**3GPP TSG-SA5 Meeting #155 *S5-242912***

Jeju, Korea, 27 - 31 May 2024

**Source: Intel, ZTE, Nokia, China Mobile, NEC**

**Title: pCR 28.9xy Add use case for management of Federated Learning**

**Document for: Approval**

**Agenda Item: 6.19.1**

# 1 Decision/action requested

***The group is asked to discuss and approve.***

# 2 References

None.

# 3 Rationale

When FL is used in 5GS, such as by NWDAFs, an ML model is collaboratively trained by a group of ML training functions including one acting as FL server and the others acting as FL clients. The ML training functions involved in FL need to be managed considering their roles.

This contribution is to add the use case and potential requirements for management of Federated Learning in 5GS.

# 4 Detailed proposal

It proposes to make the following changes to TR 28.9xy or AI/ML management phase 2.

|  |
| --- |
| **1st modified section** |

## 5.x Management Capabilities for ML training

### 5.x.1 Management of Federated Learning

#### 5.x.1.1 Description

Federated Learning (FL) is a distributed machine learning approach that allows multiple ML training functions to collaboratively train an ML model on local datasets contained in each ML training function without explicitly exchanging data samples.

FL is supported by a group of ML training functions, which contains an ML training function acting as FL server and multiple ML training functions acting as FL clients. The FL client keeps the data localized and private, and trains the ML model directly on the local nodes (client) where the data is generated or stored.

Federated learning can be categorized into two main types: Horizontal federated learning (HFL) and Vertical federated learning (VFL), based on the nature of the data distribution and the way the model training is orchestrated among participants. These categories reflect different data scenarios and use cases.

* Horizontal Federated Learning, the local data set in different FL clients have the same feature space for different samples (e.g. UE IDs).
* Vertical Federated Learning, the local data set in different FL clients have the same sample space for different features (e.g. PMs, KPIs).

For HFL, the process is as follows:

* Client Discovery and Selection: The FL Server discovers and selects FL Clients in an FL process;
* FL Initialization: The FL server initiates the federated learning process and distributing an initial global model to the FL clients for local training;
* ML model distribution and aggregation: FL clients train the ML model locally, and send the interim local ML model to FL server. FL server receives interim ML models from the FL clients, aggregates these interim ML models to update the global ML model, and then distributes the updated global ML model back to the FL clients. This step is repeated for a number of iterations, until the global ML model meets the training requirements.
* Stop: The FL server coordinates with FL clients to stop the FL process.



* Figure 5.x.1.1-1: ML model distribution and aggregation for HFL

In 5G system, the deployment options for FL are shown in Table 5.x.1.1.

**Deployment scenario 1**:

In 3GPP management domain, federated learning function can be located in the RAN domain management function. The RAN domain management function could act as a FL server, where gNB can act as a FL client. In this case, for managing the FL training, needs allow the RAN domain MnS consumer to request the FL training, control the producer-initiated FL training, and manage its process.

****

**Figure 5.x.1.1-2: Federated Learning (FL) is located in RAN management function**

**Deployment scenario 3**:

In 3GPP management domain, federated learning function can be located in the CN domain management function. The CN domain management function (e.g. NWDAF) could act as a FL server and FL client.



Table 5.x.1.1-1

|  |  |  |
| --- | --- | --- |
|  | FL server | FL client(s) |
| Case 1 | NWDAF | NWDAF(s) |

NOTE: A prior agreement needs to exist between the FL server and the FL clients to exchange ML models.

#### 5.x.1.2 Use cases

##### 5.x.1.2.1 Management of different roles in Federated Learning

For FL, an ML model is collaboratively trained by a group of ML training functions (e.g., MTLF in NWDAFs) including one acting as FL server and the others acting as FL clients. Federated Learning training allows multiple ML training functions to collaboratively train an ML model on local datasets, it means that the local training of each FL client needs to start and complete almost at the same time to ensure the performance of ML model aggregation performing in FL server.

For managing the FL, the ML training MnS consumer needs to know the ML training functions involved in the FL, and the role (FL server, FL client) of each ML training function, so that the consumer understands the impact of ML training function and can manage it correspondingly.

When receiving an ML Training request, the ML Training MnS Producer should evaluate whether FL process needs to be started according to the training requirements (e.g. the number of data samples) provided by the ML training consumer. Based on the received requirements, the ML training function may select appropriate FL Clients.

For managing the FL collaborative relationship, the FL MnS producer needs allow consumer to request FL with configuring the FL availability time requirement (e.g. time duration for the FL process) for selecting the FL Client which is available in the required time for training ML local Model.

To evaluate the performance of FL, the consumer needs to know the performance of the final global ML model on the participating FL clients. For instance, if an FL server cannot generate a global ML model with satisfied performance for the FL clients, the consumer may interact with the MnS ML training producer to optimize the FL for future training, e.g., updating the criteria for selecting FL clients.

#### 5.x.1.3 Potential requirements

**REQ-FL\_MGMT-1:** The ML training MnS producer should have a capability allowing an authorized consumer to get the FL role (FL server or FL client) of an ML Training Function in Federated Learning.

**REQ-FL\_MGMT-3:** The ML training MnS producer should have a capability allowing an authorized consumer to provide FL training requirements to the ML Training Function acting as FL server.

**REQ-FL\_MGMT-4:** The ML training MnS producer should have a capability allowing an authorized consumer to provide requirements for selecting FL clients in Federated Learning to the ML Training Function acting as FL server.

**REQ-FL\_MGMT-02:** The ML training MnS producer should have a capability report the FL status to MnS consumer.

**REQ-FL\_MGMT-5:** The ML training MnS producer should have a capability allowing an authorized consumer to get the performance about final global ML model on each participating FL client.

|  |
| --- |
| **End of modified sections** |