Gossip Learning Inspired Group Federated Learning In Agricultural Sector

Project Seminar
-by Piriya Sai Swapnika
19CS30035

Introduction:

In classical machine learning, a central ML model is developed in a centralised setting using all available training data. However, in mobile computing, users expect quick responses, and the time it takes for a user device to communicate with a central server may be too long for a satisfactory user experience. To overcome this, the model might be installed on the end user device, but continuous learning would be difficult because models are trained on a whole dataset, which the end user device does not have access to.

Federated Learning, and Gossip learning are two good concepts which help in overcoming this issue, but they have their own limitations. Hence, we have proposed a new concept combining the concepts of Gossip learning and Federated Learning, and have also discussed about its benefits.

Understanding CNN

Convolutional neural networks are a type of deep neural network that is commonly used to evaluate visual imagery. It employs a technique known as Convolution. Convolution means taking two functions and resulting in a third function that shows how the shape of one is modified by the other. In the end, the ConvNet's job is to compress the images into a format that is easier to handle while preserving elements that are important for making a decent prediction.

CNN’s are made of multiple layers of artificial neurons. When we feed an image into a Convolutional Neural Network, each layer generates a set of activation functions that are sent along to the next layer.

In the first layer, basic features such as horizontal or diagonal edges are usually identified. This is then passed on to the next layer, which detects more complicated features like corners and combinational edges. As we go deeper into the network, the model gets trained enough to recognise even more complex elements like objects, faces, and so on.

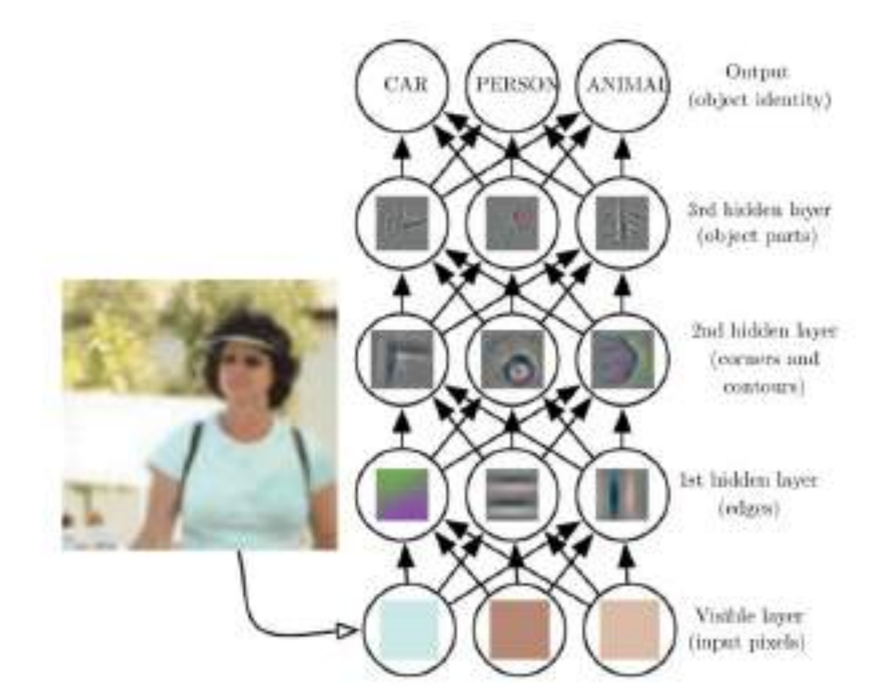
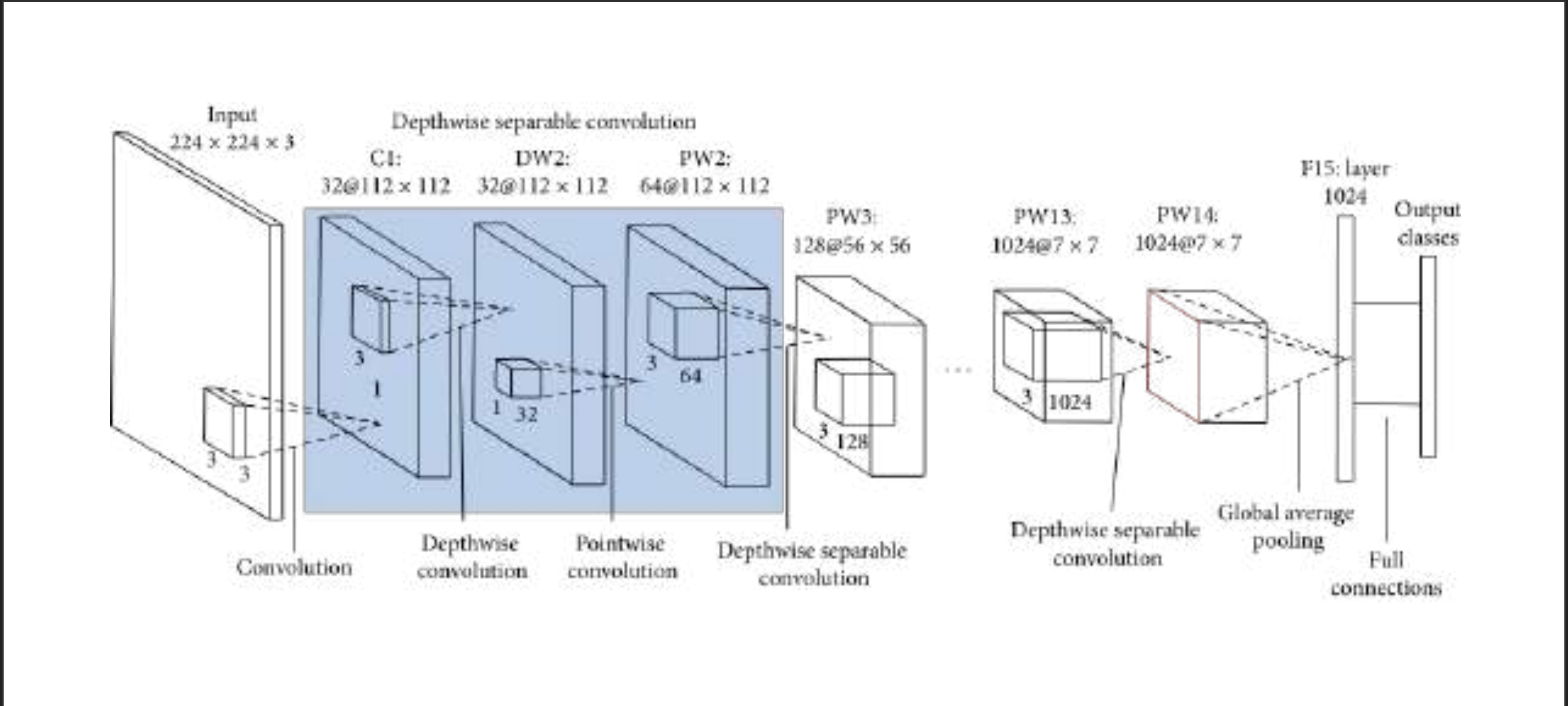

CAR
PERSON
ANIMAL
Output (object identity)
3rd hidden layer (object parts)
2nd hidden layer (corners and contours)
1st hidden layer (edges)
Visible layer (input pixels)

Image Source: https://harishnarayanan.org/writing/artistic-style-transfer/

Understanding MobileNet

MobileNet has less parameters being a lightweight deep neural network, and is more accurate at categorization. To reduce the amount of network parameters and increase classification accuracy, dense blocks are integrated into MobileNet. The new network topology can make full use of the output feature maps generated by the previous convolution layers in dense blocks, allowing for the generation of a large number of feature maps with fewer convolution cores while also allowing the features to be used repeatedly. The network decreases the parameters and calculation cost by setting a low growth rate.



Architecture of MobileNet

Image Source: https://medium.com/analytics-vidhya/image-classification-with-mobilenet-cc6fbb2cd470

Federated Learning

In Machine Learning, we often train data that is pooled from numerous edge devices such as mobile phones, laptops, and other computers and then centralised. Machine Learning systems then use this data and train themselves, eventually predicting outcomes for new data. Data privacy is addressed via the Federated Learning technique. The following is a step-by-step description of FL:

1. Each device will have a local copy of the centralised machine learning programme, which users can utilise as needed.

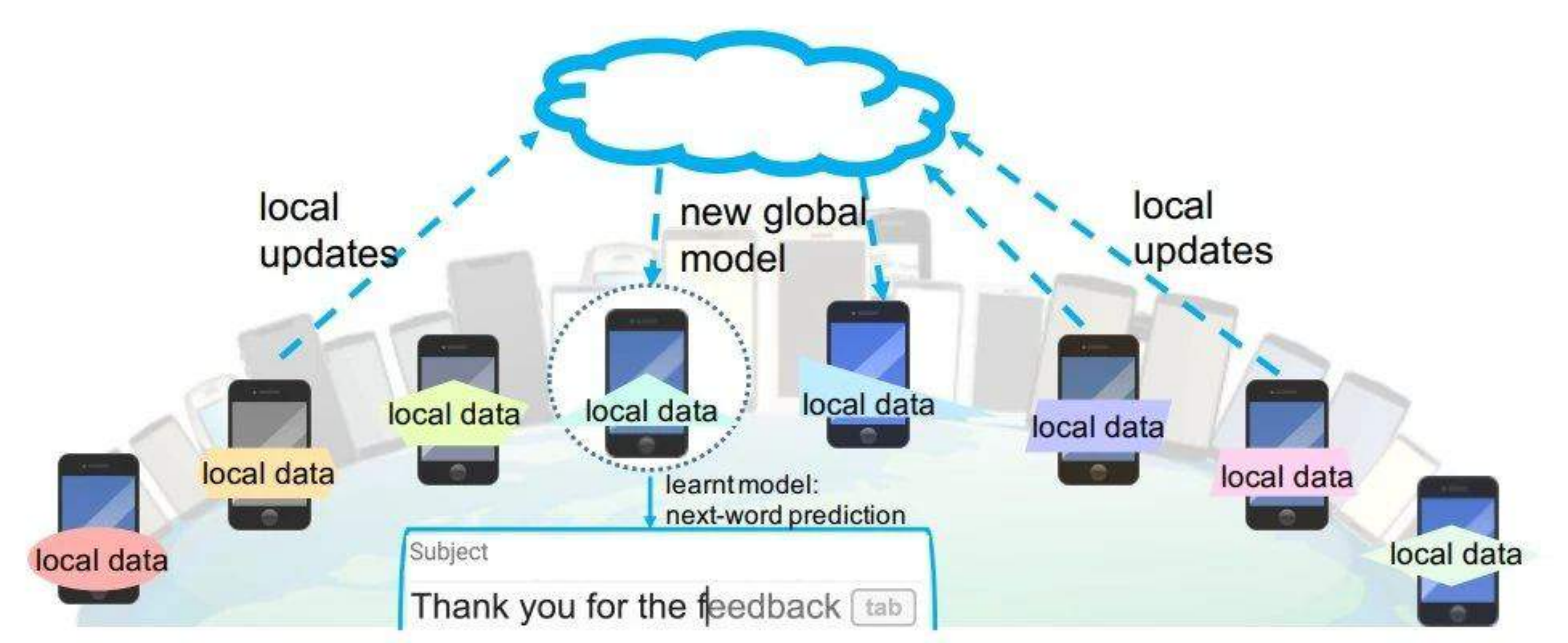
2. The model will now learn and train itself based on the information provided by the user, gradually becoming smarter over time.

3. The devices are then given permission to send the training results from the machine learning app's local copy back to the central server.

4. The same thing happens on several devices that have a local copy of the app. The findings will be compiled in a centralised server, but without any user data this time.

5. The centralised cloud server now uses the pooled training data to update its core machine learning model, which is now significantly better than the prior version.

6. The development team now upgrades the model to a newer version, and users update the programme to use the better model, which was generated using their own data!



Federated Learning Example Application for next-word prediction on mobile phones

Image Source: https://arxiv.org/pdf/1908.07873.pdf

Benefits of Federated Learning

- Data security: Since the data for training dataset is with the devices itself, the need for a data pool for the model is eliminated.
- Data diversity: In the event of network failure, FL allows access to heterogeneous data, even if data sources cannot communicate at all instances
- Continuous learning: Models are constantly upgraded utilising node data, with no need to collect the data.
- Hardware Complexity: Because federated learning models do not require a single complex central server to interpret data, this approach requires less complex hardware.

Drawbacks of Federated Learning

1. Cost Factor: FL models may require frequent communication between nodes and servers. So high storage capacity and high bandwidth are highly required.
2. Data Privacy:In FL, data is not collected on a single entity; instead, it is collected and analysed by numerous devices. This broadens the attack area.Even though only models, not raw data, are sent to the central server, models could be reverse engineered to reveal client information.
3. Performance limitations: Data heterogeneity: In federated learning, models from various devices are combined to create a superior model. Device-specific factors may hinder the generalisation of models from some devices, lowering the accuracy of the model's next version.Researchers have examined scenarios in which one of the federation's members could intentionally attack others by introducing hidden backdoors into the joint global model. Federated learning is a relatively new method of machine learning. To improve its performance, new studies and research are required. Model updates are lost in part or whole owing to node failures affecting the global model. There are many cases, where on the client side, there are no annotations or labels.

Centralized MobileNet vs Federated Learning

Upon running the MobileNet model on MNIST using tensorflow, the results and specificalities were as follows:

The accuracy on Test Dataset is : **96.50999903678894%**

and the loss is :   **0.112504125**

The pre-run FL model results after 100 communication rounds are as follows:

The accuracy on Test Dataset is : **96.50%** and the global loss is : **1.5140432**

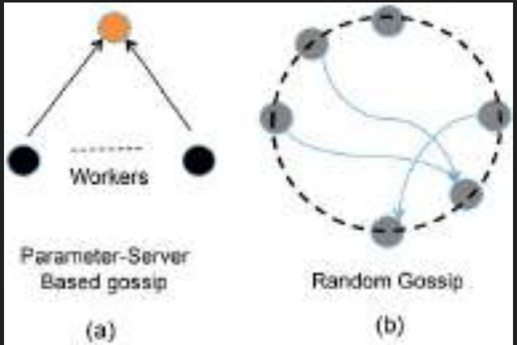
We can observe that the model efficiency of FL model is as good as that of MobileNet with added advantages of Data Security and real time learning.

Gossip Learning

Gossip Learning is a non-centralized method for learning models from fully distributed data. The node first creates a local model. This is then sent to another node in the network on a regular basis. A so-called sample service helps with node selection. The node merges a model with its local model and updates it using the local data set when it receives one. Averaging the model parameters is a common method for merging. The receiving model simply overwrites the local model in the simplest scenario. As a result of this approach, the models take random walks over the network and are updated when they reach a node. The update mechanisms are the same as they are for FL, and compression can also be used.

Advantages of Gossip Learning:

- Scalability: The benefits of GL are self-evident. Gossip learning has a far lower cost of scalability and a much higher level of robustness, because no infrastructure is required and there is no single point of failure,
- Data Security: Gossip Learning fully utilises node-to-node bandwidth and lowers communication with the central node, resulting in a more secure system.
- Faster Results: Gossip Learning makes full use of node-to-node bandwidth and reduces interaction with the central hub, leading to a more efficient system.



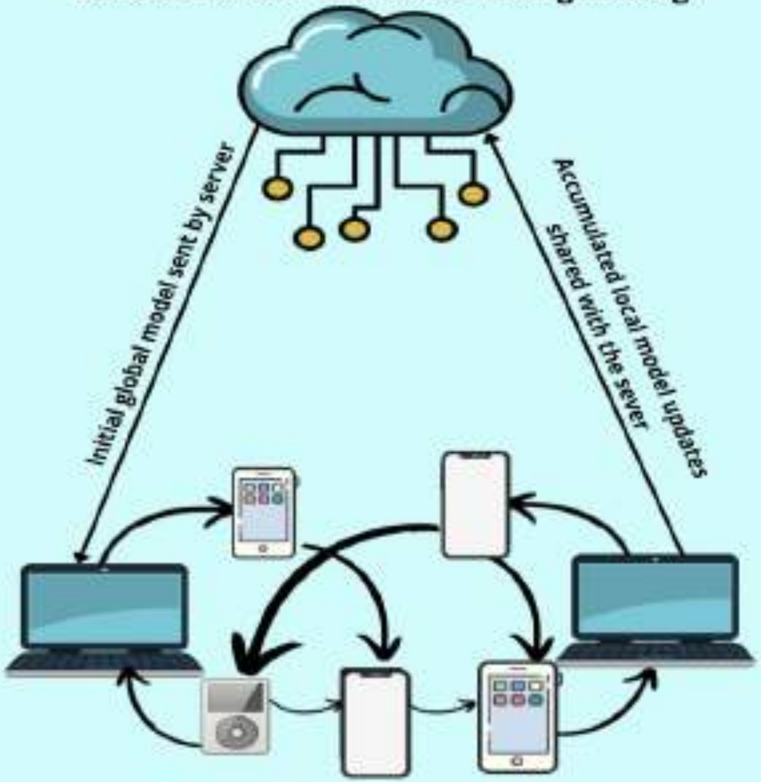
Proposed Integrated Model:

The Proposed Project aims to combine the benefits of Gossip Learning and Federated Learning. We distribute a data-set among several data-nodes and instead of following the vertical Federated Learning Model, we will use a horizontal Gossip Learning framework instead but with a Federated Learning modification in order to create a central global model.

Algorithm:

1. The process starts with a node receiving an initial model from the central server.
2. That node updates the model with its local data, trains it, and passes the model forward to another node of its choice following pass-through-gossiping.
3. The second node repeats the same to send the freshly updated model to another fresh node.
4. When all the nodes have been covered, the model, which has been updated by all the existing nodes, is passed back to the server.
5. Then the next iteration starts.
6. Data in Agricultural Nodes is often continuous so each iteration of gossip learning will use new groups of data that wasn’t used earlier. This is best explained through a diagram.
7. After every few iterations, the Central Server computes a new Global model using FedAvg.

Central Server computes a new global model per iteration of communicationusing FedAvg


Initial global model sent by server
Accumulated local model updates shared with the sever

Sharing of Local Model Updates among local nodes

Benefits of the proposed Algorithm

With the help of a federated learning inspired gossip learning framework, we can arrive at a global convergable model while accounting for:

- Low bandwidth requirement: In the case of FL, there is a lot of communication between the nodes and the server, which necessitates a lot of bandwidth. However, because communication in our suggested paradigm occurs mostly between nodes, this disadvantage is mitigated.
- Better Data Security: In some circumstances, models have been reverse engineered in order to steal client data while conducting FL. It's difficult to accomplish the same with our suggested paradigm because there's limited node-server connection, and it's almost impossible to figure out which data has been posted by which node.
- Less Expenditure: Communication between nodes is typically much less expensive with a message broker system than with a central server.
- Communication speed: This model uses less bandwidth and spends less money, and it also works faster because it communicates with the server less.
- Network Topology: In many real-world circumstances, the devices may be constrained to communicating with a small, fixed number of neighbours due to an underlying restricted topology. Pass-through gossiping is a strategy that is used in Gossip Learning. This method entails using hubs to act as "bridges" between low-degree nodes, allowing them to gossip with one another without revealing the network's power-law structure. The same principle can be used in our model.As a result, if the sender has a lesser level than the recipient, the receiver can save the incoming model as its current model and transmit it later, skipping the usual update and merging steps.

Drawbacks of the Proposed Model

1. Fully Distributed Data Model:The completely distributed data model, in which each device is supposed to own a single, private data point, is the first limitation.This may be true in some cases, such as with some recommender systems; nevertheless, there are numerous scenarios in which a single user has multiple useful data points, such as text completion and image classification.

2. Network Connectivity:The second drawback stems from the usage of gossip communications to disseminate models over the whole network. This approach's robustness and efficiency are predicated on the assumption that each node may choose its peer evenly at random from the whole network at each cycle.

This is usually accomplished by employing a peer sampling service, which distributes a uniform random selection of network members to each node. Unfortunately, this still necessitates that any device be able to interact with any other protocol member. Due to the security or privacy concerns, this may be impossible in some applications. In some circumstances, it may be ineffective and potentially costly.

3. Communication Speeds:The third and last constraint is the device communication speeds. Previous studies assumed that speeds were regularly distributed, resulting in a scenario in which the majority of speeds were clustered around the mean, with just a few outliers. The speed distribution may be substantially more heterogeneous in many real-world situations where various types of devices must interact.

Conclusion

The applicability of an integrated Gossip Learning inspired Federated Learning model to real-world circumstances was investigated in this paper. Federated Learning and Gossip Learning have restrictions that limit their use greatly. Each of these was examined to determine their impact, and where possible, new extensions to broaden the protocol's applicability were offered.

We illustrate how our model can be expanded to evaluate and learn from decentralised data, overcoming issues like data privacy, over-spending, and time limitations. However, it appears that state-of-the-art gossip learning based on the Federated Learning model has flaws that limit its application in real-world situations. With more research and testing, it could be a viable answer to one of the most pressing issues in practically all industries.

thank you