**Learning-based Incentive Mechanism for Federated Learning**

**Summary**

In this paper the authors find an incentive-based mechanism for federated learning that motivates edge nodes to contribute to the model training. The authors have made a Deep Reinforcement Learning-based incentive mechanism to determine the optimal strategy.

A major problem faced by FL is to incentivize people to join FL by contributing their computation power and data. For this a solution can, be to reward the participants according to their contributions. This solution though has some difficulties and is unfit in FL, one of the reasons for this solution being unfit in FL is that the relationship between the model accuracy and the amount of training data is nonlinear. The model accuracy depends on the model complexity and data quality and cannot be predicted in advance. Without the accurate predictions the previous used incentive mechanisms could not correctively reward participants, leading to financial loss or low participation rate.

The authors have proposed a new incentive mechanism that integrates mode updation using fresh data for federated learning in IoT applications which usually includes a parameter server which resides in the cloud and some edge nodes which is in charge of some IoT device. The parameter server aims to minimize the total reward, while each edge node maximizes the revenue which is the difference of the reward received from the parameter server and the cost of data collection, model training.

Since the authors are working with an edge-based Federated Learning, we must know what it is. Edge-based federated learning is a privacy-preserving machine learning technique that enables the edge nodes to train a shared global model without the need of uploading private local data to a central server. For each node participate in the model training, it collects data from the IoT devices and trains a model. The training cost of the edge nodes has two parts the computational cost and the communication cost (amount of data used for training).

To derive this new incentive-based mechanism the authors have used the concepts of Stackelberg game and then have derived the Nash Equilibrium. Nash equilibrium id derived of the participants’ decisions and the parameter servers’ contributions to the training accuracy. They have further designed an algorithm so that the parameter server and the edge nodes can adjust to optimize their interests without the knowledge of their participants decisions.

The author has chosen to use the Stackelberg Game concepts which tells that in the first stage, the parameter server will announce a total reward, followed by a second stage where each user will determine its training strategy to maximize its own utility. The strategy of the parameter server will be the reward and that of each node will be the amount of contributed training data. A Nash equilibrium is then taken out, it will exist if the player set is finite, the strategy sets are closed, bounded and the utility functions are continuous. A unique Stackelberg equilibrium will exist if the reward is the maximizing factor of the parameter server utility.

The author has then proposed a DRL based incentive mechanism without any prior information. For this they have introduced a basic learning mechanism of applying DRL into the decentralized incentive mechanism design problem. In this the DRL learns a general action decision from past experience based in the current state and the reward. The edge nodes interact with each other to determine the optimal participation level strategies after the action is taken placed. Since this mechanism is proposed without any prior information thus the edge nodes do not know any information about the other nodes so the find the optimal strategy each edge node learns the Nash Equilibrium and then update the model based on their local data and upload the updated model to the parameter server.

The authors have also designed a DRL approach to determine the optimal strategies for the parameter server and the edge nodes. The state space of the parameter server tells us that the parameter server trains a ML model and that the server can only observe the past strategies of edge nodes. The state space of the edge nodes tells us that the training period of each edge node. Due to the incomplete information criteria the nodes have to learn their optimal training strategies in a simulated environment. Further the parameter sever agents will take an action to incentivize the edge nodes when they receive a state. The edge nodes determine the training strategies while receiving the state. Each edge node continuously learns until they reach the Nash Equilibrium under the current parameter server’s strategy. These were the model parameters that were provided to the nodes and the server.

The training methodology includes that for the parameter server, it will maintain a policy or the actor network and an estimation of the value function or the critic network. The training methodology is based on actor-critic model. Similarly for the edge nodes, it will have a policy and an estimate of the value function. When all the edge nodes will receive the payment issued by the server, each node will then take actions. The edge node agents will the update both the policy and the value function based on the returns of each action. After updation ie. Receiving the optimal training strategy of each node the parameter server will also update its policy and the value function. Once this actor-critic model is well trained the parameter server and the edge nodes will determine their own strategies based on the output of their actor networks.

The authors have proposed that the incentive mechanism can motivate the edge nodes to participate in the federated learning training. The DRL-based incentive mechanism as used by the authors can learn the optimal strategies for the parameter server and edge nodes. On applying this mechanism, the authors were able to observe that the parameter server decreases its payment as the training cost increases. If the training cost is less then the server will be able to incentivize each node better. We also observed that the participation level of each node decreases as the training cost increases. Another observation that is made is that when the parameter server increases its payment to incentivize mode edge nodes, it leads to competition between the nodes so for that a solution could be that each edge node receives less reward from the parameter server.