSGD and FedAvg has learning rate 0.0001. The total number of clients is 10 and out of 10 clients 10 clients take participation. All the other parameter are same in both the case.

S.No.	Loss FedSGD	Accuracy SGD	Loss FedAvg	Accuracy FedAvg
1	1.9	0.245	1.84	0.245
2	1.87	0.244	1.76	0.354
3	1.86	0.264	1.8	0.418
4	1.87	0.331	1.76	0.451
5	1.78	0.354	1.72	0.471
6	1.83	0.364	1.67	0.474
7	1.77	0.391	1.68	0.484
8	1.79	0.399	1.64	0.498
9	1.76	0.412	1.71	0.511

10	1.75	0.419	1.61	0.522
11	1.78	0.427	1.66	0.53
12	1.76	0.435	1.62	0.537
13	1.76	0.447	1.66	0.533
14	1.79	0.454	1.56	0.55
15	1.78	0.463	1.62	0.546
16	1.76	0.455	1.67	0.552
17	1.68	0.458	1.58	0.556
18	1.69	0.471	1.55	0.562
19	1.68	0.468	1.63	0.566
20	1.69	0.476	1.6	0.57
21	1.73	0.481	1.65	0.578
22	1.7	0.48	1.6	0.577
23	1.74	0.482	1.61	0.584
24	1.71	0.482	1.59	0.582
25	1.75	0.486	1.57	0.59
26	1.64	0.495	1.59	0.586
27	1.71	0.494	1.66	0.592
28	1.67	0.502	1.63	0.591
29	1.67	0.502	1.59	0.6
30	1.72	0.499	1.63	0.601
31	1.65	0.506	1.55	0.608
32	1.73	0.509	1.55	0.611
33	1.66	0.499	1.66	0.6
34	1.67	0.515	1.66	0.62
35	1.66	0.511	1.59	0.62
36	1.7	0.519	1.56	0.628
37	1.67	0.524	1.54	0.629
38	1.72	0.522	1.56	0.63
39	1.62	0.522	1.58	0.638
40	1.67	0.532	1.53	0.634
41	1.69	0.526	1.58	0.63
42	1.6	0.53	1.62	0.634
43	1.61	0.522	1.58	0.638
44	1.63	0.537	1.58	0.632
45	1.68	0.534	1.58	0.631
46	1.61	0.534	1.5	0.638
47	1.57	0.538	1.59	0.638
48	1.65	0.534	1.56	0.639

Table 4: Loss and accuracy using FedAvg and FedSGD

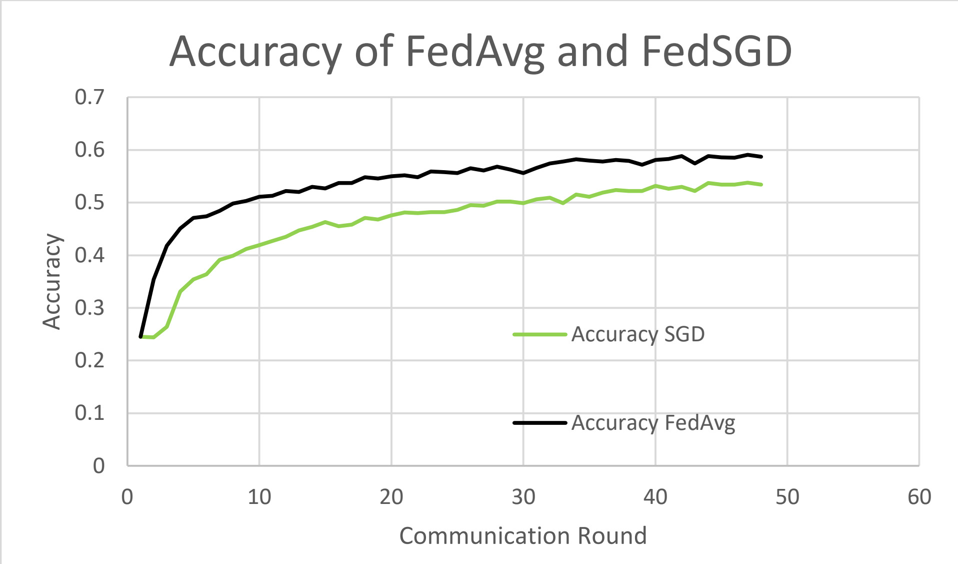

Figure 18: Accuracy using FedAvg and FedSGD

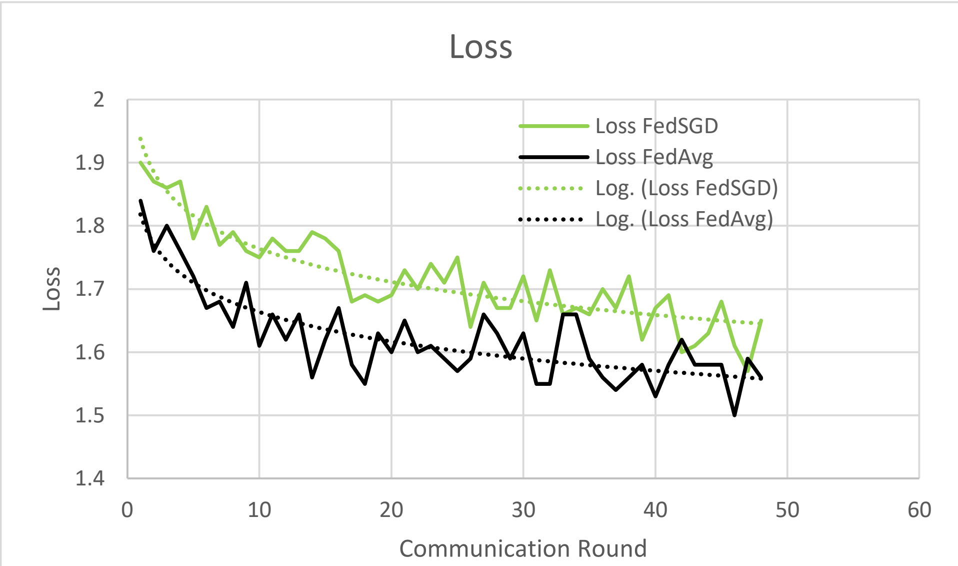

Figure 19: Loss using FedAvg and FedSGD

From the graph it can be observed that accuracy of FedAvg is better than FedSGD. Maximum accuracy is obtained at 48^{th} round having accuracy of 53.8% and 63.9% The doted line is extrapolated result.

In another experiment accuracy and loss are compared with learning rate 0.0001. The number of clients taken is 10 and out of 10, different clients take part with all the other parameter remaining constant.

S.No.	Loss (N=10 n=5)	Accuracy (N=10 n=5)	Loss (N=10 n=10)	Accuracy (N=10 n=10)	Loss (N=10 n=7)	Accuracy (N=10 n=7)
1	1.91	0.244	1.76	0.256	1.91	0.246
2	1.87	0.245	1.64	0.432	1.86	0.273
3	1.87	0.245	1.73	0.475	1.78	0.362
4	1.89	0.245	1.65	0.494	1.85	0.386
5	1.9	0.245	1.64	0.508	1.77	0.395
6	1.9	0.252	1.62	0.526	1.69	0.41
7	1.89	0.27	1.59	0.533	1.72	0.449
8	1.83	0.296	1.6	0.543	1.7	0.466
9	1.83	0.32	1.58	0.542	1.71	0.47
10	1.83	0.328	1.57	0.543	1.72	0.476
11	1.79	0.334	1.56	0.553	1.69	0.487
12	1.78	0.342	1.58	0.561	1.65	0.478
13	1.76	0.358	1.58	0.571	1.64	0.487
14	1.8	0.368	1.56	0.568	1.67	0.494
15	1.74	0.386	1.51	0.573	1.71	0.498
16	1.79	0.391	1.55	0.568	1.71	0.495
17	1.77	0.396	1.6	0.569	1.64	0.501
18	1.77	0.399	1.59	0.573	1.65	0.508
19	1.79	0.413	1.57	0.573	1.67	0.503
20	1.69	0.416	1.56	0.58	1.63	0.508
21	1.73	0.416	1.54	0.586	1.6	0.505
22	1.76	0.413	1.5	0.583	1.61	0.508
23	1.76	0.419	1.54	0.582	1.64	0.511
24	1.77	0.42	1.55	0.594	1.61	0.51
25	1.74	0.433	1.64	0.585	1.62	0.509
26	1.78	0.426	1.52	0.59	1.6	0.514
27	1.73	0.434	1.53	0.589	1.59	0.518
28	1.75	0.438	1.46	0.592	1.68	0.516
29	1.7	0.439	1.55	0.591	1.66	0.525
30	1.7	0.439	1.49	0.602	1.66	0.528
31	1.74	0.434	1.53	0.596	1.58	0.529
32	1.72	0.444	1.55	0.593	1.68	0.533
33	1.72	0.448	1.43	0.602	1.63	0.534
34	1.77	0.451	1.44	0.57	1.66	0.536
35	1.73	0.453	1.53	0.601	1.61	0.546
36	1.68	0.451	1.49	0.61	1.59	0.546

37	1.71	0.449	1.53	0.604	1.61	0.554
38	1.71	0.45	1.5	0.618	1.67	0.544
39	1.71	0.461	1.56	0.62	1.58	0.546
40	1.7	0.454	1.57	0.619	1.58	0.55
41	1.75	0.459	1.53	0.619	1.61	0.558
42	1.72	0.461	1.46	0.616	1.58	0.557
43	1.72	0.463	1.47	0.62	1.55	0.555
44	1.71	0.465	1.44	0.62	1.58	0.555
45	1.68	0.464	1.53	0.622	1.57	0.557
46	1.7	0.462	1.5	0.623	1.56	0.561
47	1.67	0.469	1.48	0.622	1.55	0.563
48	1.67	0.471	1.45	0.624	1.52	0.564
49	1.7	0.469	1.53	0.627	1.64	0.575

Table 5: Loss and accuracy using different Federated Learning Algorithm

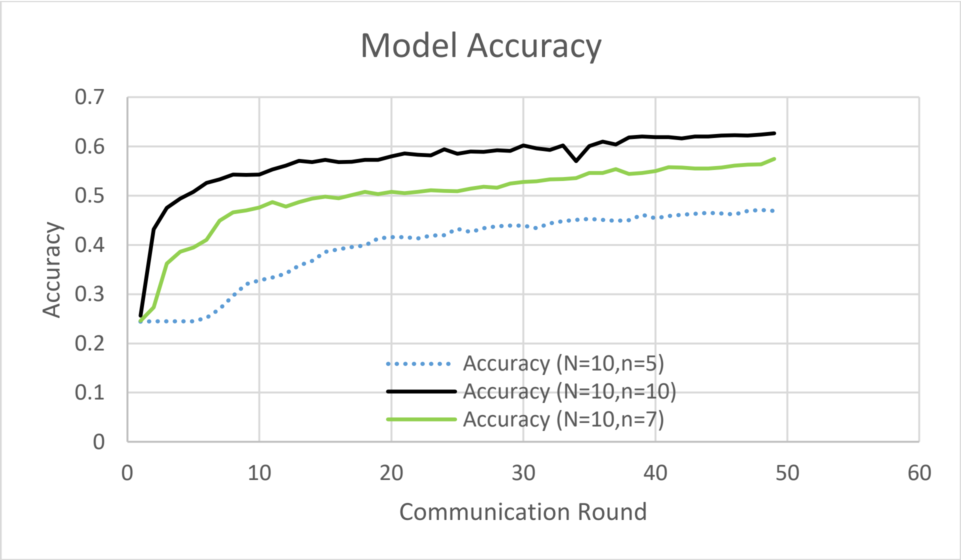

Figure 20: Accuracy of FL with different n

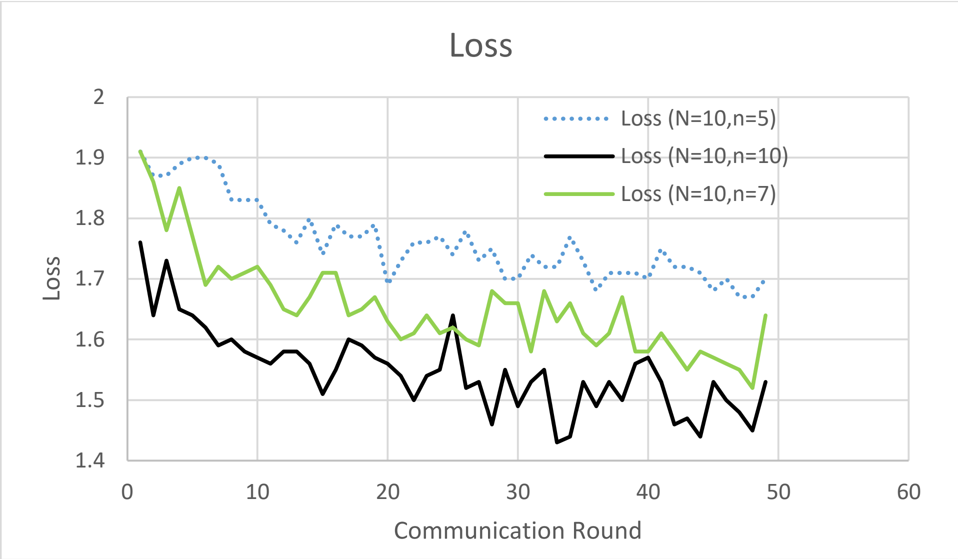

Figure 21: Loss of FL with different n

From the above graph it is clear that FL with N=10, n= 5 has highest accuracy and low loss. The client that takes part in federated learning also change in accuracy. If there is more dataset for a client than its accuracy increases to some level.

Here are some of the result. The first word in every picture are actual values whereas word within the bracket are predicted output.

Figure 22: Actual vs Predicted Classes

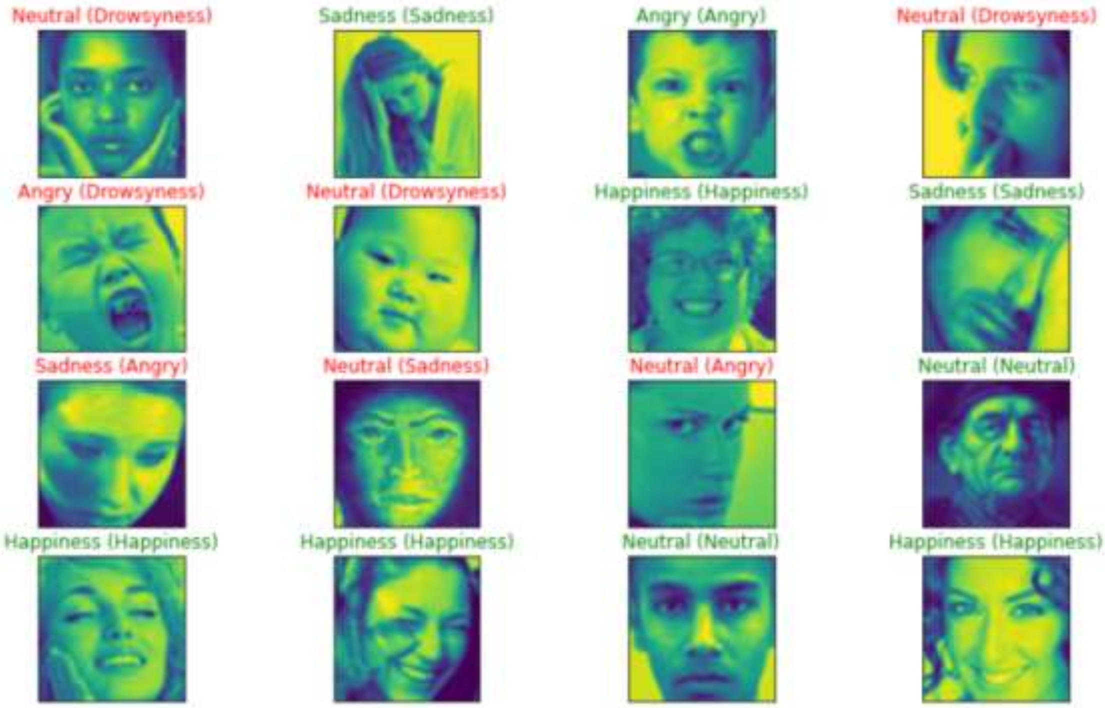

Figure 23: Actual vs predicted classes.

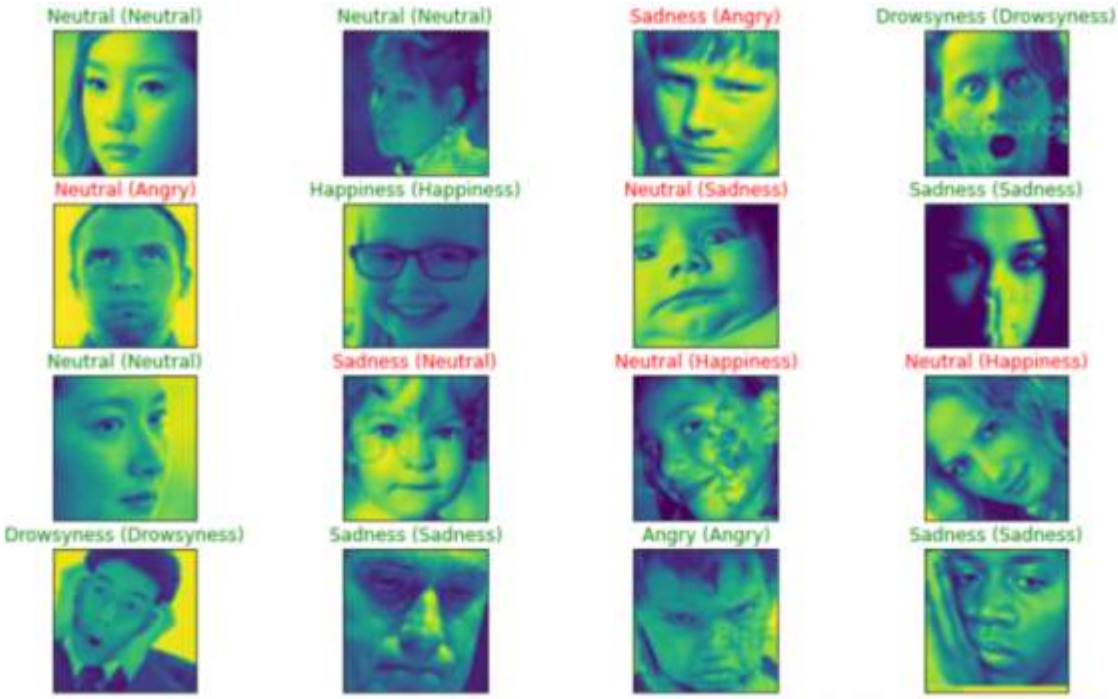

Figure 24: Actual vs Predicted Class

5. CONCLUSION

This work proposed a concept of federated machine learning in driver behavior detection while driving. Even though centralize ML has higher accuracy than FL, FL still has advantage of user privacy. Two different algorithms i.e., FedAvg and FedSGD are analyzed and found that the overall accuracy of FedAvg is better than FedSGD. FedSGD takes more steps to converge than FedAvg.

Future Enhancement

Although the accuracy of FedAvg is better than FedSGD the convergence steps is still high. Due to limited resources only 45-49 communication round is performed. In future, using a high resource computer, more communication round can be performed. Other technique in FL can be explored to reduce the conversion steps. Also, different convolutional neural architectures can be compared.

REFERENCES

[1] "What is Machine Learning? | IBM." https://www.ibm.com/cloud/learn/machine-learning (accessed May 18, 2022).

[2] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," Feb. 2016, [Online]. Available: http://arxiv.org/abs/1602.05629

[3] J. Luo and S. Wu, "Fedsld: Federated Learning with Shared Label Distribution for Medical Image Classification," *Proc. - Int. Symp. Biomed. Imaging*, vol. 2022-March, pp. 1–10, 2022, doi: 10.1109/ISBI52829.2022.9761404.

[4] *et al.*, "Hybrid Facial Expression Recognition (FER2013) Model for Real-Time Emotion Classification and Prediction," *BOHR Int. J. Internet things, Artif. Intell. Mach. Learn.*, vol. 1, no. 1, pp. 63–71, 2022, doi: 10.54646/bijiam.011.

[5] J. Vyas, D. Das, and S. K. Das, "Vehicular edge computing based driver recommendation system using federated learning," *Proc. - 2020 IEEE 17th Int. Conf. Mob. Ad Hoc Smart Syst. MASS 2020*, pp. 675–683, 2020, doi: 10.1109/MASS50613.2020.00087.

[6] C. Zhao, Z. Gao, Q. Wang, K. Xiao, Z. Mo, and M. J. Deen, "FedSup: A Communication-Efficient Federated Learning Fatigue Driving Behaviors Supervision Framework," 2021, [Online]. Available: http://arxiv.org/abs/2104.12086

[7] J. S. Femilda Josephin, C. Lakshmi, and S. Joseph James, "A review on the measures and techniques adapted for the detection of driver drowsiness," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 993, no. 1, 2020, doi: 10.1088/1757-899X/993/1/012101.

[8] Z. Mardi, S. Naghmeh, M. Ashtiani, and M. Mikaili, "Jmss-1-130," vol. 1, no. 2,

pp. 130–137, 2011.

[9] J. Cui *et al.*, “Subject-Independent Drowsiness Recognition from Single-Channel EEG with an Interpretable CNN-LSTM model,” *Proc. - 2021 Int. Conf. Cyberworlds, CW 2021*, pp. 201–208, 2021, doi: 10.1109/CW52790.2021.00041.

[10] Z. Li, S. E. Li, R. Li, B. Cheng, and J. Shi, “Online detection of driver fatigue using steering wheel angles for real driving conditions,” *Sensors (Switzerland)*, vol. 17, no. 3, pp. 1–12, 2017, doi: 10.3390/s17030495.

[11] O. Ursulescu, B. Ilie, and G. Simion, “Driver Drowsiness Detection Based on Eye Analysis,” *2018 13th Int. Symp. Electron. Telecommun. ISETC 2018 - Conf. Proc.*, no. June, 2018, doi: 10.1109/ISETC.2018.8583852.

[12] “Python AI: How to Build a Neural Network & Make Predictions – Real Python.” https://realpython.com/python-ai-neural-network/ (accessed May 18, 2022).

[13] “About Keras.” https://keras.io/about/ (accessed May 21, 2022).