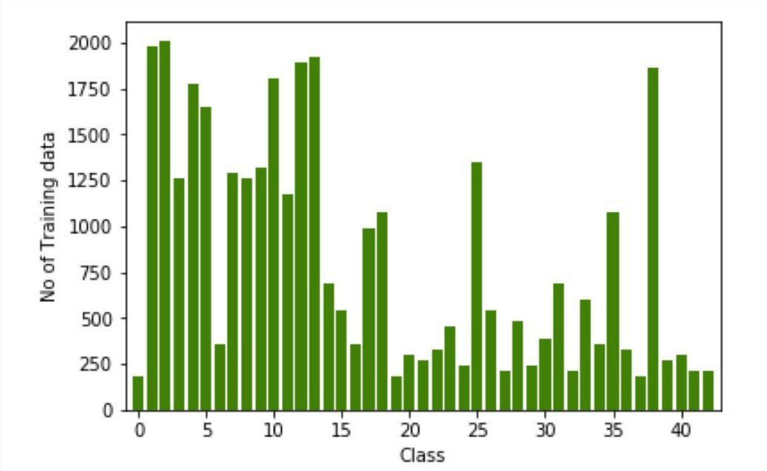
Traffic Sign Recognition
Federated Learning

Abstract :

The main aim of the following model is traffic sign recognition. The data set is related to traffic signs and this dataset contains around 50000 images of different traffic signs. The deep learning model has been made in which three-fourth of the data set has been used for the training, one-fourth for the testing and the accuracy is plotted. The federated learning model has been made in which we assume that there are three users who have the equal amount of the data, the final model is tested on test data and the accuracy is plotted(we divided the three-fourth of data set such that each user has the same amount of data and one-fourth of the data set has been used to test the model).

Dataset Understanding :

The dataset used here contains around 40000 train data and 12000 test data.The data is distributed among 43 different classes,here classes mean the different traffic signs and their distribution is as follows



Acknowledgement :

Nowadays we see a rising demand for self driving cars or autonomous vehicles, in which the automated system should be able to recognize the traffic signs in front of the car to take the required decision and corresponding action. So there is a need for a highly accurate model which recognizes the signs, for this we require huge amounts of data to train. So federated learning helps us to achieve accuracy by following the privacy protocols.

Model Architecture :

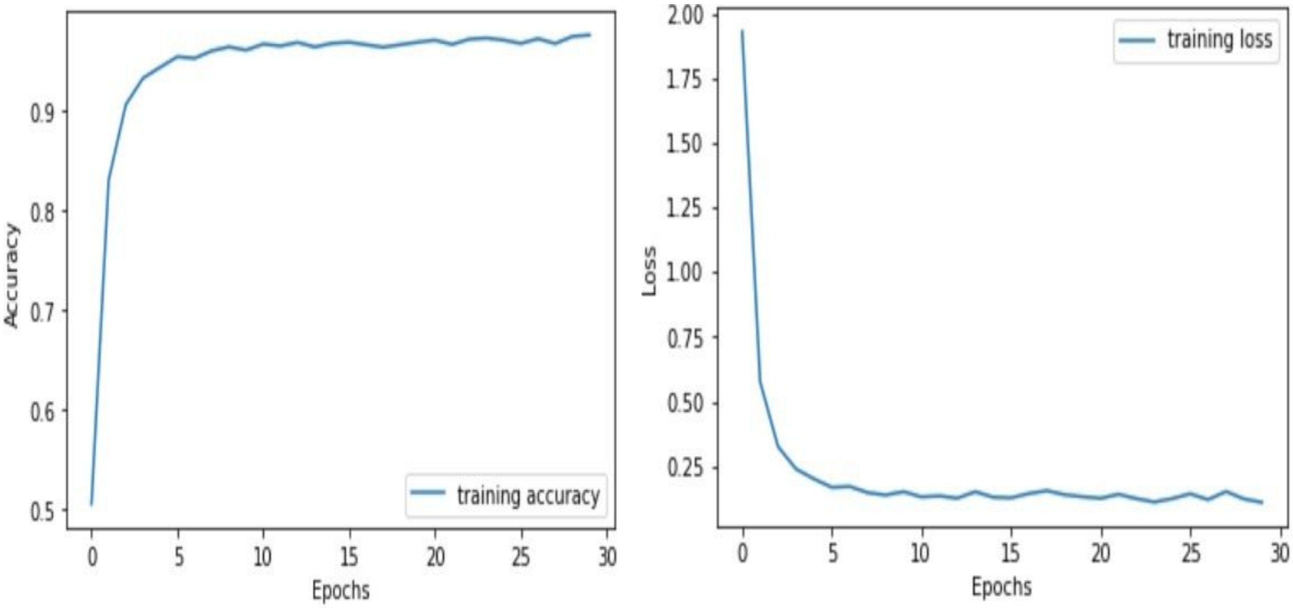
This is a multi-classification problem. The model used here is **Convolution neural network** which consists of Convolutional two dimensional layer, Max Pooling layer, Dense layer, Dropout regularization and flattening of convolution layer into simple neural network. There are totally 12 layers in the model architecture and they are described as follows
We have chosen a sequential model of keras.

- **2 Conv2D layer (filter=32, kernel_size=(5,5), activation=relu)**
- **maxPool2D layer (pool_size=(2,2))**
- **dropout layer (rate=0.25)**
- **2 Conv2D layer (filter=64, kernel_size=(3,3), activation=relu)**
- **maxPool2D layer (pool_size=(2,2))**
- **dropout layer (rate=0.25)**
- **flatten layer**
- **dense Fully connected layer (256 nodes, activation=relu)**
- **dropout layer (rate=0.5)**
- **dense layer (43 nodes, activation=softmax)**

Procedure:

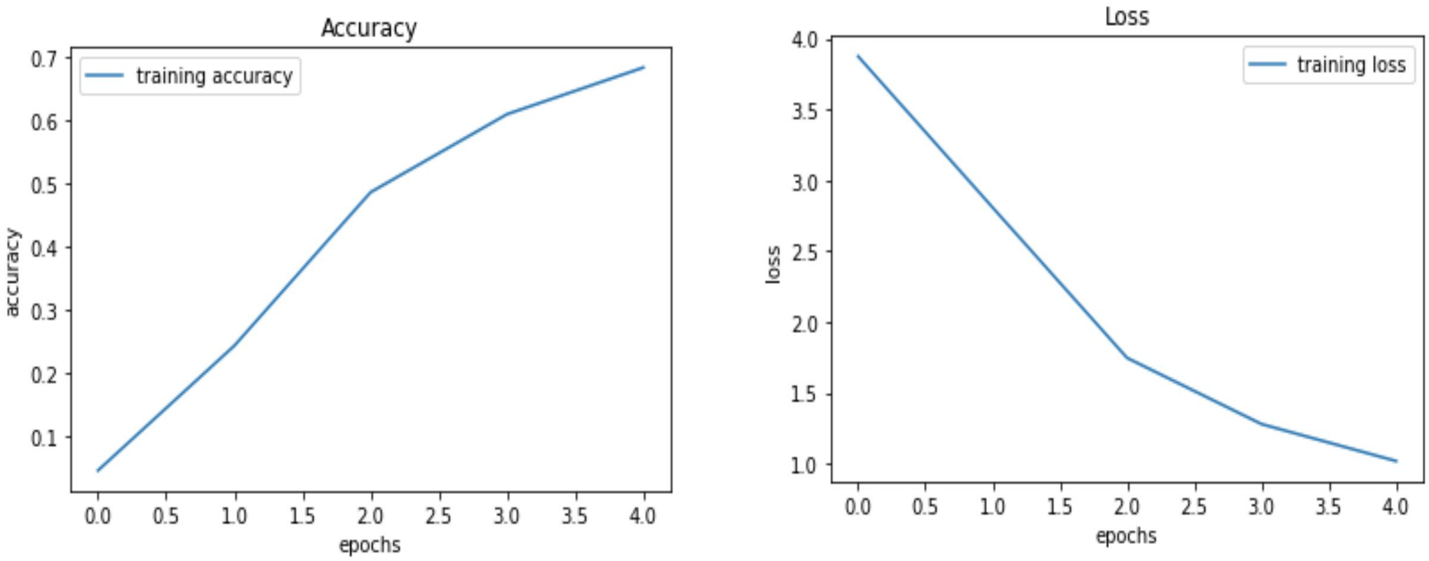
- The data visualization of the obtained data is done in the initial step.
- The deep learning model is implemented
    - The data set is divided such that three-fourth of it is used for training the model and the one-fourth is used to test the model.
- The federated learning model is implemented
    - There are three users with the equal data which is one-fourth the size of the complete data set.
    - The initial parameters are sent to the users from the organization, who train the model with their data and send the parameters to the organization where all the parameters are weighted averaged.
    - The process goes on several times (Here it is 6 iterations with each having 5 epochs).
    - The model is designed with the global parameters obtained in the organization on the test data which is one-fourth of the complete data set.

Experimental evaluation and Conclusion:

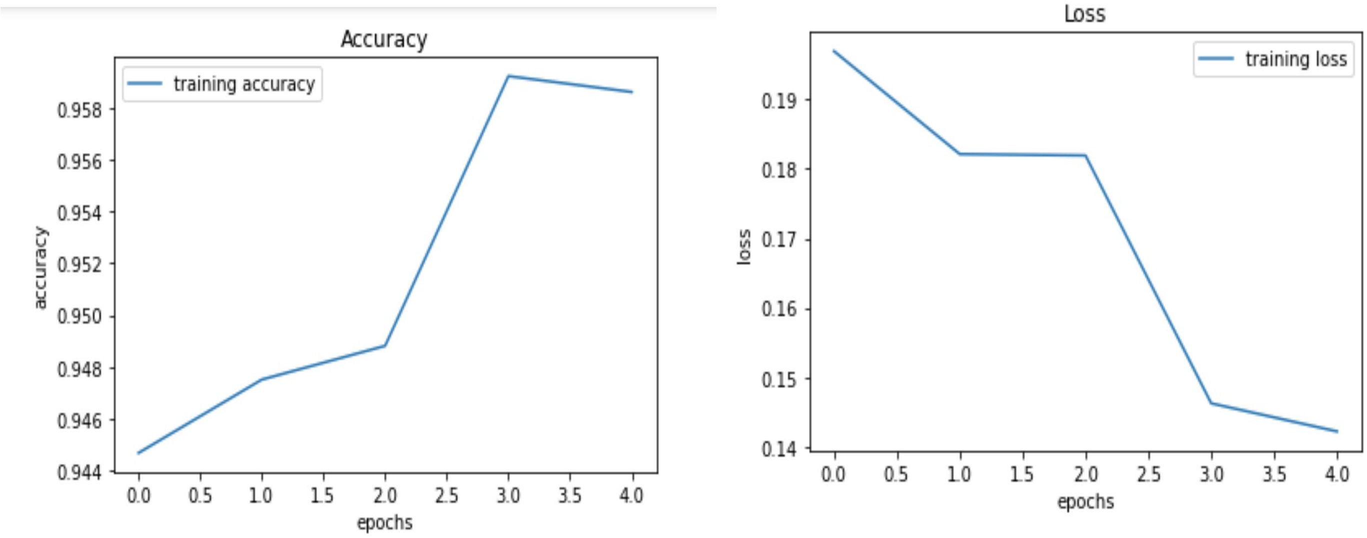


The accuracy vs epoch and loss vs epoch graphs of deep learning model during every iteration

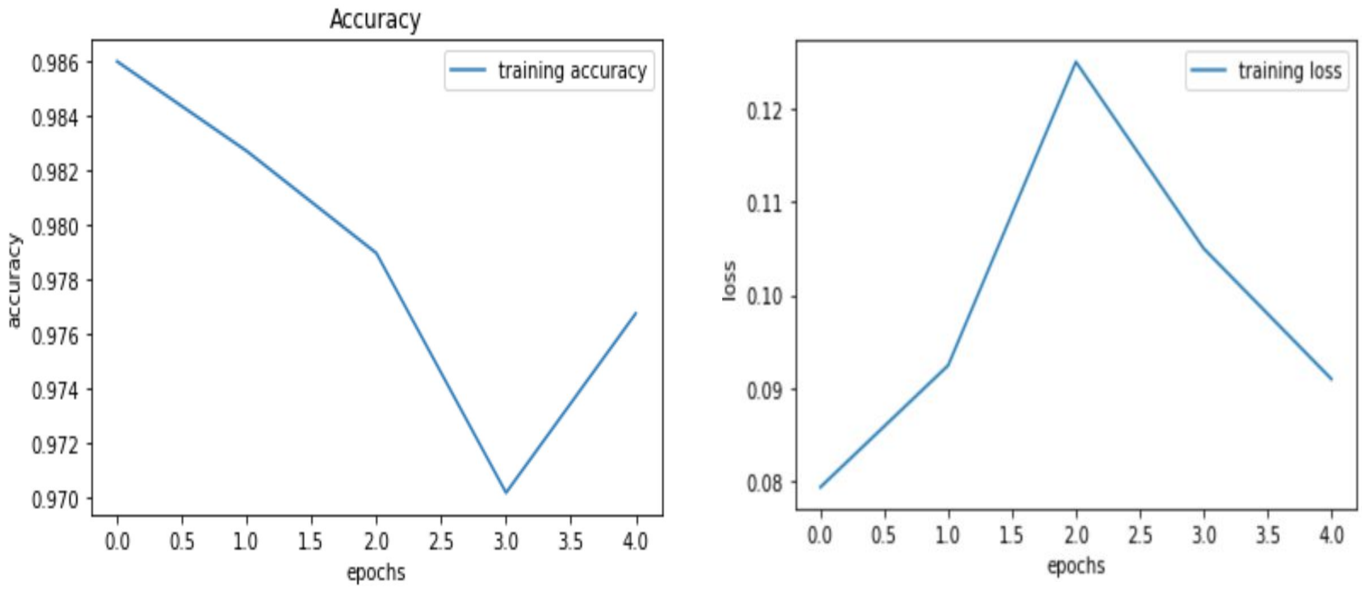
The accuracy of 96.69 has been achieved for the deep learning model after 30 epochs.



The accuracy vs epoch and loss vs epoch graphs for the user A during the first iteration



The accuracy vs epoch and loss vs epoch grapes of user A during the third iteration



The accuracy vs epoch and loss vs epoch grapes of user A during the sixth iteration

The accuracy_score metric has been used for the evaluation of the model.
We can conclude by using plots of accuracy vs epochs and loss vs epochs during iterations of federation learning
Let's take a single user and plot the above said graphs.
In this last iteration,there are some fluctuations but if we clearly observe the Y-axis, we can see that the scaling is very small and variation is also very small.
The final accuracies for the deep learning model and federated learning model is as follows,
Deep learning-96.96
Federated learning-97.45

Here the FL model performed better than DL model, but in reality we see these type of results are seen very rare because the distribution of data may not be equal among the users(mean some traffic signs data is present with some users and some other users may not have this traffic signs data).Real world federated data held by users are mostly non independent and identically distributed. So we can conclude that, generally, FL gets less accuracy than that of DL models.

Future work:

We assumed that the users have the same amount of data and same quality of data which is not true in real life hence suitable methods for the federated average learning should be developed to work with the different distributions of data among the users. In real life situations, traffic signs can be covered with snowfall and dust so accuracy of these cases should also be increased. There may be some signs present which are not traffic signs so that should be taken into consideration.